NEC

NEC Scalable Technology File System (ScaTeFS)

Administrator's Guide

SX-Aurora TSUBASA

Proprietary Notice

The information disclosed in this document is the property of NEC Corporation (NEC) and/or its licensors. NEC and/or its licensors, as appropriate, reserve all patent, copyright, and other proprietary rights to this document, including all design, manufacturing, reproduction, use and sales rights thereto, except to the extent said rights are expressly granted to others.

The information in this document is subject to change at any time, without notice.

Linux is a registered trademark of Linus Torvalds in the United States and other countries.
Red Hat and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries.
EXPRESSCLUSTER X is a registered trademark of NEC Corporation.
Windows are registered trademarks of Microsoft Corporation in the United States and other countries.
NEC Storage PathManager is a registered trademark of NEC in Japan.
InfiniBand is a trademark or service mark of InfiniBand Trade Association.
Mellanox is trademark or registered trademark of Mellanox Technologies in Israel and other countries.
Docker is a trademark of Docker, Inc. in the U.S. and/or other countries.
All other product, brand, or trade names used in this publication are the trademarks or registered trademarks of their respective trademark owners.

© NEC Corporation 2018-2024

Preface

This document explains how to setup NEC Scalable Technology File System, operation, and how to optimize IO, etc.

How to use this document

This document consists following chapters.

Target readers are different every chapter, indicated to right column of table.

Chapter	Title	Contents	Target
1	Overview of NEC Scalable Technology File System	Explain ScaTeFS overview.	Administrator User
2	Network configuration	Explain how to setup network for ScaTeFS.	Administrator
3	Hardware configuration of IO server	Explain HW of IO server.	Administrator
4	Hardware configuration of client side	Explain HW of client.	Administrator
5	Configuring IO servers	Explain how to setup IO server	Administrator
6	Setting the Linux client	Explain how to setup Linux client. This chapter is included how to use ScaTeFS on SX-Aurora TSUBASA. And explain logs at trouble occurs.	Administrator
7	Setting the SX-ACE Client	Explain how to setup SX-ACE client.	Administrator
8	Setting to use ScaTeFS on a Docker’s container	Explain setting to use ScaTeFS on a Docker's container.	Administrator
9	Operation management	Explain function for operation, ex QUOTA.	Administrator User
10	Maintenance	Explain how to backup, check consistency, etc.	Administrator

Chapter	Title	Contents	Target
11	Configuration and instructions for end users	Explain TIPS for optimizing IO and how to use "VE direct IB Library" at SX-Aurora TSUBASA by understanding file management of ScaTeFS.	Administrator User
12	Specification	Explain specifications.	Administrator User

Related Documents

- SX-Aurora TSUBASA Installation Guide
- HPC Software License Management Guide
- SX-Aurora TSUBASA Fortran Compiler User's Guide
- NEC Network Queuing System V(NQSV) User's Guide [Reference]
- NEC Network Queuing System V(NQSV) User's Guide [Management]
- SX Cross Software Node-lock License Installation Guide (*)

Refer to the SX-Aurora TSUBASA documents from the following Web site:

https://sxauroratsubasa.sakura.ne.jp/documentation/

(*) On the SX-Aurora TSUBASA system, ScaTeFS does not use Node-lock license. Therefore, "SX Cross Software Node-lock License Installation Guide" is not referred on SX-Aurora TSUBASA environment.

Definitions and Abbreviations

Term	Description
ScaTeFS	Abbreviation for NEC Scalable Technology File System.
IO server	Servers of which ScaTeFS is comprised. At least 2 of them are required.
VE	An abbreviation for Vector Engine. VE is the hardware (NEC proprietary architecture – based on NEC SX architecture) on which applications are running.
VH	An abbreviation for Vector Host. VH is a commodity off-the-shelf server (Xeon x86-64) on which common operating systems are running. Currently the OS on the VH is a Linux server.
IO server v1 for standard model	An IO server consisting of 4 storages per 2 servers. Express product name: Express5800/R120e-2M
IO server v1 for small-scale model	An IO server consisting of 2 storages per 2 servers. Express product name: Express5800/R120e-2M
IO server v3 for standard model	An IO server consisting of 2 storages per 2 servers. Express product name: Express5800/R120g-2M
IO server v4 for standard model	An IO server consisting of 2 storages per 2 servers. Express product name: Express5800/R120h-2M
IO server v4+ for standard model	An IO server consisting of 2 storages per 2 servers. Express product name: Express5800/R120h-2M 2nd-Gen
IO server v4++ for standard model	An IO server consisting of 2 storages per 2 servers. Express product name: Express5800/R120h-2M 3rd-Gen
Root IO server	A type of IO server. The mkfs command is executed on this server, and clients are mounted on this server. During system operation, it processes data in the same way as the other IO servers.
IO server daemon	The daemon that runs on the IO server
Virtual file	A file created on the virtual file system. A regular file on ScaTeFS.
Real file	A fragment of a virtual file spanning multiple IO servers. It actually refers to a file on the real file system.
Virtual file system	This is a client-visible file system. The file system consists of multiple IO targets. It is the ScaTeFS itself.

Term	Description
Real file system, IO target	Basic units that make up the virtual file system. It is created under each IO server. It is an ordinary file system available to Linux.
Fair share I/O scheduling	This function distributes IO-server resources fairly to each user or node.
Storage group	This function assigns different media with different access speeds such as NL SAS and SSD to their respective directories of the same file system based on their purposes. For example, a certain directory consisting of SSD can be used as a high-speed temporary area. The other directories consisting of NL SAS are inexpensive and more suitable for storing large-scale files.
Premap	This function generates in advance on each real file system as real files as the number of files corresponding to the specified size. The purpose is to use premapping to reduce the overhead of generating real files in case of parallel 'write' operations onto a virtual file. It uses scatefs_premap(1).
Parallel I/O	To write and read a file by transferring data in parallel using multiple computing nodes. The main purpose is to increase the I/O efficiency for large-scale files.
TOE	An abbreviation for TCP Offload Engine. The complex TCP function implemented on the hardware reduces the load on the CPU.
NIC	An abbreviation for Network Interface Card. This hardware is for communicating with other nodes.
10GbE	An abbreviation for 10Gigabit Ethernet.
GbE	An abbreviation for Gigabit Ethernet.
IB	An abbreviation for InfiniBand.
HCA	An abbreviation for Host Channel Adapter. A hardware to communicate with other nodes using InfiniBand.
IPoIB	An abbreviation for IP over InfiniBand. IP protocol works on InfiniBand network.
Verbs	The native API of InfiniBand. Verbs enables faster communication than IPoIB.
bonding	A method which aggregates multiple NICs or HCA ports virtually for redundancy or load-balancing.
Ib-bonding	A function which provides bonding for IPoIB.

Term	Description
Subnet manager	A software which manages and controls IB subnet. IB switch vendor may provide subnet manager. Also OpenSM is available for subnet manager.
QoS	An abbreviation for Quality of Service. In this manual, QoS means the QoS function of IB network and subnet manager.
Virtual Lane	A method of providing independent data streams on the same physical link of IB.
Service Level	A value for assigning IB packet to virtual lane.
Standard model	SX-ACE (A cluster system consisting of more than 64 nodes.) or Linux(RHEL) clients.
Small-scale model	SX-ACE Lite (A cluster system consisting of 16 or 32 nodes.) or Linux(RHEL) clients.
ScaTeFS IB Library	The library which issues ScaTeFS IO by InfiniBand on user space for performance improvement.
ScaTeFS VE direct IB Library	The library which issues ScaTeFS IO by InfiniBand on user space of VE for performance improvement.
ScaTeFS InfiniBand high performance Library	The library which issues ScaTeFS IO by InfiniBand on user space for performance improvement. On scalar machine, it is ScaTeFS IB Library. On VE, it is ScaTeFS VE direct IB Library.
Control communication	A communication which is issued internally by ScaTeFS client on IPoIB. Control communication is used for establishing or disconnecting IB Verbs communication.
NUMA	An abbreviation for Non-Uniform Memory Access. A kind of the memory shared multi-processors system. The memory access time depends on the memory location relative to the processor.
DDN	DataDirect Networks
SFA7990XE	The storage appliance product provided by DDN.
VM	An abbreviation for Virtual Machine. Software and flamework for emutation of computers.

Contents

List of tables

List of figures

Chapter1　Overview of NEC Scalable Technology File System

1.1 Introduction to NEC Scalable Technology File System

The NEC Scalable Technology File System (ScaTeFS: pronounced as "Skéɪt F S") is a distributed and parallel file system that supports a large-scale HPC system and enables greater data capacity. To realize load balancing and scale-out, all requests of all basic functions as a file system (read/write operation, file/directory generation, etc.) can be distributed to multiple IO servers uniformly since ScaTeFS does not need a master server for managing the entire file system such as a metadata server. Therefore, the throughput of the entire system increases, and parallel I/O processing can be used for large files.

Meanwhile, migration to SX-ACE is possible without needing to change the program because ScaTeFS is compliant with POSIX. Also, the front-end machines and PC clusters can share the same file system in order to support a heterogeneous environment.

And for SX-Aurora TSUBASA, ScaTeFS InfiniBand high performance library is available. It realizes the optimized IO for the SX-Aurora TSUBASA architecture.

Additional IO servers and storage, which can be added during operation, and the failover in case of IO server failure improve the operational continuity. There is no need to build a large-scale FC-SAN environment because the system is based on the IB and 10GbE networks, thus reducing the system administration costs.

1.2 Basic components

ScaTeFS consists of 4 major components as shown in Figure 1-1　Conceptual diagram of ScaTeFS

- Client (computing node)
- Network
- IO server
- Storage

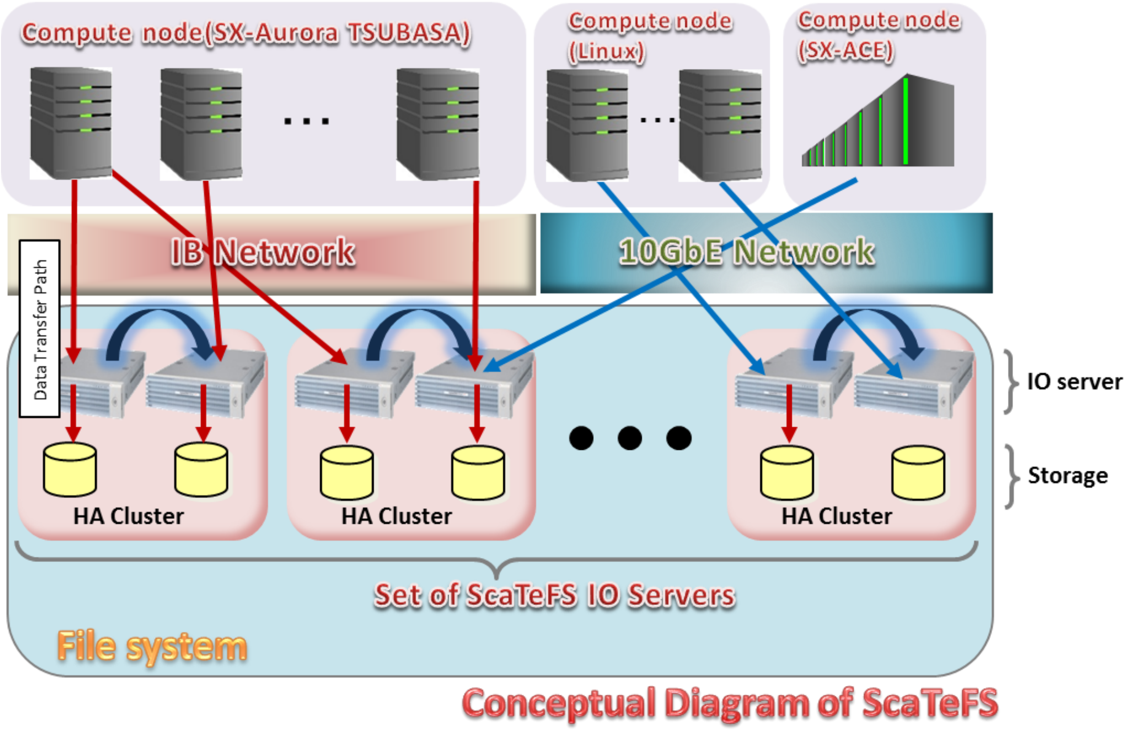


Figure 1-1　Conceptual diagram of ScaTeFS

1.2.1 Client

SX-Aurora TSUBASA, Linux machine such as computing node and front-end machine, and SX-ACE can be a client.

1.2.2 Network

On SX-Aurora TSUBASA, data are communicated between a client and IO servers via IB network. Only IB network can be used. 10GbE network cannot be used.
On Linux machine such as computing node and front-end machine, IB or 10GbE network can be used.

On SX-ACE only 10GbE network can be used.

1.2.3 IO server

Based on the client's request, each IO server operates metadata and fragments of file data stored in the storage devices connected under it. In addition to that, configuring an HA cluster by two IO servers is necessary in order to continue that operation even if one IO server in its cluster is broken.

1.2.4 Storage

It is connected to each IO server and stores metadata and fragments of file data.

1.3 Principal features of ScaTeFS

Principal features of ScaTeFS are as follows:

(1) Large-capacity and high-speed I/O functions

Load balancing with multiple IO servers ensures high throughput.

A large-scale file system in proportion to the number of IO servers can be created.

A function to add IO servers and storage without stopping the operation for better performance and higher capacity

A function to create large files

A function to update the same file simultaneously from multiple computing nodes by parallel I/O processing.

A function to cache data and metadata for efficient processing

Lossless communication by DCB (Data Center Bridging) with 10GbE

QoS function specifying the service level with IB

ScaTeFS IB Library supports the lightweight and high performance IO through a user space using InfiniBand. (See 9.12 ScaTeFS InfiniBand high performance library)

(2) Availability

A failover function of IO servers to preserve data by journaling in case of IO server failure

A path failover function in case of path failure between the IO server and the storage media

Addressing network interface failure of IO servers

(3) Easy configuration and operation of the system

Maintenance of IO servers can be performed without stopping the operation

Only one command is needed to build a file system spanning multiple IO servers

Consistency check of the file system and its recovery function

A function to collect logs and statistics

(4)　Supports various environments and usage

SX-Aurora TSUBASA, front-end machines, PC clusters and SX-ACE can use the same file system.

A fair share I/O scheduling enables fair I/O processing for a multi-user environment.

Storage groups enable a variety of storage use.

Flexibly supports a various models of SX-Aurora TSUBASA.

Flexibly supports a range of models, from small-scale models such as SX-ACE Lite to standard models such as SX-ACE.

High performance IO in user space using IB library. Using the NFS server on the Linux client, a file system can be exported to the NFS client.

Constructing a Samba server on the Linux client and making it publicly available, it allows access from Windows. (see Procedure for Accessing ScaTeFS from Windows)

Support SFA7990XE the DDN storage appliance as IO servers.
High throughput is achieved by taking advantage of the features of SFA7990XE with the ScaTeFS IB library.

Chapter2　Network configuration

2.1　Getting started

The network environment needs to be configured to use ScaTeFS. When configuring the network environment, it is important to consider the deployment of components such as the computing nodes, L3 switches, IO servers, and to consider IP addressing (rules for IP address assignment) and routing configuration. For example, fewer L3 switches results in easier management but reduced performance because communication with many entities must be handled by a small number of switches.

Therefore, deliberate consideration is required before configuration.
An example of network configuration using ScaTeFS is described below, for which the configuration of components and IP addressing are considered.

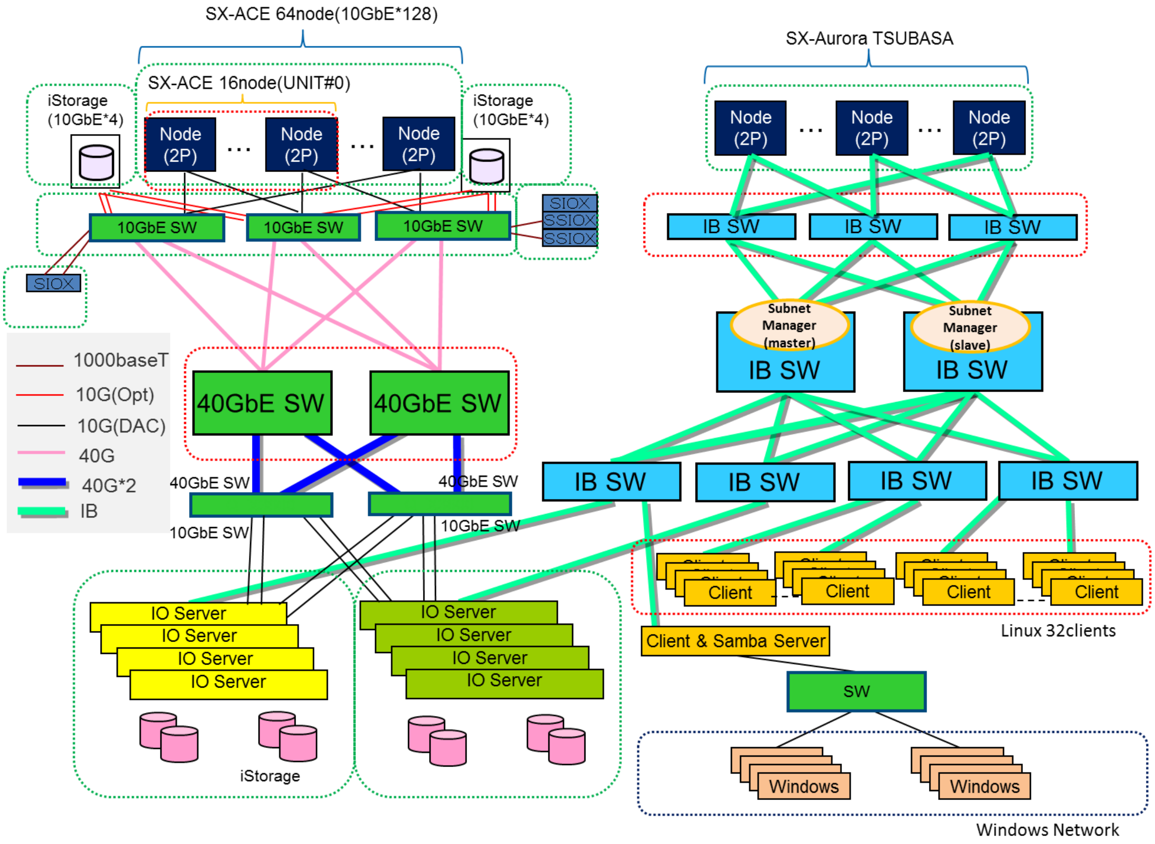


Figure 2-1　Network configuration example

2.2 Using InfiniBand

2.2.1 Network configuration

In InfiniBand network, ScaTeFS clients and IO servers must be placed on same subnet. On ScaTeFS clients and IO servers, an IPv4 address will be assigned to IPoIB. The number of IPv4 address which will be assigned to each machines is one. If the machine has two or more HCAs, ib-bonding is available.

2.2.2 Communication protocol

ScaTeFS uses two communication protocols, IB Verbs and IPoIB.

- IB Verbs protocol
  IB Verbs is a high speed communication protocol which uses kernel native APIs with HCA device name and port number. On ScaTeFS, IB Verbs will be used generally to access filesystem. The multipath function of ScaTeFS client enables redundancy communication.
- IPoIB protocol
  IPoIB uses IPv4 address and TCP port number for communication. On ScaTeFS, IPoIB will be used to establish IB Verbs connection (control communication).
  Using ib-bonding to IPoIB network interfaces enables redundancy communication.
  Nmtui command or nmcli command on Linux are used to configure ib-bonding.

On the runtime of an application, if “ScaTeFS IB Library” used, IOs are issued from the library by IB communication. The performance of the application which will issue the large amount of IO will be improved by the library. For more details about ScaTeFS IB Library, please refer 9.12 ScaTeFS InfiniBand high performance library.

2.2.3 Multi path function

On ScaTeFS clients, HCA device name and port number must be described in the configuration file. If two or more HCA devices are described in the configuration file, multiple HCAs will be used as Active/Active.
When a fault occurs on HCA device or route, remaining path will be used to continue communication. The path which cannot be used to communication becomes monitoring state. When the recovery of the path is detected, the path will be used again automatically. For more details about the configuration of the multi path function

and the configuration file, please refer 6.1.6 and 6.1.7.

2.2.4　QoS (Quality of Service)

Any service level can be specified to communications for metadata and IO. The service level is specified by mount option. Please refer 6.1.6 for details of mount option.

2.3　Using 10GbE

2.3.1　Example of configuration

In Table 2-1 Setting values for assigning address, IP addresses and VLAN-IDs are assigned based on the rules described below.

- IP address

  Use private addresses (class B) 172.16.0.0 through 172.31.255.255 for assigning addresses over the entire system. IP addresses are assigned according to the rules below.

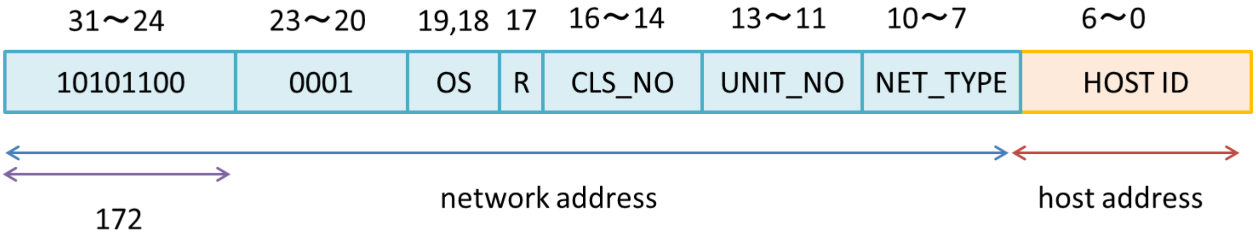


Figure 2-2　IP addressing example

Table 2-1　Setting values for assigning address

Setting value	**Description**
OS	Specify a 2-bit value according to the following: 00: The machine being assigned an IP address is an IO server 01: The machine being assigned an IP address is SUPER-UX 10: The machine being assigned an IP address is SVP 11: Machines other than the above
R	Reserved(MBZ), specify 0.
CLS_NO	Specify a 3-bit value according to the cluster number. Example: Cluster number 2(010) The cluster consists of 64 nodes.
UNIT_NO	Specify a 3-bit value according to the UNIT number Example: UNIT number 2(010) The UNIT consists of 16 nodes.
NET_TYPE	Four bits of NET_TYPE are as follows: 10 09 08 07 +--+--+--+--+ |type |IOC/P| +--+--+--+--+ Network Type/ IOC/ Port-No (4bit) Type: Specify a value according to the Ethernet. 00: Control Ethernet (SX-ACE) or IB (Linux-Client) 01: Operation Ethernet 10: TOE 11: iSCSI IOC/P: Specify a value according to the IOC and port 00: IOC=0,port=0 01: IOC=0,port=1 10: IOC=1,port=0 11: IOC=1,port=1
HOST_ID	Specify a 7-bit value according to the following: 0000000: Reserved 0000001 to 1000000: Assigned to Node(0) through Node(63). 1000001 to 1111101: Assigned to iSCSI target (NEC Storage). 1111110: Assigned to the gateway. 1111111: Reserved

・VLAN-ID

11,10	9～7	6～4	3～0
OS	CLS_NO	UNIT_NO	NET_TYPE

Figure 2-3　LAN-ID assignment example

For each setting value, see Table 2-1 Setting values for assigning address.

For a large-scale network environment like in the configuration example, it is recommended that you define rules to set the IP addresses and VLAN-IDs.
Refer to Chapter Chapter5, Chapter6 and Chapter7 for more information of configuring machines.

2.3.2　Routing table

(1)　Client

The network interfaces the client uses by default are selected according to the routing table, and therefore the routing table needs to be configured properly.

For example, assume the network interfaces for the client and the server are as follows:

Client

eth0:xx.xx.195.10
eth1:xx.xx.196.10

IO server

bond0:xx.xx.200.1
bond1:xx.xx.201.1
For bonding(bond0, bond1), see Chapter 3.1.6.

*) Default Ethernet interface names are enXXXXX on RHEL 7. In this case, interface names in this guide should be read as the actual interface name.

To establish a connection between eth0:xx.xx.195.10 and bond0:xx.xx.200.1, and between eth1:xx.xx.196.10 and bond1:xx.xx.201.1, routing table is the image below.

- Show the routing table by ip command:

```
# ip route
xx.xx.200.0/25   via   yy.yy.yy.yy   dev   eth0   proto   static   metric   NNN
xx.xx.201.0/25   via   zz.zz.zz.zz   dev   eth1   proto   static   metric   NNN
```

- Show the routing table by netstat command:

```
# netstat -r
Kernel IP routing table
Destination       Gateway           Genmask            Flags    MSS Window   irtt Iface
xx.xx.200.0       yy.yy.yy.yy       255.255.255.128    UG       0       0          0    eth0
xx.xx.201.0       zz.zz.zz.zz       255.255.255.128    UG       0       0          0    eth1
```

(2)　IO Server

Due to the restriction that Linux cannot be connected over the subnet externally if the outward and return routes are different, iproute2 needs to be used when connecting via multiple network interfaces.

The iproute2 is a routing control package that provides a function to ensure that the outward and return routes are the same.

Use iproute2 to configure the IO server routing.

For an example of configuring the routing, see 5.1.18.3.

2.3.3　DCB

Data Center Bridging can prioritize each traffic type (data and metadata).

ScaTeFS minimizes the delay of metadata transfer by using DCB to set Priority on each traffic type and using a destination port per traffic type.

Specify the destination port number for metadata for cport, and specify the destination port number for data for cdport in /etc/scatefs/system.info. If cdport is not set, the number specified for cport is used for both port numbers to establish connections.

You can use this function on only 10GbE network. When you use only IB network, you cannot use it.

2.3.4　Setting the priority

Each port can be assigned priority 0 (low) through 6 (high). Ports used for ScaTeFS are assigned priorities as follows:

Table 2-2　Ports and Priority

Port	Priority
Port for meta data	6
Port for data	4
Port between IO servers	5

For details about the priority settings, see 5.1.18.4.

Chapter3　Hardware configuration of IO server

3.1　HA cluster configuration

IO server consists of Active-Active HA cluster configuration.

The configurations for each models of IO servers are described below:

3.1.1　IO server v1 for standard model and small-scale model

IO server is connected to a client with two 2port 10GbE HBAs, and connected to four storage units with two 2port 8G-FC HBAs.

In small-scale models, an IO server is connected to a client with one 2port 10GbE HBA and is directly connected to two storage units without using an FC-Switch.

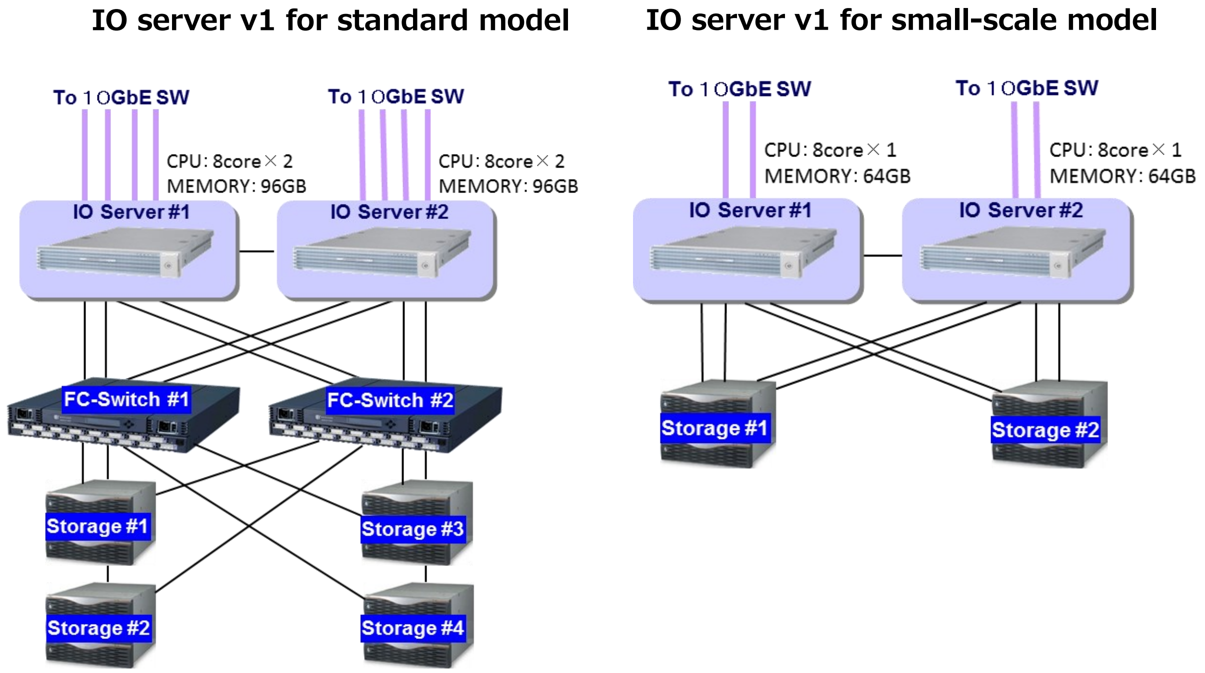


Figure 3-1　Example of Configuration at IOSv1

3.1.2　IO server v3 for standard model

IO server for standard-scale model is connected to a client with two 2port 10GbE HBAs, and is directly connected to two storage units with two 2port 16G-FC HBAs without using an FC-Switch. IO server can be connected to a Linux client with IB network by installing IB HCA as option. IO server is connected to a client with one port even if the HCA has two ports.

IO server v3 for standard model

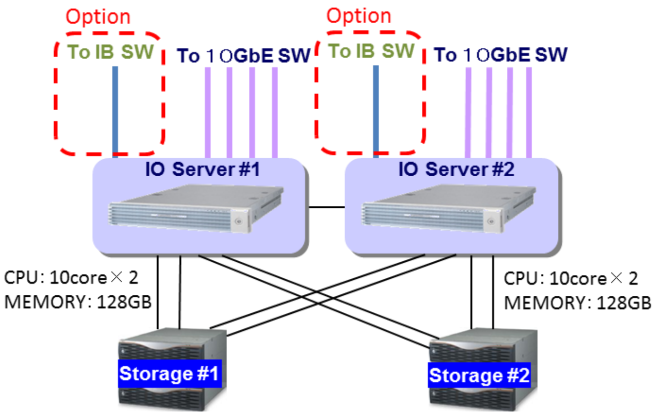


Figure 3-2　Example of Configuration at IOSv3

3.1.3　IO server v4 for standard model

IO server for standard-scale model is directly connected to two storage units with two 2port 16G-FC/2port 32G-FC HBAs. The IO server is connected to a client with one or two IB HCAs. HCA can be selected EDR 1port or 2port. IO server can be connected to a client installed 10GbE HBA and SX-ACE with 10GbE network by installing 10GbE HBA as option.

IO server v4 for standard model

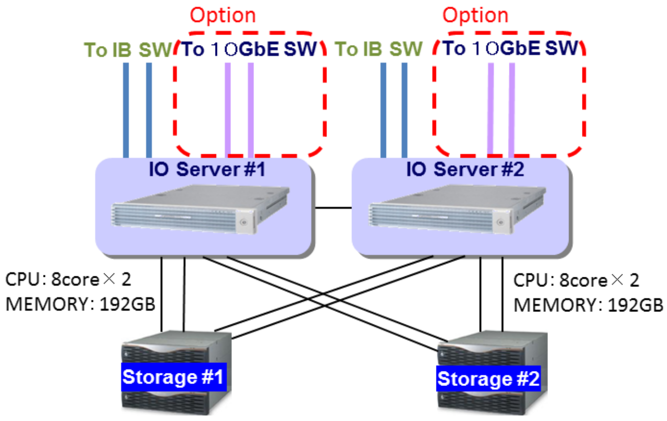


Figure 3-3　Example of Configuration at IOSv4

3.1.4　IO server v4+ for standard model

IO server for standard-scale model is directly connected to two storage units with two 2port 16G-FC/2port 32G-FC HBAs or two SAS HBAs. The IO server is connected to a client with one or two IB HCAs. HCA can be selected EDR/HDR100 1port or 2port. IO server can be connected to a client installed 10GbE HBA and SX-ACE with 10GbE network by installing 10GbE HBA as option.

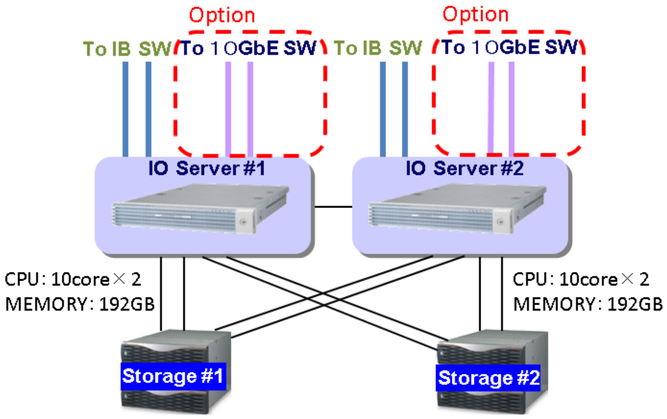


Figure 3-4　Example of Configuration at IOSv4+

3.1.5　IO server v4++ for standard model

IO server for standard-scale model is directly connected to two storage units with two 2port 16G-FC/2port 32G-FC HBAs or two SAS HBAs. The IO server is connected to a client with one or two IB HCAs. HCA can be selected HDR100 1port or 2port. IO server can be connected to a client installed 10GbE HBA and SX-ACE with 10GbE network by installing 10GbE HBA as option.

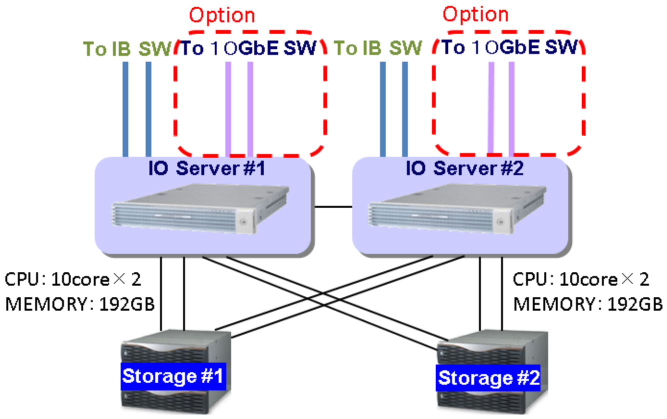


Figure 3-5　Example of Configuration at IOSv4++

3.1.6　SFA7990XE

SFA7990XE storage appliance is directly connected to SS9012 disk enclosure units with SAS intereface. The IO server is connected to a client with two IB HCAs. HCA can be selected HDR100. ScaTeFS IO server service runs on the VMs on the two controllers of the SFA7990XE.

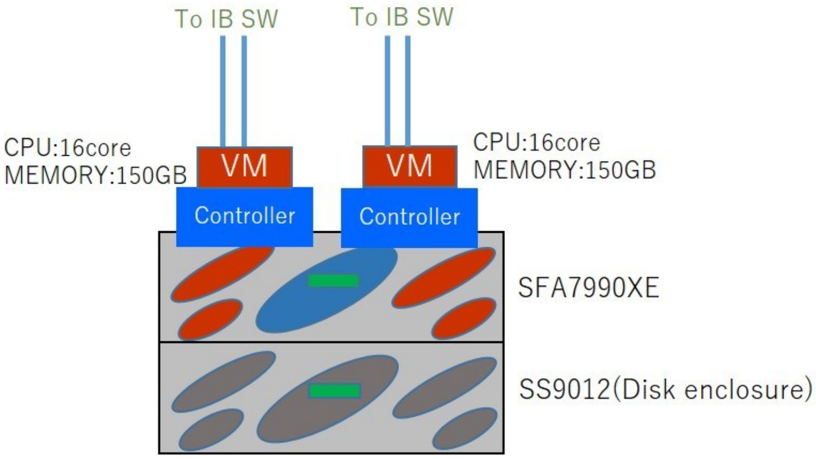


Figure 3-6　Example of Configuration at SFA7990XE and SS9012

3.2 Bonding of 10GbE

Bonding means to equip a machine with multiple NICs and Ethernet ports and operate it as a virtual network interface. Configuring bonding on IO servers is recommended in order to improve load balancing, bandwidth availability, and failure resistance.

Although several modes are available for bonding, 802.3ad (LACP, dynamic link aggregation) is required for an environment supporting both Linux and SX-ACE, and the L3 switches must support 802.3ad as well. For bonding, target NICs must be connected under the same L3 switch.

For an example of configuring the bonding, see 5.1.18.

Chapter4　Hardware configuration of client side

4.1　Specifications for Linux machines（SX-Aurora TSUBASA）

A Linux machine used as a client must support the x86-64 architecture. Other architectures such as x86 cannot be used.

When a client is connected to IO servers with IB network, a client machine needs to be installed an IB HBA. The IB HCA which ScaTeFS supports is ConnectX-4 and ConnectX-6 of NVIDIA. ConnectX-6 is supported on RHEL/CentOS 7.6 or later.

4.2　SX-ACE

The SX-ACE network interface uses 10GbE-NICs.

Two 10GbE-NICs at most are loaded on board, and the following four virtual interfaces are used for each physical port:

- The control system network (GbE equivalent)
- The operational system network (GbE equivalent)
- The network for IO servers (10GbE equivalent, using TOE)
- The network for local storage (10GbE equivalent, for iSCSI SRV/local storage)

VLAN is set per virtual interface.

The figure below describes the assignment of channel numbers.

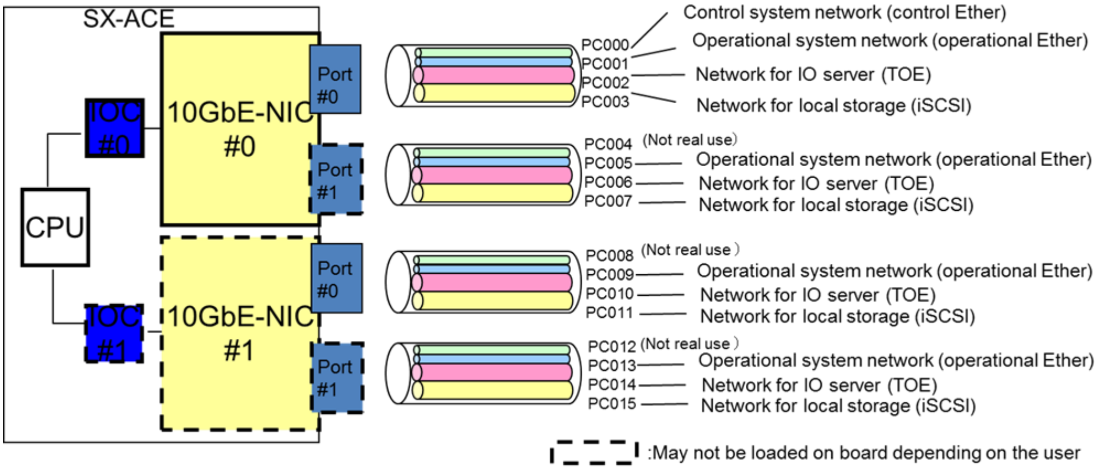

SX-ACE
IOC #0
10GbE-NIC #0
Port #0
Port #1
CPU
IOC #1
10GbE-NIC #1
Port #0
Port #1
PC000
PC001
PC002
PC003
Control system network (control Ether)
Operational system network (operational Ether)
Network for IO server (TOE)
Network for local storage (iSCSI)
PC004 (Not real use)
PC005
PC006
PC007
Operational system network (operational Ether)
Network for IO server (TOE)
Network for local storage (iSCSI)
PC008 (Not real use)
PC009
PC010
PC011
Operational system network (operational Ether)
Network for IO server (TOE)
Network for local storage (iSCSI)
PC012 (Not real use)
PC013
PC014
PC015
Operational system network (operational Ether)
Network for IO server (TOE)
Network for local storage (iSCSI)
:May not be loaded on board depending on the user

Figure 4-1　10GbE-NIC Configuration

Chapter5　Configuring IO servers

Refer to 5.1 to 5.4 if you use and build NEC’s IO servers and storage.

Refer to 5.5 for DDN SFA7990XE.

5.1　Preparing IO servers

The following program products should be installed to each IO server nodes:

- NEC Storage Manager Agent Utility
- StoragePathManager for Linux driver package
- EXPRESSCLUSTER X for Linux
- NEC Scalable Technology File System/Server (ScaTeFS server function)

Note:

When installing an OS on an IO server, select "Standard Partition" for the device type of "/ (root)".

The supported versions of the programs are as follows:

[IO server v4++ for standard model]

Table 5-1 IO server v4++ Supported distribution, kernel and software versions

Distribution	kernel	MLNX_OFED	EXPRESS CLUSTER	SPS	iSMccs
RHEL7.7	3.10.0-1062.el7.x86_64	4.7-1.0.0.1	4.3.4-1	7.3.1	-

[IO server v4+ for standard model]

Table 5-2 IO server v4+ Supported distribution, kernel and software versions

Distribution	kernel	MLNX_OFED	EXPRESS CLUSTER	SPS	iSMccs
RHEL7.6	3.10.0-957.el7.x86_64	4.6-4.1.2.0	4.1.1-1	7.2	10.3-005

[IO server v4 for standard model]

Table 5-3 IO server v4 Supported distribution, kernel and software versions

Distribution	kernel	MLNX_OFED	EXPRESS CLUSTER	SPS	iSMagent
RHEL7.4	3.10.0-693.el7.x86_64	4.2-1.2.0.0	3.3.5-1	7.0 6.7	9.7-003

[IO server v3 for standard model]

Table 5-4 IO server v3 Supported distributions, kernels and software versions

Distribution	kernel	MLNX_OFED	EXPRESS CLUSTER	SPS	iSMagent
RHEL7.4	3.10.0-693.el7.x86_64	4.2-1.2.0.0	3.3.5-1	6.7	9.7-003
RHEL7.3	3.10.0-514.26.2.el7.x86_64				

[IO server v1 for standard model and small-scale model]

Table 5-5 IO server v1 Supported distribution, kernel and software versions

Distribution	kernel	MLNX_ OFED	EXPRESS CLUSTER	SPS	iSMagent
RHEL6.4	2.6.32-358.23.2.el6.x86_64	-	3.2.0-1	6.2	8.4-002

Log in to each IO server as an administrator (that is, with root privileges) and configure the following settings: And, because the setting differs depending on a kind of distribution, sets the configuration for your distribution.

- IO targets design
- Creating the EXPRESSCLUSTER cluster configuration information
- Installing the NEC Storage Manager Agent Utility(iSMagent)
- Registering the host
- Assigning the logical disks
- Installing and setting up the PathManager for Linux driver package
- Installing the EXPRESSCLUSTER X for Linux
- Installing the DCB-compliant 10GbE-NIC driver
- Installing the IB driver

- Installing rsh-related packages
- Installing the ScaTeFS package
- Registering the ScaTeFS license
- Disabling SELinux
- Disabling firewalls
- Disabling prelink
- Disabling abrtd
- Configuring the network
- Setting the time (ntp)
- Setting the file system administration (fsadmin) account
- Setting up the internal disk (SSD)
- Setting the kernel parameter
- Setting syslog log rotation
- Setting updatedb.conf file
- Integrating as a ScaTeFS (scatefs_addios) IO server

Perform the above after storage configuration (RAID creation, zoning, etc.) is complete.
In the following explanation, two IO servers are described as “iosv00” and “iosv01”.

5.1.1 IO targets design

An IO target is a data store fundamental to the ScaTeFS file system. File data written from a client node are distributed to IO servers and then distributed and stored in IO targets of each IO server.
The IO target has a data region for storing data itself and a metadata region for storing the file type, update time, and other data. Multiple IO targets can be created, and the number of data regions and the number of metadata regions are always the same and in pairs.
An example of IO targets configuration for two IO servers is as follows:

Table 5-6 IO targets configuration IO server v1 for standard model and small-scale model

model	data region					metadata region					IO target
	HDD		Pool			HDD		Pool			
	capacity	number	RAID	number	LD	capacity	number	RAID	number	LD	
standard	1TB	36	6 (4+PQ)	6	6	600GB	6	10	1	1	6
	2TB				12						12
	3TB				18						18
	4TB				24						24
small-scale	1TB	36	6 (4+PQ)	6	6	600GB	6	10	1	1	6
	2TB				6						6
	3TB				12						12
	4TB				12						12

＊ The number of HDDs and pools is the number per a storage.

Table 5-7 IO targets configuration IO server v3 and v4 for standard model

model	data region					metadata region					IO target
	HDD		Pool			HDD		Pool			
	capacity	number	RAID	number	LD	capacity	number	RAID	number	LD	
standard	1TB	72	6 (4+PQ)	12	12	600GB	12	10	2	2	6
	2TB				24						12
	4TB				48						24
	6TB				72						36
	10TB		6 (8+PQ)	7	14						28
	12TB	80		8	16						32

＊ The number of HDDs and pools is the number per a storage.

IO server v4+ for a standard model can choose xfs as the data regions of the IO targets. In case of NLSAS, xfs is recommended as the disk type. In case of SAS and SSD, ext4 is recommended as the disk type.

Table 5-8 IO targets configuration IO server v4+ for standard model or later data region

data region							
Disk			Pool			recommended filesystem type	IO target
type	capacity	number	RAID	number	LD		
NLSAS	4TB	72	6(4+PQ)	12	12	xfs	12
	8TB	80	6(8+PQ)	8	8		8
	12TB	80	6(8+PQ)	8	8		8
SAS	1.2TB	72	6(4+PQ)	12	12	ext4	12
SSD	1.6TB	24	6(4+PQ)	4	4		4

＊ The number of disks and pools is the number per a storage.

Table 5-9 IO targets configuration IO server v4+ for standard model or later metadata region

metadata region							
Disk			Pool			recommended filesystem type	IO target
type	capacity	number	RAID	number	LD		
SAS	600GB	12	10	2	2	ext4	the number of data regions
SSD	400GB	6	10	1	1		

＊ The number of disks and pools is the number per a storage.

IO target ID is assigned by executing scatefs_addiot command (refer 5.2.5). Example of assigning of the IO target ID of the IO server composition (server:4, IO target:12) is described below:

Table 5-10 Assigning the IO target ID

IO Server	IOS#0	IOS#1	IOS#2	IOS#3
IO target ID	0	3	6	9
	1	4	7	10
	2	5	8	11

The above example is used in the following explanation.

5.1.2 LVM design

Design the number of metadata region partitions, the number of data region partitions, the number of striping ways and the order of IO target according to the configuration (pools, logical disks) of the NEC Storage disk array unit connected to the IO servers. The design examples for each IO server models are shown below:

[IO server v1 for standard model]

A design example when a data region consisting of 1-TB HDD is used is shown below:

- ScaTeFS data region
  When creating an LV with 4way striping, do as follows to distribute the load (*).
  *The check of SPS path is done in 5.2.2.
  Create the LV with 4way striping by using LDs in POOL1 and POOL2.
  Use the same LD number combinations for Storage1 and Storage3, and for Storage2 and Storage4.
  Do the same for POOL3 and POOL4, and POOL5 and POOL6.

Pool configuration

Pool	Storage1	Storage2	Storage3	Storage4
POOL1	LD1	LD1	LD1	LD1
POOL2	LD2	LD2	LD2	LD2
POOL3	LD3	LD3	LD3	LD3
POOL4	LD4	LD4	LD4	LD4
POOL5	LD5	LD5	LD5	LD5
POOL6	LD6	LD6	LD6	LD6

LVM configuration

Storage				iosv00			iosv01		
1	2	3	4	LV	IOT	order	LV	IOT	order
LD1	LD2	LD1	LD2	lv_data01	0	1	-	-	-

LD2	LD1	LD2	LD1	lv_data02	1	2	-	-	-
LD3	LD4	LD3	LD4	lv_data03	2	3	-	-	-
LD4	LD3	LD4	LD3	-	-	-	lv_data04	3	1
LD5	LD6	LD5	LD6	-	-	-	lv_data05	4	2
LD6	LD5	LD6	LD5	-	-	-	lv_data06	5	3

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 1 2
iosv01	3 4 5

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 1 2 3 4 5

- ScaTeFS metadata region

  When creating an LV with 2way striping, create logical disk combinations consisting of “Storage1 and Storage2” and “Storage3 and Storage4”.

Pool configuration

Pool	Storage1	Storage2	Storage3	Storage4
POOL0	LD0-2,3,4	LD0-2,3,4	LD0-2,3,4	LD0-2,3,4

*X in “LD0-X” means a partition.

LVM configuration

Storage				iosv00		iosv01	
1	2	3	4	LV	IOT	LV	IOT
LD0-2	LD0-2	-	-	lv_ctrl01	0	-	-
LD0-3	LD0-3	-	-	lv_ctrl02	1	-	-
LD0-4	LD0-4	-	-	lv_ctrl03	2	-	-
-	-	LD0-2	LD0-2	-	-	lv_ctrl04	3
-	-	LD0-3	LD0-3	-	-	lv_ctrl05	4

-	-	LD0-4	LD0-4	-	-	lv_ctrl06	5

[IO server v3 for standard model]

A design example when a data region consisting of 4-TB HDD is used is shown below:

- ScaTeFS data region

When creating an LV with 4way striping, use the same LD number combinations in POOL2 and POOL3 for Storage1 and Storage2. Do the same setting for POOL4 and others.

Pool configuration

Pool	Storage1	Storage2
POOL2	LD2,LD3,LD4,LD5	LD2,LD3,LD4,LD5
POOL3	LD6,LD7,LD8,LD9	LD6,LD7,LD8,LD9
POOL4	LDA,LDB,LDC,LDD	LDA,LDB,LDC,LDD
POOL5	LDE,LDF,LD10,LD11	LDE,LDF,LD10,LD11
POOL6	LD12,LD13,LD14,LD15	LD12,LD13,LD14,LD15
POOL7	LD16,LD17,LD18,LD19	LD16,LD17,LD18,LD19
POOL8	LD1A,LD1B,LD1C,LD1D	LD1A,LD1B,LD1C,LD1D
POOL9	LD1E,LD1F,LD20,LD21	LD1E,LD1F,LD20,LD21
POOL10	LD22,LD23,LD24,LD25	LD22,LD23,LD24,LD25
POOL11	LD26,LD27,LD28,LD29	LD26,LD27,LD28,LD29
POOL12	LD2A,LD2B,LD2C,LD2D	LD2A,LD2B,LD2C,LD2D
POOL13	LD2E,LD2F,LD30,LD31	LD2E,LD2F,LD30,LD31

LVM configuration

Storage1	Storage2	iosv00			iosv01		
		LV	IOT	order	LV	IOT	order
LD2,LD6	LD2,LD6	lv_data01	0	1	-	-	-
LD3,LD7	LD3,LD7	lv_data02	1	2	-	-	-
LD4,LD8	LD4,LD8	lv_data03	2	3	-	-	-
LD5,LD9	LD5,LD9	lv_data04	3	4	-	-	-
LDA,LDE	LDA,LDE	lv_data05	4	5	-	-	-

LDB,LDF	LDB,LDF	lv_data06	5	6	-	-	-
LDC,LD10	LDC,LD10	lv_data07	6	7	-	-	-
LDD,LD11	LDD,LD11	lv_data08	7	8	-	-	-
LD12,LD16	LD12,LD16	lv_data09	8	9	-	-	-
LD13,LD17	LD13,LD17	lv_data10	9	10	-	-	-
LD14,LD18	LD14,LD18	lv_data11	10	11	-	-	-
LD15,LD19	LD15,LD19	lv_data12	11	12	-	-	-
LD1A,LD1E	LD1A,LD1E	-	-	-	lv_data13	12	1
LD1B,LD1F	LD1B,LD1F	-	-	-	lv_data14	13	2
LD1C,LD20	LD1C,LD20	-	-	-	lv_data15	14	3
LD1D,LD21	LD1D,LD21	-	-	-	lv_data16	15	4
LD22,LD26	LD22,LD26	-	-	-	lv_data17	16	5
LD23,LD27	LD23,LD27	-	-	-	lv_data18	17	6
LD24,LD28	LD24,LD28	-	-	-	lv_data19	18	7
LD25,LD29	LD25,LD29	-	-	-	lv_data20	19	8
LD2A,LD2E	LD2A,LD2E	-	-	-	lv_data21	20	9
LD2B,LD2F	LD2B,LD2F	-	-	-	lv_data22	21	10
LD2C,LD30	LD2C,LD30	-	-	-	lv_data23	22	11
LD2D,LD31	LD2D,LD31	-	-	-	lv_data24	23	12

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 1 2 3 4 5 6 7 8 9 10 11
iosv01	12 13 14 15 16 17 18 19 20 21 22 23

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

- ScaTeFS metadata region

  Create LV without striping.

Pool configuration

Pool	Storage1	Storage2
POOL0	LD0-2,3,4,5,6,7	LD0-2,3,4,5,6,7
POOL1	LD1-2,3,4,5,6,7	LD1-2,3,4,5,6,7

*X in “LD0-X” means a partition.

LVM configuration

Storage1	Storage2	iosv00		iosv01	
		LV	IOT	LV	IOT
LD0-2		lv_ctrl01	0	-	-
LD0-3		lv_ctrl02	1	-	-
LD0-4		lv_ctrl03	2	-	-
LD0-5		lv_ctrl04	3	-	-
LD0-6		lv_ctrl05	4	-	-
LD0-7		lv_ctrl06	5	-	-
LD1-2		lv_ctrl07	6	-	-
LD1-3		lv_ctrl08	7	-	-
LD1-4		lv_ctrl09	8	-	-
LD1-5		lv_ctrl10	9	-	-
LD1-6		lv_ctrl11	10	-	-
LD1-7		lv_ctrl12	11	-	-
	LD0-2	-	-	lv_ctrl13	12
	LD0-3	-	-	lv_ctrl14	13
	LD0-4	-	-	lv_ctrl15	14
	LD0-5	-	-	lv_ctrl16	15
	LD0-6	-	-	lv_ctrl17	16
	LD0-7	-	-	lv_ctrl18	17
	LD1-2	-	-	lv_ctrl19	18
	LD1-3	-	-	lv_ctrl20	19
	LD1-4	-	-	lv_ctrl21	20
	LD1-5	-	-	lv_ctrl22	21
	LD1-6	-	-	lv_ctrl23	22
	LD1-7	-	-	lv_ctrl24	23

[Creating LV without striping in the data region]

When creating LV without striping in the data region, design the order of the IO target to distribute the load of the IO server and the storage as much as possible. This order is used to the setting item iotid of 5.3.1 Creating ScaTeFS. Design the order which meets the following conditions as much as possible:

a) The IO server uses storages alternately.

b) The IO server uses an odd number, an even number alternately in the pool number in the storage.

c) When more than one LD exists at one pool, use the next LD after using the first LD of the pool.

Example of the design of LVM of the storage composition (pool:4, LD:8) is described below:

Pool	Storage 1	Storage 2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order
POOL2	LD2	-	lv_data01	0	1	-	-	-
	LD3	-	lv_data02	1	5	-	-	-
POOL3	LD4	-	lv_data03	2	3	-	-	-
	LD5	-	lv_data04	3	7	-	-	-
POOL4	-	LD6	lv_data05	4	4	-	-	-
	-	LD7	lv_data06	5	8	-	-	-
POOL5	-	LD8	lv_data07	6	2	-	-	-
	-	LD9	lv_data08	7	6	-	-	-
POOL2	-	LD2	-	-		lv_data09	8	1
	-	LD3	-	-		lv_data10	9	5
POOL3	-	LD4	-	-		lv_data11	10	3
	-	LD5	-	-		lv_data12	11	7
POOL4	LD6	-	-	-		lv_data13	12	4
	LD7	-	-	-		lv_data14	13	8
POOL5	LD8	-	-	-		lv_data15	14	2
	LD9	-	-	-		lv_data16	15	6

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 6 2 4 1 7 3 5
iosv01	8 14 10 12 9 15 11 13

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 6 2 4 1 7 3 5 8 14 10 12 9 15 11 13

A design example when a data region consisting of 10TB HDD is used is shown below:

- ScaTeFS data region

Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order
POOL2	LD2	-	lv_data01	0	1	-	-	-
	LD3	-	lv_data02	1	8	-	-	-
POOL3	LD4	-	lv_data03	2	3	-	-	-
	LD5	-	lv_data04	3	10	-	-	-
POOL4	LD6	-	lv_data05	4	5	-	-	-
	LD7	-	lv_data06	5	12	-	-	-
POOL5	-	LD8	lv_data07	6	2	-	-	-
	-	LD9	lv_data08	7	9	-	-	-
POOL6	-	LDA	lv_data09	8	4	-	-	-
	-	LDB	lv_data10	9	11	-	-	-
POOL7	-	LDC	lv_data11	10	6	-	-	-
	-	LDD	lv_data12	11	13	-	-	-
POOL8	-	LDE	lv_data13	12	7	-	-	-
	-	LDF	lv_data14	13	14	-	-	-
POOL2	-	LD2	-	-	-	lv_data15	14	1
	-	LD3	-	-	-	lv_data16	15	8
POOL3	-	LD4	-	-	-	lv_data17	16	3

	-	LD5	-	-	-	lv_data18	17	10
POOL4	-	LD6	-	-	-	lv_data19	18	5
	-	LD7	-	-	-	lv_data20	19	12
POOL5	LD8	-	-	-	-	lv_data21	20	2
	LD9	-	-	-	-	lv_data22	21	9
POOL6	LDA	-	-	-	-	lv_data23	22	4
	LDB	-	-	-	-	lv_data24	23	11
POOL7	LDC	-	-	-	-	lv_data25	24	6
	LDD	-	-	-	-	lv_data26	25	13
POOL8	LDE	-	-	-	-	lv_data27	26	7
	LDF	-	-	-	-	lv_data28	27	14

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 6 2 8 4 10 12 1 7 3 9 5 11 13
iosv01	14 20 16 22 18 24 26 15 21 17 23 19 25 27

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 6 2 8 4 10 12 1 7 3 9 5 11 13 14 20 16 22 18 24 26 15 21 17 23 19 25 27

- ScaTeFS metadata region

  Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00		iosv01	
			LV	IOT	LV	IOT
POOL0	LD0-2	-	lv_ctrl01	0	-	-
	LD0-3	-	lv_ctrl02	1	-	-
	LD0-4	-	lv_ctrl03	2	-	-
	LD0-5	-	lv_ctrl04	3	-	-
	LD0-6	-	lv_ctrl05	4	-	-

	LD0-7	-	lv_ctrl06	5	-	-
	LD0-8	-	lv_ctrl07	6	-	-
POOL1	LD1-2	-	lv_ctrl08	7	-	-
	LD1-3	-	lv_ctrl09	8	-	-
	LD1-4	-	lv_ctrl10	9	-	-
	LD1-5	-	lv_ctrl11	10	-	-
	LD1-6	-	lv_ctrl12	11	-	-
	LD1-7	-	lv_ctrl13	12	-	-
	LD1-8	-	lv_ctrl14	13	-	-
POOL0	-	LD0-2	-	-	lv_data15	14
	-	LD0-3	-	-	lv_data16	15
	-	LD0-4	-	-	lv_data17	16
	-	LD0-5	-	-	lv_data18	17
	-	LD0-6	-	-	lv_data19	18
	-	LD0-7	-	-	lv_data20	19
	-	LD0-8	-	-	lv_data21	20
POOL1	-	LD1-2	-	-	lv_data22	21
	-	LD1-3	-	-	lv_data23	22
	-	LD1-4	-	-	lv_data24	23
	-	LD1-5	-	-	lv_data25	24
	-	LD1-6	-	-	lv_data26	25
	-	LD1-7	-	-	lv_data27	26
	-	LD1-8	-	-	lv_data28	27

[IO server v4 for standard model]

A design example when a data region consisting of 12TB HDD is used is shown below:

- ScaTeFS data region

Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order

POOL2	LD2	-	lv_data01	0	1	-	-	-
	LD3	-	lv_data02	1	9	-	-	-
POOL3	LD4	-	lv_data03	2	3	-	-	-
	LD5	-	lv_data04	3	11	-	-	-
POOL4	LD6	-	lv_data05	4	5	-	-	-
	LD7	-	lv_data06	5	13	-	-	-
POOL5	LD8		lv_data07	6	7	-	-	-
	LD9		lv_data08	7	15	-	-	-
POOL6	-	LDA	lv_data09	8	8	-	-	-
	-	LDB	lv_data10	9	16	-	-	-
POOL7	-	LDC	lv_data11	10	6	-	-	-
	-	LDD	lv_data12	11	14	-	-	-
POOL8	-	LDE	lv_data13	12	4	-	-	-
	-	LDF	lv_data14	13	12	-	-	-
POOL9	-	LD10	lv_data15	14	2	-	-	-
	-	LD11	lv_data16	15	10	-	-	-
POOL2	-	LD2	-	-	-	lv_data17	16	1
	-	LD3	-	-	-	lv_data18	17	9
POOL3	-	LD4	-	-	-	lv_data19	18	3
	-	LD5	-	-	-	lv_data20	19	11
POOL4	-	LD6	-	-	-	lv_data21	20	5
	-	LD7	-	-	-	lv_data22	21	13
POOL5	-	LD8	-	-	-	lv_data23	22	7
	-	LD9	-	-	-	lv_data24	23	15
POOL6	LDA	-	-	-	-	lv_data25	24	8
	LDB	-	-	-	-	lv_data26	25	16
POOL7	LDC	-	-	-	-	lv_data27	26	6
	LDD	-	-	-	-	lv_data28	27	14
POOL8	LDE	-	-	-	-	lv_data29	28	4
	LDF	-	-	-	-	lv_data30	29	12
POOL9	LD10		-	-	-	lv_data31	30	2
	LD11		-	-	-	lv_data32	31	10

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 14 2 12 4 10 6 8 1 15 3 13 5 11 7 9
iosv01	16 30 18 28 20 26 22 24 17 31 19 29 21 27 23 25

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 14 2 12 4 10 6 8 1 15 3 13 5 11 7 9 16 30 18 28 20 26 22 24 17 31 19 29 21 27 23 25

- ScaTeFS meta region

Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00		iosv01	
			LV	IOT	LV	IOT
POOL0	LD0-2	-	lv_ctrl01	0	-	-
	LD0-3	-	lv_ctrl02	1	-	-
	LD0-4	-	lv_ctrl03	2	-	-
	LD0-5	-	lv_ctrl04	3	-	-
	LD0-6	-	lv_ctrl05	4	-	-
	LD0-7	-	lv_ctrl06	5	-	-
	LD0-8	-	lv_ctrl07	6	-	-
	LD0-9	-	lv_ctrl08	7	-	-
POOL1	LD1-2	-	lv_ctrl09	8	-	-
	LD1-3	-	lv_ctrl10	9	-	-
	LD1-4	-	lv_ctrl11	10	-	-
	LD1-5	-	lv_ctrl12	11	-	-
	LD1-6	-	lv_ctrl13	12	-	-
	LD1-7	-	lv_ctrl14	13	-	-
	LD1-8		lv_ctrl15	14	-	-

	LD1-9		lv_ctrl16	15	-	-
POOL0	-	LD0-2	-	-	lv_ctrl17	16
	-	LD0-3	-	-	lv_ctrl18	17
	-	LD0-4	-	-	lv_ctrl19	18
	-	LD0-5	-	-	lv_ctrl20	19
	-	LD0-6	-	-	lv_ctrl21	20
	-	LD0-7	-	-	lv_ctrl22	21
	-	LD0-8	-	-	lv_ctrl23	22
	-	LD0-9	-	-	lv_ctrl24	23
POOL1	-	LD1-2	-	-	lv_ctrl25	24
	-	LD1-3	-	-	lv_ctrl26	25
	-	LD1-4	-	-	lv_ctrl27	26
	-	LD1-5	-	-	lv_ctrl28	27
	-	LD1-6	-	-	lv_ctrl29	28
	-	LD1-7	-	-	lv_ctrl30	29
		LD1-8	-		lv_ctrl31	30
		LD1-9	-		lv_ctrl32	31

[IO server v4+ for standard model or later]

A design example when a data region consisting of 12TB HDD is used is shown below:

- ScaTeFS data region

Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order
POOL2	LD2	-	lv_data01	0	1	-	-	-
POOL3	LD3	-	lv_data02	1	3	-	-	-
POOL4	LD4	-	lv_data03	2	5	-	-	-
POOL5	LD5	-	lv_data04	3	7	-	-	-
POOL6	LD6	-	lv_data05	4	8	-	-	-
POOL7	LD7	-	lv_data06	5	6	-	-	-

Pool	Storage 1	Storage 2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order
POOL8	LD8	-	lv_data07	6	4	-	-	-
POOL9	LD9	-	lv_data08	7	2	-	-	-
POOL2	-	LD2	-	-	-	lv_data09	8	1
POOL3	-	LD3	-	-	-	lv_data10	9	3
POOL4	-	LD4	-	-	-	lv_data11	10	5
POOL5	-	LD5	-	-	-	lv_data12	11	7
POOL6	-	LD6	-	-	-	lv_data13	12	5
POOL7	-	LD7	-	-	-	lv_data14	13	6
POOL8	-	LD8	-	-	-	lv_data15	14	4
POOL9	-	LD9	-	-	-	lv_data16	15	2

The order of the IO target of the LVM configuration is described below:

IO server	order of the IO target
iosv00	0 7 1 6 2 5 3 4
iosv01	8 15 9 14 10 13 11 12

Setting item iotid of 5.3.1 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

Item	Setting value
iotid	0 7 1 6 2 5 3 4 8 15 9 14 10 13 11 12

- ScaTeFS meta region

Create LV without striping.

LVM configuration

Pool	Storage 1	Storage 2	iosv00		iosv01	
			LV	IOT	LV	IOT
POOL0	LD0-2	-	lv_ctrl01	0	-	-
	LD0-3	-	lv_ctrl02	1	-	-
	LD0-4	-	lv_ctrl03	2	-	-
	LD0-5	-	lv_ctrl04	3	-	-

Pool	Storage 1	Storage 2	iosv00		iosv01	
			LV	IOT	LV	IOT
POOL1	LD1-2	-	lv_ctrl05	4	-	-
	LD1-3	-	lv_ctrl06	5	-	-
	LD1-4	-	lv_ctrl07	6	-	-
	LD1-5	-	lv_ctrl08	7	-	-
POOL0	-	LD0-2	-	-	lv_ctrl09	8
	-	LD0-3	-	-	lv_ctrl10	9
	-	LD0-4	-	-	lv_ctrl11	10
	-	LD0-5	-	-	lv_ctrl12	11
POOL1	-	LD1-2	-	-	lv_ctrl13	12
	-	LD1-3	-	-	lv_ctrl14	13
	-	LD1-4	-	-	lv_ctrl15	14
	-	LD1-5	-	-	lv_ctrl16	15

*X in “LD0-X” means a partition.

5.1.3 Creating the EXPRESSCLUSTER cluster configuration information

Before configuring the IO servers, create the EXPRESSCLUSTER cluster configuration information. This information is used in “5.4 Setting the EXPRESSCLUSTER”.

For details, see "APPENDIX A Procedure for Creating EXPRESSCLUSTER Cluster Configuration Information (Offline version)".

5.1.4 Installing the NEC Storage Manager Agent Utility(iSMagent)

[IO server v4+ for standard model or later]

To simplify the IO server settings, go to 5.1.5 .

[IO server v1, v3 and v4 for standard model]

See “Installation of Storage Manager Agent Utility (Linux)” in the NEC Storage Manager Installation Guide.

5.1.5 Registering the host

Stop Access Control for configuring storage settings.

This can be done on the storage management screen of NEC Storage Manager.

See “10.3.3.4 Access Control Advanced Settings” in the NEC Storage Software

Configuration Setting Tool User's Manual (GUI) for the M Series.

[IO server v4+ for standard model or later]

To simplify the IO server settings, go to 5.1.7 .

[IO server v1, v3 and v4 for standard model]

See (1)-[2] “Collection of host information in new Linux server via disk arrays” in “Appendix G Configuration Settings Simplified by Host Information Collection/Storage” in the NEC Storage Manager Installation Guide.

Confirm the target files.

* The host* files under /sys/class/fc_host/ are the target files.

```
# ls -l /sys/class/fc_host/host*/issue_lip
--w------- 1 root root 4096 Apr   8 16:36 /sys/class/fc_host/host1/issue_lip
--w------- 1 root root 4096 Apr   8 16:36 /sys/class/fc_host/host2/issue_lip
--w------- 1 root root 4096 Apr   8 16:36 /sys/class/fc_host/host3/issue_lip
--w------- 1 root root 4096 Apr   8 16:36 /sys/class/fc_host/host4/issue_lip
```

Have the volume recognized by the OS.

```
# echo "1" > /sys/class/fc_host/host1/issue_lip
# echo "1" > /sys/class/fc_host/host2/issue_lip
# echo "1" > /sys/class/fc_host/host3/issue_lip
# echo "1" > /sys/class/fc_host/host4/issue_lip
```

Execute the host information collection command (iSMcc_hostinfo command).

```
# iSMcc_hostinfo -store
iSMcc_hostinfo: Info:      iSM11700: Please wait a minute.
iSMcc_hostinfo: Info:      iSM11770: Host Information was exported successfully. (Disk Array=iost05)
(code=5ec6-5900-00a2-0000)
iSMcc_hostinfo: Info:      iSM11770: Host Information was exported successfully. (Disk Array=iost07)
(code=5ec6-5900-00a2-0000)
iSMcc_hostinfo: Info:      iSM11770: Host Information was exported successfully. (Disk Array=iost08)
(code=5ec6-5900-00a2-0000)
iSMcc_hostinfo: Info:      iSM11770: Host Information was exported successfully. (Disk Array=iost06)
(code=5ec6-5900-00a2-0000)
iSMcc_hostinfo: Info:      iSM11100: Command has completed successfully.
```

Note:
The following warning messages might be output depending on the configuration, but this can be ignored.

```
# iSMcc_hostinfo -store
iSMcc_hostinfo: Info:        iSM11700: Please wait a minute.
iSMcc_hostinfo: Warning:    iSM11773: Information collection was skipped. (IP Address) (code=2fa3-5700-0001-
0000)
iSMcc_hostinfo: Warning:    iSM11774: A part of Host Information was exported. (Disk Array=iost05)
(code=2fa3-5900-00a0-0000)
iSMcc_hostinfo: Warning:    iSM11774: A part of Host Information was exported. (Disk Array=iost06)
(code=2fa3-5900-00a0-0000)
iSMcc_hostinfo: Warning:    iSM11774: A part of Host Information was exported. (Disk Array=iost08)
(code=2fa3-5900-00a0-0000)
iSMcc_hostinfo: Warning:    iSM11774: A part of Host Information was exported. (Disk Array=iost07)
(code=2fa3-5900-00a0-0000)
iSMcc_hostinfo: Warning:    iSM11775: Command has completed with warning status. (code=2fa3-2703-0004-
0000)
```

5.1.6　Assigning the logical disks

[IO server v4+ for standard model or later]

To simplify the IO server settings, go to 5.1.7 .

[IO server v1, v3 and v4 for standard model]

Assign logical disks to the connected IO servers from the storage management screen of NEC Storage Manager.

Assign logical disks to the IO servers.

Assign all logical disks created in the storage units.

See “10.1 Assignment of Logical Disk” in the NEC Storage Software Configuration Setting Tool User's Manual (GUI) for the M Series.

Start Access Control for configuring storage settings.

This can be done on the storage management screen of NEC Storage Manager.

See “10.3.3.4 Access Control Advanced Settings” in the NEC Storage Software Configuration Setting Tool User's Manual (GUI) for the M Series.

5.1.7　Installing and setting up the PathManager for Linux driver package

Install the PathManager for Linux driver package by the following procedure. See the the NEC Storage PathManager for Linux Installation Guide and the NEC Storage PathManager User's Guide (Linux) for more details.

(1) Install

a) When sg3_utils and lvm2 package package are not installed, install them from the OS distribution.

b) Go to the directory mounted to the PathManager Installation CD.

c) Execute the install script.

```
# sh install.sh -i --silent
```

Note that the OS will be restarted automatically after the package is installed.

(2) Check the SCSI disk is NEC Storage

Execute the sg_scan command and confirm the SCSI disks recognized by the OS. If "NEC" and "DISK ARRAY" are displayed, this SCSI disk is NEC Storage.

```
# sg_scan -i /dev/sdc
/dev/sdc: scsi8 channel=0 id=0 lun=0 [em]
    NEC         DISK ARRAY          1000 [rmb=0 cmdq=1 pqual=0 pdev=0x0]
#
```

(3) Modify the setting file of LVM (/etc/lvm/lvm.conf)

Be sure to change the filter settings as described in “Appendix B How to add to LVM” in the NEC Storage PathManager for Linux Installation Guide.

a) Setting the filter of device

[In case of RHEL7]

Modify the " global_filter " entry in "devices{}" area.

The example which permits all PathManager devices.

```
global_filter = [ "a|/dev/dd.*|", "r|/dev/.*|" ]
```

[In case of RHEL6]

Modify the "filter" entry in "devices{}" area.

The example which permits all PathManager devices.

```
filter = [ "a|/dev/dd.*|", "r|/dev/.*|" ]
```

b) Add the "types" entry in "devices{}" area.

```
types = [ "dd", 16 ]
```

5.1.8 Installing the EXPRESSCLUSTER X for Linux

Install the EXPRESSCLUSTER X for Linux by the following procedure. See the EXPRESSCLUSTER X for Linux Installation and Configuration Guide for more details.

For how to set up EXPRESSCLUSTER, see “5.4 Setting the EXPRESSCLUSTER” in this guide.

(1) Install

Installing the EXPRESSCLUSTER package.

```
# rpm -ivh expresscls-<version>.<architecture>.rpm
```

(2) Registering the EXPRESSCLUSTER license

Registering the license by specifying the license file.

[EXPRESSCLUSTER X 4.x]

```
# clplcnsc -i filepath
```

[EXPRESSCLUSTER X 3.x]

```
[In case of RHEL7]
# clplcnsc -i filepath -p BASE33
[In case of RHEL6]
# clplcnsc -i filepath -p BASE32
```

(3) [In case of RHEL7] LVM metadata daemon settings

Be sure to change the LVM metadata daemon settings as described in "Chapter 5 Notes and Restrictions" in the EXPRESSCLUSTER X for Linux Getting Started Guide.

a) Execute the following command to stop the LVM metadata daemon.

```
# systemctl stop lvm2-lvmetad.service
```

b) Edit /etc/lvm/lvm.conf to set the value of use_lvmetad to 0.

```
use_lvmetad = 0
```

5.1.9 Installing the DCB-compliant 10GbE-NIC driver

Only when you use DCB-compliant 10GbE-NIC, do this setting.

The RPM binary package provided by the 10GbE-NIC vendor may not support the DCB function as is. Install the 10GbE-NIC driver according to the installation procedure obtained from the NEC support department.

5.1.10 Installing the IB driver

Only when you use IB HCA, do this setting.

Install the IB driver by the following steps. ScaTeFS supports MLNX_OFED 3.4-1.0.0.0 or later.

(1) Get MLNX_OFED package

MLNX_OFED versions supported by IO Server are shown in the following table.

OS	MLNX_OFED version
RHEL7.3	4.2-1.2.0.0
RHEL7.4	
RHEL7.6	4.6-4.1.2.0
RHEL7.7	4.7-1.0.0.1

Please download the applicable MLNX_OFED from the official site of NVIDIA.

https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/

(*) MLNX_OFED 4.6-4.1.2.0 is not published at the above URL. Please download from the following URL:

https://mellanox.my.salesforce.com/sfc/p/#500000007heg/a/1T000000cCrw/jAKX3brAtwtWng6sVqHpSXf2pT8UrSUL2rMKpn3c4ng

Password: mgIdJQfI

If you cannot download MLNX_OFED, please contact the NEC support department.

(2) Install

a) When the following packages are not installed, install them from the OS distribution.

lsof gtk2 atk cairo tcl tcsh tk pciutils

b) Mount the ISO file on a directory. It is mounted on /mnt/iso in the following example.

```
# mount -t iso9660 -o loop MLNX_OFED_LINUX-4.2-1.2.0.0-rhel7.3-x86_64.iso /mnt/iso
```

c) Execute the install script.

```
# /mnt/iso/mlnxofedinstall
```

You will be asked if you delete the old IB related packages and continue to install, input “y”.

```
This program will install the MLNX_OFED_LINUX package on your machine.
Note that all other Mellanox, OEM, OFED, or Distribution IB packages will be removed.
Do you want to continue?[y/N]:
```

d) Unmount the ISO file.

```
# umount /mnt/iso
```

(3) Reboot OS and load the IB driver

```
# reboot
```

5.1.11 Installing rsh-related packages

If remote shell (rsh) related packages are not installed, install the following packages from the OS distributions.

rsh

rsh-server

[In case of RHEL6]xinetd

[In case of RHEL7]

Enable the rsh server functions.

```
# systemctl enable rsh.socket
```

Open /etc/systemd/system/sockets.target.wants/rsh.socket in an editor such as vi. And add the following parameter:

add
[Unit] Description=Remote Shell Facilities Activation Socket

```
[Socket]
ListenStream=514
Accept=true
MaxConnections=10000 (Added)

[Install]
WantedBy=sockets.target
```

Start the rsh server functions.

```
# systemctl start rsh.socket
# systemctl daemon-reload
```

[In case of RHEL6]

Open /etc/xinetd.d/rsh in an editor such as vi. And add and delete the following parameter:

add	
per_source	= UNLIMITED
instances	= UNLIMITED
cps	= 10000 10

delete	
log_on_success	+= USERID
log_on_failure	+= USERID

Enable the rsh server functions.

```
# chkconfig rsh on
# /etc/init.d/xinetd start
```

5.1.12 Installing the ScaTeFS package

Install the following ScaTeFS/Server package in all IO server nodes.

The version of the scatefs-srv package must be consistent among all IO servers.

Note that you need to install the sos package since version 3.5.

The procedures of installation and update of ScaTeFS/Server package are below:

5.1.12.1 When using the HPC Software License

The following packages are used for installation of ScaTeFS/Server:

a) ScaTeFS/Server

b) TSUBASA-soft-release-ve1

c) License Access Library

a) is paid software package. The ways to get ScaTeFS/Server package are different depending on whether you have the PP support contract of ScaTeFS/Server or not.
b) and c) are the free software packages. These packages are registered with the NEC yum repository (the yum repository for the free software) and are installed using the yum command.
c) is installed in the section 5.1.13.

The procedures of installation, update and uninstallation of ScaTeFS/Server package is different depending on whether you have the PP support contract of ScaTeFS/Server or not. The following explanation is divided into two cases where the PP support is contracted and the PP support is not contracted.

[If you have the PP support contract of ScaTeFS/Server, see below:]

(1) Setting yum repository (PP support contract of ScaTeFS/Server available)

Set the configuration of the yum repositories to install necessary software. You can use the yum repositories on the Internet online, or you can build the yum repositories locally and use them offline.
For the procedure of setting the yum repository, refer to "3.1 Prepration of installation" in "SX-Aurora TSUBASA Installation Guide". At this time, replace VH in the sentence with the target machine and read VE1 part for the architecture.
[In case of RHEL7.3 – RHEL7.6]
Change the end of "baseurl" to "scatefs_el7.7" in the configuration file for the paid software.

(2) Installing (PP support contract of ScaTeFS/Server available)

Install the ScaTeFS/Server package.

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-server
# yum group install scatefs-server
```

If you use the monitoring function which collect and monitor the ScaTeFS filesystems statistics in real time, install the monitoring package.

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-server-monitoring
# yum group install scatefs-server-monitoring
```

(3) Updating (PP support contract of ScaTeFS/Server available)

Update the ScaTeFS/Server package.

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-server
# yum group update scatefs-server
```

When you are using the monitoring function which collect and monitor the ScaTeFS filesystems statistics in real time, update the monitoring package.

```
# systemctl stop zabbix-agent
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-server-monitoring
# yum group update scatefs-server-monitoring
# systemctl start zabbix-agent
```

[If you do not have the PP support contract of ScaTeFS/Server, see below:]

(1) Setting yum repository (No PP support contract of ScaTeFS/Server)

Set the configuration of the yum repository to install necessary software. You can use the yum repository on the Internet online, or you can build the yum repository locally and use it offline.

For the procedure of setting the yum repository, refer to "3.1 Prepration of installation" in "SX-Aurora TSUBASA Installation Guide". At this time, replace VH in the sentence with the target machine and read VE1 part for the architecture.

(2) Getting zip file including ScaTeFS/Server package (No PP support contract of ScaTeFS/Server)

Download the zip file including the ScaTeFS/Server package using the internet delivery product download service.

Transfer the downloaded zip file to the installation target machine and extract the files from it.

```
# unzip ScaTeFS_S-YYYYMMDD.zip
(YYYYMMDD is a date)
```

(3) Installing (No PP support contract of ScaTeFS/Server)

Install the ScaTeFS/Server package.

[RHEL/CentOS7.7]

```
# yum install scatefs-server/el7.7/scatefs-server/*.rpm
```

[RHEL7.6]

```
# yum install scatefs-server/el7.6/scatefs-server/*.rpm
```

If you use the monitoring function which collect and monitor the ScaTeFS filesystems statistics in real time, install the monitoring package.

[RHEL/CentOS7.7]

```
# yum install scatefs-server/el7.7/scatefs-server-monitoring/*.rpm
```

[RHEL7.6]

```
 # yum install scatefs-server/el7.6/scatefs-server-monitoring/*.rpm
```

5.1.12.2　When using the SX Cross Software Node-lock License

Install the following package in all IO server nodes:

scatefs-srv-VER.x86_64

In case of updating the package, update as following:

```
# rpm -Uvh scatefs-srv-VER.x86_64.rpm
```

5.1.13 Registering the ScaTeFS license

Register the license.

See the HPC Software License Management Guide about details of the procedure.

* When using the SX Cross Software Node-lock License, see the SX Cross Software Node-lock License Installation Guide instead of the HPC Software License Management Guide.

5.1.14 Disabling SELinux

SELinux can be enabled and disabled by using the following command:

```
# /usr/sbin/getenforce
Disabled
```

If Enabled or Enforcing is displayed, edit the /etc/selinux/config file to specify SELINUX=disabled. The OS must be restarted to enable this setting.

5.1.15 Disabling firewalls

[In case of RHEL7]

Check the firewall setting by using the systemctl command.

```
# systemctl list-unit-files | grep firewalld
firewalld.service                              enabled
```

If the firewall is enabled, disable it by using the following procedure:

```
# systemctl disable firewalld
# systemctl list-unit-files | grep firewalld
firewalld.service                              disabled
# systemctl stop firewalld
```

[In case of RHEL6]

Check the firewall setting by using the chkconfig command.

```
# /sbin/chkconfig --list iptables
iptables        0:off    1:off    2:on     3:on     4:on     5:on     6:off
```

If the firewall is enabled, disable it by using the following procedure:

```
# /sbin/chkconfig iptables off
# /sbin/chkconfig --list iptables
iptables        0:off    1:off    2:off    3:off    4:off    5:off    6:off
# /etc/init.d/iptables stop
```

5.1.16 Disabling prelink

[In case of RHEL6]

Edit the /etc/sysconfig/prelink file to specify PRELINKING=no.

```
# vi /etc/sysconfig/prelink
                ----
                PRELINKING=no
                ----
```

Disable prelink.

```
# prelink -ua
```

Note that the following error messages might be displayed, but this can be ignored:

```
prelink: /usr/lib64/samba/libserver-role-samba4.so: Could not find one of the dependencies
prelink: /usr/lib64/firefox/plugin-container: Could not find one of the dependencies
```

5.1.17 Disabling abrtd

[In case of RHEL7]

Check the abrt settings by using the systemctl command.

```
# systemctl list-unit-files | grep abrt
abrt-ccpp.service                         enabled
abrt-oops.service                         enabled
abrt-pstoreoops.service                   disabled
abrt-vmcore.service                       enabled
abrt-xorg.service                         enabled
abrtd.service                             enabled
```

If the abrt settings are enabled, disable it by using the following procedure:

```
# yum remove abrt abrt-libs
```

[In case of RHEL6]

Check the abrtd setting by using the chkconfig command.

```
# /sbin/chkconfig --list abrtd
abrtd           0:off   1:off   2:off   3:on    4:off   5:on    6:off
```

If the abrtd is enabled, disable it by using the following procedure:

```
# /sbin/chkconfig abrtd off
# /sbin/chkconfig --list abrtd
abrtd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
# /etc/init.d/abrtd stop
```

5.1.18 Configuring the network

Multiple network ports are used for IO servers. Configure the IP address setting for each network port.

- Operational/management port
  It is used to log in to IO servers via the network. It is also used for time synchronization between servers using ntp, and for communication between servers by the ScaTeFS commands.

- File system port (10GbE)
  It is used for file access from the ScaTeFS client. For ports with the same NIC, use bonding to bundle two ports. Also, set the floating IP address using EXPRESSCLUSTER.

- File system port (IB)
  It is used for file access from the ScaTeFS client. Create the network interface file and configure it. Also, set the floating IP address using EXPRESSCLUSTER.

- Port for interconnect between IO servers

It is used for communication between IO server nodes. This network is a closed network consisting of a pair of IO servers, so therefore ensure the network addresses do not conflict with the network environment in use. Two IP addresses are needed. Assign different IP addresses to the IO servers with an even ID number and an odd ID number.

Example network settings for the file system ports (10GbE) and ports connecting the IO servers are shown below:

5.1.18.1 Configuring the network interfaces of the file system ports for 10GbE and the ports for interconnect between IO servers (bonding).

This section describes how to configure bonding using the example below:
When not using 10GbE as file system ports, the setting for a bonding of file system ports is not needed. Set only a bonding of ports for interconnect between IO servers.

Example:
Target machine: IO server
File system port

[In case of RHEL7]

ens28f4, ens28f4d1 : bond0(172.16.6.6)
ens61f4, ens61f4d1 : bond1(172.16.7.6)

[In case of RHEL6]

eth0,eth1 : bond0(172.16.6.6)
eth2,eth3 : bond1(172.16.7.6)
netmask : 255.255.255.128
vlanid of bond0:12
vlanid of bond1:14

Port for interconnect between IO servers (IO server0,IO server1)

[In case of RHEL7]

IO server0 ens27f0, ens27f1: bond2(10.2.0.10)
IO server1 ens27f0, ens27f1: bond2(10.2.0.11)

[In case of RHEL6]

IO server0 eth4, eth5: bond2(10.2.0.10)
IO server1 eth4, eth5: bond2(10.2.0.11)
netmask: 255.255.255.0

For Red Hat Enterprise Linux, the administrator can bind multiple network interfaces to a single channel by using the bonding kernel module and the special network interface called channel bonding interface.

[In case of RHEL7]

To create a channel bonding interface, create a file using nmcli or nmtui command.The nmcli command execution examples for each ports are shown below:

[File system port (10GbE)]

bond0.12

```
# nmcli connection add type bond con-name bond0 ifname bond0
# nmcli connection add type ethernet autoconnect yes ifname ens28f4 master bond0
# nmcli connection add type ethernet autoconnect yes ifname ens28f4d1 master
bond0
# nmcli connection modify bond0 ipv4.never-default true
# nmcli connection modify bond0 ipv4.method disabled ipv6.method ignore
# nmcli connection modify bond0 +bond.options
mode=802.3ad,miimon=100,xmit_hash_policy=layer2+3
# nmcli connection up bond0
# nmcli connection add type vlan con-name bond0.12 dev bond0 id 12
# nmcli connection modify bond0.12 ipv4.never-default true
# nmcli connection modify bond0.12 ipv4.method disabled ipv6.method ignore
```

bond1.14

```
# nmcli connection add type bond con-name bond1 ifname bond1
# nmcli connection add type ethernet autoconnect yes ifname ens61f4 master bond1
# nmcli connection add type ethernet autoconnect yes ifname ens61f4d1 master
bond1
# nmcli connection modify bond1 ipv4.never-default true
# nmcli connection modify bond1 ipv4.method disabled ipv6.method ignore
# nmcli connection modify bond1 +bond.options
mode=802.3ad,miimon=100,xmit_hash_policy=layer2+3
# nmcli connection up bond1
# nmcli connection add type vlan con-name bond1.14 dev bond1 id 14
# nmcli connection modify bond1.14 ipv4.never-default true
# nmcli connection modify bond1.14 ipv4.method disabled ipv6.method ignore
```

[Port for interconnect between IO servers]

bond2

```
# nmcli connection add type bond con-name bond2 ifname bond2
# nmcli connection add type ethernet autoconnect yes ifname ens27f0 master bond2
# nmcli connection add type ethernet autoconnect yes ifname ens27f1 master bond2
# nmcli connection modify bond2 ipv4.never-default true
# nmcli connection modify bond2 ipv6.method ignore
# nmcli connection modify bond2 +bond.options
mode=802.3ad,miimon=100,xmit_hash_policy=layer2+3

IO server0
  # nmcli connection modify bond2 ipv4.method manual ipv4.address "10.2.0.10/24"
IO server1
  # nmcli connection modify bond2 ipv4.method manual ipv4.address "10.2.0.11/24"

# nmcli connection up bond2
```

[In case of RHEL6]

To create a channel bonding interface, create a file with the name ifcfg-bondN in the directory /etc/sysconfig/network-scripts, and replace N with the interface number. The contents of the channel bonding configuration file are as follows:

[File system port (10GbE)]

/etc/sysconfig/network-scripts/ifcfg-bond0 (newly created)

```
DEVICE=bond0
BOOTPROTO=none
NM_CONTROLLED=yes
ONBOOT=yes
IPV6INIT=no
USERCTL=no
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
```

/etc/sysconfig/network-scripts/ifcfg-bond1 (newly created)

```
DEVICE=bond1
BOOTPROTO=none
NM_CONTROLLED=yes
ONBOOT=yes
IPV6INIT=no
USERCTL=no
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
```

/etc/sysconfig/network-scripts/ifcfg-bond0.12 (newly created)

```
DEVICE=bond0.12
BOOTPROTO=none
ONBOOT=yes
IPV6INIT=no
USERCTL=no
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
VLAN=yes
```

/etc/sysconfig/network-scripts/ifcfg-bond1.14 (newly created)

```
DEVICE=bond1.14
BOOTPROTO=none
ONBOOT=yes
IPV6INIT=no
USERCTL=no
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
VLAN=yes
```

[Port for interconnect between IO servers]

/etc/sysconfig/network-scripts/ifcfg-bond2 (newly created) IO server0

```
DEVICE=bond2
BOOTPROTO=none
NM_CONTROLLED=yes
ONBOOT=yes
IPADDR=10.2.0.10
IPV6INIT=no
USERCTL=no
NETMASK=255.255.255.0
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
```

/etc/sysconfig/network-scripts/ifcfg-bond2 (newly created) IO server1

```
DEVICE=bond2
BOOTPROTO=none
NM_CONTROLLED=yes
ONBOOT=yes
IPADDR=10.2.0.11
IPV6INIT=no
USERCTL=no
NETMASK=255.255.255.0
BONDING_OPTS="mode=802.3ad miimon=100 xmit_hash_policy=layer2+3"
```

Set IPADDR, NETMASK and, regardless of the settings of EXPRESSCLUSTER the case of port for interconnect between IO servers.
After creating a channel bonding configuration file, to set the network interface to be bound, you need to add a MASTER directive and SLAVE directive in the configuration file.
The configuration file of each channel bonding interface may be nearly identical.

[File system port (10GbE)]

/etc/sysconfig/network-scripts/ifcfg-eth0 (changed)

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:07:43:13:59:E0
NM_CONTROLLED=yes
ONBOOT=yes
TYPE=Ethernet
UUID="a328a3bb-bd19-4b46-ab89-920203554a42"
IPV6INIT=no
USERCTL=no
MASTER=bond0 (Added)
SLAVE=yes (Added)
```

/etc/sysconfig/network-scripts/ifcfg-eth1(changed)

```
DEVICE=eth1
BOOTPROTO=none
HWADDR=00:07:43:13:59:E8
NM_CONTROLLED=yes
ONBOOT=yes
TYPE=Ethernet
UUID="77227c15-4565-40c0-8c73-9e04f329ac6b"
IPV6INIT=no
USERCTL=no
MASTER=bond0 (Added)
SLAVE=yes   (Added)
```

/etc/sysconfig/network-scripts/ifcfg-eth2 (changed)

```
DEVICE=eth2
BOOTPROTO=none
HWADDR=00:07:43:13:56:C0
NM_CONTROLLED=yes
```

```
ONBOOT=yes
TYPE=Ethernet
UUID="f90a96a8-8ec6-4003-a22d-cccad74bb6a7"
IPV6INIT=no
USERCTL=no
MASTER=bond1 (Added)
SLAVE=yes (Added)
```

/etc/sysconfig/network-scripts/ifcfg-eth3 (changed)

```
DEVICE=eth3
BOOTPROTO=none
HWADDR=00:07:43:13:56:C8
NM_CONTROLLED=yes
ONBOOT=yes
TYPE=Ethernet
UUID="a7d977b0-2250-42a5-a153-3228ea64d05d"
IPV6INIT=no
USERCTL=no
MASTER=bond1 (Added)
SLAVE=yes (Added)
```

[Port for interconnect between IO servers]

/etc/sysconfig/network-scripts/ifcfg-eth4 (changed)

```
DEVICE=eth4
BOOTPROTO=none
HWADDR=8C:89:A5:5F:3E:A9
NM_CONTROLLED=yes
ONBOOT=yes
TYPE=Ethernet
UUID="61e28110-46fb-4bf6-b308-e2aacf7b11e0"
IPV6INIT=no
USERCTL=no
MASTER=bond2 (Added)
SLAVE=yes (Added)
```

/etc/sysconfig/network-scripts/ifcfg-eth5 (changed)

```
DEVICE=eth5
BOOTPROTO=none
HWADDR=8C:89:A5:5F:3E:AB
NM_CONTROLLED=yes
ONBOOT=yes
TYPE=Ethernet
```

```
UUID="59eab730-1ac2-4593-b988-7c9f83717a17"
IPV6INIT=no
USERCTL=no
MASTER=bond2 (Added)
SLAVE=yes (Added)
```

To enable a channel bonding interface, the kernel module needs to be installed.

To ensure the module is loaded when the channel bonding interface is active, create a new file 'bonding.conf' in the directory /etc/modprobe.d as the root user.

Any file name is possible so long as the file extension is '.conf'.

/etc/modprobe.d/bonding.conf (newly created)

```
alias netdev-bond0 bonding
alias netdev-bond1 bonding
alias netdev-bond2 bonding
```

When all configuration files are prepared, restart the IO server to apply the settings.

Then execute ifconfig and ensure bond0, bond1 and bond2 that have been configured are displayed.

5.1.18.2 Setting of the network interface of file system port for IB

[In case of RHEL7]

To create a channel bonding interface, create a file using nmcli or nmtui command.The nmcli command execution examples for each ports are shown below:

[In case of HCA 1port]

ib0

```
# nmcli connection modify ib0 connection.autoconnect yes
# nmcli connection modify ib0 ipv4.never-default true
# nmcli connection modify ib0 ipv4.method disabled ipv6.method ignore
# nmcli connection up ib0
```

[In case of HCA 2port]

ibbond0

```
# nmcli connection add type bond con-name ibbond0 ifname ibbond0
# nmcli connection add type infiniband autoconnect yes ifname ib0 master ibbond0
# nmcli connection add type infiniband autoconnect yes ifname ib1 master ibbond0
# nmcli connection modify ibbond0 ipv4.never-default true
# nmcli connection modify ibbond0 ipv4.method disabled ipv6.method ignore
# nmcli connection modify ibbond0 802-3-ethernet.mtu 2044
# nmcli connection modify ibbond0 +bond.options mode=active-backup,primary=ib0,miimon=100
# nmcli connection up ibbond0
```

When all configuration files are prepared, restart the IO server to apply the settings. Then execute ip command and ensure network interface that have been configured are displayed.

5.1.18.3 Routing configuration

This configuration is needed only when you use 10GbE.

In case of RHEL7, download and install the following RPM package from the download center.

NetworkManager-dispatcher-routing-rules

An example of setting so that the bond0 (IP address: 10.0.0.10; gateway: 10.0.0.100) outward and return routes match is shown below:

ip rule setting

Description image of /etc/sysconfig/network-scripts/rule-bond0
table 200 from 10.0.0.0/25 ← The table ID is 200.

setting of routing and ip route

/opt/scatefs/script/routeadd.sh

```
#!/bin/sh

# routing add script

## ip route
ip route add table 200 10.0.0.0/25 dev bond0.12 proto kernel src 10.0.0.10
ip route add table 200 default via 10.0.0.100

exit 0
```

routeadd.sh is a script that will run after the floating IP address is set by

EXPRESSCLUSTER.

To check whether the settings have been applied or not, run ip rule and ip route show table table-ID after configuring the IO servers.

5.1.18.4 Setting the DCB

This configuration is needed only when you use DCB-compliant 10GbE.

Enable the DCB priorities in the IO servers by using the vconfig command.

For example, make the following script to assign Priority 4, 5, and 6 to bond0.12 that belongs to VLAN-ID12. Also, set the EXPRESSCLUSTER this script to run at startup EXPRESSCLUSTER.

Newly create the following file:

```
# vi /opt/scatefs/script/dcb.sh
```

Describe the following contents in the file:

[In case of RHEL7]

```
#!/bin/sh

ip link set bond0.12 type vlan egress 4:4
ip link set bond0.12 type vlan egress 5:5
ip link set bond0.12 type vlan egress 6:6
ip link set bond0.12 type vlan ingress 4:4
ip link set bond0.12 type vlan ingress 5:5
ip link set bond0.12 type vlan ingress 6:6

ip link set bond1.14 type vlan egress 4:4
ip link set bond1.14 type vlan egress 5:5
ip link set bond1.14 type vlan egress 6:6
ip link set bond1.14 type vlan ingress 4:4
ip link set bond1.14 type vlan ingress 5:5
ip link set bond1.14 type vlan ingress 6:6

exit 0
```

[In case of RHEL6]

```
#!/bin/sh

vconfig set_egress_map bond0.12 4 4
vconfig set_egress_map bond0.12 5 5
vconfig set_egress_map bond0.12 6 6
vconfig set_ingress_map bond0.12 4 4
vconfig set_ingress_map bond0.12 5 5
vconfig set_ingress_map bond0.12 6 6

vconfig set_egress_map bond1.14 4 4
vconfig set_egress_map bond1.14 5 5
vconfig set_egress_map bond1.14 6 6
vconfig set_ingress_map bond1.14 4 4
vconfig set_ingress_map bond1.14 5 5
vconfig set_ingress_map bond1.14 6 6

exit 0
```

Enable execution of dcb.sh.

```
# chmod +x /opt/scatefs/script/dcb.sh
```

5.1.19 Disabling IPv6

[In case of RHEL7]

Disable ipv6 built-in kernel module.

Edit /etc/default/grub and append ipv6.disable=1 to GRUB_CMDLINE_LINUX like the following sample:

```
GRUB_CMDLINE_LINUX="rd.lvm.lv=rhel/swap crashkernel=auto rd.lvm.lv=rhel/root ipv6.disable=1"
```

Run the grub2-mkconfig command to regenerate the grub.cfg file:

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

Restart the IO server.

[In case of RHEL6]

Add the items indicated by * below to the /etc/sysconfig/network file.

```
NETWORKING=yes
NETWORKING_IPV6=no *Added
HOSTNAME=iosv00
```

Create a file /etc/modprobe.d/ipv6.conf with the following contents:

```
options ipv6 disable=1
```

Disable the ip6tables service from starting at boot by issuing the following command:

```
# chkconfig ip6tables off
```

Then rebuild the Initial RAM Disk Image using:

```
# dracut -f
```

Restart the IO server.

5.1.20 Setting the time

Synchronize IO servers to ensure the time is consistent among all IO servers.

[In case of RHEL7]

Use the chronyd or ntp, ntpdate commands to ensure there is no time differential between servers.

If using ntp, chronyd must be stopped.

For details of time synchronization, see the Red Hat Enterprise Linux Server manual.

[In case of RHEL6]

Use the ntp and ntpdate commands to ensure there is no time differential between servers.

```
# vi /etc/ntp.conf
# chkconfig ntpd on
# vi /etc/ntp/step-tickers
# chkconfig ntpdate on
```

The ntpdate command is optional. Use it in case of a large time differential between servers.

For details of time synchronization, see the Red Hat Enterprise Linux Server manual.

5.1.21 Setting the file system administration (fsadmin) account

Use the fsadmin account for operation and management of the file system on IO servers.

The fsadmin account is created on the IO servers when the scatefs-srv package is installed. Enable remote operation between servers with the fsadmin account.

```
# su - fsadmin
-bash-4.1$ vi .rhosts
-bash-4.1$ chmod 600 .rhosts
```

```
-bash-4.1$ exit
#
```

Specify the IP addresses of all IO servers in the .rhosts file of the fsadmin account. Specify both the operational/management port address and file system port address. With this setting, remote execution between IO servers via fsadmin is possible.

```
# su - fsadmin
-bash-4.1$ rsh iosv01 hostname
iosv01
-bash-4.1$
```

5.1.22 Setting up the internal disk (SSD)

Assign the SSD device (/dev/sdb) of the IO server as the following table:

The device name for the SSD may not be /dev/sdb. In that case, paraphrase /dev/sdb into the actual device name.

device name	mount point	capacity	filesystem	Description
/dev/sdb1	/mnt/ssd	10GB	ext4	journal log area
/dev/sdb2	/mnt/core	remaining area	ext4	dump area

Divide the SSD device (/dev/sdb) into two partitions: one of 10 GB and one of the remaining area.

Use the following commands to execute this process:

```
# parted /dev/sdb
(parted) print
(parted) mkpart primary ext4 0% 10GB
(parted) mkpart primary ext4 10GB 100%
(parted) print
(parted) quit
```

If the device is partitioned correctly, the following will be displayed:

```
Number   Start    End      Size     Type       File system   Flags
 1       1049kB   10.0GB   9999MB   primary
 2       10.0GB   199GB    189GB    primary
```

Create file systems on the partitions and respectively mount /mnt/ssd and /mnt/core.

```
# mkfs.ext4 -E lazy_itable_init /dev/sdb1
# mkfs.ext4 -E lazy_itable_init /dev/sdb2
# mkdir -p /mnt/ssd
# mkdir -p /mnt/core
```

Open /etc/fstab in an editor such as vi and add the following lines with the deivce name or the UUID:

Example:device name

```
/dev/sdb1            /mnt/ssd             ext4     defaults        0 0
/dev/sdb2            /mnt/core            ext4     defaults        0 0
```

Example:UUID

```
UUID=7f879fd4-13ed-4d66-9577-e88e3abc70f8   /mnt/ssd   ext4     defaults        0 0
UUID=f2b97605-5592-46b1-a73c-ec8fe3e473c8 /mnt/core   ext4     defaults        0 0
```

* The UUIDs for the SSD device can be referred by lsblk command.

Mount the two created file systems by using the mount -a command.

5.1.23 Setting the kernel parameter

Add the following to the end of the /etc/sysctl.conf file:

```
# ScaTeFS
vm.dirty_writeback_centisecs = 2
vm.dirty_expire_centisecs = 10
vm.swappiness = 0
net.core.somaxconn = 4000
net.ipv4.ip_local_reserved_ports = 50000-50009
kernel.core_pattern = /mnt/core/core.%e
kernel.core_uses_pid = 0
kernel.unknown_nmi_panic = 1
kernel.panic_on_unrecovered_nmi = 1
```

Run the following command as the root user to apply the above settings:

```
# sysctl -p
```

Note:
The following error messages might be displayed depending on the distribution, but this can be ignored:

```
error: "net.bridge.bridge-nf-call-ip6tables" is an unknown key
error: "net.bridge.bridge-nf-call-iptables" is an unknown key
error: "net.bridge.bridge-nf-call-arptables" is an unknown key
```

5.1.24 Setting syslog log rotation

Configure the syslog settings as follows:

Parameter	Setting Value
Rotation	30times
Timing	weekly
Compression	compress

Add the items indicated by * below to the /etc/logrotate.d/syslog file.

```
/var/log/cron
/var/log/maillog

/var/log/messages

/var/log/secure
/var/log/spooler
{
    rotate 30     *Added
    weekly         *Added
    compress       *Added


    sharedscripts
    postrotate
        /bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null | | true
    endscript
}
```

5.1.25 Setting updatedb.conf file

[In case of RHEL6]

Add /mnt/iot to the PRUNEPATHS parameter in /etc/updated.conf as follows.

```
RUNE_BIND_MOUNTS = "yes"
PRUNEFS = "9p afs anon_inodefs auto autofs bdev binfmt_misc cgroup"
PRUNENAMES = ".git .hg .svn"
PRUNEPATHS = "/afs /media /net    /var/tmp /mnt/iot"
```

5.1.26 Integrating as a ScaTeFS (scatefs_addios) IO server

Run the scatefs_addios command so that nodes operate as IO servers.

To run this command, prepare the file defining IP addresses of all IO servers.

The items to be set are shown below:

Table 5-1 Setting items of scatefs_addios

Item	Description	IB	10GbE
ipaddr	IP address of the operational/management port	Required	Required
fipaddr	IP address of the file system port	Required	Required
inipaddr	IP address of the port for interconnect between IO servers	Required	Required
cport	Port number for client connection. Can be omitted when keeping the default value of 50000.	Required	Required
sport	Port number for communication between servers. Can be omitted when keeping the default value of 50001.	Required	Required
cdport	Port number for client connection for data transfer. Specify 50002.	Required	Required
iftypes	The kinds of interfaces specified in “fipaddr”. Specify 1 for 10GbE, 2 for IPoIB. Specify iftypes with same number as “fipaddr” separated by space. iftypes can be omitted, and the default is 10GbE.	Required	-
pciid@hcaport	Port number of HCA for IB communication. Format is pciid@hcaport. For example	Required	-

	0000:83:00.0@1. Using space if specify some parameters. See 6.1.7 for how to check pciid.		

The examples of the definition file are shown below:

- In case using only10GbE as file system port

```
-bash-4.1$ cat datafile1
# Setting up IOS#0
ipaddr    10.0.0.1
fipaddr   10.0.1.1   10.0.1.2
inipaddr 10.2.0.10
cport 50000
sport 50001
cdport 50002

# Setting up IOS#1
ipaddr    10.0.0.2
fipaddr   10.0.1.3   10.0.1.4
inipaddr 10.2.0.11
cport 50000
sport 50001
cdport 50002
```

- In case using only IB as file system port

```
-bash-4.1$ cat datafile1
# Setting up IOS#0
ipaddr    10.0.0.1
fipaddr   10.0.2.1
inipaddr 10.2.0.10
cport 50000
sport 50001
cdport 50002
pciid@hcaport 0000:83:00.0@1
iftypes 2

# Setting up IOS#1
ipaddr    10.0.0.2
fipaddr   10.0.2.2
inipaddr 10.2.0.11
cport 50000
sport 50001
cdport 50002
```

```
pciid@hcaport 0000:83:00.0@1
iftypes 2
```

- In case using both 10GbE and IB as file system port

In the following example, 10.0.1.1 and 10.0.1.2 are IP addresses for 10GbE, and 10.0.2.1 is IP address for IB.

```
-bash-4.1$ cat datafile1
# Setting up IOS#0
ipaddr    10.0.0.1
fipaddr   10.0.1.1   10.0.1.2 10.0.2.1
inipaddr 10.2.0.10
cport 50000
sport 50001
cdport 50002
pciid@hcaport 0000:83:00.0@1
iftypes 1 1 2

# Setting up IOS#1
ipaddr    10.0.0.2
fipaddr   10.0.1.3 10.0.1.4 10.0.2.2
inipaddr 10.2.0.11
cport 50000
sport 50001
cdport 50002
pciid@hcaport 0000:83:00.0@1
iftypes 1 1 2
```

Run the scatefs_addios command with this file specified as the argument.

```
# su - fsadmin
-bash-4.1$ scatefs_addios -f datafile1
```

Run the scatefs_addios command on one IO server. No need to run this command on all IO servers. One execution applies to all IO servers at one time.

Use the scatefs_detail -s command to confirm the IO servers configured.

```
# su - fsadmin
-bash-4.1$ scatefs_detail -s
--------------------------------------------------------------------------
IOSID    MATE     IP[0]              IPCNT    FIP[0]            FIPCNT   IOTCNT   FSCNT
--------------------------------------------------------------------------
     0       1     10.0.0.1              1     10.1.1.1             2         0        0
     1       0     10.0.0.2              1     10.0.1.3             2         0        0
```

```
---------------------------------------------------------------------------
ALL:2    CAPACITY:256
```

5.2 Configuring IO targets

Create IO targets according to the following procedure:

(1) Check the device name of the PathManager device

(2) Check the path status of the PathManager

(3) Partitioning (parted)

(4) Create LVM devices (pvcreate, vgcreate, lvcreate)

(5) Create IO targets (scatefs_addiot)

5.2.1 Check the device name of the PathManager device

Check the device name of the PathManager device which is designed as LVM construction in 5.1.2.

*The device name is used when creating LVM resources (PV, VG).

Check followings in the storage management view of iStorageManager.

- The serial number of the storage connected to IO servers
- LUN

  *LUNs are set in “5.1.6 Assigning the logical disks”.

You can find the device name of the PathManager device from the file /etc/sps.conf on IO servers (iosv00, iosv01). Open /etc/sps.conf by less(1) or view(2) and search the serial number and the LUN got above step. In the following example, the A is the serial number, the B is the LUN and the device name of the PathManager device is “/dev/dda”.

[iosv00]

```
device:/dev/dda
        disk-info:NEC     ,DISK ARRAY      ,0000000942801512,00000
                                            ^^^^^^^^^^^^^^^^ ^^^^^
                                            A                B
        LoadBalance:D2
        path-info:auto Watch:Enable
A:the serial number of the connected disk array
B:the LUN Number
```

5.2.2 Check the path status of the PathManager

Check if the load of the LVM designed in 5.1.2 is distributed to all ports uniformly.

With PathManager, paths that are Status=Active are used uniformly.

Check that load is distributed to all ports of the four logical disks that configure the striping by referring to the spsadmin command output example shown below:

The first number of 4 numbers in “ScsiAddress” is the port number on the IO server side.

The following example is the setting of the data region “lv_data01” for IO server v3 for standard model.

*The LD number is same as the LUN in the following example.

LVM configuration

LV		Storage1	Storage2
iosv00	iosv01		
lv_data01	-	LD2,LD6	LD2,LD6

In the following output example, if striping is configured with four logical disks whose LUN = 2 and 3 against the LVM design, only ports 7 and 8 on the IO server side are used. On the other hand, if striping is configured with four disks whose LUN = 2 and 6 according to the LVM design, all four ports on the IO server side are used uniformly.

```
# spsadmin --lun

+++ LogicalUnit 11:0:0:2 /dev/ddc [Normal] +++
  SerialNumber=0000000J1BN00180, LUN=0x00002
  LoadBalance=LeastSectors
  2: ScsiAddress=7:0:0:2, ScsiDevice=/dev/sde, Priority=1, Status=Active
  102: ScsiAddress=9:0:0:2, ScsiDevice=/dev/sdda, Priority=2, Status=Standby

+++ LogicalUnit 11:0:0:3 /dev/ddd [Normal] +++
  SerialNumber=0000000J1BN00180, LUN=0x00003
  LoadBalance=LeastSectors
  3: ScsiAddress=7:0:0:3, ScsiDevice=/dev/sdf, Priority=1, Status=Active
  103: ScsiAddress=9:0:0:3, ScsiDevice=/dev/sddb, Priority=2, Status=Standby

+++ LogicalUnit 11:0:0:6 /dev/ddg [Normal] +++
  SerialNumber=0000000J1BN00180, LUN=0x00006
  LoadBalance=LeastSectors
  106: ScsiAddress=9:0:0:6, ScsiDevice=/dev/sdde, Priority=1, Status=Active
  6: ScsiAddress=7:0:0:6, ScsiDevice=/dev/sdi, Priority=2, Status=Standby
```

```
+++ LogicalUnit 11:0:0:9 /dev/ddj [Normal] +++
  SerialNumber=0000000J1BN00179, LUN=0x00002
  LoadBalance=LeastSectors <Path thrashing suppressed>
  38: ScsiAddress=8:0:0:2, ScsiDevice=/dev/sdao, Priority=1, Status=Active
  152: ScsiAddress=10:0:0:2, ScsiDevice=/dev/sdey, Priority=2, Status=Standby

+++ LogicalUnit 11:0:0:10 /dev/ddk [Normal] +++
  SerialNumber=0000000J1BN00179, LUN=0x00003
  LoadBalance=LeastSectors <Path thrashing suppressed>
  39: ScsiAddress=8:0:0:3, ScsiDevice=/dev/sdap, Priority=1, Status=Active
  153: ScsiAddress=10:0:0:3, ScsiDevice=/dev/sdez, Priority=2, Status=Standby

+++ LogicalUnit 11:0:0:13 /dev/ddn [Normal] +++
  SerialNumber=0000000J1BN00179, LUN=0x00006
  LoadBalance=LeastSectors
  156: ScsiAddress=10:0:0:6, ScsiDevice=/dev/sdfc, Priority=1, Status=Active
  42: ScsiAddress=8:0:0:6, ScsiDevice=/dev/sdas, Priority=2, Status=Standby
```

5.2.3　Partitioning

Partition the devices recognized by the IO servers according to the LVM design details.

[ScaTeFS metadata region]

Create the following two types of partitions:

(1)　Partition for EXPRESSCLUSTER heartbeat region

Create a partition of about 16 MB at the start of the logical disk.

Example:

```
# parted /dev/dda
GNU Parted 2.1
Using /dev/dda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mklabel gpt
(parted) mkpart primary ext4 0% 16MB
(parted) print
Model: NEC DISK ARRAY (scsi)
Disk /dev/dda: 1700GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Number  Start   End     Size    File system  Name     Flags
1       1049kB  15.7MB  14.7MB               primary
```

(2)　Partition for metadata region

It makes a partition in one logical disk.

Make sure that the partitions have the same capacity.

*Adjust the number of the partitions to the LVM construction designed in 5.1.2.

[ScaTeFS data region]

The logical disk consists of a single partition, so partitioning is not required.

To recognize created partitions, restart the OS of the IO servers connecting the storage devices.

5.2.4　Creating LVM devices

Create LVM devices according to the LVM design.

Use the PathManager device files (/dev and /ddX) to create an LVM device.

* Be careful with the device combinations used for striping.

* Adjust the number of the partitions to the LVM construction designed in 5.1.2.

To recognize the LVM device, restart the OS of the IO servers connecting the storage devices.

After restarting the OS, make sure the created LVM device (device file) exists.

The command execution examples for each IO server models are shown below:

[IO server v1 for standard model]

- ScaTeFS data region

Example: Creating an LVM device with /dev/ddb, /dev/ddj, /dev/ddp, and /dev/ddx

PV

```
# pvcreate /dev/ddb
# pvcreate /dev/ddj
# pvcreate /dev/ddp
# pvcreate /dev/ddx
```

VG

```
# vgcreate vg_data01 /dev/ddb /dev/ddj /dev/ddp /dev/ddx
```

LV

```
# lvcreate -i 4 -I 512 -l 100%free -r none -n lv_data01 vg_data01
```

The -i option specifies the number of striping ways.

The -I option specifies the striping size (512 KB in the above example).

Specifying 100%free for the -l option allows all free spaces to be allocated.

- ScaTeFS metadata region

  Example: Creating an LVM device with /dev/dda2 and /dev/dde2

  PV

```
# pvcreate /dev/dda2
# pvcreate /dev/dde2
```

VG

```
# vgcreate vg_ctrl01 /dev/dda2 /dev/dde2
```

LV

```
# lvcreate -i 2 -I 512 -l 100%free -r none -n lv_ctrl01 vg_ctrl01
```

[IO server v1 for small-scale model]

- ScaTeFS data region

  Example: Creating an LVM device with /dev/ddb and /dev/ddj

  PV

```
# pvcreate /dev/ddb
# pvcreate /dev/ddj
```

VG

```
# vgcreate vg_data01 /dev/ddb /dev/ddj
```

LV

```
# lvcreate -i 2 -I 512 -l 100%free -r none -n lv_data01 vg_data01
```

- ScaTeFS metadata region

  Example: Creating an LVM device with /dev/dda2 and /dev/dde2

  PV

```
# pvcreate /dev/dda2
# pvcreate /dev/dde2
```

VG

```
# vgcreate vg_ctrl01 /dev/dda2 /dev/dde2
```

LV

```
# lvcreate -i 2 -I 512 -l 100%free -r none -n lv_ctrl01 vg_ctrl01
```

[IO server v3 and v4 for standard model]

- ScaTeFS data region

Example: Creating an LVM device with /dev/ddc, /dev/ddj, /dev/ddg, and /dev/ddn

PV

```
# pvcreate /dev/ddc
# pvcreate /dev/ddj
# pvcreate /dev/ddg
# pvcreate /dev/ddn
```

VG

```
# vgcreate vg_data01 /dev/ddc /dev/ddj /dev/ddg /dev/ddn
```

LV

```
# lvcreate -i 4 -I 512 -l 100%free -r none -n lv_data01 vg_data01
```

Example: Creating an LVM device with /dev/ddc

PV

```
# pvcreate /dev/ddc
```

VG

```
# vgcreate vg_data01 /dev/ddc
```

LV

```
# lvcreate -l 100%free -r none -n lv_data01 vg_data01
```

*The LV is created without stripe.

- ScaTeFS metadata region

Example: Creating an LVM device with /dev/dda2

PV

```
# pvcreate /dev/dda2
```

VG

```
# vgcreate vg_ctrl01 /dev/dda2
```

LV

```
# lvcreate -l 100%free -r none -n lv_ctrl01 vg_ctrl01
```

*The LV is created without stripe.

5.2.5　Creating IO targets (scatefs_addiot)

To integrate created LVM logical volumes (LV) into the system as IO targets, run the scatefs_addiot command.
To do so, prepare a file defining IO targets

It is necessary to design which IO server the created LVs will be assigned to as IO targets.
A design example when a data region consisting of 1-TB HDD for IO server v1 for standard model is used is shown below:

ScaTeFS data region
Assign the first three created LVs to iosv00 and the second three to iosv01.
　iosv00
　　lv_data01, lv_data02, lv_data03
　iosv01
　　lv_data04, lv_data05, lv_data06

ScaTeFS metadata region
Assign the first three created LVs to iosv00 and the second three to iosv01.
　iosv00
　　lv_ctrl01, lv_ctrl02, lv_ctrl03
　iosv01
　　lv_ctrl04, lv_ctrl05, lv_ctrl06

```
-bash-4.1$ cat datafile2
iosid 0
data     /dev/vg_data01/lv_data01
ctrl     /dev/vg_ctrl01/lv_ctrl01
data     /dev/vg_data02/lv_data02
ctrl     /dev/vg_ctrl02/lv_ctrl02
data     /dev/vg_data03/lv_data03
ctrl     /dev/vg_ctrl03/lv_ctrl03
iosid 1
data     /dev/vg_data04/lv_data04
ctrl     /dev/vg_ctrl04/lv_ctrl04
data     /dev/vg_data05/lv_data05
ctrl     /dev/vg_ctrl05/lv_ctrl05
data     /dev/vg_data06/lv_data06
ctrl     /dev/vg_ctrl06/lv_ctrl06
```

Meanings of the items described in the file are as follows:

Item	Meaning
iosid	IO server ID (SID). It can be confirmed by scatefs_detail -s.
data	Device name of the IO server data region
ctrl	Device name of the IO server metadata region

Run the scatefs_addiot command with this file specified as the argument.

```
# su - fsadmin
-bash-4.1$ scatefs_addiot -f datafile2
```

Like the scatefs_addios command, run the scatefs_addiot command on one IO server. This command need not be run on all IO servers. One execution applies to all IO servers at one time.

Use the scatefs_detail -t command to confirm the information of IO targets.

```
# su - fsadmin
-bash-4.1$ scatefs_detail -t
----------------------------------------
IOTID    IOS      FS:SG
----------------------------------------
    0      0     none:none
    1      0     none:none
    2      0     none:none
    3      1     none:none
```

```
      4       1       none:none
      5       1       none:none
--------------------------------------------
ALL:6   USED:0   CAPACITY:16384
```

At this point, a local file system is not yet created on the IO targets (LVs for data and metadata).

5.3 Preparation and execution of mkfs

The following procedure is used for ScaTeFS mkfs.

- Creating ScaTeFS (scatefs_mkfs)

5.3.1 Creating ScaTeFS (scatefs_mkfs)

Based on the created IO targets, run the scatefs_mkfs command to create a ScaTeFS file system.

To run this command, prepare the file defining the file system to be created.

```
-bash-4.1$ cat datafile3
name            scatefs00
iotid           0 1 2 3 4 5
```

Meanings of the items described in the file are as follows:

Item	Meaning
name	File system name Specify this when mounting from a client node. A maximum of 31 characters can be specified.
iotid	IO target ID Specify the value which was designed in 5.1.2 . It can be confirmed by scatefs_detail -t.
data_fstype	File system type of data region Specify the value designed in 5.1.1 . data_fstype can be omitted, and the default is ext4.

In this example, create the file system "scatefs00" consisting of six IO targets

Run the scatefs_mkfs command with this file specified as the argument.

```
# su - fsadmin
-bash-4.1$ scatefs_mkfs -f datafile3
```

Run the scatefs_mkfs command on one IO server. No need to run this command on all IO servers. One execution applies to all IO servers at one time.

The scatefs_mkfs command performs mkfs for IO targets on each IO server, and mounts the file system locally. Then, the file system is formatted as ScaTeFS.

Use the scatefs_detail -f command to confirm the file system information.

```
# su - fsadmin
-bash-4.1$ scatefs_detail -f
-----------------------------------------------------------------------------------------------------
FSID   NAME      ROOTIOS IOSCNT   IOTCNT   SGCNT    VERSION
-----------------------------------------------------------------------------------------------------
   0    scatefs00  0            2            6            1    0x00010000
-----------------------------------------------------------------------------------------------------
ALL:1   CAPACITY:32
```

5.3.2　Configuration file of IO server

5.3.2.1 scatefssrv.conf

The scatefssrv.conf file deployed in /etc/scatefs is a configuration file for IO server daemon tuning parameters. Typically, there is no need to deploy this file because ScaTeFS run with the recommended parameter values.

In the scatefssrv.conf file, describe the defined tag **[network]** (for network-related tuning parameters) or **[journal]** (for journal-related tuning parameters) or **[quota]** (for quota-related tuning parameters) or **[iotarget]** (for iotarget-related tuning parameters) to specify the setting values. The default value of a target setting value is used for operation if any of the following conditions is met:

- /etc/scatefs/scatefssrv.conf does not exist
- No tag name is described
- No setting value is specified
- A specified value exceeds the maximum or minimum value

In the case that you change the setting of scatefssrv.conf, use the scatefs_admin command to transfer scatefssrv.conf to each IO server, and then restart the IO server daemon of each IO server. For restart the IO server daemon and scatefs_admin, see Chapter 9.10.

The parameters are shown below: In the regular case, parameters don't need to be set.

Table 5-2 Available Network Setting Values

Setting value	Description	Minimum	Maximum	Default	Remark
RECVTHREADNUM	Number of reception threads for client	1	200	standard model:50 small-scale model:32	
RECVTHREADCNNNUM	Number of monitoring sockets per reception thread for client	10	512	256	
CLIWORKERTHREADNUM	Number of worker threads for client	1	none	standard model:64 small-scale model:32	
SRVWORKERTHREADNUM	Number of worker threads for server	10	none	standard model:192 small-scale model:96	
JNLWORKERTHREADNUM	Number of worker threads for journal	1	none	10	
FAIRPOLICY	Policy of fair share	0	2	0	0:OFF 1:UID mode 2:ClientID mode
IBSOPTIMMWAITON	The mode of optimum waiting for request to improve IO performance. 1:ON 0:OFF	0	1	0	
IBSIOMEMNODE	Specify the NUMA node number from which the memory used for data	0	1	1	

	transfer is allocated. See 9.12.4 for more details.				

Table 5-3 Available Journal Setting Values

Setting value	Description	Minimum	Maximum	Default	Remark
JMODE	A mode of journal	0	3	1	0:OFF 1: Memory 2: Shared disk 3: Memory+SSD
MEMSIZE	Memory size of the log area (MB)	1	64	32	Create a log area for each IO target.
DDLENT	Number of entries of the dirty data list	1000	500000	100000	
DDLINTVAL	Cycle for monitoring dirty data (sec)	1	600	1	
DDLSAVING	Retention time of dirty data (sec)	1	600	1	

Table 5-4 Available QUOTA function Setting Values

Setting value	Description	Minimum	Maximum	Default	Remark
QUOTAMODE	A mode of QUOTA function	0	1	1	0:OFF 1:ON

Table 5-5 Available iotarget Setting Values

Setting value	Description	Minimum	Maximum	Default	Remark
READAHEADSIZE	Readahead size	1048576	2147483647	8388608	
CACHE	Number of maximum entry of following cache ・directory name ・inode	1	none	5242880	
DECACHE	Number of maximum entry of directory entry cache	1	none	524288	
IBSSYNCMODE	The disk sync mode. See	0	1	0	

	9.12.3 for more details. 1: disk sync on write mode 0: disk sync on close mode				

5.4 Setting the EXPRESSCLUSTER

To enable the resources configured thus far to be managed by EXPRESSCLUSTER, it is necessary to configure the relevant EXPRESSCLUSTER settings.

To configure these settings by using WebManager, you will need a work PC that can communicate with the IO servers over a network.

*For how to operate WebManager and set up each resource, see the EXPRESSCLUSTER X for Linux Reference Guide.

5.4.1 Preparations

5.4.1.1 Transferring the cluster configuration information file to the work PC

Transfer the cluster configuration information file created in “5.1.1 Creating EXPRESSCLUSTER cluster configuration information” to the work PC.

5.4.1.2 Checking the network settings of the ports for connecting the IO servers

When configuring the EXPRESSCLUSTER settings, use the ports for connecting the IO servers.

If these ports are not set up, configure the appropriate network settings.

5.4.2 Starting Cluster WebUI and WebManager

[IO server v4+ for standard model or later (Cluster WebUI)]

Open the web browser.

[IO server v1, v3 and v4 for standard model (WebManager)]

Select **Run as administrator** and open the web browser.

Enter the IO server's IP address (for management) and the port number in the browser's address bar.

* If connection fails, specify the IP addresses of the other IO server.

http://10.0.0.1:29003

5.4.3 Importing the cluster configuration information file

[IO server v4+ for standard model or later (Cluster WebUI)]

Start Cluster WebUI.Select Config mode from the drop-down menu in the toolbar.

Click Import to import the cluster configuration information file.

[IO server v1, v3 and v4 for standard model (WebManager)]

When WebManager is started, a confirmation message is displayed.

Click **Import cluster configuration information file** and import the file.

5.4.4 Cluster properties

- Interconnect

  Right-click **cluster** in the tree view and open **Properties**. Double-click the **Interconnect** tab.

  Change the following heartbeat interface setting:

  Priority level 2: Type (DISK)

  Change the name to the name of the device on which the partition for the EXPRESSCLUSTER heartbeat region has been created.

* How to check the name of the device on which the partition for the heartbeat region has been created is described below:

It is necessary to specify the same storage logical disk (LUN) as that specified in the IO servers as the name of the device used for the disk heartbeat.

The serial number of connected disk array used for the disk heartbeat is 000000942801512.

Open the /etc/sps.conf file in the IO servers (iosv00 and iosv01) by using the less or view command and search for the serial number.

[iosv00]

```
device:/dev/dda
        disk-info:NEC       ,DISK ARRAY       ,0000000942801512,00000
                                             ^^^^^^^^^^^^^^^^ ^^^^^
                                             A                B
        LoadBalance:D2
        path-info:auto Watch:Enable
A:Serial number of connected disk array
B:LUN Number
```

[iosv01]

```
device:/dev/ddn
        disk-info:NEC        ,DISK ARRAY        ,0000000942801512,00000
        LoadBalance:D2
        path-info:auto Watch:Enable
```

The retrieved device is the device used for heartbeat.

The heartbeat partition has been created on this device, so change the device setting for the disk heartbeat of EXPRESSCLUSTER.

5.4.5　Apply Settings

[IO server v4+ for standard model or later (Cluster WebUI)]

Click Apply the Configuration File of Cluster WebUI.

[IO server v1, v3 and v4 for standard model (WebManager)]

Open the File menu of WebManager and click Apply Settings to apply the settings.

After the settings have been applied, restart the OS of the IO servers that configure the cluster and run the clpstat command to check the cluster status.

5.5　Configuring IO servers for DDN SFA7990XE

Build IO server on two VMs on the SFA7990XE controller. In the following explanation, two IO servers are described as “iosv00” for VM1 and “iosv01” for VM2. The supported versions of the programs are as follows.

Table 5-11 SFA7990XE Supported distribution, kernel and software versions

Distribution	kernel	MLNX_OFED	EXPRESS CLUSTER
CentOS7.7	3.10.0-1062.el7.x86_64	4.7-1.0.0.1	4.2.0-1

5.5.1　IO targets design

An IO target is a data store fundamental to the ScaTeFS file system. File data written from a client node are distributed to IO servers and then distributed and stored in IO targets of each IO server.

The IO target has a data region for storing data itself and a metadata region for storing the file type, update time, and other data. Multiple IO targets can be created, and the number of data regions and the number of metadata regions are always the same and in pairs.

An example of IO targets configuration for two IO servers is as follows:

Table 5-12 IO targets configuration SFA7990XE data region

data region							
Disk			**Pool**			**recommended filesystem type**	**IO target**
type	**capacity**	**number**	**RAID**	**number**	**LD**		
NLSAS	12TB	168	6(8+PQ)	4	4	xfs	4

* The number of disks and pools is the number per a storage.

Table 5-13 IO targets configuration SFA7990XE metadata region

metadata region							
Disk			**Pool**			**recommended filesystem type**	**IO target**
type	**capacity**	**number**	**RAID**	**number**	**LD**		
SSD	1.92GB	6	6(4+PQ)	1	1	ext4	the number of data regions

* The number of disks and pools is the number per a storage.

5.5.2 LVM design

Design the number of metadata region partitions, the number of data region partitions, the number of striping ways and the order of IO target according to the configuration (pools, logical disks) of SFA7990XE.

The design examples for each IO server models are shown below:

[SFA7990XE]

A design example when a data region consisting of NL-SAS is used is shown below:

- ScaTeFS data region

Create LV without striping.

LVM configuration

Pool	Controller1	Controller2	iosv00			iosv01		
			LV	IOT	order	LV	IOT	order
pool-0	-	VD0	-	-	-	lv_data03	2	1
pool-1	VD1	-	lv_data01	0	1	-	-	-
pool-2	VD2	-	lv_data02	1	2	-	-	-
pool-3	-	VD3	-	-	-	lv_data04	3	2

The order of the IO target of the LVM configuration is described below:

IO server	Order of the target
vm1	0 1
vm2	2 3

Setting item iotid of 5.3.22 Creating ScaTeFS is described below:

Set value the order in line which uses the IO target of iosv00, iosv01.

item	Setting value
iotid	0 1 2 3

- ScaTeFS meta region

Create LV without striping.

LVM configuration

Pool	Controller 1	Controller 2	iosv00		iosv01	
			LV	IOT	LV	IOT
pool-4	-	VD20p2	-	-	lv_ctrl03	2
	VD21p2	-	lv_ctrl01	0	-	-
	VD22p2	-	lv_ctrl02	1	-	-
	-	VD23p2	-	-	lv_ctrl04	3

*X in “LD0-X” means a partition.

5.5.3 Setting the time

Set up as follows on two VMs on the SFA7990XE controller.

(1) Setting the time zone

The following is an example of setting it to Asia/Tokyo.

```
# timedatectl set-timezone Asia/Tokyo
# timedatectl
      Local time: Fri 2020-06-19 14:51:51 JST
  Universal time: Fri 2020-06-19 05:51:51 UTC
        RTC time: Fri 2020-06-19 05:51:50
       Time zone: Asia/Tokyo (JST, +0900)
     NTP enabled: no
NTP synchronized: no
 RTC in local TZ: no
      DST active: n/a
```

(2) Setting the synchronization of time.

d) Edit /etc/chrony.conf

Add the following line. <server-ip-addr> is an IP address of chrony server.

```
server <server-ip-addr> iburst
```

e) Start chronyd

```
# systemctl enable chronyd
# systemctl start chronyd
```

5.5.4 Setting multipath

Set up as follows on two VMs on the SFA7990XE controller.

(1) Install device-mapper-multipath package.

```
# yum install device-mapper-multipath
```

*) Not required if it is already installed.

(2) Create /etc/multipath.conf

```
# mpathconf –enable
# ls -l /etc/multipath.conf
-rw------- 1 root root 2415 Jun 19 02:40 /etc/multipath.conf
```

(3) Start multipathd service

```
# systemctl start multipathd
```

(4) Add WWID to /etc/multipath/wwids

```
# cd /dev/disk/by-id
# ls -1 scsi-* | sed -e "s/scsi-/¥//g" ¥-e "s/$/¥//g" >> /etc/multipath/wwids
# cat /etc/multipath/wwids

# Multipath wwids, Version : 1.0
# NOTE: This file is automatically maintained by multipath and multipathd.
# You should not need to edit this file in normal circumstances.
#
# Valid WWIDs:
/360001ff0d004e000000002a589200000/
/360001ff0d004e000000002a689210001/
/360001ff0d004e000000002a789220002/
/360001ff0d004e000000002a889230003/
/360001ff0d004e000000002a989240014/
/360001ff0d004e000000002aa89250015/
/360001ff0d004e000000002ab89260016/
/360001ff0d004e000000002ac89270017/
```

(5)　Set up alias

Edit /etc/multipath.conf to add alias.

The following is an example of adding alias which name “7990xe_lun00”.

```
multipaths {
        multipath {
                wwid                    360001ff0d004e000000002a589200000
                alias                   7990xe_lun00
        }
}
```

(6)　Device check after service restart

Confirm that the device is displayed, after restart multipathd service.

```
# systemctl restart multipathd
# ls -l /dev/mapper/7990xe_lun*
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun00 -> ../dm-9
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun01 -> ../dm-6
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun02 -> ../dm-8
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun03 -> ../dm-7
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun20 -> ../dm-2
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun21 -> ../dm-5
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun22 -> ../dm-3
lrwxrwxrwx 1 root root 7 Jun 19 05:18 /dev/mapper/7990xe_lun23 -> ../dm-4
```

5.5.5　Installing EXPRESSCLUSTER X for Linux

Refer to 5.1.8 for installing to 2 VMs.

5.5.6 Installing the IB driver

Not required if it is alreay installed.

If not, refer 5.1.10 and install it on 2 VMs.

5.5.7 Installing rsh-related packages

Refer 5.1.11 and install them on 2 VMs.

5.5.8 Installing the ScaTeFS packages

Refer 5.1.12 and install them on 2 VMs.

5.5.9 Registering the ScaTeFS license

Refer 5.1.13 and set up on 2 VMs.

5.5.10 Disabling SELinux

If SELinux is enabled, refer 5.1.14 and disable it on 2VMs.

5.5.11 Disabling Firewalls

If firewall is enabled, refer 5.1.15 and disable it on 2VMs.

5.5.12 Disabling abrtd

If abrtd is enabled, refer 5.1.17 and disable it on 2VMs.

5.5.13 Configuring the network for file system port (IB)

Set up the IB network bonding on 2 VMs as the follows.

(1) Edit ifcfg-ib0 and ifcfg-ib1 file

Add the settings of MASTER and SLAVE as follows.

/etc/sysconfig/network-scripts/ifcfg-1b0 (changed)

```
CONNECTED_MODE=no
TYPE=InfiniBand
NAME=ib0
UUID=4ccd7d05-9b43-4cfc-8467-14827396a027
DEVICE=ib0
ONBOOT=yes
MASTER=ibbond0
SLAVE=yes
```

/etc/sysconfig/network-scripts/ifcfg-1b1 (changed)

```
CONNECTED_MODE=no
TYPE=InfiniBand
NAME=ib1
UUID=cfbc6db7-1464-471c-bff0-2f92a432b5e8
DEVICE=ib1
ONBOOT=yes
MASTER=ibbond0
SLAVE=yes
```

(2) Create bonding file (ifcfg-ibbond0)

Create the file as follows. Set UUID generated by uuidgen command to UUID parameter.

/etc/sysconfig/network-scripts/ ifcfg-ibbond0 (newly created)

```
BONDING_OPTS="miimon=100 mode=active-backup primary=ib0"
TYPE=Bond
BONDING_MASTER=yes
PROXY_METHOD=none
BROWSER_ONLY=no
DEFROUTE=no
IPV4_FAILURE_FATAL=no
IPV6INIT=no
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ibbond0
UUID=5469860b-0057-4bcc-82e4-e0f5cd467d7c
DEVICE=ibbond0
ONBOOT=yes
MTU=2044
```

(3) Restart the network service

Confirm that the state of ibbond0 interface becomes UP, after restart the network service.

```
# systemctl restart network
# ip a show dev ibbond0
7: ibbond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 2044 qdisc noqueue state UP group default qlen
1000
    link/infiniband              20:00:11:07:fe:80:00:00:00:00:00:00:b8:59:9f:03:00:f6:89:b0              brd
00:ff:ff:ff:ff:12:40:1b:ff:ff:00:00:00:00:00:00:ff:ff:ff:ff
    inet6 fe80::ba59:9f03:f6:89b0/64 scope link
```

```
        valid_lft forever preferred_lft forever
```

5.5.14 Disabling IPv6

Refer to 5.1.19 and disable IPv6 on 2 VMs. Note that in case of SFA7990XE, the setting file name is /boot/grub2/grub.cfg, not /boot/efi/EFI/redhat/grub.cfg.

5.5.15 Setting file system administrator (fsadmin) account

Refer to 5.1.21 and set up the account on 2 VMs.

5.5.16 Setting the kernel parameter

Set up the kernel parameter as the following on 2VMs.

(4) Edit /etc/sysctl.conf

Add the following to the end of the /etc/sysctl.conf file.

Set to `kernel.core_pattern` as `/mnt/iot/<IOTID>/data/core.%e` format. For `<IOTID>`, specify the lowest number of IOT which mounted to the VM. The following is an example of specifying 0 IOT.

```
vm.dirty_writeback_centisecs = 2
vm.dirty_expire_centisecs = 10
vm.swappiness = 0
net.core.somaxconn = 4000
net.ipv4.ip_local_reserved_ports = 50000-50009
kernel.core_pattern = /mnt/iot/0/data/core.%e
kernel.core_uses_pid = 0
kernel.unknown_nmi_panic = 1
kernel.panic_on_unrecovered_nmi = 1
kernel.panic = 10
vm.min_free_kbytes = 4194304
```

(5) Modify kernel parameter

```
# sysctl -p
```

5.5.17 Setting syslog log lotation

Refer to 5.1.24 and set up log lotation on 2 VMs.

5.5.18 Integrating as a ScaTeFS (scatefs_addios) IO server

Refer to 5.1.26 and set it up.

5.5.19 Partitioning

For metadata device, create an EXPRESSCLUSTER hartbeat region partition (16MB), and some metadata region partitions. The following is an example. Apply for all 4 metadeta devices.

```
# parted /dev/mapper/7990xe_lun20
GNU Parted 3.1
Using /dev/mapper/7990xe_lun20
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mklabel gpt
(parted) unit MB
(parted) mkpart primary ext4 0% 16MB
(parted) mkpart primary ext4 16MB   100%
(parted) print
Model: Linux device-mapper (multipath) (dm)
Disk /dev/mapper/7990xe_lun20: 927713MB
Sector size (logical/physical): 4096B/4096B
Partition Table: gpt
Disk Flags:

Number   Start     End        Size       File system   Name      Flags
 1       1.05MB    15.7MB     14.7MB                     primary
 2       15.7MB    927712MB   927696MB                    primary
```

5.5.20 Creating LVM devices

Create LVM devices according to the LVM design, with multipath device files (/dev/mapper/7990xe_lunXX).

To recognize the LVM device, restart the OS of the IO server VMs. Confirm that the created LVM devices exist after restarting the OS.

The command execution examples for IO server models are shown below:

【DDN SFA7990XE】

- ScaTeFS data region

PV

```
# pvcreate /dev/mapper/7990xe_lun00
# pvcreate /dev/mapper/7990xe_lun01
# pvcreate /dev/mapper/7990xe_lun02
# pvcreate /dev/mapper/7990xe_lun03
```

VG

```
# vgcreate vg_data01 /dev/mapper/7990xe_lun01
# vgcreate vg_data02 /dev/mapper/7990xe_lun02
# vgcreate vg_data03 /dev/mapper/7990xe_lun00
# vgcreate vg_data04 /dev/mapper/7990xe_lun03
```

LV

```
# lvcreate -l 100%free -r none -n lv_data01 vg_data01
# lvcreate -l 100%free -r none -n lv_data02 vg_data02
# lvcreate -l 100%free -r none -n lv_data03 vg_data03
# lvcreate -l 100%free -r none -n lv_data04 vg_data04
```

- ScaTeFS metadata region

PV

```
# pvcreate /dev/mapper/7990xe_lun20p2
# pvcreate /dev/mapper/7990xe_lun21p2
# pvcreate /dev/mapper/7990xe_lun22p2
# pvcreate /dev/mapper/7990xe_lun23p2
```

VG

```
# vgcreate vg_ctrl01 /dev/mapper/7990xe_lun21p2
# vgcreate vg_ctrl02 /dev/mapper/7990xe_lun22p2
# vgcreate vg_ctrl03 /dev/mapper/7990xe_lun20p2
# vgcreate vg_ctrl04 /dev/mapper/7990xe_lun23p2
```

LV

```
# lvcreate -l 100%free -r none -n lv_ctrl01 vg_ctrl01
# lvcreate -l 100%free -r none -n lv_ctrl02 vg_ctrl02
# lvcreate -l 100%free -r none -n lv_ctrl03 vg_ctrl03
# lvcreate -l 100%free -r none -n lv_ctrl04 vg_ctrl04
```

5.5.21 Creating IO targets (scatefs_addiot)

Refer to 5.2.5 and create IO targets.

5.5.22 Creating ScaTeFS (scatefs_mkfs)

Refer to 5.3.1 and create filesystem.

5.5.23 Setting the EXPRESSCLUSTER

Refer 5.4 and set it up.

Chapter6　Setting the Linux client

6.1 Using InfiniBand

6.1.1 Installing the InfiniBand driver

On SX-Aurora TSUBASA and Linux machines other than SX-Aurora TSUBASA, there are differences in IB driver that can be installed.

When using SX-Aurora TSUBASA
Install MLNX_OFED which is provided from NVIDIA.

When using linux machine not SX Aurora TSUBASA
Install the IB driver provided from OS distribution or MLNX_OFED provided from NVIDIA. The difference of IB driver does not affect the function of ScaTeFS. Please select IB driver for user environment, such as user applications.

The following describes how to install MLNX_OFED.

(*) Please refer to the Red Hat Enterprise Linux Server manuals for how to install InfiniBand driver which is provided from OS distribution.

How to install MLNX_OFED

(1) Get MLNX_OFED package

MLNX_OFED versions supported by ScaTeFS/Client are shown in the following table.

OS	MLNX_OFED version
RHEL/CentOS 7.3	3.4-2.1.9.0.1
RHEL/CentOS 7.4	4.2-1.2.0.0
RHEL/CentOS 7.5	4.3-3.0.2.1
RHEL/CentOS 7.6	4.6-4.1.2.0
	4.7-1.0.0.1
RHEL/CentOS 7.7	4.7-1.0.0.1
RHEL/CentOS 7.8	4.9-0.1.7.0
RHEL/CentOS 7.9	4.9-2.2.4.0
RHEL/CentOS 8.1	4.7-3.2.9.0
RHEL/CentOS 8.2	4.9-0.1.7.0

RHEL/CentOS 8.3	4.9-3.1.5.0
RHEL/CentOS 8.4	4.9-4.0.8.0
	5.5-1.0.3.2
RHEL/Rocky Linux 8.5	5.5-1.0.3.2 When using RHEL 8.5 and kernel version 4.18.0-348.12.2.el8_5.x86_64, use this version. If you use Rocky Linux 8.5 or newer kernel version, use 5.6-1.0.3.3 below.
	5.6-1.0.3.3
RHEL/Rocky Linux 8.6	5.6-2.0.9.0 When using kernel version 4.18.0-372.26.1.el8_6.x86_64, use this version. If you use newer kernel version, use 5.8-1.1.2.1 below.
	5.8-1.1.2.1
RHEL/Rocky Linux 8.8	23.04-1.1.3.0
RHEL/Rocky Linux 8.10	23.10-3.2.2.0

(*) MLNX_OFED versions supported by ScaTeFS/Client are the same as MLNX_OFED versions supported by SX-Aurora TSUBASA InfiniBand.

Please download the applicable MLNX_OFED from the official site of NVIDIA.
https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/

(*) MLNX_OFED 4.6-4.1.2.0 is not published at the above URL. Please download from the following URL:
https://mellanox.my.salesforce.com/sfc/p/#500000007heg/a/1T000000cCrw/jAKX3brAtwtWng6sVqHpSXf2pT8UrSUL2rMKpn3c4ng
Password: mgIdJQfI

If you cannot download MLNX_OFED, please contact the NEC support department.

(2) When the following packages are not installed, install them from the OS distribution.

lsof gtk2 atk cairo tcl tcsh tk pciutils

```
# yum install lsof gtk2 atk cairo tcl tcsh tk pciutils
```

(3) Mount the ISO file on a directory. It is mounted on /mnt/iso in the following example.

```
# mount -t iso9660 -o loop MLNX_OFED_LINUX-xxxx-x86_64.iso /mnt/iso
```

(4) Execute the install script.

```
# /mnt/iso/mlnxofedinstall
```

⚠ **Notice**

When the kernel has been updated, installation by mlnxofedinstall may fail. In this case, execute the install script with “--add-kernel-support” and “--kmp” option as below:

```
 # /mnt/iso/mlnxofedinstall --add-kernel-support --kmp
```

(5) You will be asked if you delete the old IB related packages and continue to install, input “y”.

```
This program will install the MLNX_OFED_LINUX package on your machine.
Note that all other Mellanox, OEM, OFED, or Distribution IB packages will be removed.
Do you want to continue?[y/N]:y
```

(6) Wait for the installation to complete. When installation is completed, go to (7).

(7) When the client OS is RHEL/CentOS 7.4 or 7.3 and using both of MLNX_OFED and 10GbE, following configuration is required. If the conditions are not met, go to (8).

Add "#" at the beginning of cxgb4 line to comment out in /etc/depmod.d/*-mlnx-ofa_kernel.conf as follows.

```
# vi /etc/depmod.d/zz01-mlnx-ofa_kernel.conf
-------------------------
#override iw_cxgb4 * weak-updates/mlnx-ofa_kernel/drivers/infiniband/hw/cxgb4
-------------------------
```

Rebuild module dependencies.

```
# depmod -a
```

(8) Restart the machine.

```
# reboot
```

(9) After rebooting, confirm that the information of HCA can be referred.

```
# ibstat
CA 'mlx5_0'
        CA type: XXXXXX
        Number of ports: 1
        Firmware version: XXXXXXXXXX
        Hardware version: 0
        Node GUID: 0xXXXXXXXXXXXXXXXX
        System image GUID: 0xXXXXXXXXXXXXXXXX
        Port 1:
                State: Active
                Physical state: LinkUp
                Rate: XXX
                Base lid: XXX
                LMC: X
                SM lid: X
                Capability mask: 0xXXXXXXXX
                Port GUID: 0xXXXXXXXXXXXXXXXX
                Link layer: InfiniBand
```

The installed MLNX_OFED version can be checked by the following command.

```
# /usr/bin/ofed_info –s
MLNX_OFED_LINUX-4.9-0.1.7.0:
```

The installation of MLNX_OFED is complete.

⚠ Notice about IB driver

- The IB driver installation must be done before the ScaTeFS/Client installation.
- Please uninstall ScaTeFS/Client before uninstalling or reinstalling MLNX_OFED. If MLNX_OFED is uninstalled with ScaTeFS/Client installed, ScaTeFS/Client service cannot start after reinstalling MLNX_OFED. In this case, reinstall ScaTeFS/Client

after reinstalling MLNX_OFED.

6.1.2　Communication confirmation

You can check if the client can communicate with IO servers through IB by the following methods.

Check if the client can communicate with IO servers on IB network

Check it by ibping command as the following steps.

ibping command can be executed as client or server. Run ibping on both the client and the IO servers like below.

(1) Run it as server on the IO server

```
# ibping -S
```

(2) Run it as client on the client

```
# ibping -L LID
```

LID is the LID of the target port on the IO server. You can get LID from “Base lid” in the output of ibstat command executed on the IO server.

Check if the client can communicate with IO servers on IP network on IB

On Linux, ping command can specify the source interface by -I option. By specifying interface used by ScaTeFS client, make sure ScaTeFS client can communicate with each interface on IO servers.

```
# ping ServerAddress -I ibbondN
```

ServerAddress is the IPoIB address of the IO server.

6.1.3　Install and Update the packages

There are 2 types of install and update the packages method, depending on machine type of linux client.

- When using SX-Aurora TSUBASA
  See the SX-Aurora TSUBASA Installation Guide.

- When using linux machine not SX Aurora TSUBASA

  See the Installation Guide (Installation_Guide_for_Scalar_E.txt) bundled in the packages.

  * When using the SX Cross Software Node-lock License, see the release memo bundled in the packages.

6.1.4 Registering the ScaTeFS license

Register the license.

See the HPC Software License Management Guide about details of the procedure.

* When using the SX Cross Software Node-lock License, see the SX Cross Software Node-lock License Installation Guide instead of the HPC Software License Management Guide.

6.1.5 Setting of ScaTeFS InfiniBand high performance Library

Following description is how to setup to use ScaTeFS IB Library. Refer to 9.12 ScaTeFS InfiniBand high performance library for more information.

- Setting the max memory size which a process can lock

  To process IOs efficiently, ScaTeFS IB Library locks the memory area which is specified as arguments of read(2)/write(2) in a user program. The more ScaTeFS IB Library can lock the memory for IOs, the more it can process them efficiently and you can expect a performance improvement. Because the default is usually tens KB (“max locked memory” in the output of “ulimit –a”), a program doing big IOs (MB order) cannot process them efficiently.

  Therefore, extend the max locked memory size by setting “memlock” in /etc/security/limits.conf. It is recommended more than 100MB.

  Because the locked memory is not unlocked until the process ends, a memory shortage might occur earlier than a usual case when many processes lock the max memory. So it's recommended that the multiplied value of “the max processes executed simultaneously using ScaTeFS IB Library” and “the max locked memory (memlock)” does not exceed the half amount of the total memory size in the client machine.

The following is the example which the max locked memory size is set unlimited for every users.

Example of /etc/security/limits.conf

```
* soft memlock unlimited
* hard memlock unlimited
```

HCA port which is used by ScaTeFS IB Library.

ScaTeFS IB Library automatically detects HCA port from the label which is specified to ibdev mount option and use the HCA port for IO. If multiple HCA port is specified in one label, the path will be changed automatically on fault of the path. Please refer 6.1.6 for the label which is specified to ibdev mount option.

6.1.6　Mounting

Mount the file system by mount command.

Following is an example to mount file system “scatefs00” on /mnt/scatefs.

```
# mount -t scatefs -o ibdev=LABEL,rsize=4194304,wsize=4194304 ServerAddress:scatefs00 /mnt/scatefs
```

IPv4 address of IPoIB on the Root IO Server is specified to ServerAddress.

The mount option "rsize" and "wsize" are transfer size for file data between client and IO server. Default value is 1MB for both. Specifying 2MB or 4MB will improve performance.

Please specify "_netdev" to the mount option, if information about the file system are described in /etc/fstab to mount the file system while booting the Linux machine. Following is an example of description in /etc/fstab.

```
ServerAddress:scatefs00 /mnt/scatefs scatefs _netdev,ibdev=HOME,rsize=4194304,wsize=4194304 0 0
```

- If "_netdev" option is missed on RHEL/CentOS 7 or 8, emergency prompt will be displayed on the console. In this case, please add "_netdev" to the mount option and reboot the machine. Following is an example of description in /etc/fstab.
- If SELinux context is not specified to the mount option, the value context="system_u:object_r:nfs_t:s0" will be used by default. Please specify the

context to the mount option to use other context.

Followings are mount options which are available only on IB environment.

- ibdev=LABEL

  The mount option which specifies HCA device and port for IB Verbs communication. Please specify the label which is defined in the configuration file. Please refer 6.1.7 for the configuration file.

  (*) If this option is omitted, IPoIB will be used for communication, and communication becomes slower than in case of IB Verbs. This option must be specified.

- mpri=N

  The mount option which specifies the priority of metadata access. Service level (0 to 14) can be specified to this option. If this option is omitted, mpri=0 will be used by default. To enable this option, QoS configuration is required on the subnet manager.

- dpri=N

  The mount option which specifies the priority of READ/WRITE. Service level (0 to 14) can be specified to this option. If this option is omitted, dpri=0 will be used by default. To enable this option, QoS configuration is required on the subnet manager.

  (*) Please refer manuals of the subnet manager for QoS configuration.

If ibdev=LABEL is specified, mount command resolves HCA device name from the LABEL which is defined in the configuration file.If the definition is wrong or not exists, the mount command will be fail.

Following is an example of mount command without arguments after mounted ScaTeFS. If mount options (ibhcaN) in red characters are displayed, the communication will use IB verbs.

```
# mount
(...)
172.28.71.1:scatefs00 on /mnt/scatefs type scatefs
(rw,relatime,hard,cto,ac,sync_on_close,ibhca1=mlx5_0:1,mpri=0,dpri=0,port=50000,rsize=4194304,wsize=4194
304,timeo=600,retrans=1,acregmin=3,acregmax=60,acdirmin=30,acdirmax=60,addr=172.28.71.1)
```

When IB verbs is effective, InfiniBand native protocol is used for data I/O. These communications are not using sockes, so connections to port number 50002 for data I/O will not be created.

```
# dd if=/dev/zero of=/mnt/scatefs/testfile bs=1M count=1
# ss -n    | grep 50002
# (not exists)
```

Note that even when IB verbs is effective, connections to port number 50000 will be created. These connections are used for control communication on IPoIB.

If ibdev=LABEL is not specified on mount, all communications are issued on IPoIB. In this case, ibhcaN, mpri and dpri options will not be displayed in the result of mount command.

```
# mount
(...)
172.28.71.1:scatefs00 on /mnt/scatefs type scatefs
(rw,relatime,hard,cto,ac,sync_on_close,port=50000,rsize=4194304,wsize=4194304,timeo=600,retrans=5,acregm
in=3,acregmax=60,acdirmin=30,acdirmax=60,addr=172.28.71.1)
```

When IB verbs is ineffective, connections to port number 50002 will be created. In this case, please check the mount option.

```
# dd if=/dev/zero of=/mnt/scatefs/testfile bs=1M count=1
# ss -n    | grep 50002
tcp     ESTAB      0       0       172.28.7.1:963             172.28.71.1:50002
tcp     ESTAB      0       0       172.28.7.1:963             172.28.71.2:50002
```

Please refer the scatefs(5) man data for details about mount options.

6.1.7　Setting the HCA device in the client

To use IB Verbs, specify the label which stands for HCA device and port to ibdev in mount option. The label is defined in configuration file "/etc/scatefs/client/ibdevice.conf".
Specify HCA device by PCI-ID as follows.

Image of /etc/scatefs/client/ibdevice.conf

```
#
# /etc/scatefs/client/ibdevice.conf
#
# This is the configuration file for ScaTeFS mount option 'ibdev'.
#
HOME1      0000:83:00.0   0000:83:00.1 … define label HOME1 with two PCI devices
HOME2      0000:83:00.1   0000:83:00.0 … define label HOME2 with two PCI devices

WORK1      0000:83:00.0                … define label WORK1 with one PCI device

# add "@port-number" at the tail of PCI-ID to specify port number of HCA
# (If omitted, "@1" will be used)
WORK2      0000:83:00.1@1
```

The format in configuration file is as follows

- One label can be defined for each line. To define multiple labels, file systems can be mounted with different configuration on each mount points.
- The maximum number of PCI-IDs in one line is 12. Space or tab can be used as separators.
- Characters can be used in label are alphabets, numbers, and underscore (_). The maximum length of label is 32.
- The line which starts with # is treated as a comment and ignored.

Followings are details of the label HOME1 in the example. HOME1 is defined with two PCI-IDs, 0000:83:00.0 and 0000:83:00.1. The mapping HCA device to PCI-ID can be confirmed as follows.

```
# ls -l /sys/class/infiniband/
total 0
(...) mlx5_0 -> ../../devices/pci0000:80/0000:80:02.0/0000:83:00.0/infiniband/mlx5_0
(...) mlx5_1 -> ../../devices/pci0000:80/0000:80:02.0/0000:83:00.1/infiniband/mlx5_1
```

From above, HOME1 means two HCA devices mlx5_0 and mlx5_1 are used for communication (multi path communication).

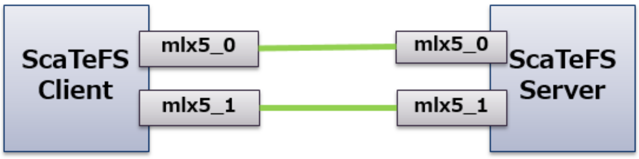
Followings are details of the label HOME2. This label is defined with same PCI-IDs as

HOME1, with the reverse order. Note that this definition is not same as HOME1, and the communication path is different from HOME1. The behavior on path fault will be different, and the performance will be affected by using longer path. Followings are details of the order of labels.

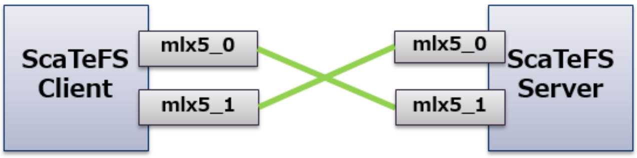
HCA devices which are used on the IO server are defined the configuration file on the IO server. The first HCA device which is defined on the IO server and the first HCA device which is defined on the client are used as endpoints of one communication path. Similarly, the second HCA device which is defined on the IO server and the second HCA device which is defined on the client are used as endpoints of another communication path.

For example, when mlx5_0 and mlx5_1 are defined as HCA devices in this order on the IO server, communication paths which are used by HOME1 and HOME2 are as follows.

[When the file system is mounted by HOME1 configuration]



[When the file system is mounted by HOME2 configuration]



Both of configurations will work, but the communication path will be changed by the order of PCI-IDs as above. For the optimal configuration, consider the configuration of HCA device on the IO server and the network environment.

6.1.8 Number of HCAs and number of connections

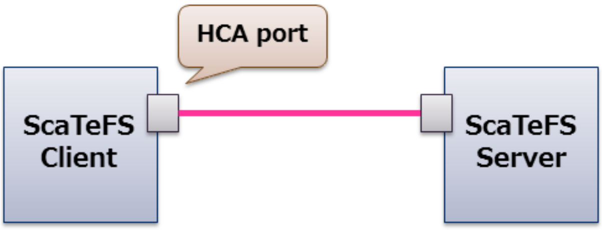
When the client has two port and the IO server has two port, there are 4 patterns for IB Verbs communication path. ScaTeFS/Client does not establish connections for all of them. The number of connections is the maximum of the number of HCAs which are defined on the client and on the IO server. From this, all of HCAs defined in the configuration file will be used, and the number of connections will be reduced.

Followings are connections in various configurations.

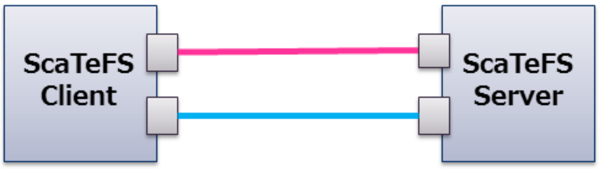
(*) The connection stands for the communication path of InfiniBand QP.

(*) In following figures, all of HCA ports on the clients are used. It is possible to leave some HCAs unused.

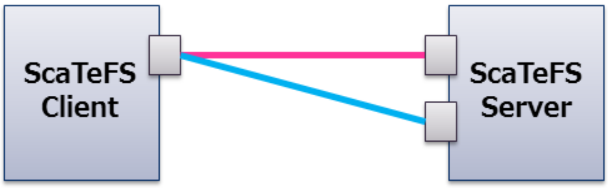
(1)　One port on the client, one port on the IO server



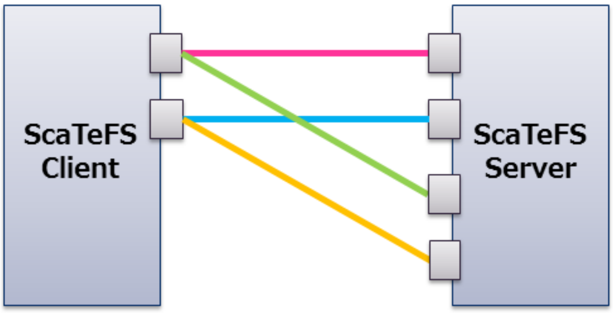
(2)　Two ports on the client, two ports on the IO server



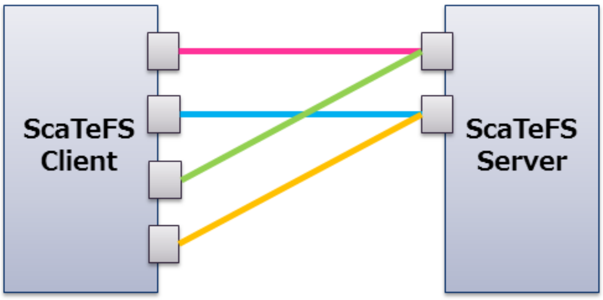
(3)　One port on the client, two ports on the IO server



(4) Two ports on the client, four ports on the IO server



(5) Four ports on the client, two ports on the IO server



6.1.9 Unmounting

Use the **umount** command to unmount the file system.

In the example below, unmount the file system mounted to /mnt/scatefs.

```
# umount /mnt/scatefs
```

In case communication with IO servers is interrupted, you can forcibly unmount the file system using the -f option. In the example below, the file system mounted to /mnt/scatefs is forcibly unmounted.

```
# umount -f /mnt/scatefs
```

6.1.10 Communication confirmation when using the ScaTeFS IB library

To use ScaTeFS IB library, refer to 11.6 and 11.6.6 then perform IO with ScaTeFS IB library, and confirm IO is executed with ScaTeFS IB library from the statistic information

6.2 Using 10GbE

6.2.1 Installing the DCB-compliant 10GbE-NIC driver

Only when you use DCB-compliant 10GbE, you need this setting.

The RPM binary package provided by the 10GbE-NIC vendor may not support the DCB function as is. Install the 10GbE-NIC driver according to the installation procedure obtained from the support department.

6.2.2　Setting of routing

ScaTeFS client communicates with the both of 10GbE Interface bond0 and bond1 setting by IO server; IO server for small scale uses bond0 only. Therefore, add the static routing for communicating with not only bond0 but also bond1 via 10GbE.
Current setting of routing can be confirmed as follows. For more information about setting the routing, see the RHEL installation guide and other relevant documents.

- Show the routing table by ip command:

```
# ip route
```

- Show the routing table by netstat command:

```
# netstat –r
```

If the routing for ScaTeFS client is incorrect, the following phenomenon may occur.

- Mount command does not return response.
  Check the routing to the Root IO Server.
  The ScaTeFS is successfully mounted, but access to the ScaTeFS is sometimes not respond.
  Part of connection may not be established. Check the routing to bond1 especially.
  On Linux, ping command can specify the source Interface by -I option. By specifying the 10GbE Interface used by ScaTeFS client, make sure ScaTeFS client can communicate with each 10GbE Interface on IO servers.

```
# ping “bond0 IP address of IO Server” -I “Client Interface name”
# ping “bond1 IP address of IO Server” -I “Client Interface name”
```

6.2.3　Install and Update the packages

- When using linux machine not SX Aurora TSUBASA
  See the Installation Guide (Installation_Guide_for_Scalar_E.txt) bundled in the packages.
  * When using the SX Cross Software Node-lock License, see the release memo

bundled in the packages.

6.2.4　Registering the ScaTeFS license

Register the license.

See the HPC Software License Management Guide about details of the procedure.

* When using the SX Cross Software Node-lock License, see the SX Cross Software Node-lock License Installation Guide instead of the HPC Software License Management Guide.

6.2.5　Mounting

Use the mount command to mount the file system.

In the example below, mount the file system "scatefs00" of the root IO server "iosv00" to /mnt/scatefs.

```
# mount -t scatefs -o rsize=4194304,wsize=4194304 iosv00:scatefs00 /mnt/scatefs
```

The rsize and wsize **mount** options indicate the size of the data being transferred when file data I/O occurs between the client and the IO server. The default value is 1 MB, but the performance can be improved by setting a value of 2 MB or 4 MB.

For details of mount options, see the man data of **scatefs**(5).

If information concerning the file system is described in the /etc/fstab file and the file system is automatically mounted when the Linux machine is started, specify _netdev for the mount option. If this option is not specified in RHEL/CentOS 6, the message "can't mount ScaTeFS file system" is output to the console when the Linux machine is started. If this option is not specified in RHEL/CentOS 7 or 8, mount of the file system fails and login prompt of emergency mode is displayed on the console when the Linux machine is started. In this case, add _netdev for the mount option in the /etc/fstab and reboot. The following is an example of description in the /etc/fstab.

```
iosv00:scatefs00 /mnt/scatefs scatefs _netdev,rsize=4194304,wsize=4194304 0 0
```

If the context of SELinux is not specified for the mount option, the default value **context="system_u:object_r:nfs_t:s0"** is used. To use a different context, specify the context as the mount option.

After mounting, perform the IO check to make clear that it is possible to communicate with all IO servers. You able to confirm this to create files in the same directory because ScaTeFS distributes them to each server in a round-robin.

The following is an example of two IO servers environment. Change the mount point and loop count to fit your machine. Notes that loop count must be equal or greater than the number of IO servers.

```
# for N in {1..2}; do dd if=/dev/zero of=/mnt/scatefs/testfile${N} bs=10M count=1; done;
```

Specify the loop number to {1...2}. For example, specify {1...4} to loop four times. Specify the mount point to /mnt/scatefs/.

Check the connection by the following command after the IO is completed.

- Check the connections by ss command:

```
# ss -nt | egrep 'State|:5000'
```

- Check the connections by netstat command:

```
# netstat -n | egrep 'Local|:5000' | sort
```

From displayed local address and foreign/peer address, make sure the assumed Interface have been used. Especially, you have to check the local address is 10GbE Interface.

The following is an example of the set up ScaTeFS with two IO servers which have two bonding Interface each other. Eight connections should be displayed because four connections for an IO server and two IO servers are exist.

```
Proto Recv-Q Send-Q Local Address          Foreign Address       State
tcp          0      0 172.28.134.43:869      172.16.6.5:50000      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.6.5:50002      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.6.6:50000      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.6.6:50002      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.7.5:50000      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.7.5:50002      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.7.6:50000      ESTABLISHED
tcp          0      0 172.28.134.43:869      172.16.7.6:50002      ESTABLISHED
```

ScaTeFS creates two connections for a bonding Interface of IO server; port number 50000 to communicate the meta data, port number 50002 to communicate the data. When IO server is configured to not use data port, only 50000 is create.

Note that all of the connection may not be displayed when a little IO is performed because the connection is created if ScaTeFS client needs to communicate with appropriate IO server.

When using the ScaTeFS IB Library, execute IOs using ScaTeFS IB Library (see 11.6 and 11.6.6) and make sure the IOs are processed by the library.

6.2.6 Unmounting

Use the **umount** command to unmount the file system.

In the example below, unmount the file system mounted to /mnt/scatefs.

```
# umount /mnt/scatefs
```

In case communication with IO servers is interrupted, you can forcibly unmount the file system using the **-f** option. In the example below, the file system mounted to /mnt/scatefs is forcibly unmounted.

```
# umount -f /mnt/scatefs
```

6.3 Additional information

6.3.1 How to export using the NFS server

Using the NFS server on the Linux client, a file system can be exported to the NFS client.

When exporting a file system, an integer which identifies a file system using the fsid option in /etc/exports must be described. The description example of /etc/exports is as follows:

```
/mnt/scatefs *(rw,no_root_squash,mp,fsid=1)
```

There are following notices:

- The NFS version supported is only 3. The protocol supported is only TCP.
- The NFS client supported is only Linux.
- If the NFS client is a Linux machine, specify 3 as the NFS version when mounting the file system on the NFS client. In some Linux distributions (such as in RHEL 6), if the NFS version is not specified. NFS version 4, which is not supported by ScaTeFS, is used by default.

  It is possible to prevent the NFS client from using NFS version 4 by setting the NFS version to 3 in the NFS server. For details, see the RHEL Storage Administration Guide for your OS version.

- When a NFS client locks a file, clients undergo influence of the locking are the other NFS client for the same NFS server and the ScaTeFS client which be the NFS server. The client which directly accesses the file system without NFS does not undergo influence of the locking.

6.3.2 Delayed synchronization at the time of closing file

It is possible to specify whether the client synchronizes the data of the file with the storage of the IO server at the time of closing the file or not by using the sync_on_close mount option (default) or the no_sync_on_close mount option.
If the sync_on_close option (default) is specified, the client sends the data written by an application to the IO sever and synchronizes the data with the storage of the IO server at the time of closing the file. The data preservation is the highest for this mode.
If the no_sync_on_close option is specified, the client sends the data written by an application to the IO sever at the time of closing the file, but does not synchronize the data with the storage of the IO server. The synchronization of the data is asynchronously performed after the file is closed. The data preservation of this mode is lower than sync_on_close, but it is possible to reduce processing time of creating small file whose size is less than tens of KB.

If the no_sync_on_close option is specified, the synchronization with the storage of the IO server is delayed after the file close so that processing time of the file close is reduced. So it is possible to reduce processing time of the tar command, the cp command and so on which creates many small files whose size are less than tens of KB.
But, even if the no_sync_on_close option is specified, the reduction effect of processing time of the application is small or there is no reduction effect in the following cases:

- The size of the data written to the file is big (more than tens of KB).
- The application performs the synchronization using fsync(2), msync(2) and so on.
- The application performs the record locking or the file locking.
- The application repeats opening and closing the same file.

When specifying the no_sync_on_close option, there are following notices:

- When both of the client and the IO server are downed at the same time after the file is closed, the data

6.4 Notice

6.4.1 Removing a file which is opened by a process

On a client, if a file which is opened by a process is removed, the file will not be removed immediately and will be renamed automatically as follows.

format: .scatefsXXX...X (X:alphanumeric character)

example: .scatefs0000000001010764000000ab

This file will be removed automatically when the process closes the file. If someone tries to remove this file manually before automatic removal, that fails with "Device or resource busy" error.

6.4.2 Notice when using DHCP on management network

In use of the SX-Aurora TSUBASA, when the IP address setting of the management network is performed by DHCP, automatic mounting of the ScaTeFS file system at the time of the system start may fail by delaying of IP address setting.

At this time, either one of the following messages appears in syslog.

```
ScaTeFS_LS:400201 ScaTeFS client license process failed. reason=<Failed to connect to IPAddress (port=7300):
Network is unreachable>
ScaTeFS_LS:400201 ScaTeFS client license process failed. reason=<Failed to connect to IPAddress (port=7300):
No route to host>
```

In this case, mount file system by manual operation after system start.

6.4.3 Notice when using the ScaTeFS IB Library

- Specifying the ScaTeFS IB Library and other libraries to LD_PRELOAD simultaneously is not supported

  You cannot use the ScaTeFS IB Library specifying it and other libraries to LD_PRELOAD. Specify only ScaTeFS IB Library when you use it.

- Specifying the ScaTeFS VE direct IB Library and other libraries to VE_LD_PRELOAD simultaneously is not supported

  You cannot use the ScaTeFS VE direct IB Library specifying it and other libraries to VE_LD_PRELOAD. Specify only ScaTeFS IB Library when you use it.

- Calling clone(2) directly is not supported

  The program using ScaTeFS IB Library cannot call clone(2) directly. You should use fork(2) or vfork(2) instead of clone(2).

6.4.4 Notice of double mount (RHEL/CentOS 8.1 or later)

When you mounted the ScaTeFS file system and mounted another file system to the same mount point, first unmount another file system using umount command, and then unmount the ScaTeFS file system using /sbin/umount.scatefs command. If you use umount command to unmount the second ScaTeFS file system, you will fail to unmount.

```
# umount /mnt/scatefs
# /sbin/umount.scatefs /mnt/scatefs
```

6.4.5 Notice when using mlocate package

If the mlocate package is installed, updatedb checks also ScaTeFS paths daily by default. Running updatedb for ScaTeFS on each client, it puts a heavy load on the system. Add "scatefs" to the PRUNEFS parameter in /etc/updatedb.conf as follows to disable the check for ScaTeFS.

```
# rpm -q mlocate
mlocate-XXX.x86_64
# grep PRUNEFS /etc/updatedb.conf
PRUNEFS = "9p afs anon_inodefs auto autofs bdev binfmt_misc cgroup cifs coda configfs cpuset debugfs devpts
ecryptfs exofs fuse fuse.sshfs fusectl gfs gfs2 gpfs hugetlbfs inotifyfs iso9660 jffs2 lustre mqueue ncpfs nfs nfs4
nfsd pipefs proc ramfs rootfs rpc_pipefs securityfs selinuxfs sfs sockfs sysfs tmpfs ubifs udf usbfs ceph fuse.ceph
scatefs"
```

Chapter7　Setting the SX-ACE Client

7.1 Setting of routing

ScaTeFS client communicates with the both of 10GbE Interface bond0 and bond1 setting by IO server; IO server for small scale uses bond0 only. Therefore, add the static routing for communicating with not only bond0 but also bond1 via 10GbE.

Current setting of routing can be confirmed as follows. Refer to the SUPER-UX Network Administrator's Guide for more information about setting of routing.

```
# netstat –r
```

If the routing for ScaTeFS client is incorrect, the following phenomenon may occur.

- Mount command does not return response.
  Check the routing to the Root IO Server.
- The ScaTeFS is successfully mounted, but access to the ScaTeFS is sometimes not respond.
  Part of connection may not be established. Check the routing to bond1 especially.
- The following message is displayed on console.
  An error occurred because it was trying to communicate via the Non-Offloaded Interface; i.e. en0.
  Correct the routing to IO server "XX.XX.XX.XX" to communicate via Offloaded Interface; i.e. ex0.

```
WARNING:ScaTeFS: RPC: connect error 151 server XX.XX.XX.XX:5000X: Not offloaded connection
```

7.2 License

The lock release code of a ScaTeFS client needs to be applied to the corresponding node, and the package “NEC Scalable Technology File System/Client” needs to be installed.

7.3 Config variables

When using ScaTeFS, specify 1 for the config variable SCATEFS and specify the capacity of the region used for the data cache for the config variable SCFS_DCACHE. The value of SCFS_DCACHE indicates the percentage (%) of region allocated from the XM cache region. The XM cache region is specified by CACHEDEV in the ISL parameter file at installation.

When not using ScaTeFS, specify 0 for the config variable SCATEFS. In this case, SCFS_DCACHE is disabled.

7.4 ScaTeFS daemon

The ScaTeFS daemon scatefs_rpcd(1M) must be running on the client.

The number of daemons is equal to the number of I/O requests that can be issued simultaneously to IO servers. To change the number of daemons to be started, change the argument passed to scatefs_rpcd(1M) in /etc/init.d/scatefs. The default value of the number of daemons is 4.

7.5 ScaTeFS path monitoring daemon

The ScaTeFS path monitoring daemon scatefs_pmond(1M) is a daemon for regularly monitoring the status of the network path between the ScaTeFS Client and an IO server when a failure has occurred on that path. Once the daemon detects that the path is recovered, the ScaTeFS Client can begin communicating again on the recovered path.

When the multi-user mode is activated, the path monitoring daemon is started.

7.6 Mounting

The command line image of mount(1M) below is to mount a file system named “fs1” with the transfer size of 4 MB and signal interruption available. The mount is done only for the root IO server.

```
# mount -t scatefs -o intr,rsize=4194304,wsize=4194304 rootsrv:scatefs00 /mnt/scatefs
```

scatefs :　File system type (fixed)

intr:　Signal interruption is available

rsize,wsize:　Transfer size. The default is 1 MB, but 2 MB or 4 MB is more efficient.

rootsrv:　Host name of the root IO server (or IP address of the root IO server).

scatefs00:　File system name. This is a string specified when executing mkfs.

/mnt/scatefs:　A directory on the client to which ScaTeFS is mounted.

For details of mount options, see mount(1M) of SUPER-UX.

After mounting, perform the IO check to make clear that it is possible to communicate with all IO servers. You able to confirm this to create files in the same directory because ScaTeFS distributes them to each server in a round-robin.

The following is an example of two IO servers environment. Change the mount point and loop count to fit your machine. Notes that loop count must be equal or greater than the number of IO servers.

```
# for N in {1..2}; do dd if=dummyfile of=/mnt/scatefs/testfile${N} bs=10240k count=1; done;
```

Specify the loop number to {1..2}. For example, specify {1..4} to loop four times.
Specify the mount point to /mnt/scatefs/.
Specify a dummy file to dummyfile which size is around 10MB.

Check the connection by the following command after the IO is completed.

```
# netstat -n | egrep 'Local|:5000' | sort
```

From displayed "Local Address" and "Foreign Address", make sure the assumed Interface have been used. Especially, you have to check the "Local Address" is 10GbE Interface; i.e. the Offloaded Interface ex0.
The following is an example of the set up ScaTeFS with two IO servers which have two bonding Interface each other. Eight connections should be displayed because four connections for an IO server, bond0 is 50000, bond1 is 50002.

```
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp          0      0 172.28.134.43:869    172.16.6.5:50000       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.6.5:50002       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.6.6:50000       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.6.6:50002       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.7.5:50000       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.7.5:50002       ESTABLISHED
tcp          0      0 172.28.134.43:869    172.16.7.6:50000       ESTABLISHED
tcp        0      0 172.28.134.43:869      172.16.7.6:50002      ESTABLISHED
```

ScaTeFS creates two connections for a bonding Interface of IO server; port number 50000 to communicate the meta data, port number 50002 to communicate the data. When IO server is configured to not use data port, only 50000 is create.

Note that all of the connection may not be displayed when a little IO is performed because the connection is created if ScaTeFS client needs to communicate with appropriate IO server.

7.7 Data cache

Mount options for data cache are as follows:

- sync, async

  Use these options to specify whether use of the data cache is enabled or disabled.

  If sync is specified, data is synchronously written to and read from the IO servers without using the data cache.

  If async is specified, the data cache is used, enabling high-speed I/O. Note, however, that if the I/O size is 1 MB or more, processing is the same whether the mount option is async or sync.

  The default is async.

- csize

  This option is used to specify the threshold value at which requests are immediately sent to the IO server when using the data cache. This option is available only for write-related system calls whose I/O size is less than 1 MB and when the data cache is enabled. The default value of csize is 1 MB.

  The processing differs as follows according to the I/O size and the setting of csize.

  csize ≦ I/O size ＜ 1MB: As soon as data enters the data cache, it is sent to the IO server and a completion notice is returned to the client even if all the data has not been written to the disk.

  I/O size ＜ csize ＜ 1MB: Immediately write back after the data enters the data cache.

  For details of mount options, see mount(1M) of SUPER-UX.

7.8 Configuration file

The configuration file (/etc/fstab, /etc/scatefs/client.conf) needs to be distributed to each node. For how to distribute the configuration file to each node, see the Installation guide for SUPER-UX R21.1 or later.

/etc/scatefs/client.conf is a configuration file reserved for future functional expansion, and need not be modified.

7.9 Unmounting

Use the **umount**(1M) command to unmount the file system.

In the example below, unmount the file system mounted to /mnt/scatefs.

```
# umount /mnt/scatefs
```

Chapter8　Setting to use ScaTeFS on a Docker’s container

8.1　Setting the configuration file for ScaTeFS

This setting is needed only when using SX-Aurora TSUBASA.

Create the configuration file (/etc/scatefs/client/libscatefsib.conf) on all clients where a docker image runs. And write the following description in it.

```
RDMA_FROM_VH_ON 0
```

8.2　Setting an image for a container

The ScaTeFS package group for container needs to be installed to an image. The package group is different between SX-Aurora TSUBASA and a Linux machine except SX-Aurora TSUBASA. Add the following line to a Dockerfile for each case and build an image.

- In case of SX-Aurora TSUBASA

```
RUN     yum -y group install scatefs-client-tsubasa-container
```

*Add the above line after the installation of the package group “ve-container-infiniband” in a Dockerfile.

- In case of a Linux machine except SX-Aurora TSUBASA

```
RUN     yum -y group install scatefs-client-scalar-container
```

Refer to “NEC Network Queuing System V (NQSV) User's Guide [Management]” about the other settings for an image.

8.3　Setting the script to run a container

Add the following options to docker-run’s options in the script to run a container.

Option	Description
-v /*scatefs-dir-on-host*: /*mount-dir-on-container*:rw	Specify the directory path on the host (*scatefs-dir-on-host*) where a job accesses and the mount point on the container (*mount-dir-on-container*).

-v /var/run/scatefs: /var/run/scatefs:z	This setting is needed for a process in a container to access the ScaTeFS’s daemon. Add the option in the left column without a modification.
-v /etc/scatefs/client/: /etc/scatefs/client/:ro	This setting is needed only when using SX-Aurora TSUBASA. This setting is needed for a pcocess in a container to refer to the configuration file for ScaTeFS. Add the option in the left column without a modification.

Refer to “NEC Network Queuing System V (NQSV) User's Guide [Management]” about the other settings for the script to run a container.

8.4 Notice

ScaTeFS cannot be mounted on a Docker’s container by mount command. Mount ScaTeFS on the host and specify a ScaTeFS’s directory where the job accesses as the argument of docker-run’s –v option.

Chapter9　Operation management

System operation is sometimes stopped by IO server operation management processing. To stop this processing, you need to stop the IO server daemon. The IO server daemon is stopped and started by an EXPRESSCLUSTER command. Run this command on one of the two IO servers.

- To stop the IO server daemon

```
# clprsc -t exec1
# clprsc -t exec2
```

- To start the IO server daemon

```
# clprsc -s exec1
# clprsc -s exec2
```

9.1 Resource constraints (QUOTA)

The QUOTA functions shown below are available for file system and storage group.

Table 9-1　QUOTA function

Type	Quota function	Classification	
		Soft limit	Hard limit
User	Number of files	✓	✓
	Disk capacity	✓	✓
Group	Number of files	✓	✓
	Disk capacity	✓	✓
Directory	Number of files	✓	✓
	Disk capacity	✓	✓

It's possible to set QUOTA every user, group and directory. The QUOTA setting is applied to the number of files and the file capacity, for each of which a hard limit and soft limit can be set.

The hard limit is a value above which files cannot be allocated. When the hard limit is reached, EDQUOT is returned in response to a write request.

The soft limit is a value that can be exceeded, but only for a certain period of time (grace period). If the soft limit value is exceeded for longer than the specified grace period, the value is treated as a hard limit and files can no longer be allocated. The grace period is set to seven days by default, but this setting can be changed to between 1 second and 232-1 seconds for each file system and storage group (Refer to the section 9.1.1.2 for how to set up.

If files cannot be written because the hard limit has been reached or the grace period of the soft limit has passed, it is necessary to either delete files until the number or capacity falls below the hard or soft limit value, or change the QUOTA upper limit by using the **scatefs_edquota** command.

The file capacity is calculated by using the file size of the real files to be stored in each IO target. This means that the hole size of the real files is also included in the used capacity.

QUOTA function is enabled when after configuring the IO servers. Also, if the QUOTA function is disabled, the number of files and used file capacity are not counted.

- Directory QUOTA

Since user/group QUOTA are standard Linux functions, the following explains only directory QUOTA. Directory QUOTA is a function that sets QUOTA limit on per-directory. Directory QUOTA and user/group QUOTA work simultaneously. By using directory QUOTA, you can manage resources more flexibly.

Figure 9-1 shows an image in which QUOTA limit are set on each proj1/proj2 directory separately from the QUOTA limit for the user/group of the file system (FS1).

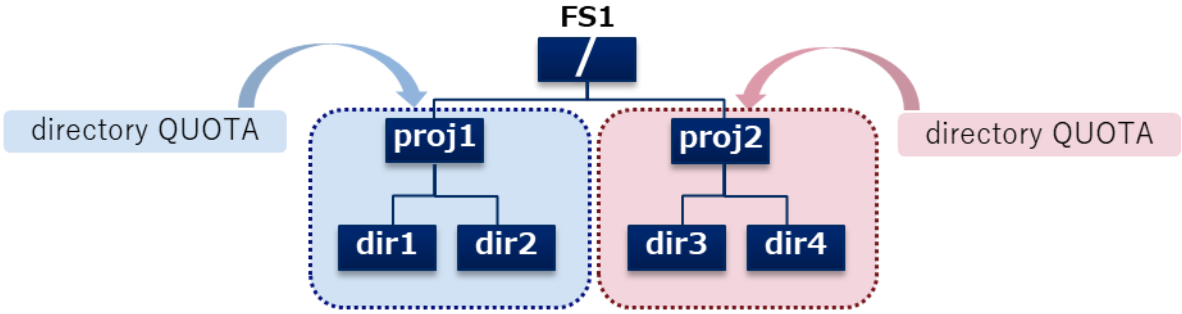


Figure 9-1 Image of directory QUOTA

Follow the steps below to operate directory QUOTA function.

(1) Create QUOTA control directory

To use directory QUOTA function, first create a target directory. The target directory is called the QUOTA control directory. In Figure 9-1, proj1/proj2 are the QUOTA control directory. To create the QUOTA control directory, use the scatefs_mkqdir command. See 9.1.1.5 about scatefs_mkqdir command.

(2) Editing QUOTA information

To edit directory QUOTA information, use scatefs_edquota command. See 9.1.1.2 about scatefs_edquota command.

(3) Checking QUOTA setting

To check the QUOTA setting, use scatefs_quota/scatefs_repquota commands. See 9.1.1.3 about scatefs_quota command and 9.1.1.4 about scatefs_repquota command.

You can check the usage of the directory QUOTA by specifying the QUOTA control directory or a file under it as an argument to the df command.

In this case, the df command is displayed as follows.

- Used　　: Usage in the directory QUOTA.
- Available: Remaining amount up to the hard limit. (※)

※　If the actual file system free space is less than the hard limit, the file system free space is displayed as the available amount.

Example:

```
# mount -t scatefs HOST:FS1 /mnt/scatefs
# df /mnt/scatefs/proj1
Filesystem  1K-blocks  Used  Available Use%  Mounted on
HOST:FS1       200704     0     200704    0%  /mnt/scatefs
```

If you mount the QUOTA control directory as a subdirectory, the QUOTA information for that directory is displayed in the results of the df command. See 9.13 about subdirectory mounts.

Example:

```
# mount -t scatefs HOST:FS1/proj1 /mnt/subdir
# df
Filesystem     1K-blocks  Used  Available Use%  Mounted on
 :
HOST:FS1/proj1    200704     0     200704   0%  /mnt/subdir
```

(4) Delete the QUOTA control directory

To delete the QUOTA control directory, use the scatefs_rmqdir command. See 9.1.1.6 about scatefs_rmqdir command.

9.1.1　Command

QUOTA settings can be configured either by logging in to the IO server and executing the QUOTA command for ScaTeFS, or by executing the QUOTA command for ScaTeFS from the pre-registered Linux client or SX-ACE client via a remote CLI.

The QUOTA commands can only be executed when the IO server daemon of each IO server is running and the QUOTA function is enabled.

An overview of each QUOTA command and a typical execution example are shown below: For. Refer to the man deta of each command for more details.

Command	Overview
scatefs_quotacheck	Check and repair quota files of ScaTeFS
scatefs_edquota	Edit user, group and directory quotas for ScaTeFS
scatefs_quota	display disk usage and limits for ScaTeFS
scatefs_repquota	Summarize quotas for ScaTeFS
scatefs_mkqdir	Create a quota control directory for ScaTeFS
scatefs_rmqdir	Remove a quota control directory for ScaTeFS

9.1.1.1 scatefs_quotacheck command

The **scatefs_quotacheck** command verifies the integrity of QUOTA information of each file system and storage group during operation, and fixes it in case of failure. This command can only be executed on the IO server. Execute the **scatefs_quotacheck** command while stopping operation of file system.

Example: To verify the consistency of the QUOTA information among the users, groups and directories of file system scatefs00:

```
# su fsadmin
$ scatefs_quotacheck scatefs00
```

Example: To verify the consistency of QUOTA information among the users, groups and directories of all file system:

```
# su fsadmin
$ scatefs_quotacheck -a
```

Example: To clear the specified hardand soft limits of filesystem scatefs00 and recalculate the used capacity information:

```
# su fsadmin
$ scatefs_quotacheck -c -g scatefs00
```

9.1.1.2 scatefs_edquota command

The scatefs_edquota command provides a function to set QUOTA for users, groups and directories. Only the administrator can execute this command to input/output QUOTA information for users and groups configured in any file system or storage group.

Example: To edit the QUOTA information of a user (UID 500) of file system scatefs00 on the IO server:
(The editor specified by the EDITOR environmental variable will open.)

```
# su fsadmin
$ export EDITOR=/bin/vi
$ scatefs_edquota -u 500 scatefs00
```

Example: To set 1000KB as soft limit and 2000KB as hard limit to a user (UID 500) of the file system scatefs00 from IO server:

```
# su fsadmin
$ scatefs_edquota -u 500 -b 1000:2000 scatefs00
```

Example: To set 5000 as soft limit of files and 10000 as hard limit of filesto a user (UID 500) of the directory "/dquota" in the file system scatefs00 from IO server:

```
# su fsadmin
$ scatefs_edquota –d /dquota -i 5000:10000 scatefs00
```

Example: To set QUOTA for a group (GID 500) of file system scatefs00 on IO server server00 from the Linux client:

```
$ scatefs_rcli server00 edquota -g 500 -b 1000:2000 -i 5000:10000 scatefs00
```

The scatefs_edquota command also provides a function to configure the following settings for the grace period that passes after the soft limits of the number of files and file capacity are exceeded. Only the administrator can configure these settings.

- Remaining grace period of each user, group or directories(grace time)
- Default grace period that passes after all users, groups or directories that belong to the file system or storage group exceed the soft limit (period time)

Example: To edit the grace period for exceed the group soft limit on file system scatefs00 from IO server:
(The editor specified by the EDITOR environmental variable will open.)

```
# su fsadmin
$ export EDITOR=/bin/vi
$ scatefs_edquota -T -u 500 scatefs00

Times to enforce softlimit for (user 0):
Time units may be: days, hours, minutes, or seconds
                 Filesystem    block grace    inode grace
                  scatefs00    3550seconds            unset
```

Example: To set QUOTA of a user (UID 500) of file system scatefs00 on the IO server: In the above example, the grace period for the file capacity is set to 7 days remaining (604,800 seconds).

```
# su fsadmin
$ scatefs_edquota -T -u 500 -b 604800 scatefs00
```

Example: To edit the grace period of the number of files to set 1 hour (3,600 seconds) for a user (UID 500) of file system scatefs00 on IO server server00 from the Linux client:

```
$ scatefs_rcli server00 edquota -T -u 500 -i 3600 scatefs00
```

Example: To edit the default grace period of the user soft limit on file system scatefs00 from IO server:

(The editor specified by the EDITOR environmental variable will open.)

```
$ export EDITOR=/bin/vi (In case of bash)
$ scatefs_edquota -t u scatefs00

Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
                 Filesystem    block grace period    inode grace period
                  scatefs00                   7days              3600seconds
```

Example: To edit the grace period of the group soft limit on file system scatefs00 from IO server:

In the above example, the grace period for the file capacity is set to 1 day remaining (86,400 seconds).

```
$ scatefs_edquota -t g -b 86400 scatefs00
```

Example: To set 10,000 seconds to grace period for number of files of all directory quotas of file system scatefs00 on IO server00 from the Linux client:

```
$ scatefs_rcli server00 edquota -t d -i 10000 scatefs00
```

9.1.1.3 scatefs_quota commands

scatefs_quota command displays quota informaton of the file system.

An administrator and users can execute this command, and it's possible to use by remote CLI command (9.8 Remote CLI). Users can confirm the user quotas, the group quotas and directory quotas using the remote CLI command.

Execute scatefs_quotacheck command (9.1.1.1 scatefs_quotacheck command) in advance if you want to get the accurate information.

Example: Display a list of QUOTA information for the user (UID=500) of file system scatefs00 on the IO server:

```
# su fsadmin
```

```
$ scatefs_quota -u 500 scatefs00 ScaTeFS quotas for user  (uid 500)
         Filesystem:sgname          blocks          quota          limit   grace    files      quota      limit
grace
-----------------------------------------------------------------------------------------------------------
            scatefs00:ROOT               0          488.2K           9.5M        -           0      10.0K
20.0K         -
```

Example: Display quota of qdir(DIRID 1000) directory on the file system scatefs00 at IO Serve

```
# su fsadmin
$ scatefs_quota -s -d qdir scatefs00
ScaTeFS quotas for directory /qdir500 (dirid 1000)
        Filesystem:sgname         blocks         quota          limit   grace        files       quota       limit   grace
 ------------------------------------------------------------------------------------------------------------------------------------------------
            scatefs00:ROOT              0          488.2K           9.5M        -            0       10.0K       20.0K          -
```

Example: Display group(group500,GID 500) quota of the file system scatefs00 on IO server server00 at Linux client:

```
# su fsadmin
$ scatefs_rcli server00 quota –s -g group500 scatefs00:sg000
ScaTeFS quotas for group group500 (gid 500)
        Filesystem:sgname         blocks         quota          limit   grace        files       quota       limit   grace
 --------------------------------------------------------------------------------------------------------------------------------------------------
            scatefs00:ROOT              0          500000      10000000        -            0       10000    1000000   -
```

9.1.1.4 scatefs_repquota commands

scatefs_repquota provides displaying all quotas of the file system. Only an administrator can execute this command, and it's possible to use by remote CLI command (9.8 Remote CLI).

Execute scatefs_quotacheck command (9.1.1.1 scatefs_quotacheck command) in advance if you want to get the accurate information.

Example: Display all user quotas of the file system scatefs00 at IO server:

```
# su fsadmin
$ scatefs_repquota -u scatefs00
*** Report for user quotas on scatefs00:ROOT
Block grace time: 7days; Inode grace time: 7days
             Block limits                                    File limits
   user(id)          used          soft          hard   grace        used       soft       hard   grace
  ------------------------------------------------------------------------------------------------------------
          0             0         32768         65536      -            0      10000      10000      -
        512             0         32768         65536      -            0      20000      30000      -
       1024             0         32768         65536      -            0      50000      60000      -
       2048        225416        524288       1048576       -         729        512       1024   6days
```

Example: Display all directory quotas of the file system scatefs00 at IO server:

```
# su fsadmin
$ scatefs_repquota -d scatefs00
*** Report for directory quotas on scatefs00:ROOT
Block grace time: 7days; Inode grace time: 7days
                              Block limits                                 File limits
   directory(name)    used          soft          hard    grace       used       soft      hard         grace
  ------------------------------------------------------------------------------------------------------------------
   qdir00            32768       2097152       4194304       -          750        500      1000         6days
   qdir01            65536       2097152       4194304       -          256        500      1000           -
   qdir02          1048576       2097152       4194304       -          128        500         0           -
   qdir03           524288       2097152       4194304       -          300        500         0           -
```

Example: Display all group quotas of the file system scatefs00 on IO server server00 at Linux client:

```
# scatefs_rcli server00 repquota -g scatefs00
```

```
(omit the output image)
```

The backup function enables information to be displayed on standard output and output to a backup file at the same time. The backup function can only be executed on the IO server.

Example: To output a backup file containing the same information after outputting a list of QUOTA information for users in the scatefs01 file system on the IO server:

```
#su fsadmin
$ scatefs_repquota -u -b scatefs01
/opt/scatefs/bin/scatefs_edquota -t u      -b 604800          -i 604800 scatefs01 || echo "error: user grace
scatefs01:SG1"
/opt/scatefs/bin/scatefs_edquota -u 1024 -b 102400:204800 -i 128:256 scatefs01 || echo "error: uid 1024
scatefs01"
/opt/scatefs/bin/scatefs_edquota -u 2048 -b 102400:204800 -i 128:256 scatefs01 || echo "error: uid 2048
scatefs01:SG1"
/opt/scatefs/bin/scatefs_edquota -u 3072 -b 102400:204800 -i 128:256 scatefs01 || echo "error: uid 3072
scatefs01 "
$ ls -l
-rw-rw-r-- 1 root fsadmin 630   Sep 18 16:58 scatefs_quota.fsid1.sgid1.user
```

Example: To recover the list of QUOTA information for users in the SGI storage group in the scatefs01 file system on the IO server by using backup file:

```
#su fsadmin
$ ls -l
-rw-rw-r-- 1 root fsadmin 630   Sep 18 16:58 scatefs_quota.fsid1.sgid1.user
$ sh ./scatefs_quota.fsid1.sgid1.user
```

9.1.1.5 scatefs_mkqdir commands

scatefs_mkqdir command provides creating a quota control directory which is able to set directory quota. This command can be executed at the IO server only. quota information are managed each created directory, and includes usage, hardlimit, softlimit, and remaining time.

It is necessary to use scatefs_rmqdir(9.1.1.6 scatefs_rmqdir command) for removing the directory.

Example: Create a quota control directory dquota00 under the root directory of the file system scatefs00 at IO server:

```
# su fsadmin
$ scatefs_mkqdir scatefs00 /dquota00
```

Example: Create a quota control directory dquota01 under the directory "work" of the file system(FSID=1) at IO server:

```
# su fsadmin
$ scatefs_mkqdir 1 /work/dquota01
```

9.1.1.6 scatefs_rmqdir commands

scatefs_rmqdir command provides removing a quota control directory which is able to set directory quota . This command can be executed at the IO server only.

Example: Remove a quota control directory dqupta00 of file system scatefs00 on the IO server:

```
# su fsadmin
$ scatefs_rmqdir scatefs00 /dquota00
```

Example: Remove a quota control directory dquota/dqupta01 of file system (FSID=1) on the IO server:

```
# su fsadmin
$ scatefs_rmqdir 1 /work/dquota01
```

9.2 Forced release of the record lock

ScaTeFS provides standard record locking defined by POSIX.1. Normally, record locking is preformed when the resource is used by a specific computing node exclusively, and it is released when the use is finished. However, if an error occurs on the computing node at which record locking is performed, the record locking for that node may not be released for some time depending on the operation. Therefore, ScaTeFS provides a function to forcibly release the record locking information of a specific node.

9.3 Expansion of file system

The file system can be extended by adding IO servers and IO targets. File system operation must be stopped to extend the file system. Extend the file system as follows:

(1) Unmount the file system from all clients.

(2) Follow the procedure described in “Configuring IO servers” to add new IO servers and IO targets to the system.

(3) Stop the IO server daemon on all IO servers.

(4) Use the **scatefs_extendfs** command when adding a file system if you want to extend the file system of the IO target that is being added.

Create the definition file and specify the filesystem to be extended and the IO target to be added.

```
-bash-4.1$ cat datafile
# File system ID of the file system you want to extend
fsid            0
# The IO target ID to add to the file system
addiotid        1
```

Execute scatefs_extendfs specifying the definition file as argument.

```
# su - fsadmin
-bash-4.1$ scatefs_extendfs -f datafile
```

(5) Specify the ID of the extended file system, check its consistency, and then recover the file system.

Example: If the file system ID is 0:

```
$ scatefs_fsck 0
```

(6)　Start the IO server daemon on all IO server.

(7)　Check and recover consistency of QUOTA information using quotacheck.

```
Example: If the file system name is scatefs00
$ scatefs_quotacheck scatefs00
```

9.4 Fair share

The fair share IO scheduling function is provided for IO servers. Unlike conventional job scheduling, this function enables a fair share of IO resources on the IO servers. This function performs efficient load balancing to avoid performance degradation of the whole system caused by the processing load on a specific user or computing node.

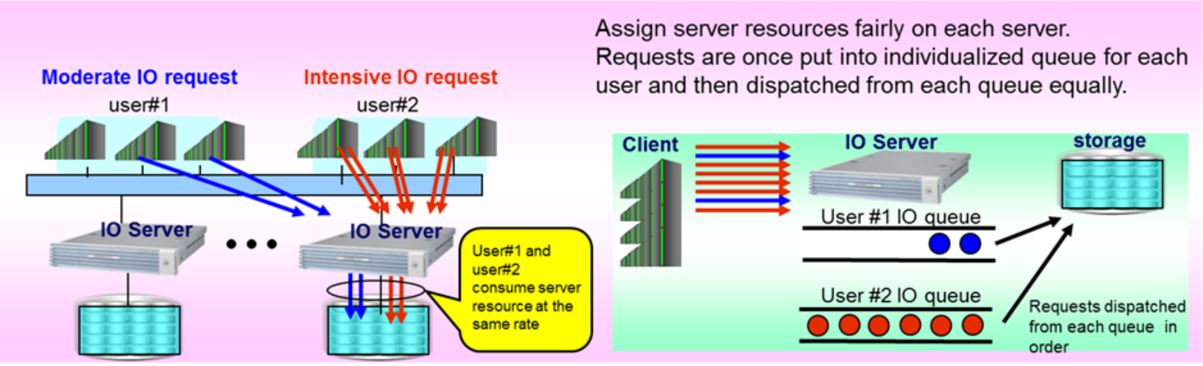

Moderate IO request
user#1
Intensive IO request
user#2
IO Server
IO Server
User#1 and user#2 consume server resource at the same rate
Assign server resources fairly on each server.
Requests are once put into individualized queue for each user and then dispatched from each queue equally.
Client
IO Server
storage
User #1 IO queue
User #2 IO queue
Requests dispatched from each queue in order

Figure 9-2 Image of fair share

To use this function, register the information in the configuration files of the IO servers. Note that dynamic changes during operation are not supported.

9.4.1 Policies

The IO scheduling function can be selected based on the following three policies.

- No fair share (default)
- Equalization per user (UID)
- Equalization per ClientID (unique ID per client)

The policy shall be the same for all IO servers. If it is changed, the IO servers need to be restarted.

9.4.2 How to change the policy

Follow the steps below to change the policy.

(1) Change FAIRPOLICY in the configuration file scatefssrv.conf.

Available setting values are as follows:

0: No fair share (default)

1: Equalization per UID

2: Equalization per ClientID

(2) Use the scatefs_admin command to distribute the modified scatefssrv.conf to all IO servers.

(3) Restart each IO server.

9.5 Storage group

The IO servers provide a function to group and manage multiple IO targets that make up the file system. These storage groups are associated with file system directories, and used as the managing units for QUOTA. These groups shall be called storage groups.

Using this function, for example, to group low-speed disks and high-speed disks enables various usages and charging depending on data characteristics.

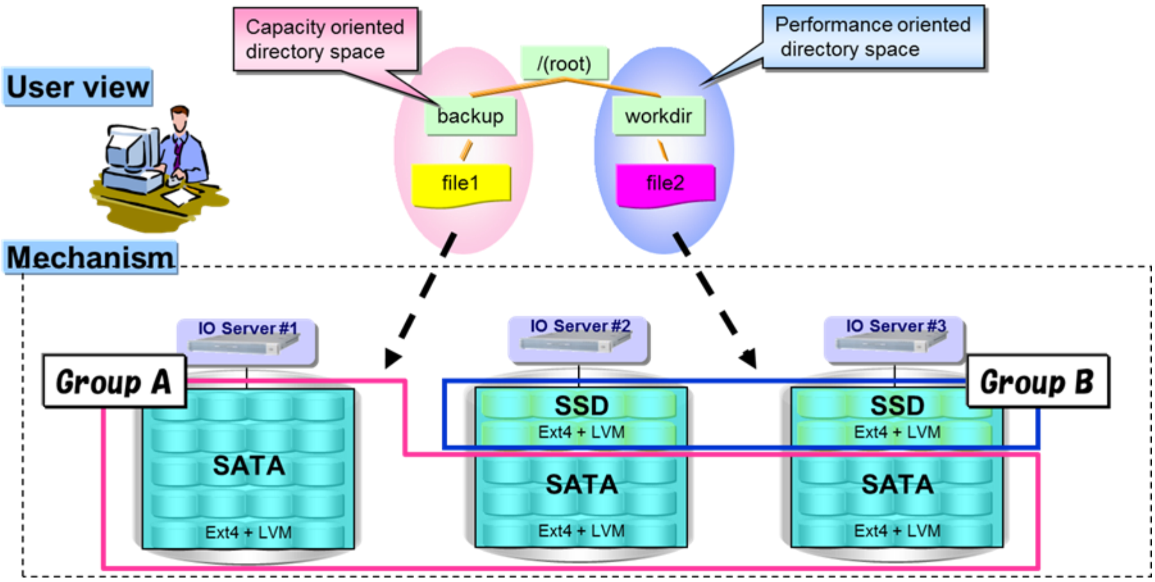


Figure 9-3 Conceptual diagram of storage group

For configuration, log in to any of the IO servers and run the ScaTeFS command. First, run the scatefs_extendfs command to add a storage group to a specific file system. Then, run the scatefs_mksgdir command to associate the storage group previously registered with a specific directory. When adding a storage group by using the scatefs_extendfs command, operation of the storage group must be stopped first. Also, it is necessary to start the IO server daemon before associating the storage group with a directory by using the scatefs_mksgdir command. Create a storage group as follows:

(1) Unmount the file system from all clients.

(2) Follow the procedure described in “Configuring IO servers” to add new IO servers and IO targets to the system.

(3) Stop the IO server daemon on all IO servers.

(4) Run the scatefs_extendfs command to specify the IO target to be added and create the storage group.

Create the definition file and specify the file system to be extended and the IO target to be added.

```
-bash-4.1$ cat datafile
# Specify a storage group to be added in the format <file system name>:<storage group name>.
name                    scatefs00:sgA
# The IO target ID to be assigned to the storage group.
# It must not have been registered with the file system.
iotid                   1
```

Execute scatefs_extendfs specifying the definition file as argument.

```
# su - fsadmin
-bash-4.1$ scatefs_extendfs -f datafile -addsg
```

(5) Specify the ID of the file system to which the storage group will be added, check its consistency, and then recover the file system.

```
    Example: If the file system ID is 0
    $ scatefs_fsck 0
```

(6) Check that the IO servers are running.

Use the following command to check that all the IO servers are running:

```
$ scatefs_admin --check system
IOSID   CONFIGFILE                  MD5SUM
    0   system.info             xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
    1   system.info             xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Check that no error messages have been output.

(7) Start the IO server daemon on all IO servers.

(8) Run the scatefs_mksgdir command to associate the storage group with a directory.

Example: $ scatefs_mksgdir scatefs sgA /sgAdir

If the mount point is '/mnt/scatefs', the directory '/mnt/scatefs/sgAdir' that belongs to sgA is created.

* Use the scatefs_rmsgdir command to delete a directory created by using the scatefs_mksgdir command.

Example: To delete a created directory:

```
$ scatefs_rmsgdir /sgAdir
```

(9)　Check and recover consistency of QUOTA information using scatefs_quotacheck

Example: If the file system name is scatefs00

```
$ scatefs_quotacheck scatefs00
```

9.6　Capacity management

When it receive a request for writing on an IO target whose capacity exceeds the threshold, ScaTeFS selects and uses another IO target with enough capacity in order to continue the IO operation. However, this function creates heavy demands for processing, and therefore it is preferable to avoid using it. To address this situation, consider re-balancing the file system while the system load remains low.

9.7 Rebalance

When the filesystem was extended, partiality of access to existence files and new files may occur. ScaTeFS provides Rebalance function which can equalize partiality of access and use IO bandwidth effectively. This function can be used without stopping operation of file system.

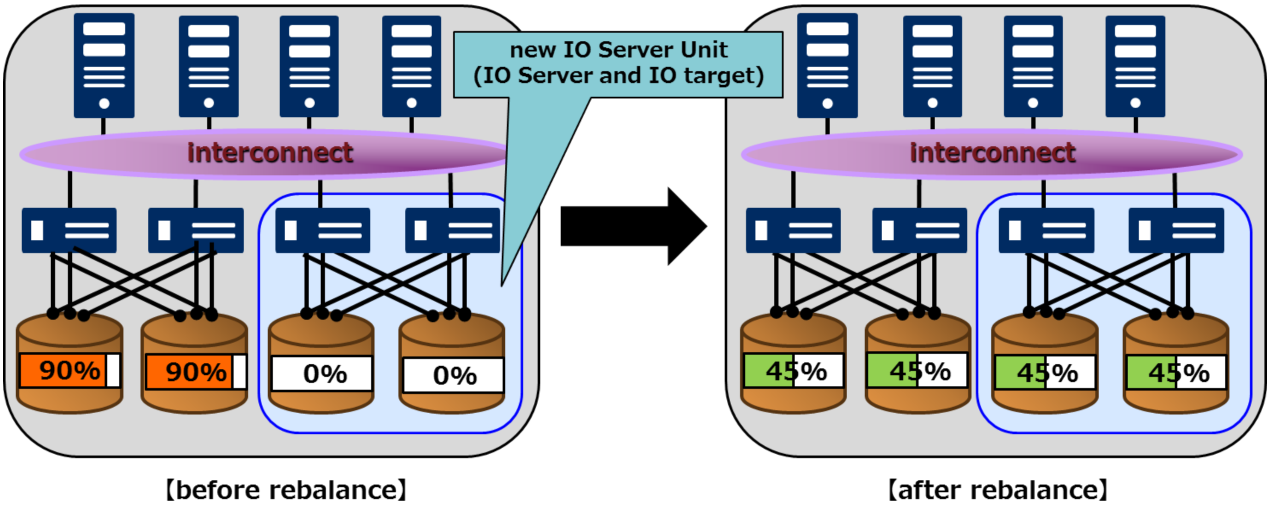


Figure 9-4　Example of execution rebalance at adding IO Server Unit

How to use Rebalance is following procedures:

(1)　Extracte the rebalance target files

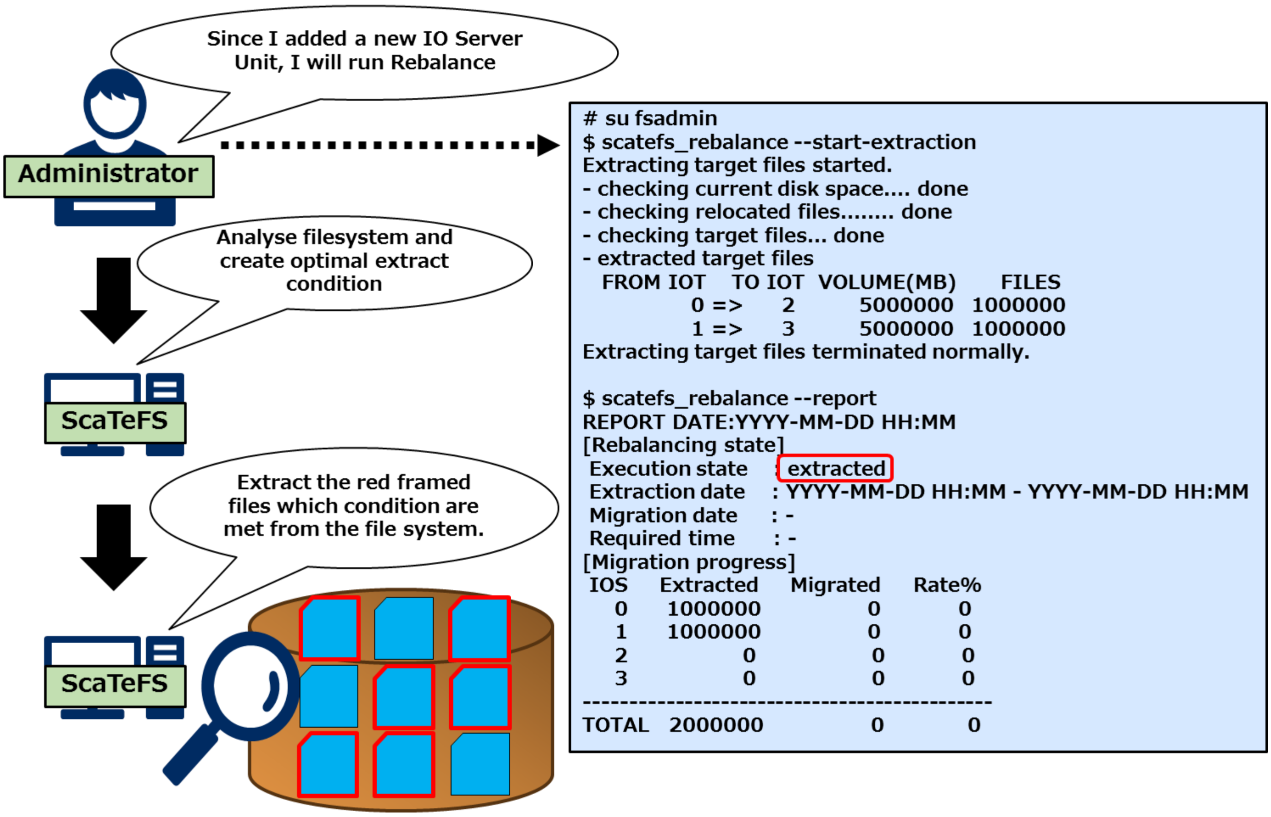
(2)　Migrate the rebalance target files

(3)　Clear the results of extraction

(4)　Clear the migration information (execute at the system maintenance)

(1)　Extracte the rebalance target files

Extracte rebalance target files using scatefs_rebalance command at IO Server. The status of extraction procedure can be checked using Report function.


Since I added a new IO Server Unit, I will run Rebalance
Administrator
Analyse filesystem and create optimal extract condition
ScaTeFS
Extract the red framed files which condition are met from the file system.
ScaTeFS

```
# su fsadmin
$ scatefs_rebalance --start-extraction
Extracting target files started.
- checking current disk space.... done
- checking relocated files........ done
- checking target files... done
- extracted target files
   FROM IOT   TO IOT  VOLUME(MB)      FILES
          0 =>      2          5000000  1000000
          1 =>      3          5000000  1000000
Extracting target files terminated normally.

$ scatefs_rebalance --report
REPORT DATE:YYYY-MM-DD HH:MM
[Rebalancing state]
 Execution state      : extracted
 Extraction date     : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Migration date       : -
 Required time        : -
[Migration progress]
 IOS      Extracted     Migrated      Rate%
    0      1000000              0             0
    1      1000000              0             0
    2                  0              0             0
    3                  0              0             0
-----------------------------------------------
TOTAL   2000000               0             0
```

Figure 9-5　Example of extract rebalance target file

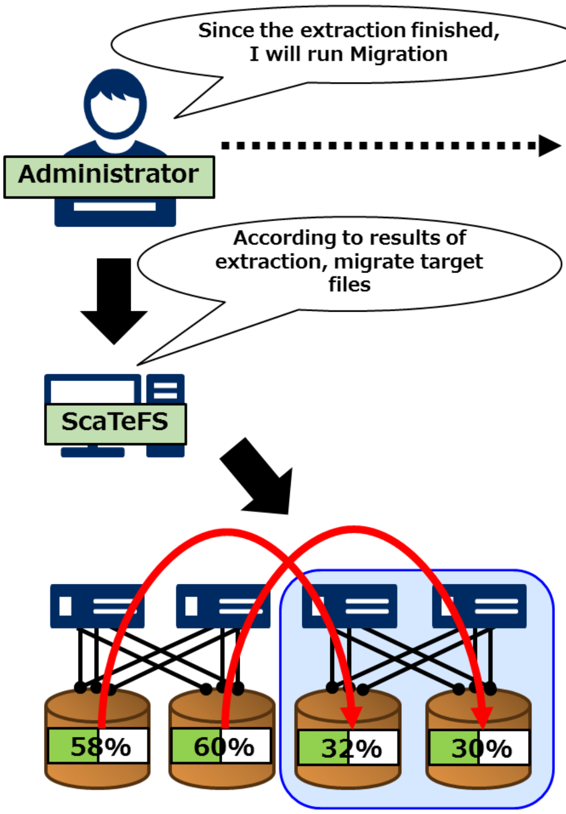
In case of extraction again, execute extraction after clearing the results.

```
# su fsadmin
$ scatefs_rebalance --clear
scatefs_rebalance: rebalance information was cleared.
$ scatefs_rebalance --start-extraction
Extracting target files started.
...
```

Also you can specify rebalance target file by scatefs_rebalance_import command at Linux client.

(2) Migrate the rebalance target files

Start migration service using scatefs_rebalance command at IO Server after extraction of rebalance target file was finished. So the target file will be migrated. The status of migration procedure can be checked using Report function.


Since the extraction finished,
I will run Migration
Administrator
According to results of extraction, migrate target files
ScaTeFS
58%
60%
32%
30%

```
# su fsadmin
$ scatefs_rebalance --start-migration
scatefs_rebalance: The migration service started normally.

$ scatefs_rebalance --report
REPORT DATE:YYYY-MM-DD HH:MM
[Rebalancing state]
 Execution state     : migrating
 Extraction date     : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Migration date      : YYYY-MM-DD HH:MM -
 Required time       : HH:MM:SS
[Migration progress]
 IOS     Extracted     Migrated     Rate%
   0      1000000       700000        70
   1      1000000       650000        65
   2            0            0         0
   3            0            0         0
--------------------------------------------
TOTAL   2000000     1350000         67
```

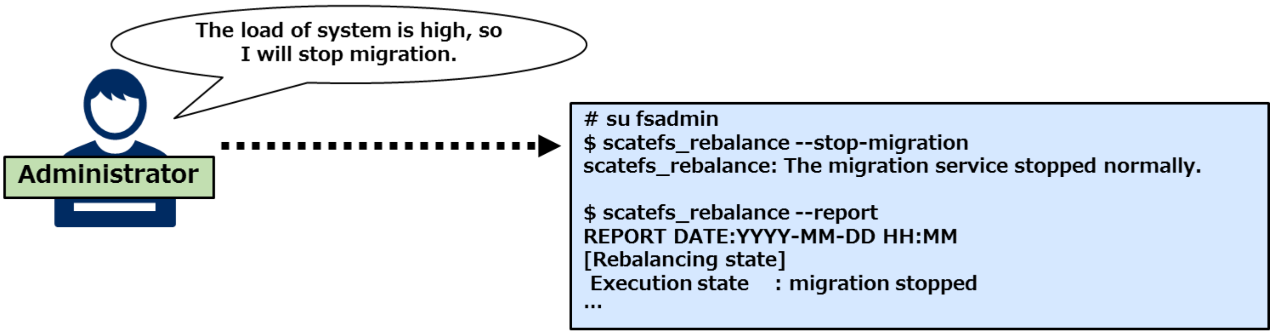
Figure 9-6 Example of migration rebalance target file

Stop the migration service after migration was finished.

```
# su fsadmin
$ scatefs_rebalance --report
REPORT DATE:YYYY-MM-DD HH:MM
[Rebalancing state]
 Execution state     : migrated
 Extraction date     : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Migration date      : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Required time       : HH:MM:SS
[Migration progress]
 IOS     Extracted     Migrated     Rate%
   0      1000000      1000000       100
   1      1000000      1000000       100
   2            0            0         0
   3            0            0         0
--------------------------------------------
TOTAL   2000000     2000000       100

$ scatefs_rebalance --stop-migration
scatefs_rebalance: The migration service stopped normally.
```

If necessary, the running migration service can be stoped and restarted. Stopping migration service may not be accepted in some case, so execute stop command again.



```
# su fsadmin
$ scatefs_rebalance --stop-migration
scatefs_rebalance: The migration service stopped normally.

$ scatefs_rebalance --report
REPORT DATE:YYYY-MM-DD HH:MM
[Rebalancing state]
 Execution state    : migration stopped
...
```

Figure 9-7　Example of stopping migration service

(3)　Clear the extraction results

Clear the extraction results using scatefs_rebalance command at the IO server after migration was finished.

```
# su fsadmin
$ scatefs_rebalance --clear
scatefs_rebalance: rebalance information was cleared.
```

Rebalance is completed at this point.

(4)　Clear the migration information (execute at the system maintenance)

Clear the migration information using scatefs_migrate at IO server after finishing migration and unmounting all client (*). Do not clear this information while the file system is mounted by one or more clients.

```
# su fsadmin
$ scatefs_migrate --clear
scatefs_migrate: The migration information was cleared.
```

(*) When the ScaTeFS file system is exported by NFS on the ScaTeFS client, unmount the exported file system on all NFS clients that mount the exported file system. And then, stop the nfs service on the ScaTeFS client.

If the migration information can not be cleared in spite of the file system is not mounted, specify --force option.

```
# su fsadmin
$ scatefs_migrate --clear
scatefs_migrate: cannot clear.
$ scatefs_migrate --clear --force
scatefs_migrate: The migration information was cleared forcibly.
```

9.8 Remote CLI

A remote CLI (scatefs_rcli) is provided as a mechanism to execute some of the commands on the IO server from the client. The subcommands that can be executed by scatefs_rcli are shown in the table below:

Table 9-2　Remote CLI Subcommand

Subcommand name	Overview	Execution user restrictions
df	Display the status of use of the ScaTeFS	None
detail	Display the configuration information of the ScaTeFS	Privileged users only
logcollect	Outputs IO server logs	Privileged users only
quota	Display the QUOTA information of the ScaTeFS	None
repquota	Outputs a ScaTeFS QUOTA information list	Privileged users only
edquota	Edits ScaTeFS user and group QUOTA settings	Privileged users only
ifstat	Display of IO server interface state	Privileged users only
mkqdir	Create a directory which is set QUOTA for ScaTeFS	Privileged users only
rmqdir	Remove a directory which is set QUOTA for ScaTeFS	Privileged users only

9.8.1 Privileged users

To provide users other than root users with privileges that enable them to execute the remote CLI, it is necessary to assign them to the fsadmin group. Users assigned to the fsadmin group become privileged users with the right to execute the remote CLI.

Example:

```
Add the fsadmin group.
# groupadd fsadmin

Add fsadmin to the group to which the foo user belongs.
    # usermod foo -G xxx,yyy,fsadmin
      * xxx and yyy are the groups to which foo already belongs.
```

9.8.2 Registering remote CLI users

A user who use scatefs_rcli from a client must be registered on the IO server. Use the scatefs_rcliadm command for user registration. After registration, refer to the example of 9.8.3 and confirm that operation.

Example:

```
Register the foo user of clientA.
$ scatefs_rcliadm add clientA foo

Confirm the registration
$ scatefs_rcliadm info
clientA foo

Delete the foo user of clientA
$ scatefs_rcliadm delete clientA foo
```

9.8.3 Executing commands using the remote CLI

Users registered by using the scatefs_rcliadm command can execute the scatefs_rcli command.

Example:

- The foo user of clientA specifies FSID#0 of serverB and executes the df subcommand.

```
$ scatefs_rcli serverB df 0
    IOT IOS SGID        1K-blocks            Used        Available   Use%   Mounted on
    0     0      0        11867221           305180        10974464      3%   /mnt/iot/0
    2     1      0        11867221           305180        10974464      3%   /mnt/iot/2
    1     0      0        11867213           305180        10974457      3%   /mnt/iot/1
    3     1      0        11867213           305180        10974457      3%   /mnt/iot/3
    --------------------------------------------------------------------------
    TOTAL                  47468868           1220720          43897842       3%
```

- Result if an unregistered user executes the scatefs_rcli command:

```
$ scatefs_rcli serverB df scatefs
Permission denied.
scatefs_rcli: df to serverB failed
```

9.9 Information display

The interface for retrieving various kinds of information regarding system configuration is provided as the commands deployed on IO servers.

- scatefs_df

  Display the status of use of the file system.

  Example:

```
Disk usage of the file system
$ scatefs_df scatefs00
 IOT IOS SGID       1K-blocks          Used       Available    Use%   Mounted on
   0    0    0    14276233588    8482274292    5492881741    60%   /mnt/iot/0
   3    1    0    14276233588    8488604028    5486868491    60%   /mnt/iot/3
   1    0    0    14276233588    8471883748    5502752757    60%   /mnt/iot/1
   4    1    0    14276233588    8461444560    5512669986    60%   /mnt/iot/4
   2    0    0    14276233588    8471705888    5502921724    60%   /mnt/iot/2
   5    1    0    14276233588    8471343548    5503265947    60%   /mnt/iot/5
 ---------------------------------------------------------------------------------------------------------------
 TOTAL              85657401528    50847256064    33001360646    60%

'inode' usage of the file system
$ scatefs_df scatefs00 -i
 IOT IOS SGID         Inodes         IUsed          IFree   IUse%    Mounted on
   0    0    0      32527525        816564       31710961      3%    /mnt/iot/0
   3    1    0      32527531        816625       31710906      3%    /mnt/iot/3
   1    0    0      32527531        816671       31710860      3%    /mnt/iot/1
   4    1    0      32527531        816755       31710776      3%    /mnt/iot/4
   2    0    0      32527531        816573       31710958      3%    /mnt/iot/2
   5    1    0      32527531        816734       31710797      3%    /mnt/iot/5
---------------------------------------------------------------------------------------------------------------
 TOTAL              195165180       4899922    190265258      3%

Disk usage of the storage group
$ scatefs_df -g scatefs00
 SGID        1K-blocks            Used        Available   Use%
    0        47093604           897648        43848570      2%

'inode' usage of the storage group
$ scatefs_df -g -i scatefs00
 SGID          Inodes           IUsed           IFree   IUse%
    0        4225772               12         4225760      0%
```

- scatefs_detail

  Display the configuration information of the file system.

  Example:

```
The entire file system
$ scatefs_detail -f 0
display detail FS#0
 FS Name          =>      scatefs00
 Root IOS         =>      IOS#0(IOT#0)
  IP              =>      10.0.0.1
  FIP             =>      10.0.1.1 10.0.2.1
PCI-ID@PORT       =>      0000:83:00.1@1
  INIP            =>      10.0.3.1
Number of IOS     =>      2
 Number of IOT    =>      6 / 1024
 Number of SG     =>      1 / 8
 Data FS type     =>      ext4
 Ctrl FS type     =>      ext4
 Version          =>      0x00010000
 IOTs             =>      0 3 1 4 2 5
 SG               =>      ROOT

Show in IOS units
$ scatefs_detail -s 0
display detail IOS#0
 IP ADDRESS                 =>      10.0.0.1
 Floating IP ADDRESS        =>      10.0.1.1 10.0.2.1
PCI-ID@PORT                 =>      0000:83:00.1@1
 Inner IP ADDRESS           =>      10.0.3.1
PORT for Client             =>      50000
 PORT for Server            =>      50001
 PORT for Client Data       =>      50002
 Defined IOTs               =>      0 1 2
 Defined FS                 =>      0

Display each IOT
$ scatefs_detail -t 0
display detail IOT#0
 defined server =>      IOS#0
 filesystem     =>      scatefs00
 storagegroup   =>      ROOT
 data device    =>      /dev/vg_data01/lv_data01
 ctrl device    =>      /dev/vg_ctrl01/lv_ctrl01
```

- scatefs_statcollect

  Display the statistics information of the IO server.

Example:

```
Display all statistics information of all IOS
$ scatefs_statcollect -a
[IOS#0]
 :
[IOS#1]
 :

Display the statistics information of the procedures on IO server ID#0.
$ scatefs_statcollect -n 0 -p
[IOS#0]
 :

Display the statistics information of the functions on IO server ID#1.
$ scatefs_statcollect -n 1 -f
[IOS#1]
 :
```

- scatefs_logcollect

  Display the logs of the IO servers

  ＊ To save the logs to a file, redirect the logs.

  Example:

```
Display the logs of all IO servers
$ scatefs_logcollect -a
    * To save the results:
  $ scatefs_logcollect -a > ioserver.log

  Display all logs of all IO servers (including rotated files and gz compressed files)
  $ scatefs_logcollect -a -m

  Display the log of the IO server ID#0
  $ scatefs_logcollect -n 0

Display the log of the IO server ID#1 and #2
 $ scatefs_logcollect -n 1,2
```

9.10 Managing system files

The scatefs_admin command is provided to manage system files on the IO servers. The scatefs_admin command can be used to perform operations such as checking the system files under the /etc/scatefs directory for consistency between IO servers, transferring files to and rolling back files from the specified IO server, and creating tuning parameter files. For details of the scatefs_admin command, see the man data on the IO server.

Example:

To check whether the ScaTeFS information file (system.info) is consistent between the IO servers:

```
$ scatefs_admin --check all system
```

To create a default tuning parameter configuration file (scatefssrv.conf) for the IO server daemon:

```
$ scatefs_admin --create tune
```

To transfer the IO server daemon tuning parameter configuration file (scatefssrv.conf) to all IO servers:

```
$ scatefs_admin --trans all tune
```

9.11 Monitoring the ScaTeFS filesystems

ScaTeFS provides Monitoring function which collect and monitor the ScaTeFS filesystems statistics in real time. By installing and configuring the required software and importing the templates included with the package, you can monitoring the ScaTeFS filesystems on a GUI basis.

The following statistics are supported:

Table 9-3 statistics

Source	Statistics
IO Server	Read/write throughput and metadata operation performance data per

	file system, IO server, and user ID. Network traffic and CPU information per IO server. Usage per file system. Profile information for each file system, such as the number of files in the directory and the distribution by file size (*).
IO Target	Read/write number per IO target. Usage per IO target.

(*) Collect profile information for each file system by executing the scatefs_profstat command (no arguments).
Execute the command according to the monitoring interval.
The time required for collection depends on your system environment.

Describes the software and software requirements that make up this feature.

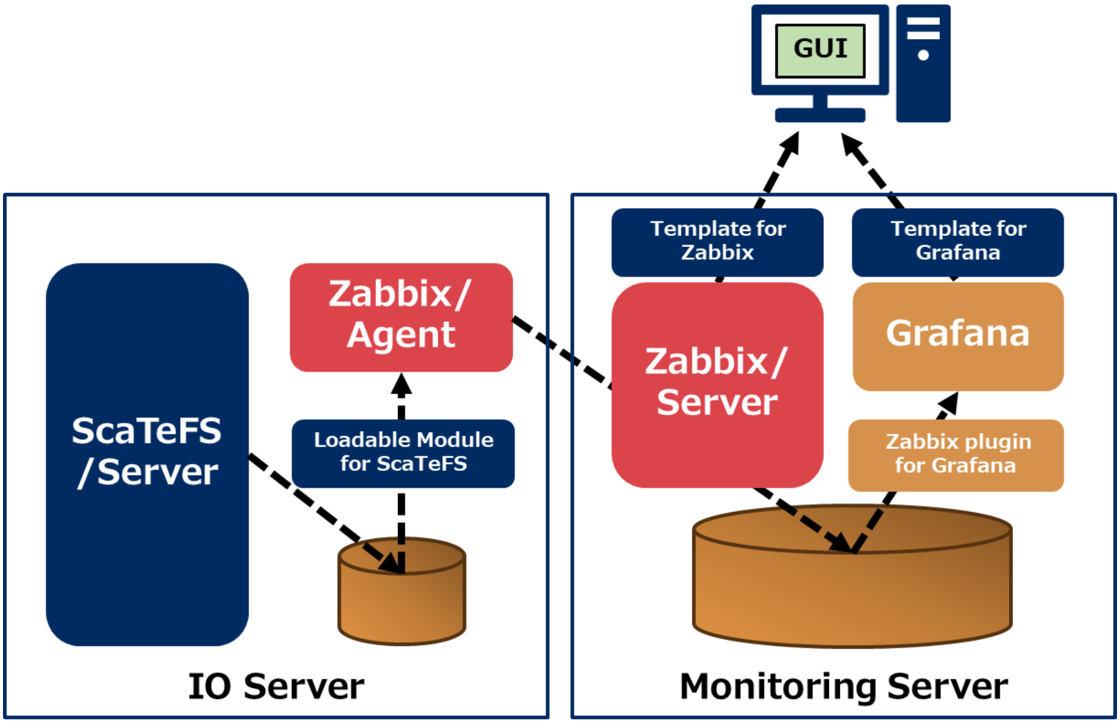


Figure 9-8 Configuration diagram

Table 9-4 Software

Software	Version
ScaTeFS/Server	scatefs-srv 3.3 or later

	scatefs-mon 3.3 or later The following is included with the scatefs-mon package. Loadable Module for ScaTeFS, Template for Zabbix, Template for Grafana
Zabbix/Server	zabbix-server 4.0 LTS or later confirmed: zabbix-server-mysql-4.0.17-2.el7
Zabbix/Agent	zabbix-agent 4.0 LTS confirmed: zabbix-agent-4.0.17-2.el7
Grafana	grafana-6.6 or later confirmed: grafana-6.6.1-1
Zabbix plugin for Grafana	v3.11.0 or later confirmed: alexanderzobnin-grafana-zabbix-v3.11.0-1-g52f24ec.zip

After installing ScaTeFS/Server, learn how to configure it to use this feature. Please refer to the community-provided documentations for basic settings for using Zabbix and Grafana.

- Loadable Module for ScaTeFS

  Get and install the scatefs-mon package as well as how to obtain the scatefs-srv package.
- Zabbix/Agent

  Get and install the software from the Zabbix community.

  To use Loadable Module for ScaTeFS, add the following settings to zabbix_agentd.conf:

```
LoadModulePath=/opt/scatefs/lib/
LoadModule=libscatefszbx.so
UserParameter=scatefs.alive.daemon, pgrep scatefs_server > /dev/null 2>&1; echo $?
```

- Zabbix/Server

  Get and install the software from the Zabbix community. To use the template, make the following settings:

(1) Import the template for Zabbix installed with the scatefs-mon package.

(2) Register the IO servers that make up the file system with the monitored host.
Configure these IO servers to belong to the same host group.

(3) Add the template installed in (1) to the monitored host that you added.

(4) Add the following to the macros for the monitored hosts that you added:
Macro Name:{$SCATEFS_HOSTGROUPNAME}
Value: "Host group name set by (2)" (enclosed in double quotes)

(5) Add the following settings to "/etc/zabbix/zabbix_server.conf":
For each set of IO server, specify 16MB for CacheSize and 8MB for TrendCacheSize.

```
CacheSize=16MB
TrendCacheSize=8MB
```

- Grafana and Zabbix plugin for Grafana

  Get and install the software from the Grafana community. To use the templates, make the following settings:

  (1) Enable Zabbix plugin for Graffana and add a data source.

  (2) Import the grafana templates that includes with the scatefs-mon package.

Describes the contents of the templates.

- Template for Zabbix

  Defines the monitoring items required for monitoring. It also defines the following failure monitoring triggers:

  - Alive monitoring of the ScaTeFS/Server Daemon.

    Monitor for ScaTeFS/Server daemon processes.
  - Monitoring of ScaTeFS File System Usage.

    Monitor usage at three levels.
- Templates for Grafana

  Three screens are defined.

  - Data screen of ScaTeFS

    Displays various statistics on read/write operations for each file system and IO server.
  - Metadata screen of ScaTeFS

    Displays various statistics about metadata operations for each file system and

IO server.

- ScaTeFS throughput/IO size per UID

  Displays various statistics about read/wiite and metadata operations by user ID.

9.12 ScaTeFS InfiniBand high performance library

9.12.1 Overview of ScaTeFS IB library and ScaTeFS VE direct IB library

ScaTeFS IB Library supports lightweight and high performance IO for large IO through a user space by using IB specific API. Program1 and program3 can IO to IO server directly bypassing the kernel space of VH with these library. You can expect a performance improvement of applications which issue large IOs. Because the library hooks read(2)/write(2) related system calls in libc and change them into library processes automatically, you can use the library without any modifications of your program.

See 6.1.5 about setting and see 11.6 about how to use.

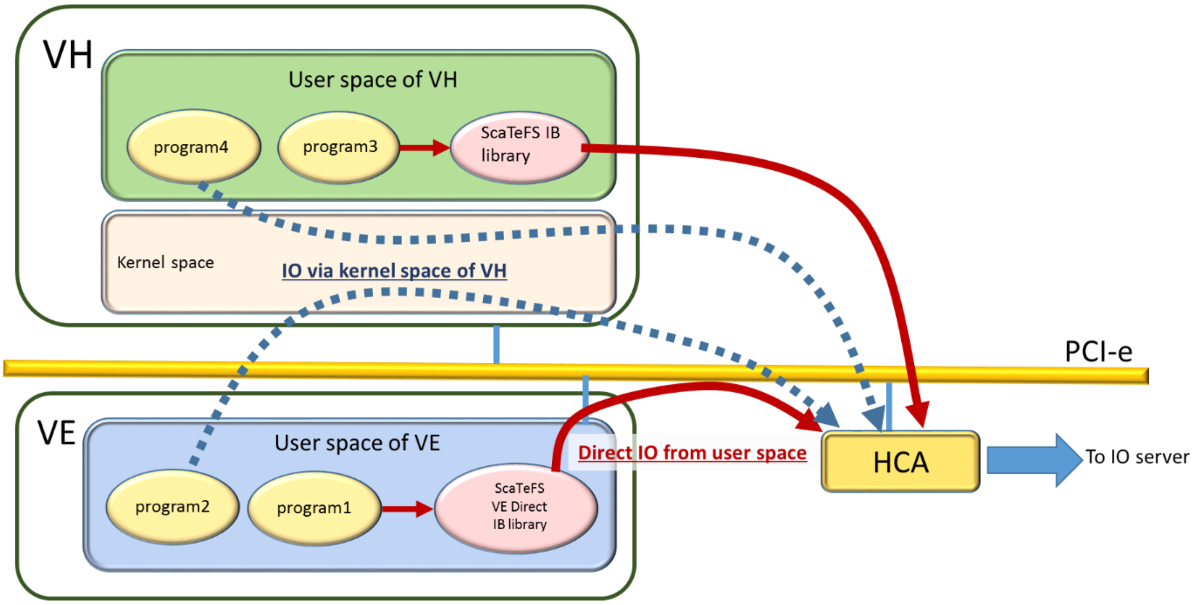


Figure 9-9　ScaTeFS InfiniBand high performance library

9.12.2 The threshold at which IO is processed by IB specific API

When an IO size specified to read(2)/write(2) related system call is greater than 1MB or equal, the library processes it by high performance IO processing using IB specific API. When an IO size is less than 1MB, the library processes it by normal kernel IO processing.

9.12.3 Setting of disk sync mode

There are two modes about syncing written data to disk as following. The relation between the two modes is trade-off of the performance and the data reliability. The default mode is disk sync on close mode.

- Disk sync on close(2) mode (default mode)

  The data written to a file is synced to disk at close(2) not write(2).

  Because the library doesn't sync to disk until close(2), the write performance is higher than “disk sync on write(2) mode” mentioned in the following.

  When an IO server failover occurs, a job on this mode will catch the error and not keep running. This behavior is different from the conventional kernel IO. When occurring this error, the library outputs the error message to the standard error output. A user needs to re-run these jobs caught this error.

  The operational image of disk sync on close(2) mode is shown in Figure 9-10.

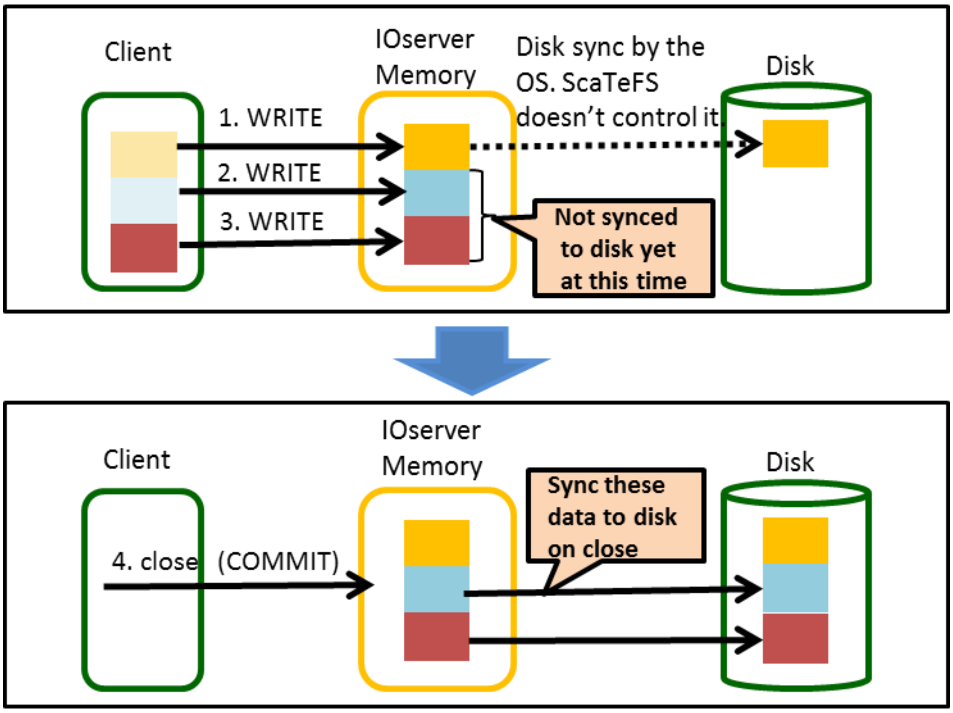

Client
IOserver Memory
Disk sync by the OS. ScaTeFS doesn’t control it
Disk
1. WRITE
2. WRITE
3. WRITE
Not synced to disk yet at this time
Client
IOserver Memory
Sync these data to disk on close
Disk
4. close (COMMIT)

Figure 9-10　Operational image of disk sync on close(2) mode

- Disk sync on write(2) mode

  The data written to a file is synced to disk at write(2).

  When an IO server failover occurs, a job on this mode will keep running same as the conventional kernel IO.

  However the write performance is lower than “disk sync on close(2) mode” because a disk sync is done on every write(2).

The operational image of disk sync on write(2) mode is shown in Figure 9-11 Operational image of disk sync on write(2) mode.

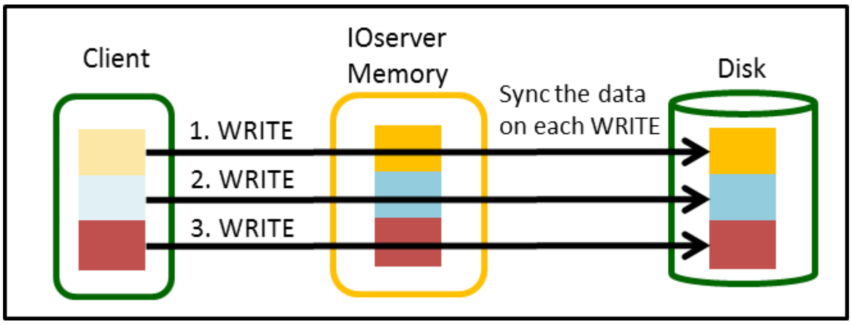


Figure 9-11　Operational image of disk sync on write(2) mode

The setting is done on each IO server. See 5.3.2.1 about the setting. The following is the summary of the differences between the two modes.

Table 9　The differences of disk sync mode

Mode	Write performance	Appropriate use case	Dealing with job failure in case IO server's failover
Disk sync on close(2) mode (default)	High	Mostly executing write(2) to comparatively large file(more than 128MB) with 4MB IO size or larger.	Find out the job which caught the error of read(2)/write(2) related systemcalls or close(2). And re-run them.
Disk sync on write(2) mode	Normal	Mostly executing write(2) to comparatively large file(more than 128MB) with 128MB IO size or larger.	A user don't need any actions because the job can be recovered automatically and keep running.

9.12.4 Setting of the IO buffer's memory location

When using IB for a file system port on “IO server v3 for small-scale model”, you need to set IBSIOMEMNODE in the IO server configuration file “scatefssrv.conf”. As the Figure 9-12　Relation between the performance and the IO buffer's memory location, the IO server consists of the two set (nodes) of CPU and main memory. These two nodes are connected with the inter-connect. On the data transfer using IB specific API, the optimum IO performance is realized by using the memory on the node installs a HCA for IO processing.

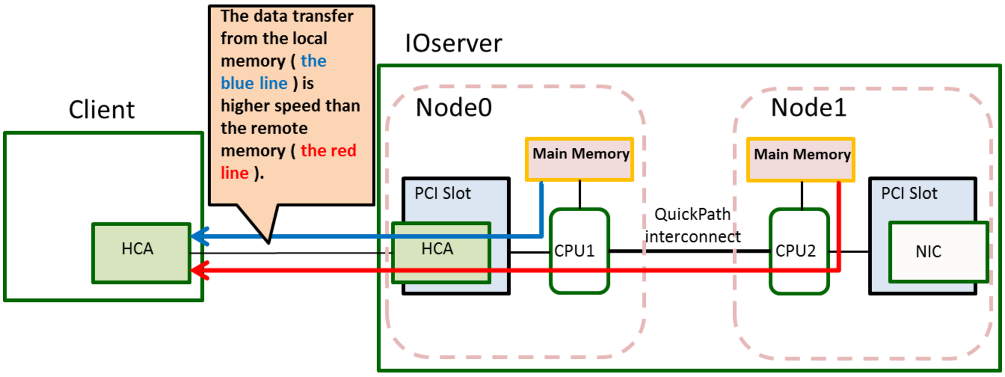

The data transfer from the local memory (the blue line) is higher speed than the remote memory (the red line).
Client
HCA
IOserver
Node0
Node1
Main Memory
Main Memory
PCI Slot
HCA
CPU1
QuickPath interconnect
CPU2
PCI Slot
NIC

Figure 9-12 Relation between the performance and the IO buffer's memory location

Because the default of IBSIOMEMNODE is 1 (Node 1), you don't need to set it when using “IO server v3 for standard model” installs a HCA in Node 1.
Set IBSIOMEMNODE to 0 in scatefssrv.conf only when using “IO server v3 for small-scale model”. See 5.3.2.1 about how to set.

9.13 Subdirectory mount

Subdirectory mount function provides the ability to mount a part of directory tree from the ScaTeFS file system. You can mount any directory in the tree of the ScaTeFS file system. It is possible to operate a part of the file system as an access target with subdirectory mount.

Figure 9-13 shows an operational image of subdirectory mount.

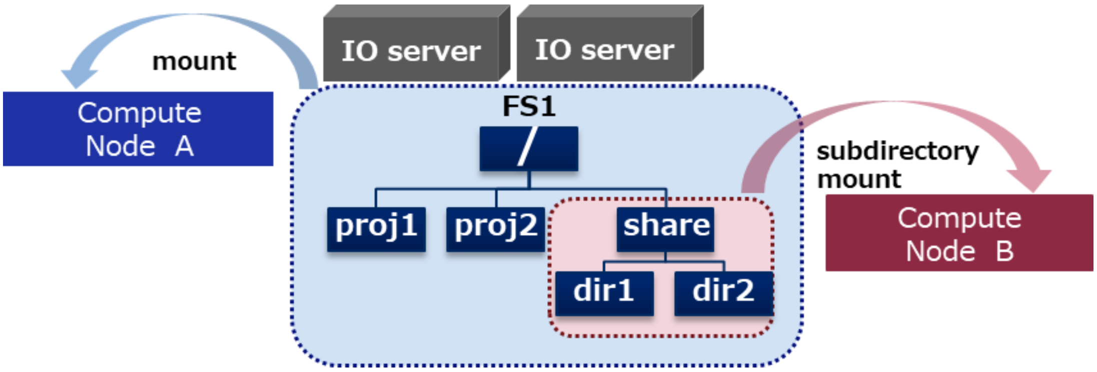


Figure 9-13 Image of subdirectory mount operation

In this figure, a ScaTeFS file system (FS1) consisting of two IO servers is mounted on two clients (Compute Nodes A/B). On compute Node A, the entire FS1 is mounted and all directories in FS1 are accessible. On compute Node B, /share directory of FS1 is partially mounted, so /share/dir1 and /share/dir2 are accessible, but /proj1 and /proj2 are not.

The accecibility of the two clients (A/B) to FS1 is as follows.

Directory	Compute Node A	Compute Node B
/	✓	-
/proj1	✓	-
/proj2	✓	-
/share	✓	✓
/share/dir1	✓	✓
/share/dir2	✓	✓

9.13.1 Mounting

When mounting a subdirectory, append the path name of subdirectory "/SUBDIR" to the target to be mounted and use the format "HOST:FSNAME/SUBDIR".

The following is an example to mount HOST:FS1/share on /mnt/subdir.

```
# mount -t scatefs HOST:FS1/share /mnt/subdir
```

9.13.2 Unmounting

Use the **umount** command to unmount the file system as before.
For example to unmount a part of file system (HOST:FS1/share) mounted on /mnt/subdir, you can unmount with one of following images:

```
# umount /mnt/subdir
```

or

```
# umount HOST:FS1/share
```

Chapter10 Maintenance

10.1 Start and stop the IO Server

How to start and stop the IO servers of the cluster configuration.

- start

  Press the power button of two IO servers continuously.

  Note that do not leave the interval of pressing the power button.

- stop

  Log in to either the IO server and executes clpstdn command.

  Two IO servers will be halted.

```
# clpstdn
```

- restart

  Log in to either the IO server and executes clpstdn -r command.

- Two IO servers will be rebooted.

```
# clpstdn -r
```

- Check that the IO servers are running.

  Log in to either the IO server and executes clpstat command. Display cluster state by the clpstat command, and check the below:

  a) All resources are Online or Normal.

  b) The server name of the group is shown to "current" of <group> tag.
     When the IO server is in the failover state, the same server name is shown to "current" of 2 <group> tags.
     When there is a problem, cancel the fault of the corresponding resource.

  The example is below:

```
# clpstat
 ========================  CLUSTER STATUS  ===========================
  Cluster : cluster
  <server>
   *iosv00 ..........: Online
      lankhb1          : Normal           Kernel Mode LAN Heartbeat
      diskhb1          : Normal           DISK Heartbeat
    iosv01 ..........: Online
```

```
       lankhb1           : Normal              Kernel Mode LAN Heartbeat
       diskhb1           : Normal              DISK Heartbeat
   <group>
     failover1 .......: Online
       current           : iosv00
       disk_c_01         : Online
       disk_c_02         : Online
       disk_c_03         : Online
       disk_d_01         : Online
       disk_d_02         : Online
       disk_d_03         : Online
       exec1             : Online
       exec_route1       : Online
       fip_ib1           : Online
       volmgr_c_01       : Online
       volmgr_c_02       : Online
       volmgr_c_03       : Online
       volmgr_d_01       : Online
       volmgr_d_02       : Online
       volmgr_d_03       : Online
     failover2 .......: Online
       current           : iosv01
       disk_c_04         : Online
       disk_c_05         : Online
       disk_c_06         : Online
       disk_d_04         : Online
       disk_d_05         : Online
       disk_d_06         : Online
       exec2             : Online
       exec_route2       : Online
       fip_ib2           : Online
       volmgr_c_04       : Online
       volmgr_c_05       : Online
       volmgr_c_06       : Online
       volmgr_d_04       : Online
       volmgr_d_05       : Online
       volmgr_d_06       : Online
   <monitor>
     diskw_c_01          : Normal
     diskw_c_04          : Normal
     fipw1               : Normal
     fipw2               : Normal
     genw1               : Normal
     genw2               : Normal
     userw               : Normal
     volmgrw1            : Normal
     volmgrw10           : Normal
```

```
    volmgrw11          : Normal
    volmgrw12          : Normal
    volmgrw2           : Normal
    volmgrw3           : Normal
    volmgrw4           : Normal
    volmgrw5           : Normal
    volmgrw6           : Normal
    volmgrw7           : Normal
    volmgrw8           : Normal
    volmgrw9           : Normal
 ==================================================================
#
```

10.2 Maintenance of servers in operation

This section describes the maintenance work for IO servers

10.2.1 Backup

ScaTeFS does not support a specific backup function. Therefore, mount a file system on the backup server and back up data in virtual file units.

10.2.2 Non-stop update of the ScaTeFS package

The non-stop update feature enables to update the scatefs-srv package during using the ScaTeFS file system service. However, this feature might not be available when all IO server which construct the file system must be synchronized. Note that this feature can be used only when the document of ILC package says the non-stop update feature is available.

Log in to each IO server as an administrator (that is, with root privileges) and configure the following procedure:

Note that access to the updating IO server is delayed for at most 3.5 minutes. When updating multiple IO server contiguously, the delay time increases proportionally to the number of IO servers. Therefore make an enough time (at least 8 minutes) between one IO server updating and other so that practical use of the system isn't affected.

The procedures of non-stop update of the ScaTeFS/Server package are below:

10.2.2.1　When using the HPC Software License

The procedures of non-stop update of ScaTeFS/Server package is different depending on whether you have the PP support contract of ScaTeFS/Server or not. The following explanation is divided into two cases where the PP support is contracted and the PP support is not contracted.

(5) Preparations

[If you have the PP support contract of ScaTeFS/Server, see below:]

a) Setting yum repository

Refer to 5.1.12.1 Check that (1) of [If you have the PP support contract of ScaTeFS/Server, see below:].

b) Check of ScaTeFS/Server package

Confirm that the new package exists in the repository.

```
# yum list available scatefs-srv
```

[If you do not have the PP support contract of ScaTeFS/Server, see below:]

a) Setting yum repository

Refer to 5.1.12.1 Check that (1) of [If you do not have the PP support contract of ScaTeFS/Server, see below:].

b) Getting zip file including ScaTeFS/Server package

Download the zip file including the ScaTeFS/Server package using the internet delivery product download service.

Refer to 5.1.12.1 Check that (2) of [If you do not have the PP support contract of ScaTeFS/Server, see below:].

(6) Confirm the environment

Check cluster state by the clpstat command. Refer to 10.1　Check that the IO servers are running for more information.

If there is a problem, some error may occur in the resource. In such situation, the error must be fixed then execute non-stop update.

(7) Update

1. Apply a package when the IO server demon is running.

[If you have the PP support contract of ScaTeFS/Server, see below:]

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-server
# yum group update scatefs-server
```

[If you do not have the PP support contract of ScaTeFS/Server, see below:]

```
# yum update scatefs-srv-VER.x86_64.rpm
```

2. Execute the following command.

```
# /opt/scatefs/sbin/scatefs_restart
```

(8) Confirm the result

When the command has normally ended (0), this update is completion.

When the command has abnormally ended (1), check cluster state by the clpstat command and recover the system by the following procedure which corresponds to the condition. After that, inform the support section.

When the IO server is in the failover state, execute the following command to return it to the package before this update. Then execute the takeback.

[If you have the PP support contract of ScaTeFS/Server, see below:]

The transaction id is confirmed from the history of yum, execute the undo command to return it to the old package.

```
# yum history list
# yum history undo X

* X: transaction id
```

[If you do not have the PP support contract of ScaTeFS/Server, see below:]

Execute the following command to return it to the old package.

```
# yum downgrade <old ScaTeFS/Server package>
```

10.2.2.2 When using the SX Cross Software Node-lock License

(1) Confirm the environment

Check cluster state by the clpstat command. Refer to 10.1 Check that the IO servers are running for more information.

If there is a problem, some error may occur in the resource. In such situation, the error must be fixed then execute non-stop update.

(2)　Update

1. Apply a package when the IO server demon is running.

```
# rpm -Uvh <new package name>
```

2. Execute the following command.

```
# /opt/scatefs/sbin/scatefs_restart
```

(3)　Confirm the result

When the command has normally ended (0), this update is completion.

When the command has abnormally ended (1), check cluster state by the clpstat command and recover the system by the following procedure which corresponds to the condition. After that, inform the support section.

When the IO server is in the failover state, execute the following command to return it to the package before this update. Then execute the takeback.

```
# rpm -Uvh -oldpackage <old package name>
```

Others

Return it to the package before this update. Then execute the scatefs_restart command.

10.3 Maintenance requiring system shutdown

Some maintenance work cannot be done while the system is in operation. In this case, shut the system down before maintenance.

- File system extension, storage group addition, and storage group extension by scatefs_extendfs
- Recovery by fsck (local file system and ScaTeFS file system)
- Integrity check and Recovery ScaTeFS QUOTA information by scatefs_quotacheck

10.4 Integrity check and recovery of the file system

The integrity check and recovery function dedicated for the ScaTeFS file system is provided. Recovery requires to stop operation of the ScaTeFS file system. There are two procedures to recover:

[Normal recovery procedure (recommended)]

Perform all maintenance within the single downtime.

(1) Stop operation for the ScaTeFS file system

(2) Perform the fsck for local file system (if necessary)

(3) Perform the fsck (integrity check and recovery) for ScaTeFS file system

(4) Perform the integrity check and recovery for QUOTA information (recommended)

(5) Restart operation for the ScaTeFS file system

[Recovery procedure to reduce downtime]

Reduce downtime by performing file system integrity check in operation.

(1) If recovery of the local file system such as a disk failure is required:

- Stop operation for the ScaTeFS file system
- Perform the fsck for local file system
- Restart operation for the ScaTeFS file system

(2) Perform the fsck (integrity check only) for ScaTeFS file system. And save the result to a file. (The fsck can be performed in operation only for integrity check)

(3) Stop operation for the ScaTeFS file system

(4) Perform the fsck for local file system (if necessary)

(5) Perform the fsck (recovery only) for ScaTeFS with the integrity check result file produced in the step (2)
(6) Perform the integrity check and recovery for QUOTA information (recommended)
(7) Restart operation for the ScaTeFS file system

Note that if recovery of the local file system such as a disk failure is required, you may not be able to access some directories and files while the step (2) and the step (3). These are resolved by the step (5).

Here's how to use each commands:

- Integrity check
  Specify the ID of the targeted file system to perform the integrity check of the file system.

Example:

```
$ scatefs_fsck -n fsid
```

- Integrity check and recovery
  Specify the ID of the targeted file system to perform the recovery of the file system.
  Stop the IO server daemon of all IO servers before recovering the file system.
  * After recovery is complete, to verify that the file system has been correctly recovered, perform recovery again.

Example:

```
$ scatefs_fsck fsid
```

- Recovery the ScaTeFS file system with the integrity check result
  Perform the fsck for ScaTeFS with the integrity check result file. Because the target files to be recovered are already extract, the fsck executing time is reduced.
  Stop the IO server daemon of all IO servers before recovering the file system.
  * After recovery is complete, to verify that the file system has been correctly recovered, perform recovery again.

Example:

```
$ scatefs_f2fsck infile
```

- Integrity check and recovery QUOTA information

  Specify the file system name of the targeted file system to perform the check and recovery of quota information. Start the IO server daemon of all IO servers before recovering the quota information.

  Example:

```
$ scatefs_quotacheck fsname
```

10.5 Switching paths in the event of a network path failure

The ScaTeFS client communicates with the IO server using multiple network paths.
If a network failure occurs on one of the paths connecting the ScaTeFS client and the IO servers, the ScaTeFS client switches to an available path and continues communicating (this is known as "path switching"). The path on which the network failure occurred is monitored by the ScaTeFS path monitoring daemon, and once the path recovery is detected, use of the path is automatically resumed. This means that no special measures are required to resume communication following recovery from a network failure.

10.6 10GbE-NIC

If the support department issues instructions to upgrade 10GbE-NIC, you need to update the driver. Although ScaTeFS uses the DCB function, the RPM binary package provided by the 10GbE-NIC vendor does not support the DCB function as is. It is therefore necessary to separately obtain the procedure for updating the 10GbE-NIC driver from the support department.

10.7 Firmware update after ConnectX-6 HCA card replacement

When replacing a failed ConnectX-6 HCA card, manual firmware update may be required.
(*) When the target machine is SX-Aurora TSUBASA, refer to the SX-Aurora TSUBASA guide, not the procedures in this section. This section is for IO server and Linux machine (scalar machine) except SX-Aurora TSUBASA.

After HCA card replacement, check the firmware version by ibstat command.

```
$ /usr/sbin/ibstat | grep -i firmware
            Firmware version: 20.27.6008
```

When the displayed version is older than the above version, update firmware by the procedures in this section.

Update procedures are described below:

(1) Download a firmware file by the NVIDIA official site.
    https://network.nvidia.com/support/firmware/connectx6ib/
    When using HDR100 1port model, select the "MCX653105A-ECA" for OPN.

(2) Transfer the downloaded firmware file to the update target machine. When the firmware file is compressed, uncompress it.

(3) Executes the following commands on the update target machine.

```
# mst start
Starting MST (Mellanox Software Tools) driver set
Loading MST PCI module - Success
Loading MST PCI configuration module - Success
Create devices
Unloading MST PCI module (unused) – Success

# mst status
MST modules:
------------
     MST PCI module is not loaded
     MST PCI configuration module loaded

MST devices:
------------
/dev/mst/mt4123_pciconf0          - PCI configuration cycles access.
                                     domain:bus:dev.fn=0000:83:00.0 addr.reg=88 data.reg=92
                                     Chip revision is: 00
```

The above is an execution example. Display may differ in the actual environment. When multiple HCAs are installed, multiple devices paths /dev/mst/mtXXXX_pciconfX are also displayed.

(4) Update the firmware by the mlxfwmanager command.

Specify the device path checked in (3) for "-d" option.

Specify the firmware file for "-i" option.

```
# mlxfwmanager -d /dev/mst/mt4123_pciconf0 -i fw-ConnectX6-rel-20_26_1040-MCX653105A-ECA_Ax-UEFI-14.19.14-
FlexBoot-3.5.803.bin –u
Querying Mellanox devices firmware ...

Device #1:
----------


  Device Type:       ConnectX6
  Part Number:        MCX653105A-ECA_Ax
  Description:       ConnectX-6 VPI adapter card; 100Gb/s (HDR100; EDR IB and 100GbE); single-port QSFP56; PCIe3.0
x16; tall bracket; ROHS R6
  PSID:               MT_0000000222
  PCI Device Name:   /dev/mst/mt4123_pciconf0
  Base GUID:          xxxxxxxxxxxxxxxx
  Versions:           Current          Available
     FW               AA.AA.AAAA       BB.BB.BBBB
     PXE              x.x.xxxx         x.x.xxxx
     UEFI             xx.xx.xxxx       xx.xx.xxxx

  Status:            Forced update required

---------
Found 1 device(s) requiring firmware update...

Device #1: Updating FW ...
Initializing image partition -   OK
Writing Boot image component -   OK
Done

Restart needed for updates to take effect.
```

New firmware version will be displayed in BB.BB.BBBB.

The above is an execution example. Display may differ in the actual environment.

(5) When multiple device path is displayed in (3), execute (4) to all devices.

⚠ **Notice**

- If the HCA card is not recognized normally, re-fit the HCA card once to identify the cause of the failure.

- It takes a while for the update to be reflected. If the update fails, re-execute once after reboot.

(6) Reboot the update target machine.

```
# reboot
```

(7) Ping to the update target machine and check the response.

```
$ ping <The machine IP addres>
```

(8) Login the update target machine and executes the hca_self_test.ofed command with root privileges.

```
# hca_self_test.ofed
---- Performing Adapter Device Self Test ----
Number of CAs Detected ................. 1
PCI Device Check ........................ PASS
Kernel Arch ............................ x86_64
Host Driver Version .................... MLNX_OFED_LINUX-4.6-4.1.2.0 (OFED-4.6-4.1.2): 3.10.0-957.27.2.el7.x86_64
Host Driver RPM Check .................. PASS
Firmware on CA #0 HCA .................. vBB.BB.BBBB
Host Driver Initialization ............. PASS
Number of CA Ports Active .............. 1
Port State of Port #1 on CA #0 (HCA)..... UP 2X HDR (InfiniBand)
Error Counter Check on CA #0 (HCA)...... PASS
Kernel Syslog Check .................... PASS
Node GUID on CA #0 (HCA) ............... b8:59:9f:03:00:00:a7:04
------------------ DONE --------------------
```

When multiple HCAs are installed, multiple results will be displayed.

For each result of HCA card, check the following:

- In the result of "Firmware on CA #N", check that the displayed firmware version matches the updated version.

- In the result of "Host Driver Initialization", check that "PASS" is displayed.
- In the result of "Error Counter Check on CA #N (HCA)", check that "PASS" is displayed.
- In the result of "Kernel Syslog Check", check that "PASS" is displayed.

(9) Check the PCI link by lspci command.

The specified PCI ID is the value checked by "mst status" command.

```
# lspci -s 83:00.0 -vvv | grep LnkSta:
    LnkSta: Speed 8GT/s, Width x16, TrErr- Train- SlotClk+ DLActive- BWMgmt- ABWMgmt-
```

Check that "Speed" is 8GT/S and "Width" is x16.

The firmware update is now complete.

10.8 Syslog messages

10.8.1 Linux client

File system operation

```
ScaTeFS:400100 commit error after file close. filesystem name=<filesystem name> dev=<device number>
code=<code> data=<internal data>
```

[Type]ERROR

[Explanation]　An error occurred in the file system during the delayed synchronization with the storage of the IO server after the file had been closed.

filesystem name: File system name

device number: Device number of the file system

code: Code which represents the error (the same number as errno)

internal data: Internal data

When the errors occur continuously in the same file system, the message is outputted every one hour.

[Measures]　　Recover the file which encountered the error by rerun of the job which creates the file and so on after removing the cause of the error.

The administrator or the user identifies the file which encountered the error by the failure date (the output date of the message), the file system information in this message, the file information in the following message ScaTeFS:400101, the access situation of the file by the application and so on.

```
ScaTeFS:400101 commit error after file close. dev=<device number> ino=<inode number> uid=<user id>
gid=<group id> code=<code> data=<internal data>
```

[Type]ERROR

[Explanation] An error occurred for the file during the delayed synchronization with the storage of the IO server after the file had been closed.

device number: Device number of the file system

inode number: Inode number of the file

user id: User ID of the file

group id: Group ID of the file

code: Code which represents the error (the same number as errno)

internal data: Internal data

If the number of output of this message exceeds 200 for 5 seconds, output of this message is restrained for 5 seconds after that and the following message ScaTeFS:400102 is outputted.

When the message ScaTeFS:400102 was outputted, it isn't possible to specify all the files which encountered the error from this message. There is the file that encountered the error, but this message wasn't outputted for.

[Measures] Recover the file which encountered the error by rerun of the job which creates the file and so on after removing the cause of the error.

The administrator or the user identifies the file which encountered the error by the failure date (the output date of the message), the file system information in the above-mentioned message ScaTeFS:400100, the file information in this message, the access situation of the file by the application and so on.

When the message ScaTeFS:400102 was outputted, it isn't possible to specify all the files which encountered the error from this message. The administrator or the user need to identify the file which encountered the error by the access situation of the file by the application just before the failure.

```
ScaTeFS:400102 drop commit error messages due to rate-limiting. data=<internal data>
```

[Type]ERROR

[Explanation] The message ScaTeFS:400101, which represents that an error occurred for the file during the delayed synchronization with the storage of the IO server after the file had been closed, was restrained.

internal data: Internal data

[Measures] Unnecessary

Data transportation (TCP)

```
ScaTeFS:RPC: all connections related to <ServerAddress>:<Port> are failed, still trying
```

[Type]WARNING

[Explanation] Communication to the IO server is failed. All paths are in fault.

[Measures] Check that there are any faults in network paths and the status of the IO server.

```
ScaTeFS:RPC: all connections related to <ServerAddress>:<Port> are failed, timed out
```

[Type]WARNING

[Explanation] Communication to the IO server is failed. All paths are in fault. The file operations will be error because the file system is mounted with soft mount option.

[Measures] Check that there are any faults in network paths and the status of the IO server.

```
ScaTeFS:RPC: retry to server <ServerAddress>:<Port> has been cancelled by signal.
```

[Type]NOTICE

[Explanation] Retransmission is aborted by a signal.

[Measures] Unnecessary.

```
ScaTeFS:RPC: server <ServerAddress>:<Port> OK
```

[Type]NOTICE

[Explanation] Retransmission succeeded.

[Measures] Unnecessary.

```
ScaTeFS:RPC: server <ServerAddress>:<Port> is unavailable. Using alternative connection path
```

[Type]WARNING

[Explanation]　The number of retries exceeds the limit on one path. The client started to use an alternative path.

[Measures]　Check that there are any faults in the network path and the status of the IO server.

```
ScaTeFS:RPC: server <ServerAddress>:<Port> not responding, still trying
```

[Type]NOTICE

[Explanation]　Communication to the IO server is timed out. The client is retrying.

[Measures]　If this occurs frequently, check that there are any faults in the network path and the status of the IO server.

```
ScaTeFS:RPC: server <ServerAddress>:<Port> not responding, timed out. (pid=<PID>, proc=<ProcedureNumber>)
```

[Type]NOTICE

[Explanation]　There is no response to the RPC request. The RPC request was failed because the file system is mounted with soft mount option.

[Measures]　Check that there are any faults in the status of the IO server and the network path.

```
ScaTeFS:pmond: connect to server <ServerAddress>:<Port> ok
```

[Type]NOTICE

[Explanation]　Communication path is recovered.

[Measures]　Unnecessary.

Data transportation (IB Verbs)

```
ScaTeFS:verbs: all connections related to <ServerAddress> for <ConnectionType> are failed, still trying.
```

[Type]WARNING

[Explanation]　Communication to the IO server is failed. All paths are in fault.

[Measures]　Check that there are any faults in network paths and the status of the

IO server.

```
ScaTeFS:verbs: all connections related to <ServerAddress> for <ConnectionType> are failed, timed out.
```

[Type]WARNING

[Explanation]　Communication to the IO server is failed. All paths are in fault. The file operations will be error because the file system is mounted with soft mount option.

[Measures]　　Check that there are any faults in network paths and the status of the IO server.

```
ScaTeFS:verbs: connection to <ServerAddress>:hca<N> is marked as disconnected. (<Internal data>)
```

[Type]NOTICE

[Explanation]　An unavailable connection was found and disconnected.

[Measures]　　Unnecessary.

```
ScaTeFS:verbs: Control request to <ServerAddress> failed. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Control communication to the IO server is failed. IPoIB communication to the server is not available.

[Measures]　　This is an assistance message. Refer messages around this message.

```
ScaTeFS:verbs: Control request to <ServerAddress> was skipped. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Control communication to the IO server is failed. IPoIB communication to the server is not available.

[Measures]　　This is an assistance message. Refer messages around this message.

```
ScaTeFS:verbs: Control request to <ServerAddress> was skipped. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Control communication to the IO server is failed. IPoIB communication to the server is not available. N means the index of the HCA on the IO server (one-based).

[Measures] This is an assistance message. Refer messages around this message.

```
ScaTeFS:verbs: detaching device done. (device=<HCA>)
```

[Type]WARN

[Explanation] Detected an abnormality in the HCA. This HCA is excluded for communication.

[Measures] Check the status of the client HCA.

```
ScaTeFS:verbs: pmond: could not connect to server <ServerAddress>:<hcaN>, still trying. (<Internal data>)
```

[Type]WARN

[Explanation] Could not connect to the IO server. The status is being monitored periodically.

N means the index of the HCA on the IO server (one-based).

[Measures] Check that there are any faults in the network path and the status of the IO server.

```
ScaTeFS:verbs: pmond: InfiniBand device is unavailable, retry after delay. (device=<HCA>, guid=<DeviceGuid>)
```

[Type]WARN

[Explanation] Detected an abnormality in the HCA. The status is being monitored periodically.

[Measures] Check the status of the client HCA.

```
ScaTeFS:verbs: re-attaching device done. (device=<HCA>)
```

[Type]NOTICE

[Explanation] The attachment of the HCA device is completed. Resume using the HCA.

[Measures] Unnecessary.

```
ScaTeFS:verbs: re-attaching device done. (device=<HCA>, not mounted)
```

[Type]NOTICE

[Explanation]　The attachment of the HCA device is completed. The HCA is available.

[Measures]　　Unnecessary.

```
ScaTeFS:verbs: server <ServerAddress>:hca<N> request transmission was not successful, still trying. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Communication to the IO server is failed. The client is retrying. N means the index of the target HCA on the IO server (one-based).

[Measures]　　If this occurs frequently, check that there are any faults in the network path and the load of the IO server.

```
ScaTeFS:verbs: server <ServerAddress>:hca<N> is unavailable (<Internal data>). Using alternative connection path.
```

[Type]WARNING

[Explanation]　The number of retries exceeds the limit on one communication path. The client started to use an alternative path. N means the index of the target HCA on the IO server (one-based).

[Measures]　　Check that there are any faults in the network path and the status of the IO server.

```
ScaTeFS:verbs: server <ServerAddress>:hca<N> not responding, still trying. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Communication to the IO server is timed out. The client is retrying. N means the index of the target HCA on the IO server (one-based).

[Measures]　　If this occurs frequently, check that there are any faults in the network path and the load of the IO server.

```
ScaTeFS:verbs: server <ServerAddress>:hca<N> OK. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Retransmission succeeded. N means the index of the target HCA on the IO server (one-based).

[Measures]　　Unnecessary.

```
ScaTeFS:verbs: server <ServerAddress>:hca<N> recovery OK. (<Internal data>)
```

[Type]NOTICE

[Explanation]　Communication path is recovered. N means the index of the target HCA on the IO server (one-based).

[Measures]　　Unnecessary.

```
ScaTeFS:verbs: start re-attaching device. (devname=<HCA>)
```

[Type]NOTICE

[Explanation]　The HCA is detected. The attachment of the HCA device is started.

[Measures]　　Unnecessary.

License

```
ScaTeFS_LS:300001 heartbeat to license server failed. continue process. errmsg=<error message> data=<internal data>
```

[Type] WARNING

[Explanation]　　An error occurred during sending heartbeat to license server. The client retries after the time of heartbeat interval has elapsed.

error message: Error message

internal data: Internal data

[Measures]　　If this occurs frequently, check whether there are any faults in the network path and　check the license server status.

```
ScaTeFS_LS:300002 heartbeat to license server recovered. data=<internal data>
```

[Type]　　WARNING

[Explanation]　Sending heartbeat to license server is recovered.

internal data: Internal data

[Measures]　　Unnecessary.

```
ScaTeFS_LS:400101 ScaTeFS client license is not valid. data=<internal data>
```

[Type]ERROR

[Explanation]　Nodelock license is not valid.

internal data: Internal data

[Measures]　　Check whether the license file is set correctly.

```
ScaTeFS_LS:400201 ScaTeFS client license process failed. reason=<reason> data=<internal data>
```

[Type]ERROR

[Explanation]　License process is failed.

reason: Error messages indicating reason of failure

internal data: Internal data

[Measures]　　Perform necessary process according to error message indicating reason of failure.

10.8.2 IO server

Describes how to monitor the failure of an IO server using syslog.

The *** indicates any string.

Storage related messages

```
lpfc***Down
or
lpfc***Reset
```

[Type] ERROR

[Explanation] Detected a failure in the server-side FC port.

[Measures] A failure may have happened on the path between the storage and the IO server. Please contact our support department.

```
sps: Warning: Detect *** path fail
or
sps: Warning: *** is not redundant
```

[Type] ERROR

[Explanation] Detected a failure in the disk port.

[Measures] Check the path configuration with the spsadmin command.

Please check the PathManager related manuals and contact our support department.

Networt related messages

```
cxgb4***link down
```

[Type] ERROR

[Explanation] Detected link down for 10G NIC (T4 card).

[Measures] Please contact our support department.

EXPRESSCLUSTER related messages

```
There was a request to restart resource(***) from the clprm process
```

[Type] WARNING

[Explanation] EXPRESSCLUSTER detected an abnormal state in the resource and restarted the resource. EXPRESSCLUSTER may perform failover.

[Measures] Check the status of ScaTeFS (*1) and resolve the error. For message details, refer to the EXPRESSCLUSTER related manuals.

```
Detected an error in monitoring ***
```

[Type] ERROR

[Explanation] EXPRESSCLUSTER detected an error in monitoring monitor resources. EXPRESSCLUSTER may perform failover.

[Measures] Check the status of ScaTeFS (*1) and resolve the error. For message details, please refer to the EXPRESSCLUSTER related manual.

```
Resource *** of server *** has stopped
```

[Type] ERROR

[Explanation] The particular resource on the IO server has stopped. EXPRESSCLUSTER performs failover.

[Measures] ScaTeFS can be used in the failover state. Check the status of ScaTeFS (*1) and resolve the error.

However, if two or more sets of IO server are in the failover state and resource error is unknown, stop using ScaTeFS immediately not to grow failure.

For message details, refer to the EXPRESSCLUSTER related manuals.

ScaTeFS related messages

```
IOS*** server started (secondary mode)
```

[Type] ERROR

[Explanation] The ScaTeFS server function is in the failover state.

[Measures] ScaTeFS can be used in the failover state. Check the status of ScaTeFS (*1) and resolve the error.

However, if two or more sets of IO server are in the failover state and resource error is unknown, stop using ScaTeFS immediately not to grow failure.

```
async event(IBV_EVENT_LID_CHANGE) at hca(***). stop the daemon.
or
async event(IBV_EVENT_CLIENT_REREGISTER) at hca(***). stop the daemon.
```

[Type] ERROR

[Explanation] Because of the subnet manager related problem e.g. reboot, the IO server daemon was rebooted.

[Measures] Check the status of the subnet manager. The reboot of the subnet manager for a maintenance should be done while the use of ScaTeFS is stopped.

```
async event(IBV_EVENT_SM_CHANGE) at hca(***). stop the daemon.
```

[Type] ERROR

[Explanation] The subnet manager was switched to a spare, the IO server daemon was rebooted.

[Measures] Check the status of the subnet manager.

```
InfiniBand timeout happened on HCA#<N> (PID=*** CLIENTID=***)
```

[Type] WARNING

[Explanation] Timeout happened on InfiniBand communication. N is the index of HCA on IO server. It is corresponding to the HCA which is specified at the Nth item of pciid@hcaport in the definition file for scatefs_addios command (zero-based).

[Measures] Check that there are any faults in the network path.

```
NET: hca(***:<hca-id1>:<hca-port1>) is replaced with hca(***:<hca-id2>:<hca-port2>)
```

[Type] WARNING

[Explanation] Because an inactive HCA was detected in the initialization phase of IO server daemon, an active HCA substituted for it and the daemon started. <hca-idX>:<hca-portX> is the combination of HCA ID and HCA port that is specified at pciid@hcaport in the definition file for scatefs_addios command.

[Measures] Check that there are any faults of HCAs on IO server.

(*1) "Check the status of ScaTeFS" refers to the following:

- Use the clpstat command to view the cluster status and verify the following:

  If it is different, there is something wrong with the resource.

  o All resources must be Online or Normal.

  o The server name of the group is displayed in the current of tag <group>.

  (If the system is failover, the same server name will appear in the current of two tags <group>.)

- Make sure that you have successfully accessed it from the client.

  o Perform IO check after mounting as described in 6.2.5 Mount method

Chapter11 Configuration and instructions for end users

11.1 The virtual file system and real file system

ScaTeFS consists of multiple IO servers, which are shown virtually to ScaTeFS clients as one file system. Therefore, it is called “virtual file system”.

As shown in Figure 11-1, the virtual file system consists of multiple Linux file systems created in the storage devices connected under each IO server. These are called “real file systems” or “IO targets”. Each IO server has at least one (generally more than one) real file system. To perform parallel I/O processing efficiently, you need to understand how many IO servers and real file systems make up the virtual file system.

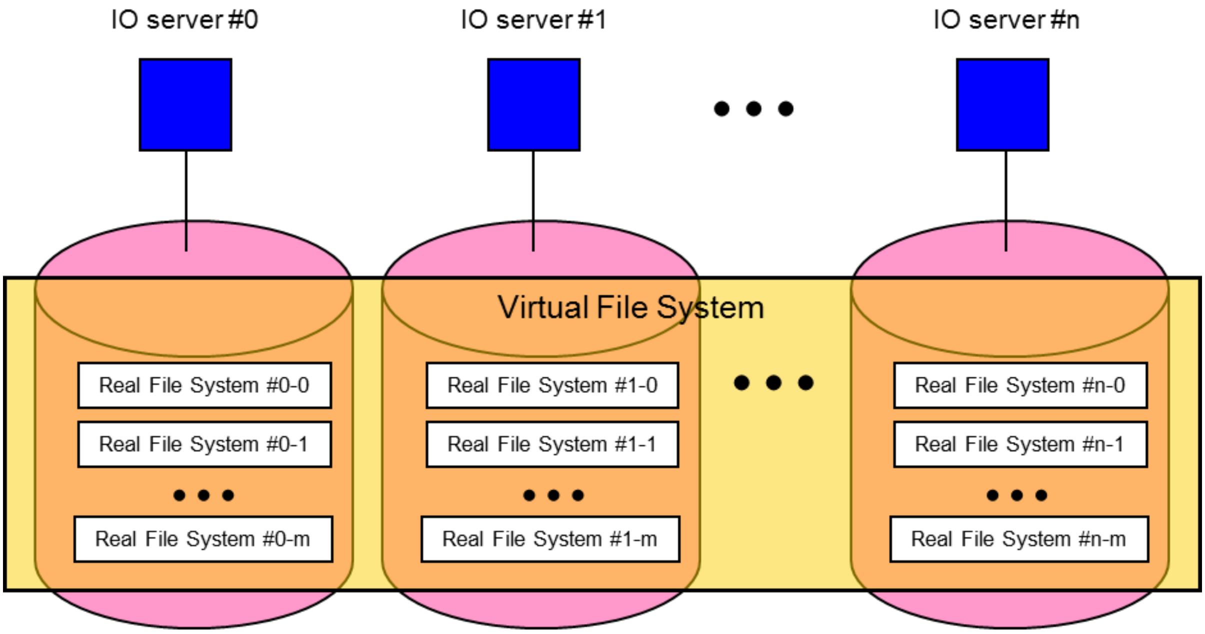


Figure 11-1　Relation between the virtual file system and real file systems

In the example in Figure 11-1, the file data are distributed to at most (n + 1) x (m + 1) real file systems.

11.2 Virtual files and real files

Fragments of the virtual file are distributed to each real file system. These fragments are called “real files”. Two file formats can be selected depending on the fragments and the way of distribution to real file systems.

Format 1: Non-stripe format

Format 2: Stripe format

The default format is non-stripe format.

11.2.1 Non-stripe format (Format 1)

Like the image of the virtual file shown in Figure 11-1, the virtual file is made up of real files consecutively connected. The unit for this connection is called "chunk size". This value can be specified by scatefs_premap(1) described later, and the default value is 256 MB.

Figure 11-1 shows the image of the virtual file in case of non-stripe format (Format 1), and an example of how the real files, which make up the virtual file, are distributed to each real file system.

In this example, 2 real file systems (targets) are created for each IO server that makes up a single ScaTeFS. The virtual file shown in Figure 11-1 comprises chunk #0 through #10, and the head of the virtual file, chunk #0, is assigned to TID:1. Then, the chunk numbers are consecutively distributed in the following TID order:

TID = (1, 2, 3, 0, 5, 6, 7, 4, 1 ···)

The chunk numbers are assigned to the targets of the same layer as the chunk #0 in each IO server, and in the next cycle, the numbers are assigned to the targets of the next layer.

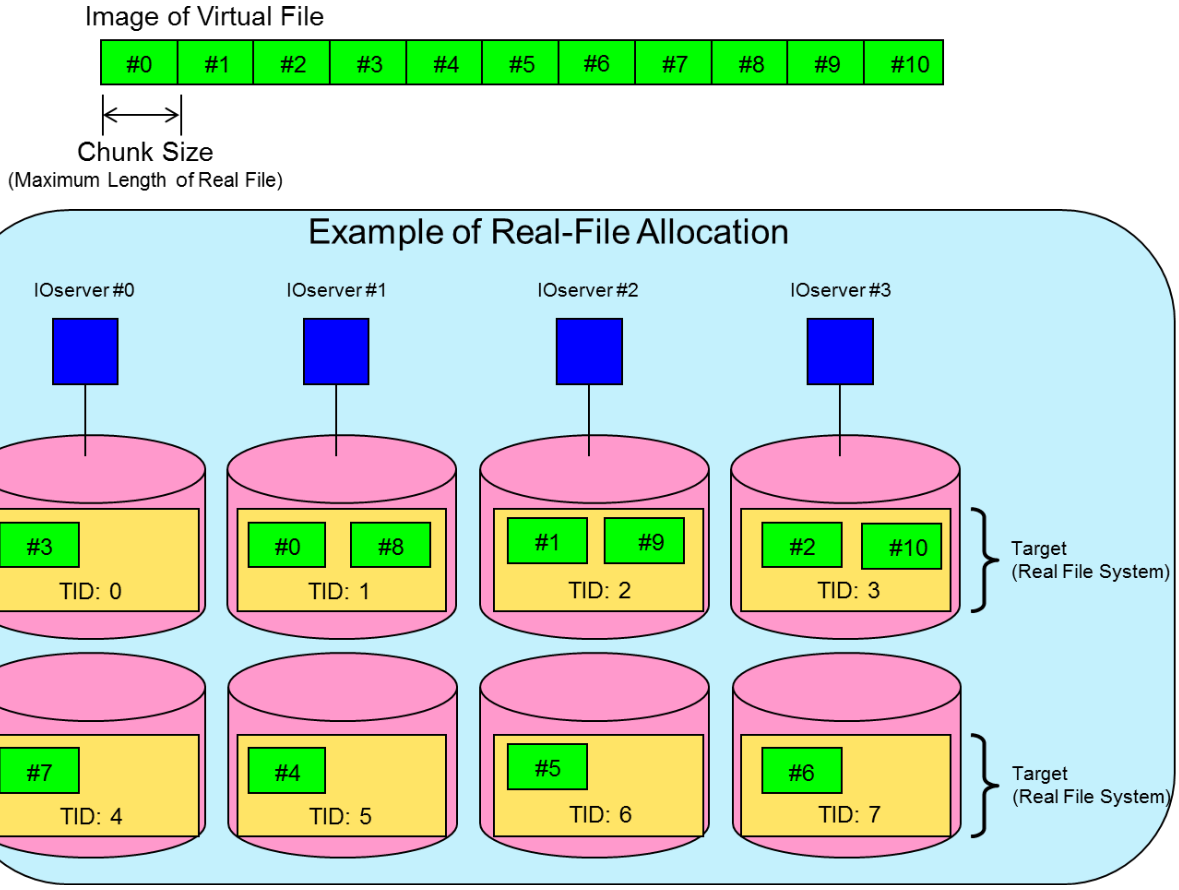


Figure 11-2 Relation between the virtual file and real files in Format 1

11.2.2 Stripe format (Format 2)

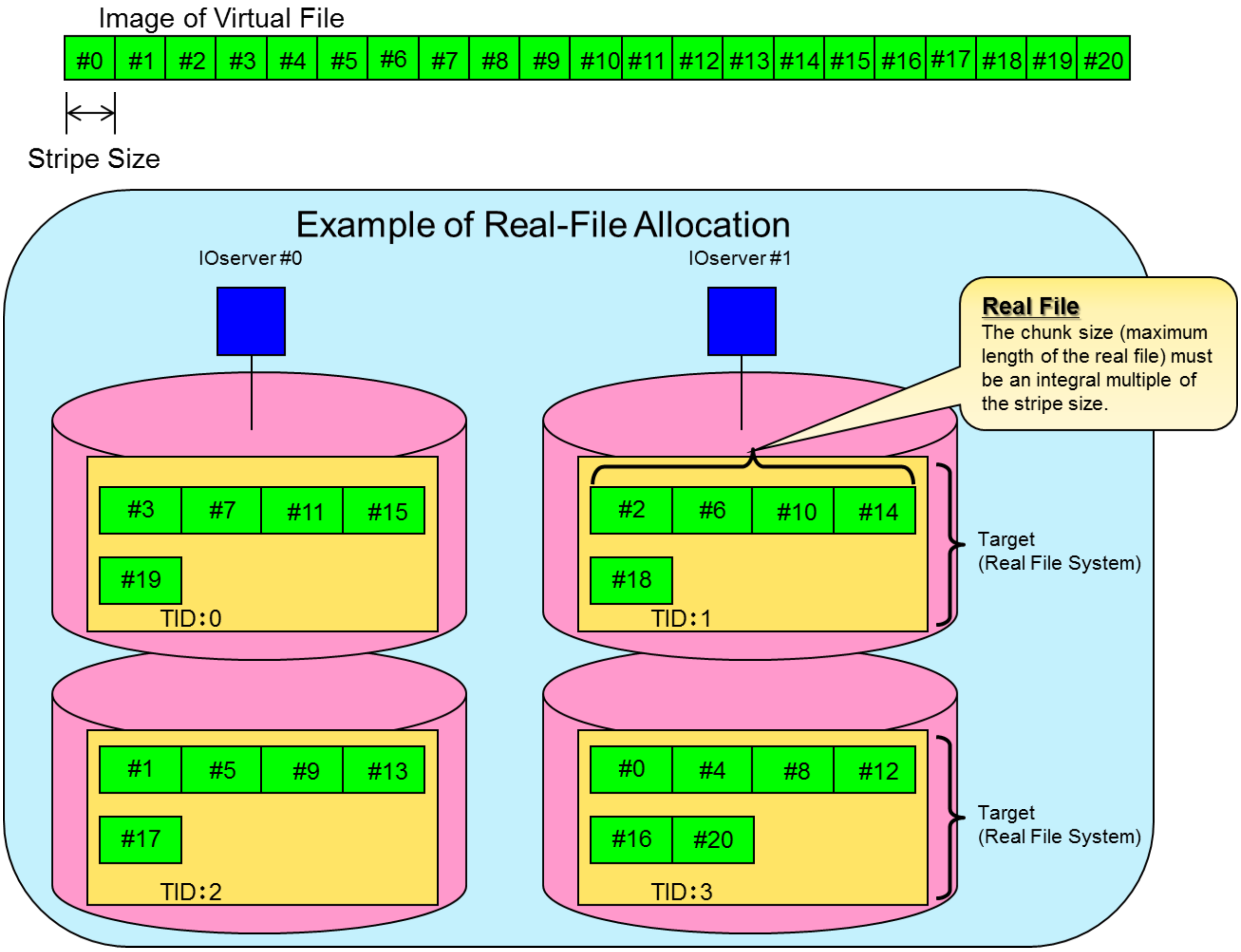
This format is useful for improving a single I/O processing because requests can be issued to multiple IO servers from a specific node simultaneously. The example in Figure 11-3 can be effective when calling read/write system calls with the size 2 or 4 times the stripe size because there are 2 IO servers, each of which has two targets. That is, read/write can be done to #0 and #1 or #0, #1, #2, and #3 of the virtual file almost simultaneously. Note that the operation is restricted by the bandwidth of the network interfaces of nodes (clients) used by ScaTeFS.

For parallel I/O (described later), a conflict may occur between nodes due to updating/referring to different offsets in the same real file.

Like shown in Figure 11-3, the stripe size is a basic unit of virtual file configuration, and the chunk size needs to be a multiple of the stripe size. Since the default file format is Format 1, use scatefs_premap(1) to specify the stripe size and chunk size explicitly in order to use Format 2. In the example in Figure 11-3, 2 real file systems (targets) are created for each IO server that makes up a single ScaTeFS. The virtual file in the figure comprises chunk #0 through #20, the head of the virtual file, chunk #0, is assigned to TID:3. Then, the chunk numbers are consecutively distributed in the following TID order:

TID = (3, 2, 1, 0, 3, 2, 1, 0 ...)

The chunk numbers are assigned to the targets of the same layer as the chunk #0 in each IO server, and in the next cycle, the numbers are assigned to the targets of the next layer. When the size of a real file reaches the chunk size, a new file is created in the same target.



Real File
The chunk size (maximum length of the real file) must be an integral multiple of the stripe size.

Figure 11-3 Relation between the virtual file of Format 2 and real files

11.3 Parallel I/O

The term parallel I/O as used in this document means to perform write and read operations on a file by transferring data in parallel using multiple computing nodes. The main purpose is to increase the I/O efficiency for large-scale files. Figure 11-4 is the simplest example of parallel I/O.

To achieve parallel I/O performance consistent with concurrency, you need to consider the number of IO servers and the number of IO targets, and then determine the format (Format 1/Format 2) and chunk size of the virtual file targeted for parallel I/O. This is to ensure that conflict does not occur between IO servers and storage devices.

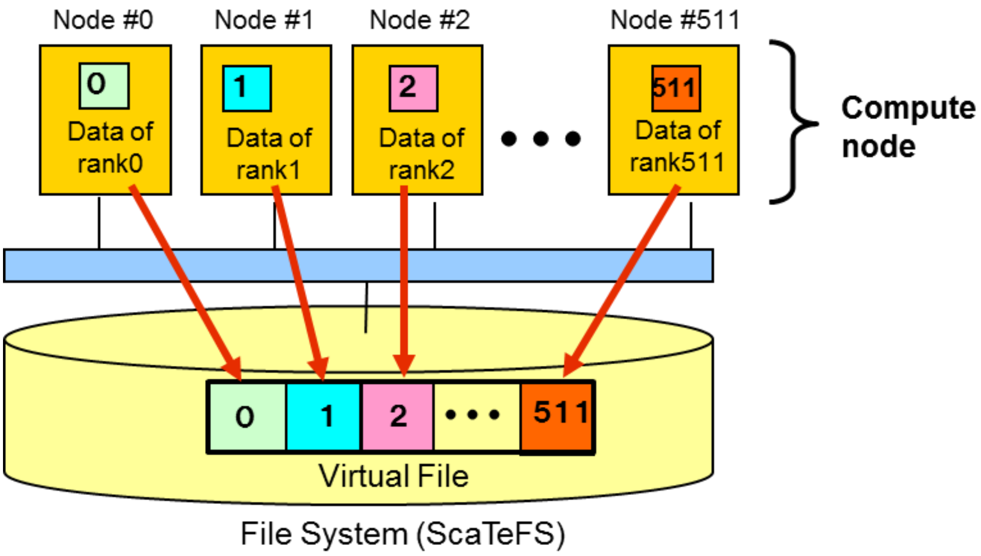


Figure 11-4　Image of the parallel I/O based on Format 1

11.4 Optimizing parallel I/O (premapping files)

Assume you perform 'write' operations from 512 nodes simultaneously to create a virtual file consisting of 512 real files as shown in Figure 11-4. In this case, 512 real files are generated almost simultaneously, and the update of management information of the virtual file can cause some overhead. The premap function is provided to reduce the overhead by generating a necessary number of real files before the 'write' operations.

This function can be executed by specifying a file size when specifying a file format (fcntl(2) of SUPER-UX or scatefs_premap(1)) which is described later. For details, see fcntl(2) and scatefs_premap(1) of SUPER-UX.

11.5 Setting and showing the file format

To set the file format, use scatefs_premap(1) if the target is a file and use scatefs_setdirattr(1) if the target is a directory. To confirm the file format, use scatefs_getinfo(1). Examples are shown below.

11.5.1 Setting up the non-stripe format (format 1)

- File

  Specify the -c option and file size for scatefs_premap(1) to create a file with Format 1. The example performs premapping with the chunk size of 2 G and the file size of 4 G.

Example:

```
$ scatefs_premap -c 2G 4G /mnt/scatefs/file000
```

To create a file with only the file format specified, specify 0 for the file size.

Example:

```
  $ scatefs_premap -c 2G 0 /mnt/scatefs/file001
```

- Directory

  Specify the -c option for scatefs_setdirattr(1) to change the format of the existing directory to Format 1. Then, the change is applied to newly created files and directories created under the directory. It is not applied to existing files and directories. In the example, specify 4 G for the chunk size.

  Example:　Directory

```
$ scatefs_setdirattr -c 4G /mnt/scatefs/dir000
```

11.5.2 Setting up the stripe format (format 2)

- File

  Specify the -s option for scatefs_premap(1) to create a file with Format 2. The example performs premapping with the stripe size of 4 M, the chunk size of 1 G, and the file size of 1 G. When specifying an existing file, premapping is only available if the file size is 0.

Example:

```
$ scatefs_premap -s 4M -c 1G 1G /mnt/scatefs/file002
```

- Directory

  Specify the -s option for scatefs_setdirattr(1) to change the format to Format 2. Then, the change is applied to newly created files and directories created under the directory. It is not applied to existing files and directories. The example changes the stripe size and chunk size of the existing directory to 4 M and 1 G, respectively.

Example:

```
$ scatefs_setdirattr -s 4M -c 1G /mnt/scatefs/dir001
```

11.5.3 Setting from the system call

The example below shows how to specify the file format using fcntl(2) of SUPER-UX. Setting from the system call can only be executed on the SUPER-UX. It cannot be executed on the Linux client.

- File

  Perform open(2) for the file to premap, and specify a value for each member of the scfs_premap structure. Specify the file descriptor for the first argument, F_SCPREMAP for the second argument, and the address of the scfs_premap structure for the third argument.

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/fcntl.h>
 :
int main (int argc, char *argv[])
{
int fd;
char *filepath;
struct scfs_premap p;

fd = open(filepath, O_RDWR);
/* Specify a value */
p.stripesize = stripesize;
        p.chunksize = chunksize;
        p.filesize = filesize;

        /* Call fcntl(2) */
        fcntl(fd, F_SCPREMAP, &p);
return 0;
}
```

The chunk size and stripe size need to be specified in units of 4 K.

For Format 1, specify the same value for the chunk size and the stripe size.

For Format 2, the chunk size must be a multiple of and greater than the stripe size.

To create a file with only the format specified, set the file size to 0.

- Directory

  Perform open(2) on the directory for which to change the format, and specify values for members of the scfs_setdirattr structure. Specify the file descriptor for the first argument, F_SCSETDIRATTR for the second argument, and the address of the scfs_setdirattr structure for the third argument.

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/fcntl.h>
        :
int main (int argc, char *argv[])
{
int fd;
char *dirpath;
struct scfs_setdirattr d;

fd = open(dirpath, O_RDONLY);

/* Specify a value */
d.stripesize = stripesize;
d.chunksize = chunksize;

/* Call fcntl(2) */
fcntl(fd, F_SCSETDATTR, d);
return 0;
}
```

The chunk size and stripe size need to be specified in units of 4 K.

For Format 1, specify the same value for the chunk size and the stripe size.

For Format 2, the chunk size must be a multiple of and greater than the stripe size.

11.5.4 Displaying the format

Use scatefs_getfinfo(1) to show the format information of the file/directory.

- File

Example:　Format 1

```
$ scatefs_getfinfo /mnt/scatefs/file001
format      ：non stripe format
iot count   ：              6
stripesize ：         268435456
chunksize   ：          268435456
filesize    ：        1610612736

format:  ...........................................................................File format
iot count:  ........................................................................Number of used IO target
stripesize:  ......................................................................Stripe size
chunksize:  .......................................................................Chunk size
filesize:  ...........................................................................File size
```

* In case of Format 1, the stripe size and chunk size are the same.

Example:　Format 2

```
$ scatefs_getfinfo /mnt/scatefs/file002
format      ：    stripe format
iot count   ：              6
stripesize ：          33554432
chunksize   ：           67108864
filesize    ：         268435456
```

Specify the -v option to show the real file distribution for each file offset. For a file with Format 1 or Format 2, information like the example below is shown.

- Displaying the file details

  Example:　Format 1

```
$ scatefs_getfinfo -hv /mnt/scatefs/file001
format       : non stripe format
iot count   :                     6
stripesize :                 256.0M
chunksize   :                  256.0M
filesize     :                  1.5G

         offset                                         no          ios         iot
------------------------------------------------------------
                      0 ...              268435455          0            0            0
          268435456 ...              536870911          1            1            3
          536870912 ...              805306367          2            0            1
          805306368 ...            1073741823          3            1            4
        1073741824 ...            1342177279          4            0            2
           1342177280 ...            1610612735          5            1            5

offset:  ..................................................................................Indicates the offset of the virtual file.
no:  .......................................................................................Indicates the real file index.
ios:  ......................................................................................Indicates the IO server ID where the real file is stored.
iot:  ......................................................................................Indicates the IO target ID where the real file is stored.
```

For Format 1, the offset and the real file index are identical. The image of real file allocation is shown below.

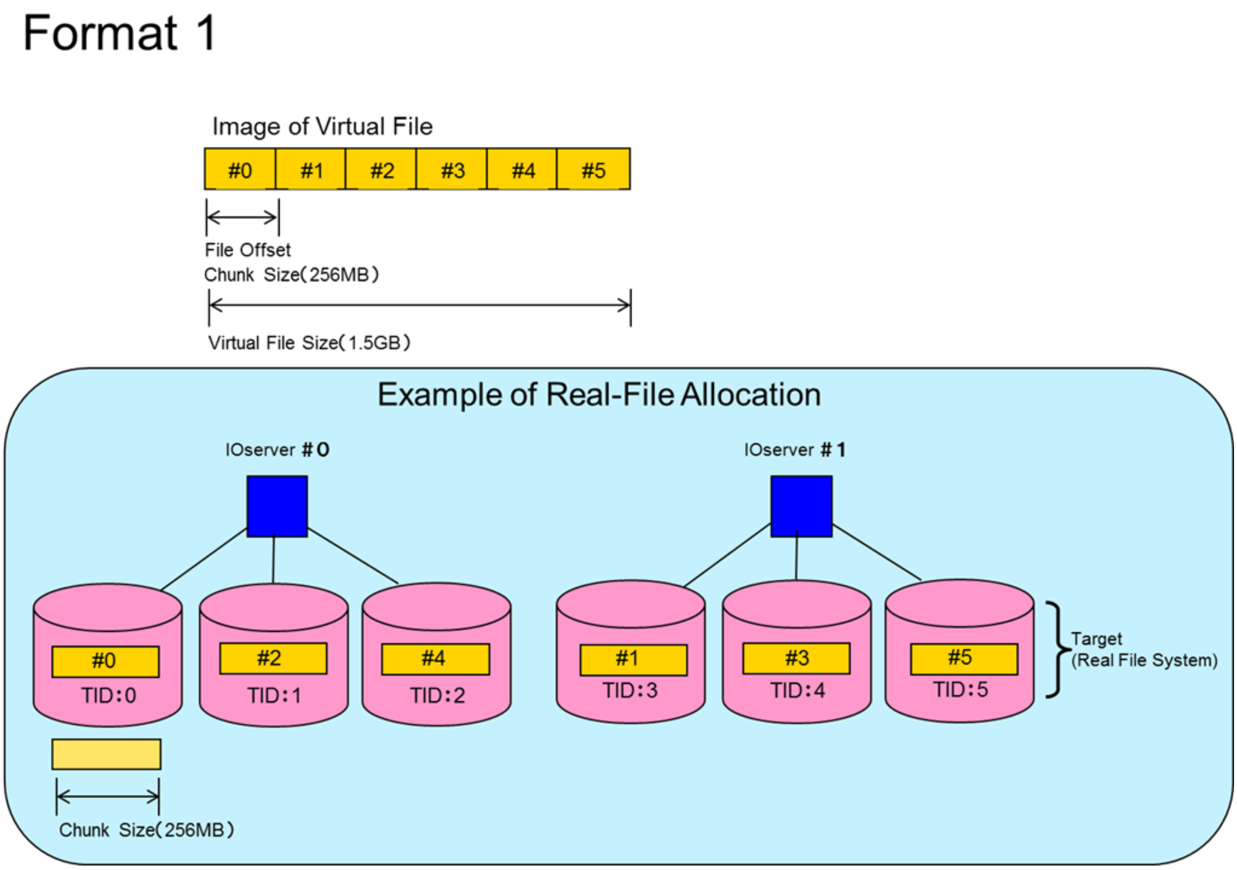


Figure 11-5　Example of real file arrangement in Format 1

Example: Format 2

```
$ scatefs_getfinfo -hv /mnt/scatefs/file002
format        :         stripe format
iot count     :                    6
stripesize :                   32.0M
chunksize     :                  64.0M
filesize      :                 256.0M

        offset                              no       ios      iot
------------------------------------------------------------------
                0 ...         33554431       0        0        0
        33554432 ...          67108863       1        1        3
        67108864 ...         100663295       2        0        1
       100663296 ...         134217727       3        1        4
       134217728 ...         167772159       4        0        2
       167772160 ...         201326591       5        1        5
       201326592 ...         234881023       0        0        0
       234881024 ...         268435455       1        1        3
```

For Format 2, the real file index corresponding to the offset delimited by the stripe size is shown. The image of real file allocation is shown below.

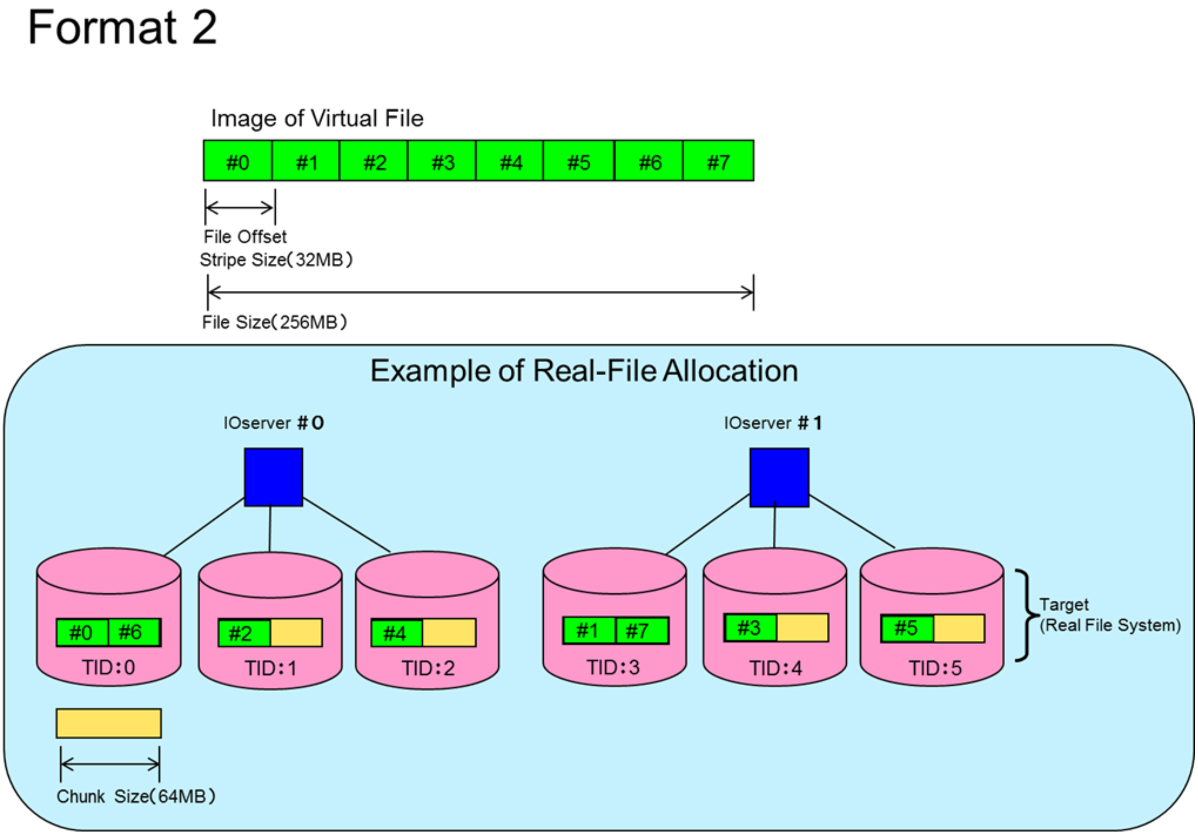


Figure 11-6　Example of real file arrangement in Format 2

- Directory

Example:　Format 1

```
$ scatefs_getfinfo -h /mnt/scatefs/dir001
format       : non stripe format
stripesize :              512.0M
chunksize   :               512.0M
```

* In case of Format 1, the stripe size and chunk size are the same.

Example:　Format 2

```
$ scatefs_getfinfo -h /mnt/scatefs/dir002
format       :      stripe format
stripesize :               32.0M
chunksize   :                  1.0G
```

*Detailed display option(-v) that target directory is invalid.

11.6 How to use ScaTeFS InfiniBand high performance library

11.6.1 How to use ScaTeFS IB Library

You can use ScaTeFS IB Library by executing a program with setting following environment variables. It is not recommended that these variable are set in .bashrc or .cshrc. Set them in a command line or a job script as following examples.

- LD_PRELOAD

  Specify the library path (/lib64/libscatefsib.so.1).

  You can use lightweight and high performance IO function by IB through a user space without modifications of user programs.

- SCATEFS_LOG_DIR

  Specify an absolute directory path which is put the log file of library.

  The directory should have write access for the user who executes a program and be created before a program execution.

  If you execute a program without specifying SCATEFS_LOG_DIR, the log file will put a current directory of the executed program. In this case, it might be hard for you to find it. So you should specify SCAETFS_LOG_DIR.

  A log file isn't created in a regular case. Only when an incident which should be investigated happens, it is created. You shouldn't remove it for a following investigation. The log file name is libscatefsib.<PID of an executed program>.

A setting method of an environment variable is different depending on execution methods of a program. The followings are setting examples of the environment variables when executing a cp(1) command by each execution methods.

- Executing on command line

  Specify the environment variables on command line.

```
$ LD_PRELOAD=/lib64/libscatefsib.so.1 SCATEFS_LOG_DIR=/home/user/log cp fileA fileB
```

- Executing in shell script

  Specify the environment variables in shell script.

  For a MPI job script of NQSV, the setting method is different from this. See the next item.

```
#!/bin/bash
export LD_PRELOAD=/lib64/libscatefsib.so.1
export SCATEFS_LOG_DIR=/home/user/log
cp fileA fileB
```

- Executing in MPI job script of NQSV

  Specify the environment variables with –x option of mpirun command.

  Even if you specify them like “export LD_PRELOAD=/lib64/libscatefsib.so.1” in a script, the setting will not be transferred to the slave node. So you should specify with –x option of mpirun command.

  The example script image of executing sample program is shown below.

  Refer to NQSV manual about setting “#PBS” and the environment variable NQSII_MPIOPTS.

```
#!/bin/bash
#PBS -T openmpi
#PBS -b 2
#PBS -l cpunum_job=4
#PBS -l elapstim_req=3600
mpirun ${NQSII_MPIOPTS} -npernode 1 -np 2 -x LD_PRELOAD=/lib64/libscatefsib.so.1 ¥
-x SCATEFS_LOG_DIR=/home/user/log /home/user/sample
```

11.6.2 How to use ScaTeFS VE direct IB library

You can use ScaTeFS VE direct IB Library by executing a program with setting following environment variables. It is not recommended that these variable are set in .bashrc or .cshrc. Set them in a command line or a job script as following examples.

- VE_LD_PRELOAD

  Specify the library name (libscatefsib.so.1).

  You can use lightweight and high performance IO function by IB through a user space without modifications of user programs.

- SCATEFS_LOG_DIR

  Specify an absolute directory path which is put the log file of library. Same as ScaTeFS IB library for setting note and output file. See the 10.6.1.

A setting method of an environment variable is different depending on execution methods of a program. The followings are setting examples of the environment variables when executing a program a.out by each execution methods.

- Executing on command line

Specify the environment variables on command line.

```
$ VE_LD_PRELOAD=libscatefsib.so.1 ./a.out
```

- Executing in shell script

  Specify the environment variables in shell script.

```
#!/bin/bash
export VE_LD_PRELOAD=libscatefsib.so.1
export SCATEFS_LOG_DIR=/home/user/logdir
./a.out
```

- Executing in MPI job script of NQSV

  Specify the environment variables in MPI job script same as above “Executing in shell script”.

  You have to specify the number of needed HCAs to “--use-hca”. If not, the IO will fail with an error.

  Refer to NQSV manual about setting “#PBS”.

```
#!/bin/sh
#PBS -T necmpi
#PBS -b 2
#PBS --venum-lhost=1
#PBS --use-hca=2
export VE_LD_PRELOAD=libscatefsib.so.1
export SCATEFS_LOG_DIR=/home/user/logdir

mpirun -ppn 1 mpi_prog
```

11.6.3 Programing tips

Programing tips for ScaTeFS VE direct IB library as follows.

- Programing tips for optimal IO performance

Recommend large size read(2)/write(2) (1MB or more).

You can expect optimal performance by calling large size read(2)/write(2)(1MB or more) a few times, not by calling small read(2)/write(2) many times.

Rather than many call small size read(2)/write(2),　Few call large size read(2)/write(2) as much as 1MB or more will be optimal performance.

Not recommend calling unnecessary stat systemcall(stat(2)/lstat(2)/fstat(2))

When calling read(2)/write(2) many times repeatedly, calling stat systemcall between read(2)/write(2) is not recommended. You can expect optimal performance by reducing stat systemcall.

- Ensure consistency of file data between VE and VH(or between different VEs).

  When you want to ensure consistency of file data accessing to same file by processes on VE and VH(or between different VEs), You must use file lock(flock(2),F_SETLK of fcntl(2)). This tips is same as accessing to same file from different clients on NFS.

11.6.4 Setting of environment variables for performance tuning

You can tune data transfer size with following environment variables. When a buffer size specified as an argument of read(2)/write(2) is large, IOs will be processed efficiently by extending data transfer size and you can expect an IO performance improvement.

However, because a load of IO servers per an IO request increases by extending it, IO performance might be degraded in case executing IOs simultaneously by many processes. So it is recommended that you set theses environment variables to same values as rsize/wsize specified as the mount command options.

These environment variables influence only processes using ScaTeFS IB Library. When you execute a program without ScaTeFS IB Library, IO transfer sizes depend on rsize/wsize specified as the mount command options.

Table 11-1 rsize/wsize option overview

Setting value	Description	Minimum	Maximum	Default
SCATEFS_WSIZE	Data transfer size for WRITE (KB)	4	4096	1024
SCATEFS_RSIZE	Data transfer size for READ (KB)	4	4096	1024

You can turn on or off the mode which can detect response from IO server rapidly with the following environment variable. You can expect the performance improvement of a program which calls read(2)/write(2) frequently. But the CPU usage during IO enabled this mode is higher than the case disabled it.

Table 11-2 cqpollhow option overview

Setting value	Description	Minimum	Maximum	Default
SCATEFS_CQPOLLHOW	The switch of the mode detecting response from IO server rapidly. ON:0, OFF:1	0	1	1

11.6.5 Performance improvement with stripe format

When an IO area specified as an argument of read(2)/write(2) stretches over more than 2 stripes (or chunks), ScaTeFS IO library issues IO requests to multiple IO servers simultaneously. Figure 11-7 shows the image which IO requests are issued to multiple IO servers simultaneously. You can expect IO performance improvement with stripe format because IOs is processed efficiently. When you tune a stripe size or an IO size in your application so that IO size is larger than a stripe size, your application can issue IO requests to multiple IO servers simultaneously. The optimum IO size is just a multiple of a stripe size.

However, when setting too small striping size, IO performance might be degraded because a data amounts per request becomes small and IOs are processed inefficiently. So it is recommended that you specify a stripe size larger than 1MB or equal.

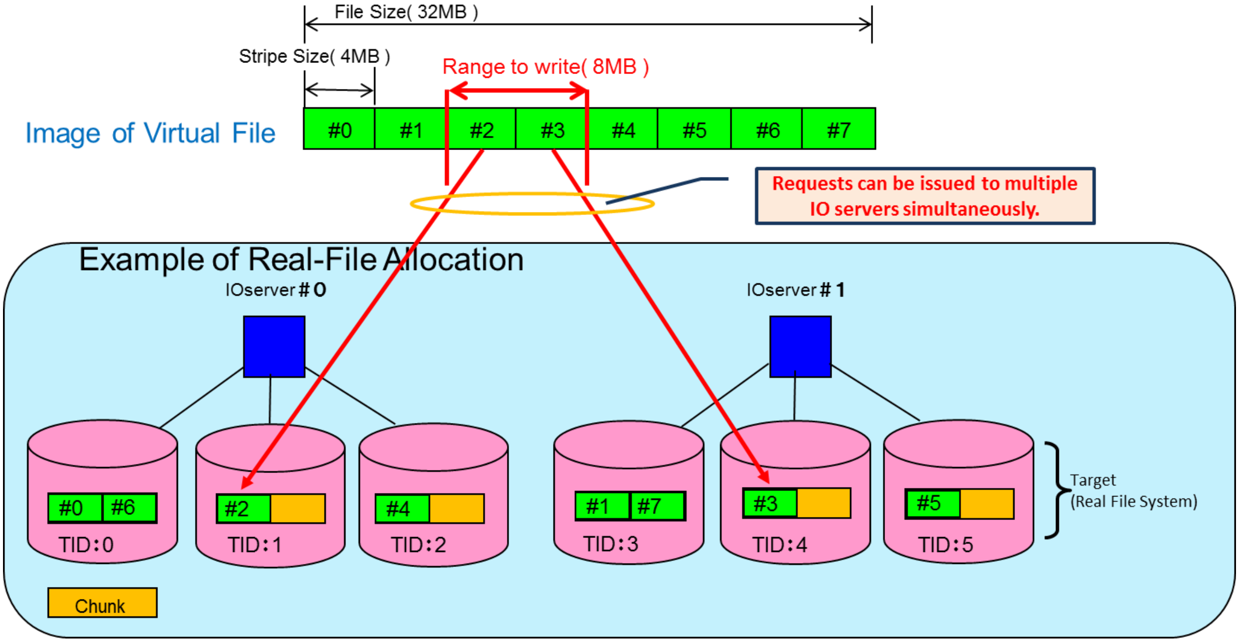


Figure 11-7　IO to files with stripe format

11.6.6 Performance tuning for program of NEC Fortran

This is the performance tuning when using ScaTeFS VE direct IB library.

When your program executes READ/WRITE statement for a small record which is less than 512KB many times, you can expect a performance improvement by extending the I/O buffer size (VE_FORT_SETBUF) to the same size as the data transfer size (SCATEFS_RSIZE, SCATEFS_WSIZE). When SCATEFS_RSIZE and SCATEFS_WSIZE are set different value, set VE_FORT_SETBUF to the largest value in SCATEFS_RSIZE and SCATEFS_WSIZE.

You can set the I/O buffer size with the environment variable VE_FORT_SETBUF. For the detail of VE_FORT_SETBUF, see "SX-Aurora TSUBASA Fortran Compiler User's

Guide". For the setting and the default value of the data transfer size, see 10.6.4. When your program handles mainly large records that are larger than 512KB like softwares generally used in HPC world, you don't need to change the setting of VE_FORT_SETBUF.

11.6.7 Statistics

You can get a statistics file for a process using the library by setting the environment variable SCATEFS_STATISTICS_ON to 1. The directory to be put it can be specified with the environment variable SCATEFS_STATISTICS_DIR. When you specify SCATEFS_STATISTICS_ON without SCATEFS_STATISTICS_DIR, a statistics file will be put on a current directory of the executed process. The following the example in case specifying them in a command line.

```
# LD_PRELOAD=/lib64/libscatefsib.so.1 SCATEFS_STATISTICS_ON=1   ¥
SCATEFS_STATISTICS_DIR=/home/user/log/ dd if=/dev/zero of=/mnt/scatefs/testfile bs=1M count=1
```

The statistics file name is libscatefs_stat.<PID>. You can see statistics by scatefs_ibstat(1) specifying a statistics file as argument. The following example shows that the data of 1048576 bytes (SIZE_TOTAL) was written. See man of scatefs_ibstat(1) for more information about scatefs_ibstat.

```
# scatefs_ibstat ./stat/libscatefs_stat.9012
Pid: 9012
Time: Tue Jul 19 10:41:19 2016
REQUEST         COUNT TAT_TOTAL TAT_AVE    SIZE_TOTAL SIZE_AVE      OK        NG
WRITE               1         2       2       1048576  1048576       1         0
READ                0         0       0             0        0       0         0
COMMIT              1         8       8             0        0       1         0
write(libc)      0         0       0             0        0       0         0
read(libc)       0         0       0             0        0       0         0
```

When an IO size specified as an argument of read(2)/write(2) is less than 1MB, an IO method is changed into Kernel IO automatically. In that case, the IO information was counted in “write(libc)” as below.

```
# scatefs_ibstat ./stat/libscatefs_stat.9015
Pid: 9015
Time: Tue Jul 19 10:46:50 2016
REQUEST         COUNT TAT_TOTAL TAT_AVE    SIZE_TOTAL SIZE_AVE      OK        NG
WRITE               0         0       0             0        0       0         0
```

```
READ              0         0       0             0        0       0        0
COMMIT             0         0       0             0        0       0        0
write(libc)     1         1       1       1048575   1048575      1        0
read(libc)      0         0       0             0        0       0        0
```

11.6.8 Dealing with job failure

In case using “disk sync on close mode”, a job fails with an error(ETIMEDOUT) of read(2), write(2) or close(2) when a failover of IO server occurs by an IO server trouble. And the following message is output to standard error output or standard error output file of NQSV.

```
ScaTeFS failed to write: process(/bin/cp)   file(4362917)
```

*4362917 is the file's inode number.

When this error occurs, a job's write data might not be synced to disk correctly. So rerun jobs faced this error.

11.6.9 Memory usage

Compared to a process not using InfiniBand high performance library, a process using the library needs the additional memory usage specified in the following table. A process which issues IO to SFA7990XE needs more memory than a process which issues IO to Express5800 IO server because of the high performance IO function taking advantage of the features of SFA7990XE. A process which issues IO to both Express5800 IO server and SFA7990XE needs same memory usage as a process which issues IO only to SFA7990XE.

Table 11-3 Additional memory usage when using the library

The case using Express5800 IO server	The case using SFA7990XE
200MB	460MB

When using ScaTeFS IB Library, a process uses the memory on the scalar machine including VH. When using ScaTeFS VE direct IB library, a process uses the memory on the VE.

Chapter12 Specification

Table 12-1　Specification

Item	Maximum number
Maximum number of IO servers that can be included in one file system	256 (128 pairs)
Maximum number of IO targets that can be included in one file system (Number of real file systems)	1024
Maximum number of file systems that can be created in the same system	20
Maximum file size	64PB (Assuming that the chunk size is 4 GB and the file format is format 1.)
Maximum file system size	500PB (Assuming that the number of IO targets is 1024.)
Maximum number of files	2 trillion files (Assuming that the number of IO targets is 1024.)
Maximum number of a directory entries	no limit (5 million confirmed.)
Maximum length of a filename	255 bytes
Maximum length of a pathname	1024 bytes
Maximum number of processes using ScaTeFS InfiniBand high performance library simultaneously for one file system	Around 35,000 processes It is limited by the HCA resources.　It might be changed depending on the usage of the HCA resources used by other programs using IB.
Maximum number of processes using ScaTeFS InfiniBand high performance library simultaneously in one client	Around 900 processes ● In case of SX-Aurora TSUBASA, it is the maximum number of the total processes running on VE and VH in one client. ● It is limited by the HCA resources.　It might be changed depending on the

Item	Maximum number
	usage of the HCA resources used by other programs using IB.

Appendix A Procedure for Creating EXPRESSCLUSTER Cluster Configuration Information (Offline version)

This procedure describes how to create EXPRESSCLUSTER cluster configuration information using the following EXPRESSCLUSTER tool before configuring the IO servers.
[IO server v4+ for standard model or later]
  EXPRESSCLUSTER X Cluster WebUI Offlime
[IO server v1, v3 and v4 for standard model]
  EXPRESSCLUSTER X builder (Offline version)

The process described in this procedure is the EXPRESSCLUSTER cluster configuration information creation process described in “5.4.1.2 Transferring the cluster configuration information file to the work PC”. After creating this EXPRESSCLUSTER cluster configuration information, configure the EXPRESSCLUSTER settings described in “5.4.1.2 Checking the network settings of the ports for connecting the IO servers” and later sections.

Create the EXPRESSCLUSTER configuration information as follows:
1. Install the EXPRESSCLUSTER tool.
2. Start the EXPRESSCLUSTER tool.
3. Create the cluster configuration information.
   3.1 Create clusters
   3.2 Create failover groups
   3.3 Create monitor resources
   3.4 Configure the recovery action settings for when a monitor resource error occurs
   3.5 Change the cluster properties

This procedure serves as a supplement to the EXPRESSCLUSTER X for Linux Installation and Configuration Guide, so refer to this guide also where appropriate. This procedure references the following sections of the above guide:
[EXPRESSCLUSTER X Cluster WebUI Offlime]
Manual(1): 6.4 Creating the configuration data of a 2-node cluster

Manual(2): 6.11 Saving the cluster configuration data
[EXPRESSCLUSTER X builder (Offline version)]
Manual(1): “Installing the Builder (Offline version)” in “Chapter 3 Installing EXPRESSCLUSTER”
Manual(2): “Creating the configuration data of a 2-node cluster” in “Chapter 5 Creating the cluster configuration data”
Manual(3): “Saving the cluster configuration data” in “Chapter 5 Creating the cluster configuration data”

The following EXPRESSCLUSTER tools are used in this procedure.
Please download EXPRESSCLUSTER tool the from the site of EXPRESSCLUSTER.
[EXPRESSCLUSTER X Cluster WebUI Offlime]
4.3.2-210913-1
[EXPRESSCLUSTER X builder (Offline version)]
Note that the default values might differ depending on the builder version, so do not change builders halfway through the process.
expressclsbuilder-3.2.0-1.linux.i686.exe
expressclsbuilder-3.3.5-1.linux.i686.exe

A.1 Introduction

Before starting this process, you need to determine the following information:

- The IO server names
- The IP addresses of ports connecting the two IO servers that configure the cluster
- The floating IP address (FIP) of the file server port (10GbE, IB)
- Each resource name

See Attachment for details of resources and their correspondence. The resource names described in the correspondence table are examples of names that comply with the recommended naming rules. Unless there is a specific reason for not doing so, use these naming rules to determine the names of the resources in your system.
Note that the device name of the partition for the heartbeat region must also be determined. However, because this name is determined after PathManager is installed, it is set in the process described in “5.4.4 Cluster properties”, after the process described in this procedure is complete.

A.2 Installing the EXPRESSCLUSTER tool

[EXPRESSCLUSTER X Cluster WebUI Offlime]

See in the "Cluster WebUI Offline Setup Guide" from the site of EXPRESSCLUSTER.

[EXPRESSCLUSTER X builder (Offline version)]

See in the manual (1).

A.3 Start the EXPRESSCLUSTER tool

[EXPRESSCLUSTER X Cluster WebUI Offlime]

See in the "Cluster WebUI Offline Setup Guide" from the site of EXPRESSCLUSTER.

[EXPRESSCLUSTER X builder (Offline version)]

See in the manual (1).

A.4 Create the cluster configuration information

Create the cluster configuration information by using the Cluster generation wizard.

[EXPRESSCLUSTER X Cluster WebUI Offlime]

See in the manual (1).

[EXPRESSCLUSTER X builder (Offline version)]

See in the manual (2).

Use the default values for items that are not described in the following procedure.

A.5 Create clusters

Create clusters.

A.6 Add clusters

[EXPRESSCLUSTER X Cluster WebUI Offlime]

Click Cluster generation wizard to start the wizard. Leave the default settings as they are on the Cluster generation wizard screen and click Next.

[EXPRESSCLUSTER X builder (Offline version)]

Open the File menu and click Cluster generation wizard. A confirmation dialog box is displayed. Click Start the Standard Cluster generation wizard. Leave the default settings as they are on the Cluster generation wizard screen and click Next.

A.7 Add servers

Add the two IO servers configuring the cluster to the server definitions on the Cluster generation wizard screen. In the following description, the names of the IO servers are iosv00 and iosv01.

Click Add next to the Server Definitions list.

Set the following item on the Add new server screen. This server will be the master server.

Server name

iosv00

Click Add next to the Server Definitions list again.

Set the following item on the Add new server screen.

Server name

iosv01

A.8 Configuring the network

Configure the network connecting the IO servers that configure the cluster.

Click Add next to the Interconnects list.

Specify an item on the Priority 1 line.

Type

Kernel mode

iosv00

IP address of the port connecting the IO servers

iosv01

IP address of the port connecting the IO servers

Click Add next to the Interconnects list.

Specify an item on the Priority 2 line.

Type

DISK

iosv00

Device name of the partition for the EXPRESSCLUSTER heartbeat region

iosv01

Device name of the partition for the EXPRESSCLUSTER heartbeat region

* Specify the device name of the partition for the heartbeat region after completing the

process described in this procedure.

A.9 Configuring the network partition resolution processing (NP resolution)

Move to the next step with do nothing.

A.10 Create failover groups

Create the failover groups that will run on the IO servers that configure the cluster. In the following description, the failover group that runs on iosv00 is called failover1, and the failover group that runs on iosv01 is called failover2.

Failover group failover1 that runs on iosv00 is created first, so carry out steps A.11 to A.15. Once failover1 is created, repeat these steps to create failover group failover2 that runs on iosv01.
If there are items that require different settings for failover groups failover1 and failover2, the settings are described separately under the headings [failover1] and [failover2].

A.11 Add failover groups

Click Add next to the Groups list on the Groups screen.

Configure the following items of the Group Definitions screen:

Name

[failover1]

failover1

[failover2]

failover2

On the List of Bootable Servers screen, clear the Failover OK check box of each server.

From the available servers, select the IO servers in the following order and click Add.

* Servers must be added in the right order.

[failover1]

iosv00

iosv01

[failover2]

iosv01

iosv00

On the Configure Group Attributes screen, change the following items from their default values:

Failback attribute

Auto failback

A.12 Add group resources (Floating IP resource)

Configure the IP addresses of the file system ports used to configure the IO server network (10GbE, IB).

The number of the resources to add differs depending on the number of file system ports. In the following example, it is assumed that both 10GbE and IB are used. In case using 4 FIPs(fip1, fip2, fip3, fip4) on 10GbE and 2 FIPs(fip_ib1, fip_ib2) on IB, fip1, fip2, fip_ib1 are added to [failover1] and fip3, fip4, fip_ib2 are added to [failover2].

(When using only 10GbE, you need to add only fip1, fip2, fip3, fip4 in the following steps. When using only IB, you need to add only fip_ib1, fip_ib2.)

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

Floating IP resource

Name

[failover1]

fip1

fip2

fip_ib1

[failover2]

fip3

fip4

fip_ib2

Leave the default settings are they are on the Dependency screen and move to the next step.

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Common tab of the Details screen:

IP Address

IP address of the file system port used to configure the IO server network (10GbE, IB)

Example:

10.0.1.1/25%bond0.12

Click Tuning button.

Configure the following items on the Parameter tab of the Floating IP Resource Tuning Properties:

This configuration is needed when using bonding interface only.

Judge NIC Link Down as abnormal

select the check box

A.13 Add group resources (Volume manager resource)

Add VGs designed for an LVM configuration as group resources. Specify the VGs used by the IO servers for IO target data and metadata regions.

Create the same number of group resources as IO targets (data and metadata regions).

Click Add next to the Group resources list on the Group resources screen.

Type

Volume manager resource

Name

[failover1]

volmgr_d_01,volmgr_d_02,・・・,volmgr_d_[n]

volmgr_c_01,volmgr_c_02,・・・,volmgr_c_[n]

[failover2]

volmgr_d_[n+1],volmgr_d_[n+2],・・・,volmgr_d_[n+n]

volmgr_c_[n+1],volmgr_c_[n+2],・・・,volmgr_c_[n+n]

*[n] indicates the number of VGs per IO server designed for an LVM configuration.

Configure the following items on the Dependency screen:

Apply default dependency

Clear the check box.

From the available resources, select a floating IP address resource and click Add.

[failover1]

fip1,fip2, fip_ib1

[failover2]

fip3,fip4, fip_ib2

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Common tab of the Details screen:

Target name

[failover1]

vg_data01,vg_data02,・・・,vg_data[n]

vg_ctrl01,vg_ctrl02,・・・,vg_ctrl[n]

[failover2]

vg_data[n+1],vg_data[n+2],・・・,vg_data[n+n]

vg_ctrl[n+1],vg_ctrl[n+2],・・・,vg_ctrl[n+n]

A.14 Add group resources (Disk resource)

Add the resources for mounting and unmounting IO target devices.

Create the same number of group resources as IO targets (data and metadata regions).

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

Disk resource

Name

[failover1]

disk_d_01,disk_d_02,・・・,disk_d_[n]

disk_c_01,disk_c_02,・・・,disk_c_[n]

[failover2]

disk_d_[n+1],disk_d_[n+2],・・・,disk_d_[n+n] disk_c_[n+1],disk_c_[n+2],・・・,disk_c_[n+n]

*[n] indicates the number of VGs per IO server designed for an LVM configuration.

Configure the following items on the Dependency screen:

Apply default dependency

Clear the check box.

From the available resources, select a floating IP address resource and the target volume manager resource and click Add.

[failover1]

fip1,fip2, fip_ib1, the target volume manager resource (for example, volmgr_d_01 for disk_d_01)

[failover2]

fip3,fip4, fip_ib2, the target volume manager resource (for example, volmgr_d_13 for disk_d_13)

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Common tab of the Details screen:

Disk type

lvm

File system

[IO server v4+ for standard model or later]

ext4 or xfs

* Specify the value designed in 5.1.1 .

[IO server v1, v3 and v4 for standard model]

ext4

Device name

LV Path

*Device names can be configured by using the VGs and LVs. (/dev/VGs/LVs)

[failover1]

Data regions

/dev/vg_data01/lv_data01,・・・,/dev/vg_data[n]/lv_data[n]

Metadata regions

/dev/vg_ctrl01/lv_ctrl01,・・・,/dev/vg_ctrl[n]/lv_ctrl[n]

[failover2]

Data regions

/dev/vg_data[n+1]/lv_data[n+]1,・・・,/dev/vg_data[n+n]/lv_data[n+n]

Metadata regions

/dev/vg_ctrl[n+1]/lv_ctrl[n+1],・・・,/dev/vg_ctrl[n+n]/lv_ctrl[n+n]

Mount point

Device mount point specified when creating the IO target

Data regions

/mnt/iot/X/data

Metadata regions

/mnt/iot/X/ctrl

*The X specifies the IO target ID.

[IO server v4+ for standard model or later]

If ext4 is specified, click the Tuning button on the Common tab of the Details screen.

Configure the following items on the Fsck tab of the Disk Resource Tuning Properties:

fsck action before mount

Not Execute

fsck Action When Mount Failed

When the check box is selected (default value)

A.15 Add group resources (EXEC resource)

Add the resources (ScaTeFS server, routing) to be run on the IO servers.

- Routing

  Click Add next to the Group resources list on the Group resources screen.

  Configure the following items of the Group Resource Definitions screen:

Type

EXEC resource

Name

[failover1]

exec_route1

[failover2]

exec_route2

Configure the following items on the Dependency screen:

Apply default dependency

Clear the check box.

From the available resources, select a floating IP address resource and click Add.

[failover1]

fip1,fip2,fip_ib1

[failover2]

fip3,fip4,fip_ib2

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Details screen:

Select the user application.

Click Edit next to the Scripts list.

Configure the following item by inputting the application path:

Start

/opt/scatefs/script/exec_route.sh

ScaTeFS server

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

EXEC resource

Name

[failover1]

exec1

[failover2]

exec2

Configure the following items on the Dependency screen:

Apply default dependency

Clear the check box.

From the available resources, select each resource and click Add.

* Add all resources to be displayed.

[failover1]

fip1,fip2,fip_ib1,Volume manager resource,Disk resource, EXEC resource (Routing)

[failover2]

fip3,fip4,fip_ib2,Volume manager resource,Disk resource, EXEC resource (Routing)

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Details screen:

Select the user application.

Click Edit next to the Scripts list.

Configure the following item by inputting the application path:

Start

/opt/scatefs/script/start.sh

Stop

/opt/scatefs/script/stop.sh

A.16 Create monitor resources

From here, instead of creating monitor resources for each failover group, you will be creating monitor resources for each IO server.

A.17 Add monitor resources (Disk monitor)

Add disk monitor resources.

For each IO server, only add the first disk resource of the metadata region.

iosv00

disk_c_01

iosv01

disk_c_[n+1]

*[n] indicates the number of VGs per IO server designed for an LVM configuration.

Click Add next to the Monitor resources list on the Monitor resources screen.

Configure the following items on the Monitor Resource Definitions screen.

Type

Disk monitor

Name

iosv00

diskw_c_01

iosv01

diskw_c_[n+1]

Configure the following items on the Monitoring (Common) screen:

Monitoring timing

Select active.

Target resource

iosv00

Click Browse and select disk_c_01.

iosv01

Click Browse and select disk_c_[n+1].

Configure the following items on the Common tab of the Monitoring (special) screen:

Monitoring method

READ(O_DIRECT)

Monitoring target

iosv00

/dev/vg_ctrl01/lv_ctrl01

iosv01

/dev/vg_ctrl[n+1]/lv_ctrl[n+1]

Configure the following items on the Recovery Action screen:

Recovery target

iosv00

Click Browse and select disk_c_01.

iosv01

Click Browse and select disk_c_[n+1].

A.18 Add monitor resources (Custom monitor)

Add EXEC resources(ScaTeFS server) monitor resources.

Click Add next to the Monitor resources list on the Monitor resources screen.

Configure the following items on the Monitor Resource Definitions screen.

Type

Custom monitor

Name

iosv00

genw1

iosv01

genw2

Configure the following items on the Monitoring (Common) screen:

Interval

15

Monitoring timing

Select active.

Target resource

iosv00

Click Browse and select exec1.

iosv01

Click Browse and select exec2.

Configure the following items on the Monitoring (special) screen:

Select User Application.

File

iosv00

/opt/scatefs/script/is_exec1_ios_running.sh

iosv01

/opt/scatefs/script/is_exec2_ios_running.sh

Monitor Type

Select Synchronous.

Configure the following items on the Recovery Action screen:

Recovery Action

Restart the recovery target, and if there is no effect with Restart, then failover.

Recovery target

iosv00

Click Browse and select exec1.

iosv01

Click Browse and select exec2.

Adding the monitor resource (custom monitor) is almost finished with the above, however, if the settings of all the process name monitor resource has been set, delete them with the following steps:

Right-click psw1 and psw2 respectively in the monitor resource list of the monitor resource screen, then click "Remove Monitor Resource". A confirmation dialog should be displayed, so click Yes.

A.19 Change monitor resources (Volume manager monitor)

Change the settings of all the volume manager monitor resources that were created automatically.

Select volmgrwX in the Monitor resources list on the Monitor Resources screen and click Properties.

Configure the following items on the Monitoring (Common) screen:

Timeout

240

Rewrite count

3

A.20 Change monitor resources (User mode monitor)

Change the settings of all the user mode monitor resources that were created automatically.

Select userw in the Monitor resources list on the Monitor Resources screen and click Properties.

Configure the following items on the Monitoring (special) screen:

Method

keepalive

Operation at Timeout Detection

PANIC

Extended Monitor Settings

Select the check boxes of the following items:

Open/Close Temporary File

Write

Create Temporary Thread

A.21 Change monitor resources (Floating IP monitor)

Change the settings of all the floating IP monitor resources that were created automatically.

This configuration is needed only use bonding interface.

Select fipwX in the Monitor resources list on the Monitor resources screen and click Properties.

Configure the following items on the Monitor(special) screen:

Monitor NIC Link Up/Down

Select the check box.

A.22 Add monitor resources (IP monitor) (10GbE)

Add the settings of IP monitor resources.

Add the IP monitor resources for each floating IP resources.

Click Add next to the Monitor resources list on the Monitor resources screen.

Configure the following items on the Monitor Resource Definitions screen.

Type

IP monitor

Name

iosv00

ipw1

ipw2

iosv01

ipw3

ipw4

Configure the following items on the Monitoring (Common) screen:

Interval

30 seconds.

Timeout

30 seconds.

Retry Count

3 times.

Monitoring timing

Select active

Target resource

iosv00

ipw1

Click Browse and select fip1.

ipw2

Click Browse and select fip2.

iosv01

ipw3

Click Browse and select fip3.

ipw4

Click Browse and select fip4.

Choose servers that execute monitoring

click Server

Select

select the checkbox.

Servers that can run the Group

ipw1, ipw2

iosv00.

ipw3, ipw4

iosv01.

Configure the following items on the Common tab of the Monitoring (special) screen:

Monitoring target

ipw1

The gateway IP address of the network which has fip1(*).

ipw2

The gateway IP address of the network which has fip2(*).

ipw3

The gateway IP address of the network which has fip3(*).

ipw4

The gateway IP address of the network which has fip4(*).

(*)Note that it is NOT the floating IP address of the IO server.

Configure the following items on the Recovery Action screen:

Recovery Action

Restart the recovery target, and if there is no effect with Restart, then failover.

Recovery Target

ipw1

Click Browse and select fip1.

ipw2

Click Browse and select fip2.

ipw3

Click Browse and select fip3.

ipw4

Click Browse and select fip4.

A.23 Configure the recovery action settings for when a monitor resource error occurs

When you click Finish after creating a monitor resource, the following popup message appears, click Yes.

[EXPRESSCLUSTER X Cluster WebUI Offlime]

Do you want to enable the following operations?

- Group Automatic Startup
- Recovery operation when group resource activation/deactivation failure detected
- Recovery Action on Monitor Failure

[EXPRESSCLUSTER X builder (Offline version)]

Set recovery action caused by monitor resource error.

A.24 Change the cluster properties

Click Properties of Cluster.

Select the Timeout tab and configure the following item:

Server Internal Timeout

300

[IO server v4 for standard model or later]

Select the Monitor tab and configure the following item:

Method

keepalive

At last, you complete creating the cluster configuration information. Save this information to the file system. You will be required to import the cluster configuration information file and use it when configuring the IO servers later.

[EXPRESSCLUSTER X Cluster WebUI Offlime]

Click Export and save the file under any directory.

See in the manual (2).

[EXPRESSCLUSTER X builder (Offline version)]

Open the File menu and click Export and save the file under any directory.

See in the manual (3).

Table 12-2 Correspondence table between the various resources

IO server name		**iosv00**		**iosv01**	
Interconnect IP address					
Device name of the partition for heartbeat region					
Failover group name		failover1		failover2	
Floating IP address	**Groupresource name**	fip1	fip2	fip3	fip4
	IP address				
Data type		**Data**	**Metadata**	**Data**	**Metadata**
Volume manager resource	**Group resource name**	volmgr_d_01 volmgr_d_02 ・・・ volmgr_d_[n]	volmgr_c_01 volmgr_c_02 ・・・ volmgr_c_[n]	volmgr_d_[n+1] volmgr_d_[n+2] ・・・ volmgr_d_[n+n]	volmgr_c_[n+1[volmgr_c_[n+2] ・・・ volmgr_c_[n+n]
	VG name	vg_data01 vg_data02 ・・・ vg_data[n]	vg_ctrl01 vg_ctrl02 ・・・ vg_ctrl[n]	vg_data[n+1] vg_data[n+2] ・・・ vg_data[n+n]	vg_ctrl[n+1] vg_ctrl[n+2] ・・・ vg_ctrl[n+n]
Disk resource	**Group resource name**	disk_d_01 disk_d_02 ・・・ disk_d_[n]	disk_c_01 disk_c_02 ・・・ disk_c_[n]	disk_d_[n+1] disk_d_[n+2] ・・・ disk_d_[n+n]	disk_c_[n+1] disk_c_[n+2] ・・・ disk_c_[n+n]
	VG name	lv_data01 lv_data02 ・・・ lv_data[n]	lv_ctrl01 lv_ctrl02 ・・・ lv_ctrl[n]	lv_data[n+1] lv_data[n+2] ・・・ lv_data[n+n]	lv_ctrl[n+1] lv_ctrl[n+2] ・・・ lv_ctrl[n+n]
Device name		/dev/vg_data01/lv_data01 /dev/vg_data02/lv_data02 ・・・ /dev/vg_data[n]/lv_data[n]		/dev/vg_ctrl[n+1]/lv_ctrl[n+1] /dev/vg_ctrl[n+2]/lv_ctrl[n+2] ・・・ /dev/vg_ctrl[n+n]/lv_ctrl[n+n]	
Mount point		/mnt/iot/0/data /mnt/iot/1/data ・・・ /mnt/iot/[n-1]/data		/mnt/iot/[n]/ctrl /mnt/iot/[n+1]/ ctrl ・・・ /mnt/iot/[n+n-1]/ ctrl	

IO server name	**EXEC resource**		**Disk monitor resource name**	**Process name monitor resources name**
	Routing	ScaTeFS server		
iosv00	exec_route1	exec1	diskw_c_01	psw1
iosv01	exec_route2	exec2	diskw_c_[n+1]	psw2

Concerning the IO target ID (mnt/iot/X/) in the mount point path

The IO target IDs assigned by the system when creating IO targets are assigned in order from 0, starting with the first IO server (iosv00).

The following shows the IO target IDs assigned when n IO targets are configured on the IO servers:

iosv00

IO target ID :　0 ～ n-1

iosv01

IO targetID : n ～ n+n-1

After creating the IO targets, use the scatefs_detail -t command to check that the IO target IDs have been assigned as intended.

For how to use this command, see “5.2.4 Creating IO targets (scatefs_addiot)”.

[Naming rules]

It is recommended to use the following rule for naming the group resources and VGs.

Data region

Group resource name

volmgr_d_01,volmgr_d_02,・・・

VGs

vg_data01,vg_data02,・・・

Metadata region

Group resource name

volmgr_c_01,volmgr_c_02,・・・

VGs

vg_ctrl01,vg_ctrl02,・・・

It is recommended to use the following rule for naming the group resources and LVs.

- Data region
  - Group resource name
    - disk_d_01,disk_d_02,・・・
  - LVs
    - lv_data01,lv_data02,・・・
- Metadata region
  - Group resource name
    - disk_c_01,disk_c_02,・・・
  - LVs
    - lv_ctrl01,lv_ctrl02,・・・

Appendix B Procedure for Accessing ScaTeFS from Windows

This procedure describes how to share ScaTeFS with Windows machine.

Samba server is used as a relay to access ScaTeFS from Windows machine. Following procedures describe how to configure the Samba server and Windows machine. Please refer "Configuration of the redundant cluster" for redundancy configuration of the Samba server.

B.1 Network Configuration

Example of network configuration

Please refer "2.3.1 Example of configuration" in "Chapter2 Network configuration" for example of network configuration.

- Image of Environment

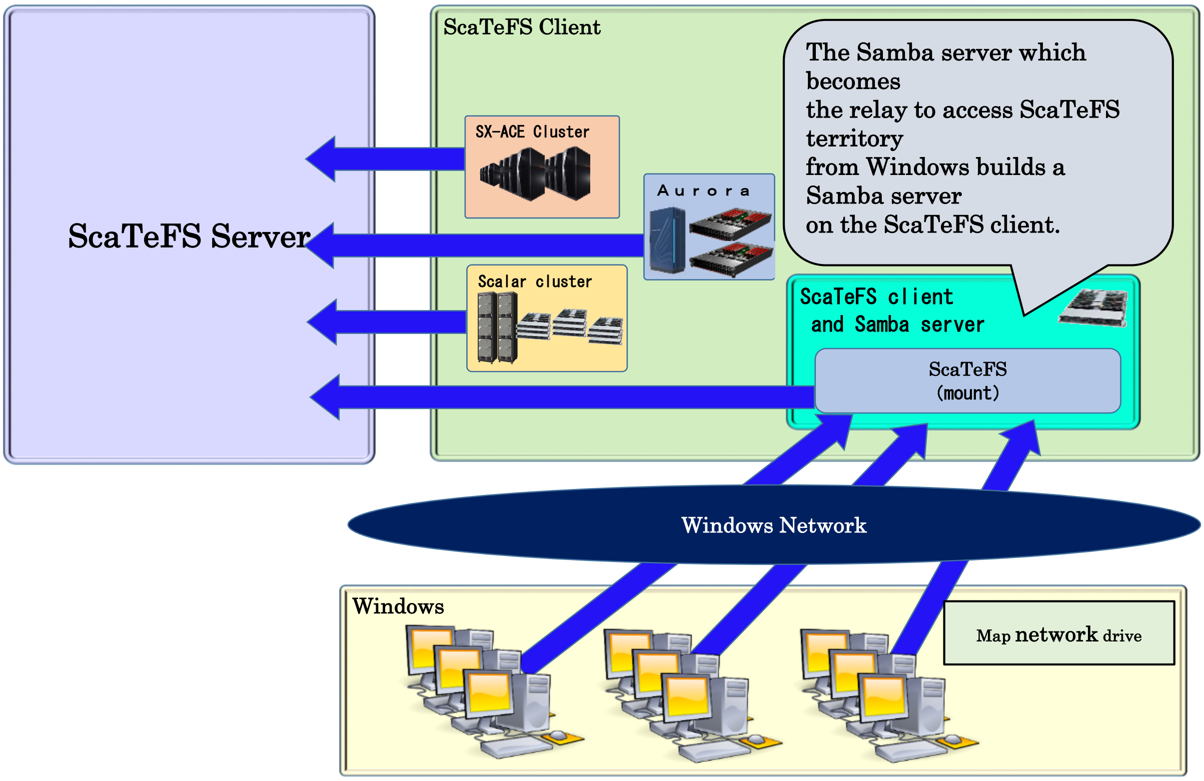

ScaTeFS Server
ScaTeFS Client
SX-ACE Cluster
Aurora
Scalar cluster
The Samba server which becomes the relay to access ScaTeFS territory from Windows builds a Samba server on the ScaTeFS client.
ScaTeFS client and Samba server
ScaTeFS (mount)
Windows Network
Windows
Map network drive

Figure 12-1 Composition image

B.2 Getting Started

This procedure configures the Samba server on ScaTeFS client machine, exports ScaTeFS by file sharing function of the Samba server, and configures Windows machine to connect the Samba server.

B.2.1 Preparation

Please prepare ScaTeFS client machine which can be accessed from Windows machine.
Following program products will be required for constructing.

- Samba
- EXPRESSCLUSTER X for Linux (required for redundancy configuration)

B.2.2 Configuration overview

The overview of configurations is as follows.

(1) Installation and configuration of Samba
Please apply procedures from “Installation of Samba4” to “Configuration of Firewall” to the machine which is prepared in “Preparation”.
(2) Configuration of Windows machine
Please prepare Windows machine to access ScaTeFS.

Please refer “Configuration of the redundant cluster” to configure EXPRESSCLUSTER for redundancy configuration of the Samba server.

B.3 Configuration of the Samba server

Following procedures require root privilege.
Please prepare RPM package of Samba4 from RHEL7 repository. Installation is issued by rpm command.
For detailed configuration for user environment, please refer “CHAPTER 15. FILE AND PRINT SERVERS” in “Deployment, Configuration, and Administration of Red Hat Enterprise Linux 7”.

B.3.1 Installation of Samba4

```
# rpm -ivh samba-4.v.v-x.ely.zzzzz.rpm
----------------------------------------
4.v.v : version number of Samba
x     : release
```

```
y       : major version of OS
zzzzz : supported architecture(x86_64)
----------------------------------------
```

B.3.2　Configuration of Samba

Edit the configuration file of Samba (/etc/samba/smb.conf) to share ScaTeFS as a shared directory with Windows machine.

At first, copy the configuration file of Samba for backup.

```
# cp -p /etc/samba/smb.conf /etc/samba/smb.conf_org
```

Edit /etc/samba/smb.conf to share ScaTeFS with Windows machine. Preferred configuration is described in the following table. Default values will be used for configuration items which are not described.

Global Settings

No.	Configuration Item	Default value	Preferred value	Note
1	csc policy	manual	disable	Offline caching configuration for files and directories. When manual, documents, or programs is specified,manually or automatically synchronization becomes available. In this case, files on shared server may be lost when synchronization is failed. Therefore, disable should be specified.
2	netbios name	DNS name of the machine	Server name	Specify network computer name. Specified server name will be displayed in the list of network computer name.

Share Definitions

No.	Configuration Item	Default value	Preferred value	Note
[ScaTeFS_share]				Specify any share name. (ScaTeFS_share is one of examples.)
3	Comment	none	Any comment	Specify the comment to be displayed when selecting Detailed display in Windows.
4	Path	none	The mount point of ScaTeFS. For example, /mnt/scatefs	Specify the path to be shared.
5	Writable	none	yes	Enable write access to shared file.

B.3.3 Mounting ScaTeFS

Please refer "6.2.5 Mounting" in "NEC Scalable Technology File System (ScaTeFS) Administrator's Guide".

B.3.4 Creating share directory

Create the directory which is shared with Windows machine under the mount directory of ScaTeFS.

Execute following commands in order.

```
# mkdir -p /mnt/scatefs/share
# chmod -R 0777 /mnt/scatefs/share
# chown -R root:root /mnt/scatefs/share
```

B.3.5 Starting Samba

Start Samba service.

```
# systemctl start smb
# systemctl start nmb
```

B.3.6 Creating Samba user

Create Linux user which is used by Samba, and register as a Samba user.

Creating Linux user

```
# useradd samba_user
# passwd samba_user
```

Registering Samba user

```
# pdbedit -a samba_user
```

B.3.7　Configuration of SELinux

If the result of confirmation is “Enforcing”, following procedure is required. If the result is “Permissive” or “Disabled”, configuration of SELinux is not required.

Confirming SELinux is enabled or not

```
# /usr/sbin/getenforce
Enforcing
```

Set “samba_share_t” for permitting access to shared directory by Samba.

Modifying configuration of SELinux

```
# chcon -t samba_share_t [directory to share]
```

B.3.8　Configuration of Firewall

If the result of confirmation is “running”, following procedure is required. If the result is “not running”, configuration of firewall is not required.

Confirming firewall is enabled or not

```
# firewall-cmd --state
```

Configuring firewall

```
# firewall-cmd --permanent --zone=public --add-service=samba
# firewall-cmd --reload
```

B.4　Configuration on Windows machine

Followings are typical methods to connect shared ScaTeFS.

B.4.1　Accessing shared ScaTeFS

Use explorer on the Windows machine to open shared ScaTeFS.

Start explorer on the Windows machine, and enter the name of Samba server and the share name in address bar.

```
Example: ¥¥samba_server¥ScaTeFS_share
samba_server: the name of server which is specified by netbios name, or IP address
        ScaTeFS_share: share name
```

Connect with the user name and password created in "B.3.6 Creating Samba users".

Shared ScaTeFS will be displayed in the library window of explorer.

B.4.2　Assigning Network Drive

Using ScaTeFS shared area from Windows machine.

(1)　Right click on “Computer” in navigation window.

(2)　Select “Map network drive” in the context menu.

(3)　On the “Map Network Drive screen”, select any “Drive”, enter the IP address and directory name of the shared server, and click "Finish" button.

(4)　Connect with the user name and password created in "B.3.6 Creating Samba users".

(5)　The drive which is specified in above step will be displayed in the navigation window.

B.5　Configuration of the redundant cluster

This chapter describes an example to configure Samba servers as a redundant cluster.
In this chapter, WebManager is used to configure EXPRESSCLUSTER. Therefore, a Windows machine which can access to the target Linux client is required for operation.

“EXPRESSCLUSTER X File Server Agent”, which is an optional product of EXPRESSCLUSTER X for Linux, enables specific configuration for monitoring Samba server. Please refer the formal web site of EXPRESSCLUSTER for this product.

B.5.1　Creating the cluster

Create the cluster.

B.5.1.1　Adding the cluster

Open the File menu and click Cluster generation wizard. A confirmation dialog box is displayed. Click Start the Standard Cluster generation wizard. Leave the default settings as they are on the Cluster generation wizard screen and click Next.

B.5.1.2 Adding servers

Add two Linux clients configuring the cluster to the server definitions on the Cluster generation wizard screen. In the following description, the names of the Linux clients are lxcl00 and lxcl01.

Click Add next to the Server Definitions list.

Set the following item on the Add new server screen. This server will be the master server.

Server name

lxcl00

Click Add next to the Server Definitions list again.

Set the following item on the Add new server screen.

Server name

lxcl01

B.5.1.3 Configuring network

Configure network between Linux clients in the cluster.

Click Add next to the Interconnects list.

Specify an item on the Priority 1 line.

Type

Kernel mode

lxcl00

IP address for interconnect between Linux clients

lxcl01

IP address for interconnect between Linux clients

(*)In this procedure, the IP address of the network setting port of the client is set as the IP address for interconnection.

B.5.1.4 Configuring the network partition resolution processing (NP resolution)

Move to the next step with do nothing.

Continue to next without any setting.

B.5.2 Creating the failover group

Create the failover group running on Linux clients in the cluster.

B.5.2.1 Adding the failover group

Click Add next to the Groups list on the Groups screen.

Configure the following items of the Group Definitions screen:

Name

[failover1]

failover1

On the List of Bootable Servers screen, clear the Failover OK check box of each server.

From the available servers, select the IO servers in the following order and click Add.

* Servers must be added in the right order.

[failover1]

lxcl00

lxcl01

On the Configure Group Attributes screen, change the following items from their default values:

Failback attribute

Auto failback

B.5.2.2 Adding the group resource (floating IP address)

Configure the IP address of the file system port used to configure the Linux client network (10GbE, IB).

The number of resources to be added differs depending on the number of ports in the network setting of the Linux client, this subsection describes an example for configuring a 10GbE as FIP (fip1) and adding fip1 to [failover1].

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

floating ip resource

Name

[failover1]

fip1

Leave the default settings are they are on the Dependency screen and move to the next step.

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Common tab of the Details screen:

IP Address

IP address of the file system port used to configure the Linux client network (10GbE, IB)

Example:

192.168.0.31/24%enp4s9

Click Tuning button.

Configure the following items on the Parameter tab of the Floating IP Resource Tuning Properties:

Judge NIC Link Down as abnormal

Turn on the checkbox

B.5.2.3 Adding the group resource (exec resource)

Add the resource (Samba) which runs on the Linux client.

Samba server

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

execute resource

Name

[failover1]

exec_samba

Configure the following items on the Dependency screen:

Apply default dependency

Clear the check box.

From the available resources, select each resource and click Add.

* Add all resources to be displayed.

[failover1]

fip1, EXEC resource (exec_samba)

Leave the default settings are they are on the Recovery Operation screen and move to the next step.

Configure the following items on the Details screen:

Select the user application.

Click Edit next to the Scripts list.

Configure the following item by inputting the application path:

Start

/root/samba_ctl/start.sh

Stop

/root/samba_ctl/stop.sh

(*)Please make the above " start.sh " and " stop.sh " according to the practical use environment.
Please designate the preservation destination which was also added to the practical use environment about a preservation place.

B.5.3 Creating monitor resources

In this section, monitor resources will be created for each Linux client, not failover group.

B.5.3.1 Configuration of the monitor resource (floating IP monitor)

Configure the floating IP monitor which was created automatically.

Select fipw1 in the Monitor resources list on the Monitor resources screen and click Properties.

Configure the following items on the Monitor (special) screen:

Monitor NIC Link Up/Down

Turn on the checkbox

B.5.3.2 Adding the monitor resource (custom monitor)

Add the monitor resource to EXEC resource (Samba server).

Click Add next to the Group resources list on the Group resources screen.

Configure the following items of the Group Resource Definitions screen:

Type

custom monitor

Name

lxcl00

genw_samba

Configure the following items on the Monitoring (Common) screen:

Interval

30 seconds.

Monitoring timing

Select active

Target resource

lxcl00

Click Browse and select exec_samba.

Configure the following items on the Monitoring (Common) screen:

Select User Application

File

lxcl00

/root/samba_ctl/is_samba_running.sh

Monitor Type

Select Synchronous.

Configure the following items on the Recovery Action screen:

Recovery Action

Restart the recovery target, and if there is no effect with Restart, then failover.

Recovery target

lxcl00

Click Browse and exec_samba.

(*) “is_samba_running.sh” is a shell script which starts the monitored daemon. Please customize contents and place of the shell script for user environment. Following is an example.

・Example of is_samba_running.sh

```
#!/bin/bash

systemctl status smb | grep Active | grep -q "active (running)"
SMB_STATUS=$(echo $?)
systemctl status nmb | grep Active | grep -q "active (running)"
NMB_STATUS=$(echo $?)

if [ ${SMB_STATUS} -eq 0 -a ${NMB_STATUS} -eq 0 ]; then
        exit 0
else
        exit 1
fi
```

B.5.4 Configuration of cluster property

Right click on cluster in tree view, and select property.

Select Cluster in the tree view and right-click Properties.

Select the Timeout tab and configure the following item:

Server Internal Timeout

300

Appendix C History

C.1 History table

Feb. 2018	1st Edition
Aug. 2018	2nd Edition
Dec. 2018	3rd Edition
May. 2019	4th Edition
Oct. 2019	5th Edition
Nov. 2019	6th Edition
Jan. 2020	7th Edition
May. 2020	8th Edition
Jul. 2020	9th Edition
Oct. 2020	10th Edition
Dec. 2020	11th Edition
May. 2021	12th Edition
Oct. 2021	13th Edition
Dec. 2021	14th Edition
Mar. 2022	15th Edition
Jun. 2022	16th Edition
Jan. 2023	17th Edition
Mar. 2023	18th Edition
Sep. 2023	19th Edition
Oct. 2024	20th Edition

C.2 Change notes

- 1st Edition

  First edition

- 2nd Edition

  Modify 9.1 Resource constraints (QUOTA) for Directory Quota

  Add 9.7 Rebalance

- 3rd Edition

  Modify 3.1 HA cluster configuration for IOSv4

  Modify 5.1.2 LVM design for IOSv4

Modify 6.5 Syslog messages

Modify 11.6.2 How to use ScaTeFS VE direct IB library for supporting glibc

Add 11.6.6 for the performance tuning of NEC Fortran program

- 4th Edition

  Modify 5.1.2 LVM design

  Modify 5.1.7 Installing and setting up the PathManager for Linux driver package

  Modify 5.1.8 Installing the EXPRESSCLUSTER X for Linux

  Modify 5.1.13 Registering the ScaTeFS license

  Modify 6.1.1 Installing the InfiniBand driver

  Modify 6.1.4 Registering the ScaTeFS license

  Add 10.1 Start and stop the IO Server

  Modify Table 12-1　Specification

  Modify A.24 Change the cluster properties

- 5th Edition

  Modify 5.1.12 Installing the ScaTeFS package

  Modify 6.1.1 description about Mellanox OFED for RHEL/CentOS 7.6

  Modify 9.1.1 description about scatefs_quotacheck command

  Modify 11.6.2 description about how to use ScaTeFS VE direct IB library

  Modify Table 12-1 Specification

- 6th Edition

  Support the IO server v4+

  - RHEL7.6

  - EXPRESSCLUSTER X 4.1

  - IB HCA HDR100 (ConnectX-6)

  - xfs as the data regions of the IO targets

  Modify 3.1 HA cluster configuration for IOSv4+

  Modify 4.1 Specifications for Linux machines（SX-Aurora TSUBASA）for ConnectX-6

  Modify 5.1.1 IO targets design for IOSv4+

  Modify 5.1.2 LVM design for IOSv4+

  Modify 5.1.4 , 5.1.5 , 5.1.6 The IO server settings was simplified

  Modify 5.1.8 Installing the EXPRESSCLUSTER X for Linux for EXPRESSCLUSTER X 4.1

  Modify 5.1.10 Installing the IB driver for RHEL7.6

  Modify 5.3.1 Creating ScaTeFS (scatefs_mkfs)

Modify 5.4.2 Starting WebManager

Modify 10.2.2 Non-stop update of the ScaTeFS package

Add 10.7 Firmware update after ConnectX-6 HCA card replacement

Modify A.14 Add group resources (disk resource)

- 7th Edition

  Modify 6.1.1 description about RHEL/CentOS 7.7 and Mellanox OFED4.7

  Add Chapter 8 Setting to use ScaTeFS on a Docker's container

- 8th Edition

  Support the IO server v4++

  - RHEL7.7
  - EXPRESSCLUSTER X 4.2

  Modify 3.1 HA cluster configuration for IOSv4+

  Modify 5.1.10 Installing the IB driver for RHEL7.7

- 9th Edition

  Add "3.1.6 SFA7990XE"

  Modify "5.1.12 Installing the ScaTeFS package"

  Add Mellanox OFED driver version for RHEL/CentOS 8.1 in "6.1.1 Installing the InfiniBand driver"

  Add "5.5 Configuring IO servers for DDN SFA7990XE"

  Add "6.4.4 Notice of double mount"

  Add "9.11 Monitoring the ScaTeFS filesystems"

- 10th Edition

  Modify the reference to the SX-Aurora TSUBASA Installation Guide in "5.1.12 Installing the ScaTeFS package"

  Add Mellanox OFED driver version for RHEL/CentOS 7.8 in "6.1.1 Installing the InfiniBand driver"

- 11th Edition

  Modify "CLUSTERPRO X for Linux" to "EXPRESSCLUSTER X for Linux" in this manual

  Modify "SPS" to "PathManager" in this manual

  Add Mellanox OFED driver version for RHEL/CentOS 8.2 in "6.1.1 Installing the InfiniBand driver"

Modify "Table 9-2 Remote CLI Subcommand" for the subcommand of "mkqdir" and "rmqdir"

Modify "9.11 Monitoring the ScaTeFS filesystems"

Add "10.8.1 Linux client" and Post the contents of Chapter 6.5

Add "10.8.2 IO server"

Add "11.6.9 Memory usage when using ScaTeFS InfiniBand high performance library"

- 12th Edition

  Add sos package installation in "5.1.12 Installing the ScaTeFS package"

  Add Mellanox OFED driver version for RHEL/CentOS 7.9 in "6.1.1 Installing the InfiniBand driver"

  Add description of RHEL 8 and CentOS in "6.1.6 Mounting"

  Add description of RHEL 8 and CentOS in "6.2.5 Mounting"

  Modify ScaTeFS related messages in "10.8.2 IO server"

  Add description of SCATEFS_CQPOLLHOW environment variable in "11.6.4 Setting of environment variables for performance tuning"

- 13th Edition

  Add Mellanox OFED driver version for RHEL/CentOS 8.3 in "6.1.1 Installing the InfiniBand driver"

  Modify "10.4 Integrity check and recovery of the file system"

- 14th Edition

  Add Mellanox OFED driver version for RHEL/CentOS 8.4 in "6.1.1 Installing the InfiniBand driver"

  Modify "6.4.1 Removing a file which is opened by a process"

- 15th Edition

  Modify "1.2.1 Client"

  Modify "1.2.2 Network"

  Add Mellanox OFED driver version for RHEL/CentOS 8.4 and RHEL 8.5 in "6.1.1 Installing the InfiniBand driver"

  Add "6.4.5 Notice when using mlocate package"

- 16th Edition

Change "Mellanox OFED" to "MLNX_OFED"

Change the URL of MLNX_OFED driver download site in “5.1.10 Installing the IB driver”

Change the URL of MLNX_OFED driver download site in “6.1.1 Installing the InfiniBand driver”

Add MLNX_OFED driver version for RHEL/Rocky Linux 8.5 in "6.1.1 Installing the InfiniBand driver"

Change the URL of firmware download site in “10.7 Firmware update after ConnectX-6 HCA card replacement”

Modify firmware version for ConnectX-6 in “10.7 Firmware update after ConnectX-6 HCA card replacement”

- 17th Edition

  Change the URL of related documents

  Modify EXPRESSCLUSTER version in "Table 5-1 IO server v4++ Supported distribution, kernel and software versions"

  Add MLNX_OFED driver version for RHEL/Rocky Linux 8.6 in "6.1.1 Installing the InfiniBand driver"

  Add "9.13 Subdirectory mount"

  Add "9.13.1 Mounting"

  Add "9.13.2 Unmounting"

  Add description of Directory QUOTA in "9.1 Resource constraints (QUOTA)"

  Modify ScaTeFS related messages in "10.8.2 IO server"

- 18th Edition

  Support the IO server v4++

  - EXPRESSCLUSTER X 4.3.4-1

  Modify “5.1.3 Creating the EXPRESSCLUSTER cluster configuration information”

  Modify the reference to "SX-Aurora TSUBASA Installation Guide" in "5.1.12.1 When using the HPC Software License"

  Modify "5.4 Setting the EXPRESSCLUSTER" for Cluster WebUI

  Change the title of 5.4.1.1 to “Transferring the cluster configuration information file to the work PC”

Change the title of 5.4.2 to “Starting Cluster WebUI and WebManager”

Change the title of 5.4.3 to “Importing the cluster configuration information file”

Modify “5.4.5 Apply Settings”

Add MLNX_OFED driver version for RHEL/Rocky Linux 8.6 in "6.1.1 Installing the InfiniBand driver"

Change the title of Appendix A to “Procedure for Creating EXPRESSCLUSTER Cluster Configuration Information (Offline vesion)” and add description about Cluster WebUI Offline

Change the title of A.2 to “Installing the EXPRESSCLUSTER tool”

Change the title of A.3 to “Start the EXPRESSCLUSTER tool”

Change the title of A.12 to “Add group resources (Floating IP resource)”

Change the title of A.13 to “Add group resources (Volume manager resource)”

Change the title of A.14 to “Add group resources (Disk resource)”

Change the title of A.15 to “Add group resources (EXEC resource)”

Change the title of A.17 to “Add monitor resources (Disk monitor)”

Change the title of A.18 to “Add monitor resources (Custom monitor)”

Change the title of A.19 to “Change monitor resources (Volume manager monitor)”

Change the title of A.20 to “Change monitor resources (User mode monitor)”

Change the title of A.21 to “Change monitor resources (Floating IP monitor)”

Modify “B.5.1.1 Adding the cluster”

Modify “B.5.1.2 Adding servers”

- 19th Edition

  Add MLNX_OFED driver version for RHEL/Rocky Linux 8.8 in "6.1.1 Installing the InfiniBand driver"

- 20th Edition

  Modify "5.5.16 Setting the kernel parameter"

  Add MLNX_OFED driver version for RHEL/Rocky Linux 8.10 in "6.1.1 Installing the InfiniBand driver"

SX-Aurora TSUBASA System Software

NEC Scalable Technology File System

(ScaTeFS)

Administrator's Guide

20th edition Oct 2024

NEC Corporation

© NEC Corporation 2018-2024