Lustre Administrators and
Developers Workshop 2014

FUJITSU
shaping tomorrow with you

# Fujitsu's Lustre Contributions
## - Policy and Roadmap-

Shinji Sumimoto, Kenichiro Sakai
Fujitsu Limited, a member of OpenSFS

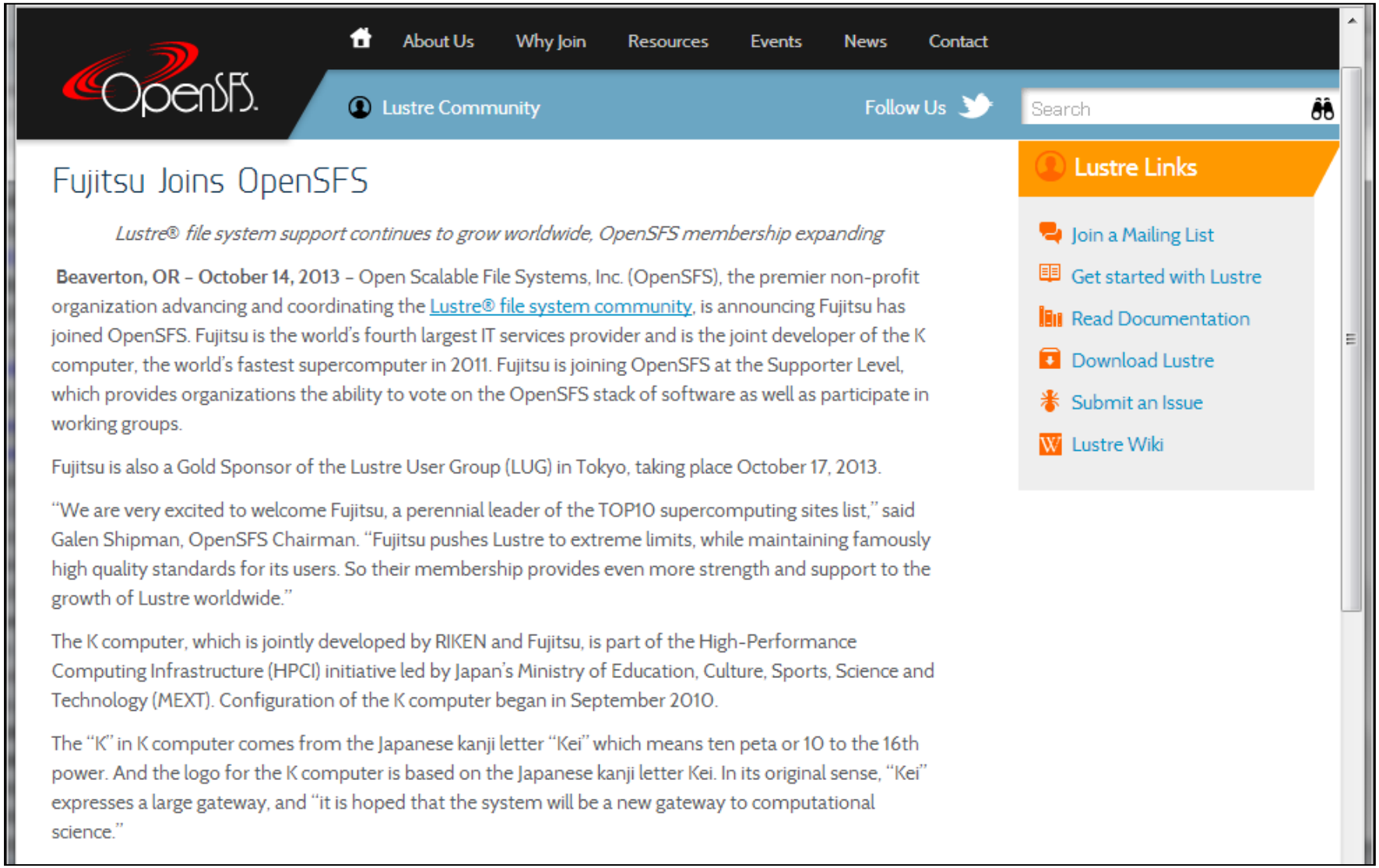
OpenSFS

# Outline of This Talk

FUJITSU

- Current Status of Fujitsu's Supercomputer Development
  - Past and Current Product Development
  - The Next Step towards Exa-scale Development
- Fujitsu's Contribution Policy to Lustre Community
  - Contribution Policy
  - Current Contribution and the Next Step
- Introduction of Contribution Feature
  - IB Multi-rail, Directory Quota etc..

# Fujitsu Joins OpenSFS, Oct. 14, 2013



About Us | Why Join | Resources | Events | News | Contact

Lustre Community | Follow Us | Search

## Fujitsu Joins OpenSFS

*Lustre® file system support continues to grow worldwide, OpenSFS membership expanding*

**Beaverton, OR – October 14, 2013** – Open Scalable File Systems, Inc. (OpenSFS), the premier non-profit organization advancing and coordinating the Lustre® file system community, is announcing Fujitsu has joined OpenSFS. Fujitsu is the world's fourth largest IT services provider and is the joint developer of the K computer, the world's fastest supercomputer in 2011. Fujitsu is joining OpenSFS at the Supporter Level, which provides organizations the ability to vote on the OpenSFS stack of software as well as participate in working groups.

Fujitsu is also a Gold Sponsor of the Lustre User Group (LUG) in Tokyo, taking place October 17, 2013.

"We are very excited to welcome Fujitsu, a perennial leader of the TOP10 supercomputing sites list," said Galen Shipman, OpenSFS Chairman. "Fujitsu pushes Lustre to extreme limits, while maintaining famously high quality standards for its users. So their membership provides even more strength and support to the growth of Lustre worldwide."

The K computer, which is jointly developed by RIKEN and Fujitsu, is part of the High-Performance Computing Infrastructure (HPCI) initiative led by Japan's Ministry of Education, Culture, Sports, Science and Technology (MEXT). Configuration of the K computer began in September 2010.

The "K" in K computer comes from the Japanese kanji letter "Kei" which means ten peta or 10 to the 16th power. And the logo for the K computer is based on the Japanese kanji letter Kei. In its original sense, "Kei" expresses a large gateway, and "it is hoped that the system will be a new gateway to computational science."

Lustre Links

- Join a Mailing List
- Get started with Lustre
- Read Documentation
- Download Lustre
- Submit an Issue
- Lustre Wiki

FUJITSU

# CURRENT STATUS OF FUJITSU'S SUPERCOMPUTER DEVELOPMENT

# History of Fujitsu Supercomputers

FUJITSU

Fujitsu has been developing HPC file system for customers

Massively Parallel

PRIMEHPC FX10

The K computer*2

No.1 in Top500
(June 2011, Nov 2011)

FX1

Most Performance Efficient
in Top500 (Nov. 2008)

SPARC Enterprise

PRIMEQUEST

PRIMERGY CX1000
Cluster node

PRIMERGY BX900
Cluster node

Japan's Largest
Cluster in Top500
(June 2010)

HX600
Cluster node

Cluster

PRIMERGY RX200
Cluster node

Japan's Largest
Cluster in Top500
(June 2004)

World's Fastest
Vector Processor (1999)

VPP5000

Vector

No.1 in Top500
(Nov. 1993)
Gordon Bell Prize
(1994, 95, 96)

NWT*1

VPP300/700

VPP500

PRIMEPOWER
HPC2500

VP-200

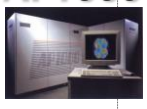
AP3000

F230-75APU

VP-100

Scalar

AP1000



Japan's First
Supercomputer
(1977)

*1 NWT Numerical Wind Tunnel, Joint development with JAXA

*2 Joint Development with Riken

~1980 1985 1990 1995 2000 2005 2010

# K computer and the Next Step

FUJITSU

- K computer: Still TOP500 Rank #4 system in the world.
  - FEFS on K computer is the first 1 TB/s sustained IOR performance file system in the world.
- We are now developing FEFS for the next Post-FX10 system.
- The next target is Exa-scale system

## FX1

4 core
VISIMPACT
8 DDR2-DIMM

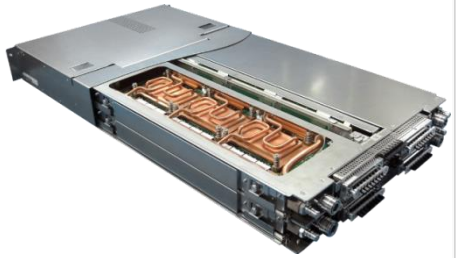
## K computer


8 core
HPC-ACE
8 DDR3-DIMM
**Tofu interconnect**

## FX10

16 core
HPC-ACE
8 DDR3-DIMM
**Tofu interconnect**

## Post-FX10



32 core
HPC-ACE2
8 Hybrid Memory Cube
**Tofu interconnect 2**

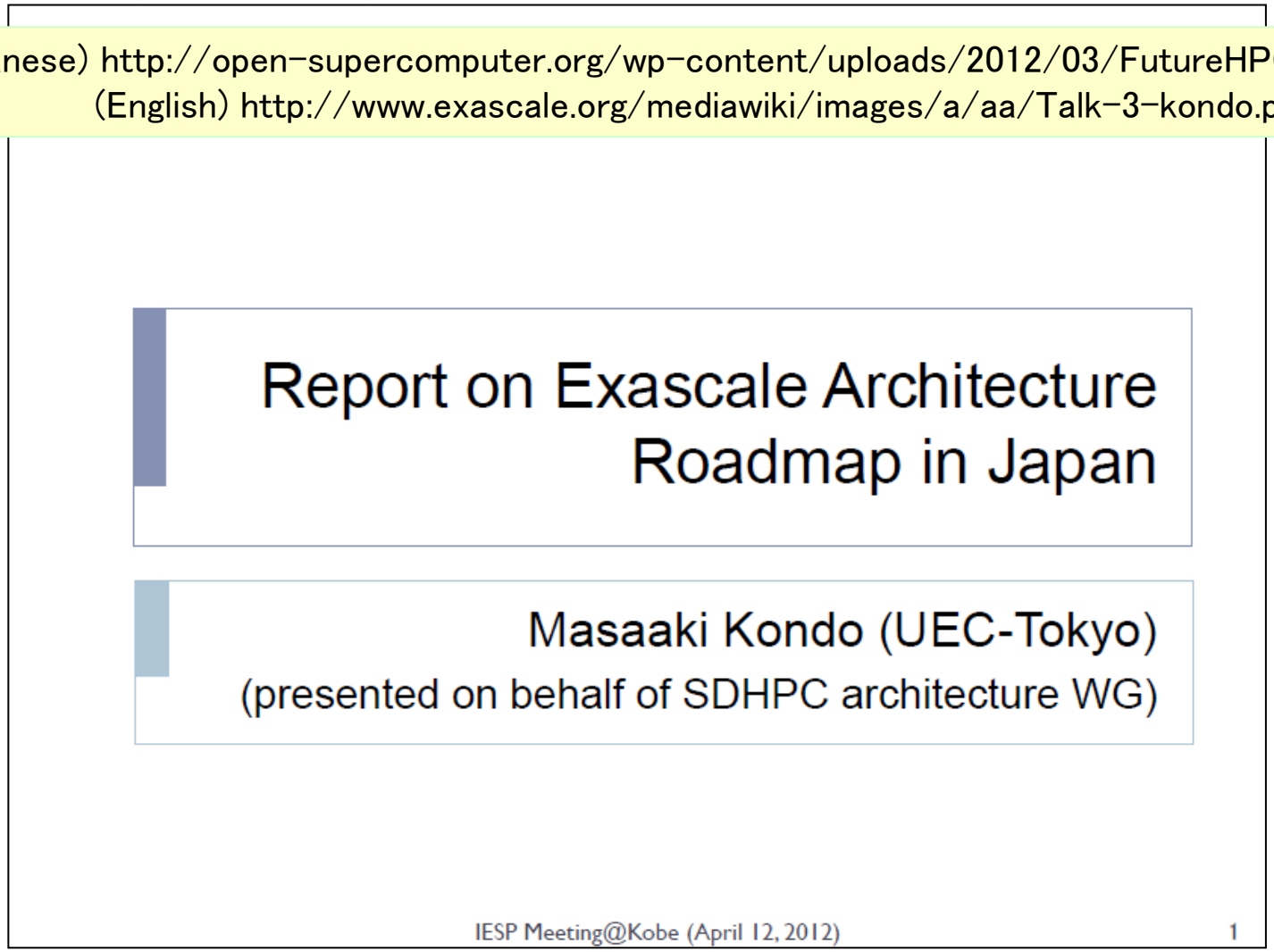
**2008** **2010** **2012** **2015**

# SDHPC Activities for Exascale System

FUJITSU

- Japanese researchers wrote roadmap papers for the exascale system (2010/8 - )

(Japanese) http://open-supercomputer.org/wp-content/uploads/2012/03/FutureHPCI-Report.pdf
(English) http://www.exascale.org/mediawiki/images/a/aa/Talk-3-kondo.pdf



# Storage and System Requirement from the Architecture Roadmap

FUJITSU

## Performance Projection

- Performance projection for an HPC system in 2018
  - Achieved through continuous technology development
  - Constraints: 20 – 30MW electricity & 2000sqm space

*Node Performance*

| | Total CPU Performance (PetaFLOPS) | Total Memory Bandwidth (PetaByte/s) | Total Memory Capacity (PetaByte) | Byte / Flop |
|---|---|---|---|---|
| General Purpose | 200~400 | 20~40 | 20~40 | 0.1 |
| Capacity-BW Oriented | 50~100 | 50~100 | 50~100 | 1.0 |
| Reduced Memory | 500~1000 | 250~500 | 0.1~0.2 | 0.5 |
| Compute Oriented | 1000~2000 | 5~10 | 5~10 | 0.005 |

*Network*

| | Injection | P-to-P | Bisection | Min Latency | Max Latency |
|---|---|---|---|---|---|
| High-radix (Dragonfly) | 32 GB/s | 32 GB/s | 2.0 PB/s | 200 ns | 1000 ns |
| Low-radix (4D Torus) | 128 GB/s | 16 GB/s | 0.13 PB/s | 100 ns | 5000 ns |

*Storage*

| Total Capacity | Total Bandwidth |
|---|---|
| 1 EB | 10TB/s |
| 100 times larger than main memory | For saving all data in memory to disks within 1000-sec. |

IESP Meeting@Kobe (April 12, 2012) 6

# Fujitsu's FEFS Development towards Exascale

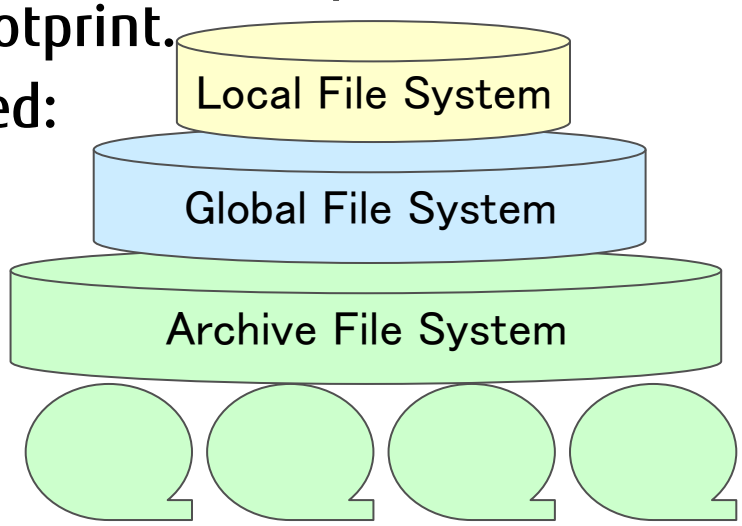
FUJITSU

- Fujitsu will continue to develop Lustre based FEFS to realize the next generation exa-scale systems.
  - Needs to continue to enhance Lustre
- FEFS already supports Exa-byte class file system size
  - However, several issues to realize real Exa-scale file system
- One of Issue is Exa-scale storage design
  - Electric Power and Footprint including Computing System and Storage: Electric Power: 20-30MW, Footprint: 2000m2(SDHPC)
  - Electric power for storage system must be minimized because most of the power should be used for computing.
  - Power Consumption of Exa-byte class Storage System: Should be Less than 1MW (as assumption)

# Exascale File System Design

FUJITSU

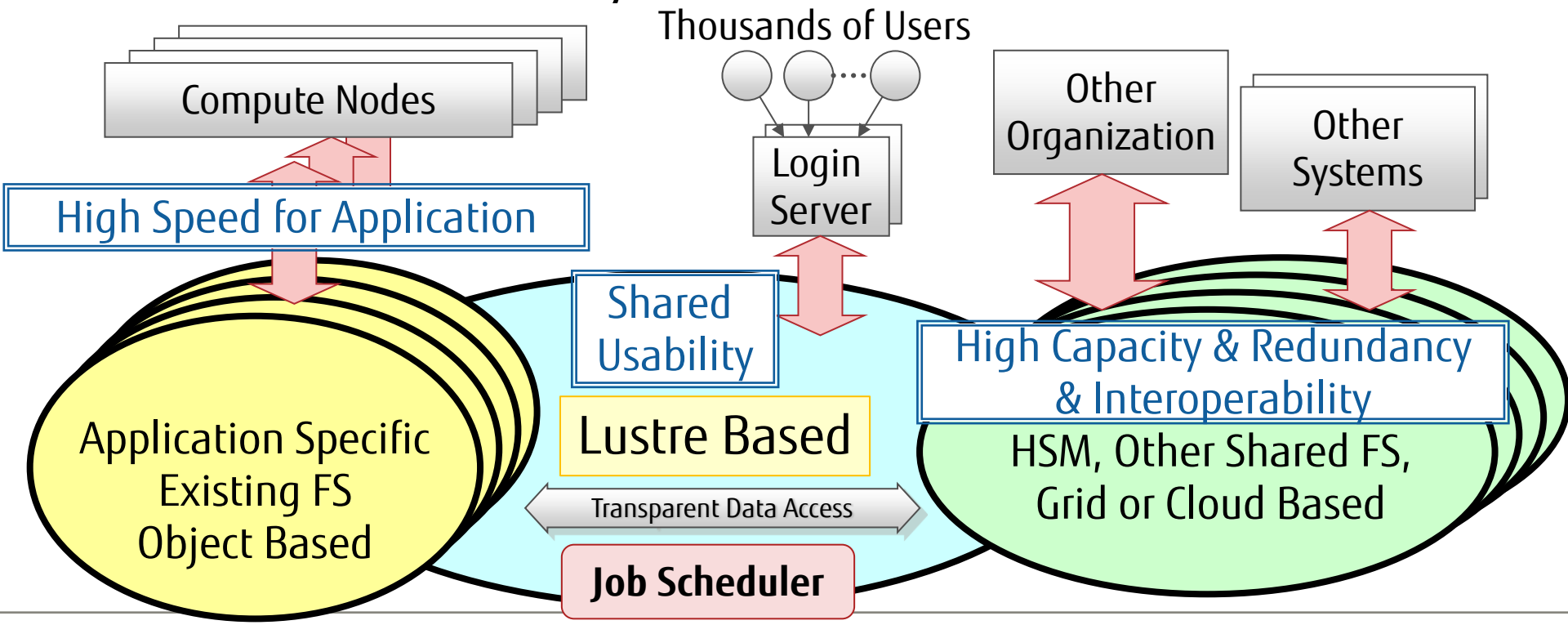
- K computer File System Design
  - How should we realize High Speed and Redundancy together?
  - How do we avoid I/O conflicts between Jobs?
  - These are not realized in single file system.
    - Therefore, we have introduced Integrated Layered File System.
- Exascale File System/Storage Design
  - Another trade off targets: Power, Capacity, Footprint
    - Difficult to realize single 1EB and 10TB/s class file system in limited power consumption and footprint.
  - Third Storage layer for Capacity is needed: Three Layered File System
    - Local File System for Performance
    - Global File System for Easy to Use
    - Archive File System for Capacity



# The Next Integrated Layered File System Architecture for Post-peta scale System (Feasibility Study 2012-2013)

FUJITSU

- Local File System o(10PB): Memory, SSD, HDD Based
  - Application Specific, Existing FS, Object Based, etc..
- Global File System o(100PB): HDD Based
  - Lustre Based, Ext[34], Object Based, Application Specific etc..
- Archive System o(1EB): HSM(Disk+Tape), Grid, Cloud Based
  - HSM, Lustre, other file system



# Issues for Post-Petascale File/Storage System

FUJITSU

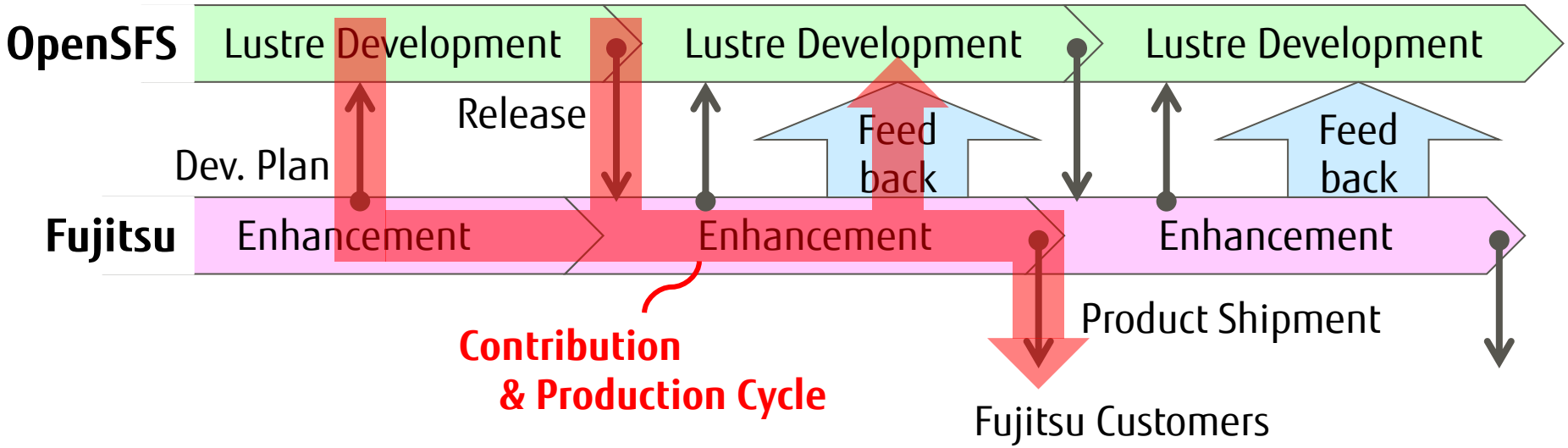
- Power Saving Storage Architecture for 1EB Class storage
  - 20MW-30MW: Total System Power including Computing System
  - Required Total(Compute and Storage) power management
- Lustre is not ready for EXA byte size systems
  - FEFS and GPFS are ready, so current Lustre needs to expand its limits. It also limits specification of Lustre 2.x based FEFS
- Issues for Realizing Post-Petascale File System:
  - How to realize application specific high speed file access to the local file system? – Needs to investigate storage access pattern of target applications
  - How to realize transparent file access among three file systems? – Lustre HSM is one of options.

FUJITSU

# FUJITSU'S CONTRIBUTION POLICY TO LUSTRE COMMUNITY

# Fujitsu' Lustre Contribution Policy

- Fujitsu will open its development plan and feed back it's enhancements to Lustre community
  - LAD is the most suitable place to present and discuss.
- Fujitsu's basic contribution policy:
  - Opening development plan
  - Feeding back its enhancements to Lustre community no later than after a certain period when our product is shipped.



# Fujitsu's Activities to Lustre Community

FUJITSU

- Step 1 (2012-2013): Basic Enhancement for Core Lustre Modules with Whamcloud/Intel

- Step 2 (2014- ): Advanced Function Contribution by Fujitsu.

# Fujitsu Contributions of Basic Enhancement

FUJITSU

- Fujitsu ported our enhancements into Lustre 2.x with Intel

| Jira | Function | Landing |
|---|---|---|
| LU-2467 | Ability to disable pinging | Lustre 2.4 |
| LU-2466 | LNET networks hashing | Lustre 2.4 |
| LU-2934 | LNET router priorities | Lustre 2.5 |
| LU-2950 | LNET read routing list from file | Lustre 2.5 |
| LU-2924 | Reduce ldlm_poold execution time | Lustre 2.5 |
| LU-3221 | Endianness fixes (SPARC support) | Lustre 2.5 |
| LU-2743 | Errno translation tables (SPARC Support) | Lustre 2.5 |
| LU-4665 | lfs setstripe to specify OSTs | Lustre 2.7 |

Bug-fixes are not included

# Fujitsu Contributions of Advanced Functions

FUJITSU

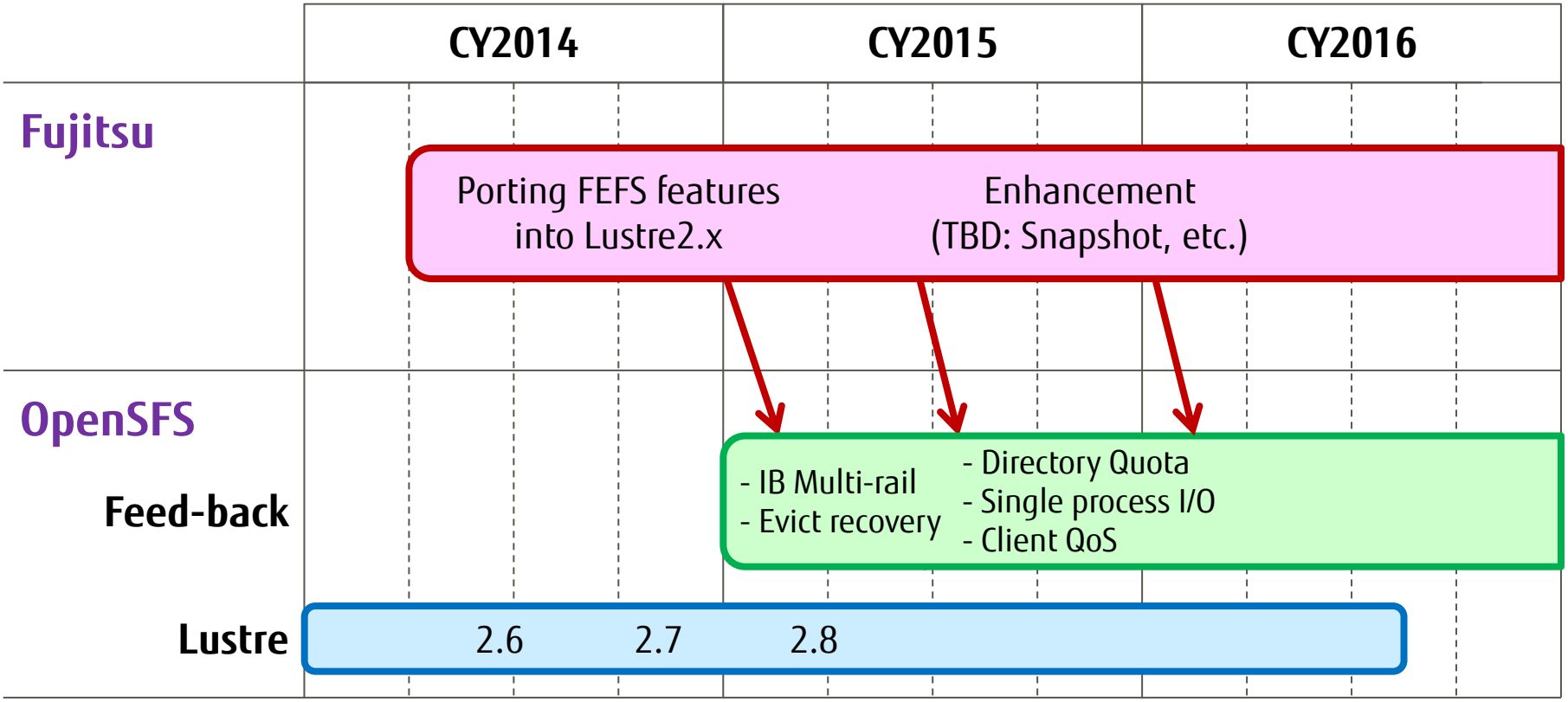
- Fujitsu's now been porting our enhancements into Lustre 2.x
  - These features were implemented in Lustre 1.8 based FEFS
  - They've been used in our customer's HPC system, including K computer
- We'll start submitting patches for Lustre in 2015

| Functions | Submitting Schedule |
|---|---|
| IB multi-rail | Jan. 2015 |
| Automated Evict Recovery | Apr. 2015 |
| Directory Quota | 2nd half of 2015 |
| Improving Single Process IO Performance | 2nd half of 2015 |
| Client QoS | 2nd half of 2015 |
| Server QoS | TBD |
| Memory Usage Management | TBD |

# Fujitsu's Contribution Roadmap

FUJITSU

- Fujitsu's development and community feedback plan
  - Schedule may change by Fujitsu's development/marketing strategy



# Advanced Function(1)

FUJITSU

- InfiniBand (IB) Multi-rail
  - Multiple InfiniBand(IB) interfaces as a single Lustre NID
  - Improving Data Transferring Bandwidth on a single Lustre node
  - Improving Redundancy against Failures of IB.
  - Achieved about 11GB/s read/write performance with two FDR IB HCAs (Single 6GB/s)
  - Tested with upto four IB HCA devices

- Directory Quota able to:
  - Use Directory Quota (DQ for short) feature in the same way of Lustre's UID/GID quota function
  - Limit the number of inodes and disk blocks to each directory specified by user
  - Be managed by lfs command like UID/GID quota of Lustre.

# Advanced Function(2)

FUJITSU

- Improvement of single process IO performance
  - Improving single process IO performance
  - Our prototype results: Over 2GB/s bandwidth twice as fast as Lustre 2.5.
- Client QoS
  - Provides Fair Share accesses among users on a single Lustre client
  - On a multi user client, when one user issues large amount of IO, the IO performance of the other users are terribly degrade.
  - Client QoS feature prevents this performance issue by controlling the number of IO requests issued by each user.
- Automated Evict Recovery
  - When a Lustre server evicts a client, the server notifies the client to reconnect the server. This occurs IO error to user application
  - Minimizing the evicting status of Lustre clients especially disable pinging feature is enabled
  - Reducing the occurring of IO error to user application.

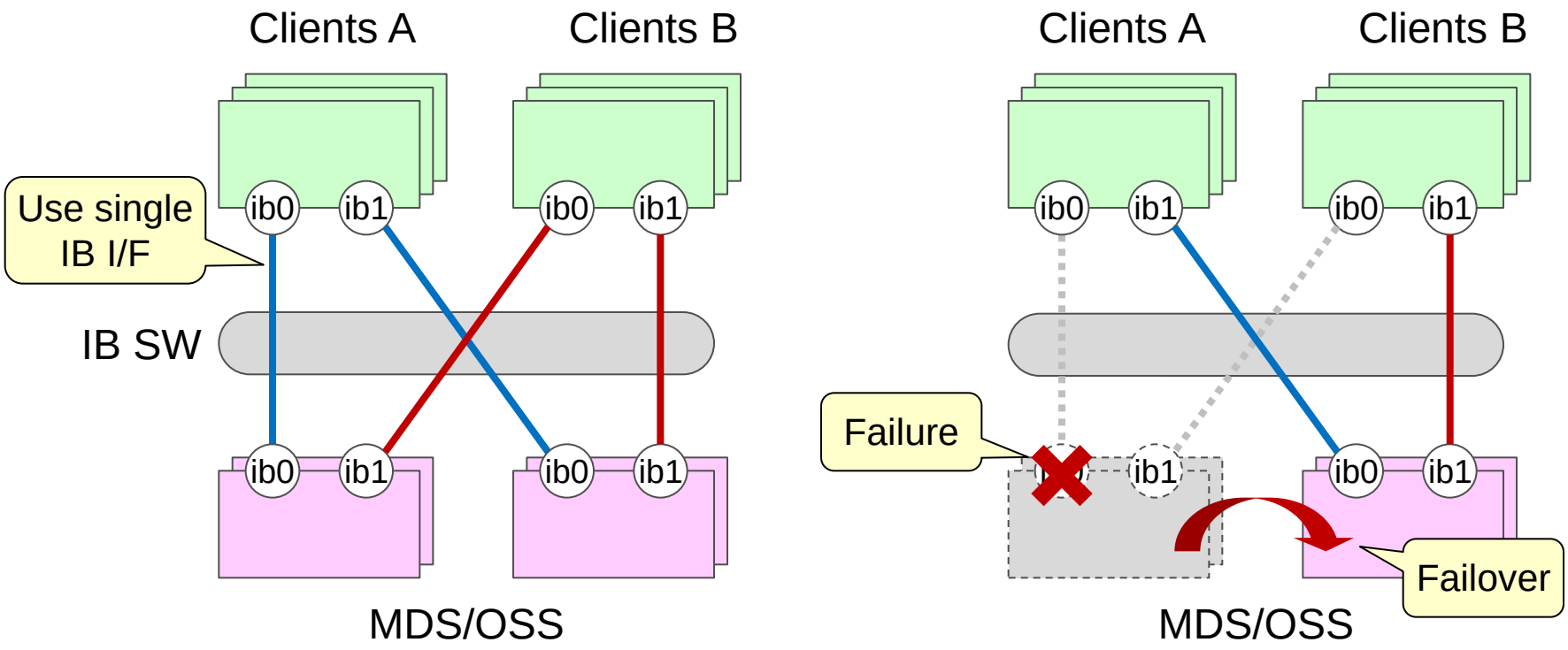
FUJITSU

# INTRODUCTION OF CONTRIBUTION FEATURES

InfiniBand (IB) Multi-rail
Directory Quota
Improving Single Process IO Performance

# Issue of Current Lustre IB Multi-rail

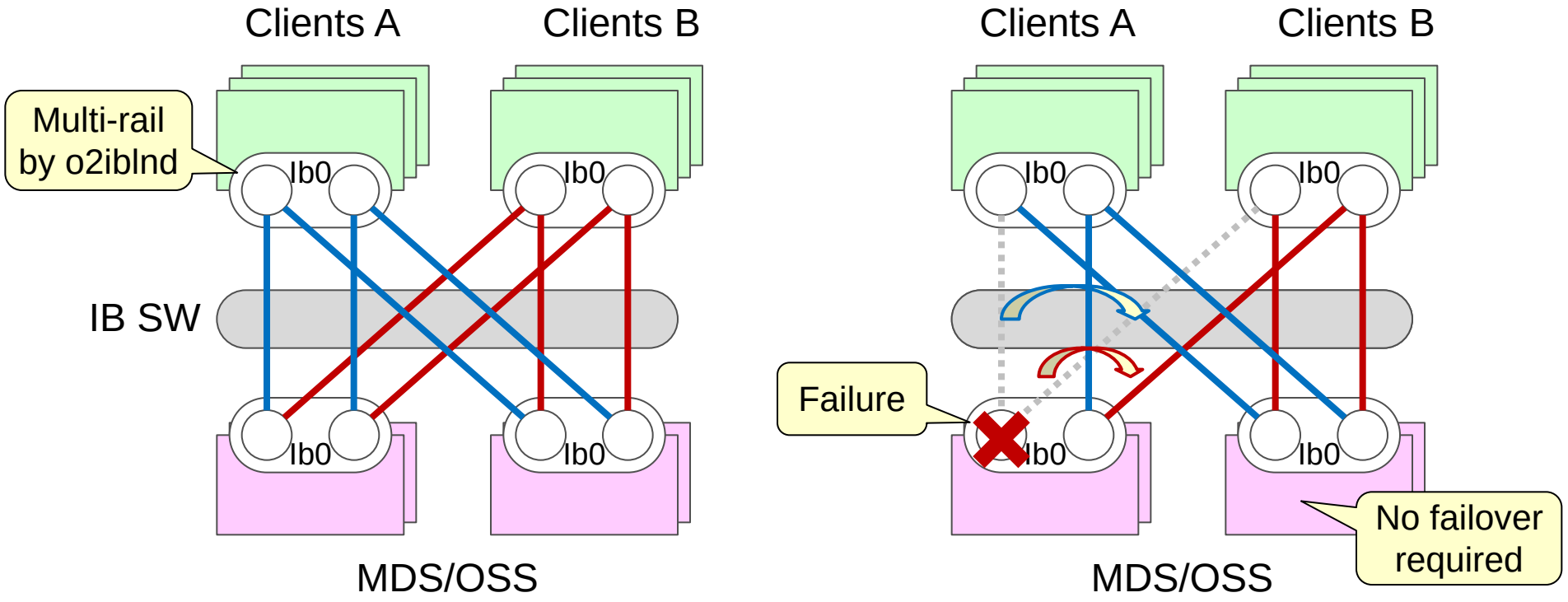
- Client, MDS and OSS can not use multiple IB I/F.
  - Single IB I/F failure in a server (MDS/OSS) cause failover.
  - Client can use only one IB I/F when accessing a server.



# FEFS IB Multi-rail

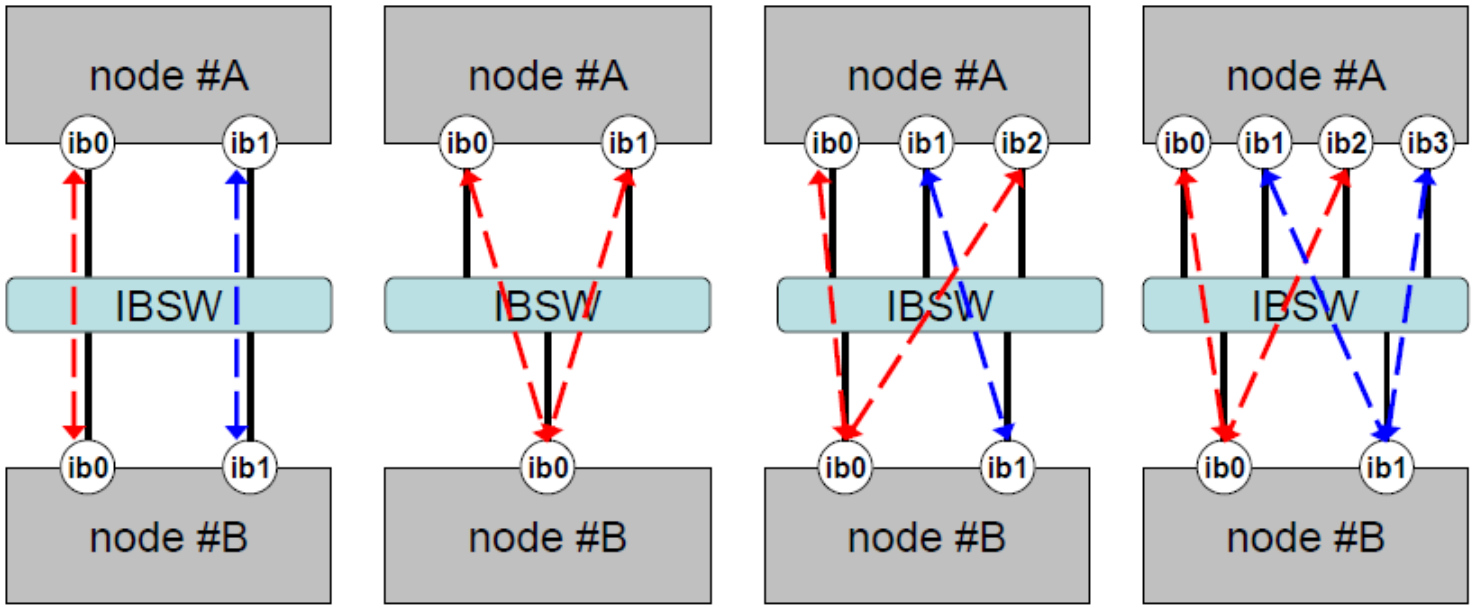
FUJITSU

- FEFS Approach: Add IB multi-rail function into Lustre network driver (o2iblnd).
  - All IB I/F on the client can be used to communicate with a server.
  - All IB connections are used by round-robin order.
- Continue communication when single point of IB failure occurs.
  - All IB connections are used by round-robin order by each requests.



# Variation of Multi-Rail

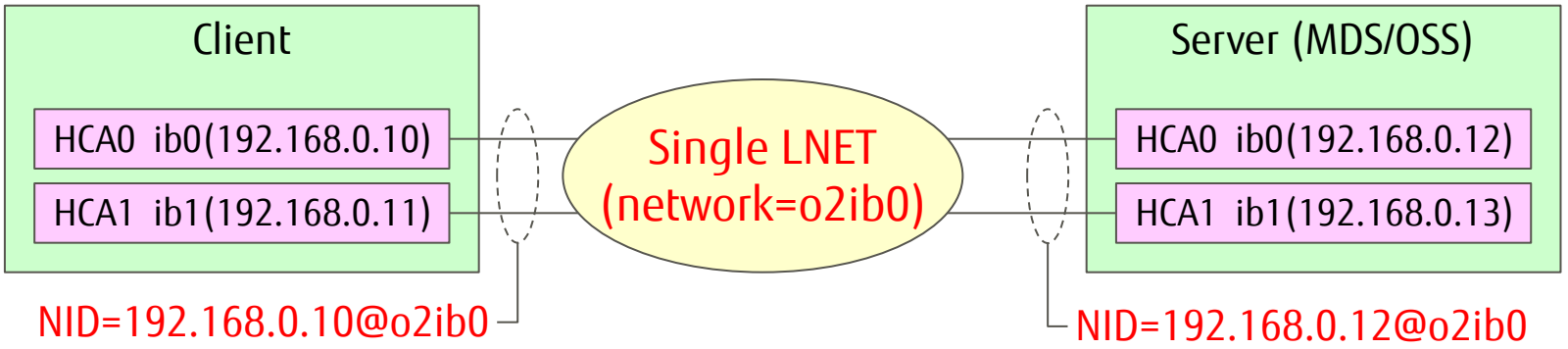
- Not only symmetric connection
  but also asymmetric connection for every node pair.
- User can realize flexible configuration



# IB Multi-Rail: How to Use

FUJITSU

## Combining single NID width multiple IB interfaces




## LNET setting (modprobe.conf)

```
options lnet networks=o2ib0(ib0,ib2)
```

## NID/IPoIB definition

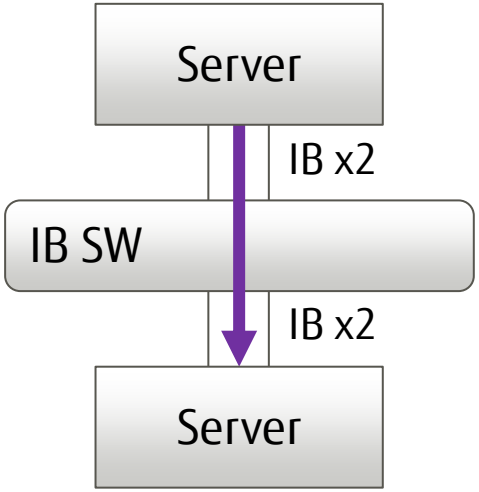
```
# lctl –net  o2ib0  add_o2ibs  192.168.0.10@o2ib0  192.168.0.10 192.168.0.11  → Client
# lctl –net  o2ib0  add_o2ibs  192.168.0.12@o2ib0  192.168.0.12 192.168.0.13  → Server
```

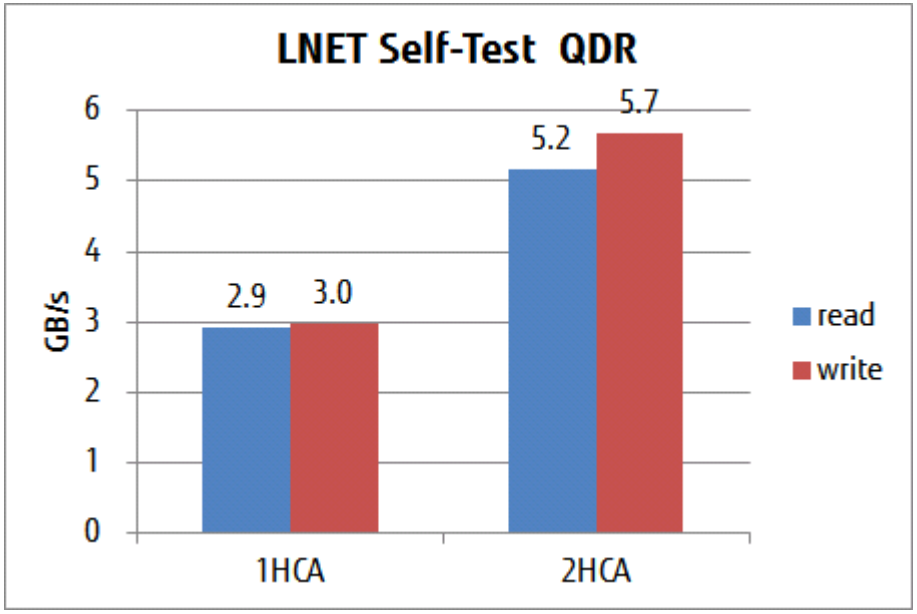
## Display multi-rail information

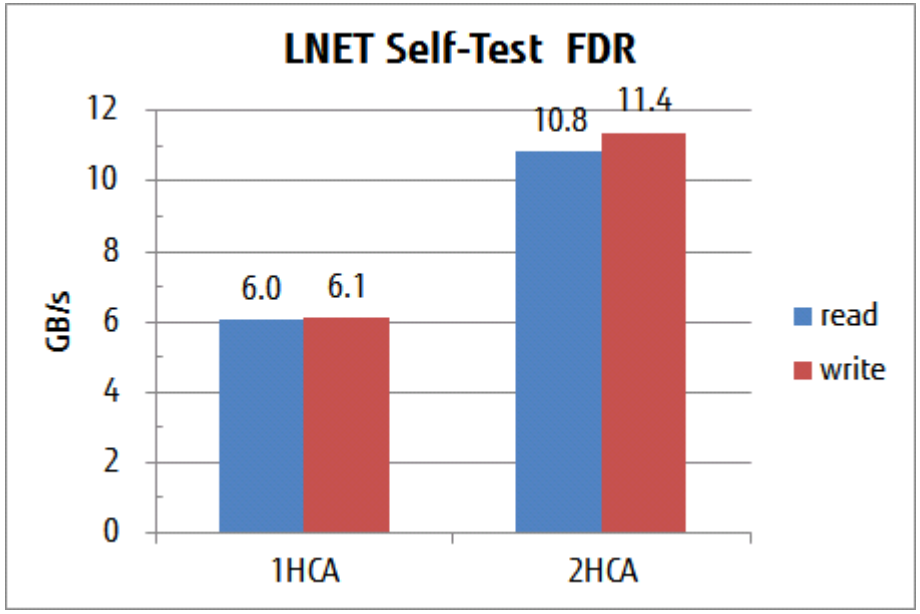
```
# lctl --net o2ib0 show_o2ibs
192.168.0.10@o2ib0 192.168.0.10 192.168.0.11
192.168.0.12@o2ib0 192.168.0.12 192.168.0.13
```

# IB Multi-Rail: LNET Performance

- Server
  - CPU: Xeon E5520 2.27GHz x2
  - IB: QDR x2 or FDR x2
- Result
  - B/W almost scales by #IBs
  - Achieves nearly HW performance



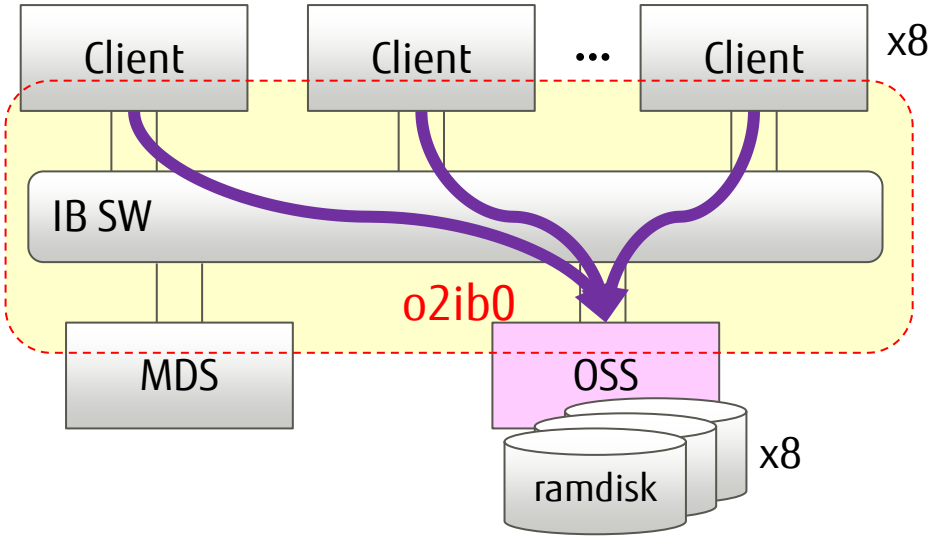



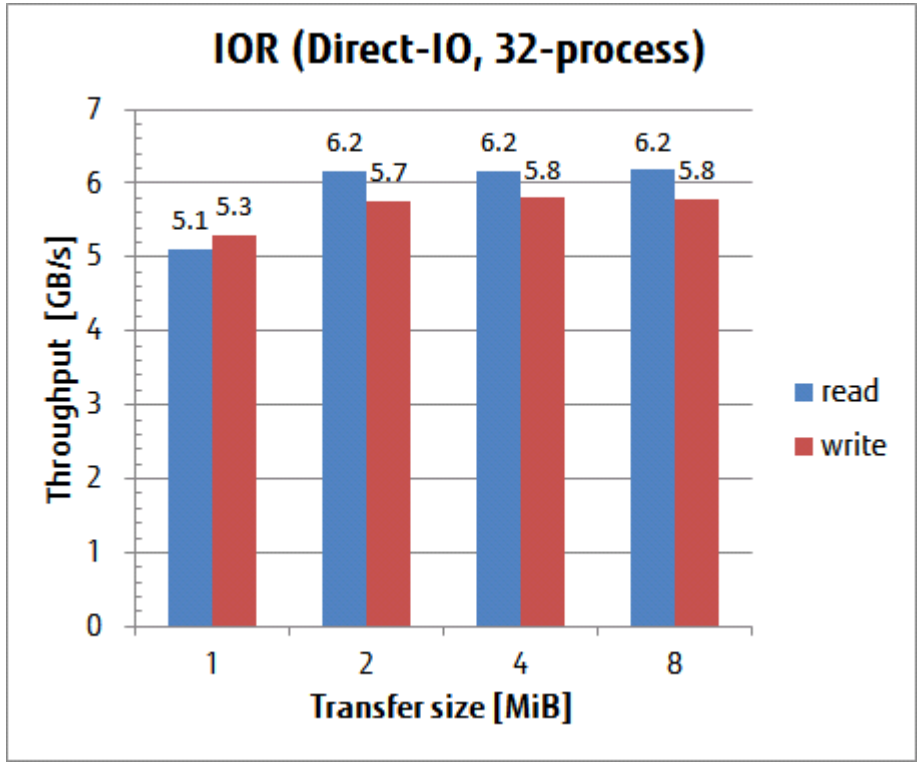

(Concurrency=32)

# IB Multi-Rail: IO Throughput of Single OSS

FUJITSU

- OSS/Client
  - CPU: Xeon E5520 2.27GHz x2
  - IB: QDR x2
- OST
  - ramdisk x8 (> 6GB/s)
- IOR
  - 32-process (8client x4)
- Result
  - Throughput almost scales by #IBs
  - Measurement of FDR is planned



# Directory Quota : Features

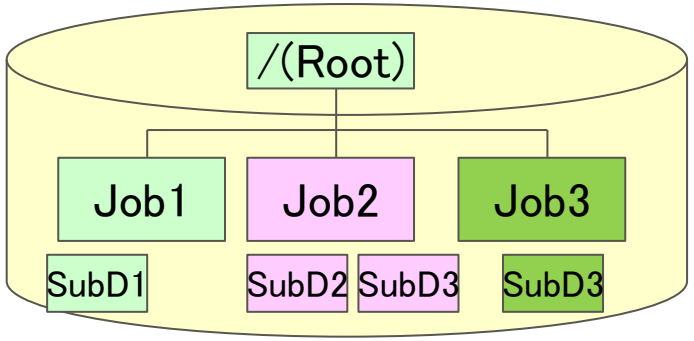
FUJITSU

- What is Directory Quota?
  - Restricting #inodes&blocks by individual directories
  - All files/directories under the DQ-enable directory are under Quota accounting
- Fujitsu is now implementing Directory Quota (DQ) function into Lustre 2.x
  - DQ of FEFS based on Lustre 1.8 has been used in production systems for more than two years.
- Will be Implemented on top of the Disk Quota framework
  - DQ can be used along with disk Quota

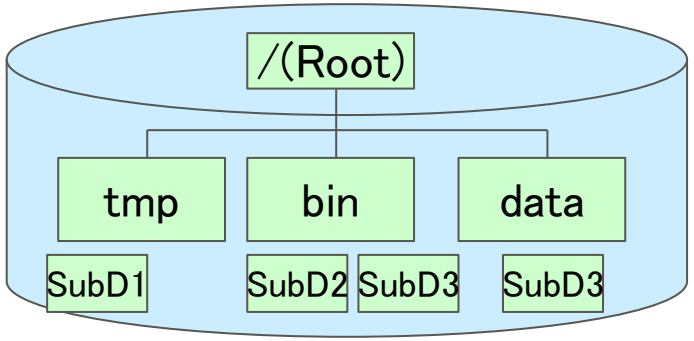
# Directory Quota(DQ) : Use Image

FUJITSU

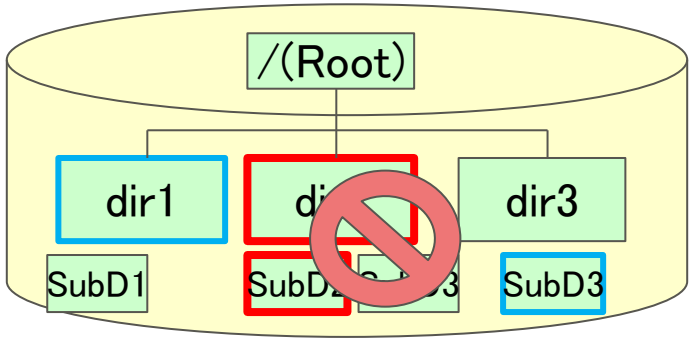
- Use Case1: for Job Directory
  - DQ can control file system usage for each job




- Use Case1: for Shared Directory
  - Of course, DQ can control shared directories for their usage




- Limitation
  - Nested DQ directories are not permitted, because of simplicity of implementation and performance



# Directory Quota: How to Use

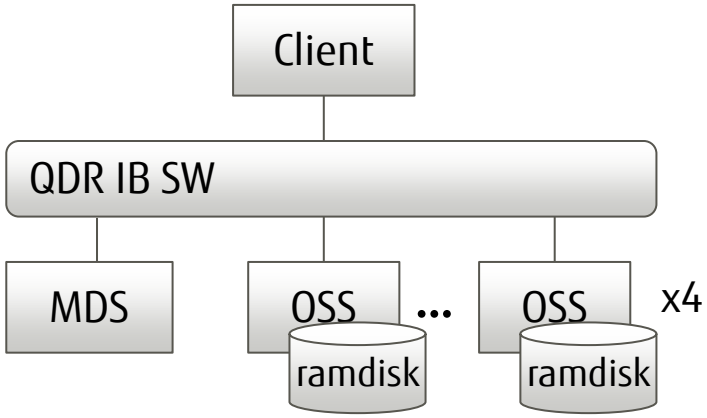
FUJITSU

- Operations are same as Lustre's UID/GID Quota
  - Only "quotacheck" operation differs
- Set DQ on target directory (=DQ-directory)
  - # lfs **quotacheck –d <target dir>**
    - Counts the number of inodes&blocks of existing files under DQ-directory
- Set limits of inodes and blocks
  - # lfs setquota **–d <target dir>** -B <#blk> -I <#inode> <mountpoint>
- Enable limiting by DQ
  - # lctl conf_param <fsname>.quota.<ost|mdt>=<ug**d**>
  - # lctl set_param -P <fsname>.quota.<ost|mdt>= <ug**d**>
- Check status
  - # lctl get_param osd-*.*.quota_slave.info

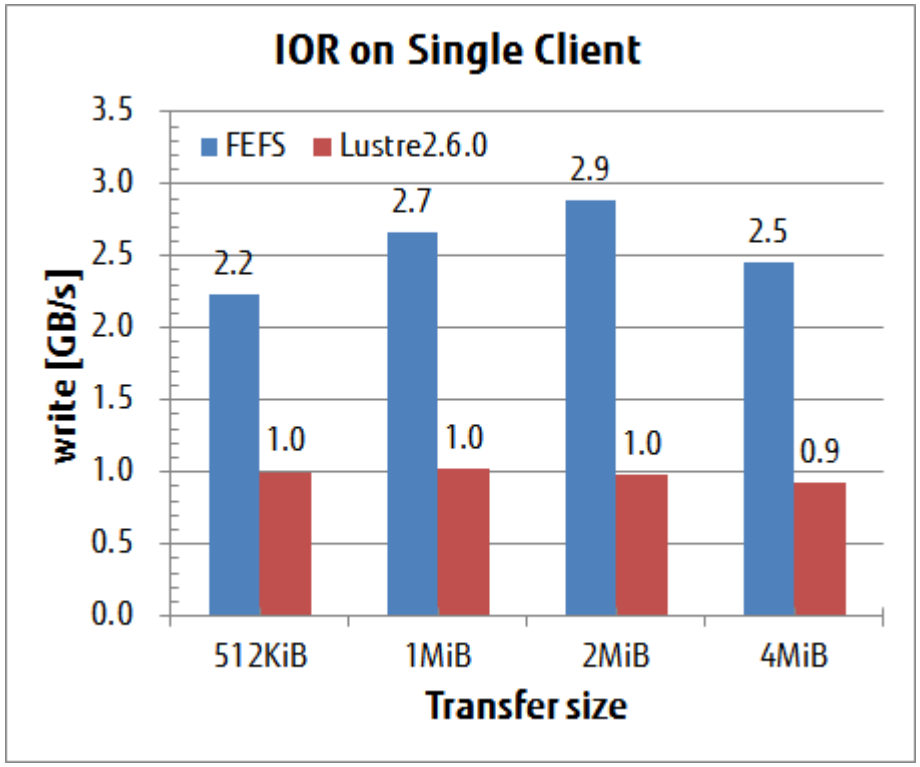
# Improving Single Process IO Performance

FUJITSU

- Comparison between Lustre 2.6.0 and prototype (Lustre 1.8 base)
  - We've been re-designing implementation suiting Lustre 2.x
- OSS/Client
  - CPU: Xeon E5520 2.27GHz x2
  - IB: QDR x1
- OST
  - ramdisk x4
- IOR
  - 1-process

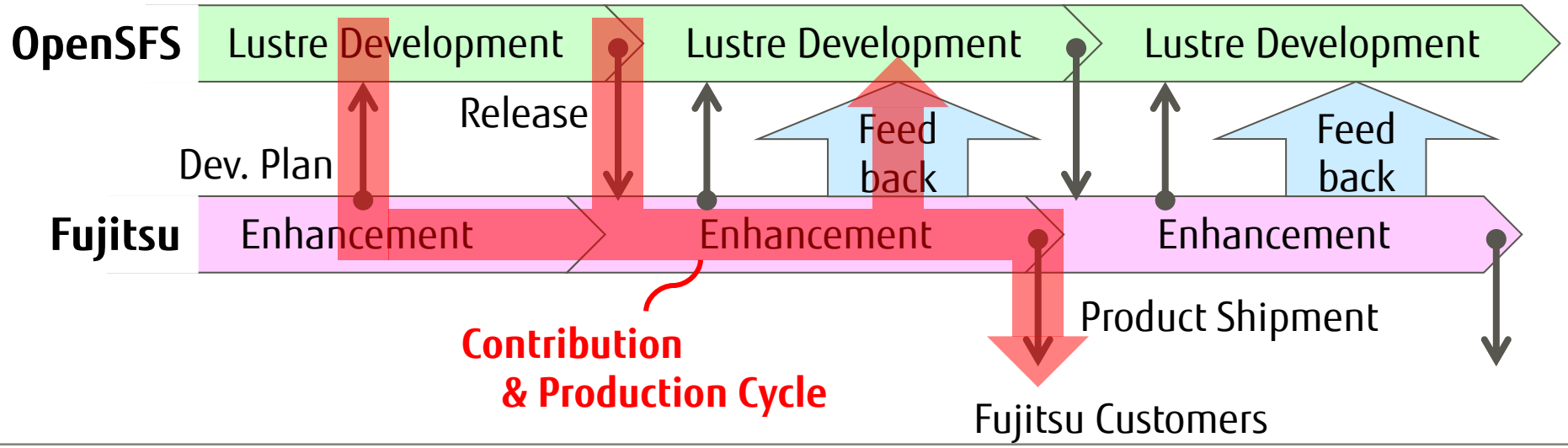


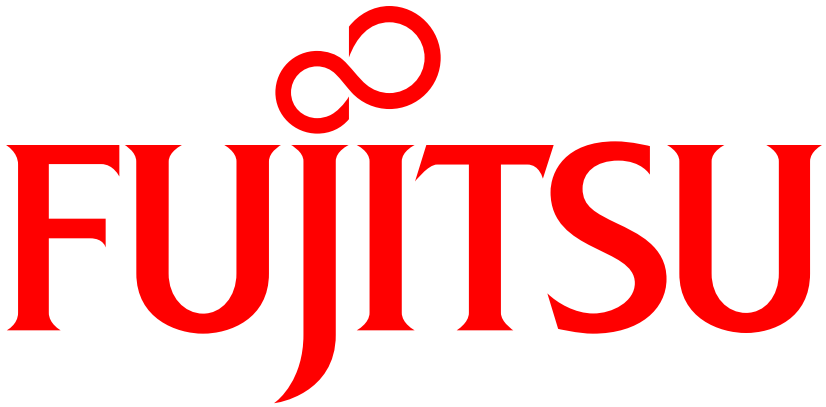

- Result
  - Lustre 2.6.0 0.9~1.0GB/s
  - Prototype 2.2~2.9GB/s



# Summary

- Fujitsu will continue to improve Lustre for exascale systems.
- Fujitsu will open its development plan and feed back it's enhancements to Lustre community
  - LAD is the most suitable place to present and discuss.
- Several Features will be scheduled to be contributed
  - InfiniBand Multi-rail, Direcotry Quota etc...






shaping tomorrow with you