Высокопроизводительная и высоконадежная файловая система для К компьютера

•Kenichiro Sakai •Shinji Sumimoto •Motoyoshi Kurokawa
FUJITSU Sci. Tech. J., Vol. 48, No. 3, pp. 302–309 (July2012)

RIKEN и Fujitsu разработали самый быстрый в мире суперкомпьютер, К компьютер. В дополнение к более чем 80 000 вычислительных узлов, К компьютер имеет систему хранения с емкостью в несколько десятков Петабайт и пропускной способностью ввода/вывода превышающей один терабайт в секунду. Это самая большая и самая быстрая система хранения в мире (прим.перев.: на момент написания статьи). Для того, чтобы воспользоваться преимуществами этой огромной системы хранения, а также достичь высокой масштабируемости и высокой стабильности, мы разработали экза-байтовую файловую систему Fujitsu (FEFS), кластерно- распределенную файловую систему. В этой статье излагается описание файловой системы К компьютера и вводятся показатели, принятые в FEFS для решения ключевых вопросов в крупномасштабной системе.

1. Введение

Драматические скачки в производительности суперкомпьютеров в последние годы привели к достижению 10 Пета операций с плавающей запятой в секунду в 2011 году. Возрастающее число вычислительных узлов и ядер, а также растущий объем оперативной памяти означает, что файловая система должна иметь большую емкость и общую пропускную способность. Емкость файловой системы и производительность, как ожидается, достигнет класса 100-Петабайт (PB) и 1-Терабайт/с (TB/s), соответственно, причем и емкость и производительность увеличиваются примерно с десятикратной скоростью в год.

По этой причине господствующие файловые системы изменились с односерверного типа на кластерный тип. Мы разработали кластерно распределенную файловую систему, называемую Экзабайтовой файловой системой Fujitsu (FEFS, Fujitsu Exabyte File System), с целью получения файловой системы, имеющей емкость и производительность, которые обеспечат вычислительную производительность К компьютера[note)], который в настоящее время (прим. перев.: 2012) является самым быстрым суперкомпьютером в мире.

В этой статье излагается описание файловой системы К компьютера и вводятся показатели, принятые в FEFS для решения ключевых вопросов в крупномасштабной системе.

2. Файловая система для К компьютера

Для достижения уровней производительности и стабильности выполнения работы, соответствующего самому быстрому в мире суперкомпьютеру, мы применили двухуровневую модель для файловой системы К компьютера. Эта модель состоит из локальной файловой системы, используемой для высокоскоростной области временного хранения только для заданий и глобальной файловой системы, используемой в качестве разделяемой системы хранения большой емкости для хранения пользовательских файлов (**Рисунок 1**). В этой двухуровневой модели задания выполняются посредством передачи данных между этими двумя файловыми системами с использованием функции управления конвейеризацией файлов. Ниже описываются соответствующие роли локальной файловой системы, глобальной файловой системы и процесса конвейеризации файлов.

note) “K computer” - английское название, которое RIKEN использовал для суперкомпьютера в данном проекте начиная с июля 2010. “K” пришло из японского слова “Kei,” которое обозначает 10пета или 10 в 16й степени.

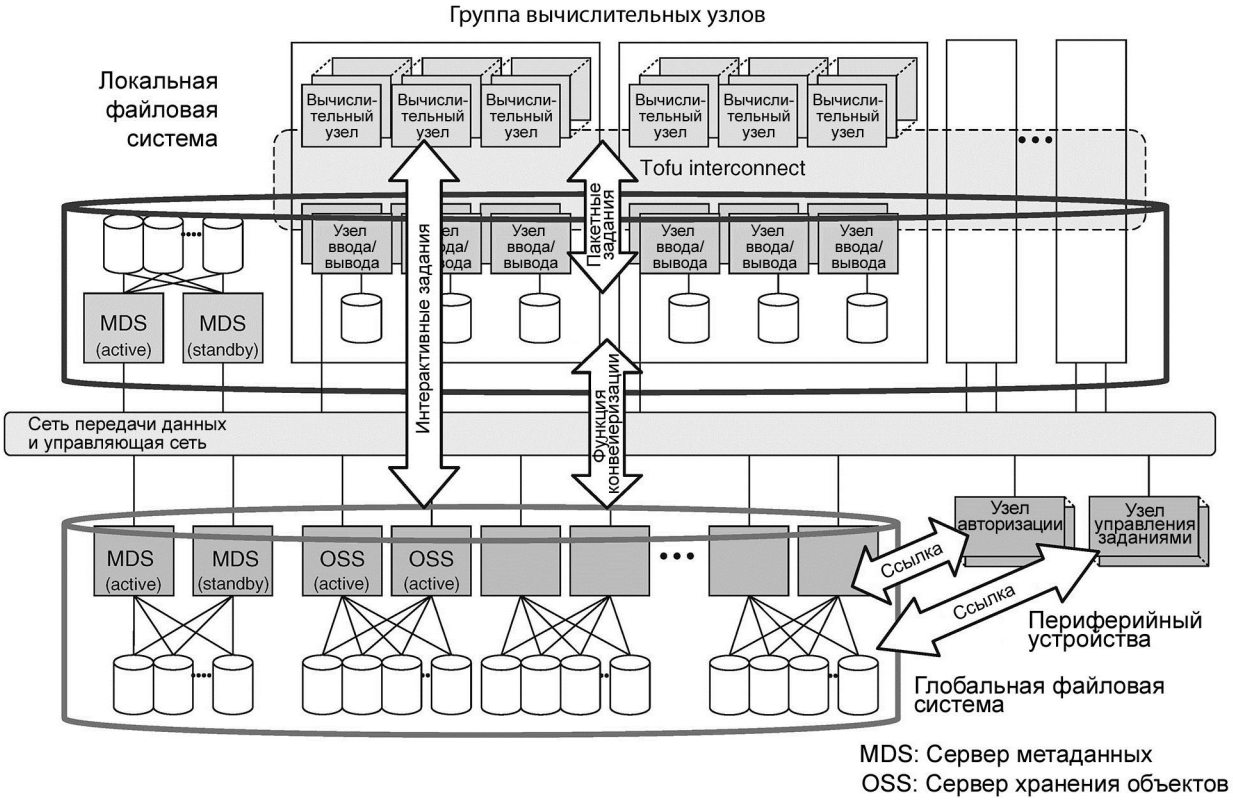

Рисунок 1
Двухуровневая модель файловой системы

1) Локальная файловая система

Это высокоскоростная область временного хранения, выделяемая выполняемому заданию для извлечения максимальной производительности файлового ввода/вывода для приложений, выполняемых как пакетные задания. Локальная файловая система перемещает файлы ввода/вывода из/в глобальную файловую систему с помощью функции конвейеризации файлов и временно хранит файлы, которые либо исполняются, либо ожидают исполнения.

Файл-сервер, используемый для доступа к блокам данных, подключается к вычислительным узлам через интерконнект Tofu с использованием выделенного узла ввода/вывода (I/O node) в вычислительной стойке посредством коммуникации удаленного доступа к памяти (RDMA). В результате достигается низколатентный обмен файлами данных с высокой пропускной способностью.

2) Глобальная файловая система

Это область разделяемой системы хранения данных большой емкости, внешняя по отношению к компьютерным стойкам, для хранения пользовательских файлов, состоящих из данных ввода/вывода заданий и другого контента. В дополнение к предоставлению доступа к файлам через узел авторизации, глобальная файловая система позволяет осуществлять прямой доступ к файлам из вычислительного узла при выполнении интерактивного задания, так что пользователь может выполнять программы отладки и настройки, проверяя вывод задания в режиме реального времени.

Вычислительный узел подключается к группе файловых серверов в глобальной файловой системе через соединение Infiniband, установленное на выделенном узле ввода/вывода. Короче говоря, узлы ввода/вывода передают данные между интерконнектами Tofu и InfiniBand в то время, как вычислительный узел обращается к файловым серверам через узлы ввода/вывода.

3) Конвейеризация файлов

Передача файлов между локальной и глобальной файловыми системами выполняется системой автоматически с помощью функции конвейеризации файлов, которая работает совместно с выполняемым заданием программного обеспечения. Перед началом работы эта функция передает входные файлы для данного задания из глобальной файловой системы в локальную файловую систему (входной конвейер), а после завершения задания она передает выходные файлы из локальной файловой системы в глобальную файловую систему (выходной конвейер). Пользователь описывает файлы входного и выходного конвейеров в скрипте задания.

3. FEFS: кластерная файловая система ультра большого масштаба

Мы разработали FEFS, чтобы обеспечить локальной и глобальной файловыми системами, соответствующими производительности топ-класса К компьютера. Были установлены следующие цели для разработки FEFS:

- Высочайшая в мире скорость ввода/вывода и высокопроизводительный MPI ввод/вывод (Message Passing Interface I/O)
- Стабильное выполнение заданий, достигаемое за счет устранения прерываний
- Наивысшая в мире емкость файловой системы
- Более высокая масштабируемость по производительности и емкости, достигаемая путем добавления аппаратных средств
- Высокая надежность (непрерывность обслуживания и сохранения данных)
- Легкость в использовании (разделение большим количеством пользователей)
- Справедливое разделение (справедливое совместное использование большим количеством пользователей)

Для достижения этих целей мы разработали FEFS на основе файловой системы с открытым кодом Lustre, которая стала мировым отраслевым стандартом, с расширением по мере необходимости определенных характеристик и функций. Как показано в **Таблице 1**, FEFS расширяет такие характеристики, как максимальный размер файловой системы и максимальный размер файла до класса 8-Экзабыйт (EB) относительно существующих спецификаций Lustre. Размер файловой системы может быть расширен даже во время работы системы за счет добавления новых серверов и устройств хранения данных.

В следующих разделах описываются перечисленные ниже меры FEFS для оперирования с ультра-крупномасштабными суперкомпьютерными системами.

- Исключение конфликтов ввода/вывода с использованием коммуникационных шин и разбиения дисков
- Постоянная доступность с помощью аппаратного дублирования и восстановления после сбоев
- Иерархический мониторинг узлов и автоматическое переключение
- Выбор QoS-политики по типу операций

Таблица 1
Сравнение Lustre и FEFS

Функции		FEFS	Lustre
Описание системы (макс. значения)	Размер файловой системы	64 PB	8 EB
	Размер файла	320 TB	8 EB
	Число файлов	4G	8 E
	Размер тома OST	16 TB	1 PB
	Число полос	160	20000
	Число записей ACL	32	8191
Масштабируемость	Число OSS	1020	20000
	Число OST	8150	20000
	Число клиентов	128 units	1 000 000 units
Размер блока (файловая система Lustre) (файловая система сервера СУБД)		4 KB	до 512 KB

4. Исключение конфликтов ввода/вывода с использованием коммуникационных шин и разбиения дисков

В качестве кластерной файловой системы, FEFS собирает вместе большое количество файловых серверов и улучшает параллельную пропускную способность пропорционально количеству файловых серверов. Однако, если доступ должен сконцентрироваться на конкретных файловых серверах, то связь может стать перегруженной и может возникнуть конфликт доступа к дискам, что приведет к падению производительности ввода/вывода и различиям во времени работы между вычислительными узлами. Поэтому очень важно для обеспечения стабильного выполнения задания полностью устранить конфликты файлового ввода/вывода.

В FEFS, конфликт файлового ввода/вывода устраняется путем разделения доступа файлового ввода/вывода на две стадии, а именно: на модули заданий и модули вычислительных узлов (**Рисунок 2**).

На уровне заданий каждое задание назначается отдельному диапазону узлов ввода/вывода, соответствующему файловому хранилищу -получателю для исключения конфликта файлового ввода/вывода на серверах, в сети и на дисках. А на уровне вычислительных узлов в пределах задания, передача и прием файловых данных осуществляется с помощью узлов ввода/вывода составляющих минимальное количество ретрансляций Tofu-коммуникацей, тем самым минимизируя конфликты файлового ввода/вывода между вычислительными узлами.

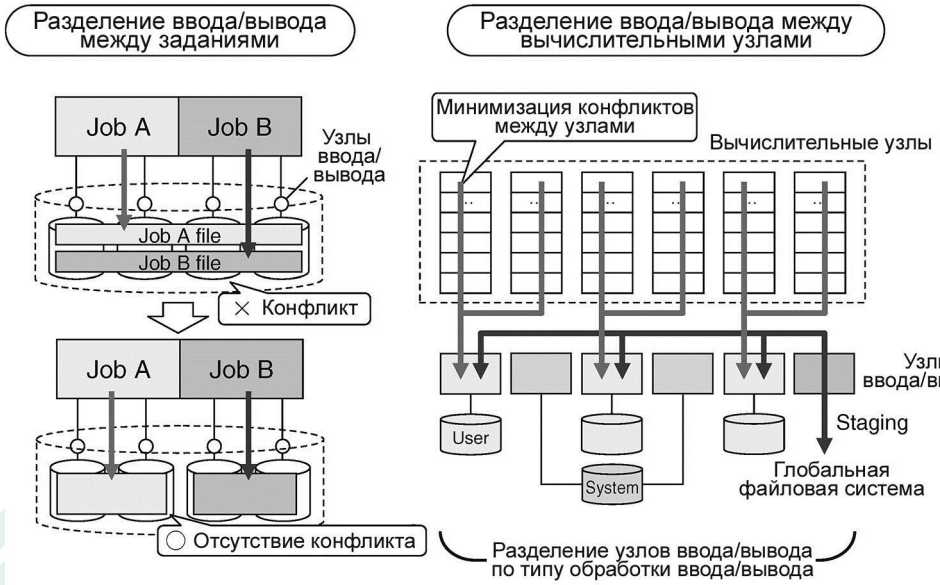

Рисунок 2
Устранение конфликтов разделением ввода/вывода

5. Постоянная доступность с помощью аппаратного дублирования и восстановления после сбоев

В дополнение к высокой производительности, другим важным требованием крупномасштабной компьютерной системы является высокая надежность. Кластерная файловая система состоит из большого количества файловых серверов, устройств хранения данных и сетевых устройств и любые неисправности или сбои в любом из них или даже временная приостановка на обслуживание не должны препятствовать продолжению обслуживания файловой системы и работы системы.

Однако при настройке FEFS с файловыми серверами и устройствами хранения данных, исчисляемыми несколькими сотнями, обязательно существуют случаи, когда аппаратные средства, такие как сетевые адаптеры и сервера находятся в состоянии неисправности или технического обслуживания. Чтобы убедиться, что операции во всей системе продолжатся даже в таких условиях, очень важно, чтобы ошибки обнаруживались автоматически и создавались обходные пути вокруг мест отказа так, чтобы обслуживание файловой системы могло беспрепятственно продолжаться.

По этой причине FEFS дублирует аппаратные средства и использует процесс управления на основе программного обеспечения для начала процесса восстановления во время возникновения аппаратного сбоя. Эта обработка переключает сервер и пути коммуникации ввода/вывода, а также восстанавливает службы до нормального состояния. Это предотвращает от останова системных служб при возникновении одиночных ошибок и позволяет продолжаться системным операциям (**Рисунок 3**).

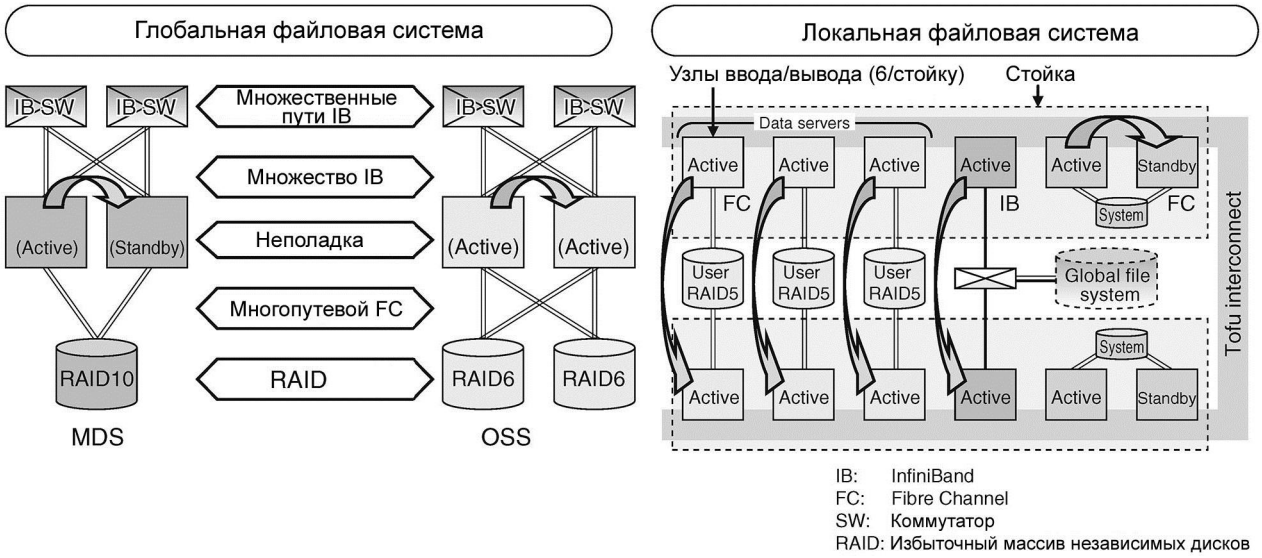

Рисунок 3
Устойчивость к сбоям за счет аппаратного дублирования

6. Иерархический мониторинг узлов и автоматическое переключение

Крупномасштабная компьютерная система должна иметь механизм для обнаружения повреждений без вмешательства человека и для автоматического уведомления узлов в пострадавшем диапазоне о необходимости исполнения переключения узлов. Один из методов для мониторинга состояния узлов заключается в наличии обмена пакетами мониторинга между вычислительными узлами и файловыми серверами, но это может генерировать большое количество пакетов в геометрической прогрессии, пропорциональной масштабу системы, которые могут помешать коммуникации MPI между вычислительными узлами и обмена данными между вычислительными узлами и файловыми серверами.

В качестве контрмеры для этой проблемы, FEFS выполняет иерархический мониторинг узлов и управление коммутацией совместно с программным обеспечением для управления системой, чтобы минимизировать нагрузку на системную коммуникацию. Такой подход обеспечивает эффективное и автоматическое переключение путем выполнения древовидного мониторинга состоящего из мониторинга в пределах вычислительной стойки, мониторинг в единицах групп узлов, объединенных несколькими стойками, а также мониторинга групп узлов более высокого порядка (**Рисунок 4**).

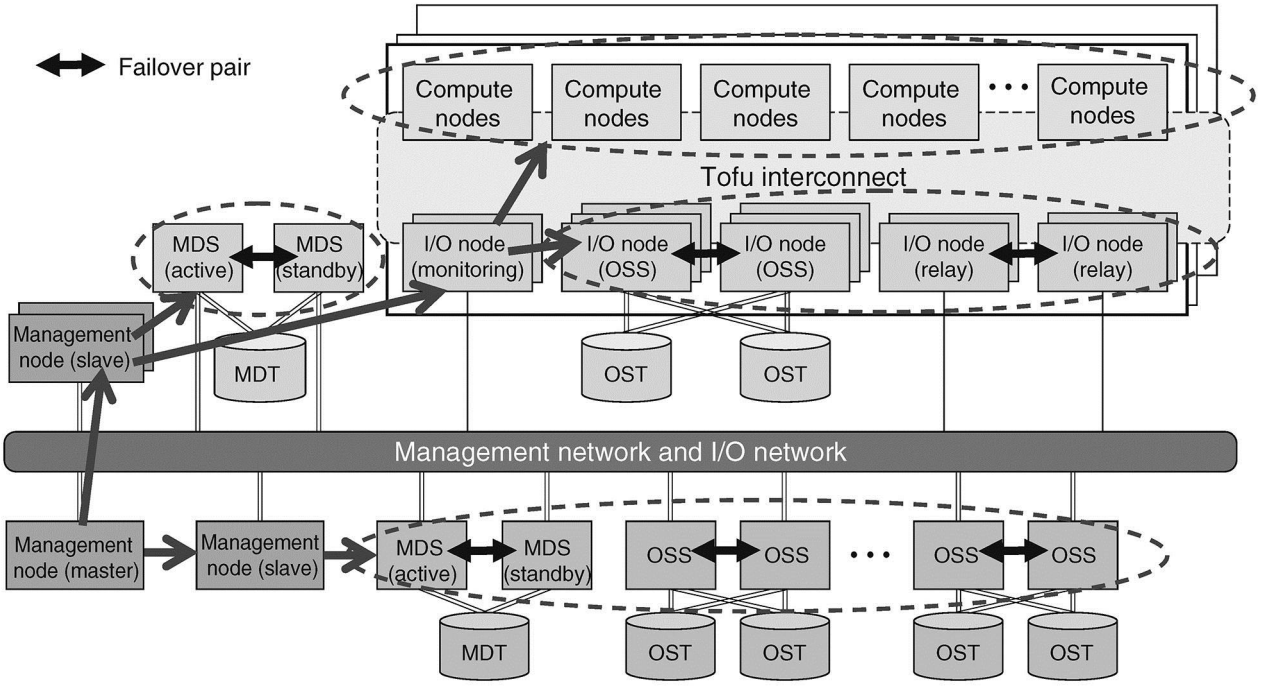

Рисунок 4
Иерархический мониторинг узлов и автоматическая коммутация

7. Выбор QoS-политики по типу операций

В крупномасштабных системах, используемых большим количеством пользователей, как и в работе дата-центров, ввод/вывод файлов большого объема, выполняемый некоторыми пользователями, не должен влиять на работу других пользователей. Кроме того, доступ к файлам из заданий не должен влиять на отклик системы для пользователей на узле авторизации.

В FEFS, эти проблемы решаются с помощью межпользовательской функции справедливого распределения для файлового ввода/вывода и функцию гарантированного ответа в режиме окружения системы разделения времени (TSS), как описано ниже.

1) Ограничение на запросы ввода/вывода, создаваемые одним пользователем

В FEFS, число запросов на ввод/вывод создаваемых на клиентской стороне и обработанных на стороне сервера управляется таким образом, что конкретный пользователь не монополизирует ресурсы ввода/вывода (сетевой ввод/вывод, серверы, дисковое оборудование), как показано на **Рисунке 5**.

На стороне клиента FEFS контролирует количество запросов ввода/вывода, которые могут быть созданы одновременно одним пользователем, чтобы не дать возможности этому пользователю создать большое количество запросов ввода/вывода и монополизировать ресурсы ввода/вывода.

Кроме того, возможна монополизация ресурсов ввода/вывода, если запросы ввода/вывода создаются приложением, принадлежащим одному и тому же пользователю от нескольких клиентов, таких как вычислительные узлы. Таким образом, на стороне файл-сервера FEFS управляет производительностью работы сервера, который может использоваться одним пользователем, чтобы предотвратить ввод/вывод ресурсов от монополизации запросами ввода/вывода от этого пользователя.

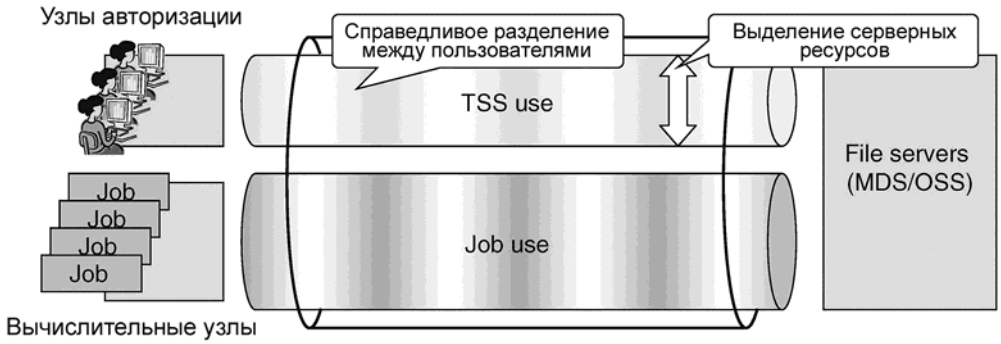

Рисунок 5
Обеспечение гарантированного отклика от узла авторизации

2) Гарантия отклика от узла авторизации

Отклик доступа, ощущаемый пользователем на узле авторизации, непосредственно связан с простотой в использовании системы и, следовательно, более важен, чем отклик доступа по отношению к заданиям.

В FEFS отклик доступа для пользователей в TSS на узлах авторизации гарантируется с помощью функции, которая выделяет ресурсы сервера для обработки запросов ввода/вывода сервером авторизации. Эта схема обеспечивает адекватный ответ для файлов доступа пользователей на узлах авторизации, даже когда задания на вычислительных узлах выполняют операции ввода/вывода файлов.

3) Использование пропускной способности ввода/вывода с максимальным результатом

Стали возможными операции с максимальным результатом так, что незадействованные ресурсы сервера могут эффективно использоваться (**Рисунок 6**). Если запросы ввода/вывода создаются одновременно и узлами авторизации и вычислительными узлами, все ресурсы сервера разделяются этими двумя типами узлов (левая часть Рисунка 6), но если никаких запросов ввода/вывода не создается на вычислительных узлах, то узлы авторизации будут использовать все ресурсы сервера (правая часть Рисунка 6).

FEFS управляет большим количеством пользователей, следовательно, файловая активность любого конкретного пользователя не должна иметь возможности влиять на других пользователей, в частности, на пользователей на узлах авторизации. Политика QoS, таким образом, выбирается типом операции, чтобы каждый пользователь получил справедливую долю системных ресурсов и, чтобы каждый пользователь TSS имел гарантированный своевременный отклик.

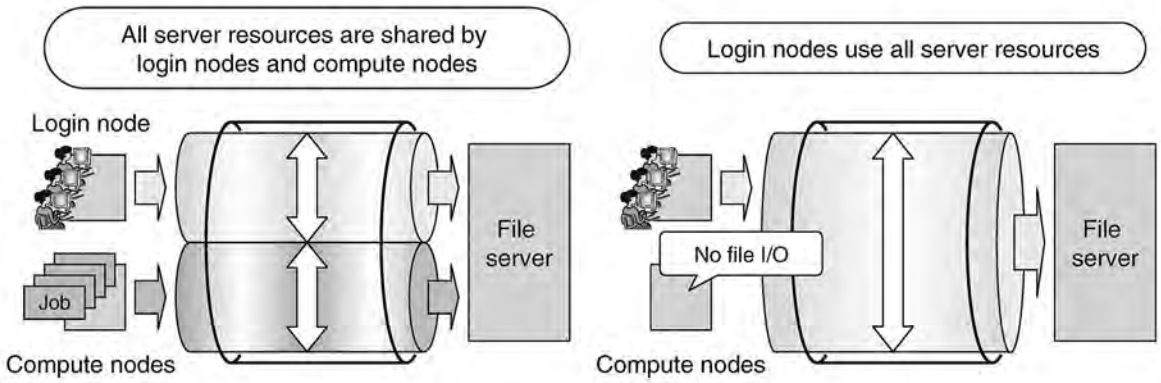

Рисунок 6
Пропускная способность ввода/вывода с максимальным результатом

8 Производительность передачи данных FEFS

К компьютер все еще находится в стадии разработки и FEFS все еще проходит эксплуатационные испытания, так как масштаб системы постепенно увеличивается. Производительность файлового ввода/вывода для двух дисковых стоек из дисковой системы, состоящей из девяти дисковых стоек, показана на **Рисунке 7**. Эти результаты были получены с использованием теста с чередующимися или случайными данными (IOR, interleaved or random), стандартно используемого для измерения ввода/вывода. Они показывают, что FEFS достигает эталонной производительности в 340GB/s при выполнении параллельных операций ввода/вывода на 580 серверах. На настоящий момент это максимально достигнутая производительность ввода/вывода в мире.

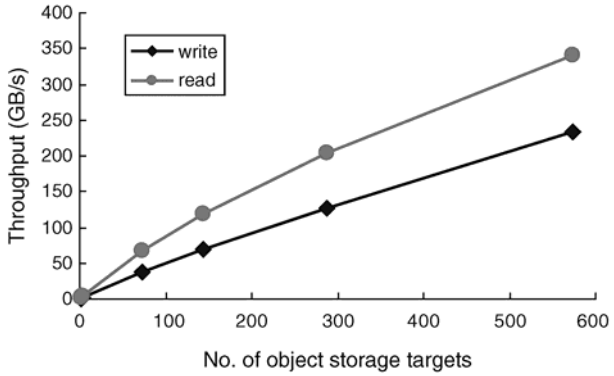

Рисунок 6
Производительность пропускной способности (IOR benchmark)

9. Заключение

Данная статья представляет кластерную распределенную файловую систему, разработанную для K компьютера. Файловая система собирает вместе файловые сервера на уровне нескольких тысяч единиц, достигая систему огромной емкости класса 100-PB и высокой производительности ввода/вывода в классе 1-TB/s. Она обеспечивает эффективный мониторинг узлов и автоматическое переключение узлов и не требует человеческого вмешательства совместно с программным обеспечением для работы K компьютера и управления им. Она обеспечивает доступ к файлам с использованием функций выбора QoS-политик, что смягчает конфликты между пользователями и заданиями, совместно использующими эту крупномасштабную систему.

Заглядывая вперед, Fujitsu намерена улучшить надежность и производительность FEFS в рамках текущих усилий разработки. Компания также стремится к стабильной работе K компьютера. Компания стремится применить наработки файловой системы, полученные на K компьютере к своему коммерческому суперкомпьютеру PRIMEHPC FX10 и кластерным системам PC, а также произвести файловые системы для суперкомпьютеров от обычного уровня до высокопроизводительных. В то же время, компания Fujitsu будет работать, чтобы отобразить свои достижения в файловой системе на K компьютере в будущих версиях Lustre с приглядкой на дальнейшую стандартизацию посредством совместных усилий с сообществом разработчиков Lustre OpenSFS и компанией-разработчиком Lustre, Whamcloud.