IBM SoFS: кластерная файловая система хранения

Публикация – продолжение серии статей, посвященных горизонтально масштабируемым и, в основном, позиционируемым как кластерные, системам хранения. Это второе объявление IBM за последние полгода, связанное с развитием данной архитектуры. Первое состоялось в сентябре 2008 г. в связи с интеграцией решения компании XIV в семейство продуктов IBM. Решение IBM SoFS полностью ориентировано на файловый доступ, в отличие от IBM XIV Storage System, которое в настоящее время поддерживает только блочный доступ.

Алексей Федосеев – Linux Center of Competence, IBM EE/A

Сергей Тараненко – "Тринити", Санкт-Петербург

Введение

В настоящее время рынок файловых СХД можно в первом приближении разделить на 3 группы: 1) файловые/настольные серверы (на базе стандартных – ориентированных, в основном, на SMB и отчасти средний бизнес); 2) корпоративные NAS-серверы (гораздо более дорогие и значительно производительные) и 3) горизонтально масштабируемые кластерные файловые СХД – сектор, который активно стал развиваться в последние несколько лет, но объявления от основных storage-вендоров прошли только за последние 6 месяцев.

Последняя группа по ценовому диапазону за единицу хранения позиционируется между первой и второй группой и имеет гораздо более высокую масштабируемость как по объему (до сотен и более терабайт), так и по доступности (до десятков и сотен гигабит в секунду) в сравнении со второй. В качестве основных строительных блоков выбирались в основном стандартные массово производимые строительные компоненты – стандартные серверы/блэйд-серверы в комплекте с дисками SAS или SATA. При этом число используемых серверов может доходить до десятков и сотен, что позволяет достигать массового параллелизма при организации доступа к данным и практически линейной масштабируемости по производительности. Интегрирующая компонента в таких системах это, в основном, ethernet/infiniband свичи. Соответственно, вся основная функциональность в таких системах привносится программным обеспечением, а не специализированными компонентами/блоками/контроллерами в модульных и монолитных системах хранения (SN № 3/36, 2008 – "Горизонтально масштабируемые СХД").

Для последней группы файловых СХД необходимо отметить еще ряд активно развиваемых особенностей – это повышенные уровни управляемости, самовосстановления и балансировки нагрузки.

Активное развитие последней группы решений обусловлено, прежде всего, потребностями современного рынка, которые уже с трудом вписываются в жесткие архитектурные рамки традиционных сетевых файловых хранилищ. В качестве некоторых причин, способствующих развитию нового класса NAS-систем можно привести следующие:

- *объемы файлового хранения возрастают от года к году и уже составляют 65–80% всех данных, с ростом 50–70% в год (Yankee, 2/07). В 2009 г. объем данных, хранимых в файлах, превысит 10,6 экзабайт (от 1,3 экзабайта в 2006 г., IDC 2/06);*
- *современные решения на основе NAS плохо масштабируются в горизонтальном направлении. В случае исчерпания объема приходится добавлять новый сервер, который управляется индивидуально (добавления слоя виртуализации усложняет архитектуру). Независимость серверов ведет к невозможности единой политики управления;*
- *одновременный доступ к данным из разных приложений становится обыденностью, особенно при обработке мультимедиа;*
- *многие "файловые" приложения по требованиям по производительности доступа к данным стали сопоставимы с HPC-приложениями (high performance computing);*
- *трудности с миграцией, интеграцией и удалением систем хранения для системного администрирования;*
- *необходимость управления циклом жизни информации в десятки и сотни терабайт требуют развитой ILM-функциональности на уровне файловой системы.*

Одной из первых попыток IBM преодолеть ограничения традиционных NAS-систем было решение SAN File System (объявление 2003 г. – см. SN № 4/18, 2003), строившееся на базе технологии сетевой виртуализации (SAN Volume Controller – SVC) и SAN СХД, оно имело глобальное пространство имен и хорошую (но ограниченную) масштабируемость за счет SVC, однако было достаточно дорогим из-за SAN-инфраструктуры.

Аббревиатура SoFS расшифровывается IBM как Scale-out File Services – "неограниченно масштабируемые файловые сервисы" в глобально распределенной среде. В данной статье рассматривается отличие классической концепции сетевых хранилищ данных и инновационных технологий, реализованных в SoFS, – новом решении компании IBM, сочетающем уникальную кластерную файловую систему и масштабируемые службы доступа к данным, включающие Samba/CIFS (доступ из сетей Windows), NFS и другие протоколы, такие как FTP или HTTP. Уникальность решения IBM SoFS в том, что оно сочетает в себе свойства системы хранения для высокопроизводительных вычислений (кла-

стерная организация, масштабируемость и скорость доступа) и классических сетевых систем хранения (простота настройки и использования).

Концепции сетевых систем хранения

Проблемы существующих сетевых систем хранения данных можно продемонстрировать, рассмотрев характерный отраслевой пример хранилищ медиа-контента. В настоящее время объем данных в этой области значительно растет при переходе к цифровому вещанию и телевидению высокой четкости. Большинству заказчиков из этой области требуется читать и записывать данные с производительностью свыше 10 Гбайт/с. При этом доступ к одному файлу на чтение и запись может осуществляться одновременно из множества приложений. Даже самые продвинутые сетевые системы хранения не справляются с такими нагрузками. Их производительность ограничена числом подключенных дисков и устройств хранения данных.

Для обеспечения таких высоких требований к производительности сетевое хранилище должно распределять все файлы между максимально возможным числом дисков, контроллеров и узлов системы хранения, так как скорость ввода-вывода каждого отдельно взятого компонента значительно ниже требуемой. Другое широко распространенное решение заключается в разделении файлов по различным системам хранения в зависимости от их назначения. Но это увеличивает сложность администрирования, добавляет новые пространства имен, тогда как ресурсы в таких системах распределяются не оптимально.

Этот пример показывает, что в области хранения данных существует потребность в новом поколении сетевых систем хранения. Такие системы должны создаваться, с одной стороны, с ориентацией на максимальное масштабирование и производительность и, с другой стороны, без потери простоты использования и управления.

Масштабируемые сетевые хранилища данных

Любая файловая система – это способ организации файлов и данных, обеспечивающий доступ к информации и поиск. Сетевая файловая система, в свою очередь, позволяет осуществить удаленный доступ к файлам. В 1985 г. корпорация Sun Microsystems разработала Network File System (NFS), которая стала первой широко распространенной сетевой файловой системой. Через 7 лет компания NetApp представила первую систему, обладавшую одной простой функцией: она обеспечивала доступ к файлам через специальный сетевой сервер, оптимизированный для выполнения данной задачи. Это решение положило начало рынку сетевых хранилищ данных (NAS).

Традиционные системы сетевого хранения данных обладают рядом недостатков. В первую очередь это ограниче-

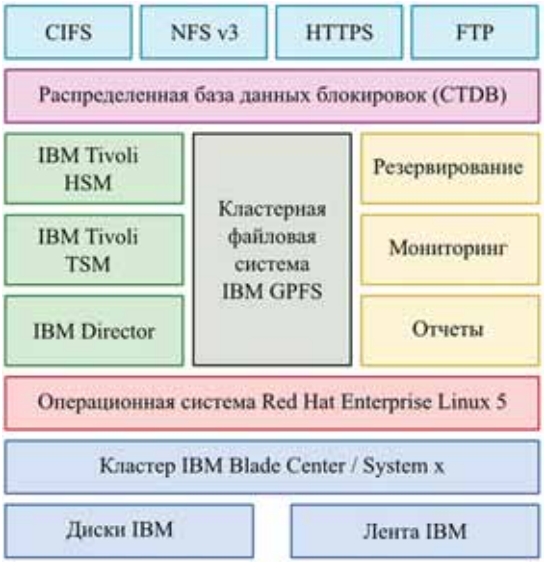

Рис. 1. Компоненты решения IBM Scale out File Services.

ния масштабируемости – при достижении предела производительности или емкости приходится переходить на другую, более мощную систему. Такое вертикальное масштабирование можно продолжать, пока не будет достигнут предел наиболее мощных существующих систем хранения.

Альтернативным подходом является построение кластерных систем для доступа к данным. В этом случае используется программно-аппаратное решение, предоставляющее пользователю доступ к одному виртуальному узлу, за которым скрывается кластерная система хранения. Увеличение размеров такой системы носит характер горизонтального масштабирования и позволяет обойти ограничения обычных систем хранения. Решение IBM SoFS, построенное на базе продуктов с открытым кодом и собственных уникальных технологий, позволяет преодолеть существующие пределы производительности и масштабирования сетевых систем хранения.

SoFS базируется на параллельной файловой системе IBM General Parallel File

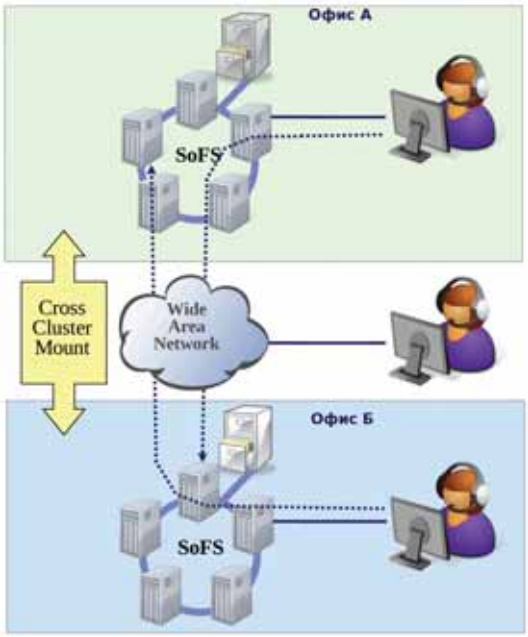

Рис. 2. Конфигурация SoFS со множественными WAN-кластерами.

System (рис. 1), одна из ключевых возможностей которой – распространение блоков одного файла по всем узлам кластера, что позволяет объединить производительность всех подключенных систем хранения и достигнуть скорости передачи данных, измеряемой сотнями гигабайт в секунду. Но возможности GPFS этим не ограничиваются: для управления жизненным циклом данных используется специальная гибкая система политик, существуют средства репликации данных и кросс-монтирования между удаленными локациями. Используя GPFS, можно объединять множество систем хранения разного класса и скорости в единую виртуальную файловую систему. Так, например, для интеграции ленточных систем хранения и прозрачной миграции данных с дисков используются продукты IBM Tivoli Storage Manager (TSM) и Tivoli Hierarchical Storage Manager (HSM).

Единое глобальное пространство имен в системе хранения SoFS достигается за счет технологий виртуализации и перенаправления данных, реализованных в GPFS-кластере. Каждый узел в кластере имеет доступ ко всем блокам данных одновременно и может заменить любой другой узел кластера в случае его неисправности. Таким образом, конечные приложения, обращаясь к какому-то конкретному узлу кластера, работают виртуально со всем массивом данных (рис. 2). Ни одна из существующих на данный момент кластерных технологий не обеспечивает прозрачный непрерывный доступ через сетевую файловую систему в случае неисправности серверного узла, с которым идет обмен. Решение SoFS базируется на чрезвычайно масштабируемой кластерной технологии, которая учитывает семантику используемых сетевых протоколов доступа к данным и обеспечивает минимальное время восстановления работы системы для клиентской стороны при очень большой скорости доступа.

Решение SoFS – это аппаратно-программный комплекс. Кластерными узлами системы могут являться блэйд-серверы IBM или серверы линейки IBM System x., а в качестве систем хранения поддерживаются все продукты из IBM System Storage. Типичная конфигурация кластера SoFS представлена на рис. 3.

IBM General Parallel File System (GPFS)

Сердцем SoFS является кластерная файловая система IBM GPFS, которая предоставляет всем узлам кластера общую файловую систему с единым пространством имен. GPFS базируется на модели разделяемых дисков: все узлы имеют одновременный доступ на чтение и запись к группам блочных устройств (например, RAID-массивам, подключенным через Fibre Channel). GPFS реализует интерфейс файловой системы, совместимый с POSIX, так что файловые серверы в SoFS – Samba и NFS – обращаются к данным на когерентной файловой системе, совер-

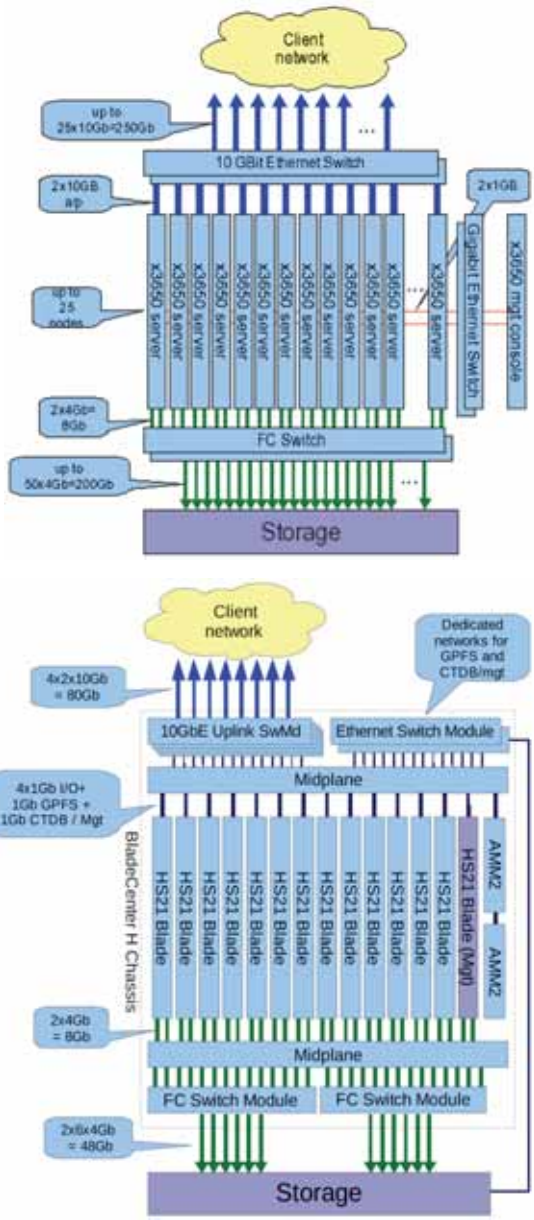

Рис. 3. Типовые аппаратные конфигурации SoFS на базе стандартных серверов x3650 (вверху, 2 GigE Adapter на сервере и до 6 Ethernet или FC Adapter опционно) и на базе шасси BladeCenter H (до 6 Ethernet коммутаторов – 1/10 GbE; два 4 Gb FC-коммутатора; два модуля управления; до 14 лезвий HS-21, из них один задействован для нужд управления кластером).

шенно аналогичной локальной файловой системе. Разделяемые диски при этом содержат как данные, так и метаданные файловой системы.

В отличие от многих других кластерных файловых систем, в GPFS нет выделенного сервера для метаданных. Узлы GPFS- или SoFS-кластера обращаются к метаданным непосредственно, синхронизируя свою работу посредством менеджера блокировок GPFS. В случае SoFS менеджер блокировок GPFS реализует две основные функции: он управляет целостностью кэша, чтобы обновление данных и метаданных могло происходить одновременно на множестве узлов, и он управляет блокировками NFS, чтобы клиенты NFS могли восстанавливать блокировки в случае неисправности одного из узлов кластера.

В GPFS есть ряд возможностей, обеспечивающих высокую надежность и отказоустойчивость. В системах хранения отказоустойчивость обычно реализуется на аппаратном уровне (технология RAID, дублирование контроллеров и т.п.), но в GPFS имеется собственная репликация блоков, которая может использоваться вместо (или в дополнение к) RAID. Для того, чтобы реагировать на отказ узла кластера, GPFS хранит журнал всех изменений метаданных на разделяемом диске. Когда узел кластера выходит из строя, он первым делом ограждается от разделяемых дисков для предотвращения возможности некорректной записи. После этого один из оставшихся узлов обновляет данные согласно журналу, чтобы восстановить все прерванные файловые операции. В GPFS всего несколько разделяемых служб: конфигурации, блокировок, кворума и квоты, все они могут работать на любом из узлов кластера. Все команды по администрированию системы (такие как добавление и удаление узлов, балансировка данных с добавлением новой системы хранения и т.п.) выполняются на лету, без остановки кластера. Длительные операции, такие как повторная балансировка, могут быть запущены повторно. То же касается и обновления программного обеспечения – GPFS позволяет устанавливать обновления без остановки работы кластера.

GPFS разрабатывалась для очень больших файловых систем, одна файловая система может включать в себя до 2000 массивов, каждый из которых может быть, к примеру, группой четности RAID. В 2007 г. самая большая система хранения под управлением GPFS содержала 2 петабайта. GPFS использует хэширование имен файлов в директориях, что повышает производительность поиска даже для большого числа файлов.

CIFS и NFS

Файловый сервер CIFS, являющийся одним из важнейших компонентов SoFS и обеспечивающий доступ к файлам из Windows-машин, базируется на известной разработке с открытым исходным кодом – проекте Samba. Сервер Samba неоднократно пытались модифицировать для работы в кластерной инфраструктуре, но все попытки упирались в низкую производительность и сложность семантики блокировок, существующих в протоколе CIFS. Эти проблемы были решены в SoFS.

Сервер Samba обеспечивает полноценную поддержку протокола CIFS (и, соответственно, операционной системы Microsoft Windows), размещая файлы на любой POSIX-совместимой файловой системе, той же GPFS. Основная задача Samba – преобразование семантики CIFS-протокола (производного от Windows API по работе с файлами) к семантике POSIX-совместимых файловых систем. Для такого корректного преобразования серверу приходится поддерживать несколько баз данных с информацией об открытых файлах, используемых ресурсах, блокировках, пользователях и т.п. Обычная версия Samba, не предназначенная для кластерных систем, использует легковесную базу данных trivial database (TDB) для хранения такой информации. База данных реализуется в виде локального файла, к которому обращаются все процессы файлового сервера, координируя свою работу.

В случае кластерной файловой системы подобной GPFS, такая база данных уже не может быть реализована в виде простого файла, расположенного на разделяемом пространстве. Такой подход не работает, так как скорость работы базы данных (а значит, и всех файловых операций) очень низка: для кластера из двух узлов скорость падает в 10-100 раз по сравнению с одиночным узлом. Очевидно, ни о каком масштабировании не может идти и речи, а значит, кластерная версия Samba должна использовать принципиально другую организацию хранения состояния сервера и обмена метаданными.

Результатом исследований участников проекта Samba и разработчиков IBM стало создание кластерного хранилища для метаданных сервера – cluster trivial database (CTDB). Эта система реализует распределенную базу данных, используя протокол обмена сообщениями между серверами в кластере вместо одного общего файла. Это позволило получить прирост производительности, ограниченный только пропускной способностью коммуникационного оборудования.

Протокол NFS, в отличие от CIFS, намного менее требовательный к сохранению состояния сервера, так что значительно лучше подходит для кластерной реализации. Семантика блокировок файлов в NFS также значительно отличается от CIFS. Важной особенностью SoFS является то, что CTDB используется для реализации межпротокольных блокировок. Таким образом, блокировки файлов на уровне различных протоколов и на уровне операционной системы находятся в соответствии, что гарантирует корректное состояние файлов при одновременном доступе по нескольким протоколам (CIFS, NFS, FTP).

Интерфейс администрирования

На рис. 1 были представлены основные компоненты решения SoFS: кластерные узлы, распределенная файловая система

Табл. 1. Пределы возможного масштабирования SoFS в сравнении с IBM System Storage N и Windows 2k3 Server

	SoFS	IBM System Storage N	Windows 2k3 Server
Number of nodes per cluster	13 (limit will be lifted with upcoming releases) Multiple clusters can export the same file system	2	n/a
Number of CPUs per cluster	52 (limit will be lifted with upcoming releases)	16	n/a
Max. capacity	33554432 Yobibytes (2^{105} Bytes)	504 Terabyte (~2^{49} Bytes)	n/a
Max. size of single file system	524288 Yobibytes (2^{99} Bytes)	16 Tebibyte (2^{44} Bytes)	256 Tebibyte (2^{48} Bytes)
Max. number of file systems	256	200	n/a
Max. size of single file	16 Exibytes (2^{64} Bytes)	16 Tebibyte (2^{44} Bytes)	16 Tebibyte (2^{44} Bytes)
Max. number of files per file system	2 billion (2^{31})	??	4 billion (2^{32})
Max. number of snapshots per fs	31	256	n/a

GPFS, кластерные версии файловых серверов, реализующих сетевые протоколы (CIFS, NFS и др.), коммуникационное оборудование. При желании такие системы можно было бы собрать самостоятельно, примеры таких высокопроизводительных систем можно увидеть в рейтинге суперкомпьютеров Top-500. Однако администрирование таких уникальных систем – отдельная сложная задача, требующая высококвалифицированных специалистов. Применение таких масштабируемых систем хранения в других областях, таких как связь, цифровое телевидение, фармацевтика или же для организации огромных хранилищ уровня предприятия подразумевает совсем другие требования к администрированию. Похожие проблемы возникают при использовании классических сетевых систем хранения: при росте объема данных приходится масштабировать хранилища путем добавления новых систем хранения и логического распределения данных между ними (например, по отделам или задачам), что в свою очередь повышает расходы на поддержку и администрирование таких систем.

Управление такой инфраструктурой должно осуществляться через единый интерфейс, аналогичный используемому в сетевых системах хранения или маршрутизаторах. Интерфейс администрирования SoFS разрабатывался из таких соображений и потому не уступает классическим сетевым систем хранения. Решения SoFS отличается возможностью масштабирования путем увеличения числа компонентов для получения нужного объема, производительности и других параметров. Пределы возможного масштабирования SoFS в сравнении с IBM System Storage N и Windows 2k3 Server даны в табл. 1.

Тестирование системы в реальных условиях

Специалистами компаний Тринити и IBM было проведено тестирование решения SoFS. Использовалась минимальная конфигурация SoFS – три кластерных узла, один из которых служит для администрирования системы. Пользовательская сеть из парка машин под управлением Windows XP была подключена через локальную сеть в 1 Гбит/с. Использовалась небольшая система хранения с десятью дисками SATA 15K RPM. Локальная сеть управляется через Active Directory на базе Windows 2003 Server, что позволило использовать клиентов как под управлением Windows, так и под Linux.

После установки и настройки кластера SoFS было произведено нагрузочное тестирование. На каждой из четырех клиентских машин был использован комплексный тест производительности файловой системы NBENCH из набора тестовых утилит сервера Samba. Результаты тестирования чтение блоками по 64 Кбайт из файлов размером в 1 Гбайт показаны ниже:

- клиент 1 – 38,7363 Мбайт/с;
- клиент 2 – 38,8454 Мбайт/с;
- клиент 3 – 100,868 Мбайт/с;
- клиент 4 – 40,1211 Мбайт/с.

Поскольку тест был запущен на всех клиентских машинах вручную, между запусками были временные разрывы. В данном случае один из клиентов успел “отъесть” больше полосы пропускания, чем другие. Тест воспроизводит типичную картину “голода” в офисе, когда возможностей файлового сервера не хватает на всех клиентов из-за ограничений полосы пропускания (в нашем случае 1 Гбит/с для каждого из клиентов). В этом случае важным становится эффективное распределение нагрузки между узлами кластера.

Видно, что один из клиентов стартовал раньше и захватил всю доступную себе полосу пропускания – из 1 Гбит/с ему было доступно 100,868 Мбайт/с, или 80,69% канальной емкости. Остальные запустились более-менее одновременно и потому распределили между собой оставшийся объем канала. Суммарно вышло 218,5708 Мбайт/с из доступных 2 Гбит/с (87.42% канальной емкости), которые SoFS мог отдать в связи с ограничением канала.

Вместо заключения

Итак, что было достигнуто в результате реализации решения SoFS:

- *добавлена поддержка NTFS ACL в Samba на основе системы разграничений в GPFS;*
- *обеспечена простая установка и настройка системы;*
- *усовершенствована Samba для более гибкой балансировки нагрузки;*
- *обеспечен множественный доступ к единой файловой системе с множества узлов с сохранением семантики, распределенных блокировок и разделения доступа к файлам;*
- *добавлена поддержка иерархического хранения данных в Samba для того, чтобы многоуровневый механизм хранения данных был прозрачен для пользователей;*
- *реализовано прозрачное восстановление после сбоев без изменения клиентских приложений.*

В результате SoFS позволила объединить кластерную файловую систему и глобальное адресное пространство при полном централизованном внедрении, управлении, архивировании данных и масштабировании. Достигнута полная горизонтальная масштабируемость: если требуется больше свободного места, просто добавляются диски; если требуется большая производительность, добавляются узлы и/или диски. SoFS обеспечивает встроенную поддержку управлением жизненным циклом информации:

- разные уровни хранения: FC, SATA, ленточные накопители;
- политики для управления размещением и миграцией данных в процессе всего срока жизни информации.

Обеспечена синхронная/асинхронная репликация данных и метаинформации на блочном уровне.

В качестве возможных применений SoFS можно назвать следующие:

- архивирование текстовых, аудио- и видеоданных;
- кэширование информации для веб-серверов и серверов приложений;
- командная работа географически распределенных коллективов;
- консолидация систем хранения;
- высокоскоростная инфраструктура восстановления данных;
- поддержка анализа и визуализации огромных массивов данных для научных и экономических задач;
- различные формы информационного слежения, например, почтовое или видеослежение;
- поддержка инфраструктуры мультимедиа компаний.

Алексей Федосеев,
IBM, alexey_fedoseev@ru.ibm.com,
Сергей Тараненко,
“Тринити”, Санкт-Петербург