ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
Федеральное государственное образовательное бюджетное учреждение высшего профессионального образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ им. проф. М. А. БОНЧ-БРУЕВИЧА»

А.Н.Губин

Сети хранения данных

Краткий конспект лекций

Раздел 6. Сети хранения данных

САНКТ-ПЕТЕРБУРГ
2017

Содержание

Раздел 6. Сети хранения данных

Сеть хранения данных (SAN - Storage Area Network) это высокоскоростная сеть, которая связывает компьютерные системы (хост-серверы) с высокопроизводительными подсистемами хранения информации (хранилищами данных).

SAN состоят из адаптеров шины, специализированных устройств для поддержки маршрутизации трафика (концентраторы и коммутаторы) и дисковых массивов хранения данных.

Технологии SAN выполняют операции чтения/записи информации на уровне блоков данных.

Чаще всего для реализации SAN используется инфраструктура оптоволоконных каналов Fibre Channel.

6.1. Архитектура сетей хранения данных. Особенности архитектуры СХД Fiber- Channel. Зонирование данных

Fibre Channel - это открытый промышленный стандарт высокоскоростного последовательного интерфейса. Он обеспечивает подключение серверов и сторедж-систем на расстоянии до 10 км (при использовании стандартного оснащения) на скорости 100 MB/s, 200 MB/s, 400 MB/s.

Изначально технология Fibre Channel предполагала поддержку только волоконно-оптических линий (fiber optic). Однако когда добавилась поддержка меди, было принято решение название в принципе сохранить, но для отсылки на стандарт использовать британское слово Fibre.

В технологии Fibre Channel предпринята попытка объединить лучшее из двух базовых разделов техники связи — каналов передачи данных и сетей. Термин канал впервые стал использоваться в мире мэйнфреймов и описывал структурированный механизм передачи данных. В большинстве случаев передача данных выполняется между компьютерной системой и периферийным устройством, например жестким диском или накопителем на магнитной ленте. К таковым каналам относятся интерфейсы SCSI (Small Computer System Interface) и HIPPI (High-Performance Parallel Interface). Работа каналов обычно реализуются средствами аппаратного обеспечения.

По сравнению с каналом сеть представляет собой более универсальный механизм для передачи данных, который, однако, менее структурирован. Кроме того, сеть может работать на значительно большем расстоянии и подключаться к большему количеству устройств, чем канал. В отличие от каналов, сети в основном реализуются средствами программного, а не аппаратного уровня.

Один из подходов в объединении систем хранения данных и сетей заключается в том, что сеть становится ключевым элементом, к которому добавляются новые возможности с одновременной компенсацией недостатков

подобного подхода. Речь идет о технологии хранения данных на базе протокола IP.
Другой подход состоит в использовании центрального хранилища данных (канальная система) и расширения существующих технологических функций. На базе этого метода создавалась технология Fibre Channel.

Одним из важнейших преимуществ Fibre Channel (FC) наряду со скоростными параметрами (которые, кстати, не всегда являются главными для пользователей SAN и могут быть реализованы с помощью других технологий) является возможность работы на больших расстояниях и гибкость топологии, которая пришла в новый стандарт из сетевых технологий. Таким образом, концепция построения топологии сети хранения данных базируется на тех же принципах, что и традиционные сети, как правило, на основе концентраторов и коммутаторов, которые помогают предотвратить падение скорости при возрастании количества узлов и создают возможности удобной организации систем без единой точки отказов.

Коммуникационный протокол FC (рис. 6.1) представляет собой объединение пяти уровней реализации функций обработки данных: от FC-0 до FC-4 (за исключением уровня FC-3, который не реализуется).

Рис. 6.1. Пакет протоколов Fibre Chanel

FC-0 определяет физический интерфейс и среду передачи данных. Спецификация FC-0 включает в себя кабели, разъемы, оптические и электрические параметры среды передачи данных.

FC-1 определяет протокол передачи данных, который включает в себя правила последовательного кодирования и декодирования информации, использования специальных символов и управление ошибками. На передающем узле 8-разрядный символ кодируется в 10-разрядный передаваемый блок и передается на приемный узел. На приемном узле 10-

разрядный блок передается на уровень FC-1, который декодирует 10-разрядный блок в оригинальный 8-разрядный символ.

FC-2 представляет собой транспортный уровень и обрабатывает данные отражающие адреса портов источника и приемника, а также информацию по управлению каналом связи и передаваемую информацию. Данный уровень обеспечивает работу с адресами FC, кроме того, здесь производится структуризация данных (кадры, последовательности, обмены), управление потоками и маршрутизация.

FC-4 определяет интерфейсы приложений и способы, которыми протоколы верхнего уровня отображаются на более низких уровнях FC. Стандарт FC определяет несколько протоколов, которые могут работать на уровне FC-4. К этим протоколам относятся:

- SCSI (Small Computer Systems Interface) - протокол для физического подключения и передачи данных между компьютерами и периферийными устройствами. SCSI стандарты определяют команды, протоколы и электрические и оптические интерфейсы;
- HIPPI (High-Performance Peripheral Interface) – обеспечивает высокоскоростной обмен между самыми мощными компьютерами (например, суперкомпьютеры);
- ESCON (Enterprise Systems Connection) – обеспечивает связь систем в масштабах предприятия;
- ATM (Asynchronous Transfer Mode) - высокопроизводительная технология коммутации и мультиплексирования, основанная на передаче данных в виде ячеек (cell) фиксированного размера;
- IP (Internet Protocol) – стек протоколов, который объединяет сегменты сети в единую сеть, обеспечивая доставку пакетов данных между любыми узлами сети через произвольное число промежуточных узлов (маршрутизаторов).

FC-архитектура

FC-архитектура поддерживает три основных топологии использования основных компонентов SAN: "точка-точка" (FC-PTP - Fibre Channel point-to-point), "управляемая петля" (FC-AL – Fibre Channel Arbitrated Loop) и "коммутируемое соединение" (FC-SW - Fibre Channel Switch).

Архитектура "точка-точка" предусматривает непосредственное подключение двух устройств друг к другу – передатчик одного устройства соединяется с приемником другого (рис. 6.2). Все отправленные одним устройством кадры предназначены для второго устройства.

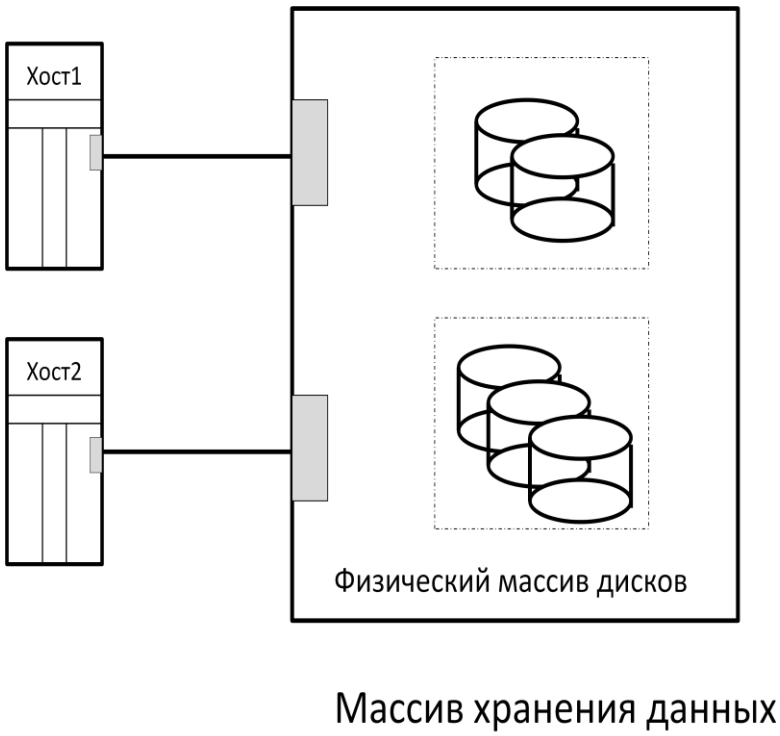


Рис. 6.2. Топология "точка-точка"

Эта конфигурация обеспечивает выделенное соединение для обмена данными между узлами, однако обладает ограниченными возможностями масштабирования системы.

Архитектура "управляемая петля" предусматривает объединение устройств в петлю — передатчик каждого устройства соединяется с приемником следующего устройства (рис. 6.3). Перед тем, как петля будет выполнять функции передачи данных, устройства составляющие петлю осуществляют "арбитраж"- договариваются о праве контроля над петлей. Для передачи данных по петле устройство должно завладеть «эстафетой» (token). В любой момент времени только одно устройство может выполнять операции ввода/вывода по петле. Для построения управляемой петли используют концентраторы, которые способны размыкать или замыкать петлю при добавлении нового устройства или выходе устройства из петли.

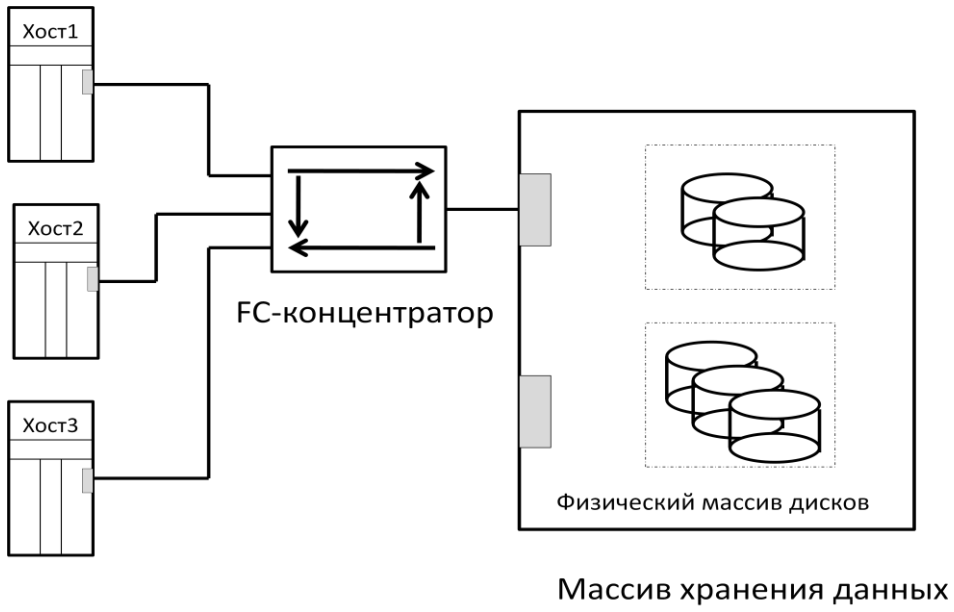


Рис. 6.3. Топология "управляемая петля" FC

Конфигурация FC-AL использует 8-разрядную адресацию и может поддерживать до 127 устройств в петле.

Добавление или удаление устройства в структуру петли приводит к повторной инициализации петли, что может вызывать кратковременные паузы в передаче данных по петле.

Архитектура "коммутируемое соединение" (коммутируемая сеть, коммутируемая фабрика) базируется на применении коммутаторов (рис. 6.4). Такая структура позволяет подключать большее количество устройств, чем в управляемой петле, при этом добавление новых устройств не влияет на передачу данных между уже подключёнными устройствами. Так как на основе коммутаторов можно строить сложные сети, на коммутаторах поддерживаются распределённые службы управления сетью (fabric services), отвечающие за маршруты передачи данных, регистрацию в сети и присвоение сетевых адресов и проч. Fibre Channel изначально разрабатывался как высокоскоростная сеть, пригодная для работы в реальном времени. В транспорте Fibre Channel заложены механизмы регулирования потока (flow control), синхронизации портов по времени и возможность повтора сбойной информации без обращения к протоколу верхнего уровня. В структурах Fibre Channel при подключении порта обязательным является выполнение операции login, так что коммутатор всегда знает о всех портах сети - какой порт где находится и какие функции может выполнять. Когда в коммутатор Fibre Channel приходит кадр данных, то коммутатор уже знает, где находится адресат и куда этот кадр маршрутизировать (в отличие от Ethernet, в котором коммутатор после прихода кадра сначала ищет, где находится адресат и только после его ответа посылает ему этот кадр, и, если истекло время старения, коммутатор Ethernet вновь будет искать маршрут для другого кадра данных от того же источника к тому же адресату, хотя оба порта были online).

Очевидно, что подход Fibre Channel требует больше ресурсов, поэтому коммутаторы по этой технологии значительно дороже, чем для Ethernet.

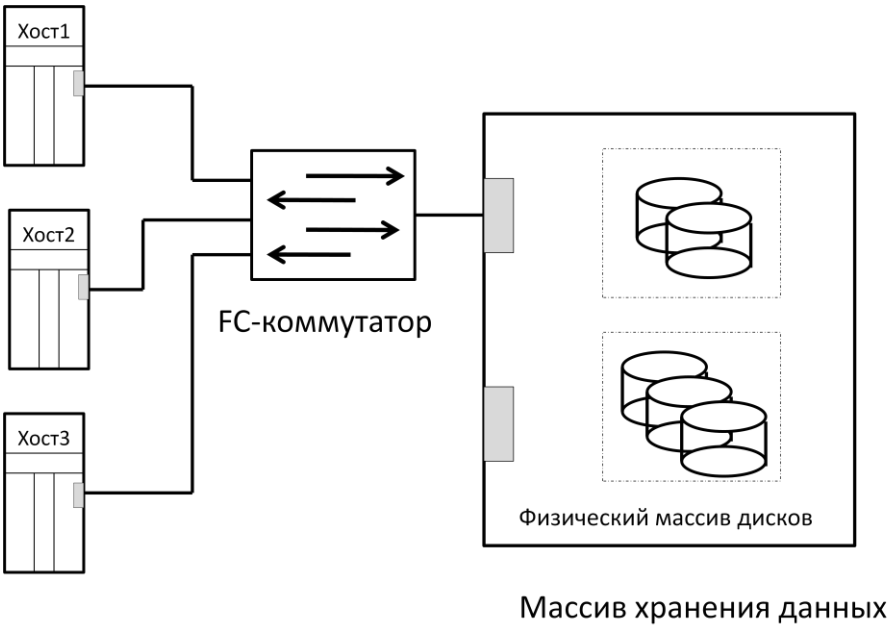


Рис. 6.4. Топология "коммутируемая фабрика" FC

Каждый порт структуры имеет уникальный 24-разрядный FC-адрес, обеспечивающий возможность выполнения операций коммутации между портами.

Развитие архитектуры коммутируемых фабрик FC привело к появлению двухуровневых и трехуровневых структур FC. Количество уровней в структуре FC определяется количеством коммутаторов между двумя, расположенными на наибольшем расстоянии друг от друга, узлами. На рис. 6.5 показаны двух- и трехуровневые архитектуры FC.

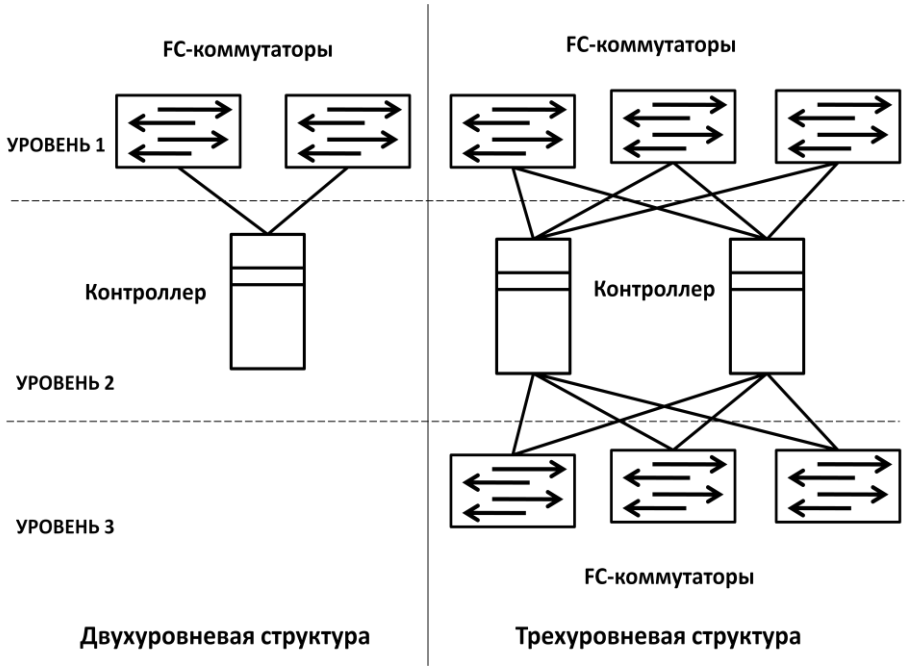


Рис. 6.5. Многоуровневая топология FC-SW

Порты SAN

Базовыми структурными элементами топологии FC являются порты, которые в составе SAN могут быть следующими.

Порты узлов:

- N_port (Node port) – порт устройства с поддержкой топологии FC-P2P («Точка-Точка») или FC-SW (с коммутатором). Конечная точка в топологии FC. Этот порт является узловым, обеспечивает подключение порта главной шины хоста или порта массива хранения данных к FC;
- NL_port (Node Loop port) – узловой порт с поддержкой петли. Обеспечивает подключение к FC-концентратору хостов и хранилищ данных.

Порты топологии FC-SW:

- E_Port (Expansion port), порт расширения. Используется для соединения FC-коммутаторов. Может быть соединён только с портом типа E_Port. Когда E_port одного коммутатора соединяется с E_port другого коммутатора, то между коммутаторами организуется межкоммутаторный канал (ISL- interswitch links). ISL является одним из основных механизмов, обеспечивающих масштабирование СХД;
- F_port (Fabric port) - порт «фабрики» (switched fabric — коммутируемая связная архитектура). Используется для подключения портов типа N_Port к коммутатору. Не поддерживает топологию петли – порт коммутатора, обеспечивающий связь с узловым N_port;

- FL_port (Fabric Loop port) – порт FC-коммутатора, обеспечивающий соединение коммутатора с NL_port FC-концентратора. При наличии такого соединения все NL_ports в структуре FC-AL могут участвовать в коммуникациях по всей сети FC-SW. Такая конфигурация называется открытой петлей. Конфигурация управляемой петли без коммутатора получила название частной петли. Частная петля содержит узлы с портами NL_ports и не имеет портов типа FL_ports;
- EX_port - порт для соединения FC-маршрутизатора и FC-коммутатора. Со стороны коммутатора он выглядит как обычный E_port, а со стороны маршрутизатора это EX_port.
- TE_port (Trunking Expansion port (E_port) - внесен в Fibre Channel компанией CISCO, сейчас принят как стандарт. Это расширенный ISL или EISL. TE_port предоставляет помимо стандартных возможностей E_port маршрутизацию множественных VSANs (Virtual SANs). Это реализовано применением нестандартного кадра Fibre Channel (vsan тегирование).

Универсальные порты:

- G_port (Generic port) – универсальный порт, который может работать как E_port, N_Port или NL_Port. Тип порта определяется автоматически во время инициализации устройства;
- L_Port (Loop port), любой порт устройства с поддержкой топологии «Петля» — NL_port или FL_port.

На рис. 6.6 показано использование основных FC-портов при построении СХД.

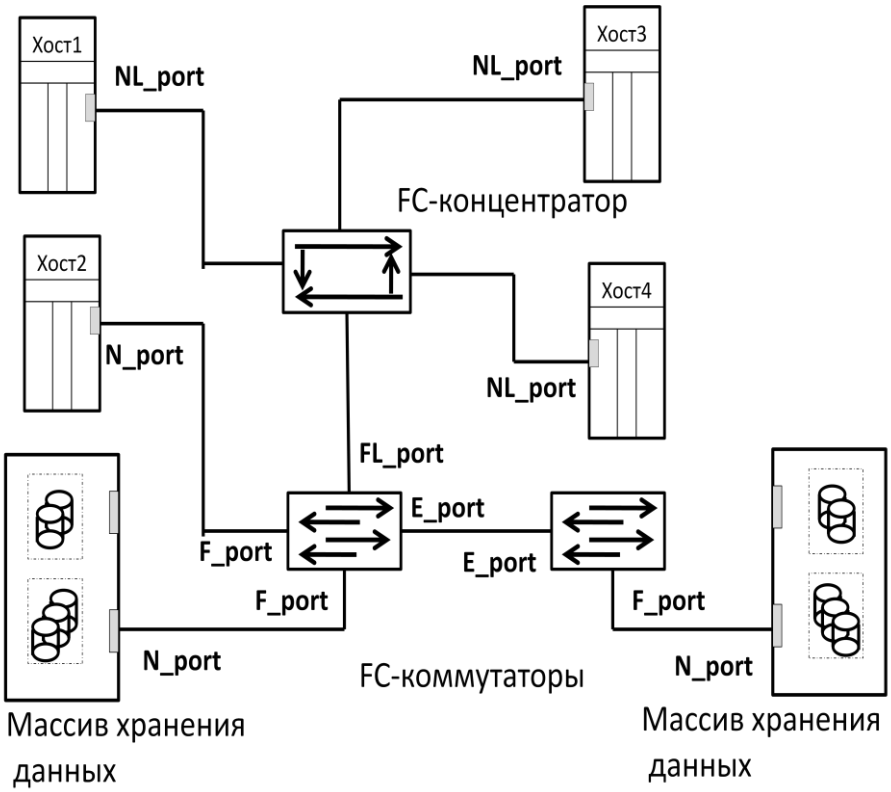


Рис. 6.6. Типы портов Fibre Channel

Адресация Fibre Channel

FC-адрес порта назначается динамически при подключении порта к FC фабрике. Формат FC-адреса определяется типом порта подключаемого узла СХД. Это могут быть порты типа N_port и NL_port в открытой петле или NL_port в частной петле.

Формат N_port содержит три восмиразрядных поля (рис.6.7).

Первое поле определяет идентификатор домена FC-коммутатора. Из 256 возможных значений идентификаторов доменов могут быть использованы только 239 значений, остальные 17 адресов зарезервированы под специальные сервисы FC структур. Так адрес FFFFC зарезервирован для сервера имен, а адрес FFFFFE – для сервиса входа в FC фабрику.

Второе поле – идентификатор области, используется для идентификации группы портов. Примером группы портов является плата коммутатора с несколькими портами.

Третье поле определяет адрес порта в пределах группы портов.

Максимальное число адресов портов типа N_port вычисляется как произведение

239 (доменов) • 256 (областей) • 256 (портов) = 15 663 104.

Рис. 6.7. 24-разрядный FC-адрес порта типа N_port

Формат адресов портов типа NL_port зависит от структуры СХД. Для частной петли два верхних байта не используются (в них заданы нулевые значения), младший байт используется для идентификации физического адреса управляемой петли (рис. 6.8).

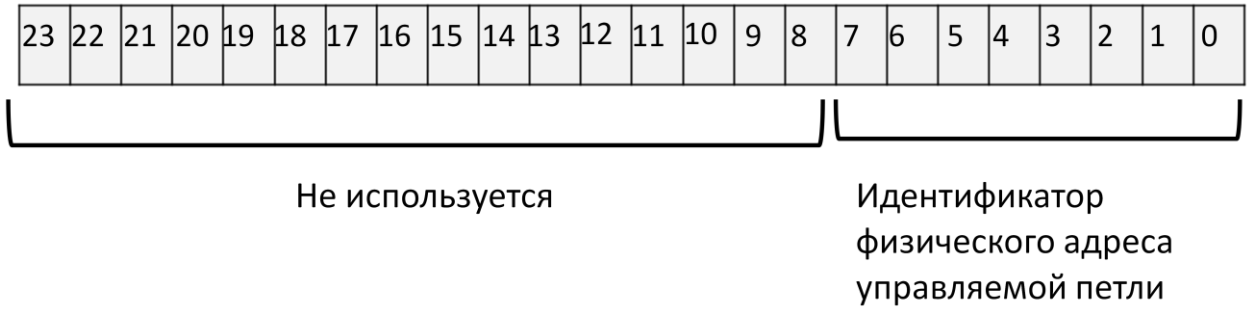


Рис. 6.8. 24-разрядный FC-адрес порта типа NL_port в частной петле

Если управляемая петля подключается к фабрике коммутаций через FL_port, то она становится открытой. В этом случае NL_port обеспечивает вход в фабрику и два верхних байта адреса используются для идентификации петли. Идентификатор петли одинаков для всех NL_port в данной петле (рис. 6.9).

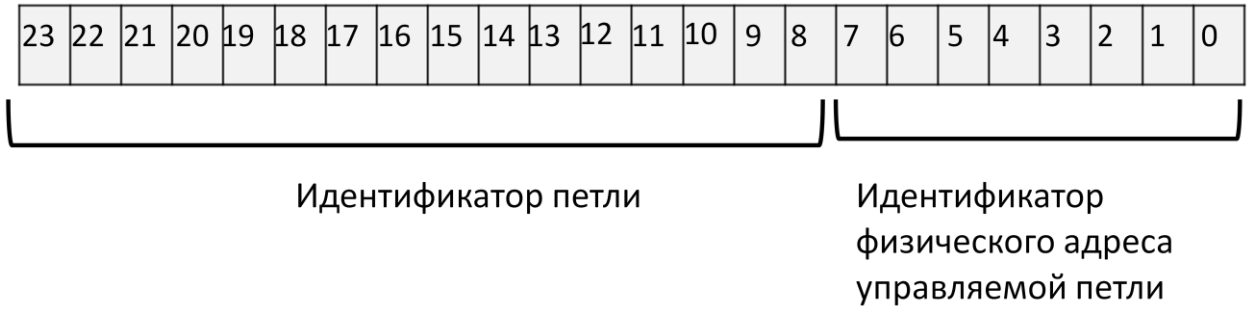


Рис. 6.9. 24-разрядный FC-адрес порта типа NL_port в открытой петле

Имена в глобальной сети

Каждому устройству в среде FC назначается уникальный 64-разрядный идентификатор, который получил название имя в глобальной сети (WWN- World Wide Name). Используется два типа имен: имя узла в глобальной сети (WWNN- World Wide Node Name) и имя порта в глобальной сети (WWPN- World Wide Port Name). В отличие от FC-адреса, который назначается динамически, имя устройства в глобальной сети является статическим и уникальным для каждого устройства. Имена назначаются производителям комитетом IEEE и встраиваются в устройство на этапе изготовления. Имена в глобальной сети аналогичны MAC-адресам (Media Access Control) в IP сетях.

На рис. 6.10 показана структура имени в глобальной сети для устройства хранения данных и для адаптера главной шины

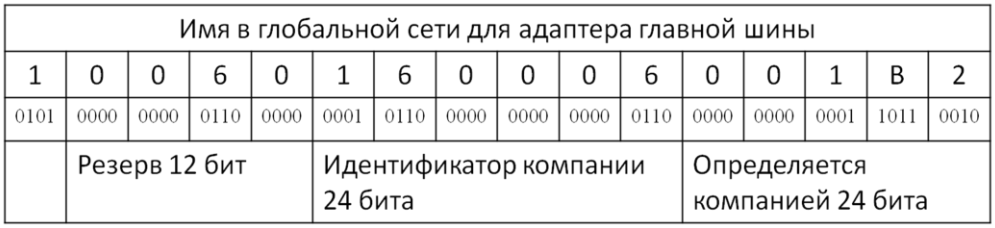


Рис. 6.10. Формат представления имен в глобальной сети

Управление в среде Fibre Channel

Как правило, структуры FC содержат несколько хостов (Initiator) и несколько единиц оборудования хранения данных (Target), в этих условиях возникает необходимость ограничить влияние некоторых хостов на устройства и подсистемы хранения данных. Ограничение доступа может быть реализовано с использованием механизмов маскирования LUN (Logical Unit Number) и зонирования.

Маскирование LUN

В FC структурах используется следующая система адресации устройств: шина (Bus) – адрес (ID) – подадрес (LUN).

Понятие LUN введено для обеспечения доступа к отдельным устройствам СХД в том случае, когда к одному адресу подключается много устройств. Например, внешние дисковые устройства подключаются к серверу одним кабелем к одному порту. Чтобы различать по этому адресу отдельные устройства и вводится понятие подадреса (LUN). В частности LUN может представлять собой не только отдельные логические диски, но и участки RAID-массивов, которые контроллер представляет операционной среде в качестве отдельного физического диска. Стандарт SCSI-2 поддерживает до 64 LUN на один порт.

С функциональной точки зрения маскирование LUN позволяет определенному хосту получать доступ только к конкретному под устройству хранилища данных, которое обладает определенным значением LUN. И, наоборот, с помощью механизма маскирования LUN можно запретить доступ к определенным LUN для определенных компьютеров и серверов.

Маскирование LUN используется как один из механизмов поддержки целостности данных в среде SAN.

Существует несколько способов обеспечения маскировки LUN. Обычно, маскировка выполняется следующими средствами:

- аппаратного обеспечения адаптера шины;

- аппаратного обеспечения коммутатора Fibre Channel;

- аппаратного обеспечения устройства хранения Fibre Channel;

- программного обеспечения узла.

Маскирование LUN средствами BIOS адаптера шины

В BIOS адаптера шины осуществляется маскировка всех LUN, которые не отображены в таблице BIOS адаптера шины. Таким образом, узел (с настроенным BIOS адаптера шины) не «замечает» существования LUN, значения которых не установлены в таблице BIOS.

К недостаткам такого метода следует отнести необходимость проведения корректной настройки адаптера. Кроме того все устройства, адаптеры шины которых настроены неправильно или не поддерживают описываемую функцию, могут получить доступ к тем LUN, к которым доступ на самом деле нежелателен. Еще одна проблема заключается в сложности динамического управления и перенастройки подобных систем

Маскирование LUN коммутаторами Fibre Channel

Коммутаторами Fibre Channel зонирование проводится достаточно просто. Входящий пакет передается или не передается дальше, что зависит от адресов исходного порта и порта назначения. Маскирование LUN возлагает дополнительную нагрузку на коммутаторы Fibre Channel, поскольку коммутатору приходится проверять первые 64 байта каждого пакета данных. Это приводит к снижению производительности большинства коммутаторов Fibre Channel, поэтому этот способ обычно не используется.

Маскирование LUN контроллерами подсистем хранения данных Fibre Channel

Этот метод маскирования LUN является принудительным для подключенных узлов или требует от узла минимального участия. Маскирование LUN реализуется контроллером подсистемы хранения данных или маршрутизатором (с помощью соответствующей прошивки). Эти устройства настроены на поддержку таблицы имен устройств (WWN) адаптера шины, отображённых на номера LUN, к которым им (контроллеру или маршрутизатору) разрешен доступ. Значительное преимущество такого подхода заключается в формировании конфигурации, независимой от промежуточных коммутаторов или концентраторов.

Недостаток метода заключается в закрытой реализации этой технологии каждым поставщиком и сложности создания единой консоли управления для перенастройки или даже получения информации о текущих параметрах, хотя каждый поставщик предоставляет интерфейсы для управления связками WWN-LUN.

Маскирование LUN программным обеспечением узла

Маскирование LUN выполняется программным обеспечением узла, в частности кодом драйвера устройства. Код должен работать в режиме ядра, так как основная идея заключается в том, чтобы предотвратить доступ операционной системы устройства к LUN.

Такая маскировка может выполняться в виде функции операционной системы или драйвера адаптеров шины.

Основная проблема такого метода – необязательная настройка, а, следовательно, необходимость частичного участия узла в процессе маскировки LUN. Это означает, что компьютеры, не имеющие модифицированного драйвера адаптера шины, не принимают участия в маскировке LUN. Кроме того, присутствуют и проблемы масштабирования, так как в особенно больших сетях хранения данных сложно настроить каждый сервер и каждый адаптер шины сервера.

Зонирование (Zoning)

Основной задачей зонирования является разграничение потоков ввода/вывода информации в сетях хранения данных.

Зонирование можно воспринимать в качестве аналога настройки виртуальных локальных сетей (VLAN) в IP сетях. В виртуальной локальной сети только устройства, входящие в состав одной и той же VLAN "видят" друг друга. Устройства, находящиеся в разных VLAN друг друга не "видят", хотя находятся в той же физической локальной сети.

Точно так же зонирование ограничивает возможности компонентов SAN (особенно инициаторов), предоставляя доступ к определенным (входящим в одну и ту же зону) единицам хранения информации, даже если в этой же физической сети хранения данных размещены и другие устройства хранения данных.

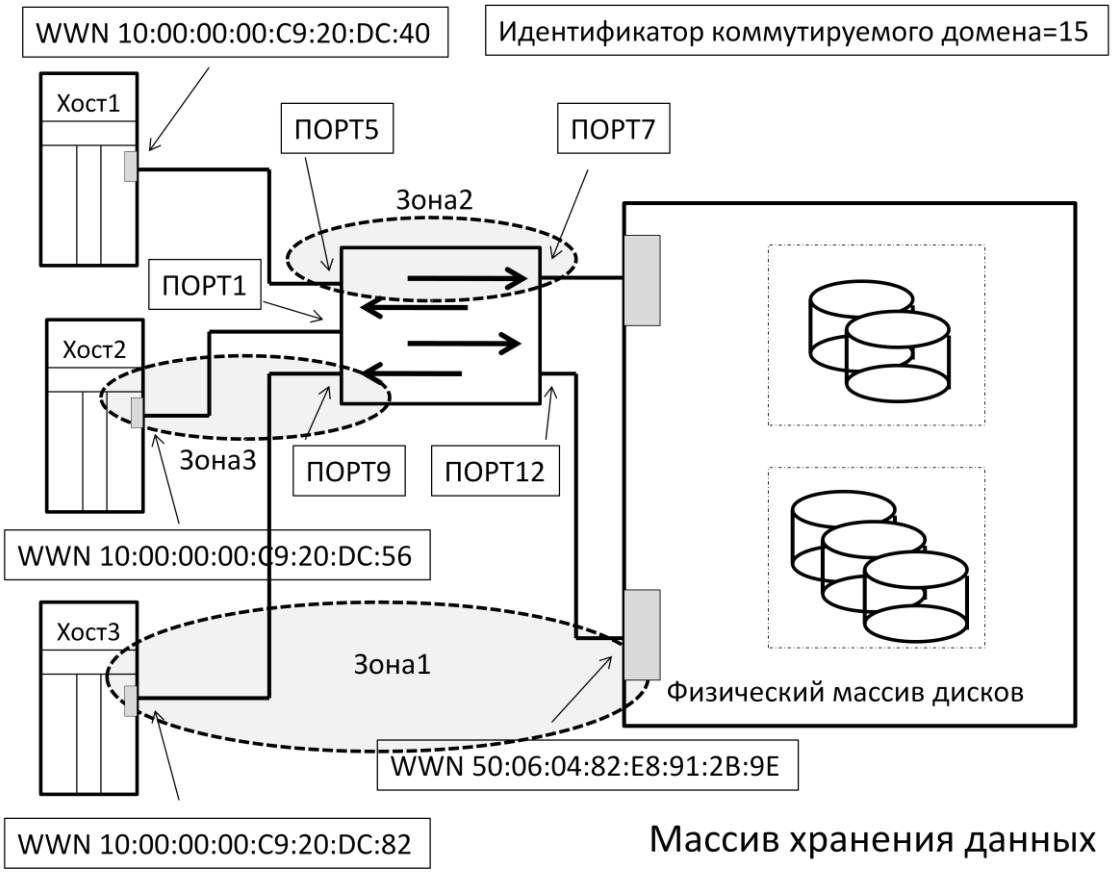
Зонирование является ресурсом всей фабрики и конфигурации зон автоматически передаются на все ее коммутаторы.

Выделяют три основных вида зонирования.

1. Зонирование по портам. Для определения зон используются FC-адреса физических портов фабрики. Устройства, включенные в порты, отнесенные к одной зоне, видят друг друга. Доступ к устройствам из других зон невозможен. Такое зонирование называется жестким. Этот метод обеспечивает достаточно высокий уровень безопасности, однако требует перенастройки зонирования в случаях, когда меняется структура фабрики

(например, добавляются новые порты). Это связано с тем, что FC-адреса назначаются динамически при регистрации порта в фабрике. Поэтому любое изменение в конфигурации фабрики затрагивает зонирование.

2. Зонирование по имени в глобальной сети. Для формирования зон используются имена устройств в глобальной сети (WWN). Устройства с WWN, отнесенными к одной зоне, видят друг друга вне зависимости от того, в какой порт они подключены. Доступ из других зон возможен, если известен WWN устройства внутри зоны. Основным преимуществом зонирования данного типа является его гибкость. Возможно переподключение устройств СХД без реконфигурирования информации по зонированию. Это объясняется тем, что WWN устройств являются статичными и определяются заводами изготовителями. Такой метод зонирования называется мягким.



Зона 1 (Зона по имени в глобальной сети) = 10:00:00:00:C9:20:DC:82; 50:06:04:82:E8:91:2B:9E
Зона 2 (Зона по портам) = 15,5; 15,7
Зона 3 (Зона смешанного типа) = 10:00:00:00:C9:20:DC:56; 15,9

Рис. 6.11. Основные типы зонирования

3. Зонирование смешанного типа. Объединяет способы зонирования предыдущих методов. Позволяет привязать определенный порт FC-структуры к имени узла в глобальной сети.

На рис. 6.11 показаны три типа зонирования в сети FC.

Зонирование используется для управления доступом серверов к хранилищу данных. Часто зонирование используется совместно с маскированием LUN. Следует подчеркнуть, что это два разных типа реализации контроля доступа – зонирование реализуется на уровне фабрики, а маскирование – на уровне массива хранения данных.

6.2. Контекстная адресация данных

Content-addressable storage (CAS) — архитектура хранения, в которой адресация осуществляется образом хранимых данных. Образ данных хэшируется и хэш используется для поиска образа на устройствах или системах хранения данных.

Архитектура обладает большой устойчивостью к дубликатам, а также может быть выполнена децентрализованно, что даёт ей существенную надёжность.

В отличие от традиционных дисковых систем (файловая, блоковая адресация), размещение информации производится не по имени файла или конкретному сектору на поверхности диска, а по его содержимому. Для каждого объекта (им может быть файл, блок данных, либо иной поток информации) вычисляется контрольная сумма (MD5, SHA-256 и т.д) – своего рода «отпечаток пальца», - которая и является адресом размещения информации. По этому адресу объект впоследствии может быть считан из устройства.

Сама архитектура подобной системы гарантирует неизменность хранимой информации. Если какой-либо объект был изменён, то у него будет уже другая контрольная сумма, и это будет уже другой объект, хранимый по другому адресу. При обращении по старому адресу объект будет считан в гарантированно первозданном виде, что исключает подмену, подделку и иные подобные действия, что неоценимо в области юриспруденции, безопасности, хранения ключевых доказательств и др.

Для каждого из записанных объектов может быть установлен определённый срок хранения, в течение которого он не может быть удалён. Этот срок может составлять от нескольких минут до нескольких лет, а также неограниченное хранение. В последнем случае удаление файла возможно только на заводе-изготовителе, либо только физическим уничтожением устройства.

Причём начало срока хранения не обязательно исчисляется с текущего момента, а может начинаться с любого определённого или даже неопределённого момента в будущем, как например хранение истории болезни в течение трёх лет с даты смерти пациента, которая на данный момент неизвестна.

Следствием архитектуры CAS является ещё одна интересная особенность: в случае если записываются несколько одинаковых файлов, но под разными именами, то реально будет записан только один объект, т.к. контрольные суммы, и следовательно, адреса размещения всех объектов совпадут, Это обстоятельство значительно экономит дисковое пространство. Однако же при чтении каждый файл будет читаться под своим именем.

Исходя из перечисленных свойств, устойства CAS получили распространение для архивного хранилища, а также для концепции «активного архива» (например при использовании ПО «Disk Extender», «E-Mail Extender»), в которых неиспользуемые данные незаметно для пользователя перемещаются на более дешёвое архивное хранилище, заменяясь ссылкой на оригинальный файл, по которой его может прочесть пользователь, не подозревающий об его реальном местонахождении

Конкретные реализации CAS-устройств представлены продуктами:

- ”Centera” фирмы EMC
- ”HCP” (“Hitachi Content Platform”) компании Hitachi, ранее имевшее название ”HCAP” (“Hitachi Content Archive Platform”).
- "HP StorageWorks Reference Information Storage System" (RISS) Hewlett-Packard
- "Sun StorageTek 5800 System" Sun Storagetek

6.3. Виртуализация систем хранения данных.

Общая структура облачных СХД

Облачные системы хранения данных – это метод организации хранилищ данных, при котором данные, принадлежащие предприятию, хранятся не в центре обработки данных (ЦОД) или на отдельных серверах этого предприятия, а на некотором множестве виртуальных серверов. Причем эти виртуальные сервера принадлежат компаниям, сдающим в аренду или продающим пользователям доступное дисковое пространство. Доступ к данным, которые хранятся в облачных СХД обеспечивается из любого места, где есть доступ к сети Интернет.

Общая структура взаимодействия пользователей с облачными системами хранения данных представлена на рис. 6.12.

Такие решения очень привлекательны для малых и средних предприятий, которым необходимо обеспечивать обработку и хранение больших объемов данных. В этом случае потребители виртуальных услуг оплачивают лишь используемый объем дискового пространства. Им не приходится оплачивать аппаратное обеспечение, обслуживание ЦОД и другие накладные расходы по содержанию центра обработки данных.

Рис. 6.12. Общая структура организации облачных систем хранения данных

В общем случае облачные вычисления (облачные СХД можно рассматривать как частный случай реализации облачных вычислений) — это модель предоставления по требованию пользователя сетевого доступа к совместно используемому пулу конфигурируемых вычислительных ресурсов (например, сетей, серверов, систем хранения, приложений и сервисов), которые могут быть предоставлены с минимальными усилиями по управлению и минимальным взаимодействием с поставщиком услуг.

Под сетевым доступом понимается обеспечение доступности сервисов облачных вычислений по сети посредством стандартных механизмов, поддерживающих использование гетерогенных платформ (таких как мобильные телефоны, ноутбуки, стационарные ПК, мониторы со встроенным «тонким клиентом»).

Пул ресурсов — Это совокупность вычислительных ресурсов поставщика услуг, которые объединяются в группы для преставления различным потребителям в рамках многопользовательской модели, при этом физические и виртуальные ресурсы могут назначаться и переназначаться в соответствии с потребностями клиентов. Потребитель сервиса как правило не знает и не контролирует точное физическое расположение предоставляемых ресурсов, но может на более высоком уровне абстракции специфицировать их размещение.

Возможны следующие модели представления услуг облачных вычислений.

Программное обеспечение как услуга (Software as a Service, SaaS). Потребителю предоставляется возможность использования приложений

поставщика, работающих в облачной инфраструктуре. Приложения доступны с различных клиентских устройств посредством тонких клиентов, таких как веб-браузер или специализированное клиентское приложение. Потребитель не управляет и не контролирует используемую облачную инфраструктуру, включая сети, серверы, операционные системы, системы хранения и даже некоторые параметры программных приложений, исключением может являться предоставление потребителю возможности управления ограниченным набором пользовательских настроек приложения.

Платформа как услуга (Platform as a Service, PaaS). Потребителю предоставляется возможность развертывания в облачной инфраструктуре собственных приложений, использующих базовую архитектуру (языки программирования, библиотеки, инструментарий) поддерживаемую поставщиком. Потребитель не управляет и не контролирует используемую базовую облачную инфраструктуру (сети, серверы, операционные системы, системы хранения), но управляет развернутыми приложениями и, возможно, рядом системных настроек, связанных с функционированием приложения.

Инфраструктура как услуга (Infrastructure as a Service, IaaS). Потребителю предоставляется возможность получения базовых ресурсов, таких как мощности систем хранения, вычислительные ресурсы, сетевые ресурсы и другие базовые вычислительные ресурсы которые могут использоваться потребителем для работы произвольного программного обеспечения, включая операционные системы и приложения. Потребитель не контролирует и не управляет базовой облачной инфраструктурой, но получает управление над операционными системами, предоставленными ресурсами систем хранения, приложениями и в некоторых случаях ограниченным набором сетевых ресурсов (например, локальный сетевой экран или виртуальный коммутатор).

Виртуализация ресурсов вычислительных систем

Одним из основных механизмов, обеспечивающих возможность предоставления услуг облачного хранения данных, является виртуализация ресурсов вычислительных систем.

Виртуализация (в области серверов) это совокупность программно-аппаратных средств, позволяющих на логическом уровне отделить вычислительные ресурсы системы от ее аппаратной части.

Обычно на одном сервере может работать только одна операционная система, управляющая работой данного сервера. Все вычислительные ресурсы этого сервера передаются этой операционной системе.

При виртуализации используются программные средства (программная прослойка, среда виртуализации), которые эмулируют заданную часть вычислительных ресурсов сервера в виде изолированного контейнера, представляющего собой виртуальную вычислительную машину (рис.42). Таких контейнеров на сервере может быть несколько и в каждом из них может

быть установлена своя операционная система. Эта программная прослойка (программная среда для создания виртуальных серверов) получила название гипервизора или монитора виртуальных машин.

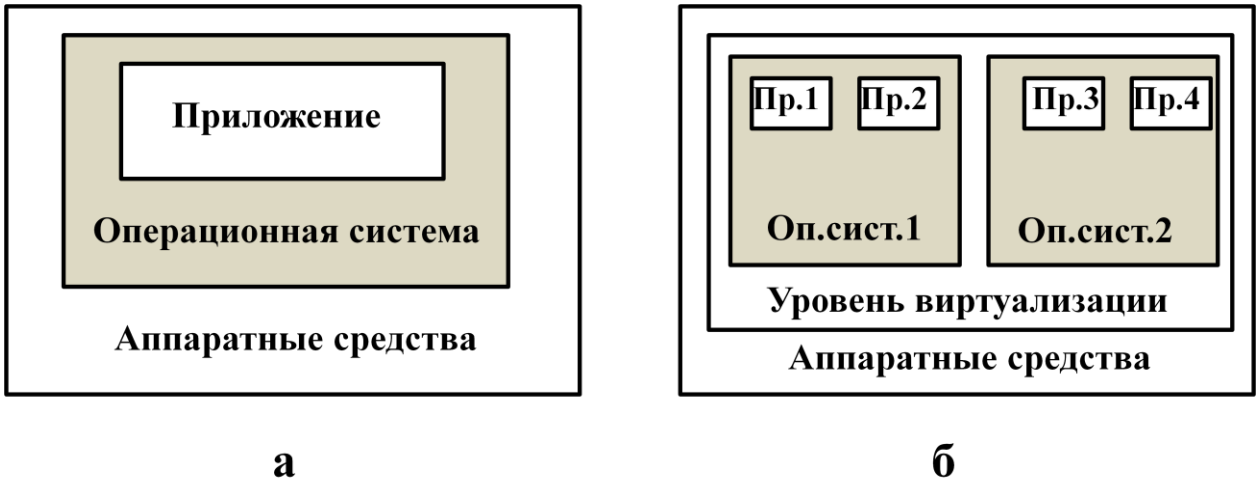


Рис.42. Виртуализация сервера (а- сервер до виртуализации, б- сервер после виртуализации)

В качестве дисковых ресурсов, которые представляются гипервизором для использования операционными системами в контейнерах, обычно используются как дисковые ресурсы локальных физических серверов, так и дисковые ресурсы внешних систем хранения данных.