УЧЕБНИК ДЛЯ ВУЗОВ

EMC²

От хранения данных к управлению информацией

Второе издание

EMC Education Services

Information Storage and Management

Storing, Managing, and Protecting Digital Information in Classic, Virtualized, and Cloud Environments

2nd Edition

Edited by
Somasundaram Gnanasundaram
Alok Shrivastava

WILEY
John Wiley & Sons, Inc.

УЧЕБНИК
ДЛЯ ВУЗОВ

От хранения данных к управлению информацией

Второе издание

Допущено Учебно-методическим объединением
вузов Российской Федерации по университетскому
политехническому образованию в качестве учебника для студентов
высших учебных заведений, обучающихся по направлениям подготовки
09.03.02 Информационные системы и технологии (уровень бакалавриата)
и 09.04.02 Информационные системы и технологии (уровень магистратуры)

ПИТЕР®

Санкт-Петербург · Москва · Екатеринбург · Воронеж
Нижний Новгород · Ростов-на-Дону
Самара · Минск

2016

ББК 32.973.233-018
УДК 004.65
О-80

О-80 От хранения данных к управлению информацией. 2-е изд. — СПб.: Питер, 2016. — 544 с.: ил.

ISBN 978-5-496-01859-3

За несколько лет, прошедших со времени выхода первого издания, известный нам мир претерпел невероятные изменения. Мы живем в эпоху цифровых технологий, когда объем имеющейся в мире информации увеличивается за два года более чем вдвое, а в следующем десятилетии IT-отделам придется справляться с информационными объемами, увеличившимися более чем в 50 раз, и это при том, что количество специалистов в области информационных технологий возрастет всего лишь в полтора раза. Теперь виртуализация и облачные вычисления для предприятий уже не просто один из возможных вариантов, а настоятельное условие для выживания на рынке. А так называемые большие данные предоставляют организациям новые, весьма действенные возможности анализа, обработки и управления возросшим объемом своего наиболее ценного актива — информации и приобретения весомых конкурентных преимуществ.

С приходом облачных вычислений появились совершенно новые технологии, компьютерные модели и дисциплины, сильно изменившие способы построения и запуска информационных технологий, а также управления ими. Чтобы идти в ногу с этими преобразованиями, были введены новые специальности, такие как технолог и архитектор облачных сред.

Книга раскроет перед вами новые перспективы и позволит разобраться с новыми технологиями и навыками, востребованными в наши дни для разработки, реализации, оптимизации и использования виртуализированных инфраструктур, а также управления ими с целью достижения тех преимуществ, которые бизнес может получить от применения облачных технологий.

12+ (В соответствии с Федеральным законом от 29 декабря 2010 г. № 436-ФЗ.)

ББК 32.973.233-018
УДК 004.65

Права на издание получены по соглашению с Wiley. Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельцев авторских прав.

ISBN 978-1118094839 англ.
978-5-496-01859-3

© Copyright © 2012 by EMC Corporation
© Перевод на русский язык ООО Издательство «Питер», 2016
© Издание на русском языке, оформление ООО Издательство «Питер», 2016

Оглавление

Предисловие ..17

Введение ..19

О редакторах ..21

Слова благодарности ..23

Раздел I. Системы хранения данных .. 25

Глава 1. Введение в хранение информации ..26

1.1. Хранение информации ..27

1.1.1. Данные ..28

1.1.2. Типы данных ..30

1.1.3. Большие данные ..31

1.1.4. Информация ..32

1.1.5. Хранение данных ..33

1.2. Эволюция архитектуры хранения данных ..33

1.3. Инфраструктура дата-центра ..35

1.3.1. Основные компоненты дата-центра ..35

1.3.2. Основные характеристики дата-центра ..37

1.3.3. Управление дата-центром ..38

1.4. Виртуализация и облачные вычисления ..39

Резюме ..40

Глава 2. Среда дата-центра ..42

2.1. Приложение ..43

2.2. Система управления базами данных ..44

2.3. Хост (вычислительная система) ..44

2.3.1. Операционная система ..45

2.3.2. Драйвер устройства ..46

2.3.3. Диспетчер томов ..46

2.3.4. Файловая система 48
2.3.5. Виртуализация вычислительных устройств 52
2.4. Соединение 54
2.4.1. Физические компоненты соединения 54
2.4.2. Протоколы обмена данными 55
2.5. Устройство хранения данных 56
2.6. Компоненты дискового накопителя 58
2.6.1. Магнитная пластина 59
2.6.2. Шпиндель 60
2.6.3. Головка чтения-записи 60
2.6.4. Кронштейн привода блока головок 61
2.6.5. Плата контроллера накопителя 61
2.6.6. Структура физического диска 62
2.6.7. Зональная побитовая запись 63
2.6.8. Адресация логических блоков 64
2.7. Производительность дискового накопителя 65
2.7.1. Время обслуживания 65
2.7.2. Загруженность дискового контроллера ввода-вывода 67
2.8. Доступ хоста к данным 69
2.9. Хранилище с прямым подключением 70
2.9.1. Преимущества и недостатки DAS 72
2.10. Проектирование хранилища на основе требований, предъявляемых приложениями, и показателей производительности дисков 72
2.11. Выстраивание очереди команд 75
2.12. Флеш-накопители 76
2.12.1. Компоненты и архитектура флеш-накопителей 77
2.12.2. Свойства флеш-накопителей корпоративного класса 78
2.13. Применение концепции на практике: VMware ESXi 79
Резюме 80

Глава 3. Защита данных: RAID-массив 82

3.1. Методы реализации RAID 83
3.1.1. Реализация RAID программными методами 83
3.1.2. Реализация RAID аппаратными методами 84
3.2. Компоненты RAID-массива 84
3.3. Методы RAID 84
3.3.1. Чередование 84
3.3.2. Зеркалирование 86
3.3.3. Контроль четности 87
3.4. RAID-уровни 89
3.4.1. RAID 0 90
3.4.2. RAID 1 90

3.4.3. Вложенный RAID 91
3.4.4. RAID 3 94
3.4.5. RAID 4 95
3.4.6. RAID 5 95
3.4.7. RAID 6. 96
3.5. Влияние RAID на производительность диска 97
3.5.1. Потребность приложений в IOPS и RAID-конфигурации 99
3.6. Сравнение RAID-конфигураций 99
3.7. Горячее резервирование 101
Резюме 101

Глава 4. Интеллектуальные системы хранения данных 103

4.1. Компоненты интеллектуальной системы хранения данных 104
4.1.1. Внешний интерфейс 105
4.1.2. Кэш-память 105
4.1.3. Внутренний интерфейс 111
4.1.4. Физический диск 112
4.2. Предоставление ресурсов хранения данных 112
4.2.1. Традиционное предоставление ресурсов хранения данных. 112
4.2.2. Виртуальное предоставление ресурсов хранения данных. 116
4.2.3. Маскирование LUN 119
4.3. Типы интеллектуальных систем хранения данных 119
4.3.1. Высокопроизводительные системы хранения данных 119
4.3.2. Системы хранения данных среднего класса 120
4.4. Практическая реализация концепций: EMC Symmetrix и VNX 121
4.4.1. Массив хранения данных EMC Symmetrix 122
4.4.2. Компоненты EMC Symmetrix VMAX 123
4.4.3. Архитектура Symmetrix VMAX 124
Резюме 124

Раздел II. Сетевые технологии хранения данных 127

Глава 5. Оптоволоконные сети хранения данных 128

5.1. Fibre Channel: обзор 129
5.2. Сеть хранения данных и ее эволюция 130
5.3. Компоненты SAN 131
5.3.1. Порты узлов 132
5.3.2. Кабели и разъемы 132
5.3.3. Соединительные устройства 134
5.3.4. Программы управления сетями хранения данных 135
5.4. Возможности соединений с применением FC 136
5.4.1. «Точка — точка» 136

5.4.2. Управляемая петля Fibre Channel .. 136
5.4.3. Коммутирующая матрица Fibre Channel. .. 138
5.5. Порты системы коммутации FC-SW .. 140
5.6. Архитектура Fibre Channel .. 141
5.6.1. Стек протоколов Fibre Channel .. 142
5.6.2. Адресация в Fibre Channel .. 143
5.6.3. Глобальные имена .. 144
5.6.4. FC-кадр .. 145
5.6.5. Структура и организация FC-данных .. 147
5.6.6. Управление потоками .. 147
5.6.7. Классы обслуживания .. 148
5.7. Службы систем коммутации .. 148
5.8. Типы регистрации в системе коммутации .. 149
5.9. Зонирование .. 150
5.9.1. Типы зонирования .. 152
5.10. Топологии FC SAN-сетей .. 153
5.10.1. Топология типа «решетка» .. 153
5.10.2. Топология систем коммутации «центр — периферия» .. 155
5.11. Виртуализация в SAN-среде .. 158
5.11.1. Виртуализация хранилища на уровне блоков .. 158
5.11.2. Виртуальная SAN-сеть (VSAN) .. 160
5.12. Практическая реализация концепций: EMC Connectrix и EMC VPLEX .. 161
5.12.1. EMC Connectrix .. 162
5.12.2. EMC VPLEX .. 164
Резюме .. 165

Глава 6. IP SAN и FCoE .. 167

6.1. Протокол iSCSI .. 168
6.1.1. Компоненты iSCSI .. 169
6.1.2. Варианты подключения iSCSI хоста .. 169
6.1.3. Топологии подключений iSCSI .. 170
6.1.4. Стек протоколов iSCSI .. 172
6.1.5. Протокольные блоки данных iSCSI .. 173
6.1.6. Обнаружение устройств в iSCSI .. 174
6.1.7. iSCSI-имена .. 175
6.1.8. Сеанс связи iSCSI .. 176
6.1.9. Выстраивание командных последовательностей iSCSI .. 177
6.2. FCIP .. 178
6.2.1. Стек протоколов FCIP .. 179
6.2.2. Топология FCIP .. 180
6.2.3. Производительность и безопасность FCIP .. 181

6.3. FCoE .. 182
6.3.1. Консолидация ввода-вывода с помощью FCoE. .. 182
6.3.2. Компоненты FCoE-сети .. 184
6.3.3. Структура кадра FCoE.. .. 187
6.3.4. Технологии, обеспечивающие работу FCoE .. 190
Резюме .. 192

Глава 7. Сетевые устройства хранения данных .. 194

7.1. Сравнение NAS-устройств с серверами общего назначения .. 195
7.2. Преимущества NAS.. .. 196
7.3. Файловые системы и совместный сетевой доступ к файлам .. 197
7.3.1. Доступ к файловой системе.. .. 197
7.3.2. Совместное сетевое использование файлов .. 198
7.4. Компоненты NAS .. 199
7.5. NAS-операции ввода-вывода. .. 200
7.6. Реализации NAS . .. 201
7.6.1. Унифицированные NAS-устройства.. .. 202
7.6.2. Возможности подключения унифицированных NAS-устройств . 202
7.6.3. Шлюзовые NAS-устройства.. .. 202
7.6.4. Возможности подключения шлюзовых NAS-устройств .. 204
7.6.5. Наращиваемые NAS-устройства .. 205
7.6.6. Возможности подключения масштабируемых NAS-устройств . .. 206
7.7. Протоколы совместного использования файлов, применяемые в NAS . .. 207
7.7.1. Сетевая файловая система.. .. 207
7.7.2. CIFS .. 209
7.8. Факторы, влияющие на производительность NAS . .. 210
7.9. Виртуализация на уровне файлов .. 213
7.10. Практическая реализация концепций: EMC Isilon и EMC VNX Gateway. .. 214
7.10.1. EMC Isilon.. .. 215
7.10.2. EMC VNX Gateway . .. 216
Резюме .. 217

Глава 8. Объектно-ориентированные и унифицированные хранилища данных .. 219

8.1. Устройства объектно-ориентированного хранения данных .. 220
8.1.1. Архитектура объектно-ориентированного хранилища .. 221
8.1.2. Компоненты OSD.. .. 222
8.1.3. Сохранение и извлечение объектов в OSD .. 223
8.1.4. Преимущества объектно-ориентированного хранилища .. 225
8.1.5. Наиболее распространенные примеры использования объектно-ориентированного хранилища . .. 226

8.2. Контентно-адресуемое хранилище. 228
8.3. Примеры использования CAS. 230
8.3.1. Решение в области здравоохранения: хранение результатов обследований пациентов. 230
8.3.2. Финансовое решение: хранение финансовых записей 231
8.4. Унифицированные хранилища 231
8.4.1. Компоненты унифицированного хранилища.. 232
8.5. Практическая реализация концепций: EMC Atmos, EMC VNX и EMC Centera 234
8.5.1. EMC Atmos 234
8.5.2. EMC VNX. 236
8.5.3. EMC Centera.. 237
Резюме.. 239

Раздел III. Резервное копирование, архивирование и репликация 241

Глава 9. Введение в обеспечение непрерывности бизнес-процессов.. .. 242

9.1. Доступность информации.. 243
9.1.1. Причины недоступности информации.. 243
9.1.2. Последствия вынужденного простоя 244
9.1.3. Оценка доступности информации. 245
9.2. Терминология обеспечения непрерывности бизнес-процессов.. 247
9.3. Жизненный цикл планирования обеспечения непрерывности бизнес-процессов. 250
9.4. Анализ сбоев.. 252
9.4.1. Единая точка отказа 252
9.4.2. Решение проблемы единых точек отказа 253
9.4.3. Программное обеспечение управления несколькими путями 255
9.5. Анализ факторов, влияющих на бизнес-процессы 256
9.6. Технологические решения по обеспечению непрерывности бизнес-процессов. 256
9.7. Практическая реализация концепции: EMC PowerPath 257
9.7.1. Свойства PowerPath.. 257
9.7.2. Динамическая балансировка нагрузки 258
9.7.3. Автоматический обход сбойных путей 261
Резюме.. 264

Глава 10. Резервное копирование и архивирование.. 266

10.1. Цели резервного копирования 267
10.1.1. Аварийное восстановление 267
10.1.2. Оперативное восстановление 268
10.1.3. Архивирование. 268
10.2. Факторы, определяющие порядок резервного копирования 268

10.3. Уровни объемов данных резервного копирования .. 270
10.4. Факторы, определяющие порядок восстановления данных .. 273
10.5. Методы резервного копирования .. 274
10.6. Архитектура резервного копирования .. 276
10.7. Операции резервного копирования и восстановления .. 277
10.8. Топологии резервного копирования .. 279
10.9. Резервное копирование в средах сетевых устройств хранения данных (NAS) .. 282
10.9.1. Резервное копирование с использованием и без использования сервера .. 283
10.9.2. Резервное копирование с использованием NDMP-протокола .. 285
10.10. Адресаты резервного копирования .. 286
10.10.1. Резервное копирование на ленту .. 287
10.10.2. Резервное копирование на диск .. 290
10.10.3. Резервное копирование на виртуальную ленту .. 291
10.11. Дедупликация данных при резервном копировании .. 295
10.11.1. Методы дедупликации данных .. 295
10.11.2. Реализация методов дедупликации данных .. 296
10.12. Резервное копирование в виртуализированных средах .. 299
10.13. Архивирование данных .. 301
10.14. Архитектура решений, связанных с архивированием данных .. 303
10.14.1. Сценарий использования: архивирование электронной почты .. 304
10.14.2. Сценарий использования: архивирование файлов .. 305
10.15. Практическая реализация концепций: EMC NetWorker, EMC Avamar и EMC Data Domain .. 305
10.15.1. EMC NetWorker .. 306
10.15.2. EMC Avamar .. 306
10.15.3. EMC Data Domain .. 308
Резюме .. 309

Глава 11. Локальная репликация .. 311

11.1. Терминология репликаций .. 312
11.2. Использование локальных реплик .. 313
11.3. Согласованность реплик .. 314
11.3.1. Согласованность реплицированных файловых систем .. 314
11.3.2. Согласованность реплицированных баз данных .. 315
11.4. Технологии локальных репликаций .. 318
11.4.1. Локальная репликация на основе использования хоста .. 318
11.4.2. Репликация на основе использования массива хранения данных .. 322
11.4.3. Локальная репликация на основе использования сети .. 328

11.5. Отслеживание изменений в источнике и реплике .. 330
11.6. Особенности восстановления и перезапуска .. 333
11.7. Создание нескольких реплик .. 334
11.8. Локальная репликация в виртуальной среде .. 335
11.9. Практическая реализация концепций: EMC TimeFinder, EMC SnapView и EMC RecoverPoint .. 336
11.9.1. EMC TimeFinder .. 336
11.9.2. EMC SnapView .. 337
11.9.3. EMC RecoverPoint .. 338
Резюме .. 339

Глава 12. Удаленная репликация .. 340

12.1. Режимы удаленной репликации .. 340
12.2. Технологии удаленной репликации .. 344
12.2.1. Удаленная репликация на основе использования хоста .. 344
12.2.2. Удаленная репликация на основе использования массивов хранения данных .. 346
12.2.3. Удаленная репликация на основе использования сети .. 350
12.3. Трехсторонняя репликация .. 352
12.3.1. Трехсторонняя репликация — каскадное решение (с несколькими транзитными участками) .. 352
12.3.2. Трехсторонняя репликация — треугольное решение (с несколькими приемниками) .. 355
12.4. Решения по осуществлению миграции данных .. 357
12.5. Удаленная репликация и миграция в виртуализированной среде .. 359
12.6. Практическая реализация концепций: EMC SRDF, EMC MirrorView и EMC RecoverPoint .. 361
12.6.1. EMC SRDF .. 362
12.6.2. EMC MirrorView .. 363
12.6.3. EMC RecoverPoint .. 363
Резюме .. 363

Раздел IV. Облачные вычисления .. 365

Глава 13. Облачные вычисления .. 366

13.1. Высокоэффективные облачные технологии .. 367
13.2. Характеристики облачных вычислений .. 368
13.3. Преимущества, получаемые от облачных вычислений .. 370
13.4. Модели облачного обслуживания .. 370
13.4.1. Инфраструктура как услуга .. 371
13.4.2. Платформа как услуга .. 372
13.4.3. Программное обеспечение как услуга .. 372

13.5. Модели развертывания облака. .. .. .

14.6. Практическая реализация концепций: продукты RSA и VMware Security .. 427
14.6.1. RSA SecureID .. 427
14.6.2. RSA Identity and Access Management .. 427
14.6.3. RSA Data Protection Manager .. 428
14.6.4. VMware vShield .. 428
Резюме .. 429

Глава 15. Управление инфраструктурой хранения данных .. 431

15.1. Мониторинг инфраструктуры хранения данных .. 432
15.1.1. Отслеживаемые параметры .. 432
15.1.2. Отслеживаемые компоненты .. 434
15.1.3. Примеры мониторинга .. 436
15.1.4. Предупреждения .. 442
15.2. Действия по управлению инфраструктурой хранения данных .. 443
15.2.1. Управление доступностью .. 444
15.2.2. Управление объемами .. 444
15.2.3. Управление производительностью .. 445
15.2.4. Управление безопасностью .. 445
15.2.5. Составление отчетов .. 446
15.2.6. Управление инфраструктурой хранения данных в виртуализированной среде .. 446
15.2.7. Примеры управления хранилищами данных .. 448
15.3. Проблемы управления инфраструктурой хранилища данных .. 453
15.4. Выработка идеального решения .. 454
15.4.1. Инициативные разработки в вопросах управления хранилищами данных .. 454
15.4.2. Платформа управления в масштабах предприятия .. 455
15.5. Управление жизненным циклом информации .. 456
15.6. Многоуровневое хранение данных .. 458
15.6.1. Многоуровневое хранение в массиве хранения данных .. 459
15.6.2. Многоуровневое хранение между массивами хранения данных .. 461
15.7. Практическая реализация концепций: средства управления инфраструктурой от компании EMC .. 462
15.7.1. EMC ControlCenter и ProSphere .. 462
15.7.2. EMC Unisphere .. 464
15.7.3. EMC Unified Infrastructure Manager .. 464
Резюме .. 465

Приложение А. Характеристики операций ввода-вывода, проводимых приложениями. 467

Произвольный и последовательный 467
Операции чтения и записи. 468
Размер запроса на ввод-вывод 469

Приложение Б. Параллельный SCSI-интерфейс. 470

Семейство стандартов SCSI. 471
Клиент-серверная модель SCSI 472
Адресация в параллельном SCSI 474

Приложение В. Упражнения по разработке SAN-сетей 475

Упражнение 1. 475
Решение 475
Упражнение 2. 476
Решение 476

Приложение Г. Упражнения по доступности информации 478

Упражнение 1. 478
Решение 478
Упражнение 2. 479
Решение 479

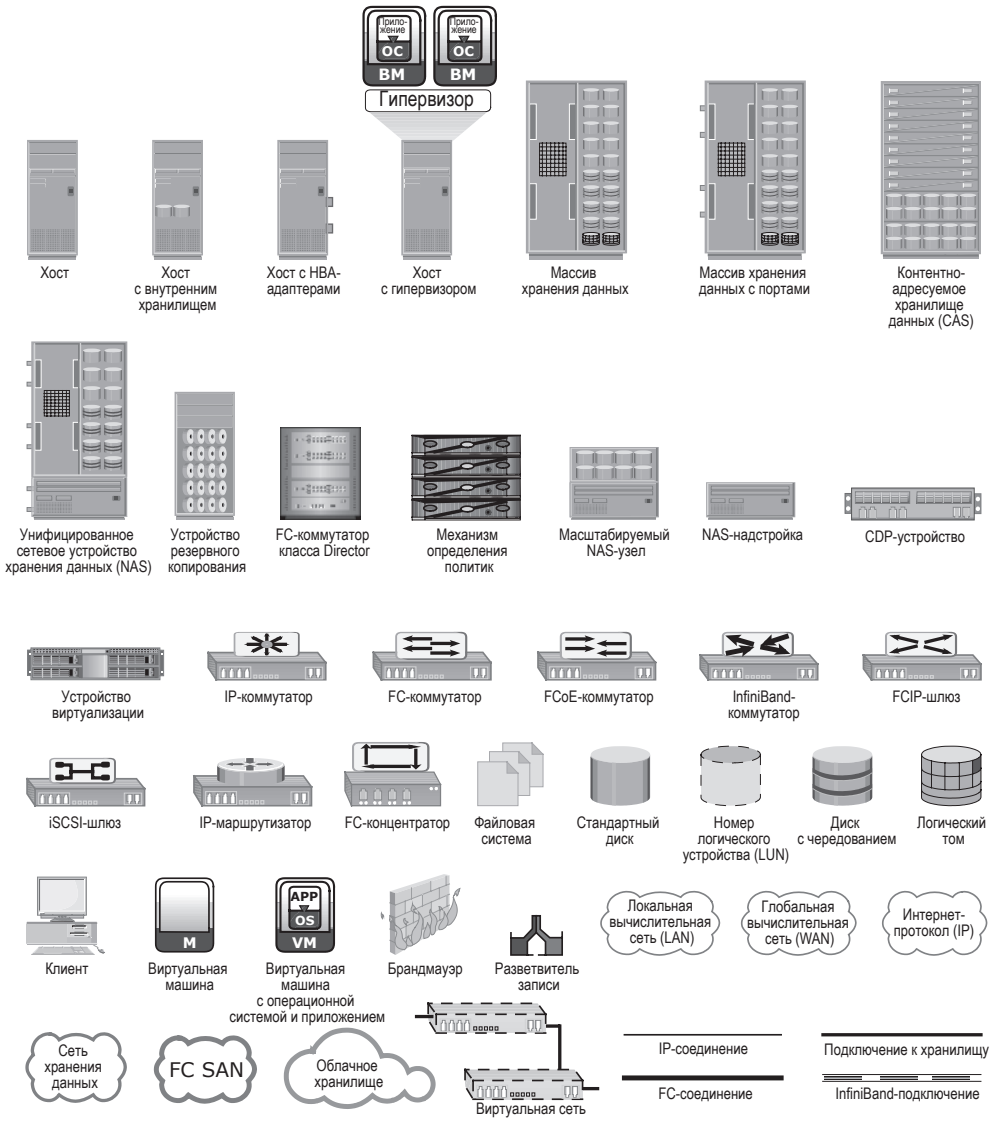
Приложение Д. Сетевые технологии для удаленной репликации 480

DWDM. 480
CWDM. 481
SONET 481

Сокращения и аббревиатуры 483

Глоссарий. 495

Значки, используемые в этой книге



Предисловие

За два коротких года со времени выхода первого издания книги известный нам мир претерпел невероятные изменения. Мы живем в эпоху цифровых технологий, когда объем имеющейся в мире информации увеличивается за два года более чем вдвое, а в следующем десятилетии IT-отделам придется справляться с информационными объемами, увеличившимися более чем в 50 раз, и это при том, что количество специалистов в области информационных технологий возрастет всего лишь в полтора раза (согласно отчету, представленному в июне 2011 года аналитической компанией IDC «Исследование цифровой вселенной», подготовку которого спонсировала компания EMC). Теперь виртуализация и облачные вычисления для предприятий уже не просто один из возможных вариантов, а настоятельное условие для выживания на рынке. А так называемые большие данные предоставляют организациям новые, весьма действенные возможности анализа, обработки и управления возросшим объемом своего наиболее ценного актива — информации и приобретения весомых конкурентных преимуществ.

В результате всего этого индустрия информационных технологий претерпевает огромные изменения. С приходом облачных вычислений появились совершенно новые технологии, компьютерные модели и дисциплины, сильно изменившие способы построения и запуска информационных технологий, а также управления ими. Чтобы идти в ногу с этими преобразованиями, были введены новые специальности, такие как технолог и архитектор облачных сред. Организации, работающие в области информационных технологий, превратились из операторов инфраструктуры операционного отдела, выполняющих задачу поддержания работоспособности оборудования, в ключевые стратегические фигуры организации бизнеса со специализацией на предоставлении информационных технологий в виде услуг.

Все эти изменения потребовали новых ключевых умений и навыков в IT-организации, нового образа технологического мышления в контексте бизнес-требований и стратегических целей и даже новой организационной

структуры дата-центра. Профессионалы в области хранения информации и управления ею должны пополнить уже имеющиеся знания и выработать дополнительные навыки в области тех технологий, которые играют наиболее важную роль в успешном освоении сложного многолетнего пути в облака, то есть освоить, в частности, технологии виртуализации, конвергентных сетей, информационной безопасности, защиты данных, а также начисления платы за услуги хранения и анализа данных.

Мы переработали эту книгу, чтобы раскрыть перед вами новые перспективы и дать возможность глубокого проникновения в суть новых технологий и навыков, востребованных в наши дни для разработки, реализации, оптимизации и использования виртуализированных инфраструктур, а также управления ими с целью достижения тех преимуществ, которые бизнес может получить от применения облачных технологий. Учиться вы будете на основе тех знаний, которыми с вами поделятся высококвалифицированные специалисты компании EMC, имеющие наиболее богатый опыт в обучении, проведении аттестации специалистов и практической работе в данной отрасли.

Кем бы вы ни были — профессионалом в области хранения информации и управления ею, находящимся в процессе виртуализации своего дата-центра или создания надежной облачной инфраструктуры, или же простым человеком, заинтересованным в изучении понятий и принципов новых парадигм, — совершенствование вашего мастерства еще не приобретало подобной степени важности. Ускоряя процесс самосовершенствования с помощью этой книги и получая преимущества от обучения и прохождения уже доступной для вас новой квалификационной аттестации, вы сможете поспособствовать закрытию критического дефицита квалифицированных специалистов в информационной отрасли, получить возможность карьерного роста и начать полноценно вносить свой вклад в успешное расширение вашей компании и достижение ею высоких показателей стабильности и рентабельности.

Проблем в этой области еще хватает, но по делам вашим вам же и воздастся. Нельсон Мандела как-то сказал: «Образование — это самое мощное оружие, с помощью которого можно изменить мир». Я надеюсь, что эта книга станет одним из главных источников вашего IT-образования и профессионального развития независимо от того, в какой роли вы пребываете в настоящий момент, и что вы непременно воспользуетесь возможностью измениться самим и помочь изменить к лучшему этот мир.

Томас П. Клэнси, вице-президент компании EMC, ответственный за предоставление образовательных услуг.
Май 2012 года

Введение

Хранение информации является важнейшей составляющей информационных технологий. Каждое мгновение частные лица и организации создают огромные объемы цифровой информации, которую нужно сохранить, защитить, оптимизировать и которой нужно управлять в классической, виртуализированной и быстро развивающейся облачной среде.

Не так давно хранилища информации представлялись только в виде накопителей на магнитных дисках или лентах, подключенных к компьютеру с целью хранения данных. Даже в наши дни та наиважнейшая роль, которую играет технология хранения информации в достижении высоких уровней доступности, производительности, интеграции и оптимизации всей IT-инфраструктуры, понятна в основном только специалистам в области индустрии хранения данных. За последнее десятилетие хранение информации развилось в весьма сложную технологию, предоставляющую разнообразные решения в области цифровой информации, касающиеся хранения, управления, подключения, защиты, обеспечения безопасности, совместного использования и оптимизации.

Широкое внедрение виртуализации, появление облачных вычислений, многократный ежегодный рост объемов данных и наличие разнообразных типов и источников данных — все эти факторы сделали современные технологии хранения данных еще важнее и актуальнее для успешной деятельности как коммерческих, так и любых других организаций. Наиболее остро по сравнению с прежними временами перед руководителями IT-отрасли встала проблема найма на работу и подготовки высококвалифицированных технических специалистов сферы хранения данных, хорошо разбирающихся в работе классических, виртуализированных и облачных сред.

Многие ведущие университеты и колледжи теперь включают в свои программы изучения обычных компьютерных или информационных технологий курсы по изучению технологий хранения данных, поскольку до сих пор IT-профессионалы, даже те, у которых за плечами многолетний опыт работы, не получили от прежнего формального обучения никаких существенных преимуществ. Поэтому многие опытные профессионалы, включая администраторов приложений, систем, баз данных и сетей, не имеют общего мнения о влиянии технологии хранения данных на их сферы деятельности.

Эта книга задумывалась и создавалась с целью помочь профессионалам и студентам обрести глубокое понимание всех составляющих технологии хранения данных. Хотя примеры промышленных образцов, использованные в данной книге, относятся к изделиям компании EMC Corporation, понимание технологических концепций и принципов подготовят вас к легкому освоению технической продукции от различных поставщиков.

Книга содержит 15 глав, сведенных в пять разделов. Темы изложены по нарастанию сложности материала, все последующие темы должны изучаться после освоения предшествующих. В разделе I вводятся понятия виртуализации и облачной инфраструктуры, которые затем упоминаются по всей книге, чтобы обеспечить рассмотрение технологий хранения данных в контексте традиционной или классической среды, виртуализированной среды и быстро развивающейся облачной среды.

Раздел I «Системы хранения данных». В четырех главах этого раздела рассматриваются проблемы информационного бума и изменения характера хранимой информации, даются определения системе хранения данных и среде дата-центра, предоставляется обзор развития технологии хранения данных и дается введение в интеллектуальные системы хранения данных. Кроме этого, в данном разделе вводятся понятия виртуализации и облачных вычислений.

Раздел II «Сетевые технологии хранения данных». В четырех главах этого раздела рассматриваются сеть хранения данных на основе использования оптоволоконной технологии — Fibre Channel storage area network (FC-SAN), сеть хранения данных на основе использования интернет-протокола — Internet Protocol SAN (IP SAN), сетевые устройства хранения данных — Network-attached storage (NAS), технологии хранения данных на основе объектов и унифицированное хранилище данных. Кроме этого, в разделе рассматриваются концепции объединений хранилищ и конвергентных сетей (FCoE).

Раздел III «Резервное копирование, архивирование и репликация». В четырех главах этого раздела рассматриваются обеспечение непрерывности бизнес-процессов, создание резервных копий и восстановления данных, дедупликация данных, архивирование данных, локальная и удаленная репликации данных как в классической, так и в виртуализированной среде.

Раздел IV «Облачные вычисления». В единственной главе этого раздела рассматриваются облачные вычисления, включая их инфраструктуру, модели обслуживания, варианты развертывания и факторы, определяющие порядок перехода к использованию облачных вычислений.

Раздел V «Обеспечение безопасности и управление инфраструктурой хранения данных». В двух главах этого раздела рассматриваются обеспечение безопасности инфраструктуры хранения данных, мониторинг инфраструктуры и управление этой инфраструктурой, включая проблемы обеспечения безопасности и управления в виртуализированной и облачной средах.

У этой книги есть веб-сайт, содержащий дополнительный актуальный на данный момент учебный материал для усвоения и ознакомления. Адрес сайта: http://education.EMC.com/ismbook.

О редакторах

Гнанасундарам Сомасундарам (Somasundaram Gnanasundaram), или Сому (Somu), является директором подразделения образовательных услуг компании EMC — EMC Education Services, ведущей всемирной системы подготовки специалистов по практической работе в сфере интересов компании. Сому является создателем открытой программы обучения компании EMC, призванной устранить существующие пробелы в знаниях в той области информационных технологий, которая касается хранения информации, а также таких быстро развивающихся технологий, как облачные вычисления. Под его непосредственным руководством получила существенное развитие программа академического партнерства ведущих университетов мира в области информационных технологий — «EMC Academic Alliance», по которой технологиям хранения и управления информацией обучаются тысячи студентов по всему миру. Основные зоны ответственности Сому включают руководство работой всемирной команды профессионалов, поиск по всему миру поставщиков образовательных услуг в области информационных технологий и установление с ними партнерских взаимоотношений, а также общее руководство всеми образовательными инициативами компании EMC по подготовке специалистов в сфере интересов компании. До назначения на эту должность Сому занимал различные управленческие и ведущие должности в компании EMC, а также других ведущих компаниях — поставщиках услуг в области информационных технологий. У него имеется степень бакалавра технологии университета Anna University (Ченнаи) и степень магистра технологии Индийского технологического института — Indian Institute of Technology (Мумбаи, Индия). Сому работает в сфере информационных технологий уже свыше 25 лет.

Алок Шривастава (Alok Shrivastava) — генеральный директор подразделения образовательных услуг компании EMC — EMC Education Services. Алок является создателем ряда успешных образовательных инициатив компании EMC, включая ведущую в индустрии программу «Профессионал, признанный

EMC» — EMC Proven Professional, а также такой программы подготовки, принадлежащей компании EMC, как «Академический союз» (Academic Alliance). Кроме того, он является автором этой уникальной и весьма полезной книги, посвященной технологиям хранения информации. Алок является инициатором новых направлений и руководителем команды талантливых специалистов, практиков и профессионалов, разрабатывающих образовательные материалы по техническому обучению мирового класса, предназначенные для работников компании EMC, ее партнеров, клиентов, студентов и других профессионалов данной отрасли и охватывающие такие технологические области, как хранение данных, виртуализация, облачные вычисления и работа с большими данными. До того как он достиг больших успехов в образовательной деятельности, Алок создал и возглавил весьма успешную команду инженеров по предпродажной подготовке изделий компании EMC в странах Тихоокеанского региона и в Японии. Ранее, по мере своего карьерного роста, Алок был системным менеджером, менеджером по хранению данных и консультантом ряда самых крупных в мире дата-центров и других IT-организаций по проведению резервного копирования и восстановления данных, а также восстановлению работоспособности хранилищ данных после возникновения чрезвычайных ситуаций. У него имеется двойная степень магистра Индийского технологического института — Indian Institute of Technology (Мумбаи, Индия) и Сагарского университета — University of Sagar (Индия). Алок проработал в области технологий хранения информации, сохраняя неизменную приверженность к ней, более 30 лет своей карьеры в IT-индустрии.

Слова благодарности

Главным вопросом, с которым нам пришлось столкнуться в самом начале работы над этой книгой 2008 году, стал подбор команды специалистов по конкретным направлениям, входящим в широкий спектр технологий, формирующих современную инфраструктуру хранения информации.

Ключевым фактором, работавшим в нашу пользу, было наличие у компании EMC технологий, инноваций и множества светлых голов, работающих в данной отрасли. Когда мы обратились к отдельным специалистам, они вдохновились перспективой издания подробной книги о технологиях хранения информации не меньше нашего. У них появилась возможность обмена опытом с профессионалами и студентами по всему миру.

Эта книга является результатом усилий ряда ключевых организации компании EMC во главе с EMC Education Services. Она создана при поддержке таких подразделений, как CTO, Global Marketing и EMC Engineering.

Первое издание этой книги вышло в 2009 году под руководством Ганеша Раджаратнама (Ganesh Rajaratnam) из EMC Education Services и доктора Дэвида Блэка (David Black) из EMC CTO. Профессионалы и студенты до сих пор считают данную книгу наиболее популярным в мире изданием, посвященным технологии хранения данных. Кроме выхода изданий на английском языке и в электронном виде, книга была издана на китайском, португальском и русском языках.

С появлением облачных вычислений и широкого внедрения в организациях технологий виртуализации мы почувствовали, что настало время обновить содержимое книги, включив в него вопросы хранения информации с использованием этих новых технологий, а также новых разработок в области хранения информации. Возглавил работу по обновлению содержимого книги с целью выпуска второго издания Ашиш Гарг (Ashish Garg) из Education Services. Содержимое книги подвергла пересмотру также команда специалистов в конкретных областях, которую возглавили Джо Милардо (Joe Milardo) и Нэнси Гесслер (Nancy Gessler).

Мы очень благодарны специалистам компании EMC за их участие в работе над различными главами данной книги и пересмотре их содержимого.

Свой вклад в содержимое книги внеслиРодриго Элвз (Rodrigo Alves), Чарли Брукс (Charlie Brooks), Дебасиш Чакрабарти (Debasish Chakrabarty), Дайана Дэвис (Diana Davis), Амит Дешмук (Amit Deshmukh), Майкл Дулавит (Michael Dulavitz), д-р Ванки Гурумурти (Vanchi Gurumoorthy), Саймон Хокшоу (Simon Hawkshaw), Анбусельви Джейакумар (Anbuselvi Jeyakumar), Сагар Котекар Патил (Sagar Kotekar Patil), Андрэ Россув (Andre Rossouw), Тони Сантамария (Tony Santamaria), Сараванарадж Сридхаран (Saravanaraj Sridharan), Ганеш Сундаресан (Ganesh Sundaresan), Джим Трейси (Jim Tracy), Ананд Варкар (Anand Varkar), д-р Визвонт В. С. (Viswanth VS).

В качестве рецензентов с нами работалиРонен Эртц (Ronen Artzi), Эрик Бэйз (Eric Baize), Грэг Бальтазар (Greg Baltazar), Эдвард Бэлл (Edward Bell), Эд Белливо (Ed Belliveau), Пол Брант (Paul Brant), Юрген Буш (Juergen Busch), Кристофер Чок (Christopher Chaulk), Брайан Коллинз (Brian Collins), Хуан Кубиллос (Juan Cubillos), Джон Дауд (John Dowd), Роже Дюпюи (Roger Dupuis), Дебора Филер (Deborah Filer), Бала Ганешан (Bala Ganeshan), Джейсон Джервикас (Jason Gervickas), Джоди Гонкальвес (Jody Goncalves), Джек Харвуд (Jack Harwood), Манодж Кумар (Manoj Kumar), Артур Джонсон (Arthur Johnson), Мишель Лавуа (Michelle Lavoie), Том МакГован (Tom McGowan), Джефри Мур (Jeffery Moore), Тоби Моррэл (Toby Morral), Уэйн Поли (Wayne Pauley), Питер Попенюк (Peter Popieniuck), Ира Шилд (Ira Schild), Шашакант Пунуру (Shashikanth Punuru), Муругесон Пурушотаман (Murugeson Purushothaman), Шекхар Сенгупта (Shekhar Sengupta), Кевин Шеридан (Kevin Sheridan), Эд ВанСикл (Ed VanSickle), Майк Уорнер (Mike Warner), Ронни Зуби (Ronnie Zubi), Эван Берли (Evan Burleigh).

Мы также благодарны Маллику Мотилалу (Mallik Motilal) из компании EMC за его работу над созданием иллюстраций, Маллешу Гурраму (Mallesh Gurram) из компании EMC — за дизайн обложки и издательству John Wiley & Sons — за их своевременное содействие в выходе этой книги из печати.

Гнанасундарам Сомасундарам,
директор подразделения образовательных услуг корпорации EMC

Алок Шривастава,
генеральный директор подразделения образовательных услуг
корпорации EMC
Март 2012 года

Раздел

I

Системы хранения данных

В ЭТОМ РАЗДЕЛЕ

Глава 1. Введение в хранение информации
Глава 2. Среда дата-центра
Глава 3. Защита данных: RAID-массив
Глава 4. Интеллектуальные системы хранения данных

Глава 1
Введение в хранение информации

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Данные и информация
- Структурированные и неструктурированные данные
- Эволюция архитектуры хранилищ данных
- Основные элементы дата-центра
- Виртуализация и облачные вычисления

С каждым днем информация играет все более важную роль в нашей повседневной жизни. Мы стали информационно зависимым обществом XXI века и живем в мире команд и запросов. Это означает, что информация необходима нам в определенное время и в определенном месте. Ежедневно мы выходим в Интернет для поиска информации, общения в социальных сетях, отправки и получения электронной почты, обмена видео и фотографиями, а также решения ряда других задач. При растущем количестве производящих контент устройств все больше информации создается частными лицами, а не организациями — промышленными, правительственными, некоммерческими и т. д. Информация, созданная частными лицами, приобретает ценность, когда ею обмениваются с другими. В момент создания информация обычно размещена на таких устройствах, как мобильные телефоны, смартфоны, планшетные компьютеры, камеры, ноутбуки. Для обмена ее необходимо загрузить через сеть в центральные хранилища данных (так называемые репозитории, или дата-центры). Хотя большая часть информации создается частными лицами, хранением этой информацией и управлением ею занимается относительно небольшое число организаций.

Значимость, взаимосвязанность и объем информации в мире бизнеса тоже продолжают расти стремительными темпами. Успех в бизнесе зависит от быстрого и надежного доступа к соответствующей информации. В качестве примера можно привести ряд коммерческих процессов или систем,

зависящих от цифровой информации: бронирование авиабилетов, дистанционную рассылку счетов, интернет-торговлю, предоставление электронных банковских услуг, обработку транзакций кредитных карт, торговлю акциями и ресурсами, обработку данных о состоянии здоровья, обработку оперативных данных научных исследований и т. д.

Растущая зависимость бизнеса от информации увеличила потребности в хранении и защите данных и управлении ими. К этим требованиям добавляются правовые, нормативные и договорные обязательства, касающиеся доступности и защиты данных.

Организации обычно используют один или несколько дата-центров, представляющих собой объекты, содержащие информационное хранилище и другие информационные технологические ресурсы (IT-ресурсы) для производства вычислений, сетевого обмена информацией и управления ею. В обычных дата-центрах каждой бизнес-единице или приложению выделяются свои ресурсы хранения. Рост количества новых приложений и существенный рост объема данных привели к появлению в этих дата-центрах обособленных инфраструктур отдельных хранилищ информации, что усложнило управление информацией и стало причиной недоиспользования ресурсов хранения данных. Виртуализация позволяет оптимизировать использование ресурсов и упрощает управление ими. Организации в своих дата-центрах применяют виртуализацию, превращая их тем самым в виртуализированные дата-центры (virtualized data centers, VDC). Облачные вычисления, являющиеся фундаментальными подвижками в способах построения и предоставления IT-ресурсов, а также управления ими, приводят к дальнейшему упрощению хранения информации и управления ею, а также сокращают время предоставления этих ресурсов. Облачные вычисления принесли с собой полностью автоматизированный процесс отправки и выполнения запросов, позволяющий пользователям по мере надобности быстро получать ресурсы для хранения данных и другие IT-ресурсы. Благодаря облачным вычислениям организация имеет возможность быстрого развертывания приложений, а базовая емкость хранилища может расширятся и сужаться исходя из деловых потребностей.

В этой главе рассказывается об эволюции архитектуры хранилищ информации от модели, основанной на применении серверов, до информационно-центрической модели. В ней также дается обзор виртуализации и облачных вычислений.

1.1. Хранение информации

Организации обрабатывают данные для извлечения информации, необходимой для своих повседневных операций. Хранилище представляет собой репозиторий, дающий пользователям возможность постоянного хранения и извлечения этих цифровых данных.

1.1.1. Данные

Данные — это набор сведений или фактов, из которых могут быть сделаны выводы. Рукописные документы, печатное издание, семейная фотография, распечатанные и надлежащим образом подписанные закладные, банковская бухгалтерия и авиабилеты — все это примеры данных.

До начала компьютерной эпохи данные создавались и передавались преимущественно на бумаге или на пленке. Сегодня эти же данные можно конвертировать в более удобные формы, например в электронное сообщение, электронную книгу, цифровое изображение или цифровое кино. Эти данные можно создавать посредством компьютера и хранить в цифровом виде, представленном строками из нулей и единиц (рис. 1.1). Такой вид данных, доступных пользователю только после компьютерной обработки, называют цифровыми данными.

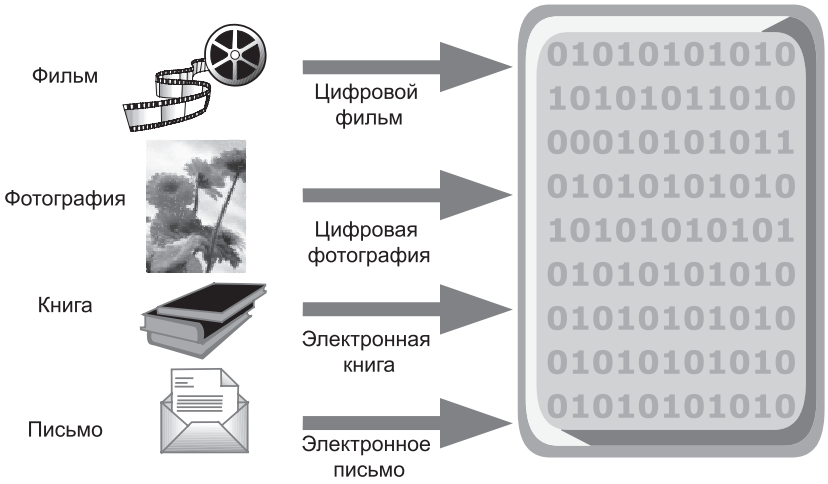


Рис. 1.1. Цифровые данные

С развитием компьютерных технологий и технических средств связи скорость создания данных и скорость обмена данными увеличились в геометрической прогрессии. Вот лишь несколько факторов, способствовавших росту цифровых данных.

- **Расширение возможностей обработки данных:** современные компьютеры дают возможность существенно увеличить производительность при обработке и хранении данных. Это позволяет преобразовывать различные традиционные виды контента на различных носителях в цифровые форматы.
- **Снижение цен на цифровые носители:** технологический прогресс и снижение стоимости устройств хранения способствовали появлению

недорогих решений в области хранения данных. Снижение цен способствовало росту объемов генерирования и хранения данных.

- **Появление доступных высокоскоростных технологий связи:** скорость обмена цифровыми данными сегодня намного выше, чем при использовании традиционных методов. Доставка адресату письма, написанного вручную, может занять неделю, а пересылка электронного сообщения получателю обычно занимает всего несколько секунд.
- **Рост количества приложений и смарт-устройств:** существенный вклад в создание цифрового контекста вносят высокотехнологичные приложения, а также смартфоны, планшетные компьютеры и иные цифровые устройства.

Недорогие и более простые пути создания, сбора и хранения всех типов данных в сочетании с растущими потребностями бизнеса и частных лиц привели к ускоренному росту объемов данных, больше известному как информационный бум. Этому буму в равной степени поспособствовали как коммерческие предприятия, так и частные лица.

Со временем важность и значимость данных меняется. Значимость большей части создаваемых данных носит краткосрочный характер, и со временем такие данные обесцениваются. Это определяет тип решений, используемых для хранения данных. Обычно самые свежие данные, обладающие высокой степенью востребованности, хранятся на более скоростном и дорогостоящем устройстве. По мере утраты актуальности они могут перемещаться на менее скоростное и дорогое, но вполне надежное устройство хранения данных.

ПРИМЕРЫ ИССЛЕДОВАНИЙ И КОММЕРЧЕСКИХ СВЕДЕНИЙ

В качестве примеров научных данных и коммерческих сведений можно привести следующую информацию.

- **Данные о клиентах:** совокупность данных о клиентах компании, в том числе подробные сведения о заказах, адресах поставок и истории покупок.
- **Данные о товарах:** включают данные о различных характеристиках товаров, например их учетные данные, описания, системы ценообразования, доступность и объемы продаж.
- **Медицинские данные:** сведения о здравоохранении, например истории болезней, радиологические снимки, подробности медикаментозного и иного лечения и информация о страховках.
- **Сейсмоданные:** сейсмология является наукой о землетрясениях. Она включает в себя сбор данных и изучение процессов с целью получения информации, помогающей выявить точные места и магнитуды землетрясений.

Коммерческие предприятия генерируют огромное количество данных, а затем извлекают из них значимую информацию с целью получения экономической выгоды. По этой причине предприятиям необходимо заниматься сохранением данных и обеспечивать возможность доступа к ним на протяжении длительного периода времени. Более того, данные могут различаться по степени важности и требовать особого подхода. Например, по закону банки должны обеспечивать сохранность и точность данных клиентских счетов. Некоторые предприятия хранят данные миллионов клиентов. Для этих данных обеспечиваются безопасность и целостность в течение длительного времени. Для этого необходимы устройства хранения данных больших объемов с улучшенными характеристиками безопасности и совместимости, позволяющие долго сохранять данные.

1.1.2. Типы данных

Данные могут быть классифицированы как структурированные и неструктурированные (рис. 1.2) — в зависимости от способа управления данными и их хранения. Структурированные данные организуются в строки и столбцы строго определенного формата, чтобы приложения могли их извлекать и эффективно обрабатывать. Структурированные данные хранятся, как правило, с применением системы управления базой данных (DBMS).

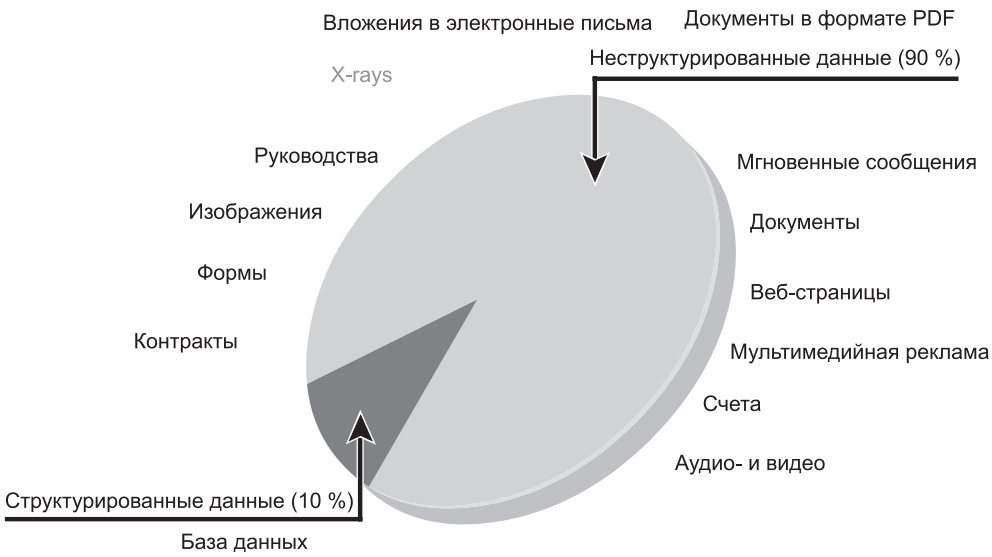


Рис. 1.2. Типы данных

Данные считаются неструктурированными, если их элементы не могут храниться в строках и столбцах, что затрудняет создание запросов и извлечение

данных приложениями предприятия. Например, контактные данные клиента могут храниться на различных носителях: на наклейках, визитках, в электронных сообщениях или даже файлах цифровых форматов — DOC, TXT или PDF. Поскольку данные не структурированы, их трудно извлекать с помощью обычного приложения, управляющего клиентскими базами. Подавляющее большинство создаваемых в наше время данных имеет неструктурированный характер. Для хранения неструктурированных данных, получаемых из многочисленных источников, управления этими данными, их анализа и извлечения из них ценной информации необходимы новые архитектурные решения, технологии, методы и знания.

1.1.3. Большие данные

Большими данными (Big Data) называется новая, развивающаяся концепция, относящаяся к наборам данных, объем которых выходит за пределы возможностей обычных программных средств по их сбору, хранению, управлению ими и обработке этих данных в приемлемые сроки. Они включают как структурированные, так и неструктурированные данные, происходящие из множества источников, в том числе транзакции бизнес-приложений, веб-страницы, видеоматериалы, изображения, сообщения электронной почты, данные социальной среды и т. д. Эти наборы данных обычно требуют сбора или обновления в реальном масштабе времени с целью анализа, предсказательного моделирования и принятия решений.

Для извлечения ценности из больших данных существует множество возможностей. Экосистема больших данных (рис. 1.3) состоит из следующих компонентов:

- устройств, производящих сбор данных из множества различных мест, а также генерирующих на их основе новые данные (метаданные);
- центров сбора данных, получающих эти данные от устройств и пользователей;
- агрегаторов данных, составляющих сводки о собранных данных с целью извлечения важной информации;
- пользователей и покупателей данных, извлекающих пользу из информации, собранной и обобщенной другими компонентами в цепочке создания ценных данных.

Традиционных IT-инфраструктур, инструментария и методологии недостаточно чтобы справиться с объемом, разнообразием, динамизмом и сложностью больших данных. Анализ больших данных в реальном масштабе времени требует применения новых технологий, архитектур и инструментария, обеспечивающих высокий уровень производительности, платформ данных с массовой параллельной обработкой (massively parallel processing, MPP) и аналитики наборов данных более высокого уровня.

Рис. 1.3. Экосистема больших данных

Наука о данных, или даталогия (data science), — развивающаяся дисциплина, позволяющая организациям извлекать из больших данных ценную бизнес-информацию. Даталогия представляет собой синтез нескольких существующих дисциплин, таких как статистика, математика, визуализация данных и информатика, позволяющий специалистам по обработке данных разрабатывать передовые алгоритмы с целью анализа огромного объема информации для управления новыми значениями и принятия дополнительных решений на основе имеющихся данных.

Предприятия, организации и учреждения, интересующиеся методиками анализа данных (медицинские и научно-исследовательские организации, организации здравоохранения, государственные и муниципальные учреждения, организации, выявляющие случаи мошенничества, общественные организации, банки, страховые компании и другие учреждения, работающие с цифровой информацией), извлекают пользу из анализа больших данных.

1.1.4. Информация

Структурированные или неструктурированные данные не соответствуют целям предприятий или частных лиц, если не представлены в смысловой форме. Предприятиям необходимо анализировать и выделять нужные

по смыслу данные. Информация — это сведения и знания, извлекаемые из данных.

Предприятия анализируют исходные данные для выявления значимых тенденций. Основываясь на этих тенденциях, компания может спланировать или изменить свой подход. Например, розничный продавец вычисляет предпочитаемые клиентом товары и торговые марки, проанализировав историю покупок и проведя инвентаризацию товара. Эффективный анализ данных не только приносит прибыль существующим предприятиям, но и создает потенциал для новых деловых возможностей, используя информацию в инновационных методах.

1.1.5. Хранение данных

Данные, создаваемые частными лицами или предприятиями, должны храниться так, чтобы они были легко доступны для дальнейшей обработки. В компьютерной среде устройства, разработанные для решения этой задачи, называются устройствами хранения данных, или просто хранилищами. Тип используемого хранилища зависит от типа данных и уровня их создания и применения. Карта памяти в мобильном телефоне или цифровой камере, DVD- и компакт-диски, а также жесткие диски — все это примеры устройств хранения данных.

Для предприятий существуют различные варианты хранения данных, включая встроенные жесткие диски, внешние дисковые массивы и ленты.

1.2. Эволюция архитектуры хранения данных

Изначально в вычислительных центрах организаций находились центральные компьютеры (главные компьютеры вычислительного центра) и устройства хранения информации (ленточные бобины и сборки магнитных дисков). Развитие открытых систем, доступность и простота их размещения дали возможность предприятиям и их отделам обзавестись собственными серверами и устройствами хранения данных. В ранних разработках открытых систем устройства хранения данных, как правило, были встроены в сервер. Эти хранилища не могли использоваться совместно с какими-либо другими серверами. Такой подход известен как архитектура, основанная на применении серверов (рис. 1.4, *а*). В данной архитектуре у каждого сервера имеется ограниченное количество устройств хранения данных, и любые действия администратора, например обслуживание сервера или наращивание объема хранилища, могут привести к недоступности информации. Рост количества серверов в отделах предприятия способствовал появлению незащищенных, неуправляемых и разобщенных информационных областей и увеличению текущих расходов.

Для решения этих проблем технология хранения данных прошла путь от архитектуры, основанной на сервер-центрической модели, до архитектуры, основанной на информационно-центрической модели (рис. 1.4, *б*). Эта архитектура предусматривает централизованное управление устройствами хранения данных и их независимость от серверов. Такие централизованно управляемые хранилища совместно используются сразу несколькими серверами. При развертывании нового сервера ему назначается устройство хранения из тех же самых совместно используемых хранилищ. Объем общего

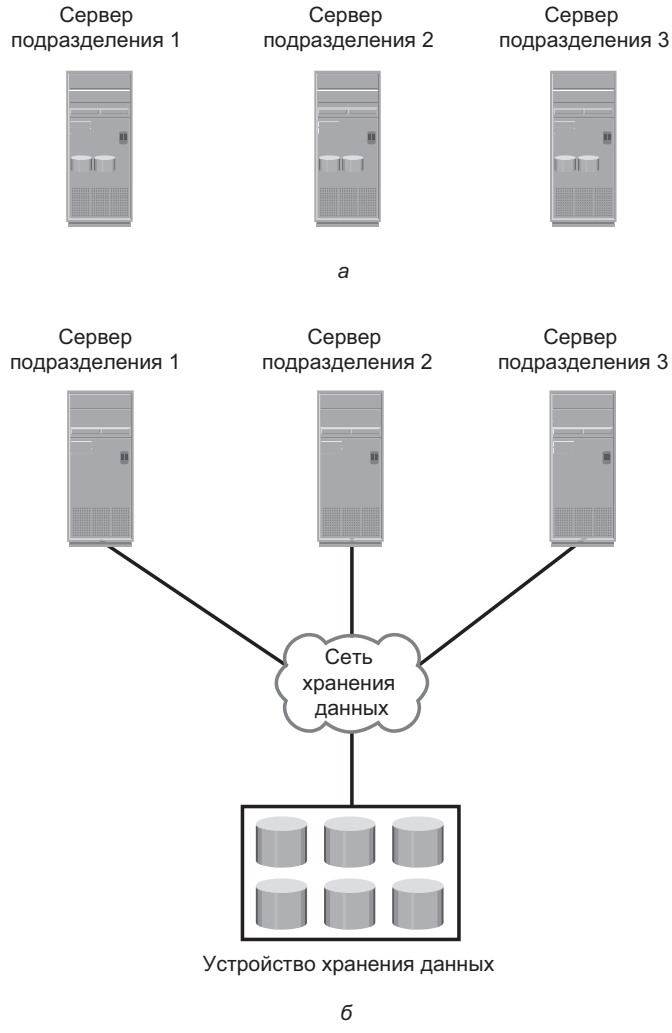


Рис. 1.4. Эволюция архитектуры хранения данных: *а* — архитектура на основе сервер-центрической модели; *б* — архитектура на основе информационно-центрической модели

хранилища может быть увеличен путем динамического добавления дополнительных устройств хранения данных без ущерба для доступности информации. Управление информацией в данной архитектуре осуществляется с меньшими организационными и финансовыми затратами.

Технология и архитектура хранения данных продолжают развиваться, что позволяет организациям объединять, защищать, оптимизировать и приводить в порядок свои данные для достижения максимальной отдачи от информационных активов.

1.3. Инфраструктура дата-центра

В организациях функционируют дата-центры для централизованной обработки информации всего предприятия. Эти центры хранят огромное количество данных и управляют ими. Инфраструктура дата-центра включает в себя аппаратные компоненты, такие как компьютеры, системы хранения данных, сетевые устройства и резервные источники питания, и программные компоненты, такие как приложения, операционные системы и управляющие программы. Они включают в себя также средства контроля состояния окружающей среды, такие как системы кондиционирования воздуха, противопожарное оборудование и системы вентиляции.

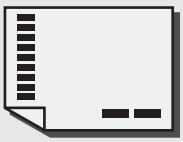
Крупные организации часто имеют несколько дата-центров для распределения нагрузки, связанной с обработкой данных, и обеспечения резервного копирования в аварийной ситуации.

1.3.1. Основные компоненты дата-центра

Перечислим пять основных компонентов, необходимых для функционирования информационного центра.

- **Приложение:** компьютерная программа, задающая логику вычислительных операций.
- **Система управления базами данных (DBMS):** обеспечивает структурированный способ хранения данных в логически организованных и взаимосвязанных таблицах.
- **Хост-система или главный компьютер:** вычислительная платформа (оборудование, программно-аппаратные средства и программное обеспечение), обеспечивающая работу приложений и баз данных.
- **Сеть:** канал обмена данными, упрощающий связь между различными устройствами, подключенными к нему.
- **Хранилище:** устройство длительного хранения данных для их последующего применения.

Как правило, эти основные компоненты рассматриваются и управляются как отдельные устройства, но для выполнения требований к обработке данных все элементы должны работать сообща.



В данной книге для представления компонента, запускающего приложения, такие понятия, как хост, главный компьютер и сервер, используются в качестве взаимозаменяемых.

На рис. 1.5 показан пример системы транзакций онлайн-заказов, включающей пять основных компонентов дата-центра и демонстрирующую их функциональные назначения в бизнес-процессе.

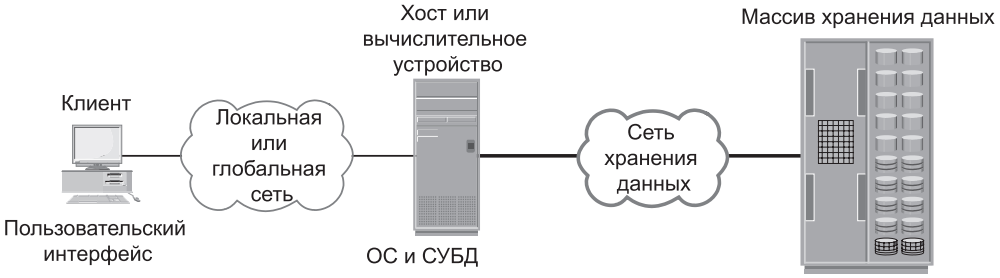


Рис. 1.5. Пример онлайновой системы транзакции заказа

Пользователь размещает заказ посредством клиентской машины, связанной локальной или глобальной сетью с хостом, на котором запущено приложение обработки заказа. Пользователь посредством приложения обращается к системе управления базами данных на хосте для предоставления информации, связанной с заказом, такой как имя пользователя, адрес, способ оплаты, заказываемые товары и заказываемое количество.

Система управления базами данных (DBMS) использует операционную систему хоста для записи этих данных на физические диски в хранилище. Сети хранения предоставляют линию связи между хостом и хранилищем и передают запрос на чтение или запись данных между ними. После получения запроса на чтение или запись от хоста хранилище выполняет операции, необходимые для сохранения данных на физических дисках.

1.3.2. Основные характеристики дата-центра

Бесперебойная работа дата-центров имеет решающее значение для безотказного и успешного ведения бизнеса. Организации должны обладать надежной инфраструктурой, обеспечивающей доступ к данным в любое время. Несмотря на то что требования, представленные на рис. 1.6, относятся ко всем компонентам инфраструктуры информационного центра, мы рассматриваем их относительно системы хранения данных. В данной книге рассматриваются различные технологии и решения, позволяющие соответствовать этим требованиям.

- **Доступность (Availability):** дата-центры должны обеспечивать доступность запрашиваемой информации. Ее недоступность может привести к бизнес-потерям в сферах финансовых служб, телекоммуникаций и электронной торговли в миллионы долларов в час.
- **Безопасность (Security):** в дата-центрах необходимо установить правила, процедуры и надлежащую интеграцию ключевых компонентов с целью предотвращения несанкционированного доступа к информации.
- **Масштабируемость (Scalability):** развитие компании часто требует развертывания большего количества серверов, установки новых приложений и дополнительных баз данных. Ресурсы дата-центра должны масштабироваться в соответствии с потребностями, при этом осуществление бизнес-операций не должно прерываться.

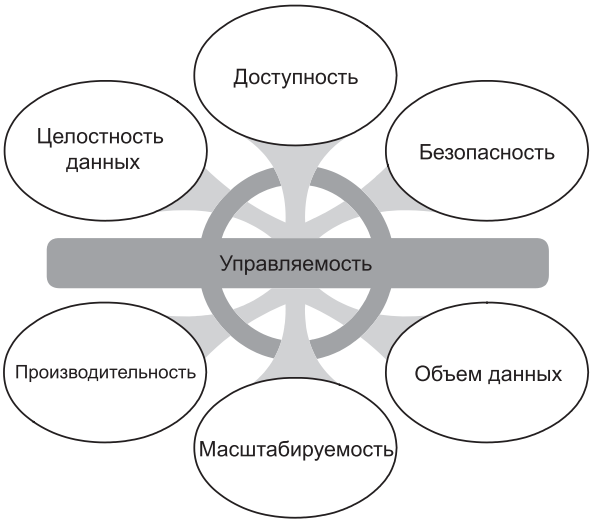


Рис. 1.6. Ключевые характеристики компонентов дата-центра

- **Производительность (Performance):** все основные компоненты дата-центра должны обеспечивать оптимальную производительность в соответствии с необходимыми уровнями обслуживания.
- **Целостность данных (Data integrity):** целостность данных означает применение таких механизмов, как коды коррекции ошибок или биты контроля четности, гарантирующих хранение и извлечение данных точно в таком же виде, в котором они были получены.
- **Объем(Capacity):** для эффективного хранения и обработки большого количества данных операции дата-центра требуют соответствующих ресурсов. При увеличении требований к объему дата-центр должен предоставлять дополнительный объем без ущерба для доступности данных или, в крайнем случае, с минимальным ущербом. Объемом можно управлять путем перераспределения существующих или добавления новых ресурсов.
- **Управляемость (Manageability):** дата-центр должен обеспечивать простое и интегрированное управление всеми своими компонентами. Хорошая управляемость может достигаться путем автоматизации и снижения роли человека (ручного управления) при выполнении стандартных заданий.

1.3.3. Управление дата-центром

Управление дата-центром включает в себя решение множества задач. К основным функциям управления можно отнести следующие.

- **Мониторинг (Monitoring)** — непрерывный сбор информации о различных компонентах и службах, запущенных в дата-центре. Мониторинг дата-центра проводится по следующим направлениям: безопасность, производительность, доступность и объем.
- **Составление отчетов (Reporting)** — периодически проводимая оценка производительности ресурса, его объема и загруженности. Это помогает оценить параметры бизнеса и выставить счета в соответствии с расходами, связанными с работой дата-центра.
- **Предоставление услуг (Provisioning)** — процесс обеспечения оборудованием, программами и другими ресурсами, необходимыми для работы дата-центра. В основном деятельность по предоставлению услуг включает в себя управление ресурсами с целью приведения их в соответствие требованиям по объему, доступности, производительности и безопасности.

Под влиянием виртуализации и облачных вычислений произошло существенное изменение способов предоставления ресурсов инфраструктуры дата-центра и управления ими. Чтобы решить проблемы оптимизации использования различных компонентов дата-центров, организации

наращивают темпы их виртуализации. Кроме того, постоянное ценовое давление в области информационных технологий и востребованность обработки данных по запросу привели к внедрению облачных вычислений.

1.4. Виртуализация и облачные вычисления

Виртуализация представляет собой технологию абстракции таких физических ресурсов, как средства вычисления, хранения, сети передачи данных и превращения этих ресурсов в логические. Виртуализация существовала в индустрии информационных технологий на протяжении многих лет в разнообразных формах. Общеизвестными примерами виртуализации могут послужить виртуальная память, используемая в вычислительных системах, и разбиение цельных дисков на разделы.

Виртуализация позволяет создавать пулы физических ресурсов и предоставлять обзор совокупных возможностей физических ресурсов. Например, виртуализация устройств хранения данных позволяет нескольким объединенным в пул устройствам появляться в виде единого большого логического объекта. Аналогично этому при виртуализации вычислительных устройств возможности виртуализированного центрального процессора объединенных в пул физических серверов могут рассматриваться как совокупность мощностей всех центральных процессоров (выражается в мегагерцах). Виртуализация также позволяет централизованно управлять объединенными в пул ресурсами.

Виртуальные ресурсы могут создаваться и предоставляться из объединенных в пулы физических устройств. Например, виртуальный диск заданного объема может быть создан из пула устройств хранения данных, а виртуальный сервер с центральным процессором конкретной мощности и заданным объемом памяти может быть сконфигурирован из пула вычислительных устройств. Такие виртуальные ресурсы совместно используют объединенные в пул физические ресурсы, благодаря чему улучшаются эксплуатационные показатели физических IT-ресурсов. На основе бизнес-требований виртуальные ресурсы могут наращиваться или сокращаться без какого-либо ущерба для приложений или пользователей. За счет улучшения условий эксплуатации IT-ресурсов организации могут экономить средства на закупку новых физических ресурсов и управление ими. Кроме того, уменьшение объемов физических ресурсов влечет за собой экономию производственных площадей и электроэнергии, что улучшает экономические и экологические показатели производства вычислений.

В современных условиях динамично изменяющейся обстановки и высокой конкуренции организации должны проявлять гибкость и умение подстраиваться под меняющиеся рыночные требования. Это приводит к быстрому расширению и обновлению ресурсов наряду с соблюдением условий по сокращению или сохранению на прежнем уровне расходов на IT-технологии.

Эффективному решению этих задач способствуют облачные вычисления, позволяющие физическим или юридическим лицам пользоваться IT-ресурсами как услугой по сети. Тем самым достигается высокая степень масштабируемости и гибкости вычислений, позволяющая предоставлять ресурсы по мере их востребования. Пользователи могут наращивать или снижать потребности в вычислительных ресурсах, включая емкость хранилища данных, с минимальными усилиями по управлению или взаимодействию с поставщиком услуг. Облачные вычисления позволяют отправлять самообслуживаемые запросы благодаря полностью автоматизированному процессу их выполнения.

Облачные вычисления позволяют учитывать объем потребления, благодаря чему потребители платят только за использованные ресурсы, например за время использования центрального процессора, объем переданных данных и количество гигабайт сохраненных данных.

Облачная инфраструктура обычно выстраивается на основе виртуализированных дата-центров, обеспечивающих объединение ресурсов в пулы и их быстрое предоставление. Более подробно хранение информации в виртуализированной и облачной среде будет рассмотрено в данной книге чуть позже.

Резюме

В данной главе рассмотрена важная роль данных, информации и инфраструктуры хранения данных в современных условиях. Использование современных устройств хранения данных должно начинаться с изучения типа данных, их ценности, а также основных свойств дата-центра.

Развитие архитектуры устройств хранения данных и основные компоненты дата-центра, рассмотренные здесь, закладывают основу изучения хранения информации и управления этим процессом. Появление виртуализации позволило превратить классические дата-центры в виртуализированные. Дальнейшие изменения в способах предоставления и потребления IT-ресурсов связаны с облачными вычислениями.

В следующих главах книги рассмотрены подробности различных аспектов хранения информации и управления этим процессом как в классической, так и в виртуализированной средах. Сначала поговорим об основных компонентах дата-центра, причем особое внимание будет уделено системам хранения данных и RAID-массивам (в главах 2–4). В главах 5–8 будут подробно рассмотрены различные сетевые технологии хранения данных, такие как сеть хранения данных — storage area network (SAN), сетевое хранилище — network attached storage (NAS) и объектно-ориентированное и унифицированное хранилища. В главах 9–12 разговор пойдет о различных решениях, предназначенных для обеспечения непрерывности бизнес-процессов, таких как резервное копирование и репликация, а также технологиях архивирования. В главе 13 будут представлены облачная инфраструктура и соответствующие

ей службы. В главах 14 и 15 будут рассмотрены вопросы обеспечения безопасности и управления хранилищами данных в традиционной и виртуализированной средах.

УПРАЖНЕНИЯ

1. Что такое структурированные и неструктурированные данные? Исследуйте проблему хранения неструктурированных данных и управления ими.
2. Проанализируйте преимущества архитектуры хранилища, основанного на информационно-центрической модели, над архитектурой хранилища, основанного на сервер-центрической модели.
3. В чем заключаются характерные особенности больших данных? Проведите исследование больших данных и подготовьте презентацию, основанную на анализе их особенностей.
4. Исследуйте вопросы использования предприятиями своих информационных ресурсов для получения конкурентных преимуществ и новых бизнес-возможностей.
5. Исследуйте вопросы управления персональными данными и подготовьте соответствующую презентацию.

Глава 2
Среда дата-центра

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Приложение, система управления базами данных, хост, система передачи данных и хранилище
- Виртуализация приложений
- Файловая система и диспетчер томов
- Виртуализация вычислительной системы, рабочего стола и памяти
- Носители данных
- Компоненты дискового накопителя
- Зональная побитовая запись
- Адресация логических блоков
- Флеш-накопители

Сегодня дата-центры стали важной и неотъемлемой частью любого бизнеса, каким бы он ни был — малым, средним или большим. Основными компонентами дата-центра являются централизованно управляемые хост, хранилище данных, система передачи данных (или сеть), приложения и система управления базами данных (СУБД). Эти компоненты осуществляют совместную обработку и хранение данных. С развитием виртуализации классические дата-центры превратились в виртуализированные — virtualized data center (VDC). В VDC физические ресурсы из классического дата-центра объединены в пулы и представлены в виде виртуальных ресурсов. Такая абстракция скрывает от пользователя сложность физических ресурсов и имеющиеся у них ограничения. Путем объединения IT-ресурсов с помощью виртуализации организации могут оптимизировать использование своей инфраструктуры и снизить совокупные расходы на ее эксплуатацию. Кроме того, в VDC виртуальные ресурсы создаются с помощью программных средств, допускающих более быстрое развертывание по сравнению с развертыванием физических ресурсов в классических дата-центрах. В этой главе будут рассмотрены все основные компоненты дата-центра, а также виртуализация вычислительной системы, памяти, рабочего стола и приложений. Виртуализация хранилища данных и сети рассматривается в следующих главах книги.

С повышением важности информационных ресурсов для ведения бизнеса хранилище данных, являющееся одним из основных компонентов дата-центра, становится особым ресурсом дата-центра. Процессу ввода хранилищ в эксплуатацию и управлению ими следует уделять особое внимание. В данной главе будут рассмотрены также подсистемы хранилища и характерные особенности его компонентов, конфигурации и параметры производительности дискового накопителя. Кроме того, будет рассмотрено соединение между хостом и хранилищем, обеспечиваемое различными технологическими средствами.

2.1. Приложение

Приложение — это компьютерная программа, предоставляющая логику вычислительных операций. Приложение отправляет запросы к операционной системе, в среде которой оно запущено, с целью осуществления операций чтения-записи на устройствах хранения данных. Приложения могут являться надстройками над базами данных, которые в свою очередь используют для выполнения операций чтения-записи на устройствах хранения данных службы операционной системы. Обычно к приложениям, развернутым в среде дата-центра, относятся бизнес-приложения, приложения управления инфраструктурой, приложения защиты данных и приложения обеспечения безопасности. К числу этих приложений можно отнести программы электронной почты, системы планирования ресурсов предприятия (enterprise resource planning, ERP), системы поддержки принятия решений (decision support system, DSS), системы управления ресурсами, системы резервного копирования, аутентификации, антивирусные программы и т. д.

ВИРТУАЛИЗАЦИЯ ПРИЛОЖЕНИЙ

Виртуализация приложений устраняет зависимость приложения от базовой платформы (операционной системы и оборудования). Виртуализация инкапсулирует приложение и запрашивает ресурсы операционной системы внутри виртуализированного контейнера. Эта технология позволяет развертывать приложения, не внося каких-либо изменений в базовую операционную систему, файловую систему или реестр вычислительной платформы, на которой они развернуты. Поскольку виртуализированные приложения работают в изолированной среде, базовая операционная система и другие приложения защищены от возможности повреждений. Существует множество сценариев, при которых возможно возникновение конфликтов, если на одной и той же вычислительной платформе установлены несколько приложений или несколько версий одного и того же приложения. Виртуализация приложений исключает возникновение подобных конфликтов путем изоляции различных версий приложения и связанных с ними ресурсов операционной системы.

Общая производительность системы хранения данных и конструкторские решения при ее создании выбираются на основе характеристик ввода-вывода, задаваемых приложением. Дополнительную информацию о характеристиках ввода-вывода приложений можно найти в приложении А.

2.2. Система управления базами данных

База данных является структурированным способом хранения данных в логически организованных взаимосвязанных таблицах. База данных помогает оптимизировать хранение и извлечение данных. Системы управления базами данных (СУБД) управляет созданием, обслуживанием и использованием баз данных. СУБД обрабатывает запросы приложений, связанных с данными, и дает указание операционной системе на перенос соответствующих данных из хранилища.

2.3. Хост (вычислительная система)

Пользователи хранят и извлекают данные с помощью приложений. Компьютеры, на которых запускаются эти приложения, называются *хостами*, или *вычислительными системами*. Хостами могут быть физические или виртуальные машины. Программы виртуализации вычислительных систем позволяют создавать виртуальные машины, являющиеся надстройками над инфраструктурой физической вычислительной системы. Виртуализация вычислительных систем и виртуальные машины будут рассмотрены в данной главе чуть позже. В качестве примеров физических хостов можно привести настольные компьютеры, серверы или кластеры серверов, ноутбуки и мобильные устройства. Хост состоит из центрального процессора, памяти, устройств ввода-вывода и набора программного обеспечения для выполнения вычислительных операций. Это программное обеспечение включает в себя операционную систему, файловую систему, диспетчер логических томов, драйверы устройств и т. д. и может быть установлено в качестве отдельных компонентов или как часть операционной системы. Центральный процессор состоит из четырех компонентов: арифметико-логического устройства (АЛУ — Arithmetic Logic Unit (ALU), блока управления, регистров и кэш-памяти первого уровня (L1). На хосте бывает два типа памяти: оперативная память — Random Access Memory (RAM) и постоянное запоминающее устройство — Read-Only Memory (ROM). Связь с хостом обеспечивают устройства ввода-вывода. Примерами устройств ввода-вывода могут послужить клавиатура, мышь, монитор и т. д.

Программное обеспечение запускается на хосте и обеспечивает обработку входных и выходных данных. Более подробно различные компоненты

программного обеспечения, являющиеся неотъемлемой частью хост-системы, будут рассмотрены в следующих разделах.

2.3.1. Операционная система

В обычной компьютерной среде *операционная система* управляет всеми аспектами вычислений. Она работает в качестве посредника между приложением и физическими компонентами вычислительной системы. Одна из функций операционной системы для приложений заключается в обеспечении доступа к данным. Операционная система также отслеживает действия пользователя и состояние среды и реагирует на них. Она планирует и контролирует работу компонентов оборудования и управляет распределением их ресурсов. Она обеспечивает проведение основных мер безопасности при доступе ко всем управляемым ресурсам и при их использовании. Операционная система также выполняет основные задачи по управлению системой хранения данных при работе с другими ключевыми компонентами, такими как файловая система, диспетчер томов и драйверы устройств.

В виртуализированной среде вычисления уровень виртуализации является посредником между операционной системой и ресурсами оборудования. Здесь операционная система может работать по-разному в зависимости от типа реализованной виртуализации вычислительной системы. При обычной виртуализации операционная система работает в качестве гостевой и выполняет только действия по взаимодействию с приложением. В этом случае функции управления оборудованием возлагаются на уровень виртуализации.

Виртуализация памяти

Память была и остается весьма дорогостоящим компонентом хоста. Ее объем определяет размер и количество приложений, запускаемых на хосте. *Виртуализация памяти* позволяет нескольким приложениям и процессам, чьи совокупные потребности в памяти превышают объем доступной физической памяти, запускаться на хосте, не оказывая отрицательного воздействия друг на друга.

Виртуализация памяти является свойством операционной системы, позволяющим виртуализировать физическую (оперативную) память хоста. Операционная система создает виртуальную память с адресным пространством, превышающим объем имеющейся в вычислительной системе физической памяти. Виртуальная память включает в себя адресное пространство физической памяти и часть дискового хранилища. Утилита операционной системы, управляющая виртуальной памятью, известна как диспетчер виртуальной памяти — virtual memory manager (VMM). VMM управляет отображением виртуальной памяти на физическую и извлекает данные из дискового хранилища, когда процесс обращается к виртуальному адресу, указывающему на данные, находящиеся в дисковом хранилище. Память, используемая VMM

на диске, известна как своп-пространство. Это пространство (также называемое страничным файлом, или своп-файлом) является частью дискового накопителя, представляемой операционной системе в качестве физической памяти.

При реализации виртуальной памяти системная память разбивается на непрерывные блоки, состоящие из страниц фиксированного размера. Процесс, известный как замещение страниц, сбрасывает неактивные страницы физической памяти в своп-файл и возвращает их в физическую память по мере надобности. Это позволяет эффективно использовать объем доступной физической памяти сразу несколькими различными приложениями. Операционная система обычно сбрасывает в своп-файл наименее востребованные страницы, чтобы наиболее активные процессы не испытывали дефицита оперативной памяти. Доступ к страницам своп-файла осуществляется медленнее, чем доступ к физической памяти, поскольку эти страницы размещены на дисковом накопителе, который работает медленнее физической памяти.

2.3.2. Драйвер устройства

Драйвер устройства — это специальная программа, позволяющая операционной системе взаимодействовать с определенным устройством, например с принтером, мышью или жестким диском. Драйвер устройства позволяет операционной системе опознавать устройство и получать доступ к устройствам и к управлению ими. Драйверы устройств являются аппаратно-зависимыми программами, учитывающими специфику операционной системы.

2.3.3. Диспетчер томов

В прежние времена дисковые накопители были видны операционной системе как некоторое количество непрерывных дисковых блоков. Файловой системе или другим информационным объектам, используемым операционной системой или приложением, должен был выделяться весь диск. Недостатком такого подхода было отсутствие гибкости. Когда исчерпывался объем дискового накопителя, не было простого способа расширить объем файловой системы. Кроме того, по мере увеличения емкости дискового накопителя выделение всего пространства диска файловой системе зачастую приводило к недоиспользованию этой емкости.

Появление *диспетчеров логических томов* — Logical Volume Managers (LVM) позволило осуществлять динамическое расширение объема файловой системы и эффективно управлять системой хранения данных. Диспетчер логических томов представляет собой запускаемую на вычислительной системе программу управления логической и физической системами хранения данных и является дополнительной прослойкой между файловой системой и физическим диском. Диспетчер может разбить диск большого

объема на несколько разделов, создав виртуальные тома меньшего объема (этот процесс называется *разбиением на разделы*), или объединить несколько дисков меньшего объема, сформировав более крупный виртуальный том (этот процесс называется объединением, или конкатенацией). Затем созданные тома показываются приложениям.

Разбиение диска на разделы было введено с целью повышения гибкости и оптимизации использования дискового накопителя. При разбиении на разделы жесткий диск разделяется на логические контейнеры, называемые логическими томами — logical volumes (LV) (рис. 2.1). Например, физический диск, имеющий большой объем, может быть разбит на несколько логических томов для обслуживания данных в соответствии с требованиями файловой системы и приложений. Разделы создаются из групп смежных цилиндров при первоначальной установке жесткого диска на хосте. Файловая система хоста получает доступ к логическим томам, даже не подозревая о том, что диск разбит на разделы и имеется некая физическая структура диска.

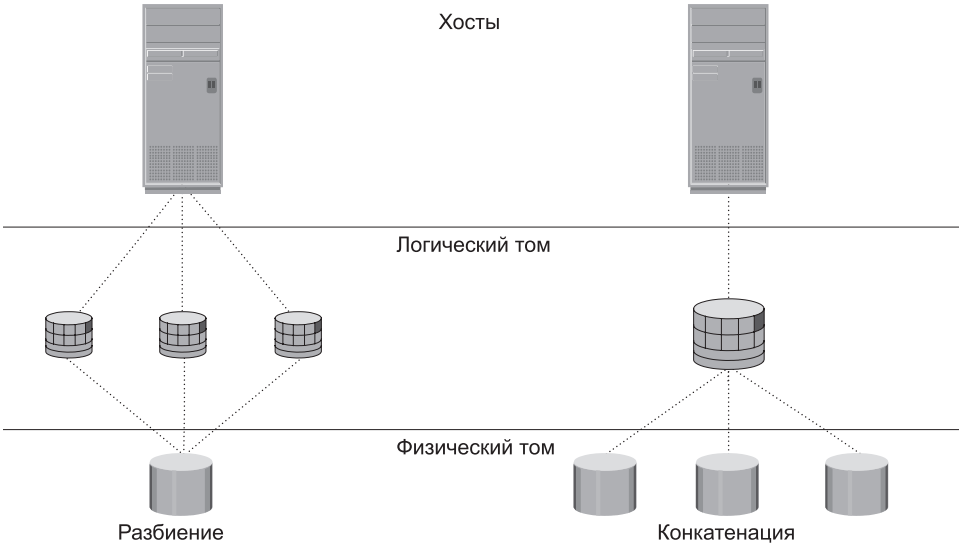


Рис. 2.1. Разбиение диска и конкатенация дисков

Конкатенация представляет собой процесс объединения в группу нескольких физических дисковых накопителей и представления их хосту в виде одного большого логического тома (см. рис. 2.1).

Диспетчер логических томов позволяет оптимизировать обращение к хранилищу и упростить управление его ресурсами. Он скрывает детали физического диска и размещения данных на нем, позволяя администраторам вносить изменения в распределение пространства хранилища даже в процессе выполнения приложения.

Основными компонентами диспетчера логических томов являются физические тома, группы томов и логические тома. Согласно LVM-терминологии, каждый физический диск, подключенный к хост-системе, является физическим томом — physical volume (PV). LVM преобразует физическую систему хранения, представленную физическими томами, в логическое представление хранилища, которое затем используется операционной системой и приложениями. Группа томов создается путем объединения одного или нескольких физических томов. При инициализации каждого физического тома для его использования диспетчером логических томов ему назначается *уникальный идентификатор физического тома* — physical volume identifier (PVID). Физические тома из группы томов можно в динамическом режиме добавлять или удалять. Они не могут совместно использоваться разными группами томов, то есть частью группы томов становится весь физический диск. При создании группы томов каждый физический том разбивается на блоки данных одинакового размера, называемые физическими экстентами.

Логические тома создаются в пределах заданной группы томов. Логический том можно представить в виде раздела диска, а сама группа томов может быть представлена как диск. Группа томов может иметь несколько логических томов. Размер логического тома складывается из нескольких физических экстентов. Для операционной системы логический том представляется физическим устройством. Логический том может быть составлен из несмежных физических экстентов и может охватывать несколько физических томов. Файловая система создается в логическом томе. Затем такие логические тома назначаются приложению. Для повышения доступности данных логический том может быть зазеркалирован.

2.3.4. Файловая система

Файл — это совокупность взаимосвязанных записей или данных, хранящихся в виде единого целого, обозначенного именем. *Файловая система* — это иерархическая структура файлов. Файловые системы обеспечивают свободный доступ к файлам данных, расположенным на дисковом накопителе, в разделе диска или логическом томе. Файловая система состоит из логических структур и программных процедур, управляющих доступом к файлам. Она предоставляет пользователям функциональные возможности по созданию, изменению, удалению файлов и получению доступа к ним. Доступом к файлам на дисках управляют права доступа, назначенные файлу его владельцем, что также регулируется файловой системой.

Файловая система организует данные в структурированном иерархическом порядке посредством каталогов, которые представляют собой контейнеры для хранения указателей на несколько файлов. Все файловые системы

обслуживают карту указателей на каталоги, подкаталоги и файлы, являющиеся частью файловой системы. Среди распространенных файловых систем можно назвать следующие:

- FAT 32 (FAT — File Allocation Table, таблица размещения файлов) для Microsoft Windows;
- NT File System (NTFS) для Microsoft Windows;
- UNIX File System (UFS) для UNIX;
- Extended File System (EXT2/3) для Linux.

Помимо файлов и каталогов файловая система содержит также ряд других связанных записей с обобщенным названием *метаданные*. Например, метаданные в среде UNIX состоят из *суперблока*, *инодов* и списка свободных и используемых блоков данных. Чтобы считаться полезными, метаданные должны быть согласованы со своей файловой системой.

Суперблок содержит важную информацию о файловой системе, например о ее типе, дате создания и изменения, размере и формате, а также расположении. Кроме того, он содержит сведения о количестве доступных ресурсов (например, о количестве свободных блоков, инодов и т. д.) и флаг, указывающий на состояние установки файловой системы. Инод связан с каждым файлом и каталогом и содержит информацию о длине файла, владельце, правах доступа, времени последнего обращения к файлу и его изменения, количестве ссылок и адресов данных.

Блок файловой системы является наименьшим «структурным элементом», выделяемым для хранения данных. Каждый блок файловой системы представляет собой непрерывную область на физическом диске. Размер блока файловой системы фиксируется во время ее создания. Размер файловой системы зависит от размера блока и общего количества содержащихся в ней блоков. Файл может занимать несколько блоков файловой системы, поскольку основное количество файлов больше по размеру предопределенного блока файловой системы. При удалении блоков или добавлении новых блоков файлы утрачивают непрерывность и становятся фрагментированными. Со временем с увеличением размеров файлов файловая система становится все более фрагментированной.

Процесс отображения пользовательских файлов на подсистему дискового хранилища, производимый диспетчером логических томов, проходит в следующем порядке (рис. 2.2).

1. Пользователи и приложения создают файлы и управляют ими.
2. Эти файлы размещаются в файловых системах.
3. Файловые системы отображаются на блоки файловых систем.
4. Блоки файловых систем отображаются на логические экстенты логического тома.

5. Логические экстенты в свою очередь отображаются на физические экстенты диска либо операционной системой, либо диспетчером логических томов.
6. Физические экстенты отображаются в подсистеме хранилища на сектора диска.

При отсутствии диспетчера логических томов отсутствуют и логические экстенты. Без LVM блоки файловой системы отображаются на сектора диска напрямую.

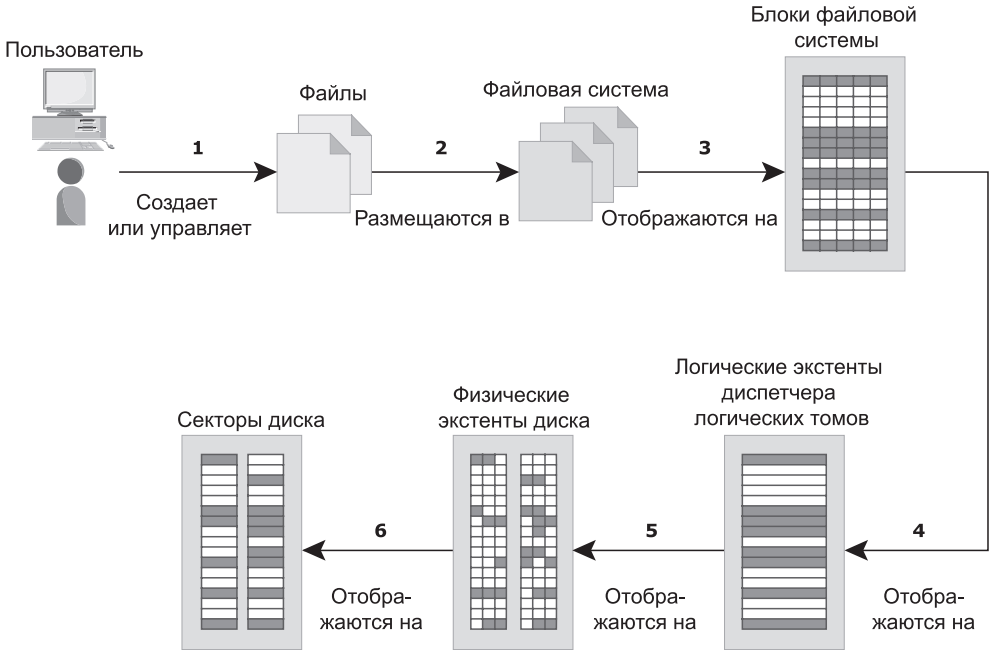
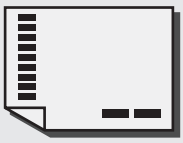


Рис. 2.2. Процесс отображения пользовательских файлов на подсистему дискового хранилища

Дерево файловой системы начинается с *корневого каталога*. У корневого каталога имеется ряд подкаталогов. Перед использованием файловая система должна быть установлена.

Файловая система может быть журналируемой или нежурналируемой. *Нежурналируемые файловые системы* предрасположены к потенциальной потере файлов, поскольку для обновления своих данных и метаданных они используют разные операции записи. Если в ходе процесса записи система дает сбой, метаданные или данные могут быть утрачены или повреждены. После перезагрузки системы файловая система пытается обновить структуры метаданных путем их обследования и восстановления. На файловых системах больших размеров данная операция занимает много времени. Если для воссоздания желаемой или первоначальной структуры информации

недостаточно, файлы могут быть потеряны или утрачены, что приведет к повреждению файловых систем.



Для проверки согласованности файловой системы на хостах под управлением UNIX и Linux запускается системная утилита fsck. К примеру, файловая система может приобрести несогласованное состояние при незавершенных изменениях, когда компьютерная система дала сбой до того, как изменения были зафиксированы на диске. Для успешного проведения начальной загрузки системы выдается команда **fsck, в результате чего сначала проверяется согласованность файловых систем**. Если файловые системы находятся в согласованном состоянии, утилита проверяет согласованность всех остальных файловых систем. Если какая-либо файловая система оказывается несогласованной, она не устанавливается. Несогласованная файловая система может быть автоматически исправлена утилитой fsck или может потребовать пользовательского вмешательства для подтверждения корректирующих действий. В операционных системах DOS, OS/2 и Microsoft Windows используется команда CHKDSK.

В *журналируемой файловой системе* используется отдельная область, называемая *логом*, или *журналом*. Этот журнал может содержать все записываемые данные (физический журнал) или только обновляемые метаданные (логический журнал). Перед внесением изменений в файловую систему они записываются в эту отдельную область. После обновления журнала может быть выполнена операция в файловой системе. Если система даст сбой в ходе операции, в журнале будет достаточно информации для «воспроизведения» журнальной записи и завершения операции. Журналирование позволяет проводить быструю проверку файловой системы, поскольку просматриваются только активные части большой файловой системы, к которым было самое последнее обращение. Кроме того, благодаря сохранению информации о незавершенной операции снижается риск потери файлов.

Недостаток журналируемых файловых систем заключается в том, что они работают медленнее других файловых систем. Это замедление обусловлено дополнительными операциями, которые нужно выполнять с журналом при каждом изменении файловой системы. Но существенное сокращение времени проверок файловой системы и ее целостность, предоставляемые журналированием, перевешивают недостатки такой системы.

Чтобы управлять большим количеством файлов по сети и совместно использовать эти файлы, нужно устанавливать выделенные файловые серверы. Такие серверы поддерживают несколько файловых систем и используют протоколы совместного использования файлов, характерные для той или иной операционной системы, например NFS и CIFS. Более подробно эти протоколы рассмотрены в главе 7.

2.3.5. Виртуализация вычислительных устройств

Виртуализация вычислительных устройств — это технология, предназначенная для маскировки или абстрагирования физического оборудования от операционной системы. Она позволяет нескольким операционным системам одновременно работать на отдельных или кластеризированных физических машинах. Эта технология дает возможность создавать переносимые виртуальные компьютерные системы, называемые *виртуальными машинами* (VM). Каждая VM запускает операционную систему и экземпляр приложения в изолированной среде. Виртуализация вычислительных устройств достигается с помощью уровня виртуализации, находящегося между оборудованием и виртуальными машинами, называемого *гипервизором*. Гипервизор предоставляет ресурсы оборудования, такие как центральный процессор, память и сеть, всем виртуальным машинам. В одном физическом сервере в зависимости от возможностей его оборудования может быть создано большое количество виртуальных машин.

Виртуальная машина является логическим объектом, который виден операционной системе как физический хост, имеющий свой собственный центральный процессор, память, сетевой контроллер и диски. Но все виртуальные машины совместно используют одно и то же базовое физическое оборудование в изолированном виде. С точки зрения гипервизора виртуальные машины являются отдельными наборами файлов, включающими файл конфигурации виртуальной машины, файлы данных и т. д.

Физический сервер часто сталкивается с проблемами конфликта ресурсов, когда два или более приложения, запущенных на сервере, предъявляют конфликтующие запросы. Например, приложениям могут понадобиться различные значения в одной и той же записи реестра, разные версии одной и той же DLL-библиотеки и т. д. Эти проблемы усугубляются при наличии у приложений требований по высокой степени доступности. В результате серверы могут одновременно обслуживать только одно приложение (рис. 2.3, *а*). Это вынуждает организации приобретать новые физические машины для каждого развертываемого ими приложения, что приводит к появлению дорогостоящей и неспособной к гибкой настройке инфраструктуры. В то же время многие приложения не используют возможности доступного им оборудования в полном объеме. В результате этого такие ресурсы, как центральный процессор, память и хранилище данных, остаются недоиспользованными. Виртуализация вычислительных устройств дает возможность пользователям преодолеть эти проблемы (рис. 2.3, *б*), позволяя нескольким операционным системам и приложениям запускаться на одной физической машине. Эта технология существенно повышает эффективность использования серверов и обеспечивает их консолидацию.

Серверная консолидация позволяет организациям вводить в эксплуатацию дата-центры с меньшим количеством серверов. Это, в свою очередь, позволяет

Рис. 2.3. Виртуализация сервера: *а* — до виртуализации вычислительного устройства; *б* — после виртуализации вычислительного устройства

ВИРТУАЛИЗАЦИЯ РАБОЧИХ СТОЛОВ

При использовании традиционного рабочего стола операционная система, приложения и профили пользователей привязаны к конкретной части оборудования. С прежними рабочими столами продуктивность бизнеса сильно зависела от сбоя или утраты клиентского устройства. Виртуализация рабочих столов исключает зависимость операционной системы, приложений, профилей пользователей и настроек от оборудования. Это позволяет IT-специалистам вносить изменения, проводить обновления и развертывания этих составляющих независимо друг от друга. Рабочие столы, имеющиеся в дата-центре, работают на виртуальных машинах, а пользователи получают к этим рабочим столам удаленный доступ с разнообразных клиентских устройств, таких как ноутбуки, настольные компьютеры и мобильные устройства (которые также называют тонкими устройствами — Thin devices). Выполнение приложений и хранение данных осуществляются централизованно в дата-центре, а не на клиентских устройствах. Поскольку рабочие столы запускаются как виртуальные машины в дата-центре организации, уменьшается риск утечки или хищения данных. Это также способствует выполнению централизованного резервного копирования и упрощает процедуры обеспечения совместимости. Виртуальные рабочие столы проще обслуживать, поскольку к ним легче применять доработки, на них легче развертывать новые приложения и операционные системы, а также централизованно изменять политики или удалять пользователей.

сэкономить на приобретении нового сервера, снижает эксплуатационные расходы и экономит площадь, занимаемую дата-центром, и пространство в аппаратных стойках. По сравнению с установкой физического сервера, на создание виртуальных машин затрачивается меньше времени, благодаря чему организации могут упростить и ускорить предоставление серверов.

Отдельные виртуальные машины могут быть перезапущены, обновлены или дать сбой, не оказывая при этом никакого влияния на другие виртуальные машины, работающие на той же физической машине. Кроме того, виртуальные машины могут быть скопированы или перемещены с одной физической машины на другую без простоев приложений. Неразрушающая миграция виртуальных машин нужна для балансировки нагрузки среди физических машин, обслуживания оборудования и обеспечения более высокой доступности.

2.4. Соединение

Соединение рассматривается как взаимное соединение хостов и таких периферийных устройств, как принтеры или устройства хранения данных. В данной книге внимание уделяется только соединению между хостами и устройством хранения данных. Соединение и передача информации между хостом и хранилищем обеспечиваются с помощью физических компонентов и протоколов обмена данными.

2.4.1. Физические компоненты соединения

К физическим компонентам соединения относятся элементы оборудования, соединяющие хост и хранилище. Существуют три таких компонента: интерфейсное устройство хоста, порт и кабель (рис. 2.4).

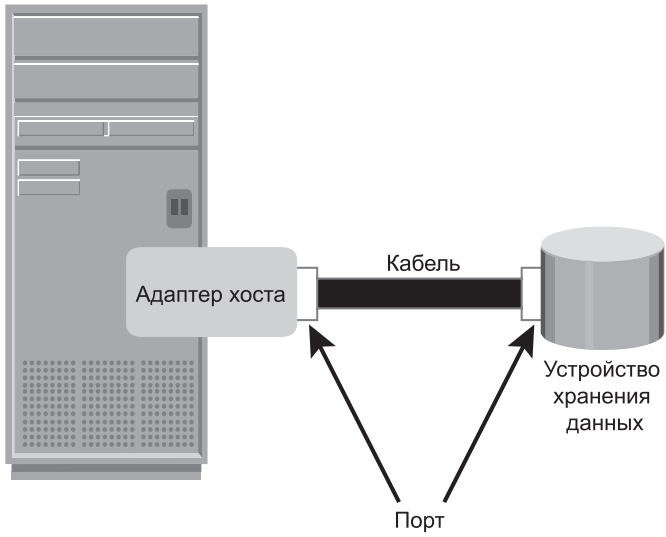


Рис. 2.4. Физические компоненты обеспечения связи

Интерфейсное устройство хоста, или адаптер хоста, подключает хост к другим хостам и устройствам хранения данных. В качестве примеров интерфейсных устройств хоста можно привести шинный адаптер хоста — host bus adapter (HBA) и сетевую интерфейсную плату — network interface card (NIC). Шинный адаптер хоста представляет собой плату со специализированной интегральной схемой — application-specific integrated circuit (ASIC), выполняющую функции интерфейса ввода-вывода между хостом и устройством хранения, освобождая при этом центральный процессор от дополнительной вычислительной нагрузки, связанной с вводом-выводом. У хоста обычно имеется несколько HBA-адаптеров.

Порт — это специализированный разъем, обеспечивающий соединение хоста с внешними устройствами. Для связи хоста с устройством хранения данных у HBA-адаптера может быть один или несколько портов. Для подключения хостов к внутренним или внешним устройствам используются медные или оптоволоконные кабели.

2.4.2. Протоколы обмена данными

Протокол делает возможным обмен данными между хостом и хранилищем. Протоколы реализуются с помощью интерфейсных устройств (или контроллеров) как на источнике, так и на приемнике данных. Среди популярных протоколов, используемых для обмена данными между хостом и хранилищем, можно назвать протоколы Integrated Device Electronics/Advanced Technology Attachment (IDE/ATA), Small Computer System Interface (SCSI), Fibre Channel (FC) и Internet Protocol (IP).

IDE/ATA и Serial ATA

IDE/ATA является популярным стандартом интерфейсного протокола, используемым для подключения устройств хранения данных, таких как дисковые накопители и приводы компакт-дисков. Этот протокол поддерживает параллельную передачу данных и поэтому называется Parallel ATA (PATA) или просто ATA. У протокола IDE/ATA имеется большое разнообразие стандартов и названий. Версия ATA под названием Ultra DMA/133 поддерживает скорость передачи данных до 133 Мбит/с. В конфигурации «главный — подчиненный» (master — slave) интерфейс ATA поддерживает два устройства хранения для каждого подключения. Но если важна производительность устройства, совместное использование порта двумя устройствами не рекомендуется.

Версия этого протокола с последовательной передачей данных поддерживает побитовую последовательную передачу и известна как Serial ATA (SATA). В самых новых системах SATA в основном заменил PATA благодаря своей высокой производительности и низкой стоимости. SATA версии 3.0 обеспечивает скорость передачи данных до 6 Гбит/с.

SCSI и Serial SCSI

Протокол SCSI появился в качестве наиболее предпочтительного протокола соединения в компьютерах высшего класса. Этот протокол поддерживает параллельную передачу данных и предлагает по сравнению с ATA повышенную производительность, масштабируемость и совместимость. Но высокая стоимость SCSI-устройств ограничивает его популярность среди пользователей домашних или персональных настольных компьютеров. Со временем протокол SCSI был усовершенствован и теперь включает широкий спектр связанных с ним технологий и стандартов. SCSI поддерживает до 16 устройств на одной шине и обеспечивает скорость передачи данных до 640 Мбит/с (в версии Ultra-640).

Версия протокола с последовательной передачей данных — Serial attached SCSI (SAS) является двухточечным последовательным протоколом, представляющий собой альтернативу параллельному SCSI. Самая новая версия serial SCSI (SAS 2.0) поддерживает скорость передачи данных до 6 Гбит/с. Более подробно SCSI-архитектура и интерфейс рассматриваются в приложении Б.

Протокол Fibre Channel

Протокол Fibre Channel получил широкое распространение и используется для высокоскоростного обмена данными с устройствами хранения. Интерфейс Fibre Channel обеспечивает гигабитную скорость передачи данных по сети. Он выполняет последовательную передачу данных по медному проводу и оптоволокну. Самая последняя версия FC-интерфейса (16FC) позволяет передавать данные со скоростью до 16 Гбит/с. Протокол FC и его свойства более подробно рассматриваются в главе 5.

Internet Protocol (IP)

IP является сетевым протоколом, который традиционно использовался для трафика типа «хост — хост». С появлением новых технологий IP-сеть стала использоваться для обмена данными типа «хост — хранилище». IP предлагает ряд преимуществ по стоимости и завершенности и позволяет организациям воспользоваться уже существующими у них сетями на основе IP-протокола. Типичными примерами использования IP-протокола для обмена данными между хостом и хранилищем являются протоколы iSCSI и FCIP. Более подробно эти протоколы рассматриваются в главе 6.

2.5. Устройство хранения данных

Устройство хранения данных является важнейшим компонентом дата-центра. В нем используются магнитные, оптические или твердотельные носители информации. Диски, ленты и дискеты используют магнитные носители,

а компакт-диски и DVD используют для хранения данных оптические носители. Примерами твердотельного носителя могут послужить перемещаемая флеш-память или флеш-накопители.

В прошлом благодаря своей дешевизне наиболее популярным вариантом хранилища резервных копий служили магнитные ленты. Но по части производительности и управления у лент имеются различные ограничения.

- Данные хранятся на ленте линейно по всей длине. Поиск и извлечение данных осуществляются последовательно, что неизменно приводит к тому, что обращение к данным занимает несколько секунд. В итоге произвольный доступ к данным происходит медленно и отнимает много времени. Это обстоятельство делает ленты нежелательным вариантом для приложений, требующих быстрого доступа к данным в реальном масштабе времени.
- В совместно используемой вычислительной среде хранящиеся на ленте данные не могут быть одновременно доступны сразу нескольким приложениям, что ограничивает их использование в любой момент времени всего одним приложением.
- На ленточном устройстве головки чтения-записи соприкасаются с поверхностью пленки, поэтому после многократного применения пленка приходит в негодность или изнашивается.
- Весьма важную роль играют требования к хранению данных на ленте и их извлечению с нее и издержки, связанные с управлением ленточным накопителем.

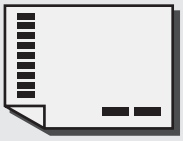
Из-за этих ограничений и по причине доступности дешевых дисковых накопителей ленты больше не являются предпочтительным вариантом средств резервного копирования для дата-центров корпоративного класса.

Хранилища, использующие оптические диски, популярны в небольших вычислительных средах с одним пользователем. Они часто используются людьми для хранения фотографий или в качестве носителя резервных копий на персональном компьютере или ноутбуке. Они также используются в качестве носителей дистрибутивов небольших приложений, таких как игры, или как средства передачи небольших объемов данных с одной компьютерной системы на другую. Оптические диски ограничены в объеме хранимой информации и скорости передачи данных, что служит препятствием для их применения в качестве решений для хранилищ деловой информации.

Возможность однократной записи и многократного считывания (WORM) — одно из преимуществ хранения данных на оптических дисках. Примером WORM-устройства может послужить компакт-диск. Оптические диски в определенной степени гарантируют отсутствие изменений в содержимом. Поэтому их можно использовать как малобюджетный альтернативный вариант для длительного хранения относительно малых объемов фиксированного контента, который не изменится после того, как был записан. Совокупность

оптических дисков в массиве, называемом *авточенжером*, до сих пор является одним из решений для хранения фиксированного контента. Другие виды оптических дисков представлены перезаписываемыми компакт-дисками (CD-RW) и различными вариантами DVD.

Наиболее популярными носителями для хранения данных, используемыми в современных компьютерах для хранения данных и предоставления доступа к ним высокопроизводительным постоянно работающим приложениям, являются *дисковые накопители*. Диски поддерживают быстрый доступ к произвольным местам расположения данных. Следовательно, данные могут быть быстро записаны или извлечены большим количеством одновременно работающих пользователей или приложений. Кроме того, диски обладают большой емкостью. Для обеспечения наращиваемого объема и улучшенной производительности применяются несколько дисков, которые сводятся в массивы дисковых хранилищ.



Обращение к дисковым накопителям осуществляется посредством предопределенных протоколов, таких как ATA, Serial ATA (SATA), SAS (Serial Attached SCSI) и FC. Эти протоколы реализуются в контроллерах дисковых интерфейсов. Ранее контроллеры дисковых интерфейсов были реализованы в виде отдельных плат, устанавливаемых в разъем материнской платы с целью обеспечения обмена данными с устройствами хранения информации. Современные контроллеры дисковых интерфейсов интегрированы с дисковыми накопителями, поэтому такие накопители известны по поддерживаемому ими протокольному интерфейсу, например SATA-диск, FC-диск и т. д.

2.6. Компоненты дискового накопителя

Ключевыми компонентами жесткого диска являются пластина, шпиндель, головка чтения-записи, кронштейн привода блока головок и плата контроллера (рис. 2.5).

Операции ввода-вывода в жестком диске выполняются за счет быстрого перемещения кронштейна блока головок над поверхностью вращающихся пластин, покрытых магнитными частицами. Данные между контроллером диска и магнитными пластинами передаются посредством головки чтения-записи, смонтированной на кронштейне. Данные можно записывать на магнитные пластины и стирать с них любое количество раз. В следующих разделах будут подробно рассмотрены различные компоненты дискового накопителя, механизм организации и хранения данных на дисках и факторы, влияющие на производительность жесткого диска.

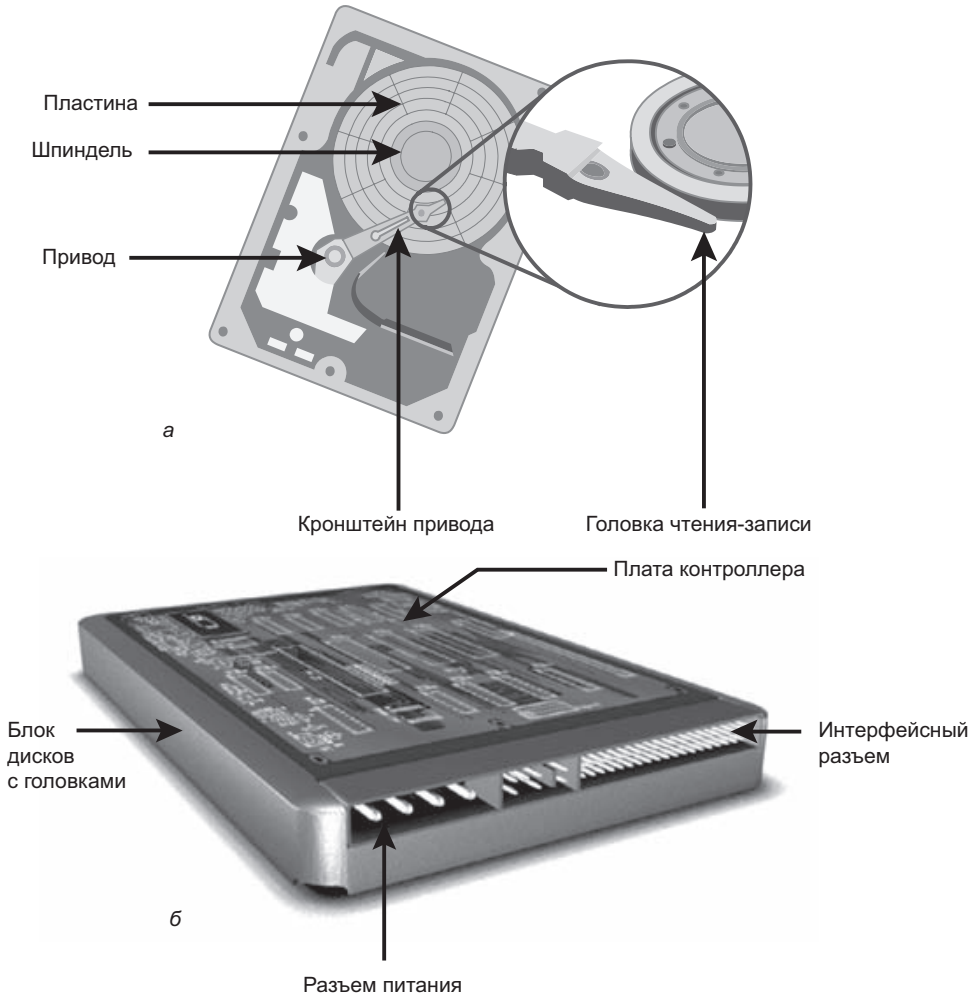


Рис. 2.5. Компоненты дискового накопителя

2.6.1. Магнитная пластина

Обычный жесткий диск состоит из одного или нескольких плоских круглых дисков, называемых *пластинами* (рис. 2.6). Запись данных на эти магнитные пластины происходит в двоичных кодах (в виде нулей и единиц). Набор вращающихся магнитных пластин в герметичном контейнере называется *блоком дисков с головками* — Head Disk Assembly (HDA). Магнитная пластина представляет собой жесткий круглый диск, покрытый с обеих сторон (снизу и сверху) магнитным материалом. Данные кодируются путем поляризации магнитных областей, или доменов, на поверхности диска. Запись или считывание данных возможны с обеих поверхностей пластины. Общий

объем диска определяется количеством пластин и вместимостью каждой пластины.

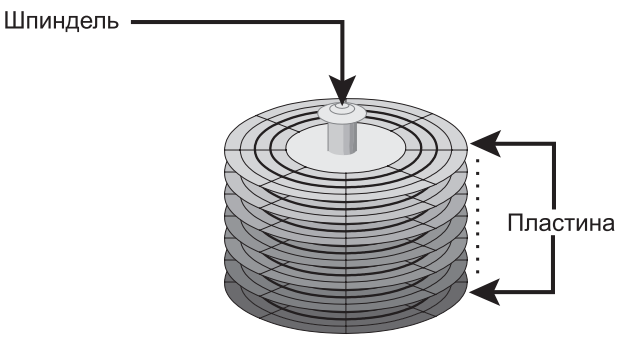


Рис. 2.6. Шпиндель и пластина

2.6.2. Шпиндель

Все пластины собираются на шпинделе (см. рис. 2.6), к которому присоединен двигатель, вращающий шпиндель с постоянной угловой скоростью.

Дисковая пластина вращается со скоростью несколько тысяч оборотов в минуту. Самыми распространенными скоростями вращения шпинделей являются 5400, 7200, 10 000 и 15 000 об./мин. С усовершенствованием технологий скорость вращения пластин возрастает, но степень допустимого возрастания ограничена.

2.6.3. Головка чтения-записи

Головки чтения-записи, показанные на рис. 2.7, осуществляют чтение данных с пластин и запись на пластины. На накопителях имеется по две головки чтения-записи на каждую пластину, по одной для каждой из поверхностей пластины. При записи данных головка чтения-записи меняет магнитную поляризацию участков поверхности пластины. При чтении данных эта головка обнаруживает магнитную поляризацию участков поверхности пластины. Когда ведутся чтение и запись, головка чтения-записи распознает магнитную поляризацию, но никогда при этом не касается поверхности пластины. При вращении шпинделя образуется микроскопический воздушный зазор между головками чтения-записи и пластинами, известный как *высота зависания головки*. Воздушный зазор исчезает, когда шпиндель прекращает вращение, и головка чтения-записи лежит на специальной области пластины рядом со шпинделем. Эта область называется *зоной парковки*. Эта зона покрывается смазочным материалом для снижения трения между головкой и пластиной.

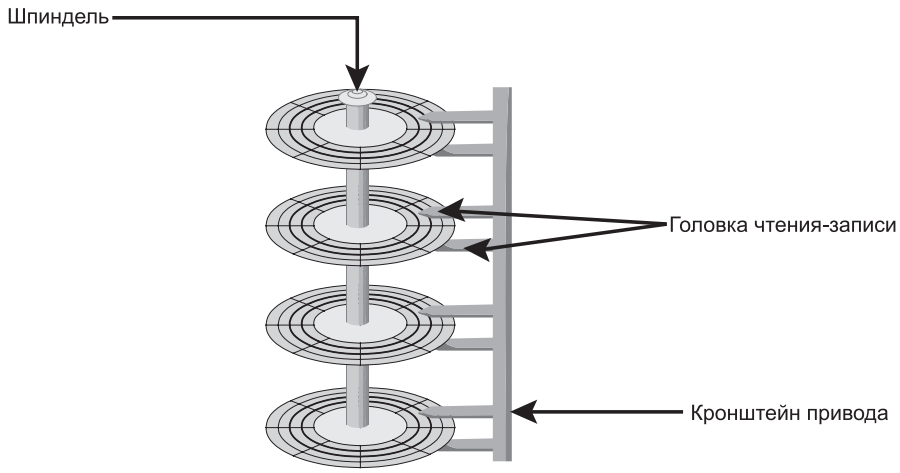


Рис. 2.7. Кронштейн привода блока головок

Логикой работы дискового накопителя обеспечивается перемещение головок в зону парковки до того, как они коснутся поверхности. Если накопитель работает неправильно и головка чтения-записи случайно коснется поверхности пластины за пределами зоны парковки, происходит *авария головки*. При аварии головки магнитное покрытие пластины царапается, что может привести к повреждению головки чтения-записи. Авария головки, как правило, ведет к потере данных.

2.6.4. Кронштейн привода блока головок

Головки чтения-записи крепятся к кронштейну привода блока головок, позиционирующего головку чтения-записи над тем местом пластины, где находятся или будут находиться данные, которые нужно считать или записать (см. рис. 2.7). Головки чтения-записи всех пластин крепятся к одному кронштейну привода блока головок и перемещаются над пластинами одновременно.

2.6.5. Плата контроллера накопителя

Контроллер (см. рис. 2.5, *б*) — это печатная плата, установленная снизу дискового накопителя. Контроллер состоит из микропроцессора, внутренней памяти, электрических цепей и программ, управляющих электропитанием двигателя шпинделя и скоростью вращения его ротора. Эти программы также управляют обменом данных между накопителем и хостом. Кроме того, они же управляют операциями чтения-записи путем перемещения кронштейна привода блока головок и переключения между различными головками чтения-записи, а также оптимизируют доступ к данным.

2.6.6. Структура физического диска

Данные на диске записываются на *дорожки*, которые представляют собой концентрические окружности на пластине вокруг шпинделя (рис. 2.8). Дорожки нумеруются от наружного края пластины, начиная с нулевой. Плотность размещения дорожек на пластине определяется количеством дорожек на дюйм — tracks per inch (TPI).

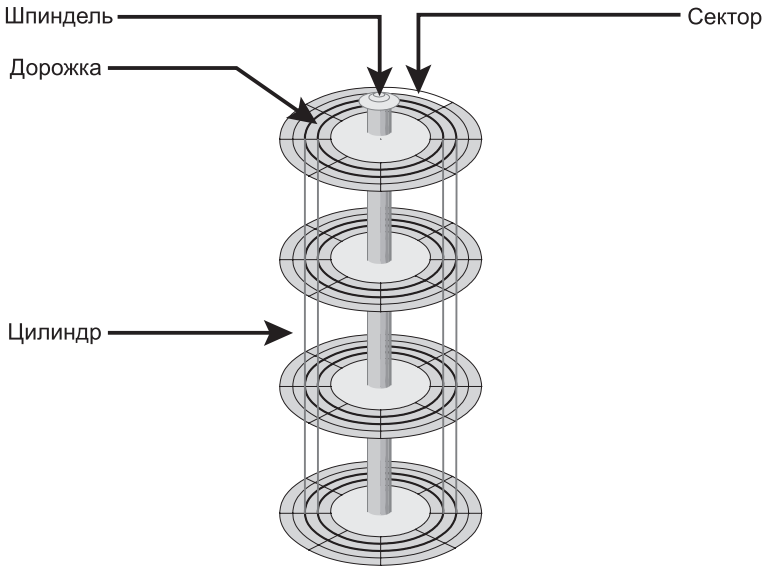


Рис. 2.8. Структура диска: секторы, дорожки и цилиндры

Каждая дорожка разбивается на более мелкие части, называемые *секторами*. Сектор — это наименьший отдельно адресуемый элемент накопителя. Структура дорожек и секторов записывается на пластину производителем устройства посредством операции низкоуровневого форматирования. Количество секторов на дорожке варьируется в зависимости от типа накопителя. У дисков первых персональных компьютеров было по 17 секторов на дорожке. У современных дисков количество секторов на одной дорожке значительно больше. В зависимости от физических размеров и плотности записи на пластине может быть несколько тысяч дорожек.

Как правило, в секторе содержится 512 байт пользовательских данных, хотя некоторые диски могут быть отформатированы под сектора бóльших размеров. Помимо пользовательских данных в секторе хранится другая информация, такая как номер сектора, головки или пластины и номер дорожки. Эта информация помогает контроллеру определять местонахождение данных на накопителе.

Цилиндр — это совокупность идентичных дорожек на обеих поверхностях каждой пластины накопителя. При определении местоположения головок чтения-записи делается ссылка на номер цилиндра, а не дорожки.

СРАВНЕНИЕ ЗАЯВЛЕННОЙ И ДОСТУПНОЙ ЕМКОСТИ ДИСКА

Заявленная емкость диска отличается от пространства, реально доступного для хранения данных. Например, диск с объявленной емкостью 500 Гбайт имеет всего лишь 465,7 Гбайт пространства для хранения пользовательских данных. Дело в том, что производители жестких дисков используют для определения их емкости числа по основанию 10, а это означает, что 1 Кбайт эквивалентен 1000 байт, а не 1024 байт, следовательно, реально доступное пространство диска всегда меньше, чем его заявленная емкость.

2.6.7. Зональная побитовая запись

На пластинах расположены концентрические дорожки, при этом дорожки, находящиеся ближе к краю, могут содержать больше данных, поскольку физически они длиннее тех дорожек, которые находятся ближе к шпинделю. На прежних дисковых накопителях внешние дорожки имели такое же количество секторов, что и внутренние, поэтому плотность записи данных на внешних дорожках была ниже. Доступное пространство в этом случае использовалось неэффективно (рис. 2.9, *а*).

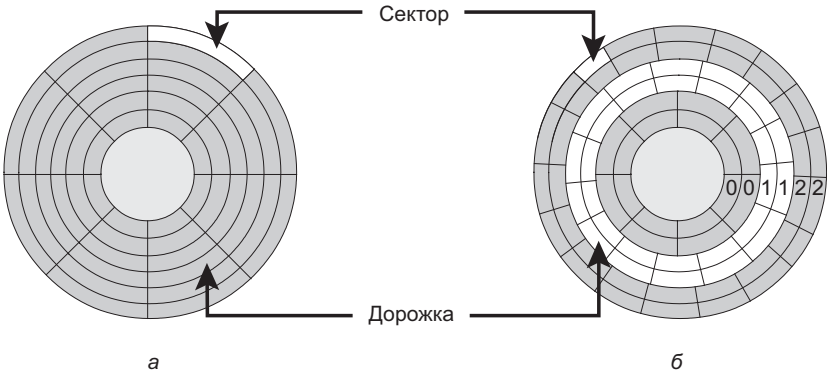


Рис. 2.9. Зональная побитовая запись: *а* — пластина без зон; *б* — пластина с зонами

Зональная побитовая запись позволяет использовать дисковое пространство более эффективно. Как показано на рис. 2.9, *б*, этот механизм группирует дорожки в зоны на основе их удаленности от центра диска. Зоны нумеруются, при этом самая крайняя зона получает нулевой номер. Каждой зоне

назначается соответствующее количество секторов на дорожку, чтобы зона, находящаяся ближе к центру пластины, имела меньше секторов на дорожке, чем зона, расположенная с краю. Но дорожки в определенной зоне имеют одинаковое количество секторов.

При обращении к данным из зоны, расположенной ближе к центру пластины, скорость передачи данных падает. У приложений, требующих высокой производительности, данные должны располагаться во внешних зонах пластины.

2.6.8. Адресация логических блоков

На первых дисках для ссылки на определенные места диска применялась физическая адресация, состоящая из номеров цилиндра, головки и сектора — cylinder, head, sector (CHS) (рис. 2.10, *а*), и основной операционной системе следовало знать геометрию каждого используемого диска. Адресация логических блоков — logical block addressing (LBA) (рис. 2.10, *б*) упростила адресацию за счет того, что при обращении к физическим блокам данных используется линейный адрес. Контроллер диска переводит LBA-адрес в адрес CHS, и хосту нужно знать лишь размер дискового накопителя, выраженный в количестве блоков. Логические блоки отображаются на физические сектора один к одному.

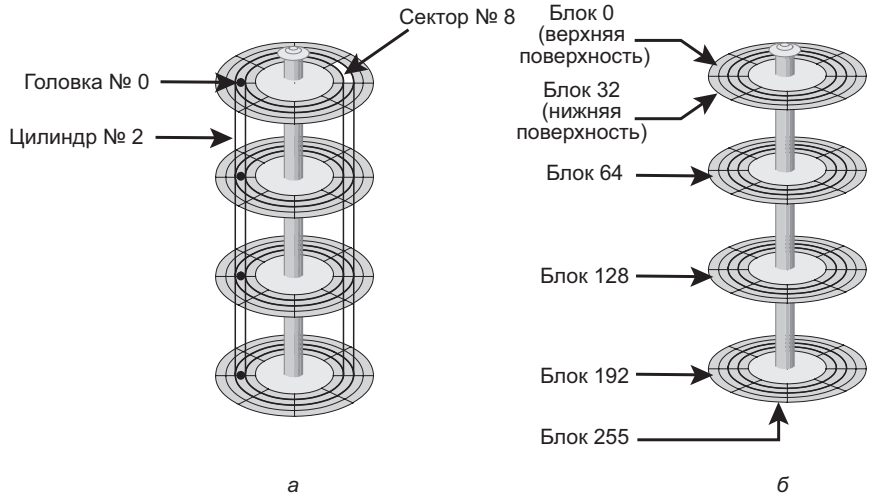


Рис. 2.10. Физический адрес и адрес логического блока: *а* — физический адрес = CHS; *б* — адрес логического блока = № блока

На рис. 2.10, *б*, у накопителя восемь секторов на одну дорожку, восемь головок и четыре цилиндра. Следовательно, всего мы имеем дело с $8 \cdot 8 \cdot 4 = 256$ блоками, а номера блоков умещаются в диапазоне от 0 до 255. У каждого блока имеется собственный уникальный адрес. При условии, что в секторе содержится 512 байт, накопитель емкостью 500 Гбайт с отформатированной емкостью 465,7 Гбайт содержит более 976 млн блоков.

2.7. Производительность дискового накопителя

Дисковый накопитель является электромеханическим устройством, влияющим на общую производительность среды, относящейся к системе хранения данных. В данном разделе рассматриваются различные факторы, влияющие на производительность дисковых накопителей.

2.7.1. Время обслуживания

Время обслуживания (Disk Service time) — это время, необходимое диску на выполнение одного запроса ввода-вывода данных. Оно складывается из следующих компонентов: времени поиска, времени задержки, связанной с вращением диска, и времени, обусловленного скоростью передачи данных.

Время поиска

Время поиска (также называемое временем доступа) — это время, затрачиваемое на позиционирование головок чтения-записи над пластиной путем их радиального перемещения (по радиусу пластины). Иными словами, это время, затрачиваемое на позиционирование и расположение кронштейна и головки над нужной дорожкой. Следовательно, чем короче время поиска, тем быстрее проводится операция ввода-вывода. Поставщики дисков заявляют следующие характеристики времени поиска.

- **Время полного хода (Full Stroke)** — время, затрачиваемое на перемещение головки чтения-записи над всей поверхностью диска, от самой близкой к шпинделю дорожки до самой удаленной.
- **Среднее время (Average)** — среднее время, затрачиваемое на перемещение головки чтения-записи от одной произвольно взятой дорожки к другой такой же дорожке; обычно в качестве этого времени приводится показатель, составляющий одну треть от времени полного хода.
- **Время перехода с дорожки на дорожку (Track-to-Track)** — время, затрачиваемое на перемещение головки чтения-записи с одной дорожки на смежную с ней дорожку.

Каждая из этих характеристик измеряется в миллисекундах. Время поиска обычно указывается производителем дискового накопителя. Среднее время

поиска на современных дисках обычно находится в диапазоне от 3 до 15 мс. Наибольшее влияние время поиска оказывает на операции чтения произвольной дорожки, а не на чтение соседних дорожек. Для сведения времени поиска к минимуму данные могут быть записаны только в одну подгруппу доступных цилиндров. Это приводит к снижению полезной емкости по сравнению с фактической емкостью дискового накопителя. Например, дисковый накопитель емкостью 500 Гбайт настраивается на использование только первых 40 % цилиндров и фактически рассматривается как накопитель емкостью 200 Гбайт. Это называется использованием дискового накопителя в режиме короткого хода.

Время задержки, связанной с вращением диска

Для получения доступа к данным кронштейн привода перемещает головку чтения-записи над пластиной к определенной дорожке, а пластина за счет вращения подводит нужный сектор под головку. Время, затрачиваемое на вращение пластины и подведение данных под головку чтения-записи, называется временем задержки, связанным с вращением диска. Эта задержка зависит от скорости вращения шпинделя и измеряется в миллисекундах. Средняя задержка, связанная с вращением диска, составляет половину времени, затрачиваемого на его полный оборот. Как и время поиска, эта задержка больше всего влияет на чтение-запись произвольных секторов на диске, а не на те же операции над смежными секторами.

Среднее время задержки, связанной с вращением диска, составляет приблизительно 5,5 мс для диска со скоростью вращения шпинделя 5400 об./мин и, как показано далее, около 2 мс для диска со скоростью вращения шпинделя 15 000 об./мин (или 250 об./с):

> Среднее время задержки, связанной с вращением диска, вращающегося со скоростью 15 000 об./мин (или 250 об./с) = 0,5/250 = 2 мс.

Время, обусловленное скоростью передачи данных

Скорость передачи данных (также называемая скоростью передачи) относится к среднему объему данных за единицу времени, доставляемому дисковым накопителем к шинному адаптеру хоста (HBA). Для вычисления скорости передачи данных прежде всего важно понять, как проходят операции чтения-записи. При операции чтения данные сначала передаются с дисковых пластин на головки чтения-записи, а затем — во внутренний буфер накопителя. И наконец, данные перемещаются из буфера через интерфейс к шинному адаптеру хоста (HBA). При операции записи данные передаются из HBA во внутренний буфер дискового накопителя через его интерфейс, а затем — из буфера к головкам чтения-записи. И наконец, они перемещаются из головок чтения-записи на пластины.

Скорости передачи данных в ходе операций чтения-записи измеряются по внутренним и внешним скоростям передачи данных (рис. 2.11).

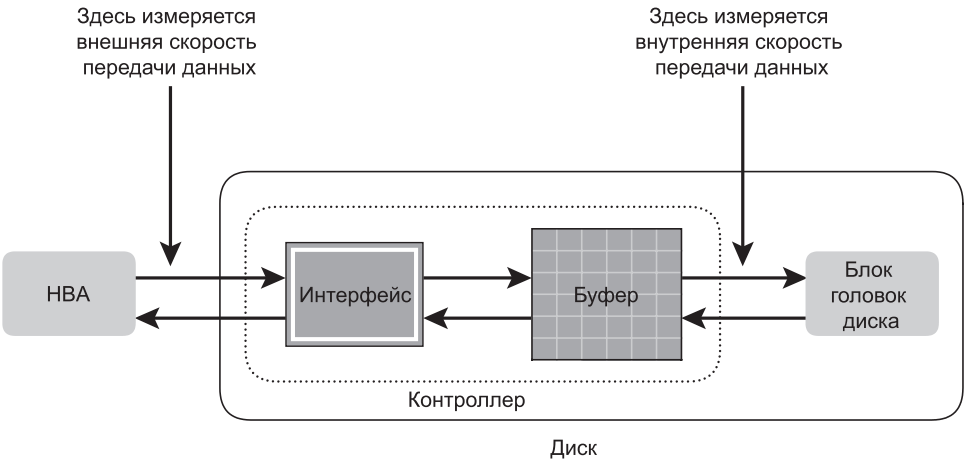


Рис. 2.11. Скорость передачи данных

Внутренняя скорость передачи данных — это скорость перемещения данных с поверхности пластины во внутренний буфер (кэш-память) диска. При вычислении внутренней скорости передачи данных в расчет берутся такие факторы, как время поиска и время задержки, связанной с вращением диска. *Внешняя скорость передачи данных* — это скорость перемещения данных через интерфейс к шинному адаптеру хоста. Внешняя скорость передачи данных обычно соответствует заявленной скорости интерфейса, например 133 Мбит/с для интерфейса ATA. Поддерживаемая внешняя скорость передачи данных ниже скорости интерфейса.

2.7.2. Загруженность дискового контроллера ввода-вывода

Загруженность дискового контроллера ввода-вывода оказывает большое влияние на время отклика на запрос ввода-вывода. Чтобы разобраться в степени этого влияния, нужно представить диск в виде черного ящика, состоящего из двух компонентов:

- **очереди** — места, где запрос на ввод-вывод ожидает своей обработки контроллером ввода-вывода;
- **дискового контроллера ввода-вывода** — устройства поочередной обработки запросов на ввод-вывод.

Запросы на ввод-вывод поступают в контроллер с той скоростью, с которой они выдаются приложением. Эта скорость также называется *частотой поступления*. Эти запросы содержатся в очереди ввода-вывода, и контроллер

ввода-вывода обрабатывает их по одному в порядке очередности (рис. 2.12). Время отклика на запрос ввода-вывода определяется частотой поступления запросов на ввод-вывод, длиной очереди и временем, которое контроллер ввода-вывода затрачивает на обработку каждого запроса. Если контроллер занят или сильно загружен, размер очереди будет большим, а время отклика на запрос — более продолжительным.

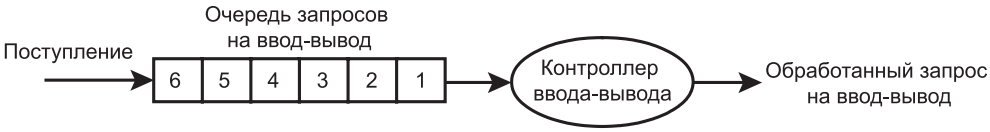


Рис. 2.12. Обработка запросов на ввод-вывод

Основываясь на фундаментальных законах производительности дискового накопителя, зависимость среднего времени отклика на запрос ввода-вывода от степени загруженности контроллера выражается следующем образом:

> Среднее время отклика (*T*R) = Время обслуживания запроса (*T*S) / (1 – Коэффициент использования),

где *T*S — время, затрачиваемое контроллером на обслуживание ввода-вывода.

Когда коэффициент использования достигает 100 %, то есть контроллер ввода-вывода оказывается перегруженным, время отклика становится близким к бесконечности. По сути, компонент, испытывающий перегрузки, или самое узкое место в системе, вынуждает выстраивать запросы на ввод-вывод в последовательность, означающую, что каждый запрос на ввод-вывод должен ожидать завершения предшествующих ему таких же запросов. На рис. 2.13 показан график зависимости времени отклика от загруженности контроллера.

На графике видно, что изменение времени отклика по мере роста коэффициента использования имеет нелинейный характер. Когда средний размер

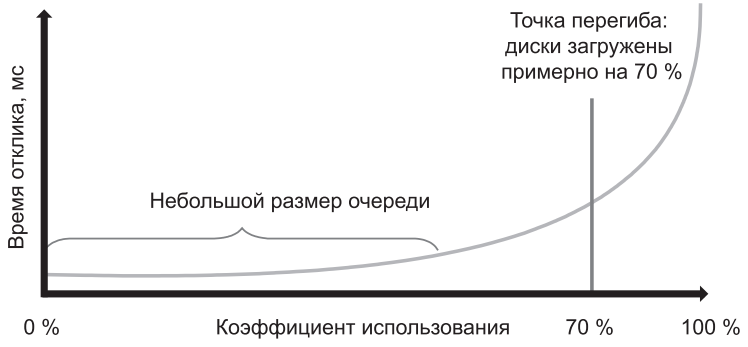


Рис. 2.13. Влияние загруженности контроллера на время отклика на запрос

очереди небольшой, время отклика остается невысоким. При повышении коэффициента использования (увеличении очереди) наблюдается медленный рост времени отклика, который при достижении 70 % начинает расти в геометрической прогрессии. Поэтому для приложений с высокими требованиями к производительности диски обычно загружаются не более чем на 70 % от их возможностей по обслуживанию запросов на ввод-вывод.

2.8. Доступ хоста к данным

Приложения получают доступ к данным и сохраняют их с помощью базовой инфраструктуры. Основными компонентами этой инфраструктуры являются операционная система (или файловая система), система передачи данных и хранилище. По отношению к хосту устройство хранения данных может быть внутренним и/или внешним. В любом случае карта контроллера хоста обращается к устройствам хранения данных с помощью предопределенных протоколов, таких как IDE/ATA, SCSI или Fibre Channel (FC). Для доступа к внешним хранилищам в небольших и персональных вычислительных средах широко используются протоколы IDE/ATA и SCSI. Протоколы FC и iSCSI используются для доступа к данным из внешнего устройства хранения данных (или из подсистем). Внешние устройства хранения данных могут быть подключены к хосту напрямую или через сеть хранения данных. Когда хранилище подключено к хосту напрямую, оно называется хранилищем с прямым подключением — direct-attached storage (DAS). Это хранилище будет подробно рассмотрено в данной главе ниже.

Поскольку доступ к данным по сети положен в основу сетевых технологий хранения данных, в нем необходимо разобраться. Данные по сети могут быть доступны одним из следующих способов: на блочном, файловом или объектном уровне.

Как правило, приложение запрашивает данные из файловой системы (или операционной системы) путем указания имени файла и того места, где он находится. Файловая система отображает атрибуты файла на адрес логического блока данных и отправляет запрос к устройству хранения. Устройство хранения данных преобразует адрес логического блока (LBA) в адрес типа «цилиндр — головка — сектор» (CHS) и извлекает данные.

При доступе на уровне блоков на хосте создается файловая система, а доступ к данным осуществляется по сети на уровне блоков (рис. 2.14, *а*). В этом случае для создания файловой системы хосту выделяются целые диски или логические тома.

При доступе на уровне файлов файловая система создается на отдельном файловом сервере или на стороне хранилища, а по сети передается запрос на уровне файлов (рис. 2.14, *б*). Поскольку доступ к данным осуществляется на уровне файлов, издержки этого метода выше, чем при доступе к данным

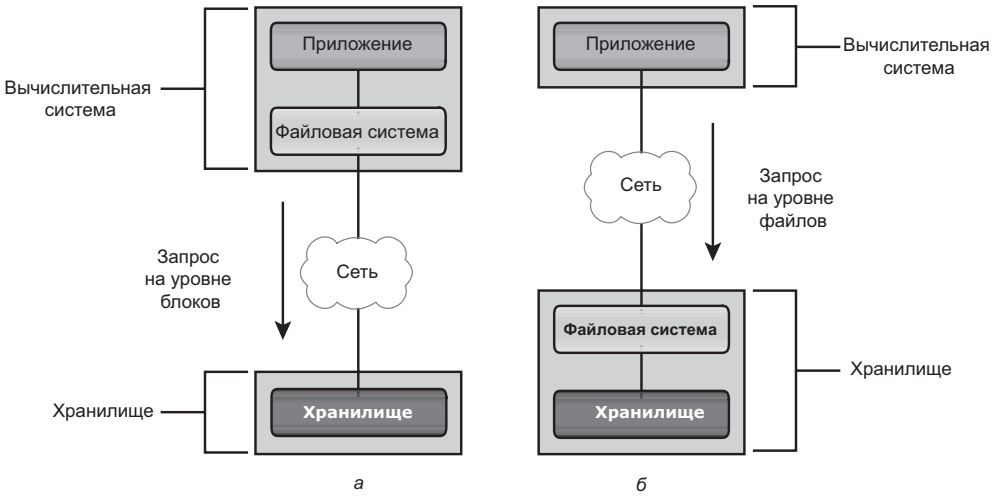


Рис. 2.14. Доступ хоста к данным: *а* — доступ на уровне блоков; *б* — доступ на уровне файлов

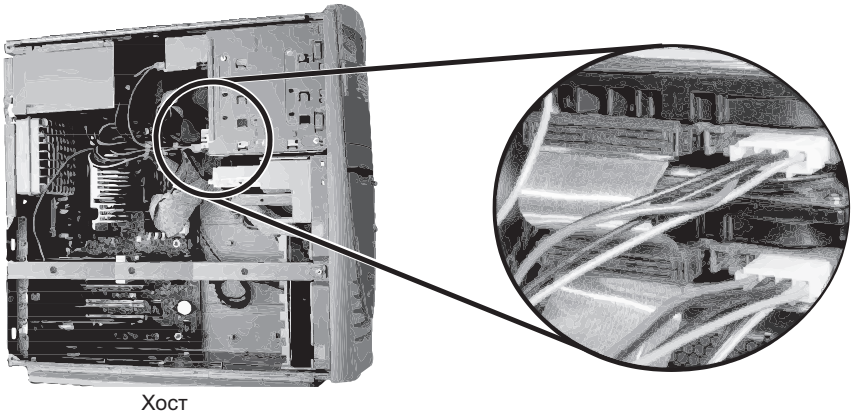
на уровне блоков. Доступ к данным на уровне объектов стал результатом интеллектуального развития средств хранения данных, и, соответственно, доступ к данным по сети осуществляется на основе независимых объектов, имеющих собственные уникальные идентификаторы. Подробности сетевых технологий хранения данных и их развертывания рассматриваются в разделе II данной книги «Сетевые технологии хранения данных».

2.9. Хранилище с прямым подключением

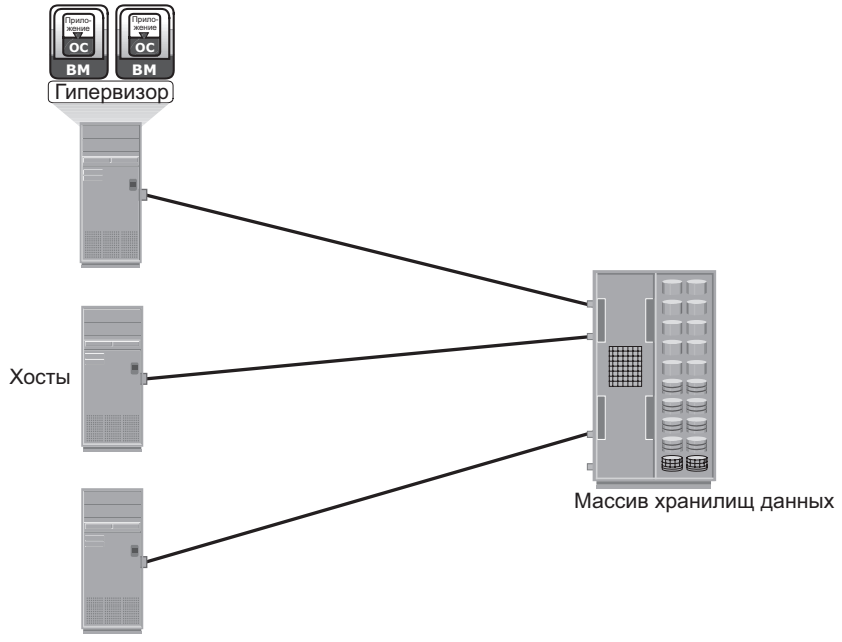
Хранилище с прямым подключением (DAS) представляет собой архитектуру, в которой само хранилище напрямую подключено к хостам. В качестве примеров DAS можно привести внутренний дисковый накопитель хоста и подключенный напрямую внешний массив хранилищ данных. Несмотря на набирающую популярность реализацию сетевых технологий хранения данных, DAS все еще считается вполне подходящим решением для локального доступа к данным в небольших вычислительных средах, таких как персональные вычисления и рабочие группы. В зависимости от расположения устройства хранения данных по отношению к хосту DAS-хранилища делятся на два класса: внутренние и внешние.

Во внутренних DAS-архитектурах устройство хранения данных имеет внутреннее подключение к хосту по последовательной или параллельной шине (рис. 2.15, *а*). У физической шины имеются ограничения по протяженности, и она может стабильно работать в режиме высокоскоростной передачи данных только на короткие расстояния. Кроме того, большинство внутренних

шин могут поддерживать только ограниченное количество устройств, и эти устройства занимают внутри хоста слишком много места, затрудняя обслуживание других компонентов.



а



б

Рис. 2.15. Внутренняя и внешняя DAS-архитектура: *а* — внутреннее DAS-хранилище; *б* — внешнее DAS-хранилище

В то же время при использовании внешних DAS-архитектур хост подключается непосредственно к внешнему устройству хранения данных и доступ к данным осуществляется на уровне блоков (рис. 2.15, *б*). В большинстве случаев обмен данными между хостом и устройством хранения данных производится по протоколу SCSI или FC. По сравнению с внутренними DAS внешние DAS-хранилища не испытывают ограничений по удаленности или количеству устройств и предоставляют возможность централизованного управления устройствами хранения данных.

2.9.1. Преимущества и недостатки DAS

По сравнению с сетевыми архитектурами хранения данных DAS-архитектура требует относительно небольших начальных вложений. Конфигурирование DAS не вызывает затруднений, и хранилище этого типа может быть развернуто быстрее и с меньшими затратами. Настройка ведется с использованием средств операционной системы хоста, что упрощает задачи управления хранилищами в небольших вычислительных средах. Поскольку архитектура DAS не отличается особой сложностью, у нее меньше управленческих задач и она содержит меньше оборудования и программ, требующих настройки и управления.

Но масштабируемость DAS оставляет желать лучшего. У массива хранилища данных ограниченное количество портов, что лимитирует количество хостов, подключаемых к нему напрямую. Когда потенциал исчерпывается, услуги могут стать недоступными. В связи с ограниченной возможностью совместного использования интерфейсных портов DAS не в состоянии обеспечить оптимальное использование ресурсов. Невостребованные ресурсы в DAS-средах не поддаются легкому перераспределению, что приводит к образованию изолированных друг от друга перегруженных и недогруженных пулов хранения данных.

2.10. Проектирование хранилища на основе требований, предъявляемых приложениями, и показателей производительности дисков

Определение требований, предъявляемых приложением, начинается с выявления запрашиваемой емкости хранилища. Эта емкость легко рассчитывается по размеру и количеству файловых систем и компонентов базы данных, используемых приложениями. Другими факторами, влияющими на выбор производительности диска, времени его отклика на запросы ввода-вывода и на конструкции систем хранения данных, являются объем операций ввода-вывода и их характеристики, а также количество запросов на ввод-вывод,

выдаваемое приложением при пиковой рабочей нагрузке. Размер блока ввода-вывода зависит от файловой системы и базы данных, на которых построено приложение. Размер блока в среде базы данных управляется ее базовым механизмом и переменными среды.

Основной мерой производительности диска является время, затрачиваемое диском на обслуживание одного запроса на ввод-вывод *T*S. Этот показатель, а также коэффициент использования диска *U* определяют время отклика на запрос ввода-вывода для приложения. Как уже упоминалось, полное время обслуживания диском одного запроса *T*S складывается из времени поиска *T*, задержки, связанной с вращением диска *L*, и внутреннего времени передачи данных *X*:

$$TS = T + L + X.$$

Рассмотрим пример со следующими характеристиками, предоставляемыми диском.

Среднее время поиска в среде с произвольным вводом-выводом равно 5 мс, то есть $T = 5$ мс.

Скорость вращения диска равна 15 000 об./мин, или 250 об./с, из чего следует, что задержка *L*, связанная с вращением диска, может быть определена как половина времени, затрачиваемого на полный оборот диска, или L = (0,5/250 об./с) [мс].

Внутренняя скорость передачи данных равна 40 Мбайт/с, из чего внутреннее время передачи данных *X* выводится на основе размера блока ввода-вывода. Например, при вводе-выводе с размером блока 32 Кбайт время X = 32 Кбайт/40 Мбайт.

Соответственно, время, затрачиваемое контроллером на обслуживание ввода-вывода блока размером 32 Кбайт рассчитывается следующим образом: $TS = 5$ мс + (0,5/250) + 32 Кбайт/40 Мбайт = 7,8 мс.

Следовательно, максимальное количество обслуживаемых запросов на ввод-вывод в секунду, или IOPS, рассчитывается как $1/TS = 1/(7{,}8 \cdot 10–3) = 128$ IOPS.

В табл. 2.1 перечислены максимальные количества IOPS, обслуживаемые при различных размерах блока с использованием прежних характеристик диска.

Таблица 2.1. Количества IOPS, обслуживаемые дисковым накопителем

РАЗМЕР БЛОКА	$T_S = T + L + X$	IOPS = 1/*TS*
4 Кбайт	5 мс + (0,5/250 об./с) + 4 Кбайт/40 Мбайт = 5 + 2 + 0,1 = 7,1	140
8 Кбайт	5 мс + (0,5/250 об./с) + 8 Кбайт/40 Мбайт = 5 + 2 + 0,2 = 7,2	139
16 Кбайт	5 мс + (0,5/250 об./с) + 16 Кбайт/40 Мбайт = 5 + 2 + 0,4 = 7,4	135
32 Кбайт	5 мс + (0,5/250 об./с) + 32 Кбайт/40 Мбайт = 5 + 2 + 0,8 = 7,8	128
64 Кбайт	5 мс + (0,5/250 об./с) + 64 Кбайт/40 Мбайт = 5 + 2 + 1,6 = 8,6	116

Количество IOPS, находящееся в диапазоне от 116 до 140 для блоков разного размера, представляет собой показатель IOPS, который может достигаться при потенциально высоких уровнях загруженности (близких к 100 %). В разделе 2.7.2 уже упоминалось, что время отклика на запрос приложения, *T*R, возрастает с возрастанием коэффициента использования контроллера диска. Для примера, аналогичного предыдущему, время отклика (*T*R) для ввода-вывода с размером блока, равным 32 Кбайт, при коэффициенте использования контроллера диска, равном 96 %, вычисляется следующим образом:

$$T_R = T_S /(1 - U) = 7{,}8/(1 - 0{,}96) = 195 \text{ мс}.$$

Если приложению требуется менее продолжительное время отклика, коэффициент использования дисков должен поддерживаться на уровне ниже 70 %. Для такого же размера блока (32 Кбайт) при коэффициенте использования 70 % время отклика существенно сокращается — до 26 мс. Но при снижении коэффициента использования диска сокращается и количество возможных IOPS, выполняемых диском. При размере блока, равном 32 Кбайт, диск может выполнять 128 IOPS при коэффициенте использования почти 100 %, а количество IOPS, выполняемых им при 70 %, составляет 89 (128 · 0,7). Это говорит о том, что количество операций ввода-вывода, которое может выполнить диск, является важным фактором и его следует брать в расчет при определении требований к хранилищу для приложения.

Следовательно, требования к хранилищу для приложения выражаются как в емкости, так и в количестве IOPS. Если приложению требуется 200 Гбайт дискового пространства, эту емкость может предоставить один обычный диск. Но если у приложения высокие требования к показателю IOPS, то такое решение может привести к снижению производительности, поскольку один диск может не обеспечить требуемое время отклика для операций ввода-вывода.

Исходя из этих рассуждений, общее количество требуемых дисков (*D*R) для приложения можно рассчитать следующим образом:

$$D_R = \text{Max}\,(D_C, D\text{I}),$$

где D_C — это количество дисков, отвечающее требованиям по *емкости*, а D_I — количество дисков, отвечающее требованиям приложения по показателю IOPS. Давайте поясним это на примере.

Рассмотрим вариант, при котором требования приложения к емкости составляют 1,46 Тбайт. Количество IOPS, выдаваемых приложением при пиковой рабочей нагрузке, приблизительно 9000 IOPS. Поставщик заявил, что дисковый накопитель емкостью 146 Гбайт со скоростью вращения шпинделя 15 000 об./мин может выполнить максимум 180 IOPS. В данном примере количество дисков, отвечающее условиям емкости, составит 1,46 Тбайт/146 Гбайт = 10.

Чтобы соответствовать требованиям приложения по показателю IOPS, количество дисков должно быть 9000/180 = 50. Но если работа приложения сильно зависит от времени отклика, количество IOPS, которое дисковый накопитель может выполнить, нужно вычислять при степени

загруженности, равной 70 %. При таких условиях количество IOPS, которое диск может выполнить при степени загруженности 70 %, составляет 180 · 0,7 = 126 IOPS. То есть количество дисков, отвечающее требованиям приложения по показателю IOPS, будет 9000/126 = 72. В результате количество дисков, отвечающее требованиям приложения, будет определено из расчета Max (10, 72) = 72 диска.

Предыдущий пример показывает, что с точки зрения емкости было достаточно и 10 дисков, но чтобы отвечать требованиям приложения по производительности, нужны 72 диска. Для оптимизации требований по количеству дисков с точки зрения производительности в реальной обстановке при развертывании хранилища может быть принято множество различных решений. Примерами таких решений могут послужить выстраивание очереди команд, использование флеш-накопителей, RAID-массива и кэш-памяти. RAID-массивы и кэш-память более подробно рассматриваются в главах 3 и 4 соответственно.

2.11. Выстраивание очереди команд

Выстраивание очереди команд представляет собой технологию, реализованную на современных дисковых накопителях и определяющую очередность выполнения полученных команд на ввод-вывод с тем, чтобы сократить ненужные перемещения головок накопителя и повысить производительность диска. Когда на контроллер диска поступает команда на ввод-вывод данных, алгоритмы выстраивания очереди команд присваивают ей тег, определяющий последовательность, в которой должны быть выполнены команды. Эти команды выполняются на основе организации данных на диске независимо от того порядка, в котором они поступили.

Наиболее распространенным алгоритмом выстраивания очереди команд является алгоритм оптимизации времени поиска. Команды выполняются на основе оптимизации перемещений головок чтения-записи, что может привести к изменению порядка выполнения команд. Без оптимизации времени поиска команды выполняются в том порядке, в котором они были получены. Например, как показано на рис. 2.16, *а*, команды выполняются в порядке A, B, C и D. Радиальное перемещение, требующееся для головки, чтобы выполнить команду C сразу же после выполнения команды A, меньше, чем оно бы было для выполнения команды B. При оптимизации времени поиска порядок выполнения команд (рис. 2.16, *б*) будет следующим: A, C, B и D.

Еще одним алгоритмом выстраивания очереди является оптимизация времени доступа. При использовании этого алгоритма команды с целью достижения оптимальной производительности выполняются на основе сочетания оптимизации времени поиска и анализа задержки, связанной с вращением диска.

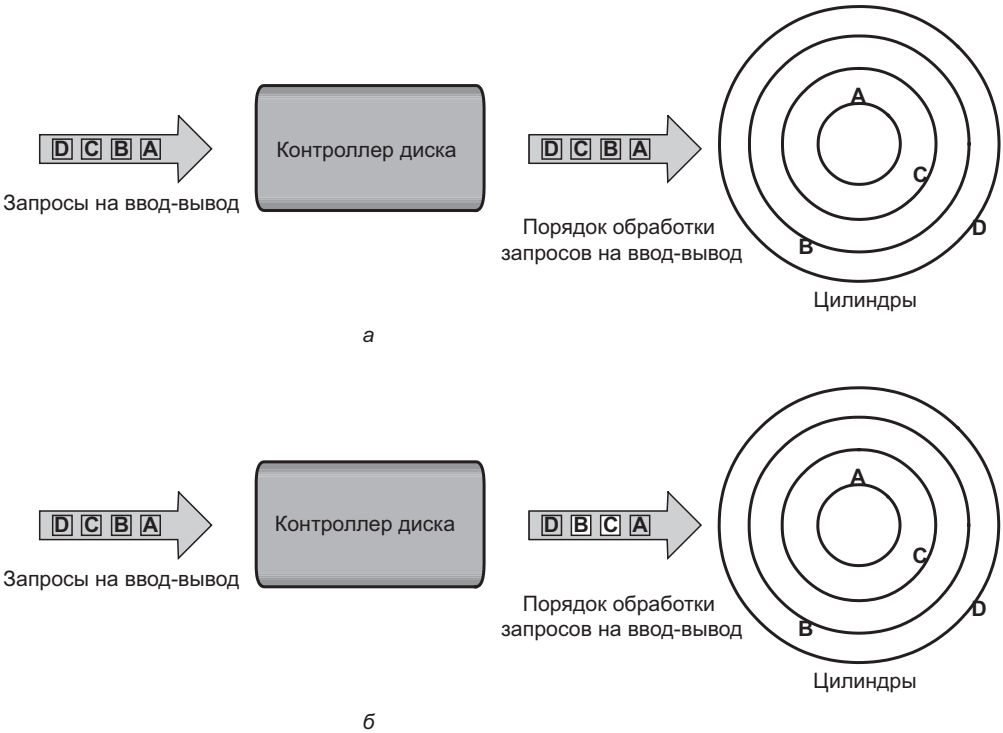


Рис. 2.16. Выстраивание диском очереди команд: *а* — без оптимизации; *б* — с оптимизацией времени поиска

Выстраивание очереди команд реализовано и на современных контроллерах дисковых массивов, что может стать дополнением к выстраиванию очереди команд, реализованному на дисковом накопителе.

2.12. Флеш-накопители

С ростом объемов информации пользователи хранилищ продолжают выдвигать постоянно возрастающие требования по повышению производительности для своих бизнес-приложений. Обычно высокие требования по операциям ввода-вывода удовлетворялись простым использованием большего количества дисков. С появлением флеш-накопителей корпоративного класса — enterprise class flash drives (EFD) сценарий изменился.

Флеш-накопители, известные также как твердотельные накопители — solid state drives (SSD), являются накопителями нового поколения, предоставляющими сверхвысокую производительность, которая требуется приложениям, особо чувствительным к этому параметру. Во флеш-накопителях для хранения и извлечения данных используется твердотельная полупроводниковая

память (флеш-память). В отличие от обычных механических дисковых накопителей, у флеш-накопителей нет движущихся частей, поэтому у них нет задержек на поиск и вращение диска. Флеш-накопители обеспечивают большое количество IOPS в сочетании с весьма низким временем отклика. Кроме того, будучи устройством на основе полупроводников, флеш-накопители по сравнению с механическими накопителями потребляют меньше энергии. Флеш-накопители особенно подходят для приложений с небольшим размером блока и рабочим циклом, связанным с произвольным чтением, требующим неизменно низкого времени отклика (менее 1 мс). От применения флеш-накопителей существенно выигрывают те приложения, которым требуется быстро обрабатывать большие массивы информации, например приложения по обмену валюты, электронные торговые системы и приложения, предоставляющие данные в реальном масштабе времени.

По сравнению с обычными механическими дисковыми накопителями, EFD предлагают существенное увеличение пропускной способности — вплоть до 30 раз — и снижение времени отклика до одной десятой (<1 мс в сравнении с 6–10 мс). Кроме того, флеш-накопители могут хранить данные, затрачивая на это до 38 % меньше энергии на 1 Тбайт, чем традиционные дисковые накопители, что приводит приблизительно к 98 %-ной экономии электроэнергии, затрачиваемой на проведение операций ввода-вывода.

В целом, флеш-накопители обеспечивают более высокий показатель совокупной стоимости владения — total cost of ownership (TCO), даже при том, что они по показателю стоимости хранения 1 Гбайт информации обходятся дороже. Внедряя флеш-накопители, потребители могут обеспечить соответствие требованиям приложений по производительности с использованием намного меньшего количества устройств (примерно в 20–30 раз по сравнению с обычными механическими накопителями). Такое сокращение не только приводит к экономии средств на приобретение накопителей, но и выливается в экономию расходов на электропитание, охлаждение и рабочие площади. Чем меньше в среде хранения данных количество отдельных накопителей, тем ниже затраты на обслуживание хранилища.

2.12.1. Компоненты и архитектура флеш-накопителей

Для обеспечения совместимости во флеш-накопителях используются такой же физический форм-фактор и такие же разъемы, как и в дисковых накопителях. Это позволяет упростить замену механического дискового накопителя в стойке массива хранения данных флеш-накопителем. Основными компонентами флеш-накопителя являются контроллер, интерфейс ввода-вывода, запоминающее устройство (набор микросхем памяти) и кэш. Контроллер управляет функционированием накопителя, а интерфейс ввода-вывода предоставляет доступ к электропитанию и данным. Запоминающее устройство представляет собой массив энергонезависимых микросхем памяти

NAND-типа, используемых для хранения данных. Кэш служит как временное пространство или буфер для транзакций данных и операций.

Для доступа к данным флеш-накопитель использует несколько параллельных каналов ввода-вывода (от его контроллера накопителя к микросхемам флеш-памяти). Как правило, чем больше количество микросхем флеш-памяти и каналов, тем шире полоса пропускания накопителя, а в итоге выше производительность накопителя. У флеш-накопителей обычно имеется от 8 до 24 каналов.

Логически микросхемы памяти флеш-накопителей организованы в блоки и страницы. Страница является наименьшим объектом, который может быть считан или записан на флеш-накопитель. Страницы сгруппированы в блоки. (Эти блоки не следует путать с 512-байтными блоками секторов механического дискового накопителя.) В блоке может иметься 32, 64 или 128 страниц. У страниц нет стандартного размера, обычно их размер составляет 4, 8 и 16 Кбайт. Поскольку флеш-накопители эмулируют механические накопители, использующие логическую адресацию блоков (LBA), страница охватывает ряд последовательных блоков данных. Например, страница размером 4 Кбайт будет охватывать восемь блоков данных по 512 байт с последовательными адресами. Во флеш-накопителях операция чтения может проходить на уровне страниц, а операция записи или операция удаления происходит только на уровне блоков.

2.12.2. Свойства флеш-накопителей корпоративного класса

Основными свойствами флеш-накопителей корпоративного класса являются:

- **NAND-технология флеш-памяти.** Технология NAND-памяти хорошо подходит для доступа к произвольным данным. Для обеспечения целостности данных и более высоких скоростей записи в NAND-устройствах используются отслеживание поврежденных блоков и код исправления ошибок — error-correcting code (ECC);
- **флеш-память с одноуровневыми ячейками — Single-Level Cell (SLC).** В NAND-технологии доступны две разные конструкции ячеек. Многоуровневая ячейка — multi-level cell (MLC) хранит более одного бита на ячейку в силу своей способности регистрировать несколько состояний, а одноуровневая ячейка может хранить только 1 бит. Для корпоративных приложений, работающих с данными, предпочтительнее SLC-технология, поскольку при ее использовании ячейки обладают более высокой производительностью и долговечностью. Скорость считывания при технологии SLC обычно вдвое быстрее, чем на устройствах, выполненных по технологии MLC, а скорость записи выше почти вчетверо. По сравнению с MLC-конструкциями у SLC-устройств обычно в 10 раз больше циклов записи-стирания. Кроме того, флеш-память, выполненная по технологии SLC, имеет более

высокую надежность, поскольку в ее ячейке хранится всего 1 бит. Следовательно, вероятность ошибки уменьшается;

- **технология выравнивания записи.** Важной составляющей максимального продления жизненного срока флеш-накопителя является обеспечение равномерного использования отдельных ячеек памяти. Это означает, что часто обновляемые данные во избежание случаев перезаписи одних и тех же ячеек записываются в разные места. Конструкция флеш-накопителей корпоративного класса обеспечивает использование для каждой новой операции записи наименее задействованного ранее блока.

2.13. Применение концепции на практике: VMware ESXi

Компания VMware является лидером в предоставлении решений виртуализации серверов. Ее продукт VMware ESXi предоставляет платформу, называемую гипервизором. Гипервизор создает абстракцию центрального процессора, памяти и ресурсов хранения данных с целью обеспечения одновременной работы нескольких виртуальных машин на одном и том же физическом сервере.

VMware ESXi является гипервизором, который устанавливается на оборудовании, имеющем архитектуру x86 с целью обеспечения виртуализации сервера. Он позволяет создавать несколько виртуальных машин (VM), которые могут работать одновременно на одной и той же физической машине. Виртуальная машина является отдельным набором файлов, который может быть перемещен, скопирован и использован в качестве шаблона. Все файлы, составляющие виртуальную машину, обычно хранятся в одном каталоге в кластерной файловой системе, которая называется файловой системой виртуальной машины — Virtual Machine File System (VMFS). Физическая машина, на которой базируется ESXi, называется ESXi-хостом. ESXi-хосты предоставляют физические ресурсы, используемые для работы виртуальных машин. У ESXi имеется два основных компонента: ядро виртуальной машины — VMkernel и монитор виртуальной машины — Virtual Machine Monitor.

VMkernel предоставляет функциональные возможности, которые можно найти в других операционных системах, такие как создание процесса, управление файловой системой и диспетчеризация процессов. Это ядро разработано именно для того, чтобы поддерживать несколько работающих виртуальных машин и предоставлять основные функциональные возможности, такие как диспетчеризация ресурсов, организация стеков ввода-вывода и т. д.

Монитор виртуальной машины отвечает за выполнение команд на центральном процессоре и выполнение двоичной трансляции — Binary Translation (BT). Монитор виртуальной машины выполняет аппаратные абстракции, чтобы предстать в виде физической машины со своими собственными центральным процессором, памятью и устройствами ввода-вывода.

Монитор виртуальной машины назначается каждой виртуальной машине, которая совместно с другими виртуальными машинами использует центральный процессор, память и устройства ввода-вывода, чтобы обеспечить ее успешную работу.

Резюме

В этой главе были подробно рассмотрены основные компоненты среды дата-центра — приложение, система управления базами данных, хост, система передачи данных и хранилище. Через эти компоненты проходят потоки данных от приложения к хранилищу. Физические и логические составляющие этих компонентов оказывают влияние на общую производительность приложения. Виртуализация различных компонентов дата-центра повышает коэффициент их использования и упрощает управление ими.

Основным компонентом среды дата-центра является хранилище. Наиболее популярным устройством хранения данных является дисковый накопитель, в котором для доступа к данным и их хранения используются магнитные носители. Последним нововведением, по многим параметрам превосходящим механические дисковые накопители, являются твердотельные накопители (SSD) на основе флеш-памяти.

Чтобы всецело отвечать требованиям приложений по производительности, в современных дисковых системах хранения данных используются сотни дисков. Управление емкостью, производительностью и надежностью такого большого количества дисков является непростой задачей. Вполне приемлемой технологией управления емкостью, производительностью и надежностью дисковых накопителей является массив независимых дисковых накопителей. Эта технология обладает избыточностью данных — RAID и более подробно будет рассмотрена в следующей главе.

УПРАЖНЕНИЯ

1. Какими преимуществами обладает виртуализированный дата-центр по сравнению с классическим?
2. Приложению предъявляются требования по емкости хранилища, составляющие 200 Гбайт, для размещения базы данных и других файлов. Также в технических условиях указано, что среда хранения данных должна поддерживать при пиковой нагрузке 5000 IOPS. Доступные для составления хранилища диски предоставляют по 66 **Гбайт полезной емкости, а их производитель заявил, что они в состоянии под**держивать максимум 140 IOPS. Приложение весьма чувствительно к времени отклика, а коэффициент использования, превышающий 60 %, не отвечает требованиям по этому показателю. Вычислите минимальное количество дисков, необходимое для составления конфигурации, отвечающей запросам приложения, и поясните свои теоретические выкладки.

3. Из каких компонентов состоит время обслуживания диском запроса на ввод-вывод? Какой из компонентов имеет в общем времени наибольший процентный показатель при обслуживании диском операции произвольного ввода-вывода?
4. Средний объем ввода-вывода приложения составляет 64 Кбайт. Производитель диска сообщает следующие характеристики накопителя: среднее время поиска 5 мс, скорость вращения шпинделя 7200 об./мин и скорость передачи данных 40 Мбайт/с. Определите максимальный показатель IOPS, который может быть достигнут с этим диском для приложения. Используя этот случай в качестве примера, объясните взаимосвязь между коэффициентом использования диска и показателем IOPS.
5. Обратитесь к упражнению 4 еще раз. На основании вычисленного времени обслуживания запроса на ввод-вывод постройте график, показывающий зависимость времени отклика от коэффициента использования, рассмотрев при этом коэффициент использования контроллера ввода-вывода 20, 40, 60, 80 и 100 %. Какие выводы можно сделать на основе этого графика?
6. Исследуйте другие компоненты дата-центра, рассмотренные в данной главе и не входящие в перечень основных, включив в их состав такие параметры управления средой, как теплоснабжение, вентиляция и кондиционирование воздуха — HVAC (heat, ventilation, and air-condition), электропитание и безопасность.

Глава 3
Защита данных: RAID-массив

КЛЮЧЕВЫЕ ПОНЯТИЯ

- RAID-массивы, реализуемые аппаратными и программными средствами
- Чередование, зеркалирование и контроль четности
- RAID-уровни
- Издержки записи в RAID-массив
- Горячее резервирование

В конце 1980-х годов стремительное внедрение компьютеров в коммерческую деятельность способствовало распространению новых приложений и баз данных, значительно увеличивая требования к емкости и производительности запоминающих устройств. В то время данные хранились на одном большом и дорогом SLED-диске (Single Large Expensive Drive). Одиночные диски не соответствовали требуемому уровню производительности, поскольку они могли обслуживать только ограниченное количество запросов на ввод-вывод.

Современные дата-центры содержат в своей инфраструктуре хранения данных сотни дисковых накопителей. У жестких дисков в силу их конструкции возможны сбои из-за механического износа и других факторов внешней среды, что может привести к потере данных. Чем больше дисковых накопителей в массиве, тем больше вероятность выхода из строя одного из них. Например, возьмем массив из 100 дисков при среднем периоде безотказной работы каждого 750 000 часов. Средний период безотказной работы всего массива составляет 7500 часов (750 000/100). Это означает, что в среднем через 7500 часов один из жестких дисков массива выходит из строя.

Технология RAID позволяет использовать несколько накопителей как часть комплекта, защищающего данные от выхода из строя жестких дисков. В целом RAID-конструкции повышают производительность систем хранения данных, обслуживая запросы на ввод-вывод одновременно с нескольких дисков. Современные массивы, оснащенные флеш-накопителями при

использовании RAID-технологий, также получают преимущества с точки зрения защиты и производительности.

В 1987 году Паттерсон, Гибсон и Катц из Калифорнийского университета в Беркли опубликовали научную работу под названием «Применение избыточного массива недорогих дисков (RAID)». В этой работе рассказывалось об использовании недорогих дисков малых объемов в качестве альтернативы объемным дискам, применявшимся в центральных компьютерах вычислительного центра. С тех пор значение термина RAID изменилось. Так стали называть независимые диски, отмечая развитие технологии хранения данных. RAID-технология из научной концепции переросла в промышленный стандарт, часто реализуемый в современных массивах хранения данных.

В этой главе представлена подробная информация о технологии RAID, RAID-уровнях, различных типах реализации RAID и их преимуществах.

3.1. Методы реализации RAID

RAID-массивы реализуются двумя методами: аппаратным и программным. У этих реализаций есть свои преимущества и недостатки, рассматриваемые в данном разделе.

3.1.1. Реализация RAID программными методами

RAID-массивы, реализуемые программными методами, используют для предоставления RAID-функций программы, имеющиеся на хост-машине. Реализация происходит на уровне операционной системы без применения для управления RAID-массивом специального контроллера.

RAID-массивы, реализуемые программными методами, зачастую дешевле и проще в эксплуатации по сравнению с RAID-массивами, реализуемыми аппаратными методами, но при этом у них есть следующие ограничения.

- **По производительности** — реализация RAID программными методами снижает общую производительность системы. Причина в дополнительных циклах центрального процессора, необходимых для проведения RAID-вычислений.
- **По поддерживаемым функциям** — реализация RAID программными методами не поддерживает все RAID-уровни.
- **По совместимости с операционной системой** — реализация RAID программными методами привязана к операционной системе хоста, следовательно, обновления программного обеспечения RAID или операционной системы требуют проверки на совместимость. В результате среда обработки данных теряет гибкость.

3.1.2. Реализация RAID аппаратными методами

RAID-массивы, реализуемые аппаратными методами, используют специализированный аппаратный контроллер либо на хост-машине, либо на самом RAID-массиве.

RAID-массив на основе платы контроллера представляет собой аппаратную реализацию RAID за счет установленного на хост-машине специализированного RAID-контроллера, к которому подключены дисковые накопители. Поставляются также материнские платы со встроенным RAID-контроллером. В среде дата-центра с большим количеством хостов использование RAID-контроллеров, размещенных на хосте, считается нецелесообразным.

Внешний RAID-контроллер является аппаратным методом реализации RAID, при котором оборудование находится на самом массиве и служит интерфейсом между хостом и дисками. Он предоставляет тома запоминающего устройства хосту, а хост управляет этими томами как физическими накопителями. RAID-контроллер выполняет следующие основные функции:

- управление группами дисков и контроль их работы;
- преобразование запросов на ввод-вывод между логическими и физическими дисками;
- восстановление данных при сбоях дисков.

3.2. Компоненты RAID-массива

RAID-массив представляет собой корпус, содержащий несколько дисковых накопителей и вспомогательное оборудование для реализации RAID-технологии. Подгруппы дисков внутри RAID-массива могут быть объединены в логические объединения, называемые логическими массивами, также известными как RAID-наборы, или RAID-группы (рис. 3.1).

3.3. Методы RAID

Методы RAID — чередование, зеркалирование и контроль четности — задают основу для определения различных RAID-уровней. Эти методы определяют степень доступности данных и характеристики производительности RAID-набора.

3.3.1. Чередование

Чередование (striping) — это метод распространения данных по нескольким накопителям для их параллельного использования. Все головки чтения-

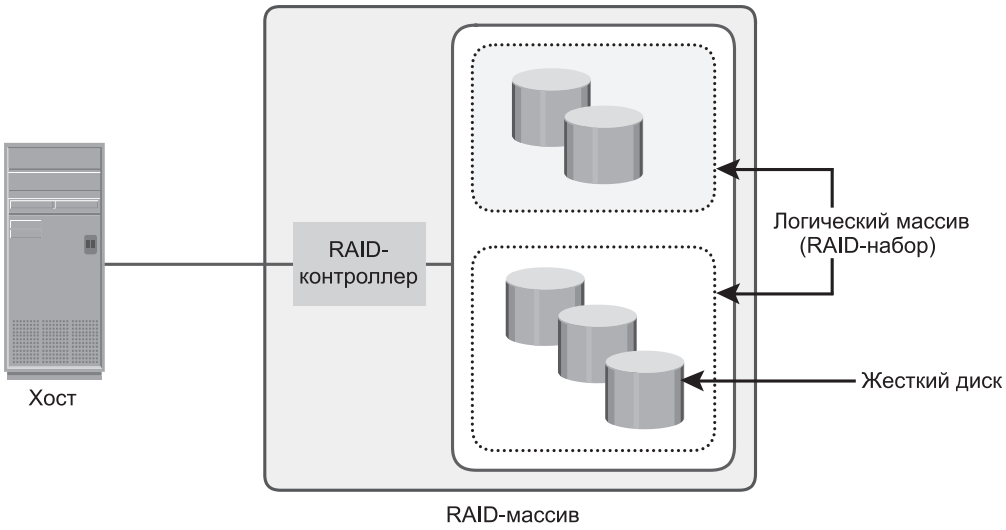


Рис. 3.1. Компоненты RAID-массива

записи работают одновременно, позволяя за меньшее время обрабатывать больше данных и повысить производительность по сравнению с чтением или записью на одном диске.

В каждом диске RAID-набора имеется предопределенное количество непрерывно адресуемых блоков, определяемых в качестве полосы, или стрипа (strip). Группа располагающихся на одной линии полос, охватывающая все диски RAID-набора, называется дорожкой, или страйпом (stripe). Физическое и логическое представления RAID-набора с чередованием показаны на рис. 3.2.

Размер полосы (также называемый глубиной дорожки) описывает количество блоков в полосе и максимальный объем данных, который может быть записан на один жесткий диск в наборе или считан с него при условии, что начало данных, к которым идет обращение, совпадает с началом полосы. Все полосы в дорожке имеют одинаковые номера блоков. Меньший размер полосы означает, что при распределении по дискам данные разбиваются на меньшие части.

Размер дорожки представляет собой произведение размера полосы на количество дисков для данных в RAID-наборе. Например, у входящих в RAID-набор пяти дисков с чередованием, при размере полосы, равной 64 Кбайт, размер дорожки получается 320 Кбайт (64 Кбайт · 5). Ширина дорожки — это количество полос данных в дорожке. В следующих разделах будет показано, что RAID-массив с чередованием не обеспечивает защиты данных, если только не применяется зеркалирование или контроль четности.

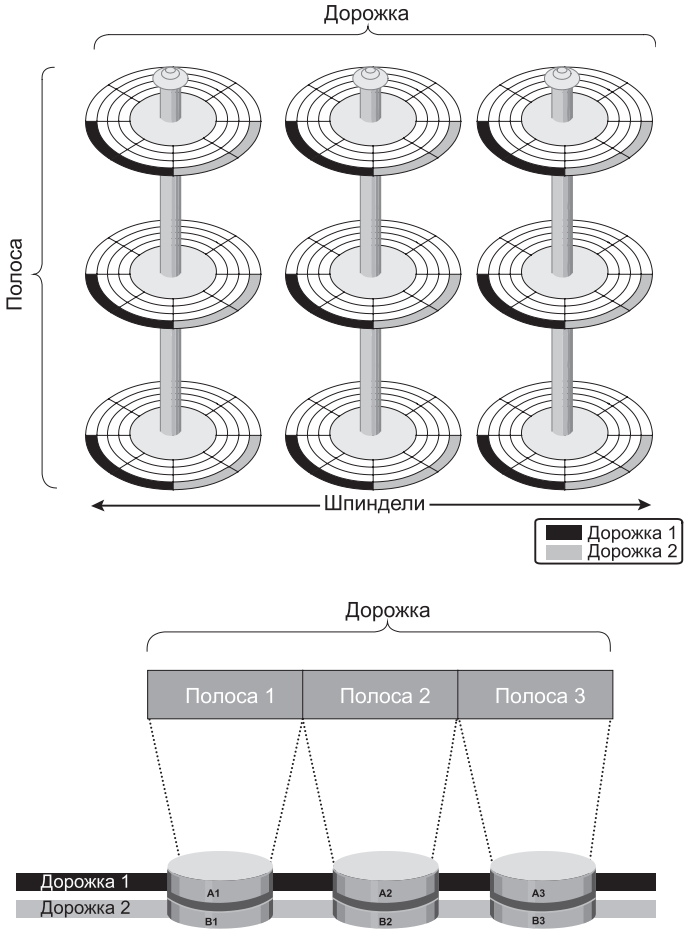


Рис. 3.2. RAID-группа с чередованием

3.3.2. Зеркалирование

Зеркалирование — это технология хранения одних и тех же данных на двух разных дисковых накопителях, в результате применения которой получаются две копии данных. При выходе из строя одного дискового накопителя данные остаются целыми на дисковом накопителе, сохранившем работоспособность (рис. 3.3), а контроллер продолжает обработку запросов хоста с применением уцелевшего диска зеркалированной пары.

Когда поврежденный диск заменяется новым, контроллер копирует данные с неповрежденного диска зеркалированной пары. Эта операция проводится незаметно для хоста.

Помимо предоставления полной избыточности данных зеркалирование позволяет быстро восстанавливать данные после отказа диска. Но оно

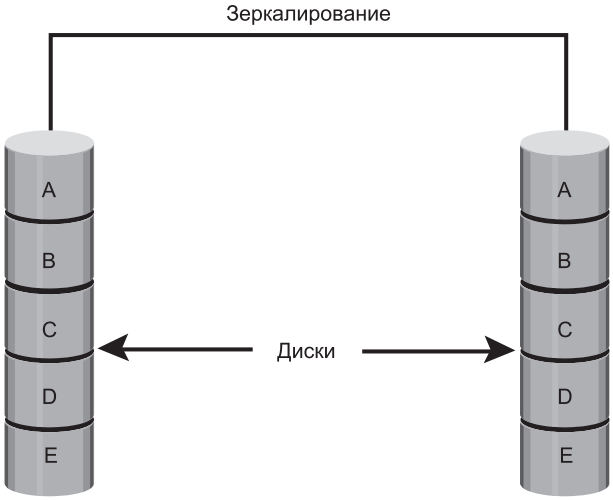


Рис. 3.3. Зеркалированные диски в массиве

обеспечивает лишь защиту данных и не может заменить операцию резервного копирования. Зеркалирование постоянно записывает изменения в данных, а резервное копирование фиксирует образы данных в определенный момент времени.

Зеркалирование приводит к дублированию данных, при этом требуемый объем хранилища вдвое превышает объем сохраняемой информации. Поэтому зеркалирование считается весьма дорогостоящей технологией, которую предпочтительно применять для крайне важных приложений, не допускающих потери данных. Зеркалирование повышает скорость чтения данных, поскольку запросы на чтение могут обрабатываться обоими дисками. Но по сравнению с записью на один диск скорость записи немного снижается, поскольку каждый запрос на запись превращается в две записи на дисковые накопители. Зеркалирование не дает таких же уровней производительности записи, как RAID-массив с чередованием.

3.3.3. Контроль четности

Контроль четности — это метод защиты чередуемых данных от выхода из строя дисковых накопителей без затрат на зеркалирование. К набору добавляется дополнительный дисковый накопитель, используемый для хранения данных контроля четности, математического построения, позволяющего восстанавливать утраченные данные. Контроль четности является избыточным методом, обеспечивающим защиту данных без обслуживания полного набора продублированных данных. Вычисление данных контроля четности возлагается на RAID-контроллер.

Информация контроля четности может храниться на отдельных, специально выделенных дисковых накопителях или же распределяться по всем дискам RAID-набора. На рис. 3.4 показан RAID-массив с контролем четности. Первые четыре диска, помеченные надписью «Диски с хранящимися данными», содержат обычные данные. Пятый диск, помеченный надписью «Диск с данными контроля четности», содержит информацию о контроле четности, которая в данном случае является суммой элементов в каждом ряду. При отказе одного из дисков с обычными данными утраченное значение вычисляется вычитанием суммы оставшихся элементов из контрольного значения.

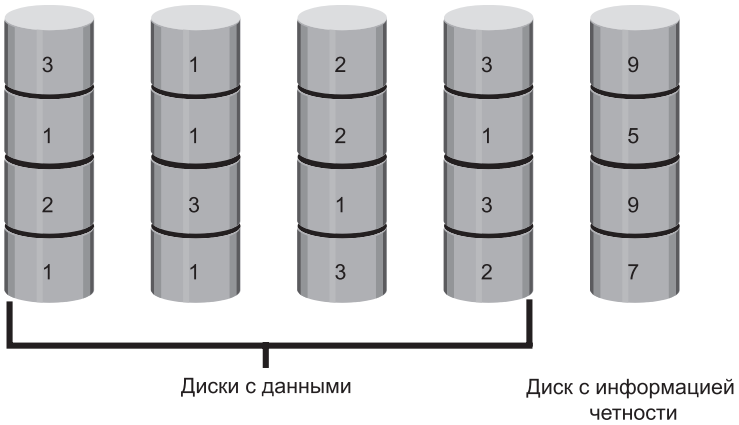


Рис. 3.4. RAID-массив с контролем четности

Здесь для простоты объяснения вычисление контрольных значений представлено арифметической суммой данных. Но на самом деле вычисление данных контроля четности представляет собой побитовую операцию XOR (логическое исключающее *ИЛИ*).

По сравнению с зеркалированием контроль четности существенно снижает стоимость защиты данных. Рассмотрим конфигурацию RAID-массива с контролем четности, который состоит из пяти дисков, где четыре диска содержат данные, а пятый — контрольную информацию. По сравнению с зеркалированием, требующим 100 % дополнительного дискового пространства, нам потребуется только 25 %. Тем не менее у технологии контроля четности имеется ряд недостатков. Контрольная информация создается на основе данных, расположенных на диске с данными. Следовательно, при каждом изменении данных происходит перерасчет. Этот перерасчет занимает время и снижает производительность RAID-массива.

Для RAID-массива с контролем четности вычисление размера дорожки не включает полосу контроля четности. Например, если RAID-массив с контролем четности состоит из 5 дисков (4 + 1) с размером полосы 64 Кбайт, размер дорожки будет равен 256 Кбайт (64 Кбайт · 4).

ОПЕРАЦИЯ XOR

Побитовая операция исключающего ИЛИ (XOR) получает две битовые последовательности одинаковой длины и проводит логическую операцию XOR над каждой парой соответствующих битов. В результате в каждой позиции получается 1, если значения двух битов отличаются друг от друга, и 0, если они оказываются одинаковыми. Ниже показана таблица истинности операции XOR. (A и B обозначают входные данные, а C — выход после выполнения операции XOR.) Если какие-либо данные из A, B или C утрачиваются, они могут быть воспроизведены путем проведения операции XOR над доступными оставшимися данными. Например, если диск, содержащий все данные из A, выходит из строя, эти данные могут быть восстановлены путем выполнения операции XOR над B и C.

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

3.4. RAID-уровни

Решение о выборе RAID-уровня принимается на основе производительности приложения, требований по доступности данных и по допустимым затратам на их хранение. RAID-уровни определяются на основе технологий чередования, зеркалирования и контроля четности. В некоторых RAID-уровнях используется только одна их этих технологий, а в других — их сочетание. Самые распространенные RAID-уровни показаны в табл. 3.1.

Таблица 3.1. RAID-уровни

УРОВНИ	КРАТКОЕ ОПИСАНИЕ
RAID 0	Массив с чередованием, не обеспечивающий отказоустойчивости
RAID 1	Массив с зеркалированием дисков
Вложенный	Комбинации RAID-уровней. Пример: RAID 1 + RAID 0
RAID 3	Массив с чередованием, параллельным доступом к данным и выделенным диском контроля четности
RAID 4	Массив с чередованием, независимым доступом к дискам и выделенным диском контроля четности
RAID 5	Массив с чередованием, независимым доступом к дискам и рассредоточенной информацией контроля четности
RAID 6	Массив с чередованием, независимым доступом к дискам и двойным рассредоточением информации контроля четности

3.4.1. RAID 0

В конфигурации RAID 0 используется технология чередования данных, при которой они распространяются по всем дискам RAID-массива. Поэтому в данной конфигурации используется весь объем запоминающего устройства RAID-набора. Чтобы считать данные, контроллер собирает их со всех полос. На рис. 3.5 показана используемая в массиве технология RAID 0, при которой данные чередуются по пяти дискам. Когда количество накопителей в RAID-наборе увеличивается, производительность возрастает, поскольку появляется возможность одновременного считывания и записи большего количества данных. RAID нулевого уровня хорошо подходит для приложений, требующих высокой пропускной способности операций ввода-вывода. Но если от приложений требуется высокая степень доступности в случае выхода из строя накопителей, то защиту и доступность данных RAID 0 обеспечить не сможет.

Рис. 3.5. RAID 0

3.4.2. RAID 1

Технология, применяемая в RAID 1, основана на зеркалировании. Данные в этой RAID-конфигурации с целью повышения отказоустойчивости

зеркалируются (рис. 3.6). Набор RAID 1 состоит из двух дисковых накопителей, и каждая запись ведется сразу на оба диска. Для хоста зеркалирование остается незамеченным. Среди всех RAID-реализаций в RAID 1 усилия, предпринимаемые для восстановления данных в случае отказа диска, являются наименьшими. Дело в том, что для восстановления данных RAID-контроллер использует их зеркальную копию на другом диске. Конфигурация RAID 1 хорошо подходит для приложений, требующих высокой работоспособности при отсутствии ограничений по финансовым затратам.

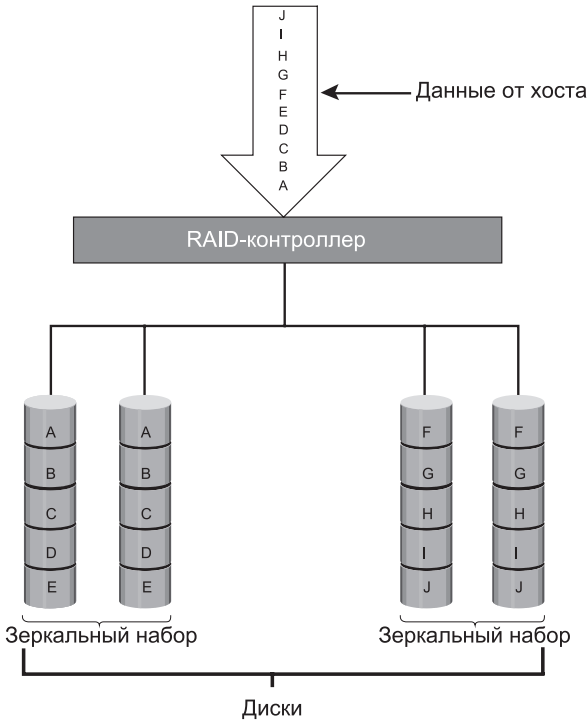


Рис. 3.6. RAID 1

3.4.3. Вложенный RAID

Большинству дата-центров от их RAID-массивов требуется как избыточность данных, так и высокий уровень производительности. Комбинации RAID 0+1 и RAID 1+0 сочетают в себе преимущества как высокой производительности RAID 0, так и избыточности RAID 1. В них применяются методы чередования и зеркалирования, что в результате дает сочетание их преимуществ. Эти RAID-конфигурации требуют четного количества дисков при минимуме 4 единицы (рис. 3.7).

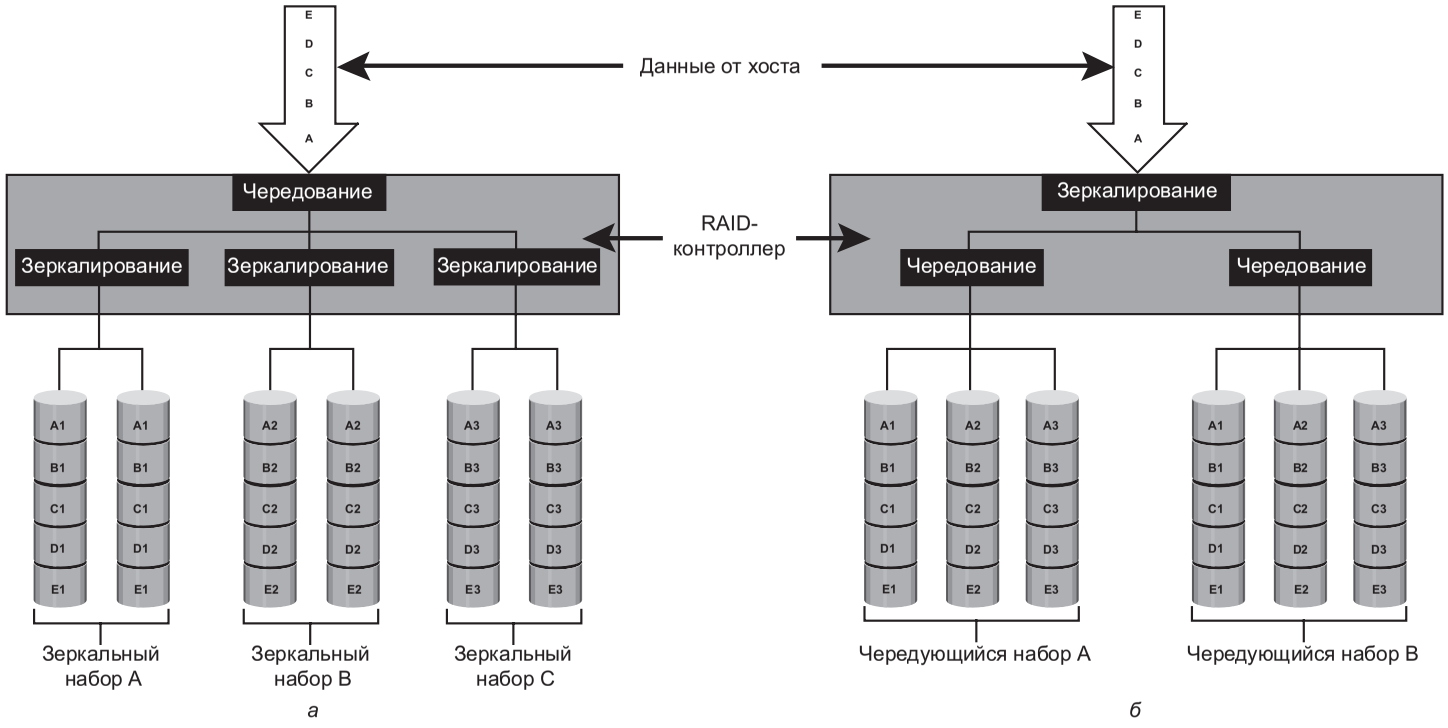


Рис. 3.7. Вложенный RAID

Конфигурация RAID 1+0 также известна как RAID 10 (десять), или RAID 1/0. По аналогии с этим конфигурация RAID 0+1 также известна как RAID 01, или RAID 0/1. Конфигурация RAID 1+0 отлично подойдет для проведения небольших операций ввода-вывода с произвольным доступом и частыми записями данных. К числу приложений, получающих преимущества от использования RAID 1+0, можно отнести:

- приложения, выполняющие большой объем оперативной обработки сетевых транзакций — Online Transaction Processing (OLTP);
- приложения, входящие в крупные системы обмена сообщениями;
- приложения баз данных с рабочей нагрузкой, связанной с частыми записями с произвольным доступом.

Существует ошибочное мнение, что RAID 1+0 и RAID 0+1 — это одно и то же. В обычных условиях RAID-конфигурации уровней 1+0 и 0+1 предлагают одинаковые преимущества. Но в случае отказа диска операции восстановления у них различаются.

Конфигурацию RAID 1+0 также называют полосатым зеркалом. Основным элементом RAID 1+0 является зеркальная пара. Это означает, что сначала данные зеркалируются, а затем обе копии данных чередуются по RAID-группе в нескольких парах дисковых накопителей. При замене отказавшего диска восстанавливается только зеркальная запись. Другими словами, контроллер дискового массива использует неповрежденный диск зеркальной пары для восстановления данных и обеспечения непрерывной работы. Данные с уцелевшего диска копируются на диск, устанавливаемый в качестве замены отказавшему.

Чтобы разобраться в работе RAID 1+0, рассмотрим пример набора из шести дисков (сначала RAID 1, затем RAID 0). Эти шесть дисков разбиты на три пары, где каждая пара работает как набор RAID 1 (как зеркалированная пара дисков). Затем данные чередуются по всем трем зеркалированным наборам, чтобы составить конфигурацию RAID 0. Следующая последовательность показывает пошаговые действия, выполняемые в RAID 1+0 (см. рис. 3.7, *а*):

Накопители 1 + 2 = RAID 1 (Зеркальный набор A);

Накопители 3 + 4 = RAID 1 (Зеркальный набор B);

Накопители 5 + 6 = RAID 1 (Зеркальный набор C).

Теперь в наборах с A по C выполняется чередование RAID 0. При такой конфигурации отказ накопителя 5 отразится только на наборе C. В нем еще остается продолжающий функционировать накопитель 6, благодаря чему сохраняется работоспособность всего массива RAID 1+0. Теперь предположим, что отказал накопитель 3, а накопитель 5 был заменен новым накопителем. В этом случае массив по-прежнему будет сохранять работоспособность, потому что накопитель 3 принадлежит другому зеркальному набору. Следовательно, при такой конфигурации без ущерба для массива

могут отказать до трех накопителей при условии, что все они принадлежат разным зеркальным наборам.

Конфигурацию RAID 0+1 называют также зеркалированной дорожкой. Базовым элементом в данном случае является дорожка. Это означает, что вначале выполняется процесс, обеспечивающий чередование данных по разным дисковым накопителям, а затем зеркалируется вся дорожка. В данной конфигурации при отказе одного накопителя происходит отказ всей дорожки. Чтобы разобраться с работой RAID 0+1 (то есть сначала RAID 0, а затем RAID 1), рассмотрим такой же пример с шестью дисковыми накопителями. В данном случае шесть дисков сгруппированы в два набора по три диска в каждом. В свою очередь, каждый из этих наборов действует как набор RAID 0, содержащий три диска, а затем эти два набора зеркалируются, чтобы сформировался массив RAID 1. Следующая последовательность показывает пошаговые действия, выполняемые в RAID 0+1 (см. рис. 3.7, *б*):

Накопители 1 + 2 + 3 = RAID 0 (Набор дорожек A);

Накопители 4 + 5 + 6 = RAID 0 (Набор дорожек B).

Теперь эти два набора дорожек зеркалируются. Если один из накопителей, скажем накопитель 3, откажет, неработоспособен будет и весь набор дорожек A. При выполнении операции восстановления копируется вся дорожка, то есть данные копируются с каждого диска, составляющего неповрежденную дорожку, на соответствующий ему диск из состава ранее отказавшей дорожки. При этом увеличивается объем ненужных операций ввода-вывода на неповрежденных дисках, а RAID-массив становится более уязвимым к повторному отказу накопителя.

3.4.4. RAID 3

В RAID 3 данные чередуются для увеличения производительности, а контроль четности применяется для повышения отказоустойчивости. Контрольная информация для восстановления данных хранится на специально выделенном диске. Например, 4 диска из 5 используются для хранения данных, а один — для контрольной информации. Следовательно, необходимый объем общего дискового пространства составляет 1,25 от объема дискового пространства, предназначенного для хранения данных. RAID 3 всегда считывает и записывает полные дорожки данных на все диски, поскольку диски работают параллельно. Частичные записи при обновлении одной из полос дорожки невозможны. Вариант реализации RAID 3 показан на рис. 3.8.

RAID 3 предоставляет высокий уровень производительности для приложений, предусматривающих последовательное обращение к данным большого объема, например для создания резервных копий или обработки видеопотоков.

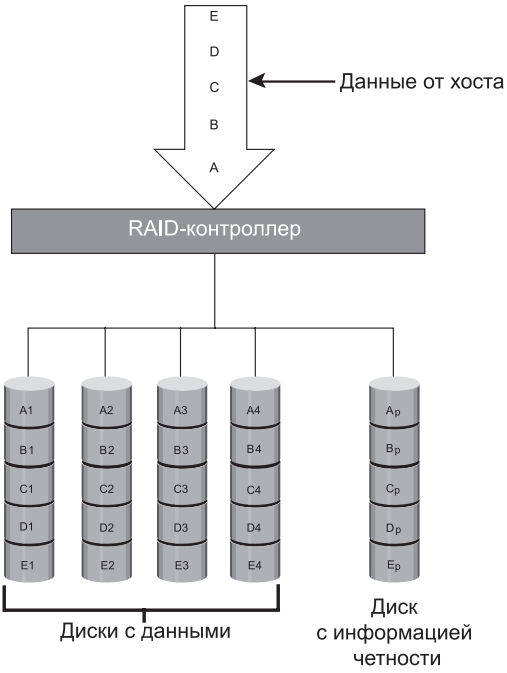


Рис. 3.8. RAID 3

3.4.5. RAID 4

Как и в RAID 3, в RAID 4 данные чередуются для увеличения производительности, а контроль четности применяется для повышения отказоустойчивости. Данные чередуются по всем дискам массива, за исключением диска с контрольной информацией. Эта информация хранится на специально выделенном диске, позволяя восстановить данные в случае отказа накопителя.

В отличие от RAID 3, в RAID 4 доступ к дискам данных может осуществляться независимо, поэтому конкретные элементы данных можно считывать или записывать на отдельный диск без считывания или записи всей дорожки. Конфигурация RAID 4 обеспечивает хорошую пропускную способность при чтении и приемлемую скорость записи.

3.4.6. RAID 5

У RAID 5 может быть весьма разнообразная реализация. Конфигурация этого массива похожа на RAID 4, поскольку использует сегментирование при независимом доступе к дискам (полосам). Разница между 4-й и 5-й конфигурациями заключается в размещении контрольных данных. В RAID 4 они записываются на отдельный диск, превращая запись на этот диск в критическую операцию. Для преодоления этого узкого места в RAID 5 контрольная

информация распределяется по всем дискам. Вариант реализации RAID 5 показан на рис. 3.9.

Рис. 3.9. RAID 5

RAID 5 хорошо подходит для приложений, использующих произвольный доступ к накопителям с большим объемом операций чтения данных, и наиболее предпочтителен для обслуживания источников информации средней производительности, систем обмена сообщениями и систем извлечения данных, а также для реализации систем управления реляционными базами данных — relational database management system (RDBMS), в которых оптимизацию доступа к данным осуществляют администраторы баз данных — database administrators (DBA).

3.4.7. RAID 6

RAID 6 работает точно так же, как и RAID 5, за исключением того, что в RAID 6 включен второй элемент контроля четности, чтобы массив смог выжить в случае отказа двух дисков в RAID-наборе (рис. 3.10). Поэтому для реализации RAID 6 требуется минимум 4 диска. В RAID 6 контрольная информация распределяется по всем дискам. Издержки записи (рассматриваемые далее) в RAID 6 выше, чем в RAID 5, поэтому производительность записи у RAID 5 выше, чем у RAID 6. Из-за наличия двойного набора контрольной

информации операция восстановления у RAID 6 может занять больше времени, чем у RAID 5.

Рис. 3.10. RAID 6

3.5. Влияние RAID на производительность диска

При выборе RAID-конфигурации нужно обязательно учитывать ее влияние на производительность диска и количество операций, выполняемых приложениями (IOPS).

В обеих RAID-конфигурациях, как с зеркалированием, так и с контролем четности, каждая операция записи вызывает для дисков дополнительные затраты на ввод-вывод, которые называются издержками записи (write penalty). При реализации RAID 1 каждая операция записи должна выполняться на два диска, составляющих зеркальную пару, а при реализации RAID 5 одна операция записи может проявляться как 4 операции ввода-вывода. При выполнении операций ввода-вывода на диск, входящий в состав RAID 5, контроллер вынужден прочитать, пересчитать и записать сегмент четности для каждой операции записи данных.

На рис. 3.11 изображена единичная операция записи в хранилище с конфигурацией RAID 5, состоящее из 5 дисков.

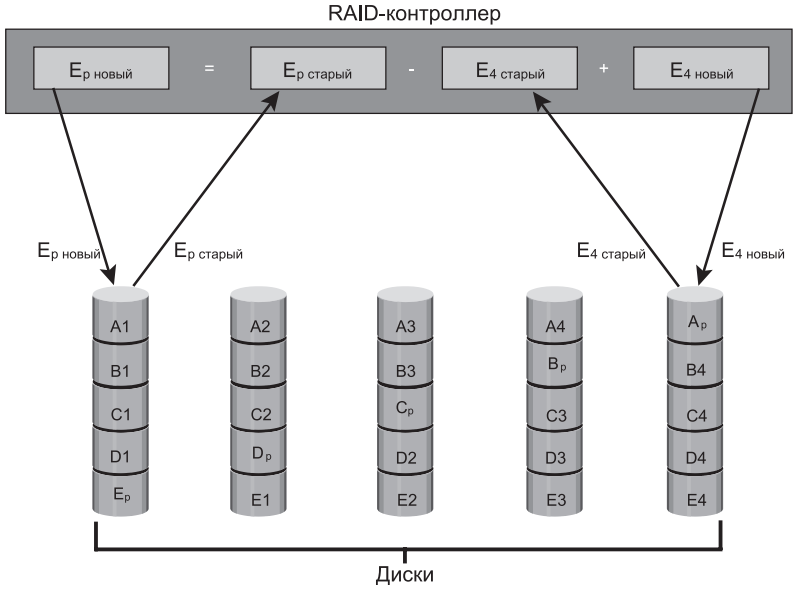


Рис. 3.11. Издержки записи в RAID 5

Информация контроля четности (P) рассчитывается в контроллере следующим образом:

$EP = E_1 + E_2 + E_3 + E_4$ (операции XOR).

При выполнении контроллером операции записи по запросу на ввод-вывод необходимо рассчитать контрольную информацию, для чего нужно считать с диска старую информацию четности (EP старый) и старые данные (E4 старый), то есть выполнить две операции считывания. Затем для расчета новой контрольной информации (EP новый) выполняется следующее действие:

$EP_{новый} = EP_{старый} – E_{4\ старый} + E_{4\ новый}$ (операции XOR).

После расчета новой контрольной информации контроллер завершает операцию записи по запросу на ввод-вывод, записывая новые данные и новую контрольную информацию на диски, что в сумме составляет две операции записи по запросу на ввод-вывод. Следовательно, для каждой записи контроллер выполняет две операции считывания с диска и две операции записи на диск, при этом издержки записи равны четырем.

При реализации RAID 6, в которой поддерживается хранение двойной контрольной информации, запись на диск требует проведения трех операций считывания: двух для контрольной информации и одной для данных. После вычисления обеих частей новых контрольных данных контроллер выполняет три операции записи: две для записи контрольной информации и одну для записи в рамках операции ввода-вывода. Следовательно, при реализации RAID 6 контроллер выполняет шесть операций ввода-вывода

для каждой записи по запросу ввода-вывода, при этом издержки записи равны шести.

3.5.1. Потребность приложений в IOPS и RAID-конфигурации

При принятии решения по количеству дисков, требуемому для приложения, важно учесть влияние RAID-конфигурации, рассчитываемое на основе количества операций ввода-вывода в секунду (IOPS), генерируемых приложением. Общая рабочая нагрузка на диск должна рассчитываться с учетом типа RAID-конфигурации и соотношения количества операций чтения и записи, запросы на которые поступают с хоста.

Следующий пример иллюстрирует методику расчета дисковой нагрузки при различных типах RAID-конфигураций.

Рассмотрим приложение, генерирующее 5200 IOPS, 60 % из которых — операции считывания.

Нагрузка при конфигурации RAID 5 рассчитывается следующим образом:

Дисковая нагрузка при конфигурации RAID 5 (считывания + записи) = $0{,}6 \cdot 5200 + 4 \cdot (0{,}4 \cdot 5200)$ [поскольку издержки записи в конфигурации RAID 5 равны 4] =

$= 3120 + 4 \cdot 2080 = 3120 + 8320 = 11\,440$ IOPS.

Дисковая нагрузка при конфигурации RAID 1 рассчитывается следующим образом:

Дисковая нагрузка при конфигурации RAID 1 = $0{,}6 \cdot 5200 + 2 \cdot (0{,}4 \cdot 5200)$ [поскольку при каждой записи на диски проводится две операции записи] =

$= 3120 + 2 \cdot 2080 = 3120 + 4160 = 7280$ IOPS.

Рассчитанная нагрузка на диск определяет количество необходимых для приложения дисков. Если в данном примере нужно использовать дисковый накопитель с заявленным максимальным показателем 180 IOPS, то количество таких дисков, требуемое для соответствия рабочей нагрузке RAID-конфигурации, будет следующим:

- RAID 5:

  11 440/180 = 64 диска;

- RAID 1:

  7280/180 = 42 диска (округлено до ближайшего четного количества).

3.6. Сравнение RAID-конфигураций

Сравнительные характеристики различных RAID-конфигураций даны в табл. 3.2.

Таблица 3.2. Сравнение наиболее распространенных типов RAID-конфигураций

RAID	МИНИМАЛЬНОЕ КОЛИЧЕСТВО ДИСКОВ	ЭФФЕКТИВНОСТЬ ХРАНЕНИЯ ДАННЫХ, %	СТОИМОСТЬ	ПРОИЗВОДИТЕЛЬНОСТЬ ЧТЕНИЯ	ПРОИЗВОДИТЕЛЬНОСТЬ ЗАПИСИ	ИЗДЕРЖКИ ЗАПИСИ	ЗАЩИТА
0	2	100	Низкая	Хорошая для чтения как произвольных, так и последовательных данных	Хорошая	Отсутствуют	Защита отсутствует
1	2	50	Высокая	Лучше, чем у отдельного диска	Хуже, чем у отдельного диска, поскольку каждая запись должна быть сделана на все диски	Средние	Зеркальная защита
3	3	$[(n-1)/n] \cdot 100$, где n — количество дисков	Средняя	Приемлемая для чтения произвольных данных и хорошая для чтения последовательных данных	От низкой до приемлемой для небольших произвольных записей и приемлемая для больших, последовательных записей	Высокие	Защита от одиночного отказа диска за счет контроля четности
4	3	$[(n-1)/n] \cdot 100$, где n — количество дисков	Средняя	Хорошая для чтения произвольных и последовательных данных	Приемлемая для произвольных и последовательных записей	Высокие	Защита от одиночного отказа диска за счет контроля четности
5	3	$[(n-1)/n] \cdot 100$, где n — количество дисков	Средняя	Хорошая для чтения произвольных и последовательных данных	Приемлемая для произвольных и последовательных записей	Высокие	Защита от одиночного отказа диска за счет контроля четности
6	4	$[(n-1)/n] \cdot 100$, где n — количество дисков	Средняя, но выше, чем у RAID 5	Хорошая для чтения произвольных и последовательных данных	От низкой до приемлемой для небольших произвольных записей и приемлемая для больших, последовательных записей	Очень высокие	Защита от отказа двух дисков за счет контроля четности
1 + 0 и 0 + 1	4	50	Высокая	Хорошая	Хорошая	Средние	Зеркальная защита

3.7. Горячее резервирование

Под горячим резервированием подразумевается использование резервного накопителя, который временно заменяет отказавший дисковый накопитель, перенимая все его особенности. При горячем резервировании в зависимости от типа RAID-реализации выполняется один из следующих методов восстановления данных.

- Если используется RAID-конфигурация с контролем четности, то данные восстанавливаются на резервном дисковом накопителе на основе контрольной информации и тех данных, которые находятся на неповрежденных дисковых накопителях RAID-набора.
- При зеркалировании копирование данных на резервный накопитель осуществляется с уцелевшего зеркального диска.

При добавлении к системе нового дискового накопителя на него копируются данные с резервного накопителя. А резервный накопитель возвращается в состояние простоя в готовности заменить следующий отказавший накопитель. Есть вариант, при котором резервный накопитель заменяет отказавший дисковый накопитель на постоянной основе. Это означает, что он перестает быть резервным накопителем и в конфигурацию массива должен быть включен новый резервный накопитель.

Емкость резервного накопителя должна быть достаточна для приема данных с отказавшего диска. Для повышения доступности данных на некоторых системах используются сразу несколько резервных накопителей.

Резервный накопитель может быть настроен на автоматический или на ручной ввод в активную работу, определяющий порядок его использования в случае отказа дискового накопителя. При настройке на автоматическую замену дисковая подсистема в рамках горячего резервирования предпринимает попытку копирования данных с диска, на котором превышен определенный порог появления восстанавливаемых ошибок. Если эта задача завершается до выхода диска из строя, подсистема переключается на резервный накопитель и помечает отказавший диск как неиспользуемый. В противном случае для восстановления данных используется информация контроля четности или диск, содержащий зеркальную копию. При настройке на ручную замену процессом восстановления управляет администратор. К примеру, чтобы не допустить снижения производительности системы, восстановление может быть проведено в ночное время. Но при этом в случае еще одного отказа накопителя система подвергается риску потери данных.

Резюме

Отдельные диски склонны к отказам, что создает угрозу недоступности данных. RAID-массивы обеспечивают выполнение требований по доступности данных путем применения методов зеркалирования и контроля четности.

Вдобавок к обеспечению преимуществ избыточности, RAID-конфигурации с технологией чередования данных повышают производительность выполнения команд ввода-вывода путем распространения данных по нескольким дисковым накопителям.

В данной главе были рассмотрены базовые концепции чередования данных, их зеркалирования и контроля четности, составляющие основу реализации различных RAID-уровней. Выбор RAID-уровня зависит от требований, предъявляемых приложением к производительности хранилища данных, его стоимости и степени защищенности данных.

RAID является основой ряда усовершенствований технологии хранения данных. В следующей главе будут рассмотрены интеллектуальные системы хранения данных, реализующие RAID со специальной операционной средой, предоставляющей высокий уровень производительности и доступа к данным.

УПРАЖНЕНИЯ

1. Почему RAID 1 не может служить заменой резервному копированию?
2. Проведите исследование RAID 6 и выполняемого в этой конфигурации вторичного вычисления четности.
3. Объясните ход процесса восстановления данных при отказе накопителя в RAID 5.
4. В чем заключаются преимущества использования RAID 3 в приложении резервного копирования?
5. Проанализируйте влияние операций произвольного и последовательного ввода-вывода в различных RAID-конфигурациях.
6. У приложения имеется 1000 весьма активных пользователей, каждый из которых создает пиковую нагрузку в 2 IOPS, и 2000 обычных пользователей, каждый из которых создает пиковую нагрузку в 1 IOPS. Предполагается, что приложение подвергается и другим нагрузкам, составляющим максимум 20 %. Соотношение чтения-записи для приложения составляет 2:1. Вычислите количество IOPS, скорректированное для применения RAID 1/0, RAID 5 и RAID 6.
7. Используя условия упражнения 6, вычислите количество накопителей, требующееся для поддержки приложения в различных RAID-средах при использовании накопителей со скоростью вращения шпинделя 10 000 об./мин и заявленным показателем 130 IOPS.
8. Каков размер дорожки массива RAID 5, состоящего из пяти дисков с размером полос 32 Кбайт? Сравните его с размером дорожки массива RAID 0, состоящего из пяти дисков, имеющих такие же размеры полос.

Глава 4
Интеллектуальные системы хранения данных

Бизнес-приложения требуют высокого уровня производительности, доступности, безопасности и масштабируемости. Ключевым элементом, от которого зависит производительность любой системы хранения данных, является дисковый накопитель. Некоторые устаревшие технологии дисковых массивов не позволяли преодолеть пределов производительности из-за ограничений дисков и их механических компонентов. Технология RAID-массивов внесла весомый вклад в повышение производительности и надежности систем хранения данных, но дисковые накопители, даже с реализацией в RAID, не могут удовлетворить требования к производительности, предъявляемые современными приложениями.

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Интеллектуальные системы хранения данных
- Зеркалирование кэша и аварийное сохранение данных из кэш-памяти
- Номер (адрес) логического устройства (LUN)
- LUN-маскирование
- МетаLUN
- Виртуальное предоставление ресурсов хранения данных
- Высокопроизводительные системы хранения данных
- Системы хранения среднего класса

По мере совершенствования технологии появился новый тип решений по хранению данных, известный как интеллектуальная система хранения данных. Интеллектуальные системы хранения данных представляют собой полнофункциональные RAID-массивы, обеспечивающие возможность обработки запросов на ввод-вывод с высокой степенью оптимизации. Эти системы хранения данных комплектуются памятью большого объема (которая называется кэш-памятью) и несколькими путями ввода-вывода, и кроме того, в них используются очень сложные алгоритмы, позволяющие соответствовать требованиям, предъявляемым приложениями, которым нужна высокая производительность. У массивов таких систем имеется операционная среда, осуществляющая интеллектуальное и оптимизированное управление ресурсами

системы хранения данных, их распределение и использование. С появлением поддержки флеш-накопителей и других современных технологий, таких как предоставление виртуальных хранилищ и автоматизированное многоуровневое хранение данных, к производительности, масштабируемости и готовности систем хранения данных было добавлено новое измерение.

В данной главе рассмотрены компоненты интеллектуальных систем хранения данных, а также вопросы предоставления приложениям ресурсов хранения данных.

4.1. Компоненты интеллектуальной системы хранения данных

Интеллектуальная система хранения данных состоит из четырех основных компонентов: внешнего интерфейса, кэш-памяти, внутреннего интерфейса и физических дисков. Эти компоненты и соединения между ними показаны на рис. 4.1. Запрос на ввод-вывод приходит от хоста на порт внешнего интерфейса, обрабатывается, проходя кэш-память и внутренний интерфейс, чтобы появилась возможность сохранить данные и извлечь их с физического диска. Если запрошенные данные находятся в кэш-памяти, то запрос на чтение может быть обслужен прямо из нее. В современных интеллектуальных системах хранения данных внешний интерфейс, кэш-память и внутренний интерфейс обычно собраны на одной плате, которую называют процессором хранилища, или контроллером хранилища.

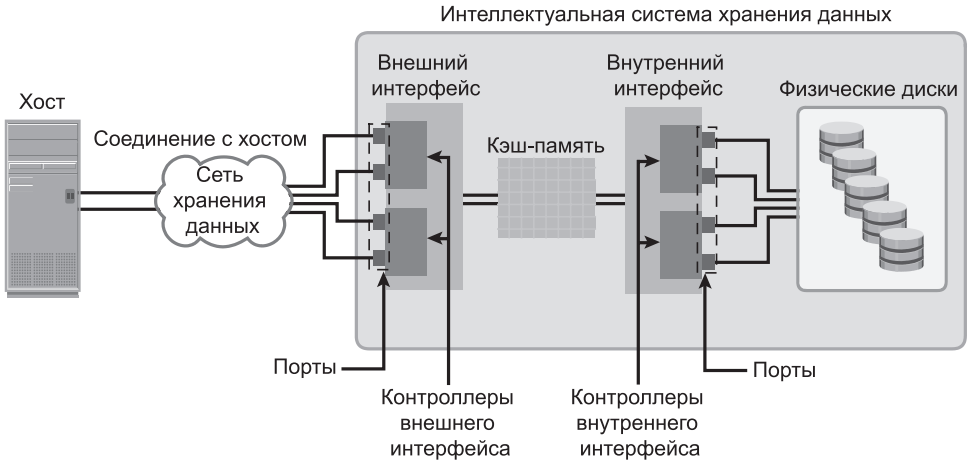


Рис. 4.1. Компоненты интеллектуальной системы хранения данных

4.1.1. Внешний интерфейс

Внешний интерфейс обеспечивает взаимодействие системы хранения данных с хостом. Он состоит из двух компонентов: портов внешнего интерфейса и контроллеров внешнего интерфейса. Обычно с целью получения высокой доступности данных внешний интерфейс имеет резервные контроллеры, и у каждого контроллера имеется несколько портов, позволяющих подключиться к интеллектуальной системе хранения данных большому количеству хостов. У каждого контроллера интерфейса имеется логика обработки запросов, выполняющая соответствующий транспортный протокол, например Fibre Channel, iSCSI, FICON или FCoE.

Контроллеры внешнего интерфейса направляют данные в кэш-память и получают их обратно по внутренней шине данных. Когда кэш-память получает данные для записи, контроллер посылает на хост подтверждающее сообщение.

4.1.2. Кэш-память

Кэш представляет собой полупроводниковую память, в которую временно помещаются данные для сокращения времени, требующегося на обслуживание запросов ввода-вывода, получаемых от хоста.

Кэш-память повышает быстродействие системы хранения данных путем изолирования хоста от механических задержек, связанных с вращающимися дисками или накопителями на жестких дисках (HDD), которые являются самыми медленными компонентами интеллектуальной системы хранения данных. Как правило, на доступ к данным, находящимся на вращающемся диске, требуется несколько миллисекунд, необходимых на поиск и поворот диска. Доступ к данным из кэш-памяти осуществляется быстрее и занимает, как правило, менее миллисекунды. В интеллектуальных массивах данные для записи сначала помещаются в кэш-память, а затем записываются на диск.

Структура кэш-памяти

Кэш-память организована в виде страниц, являющихся наименьшей единицей ее распределения. Размер кэш-страницы настраивается в соответствии с размером ввода-вывода, осуществляемого приложением. Кэш состоит из массива данных и оперативной памяти тегов. В массиве данных хранятся данные, а с помощью оперативной памяти тегов отслеживается размещение данных в массиве (рис. 4.2) и на диске.

Записи в оперативной памяти тегов указывают, где найти данные, помещенные в кэш-память, и где хранятся эти данные на диске. Оперативная память тегов включает флаг статуса данных, указывающий, была ли проведена запись данных, находящихся в кэш-памяти, на диск. В кэше также содержится информация, касающаяся времени, например время последнего обращения, используемое для идентификации кэшированной информации, к которой длительное время не было никаких обращений и от которой можно избавиться.

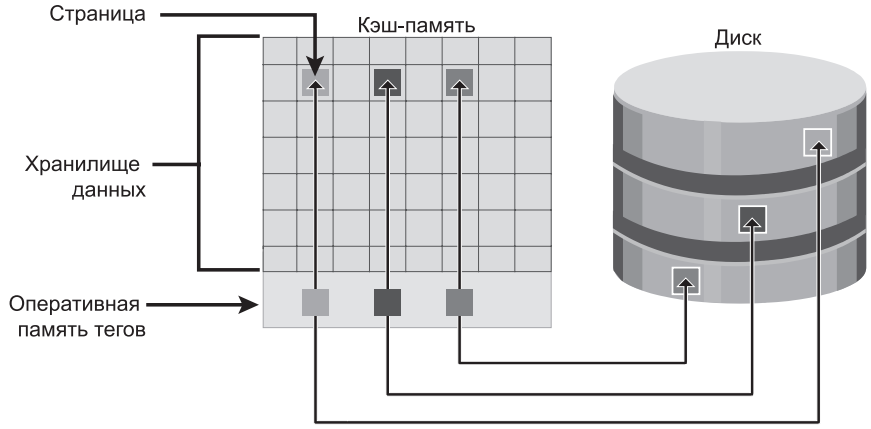


Рис. 4.2. Структура кэш-памяти

Операция чтения с использованием кэш-памяти

Когда хост отправляет запрос на чтение, контроллер хранилища обращается к оперативной памяти тегов, чтобы определить, имеются ли запрошенные данные в кэш-памяти. Если эти данные там найдены (это называется кэш-попаданием при чтении или попаданием при чтении), они отправляются непосредственно на хост без каких-либо операций с диском (рис. 4.3, *а*). Таким образом обеспечивается быстрый ответ на запрос хоста (время ответа около миллисекунды). Ситуация, когда запрошенные данные не найдены в кэш-памяти, называется кэш-промахом. Данные в этом случае должны быть считаны с диска (рис. 4.3, *б*). Внутренний интерфейс обращается к соответствующему диску и извлекает запрошенные данные. После этого данные помещаются в кэш-память и в конечном итоге отправляются на хост через внешний интерфейс. Кэш-промахи увеличивают время отклика на запросы ввода-вывода.

При последовательных запросах на чтение используется алгоритм упреждающей выборки или предварительного считывания. В этом случае извлекается смежный набор связанных блоков. При этом с диска могут быть заранее считаны и помещены в кэш-память несколько других, еще не запрошенных блоков. Когда чуть позже хост запросит эти блоки, операции чтения превратятся в кэш-попадания при чтении. Такой процесс значительно сокращает для хоста время отклика. В интеллектуальной системе хранения данных предлагаются фиксированные и переменные размеры данных предварительного считывания. Фиксированное предварительное считывание позволяет системе считывать фиксированный объем данных. Это наиболее подходящий вариант при одинаковых объемах ввода-вывода, запрашиваемого хостом. Переменное предварительное считывание позволяет интеллектуальной системе хранения считывать объем данных, кратный размеру данных, запрашиваемых хостом. Максимальный объем упреждающей выборки ограничивает количество заранее считываемых блоков данных, чтобы предотвратить занятость дисков предварительным считыванием в ущерб обработке других запросов на ввод-вывод.

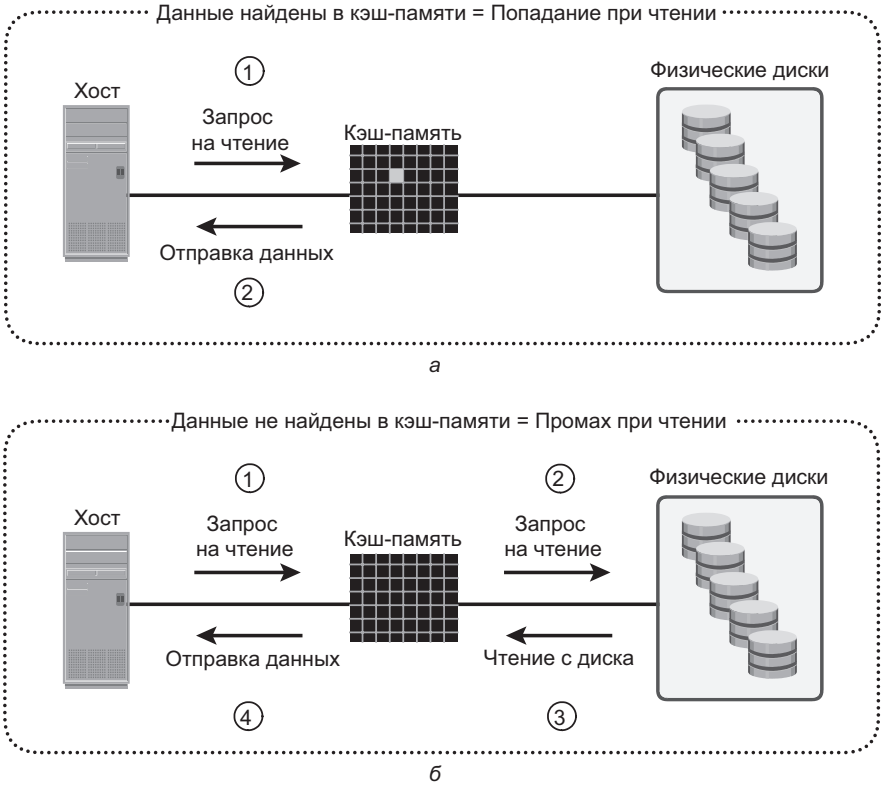


Рис. 4.3. Попадание и промах при запросах на чтение

Производительность чтения, измеряемая коэффициентом попаданий при чтении или просто коэффициентом попаданий, обычно выражается в процентах. Этот коэффициент представляет собой отношение числа попаданий при чтении к общему числу запросов на чтение. При более высоком коэффициенте попаданий при чтении производительность чтения повышается.

Операция записи с использованием кэш-памяти

Операции записи с использованием кэш-памяти имеют более высокую производительность по сравнению с записью непосредственно на диски. Когда при выполнении операции ввода-вывода запись ведется в кэш-память с возвратом подтверждения, эта операция (с позиции хоста) завершается намного раньше, чем при записи непосредственно на диск. Возможности оптимизации имеются и при последовательно проводимых записях, так как при использовании кэш-памяти множество более мелких операций записи можно объединить для получения более объемных переносов данных на дисковые накопители.

Операция записи с использованием кэш-памяти реализуется следующими способами.

- **Кэширование с отложенной записью:** данные помещаются в кэш-память, и на хост сразу же возвращается подтверждение. Чуть позже данные из нескольких запросов на запись окончательно переносятся на диск. Время отклика на запись существенно сокращается, поскольку на операции записи не влияют механические задержки диска. Тем не менее в случае отказа кэш-памяти существует риск потери не перенесенных на диск данных.
- **Кэширование со сквозной записью:** данные помещаются в кэш-память и тут же записываются на диск с отправкой подтверждения на хост. Так как данные переносятся на диск сразу же после их поступления, риск их потери снижается, а время отклика на запись из-за выполнения дисковых операций увеличивается.

При определенных обстоятельствах, например при операции записи большого объема данных, запись может вестись в обход кэш-памяти. При этом, если объем данных запроса на ввод-вывод превышает предопределенный размер, называемый размером, при котором запись ведется в обход кэша (write aside size), запись производится прямо на диск, чтобы снизить негативное влияние на кэш-память записи большого объема данных, потребляющего слишком много этой памяти. Такой режим работы особенно пригодится в среде с ограниченными кэш-ресурсами, где кэш-память требуется для проведения операций произвольного ввода-вывода с небольшим объемом данных.

Реализация кэш-памяти

Кэш-память может быть выделенной или глобальной. В выделенной кэш-памяти для операций чтения и записи резервируются отдельные участки памяти. В глобальной кэш-памяти под операции чтения и записи могут использоваться любые доступные адреса памяти. Более эффективно управление выполняется в глобальной кэш-памяти, поскольку управлять приходится только одним глобальным набором адресов.

При управлении глобальной кэш-памятью пользователи могут задавать ее объем, который будет доступен для операций чтения и записи, в процентах. Как правило, под чтение резервируется небольшой объем кэш-памяти, но если используемое приложение считывает информацию довольно часто, он должен быть увеличен. В других реализациях глобальной кэш-памяти соотношение объемов, доступных для чтения и записи, регулируется динамически в соответствии с рабочей нагрузкой.

Управление кэш-памятью

Кэш-память представляет собой ограниченный и весьма дорогостоящий ресурс, требующий надлежащего управления. Несмотря на то что современные интеллектуальные системы хранения данных поставляются с большими

объемами кэш-памяти, при заполнении всех кэш-страниц часть из них во избежание снижения производительности должна быть освобождена под новые данные. В интеллектуальных системах хранения данных для упреждающего создания набора свободных страниц и ведения списка страниц, которые могут быть в случае необходимости освобождены, реализуются различные алгоритмы управления кэш-памятью. В следующем перечне рассматриваются наиболее часто востребуемые алгоритмы.

- **Замещение наименее востребованных страниц** — Least Recently Used (LRU): алгоритм, постоянно отслеживающий обращение к данным в кэш-памяти и определяющий наименее востребованные кэш-страницы. Эти страницы либо освобождаются, либо помечаются как пригодные для повторного использования. Этот алгоритм основывается на предположении, что данные, не использовавшиеся в течение некоторого времени, не будут запрошены хостом. Но если страница содержит данные для записи, которые еще не были сброшены на диск, перед повторным использованием страницы эти данные сначала будут на него записаны.
- **Замещение самых последних использовавшихся страниц** — Most Recently Used (MRU): алгоритм, противоположный алгоритму LRU, согласно которому освобождаются или помечаются как пригодные для повторного использования те страницы, к которым были самые последние обращения. Этот алгоритм основывается на предположении, что недавно использовавшиеся данные некоторое время не будут востребованы.

Чтобы управлять доступностью пространства, по мере заполнения кэш-памяти система хранения данных должна предпринимать действия по сбросу на диск измененных страниц (данных, записанных в кэш-память, но еще не записанных на диск). Сброс представляет собой процесс передачи на диск данных, находящихся в кэш-памяти. Для управления процессом сброса на основании частоты и схемы обращения по вводу-выводу в кэш-памяти задаются верхний и нижний уровни, называемые уровнями заполнения. Верхний уровень заполнения — это уровень использования кэш-памяти, при достижении которого система хранения начинает быстрый сброс данных, находящихся в кэше. Нижний уровень заполнения — это уровень, при снижении до которого система хранения данных прекращает сброс данных на диск. Как показано на рис. 4.4, применяемый режим сброса управляется степенью использования кэш-памяти.

- ***Ленивый сброс во время простоя*** (Idle flushing): происходит постоянно умеренными темпами при степени использования кэш-памяти, соответствующей ее заполнению между верхним и нижним уровнями.
- **Сброс на верхнем уровне заполнения** (High watermark flushing): активизируется, если использование кэш-памяти достигло верхнего уровня заполнения. Для выполнения сброса система хранения данных выделяет некоторые дополнительные ресурсы. Используемый

в данном случае режим сброса влияет на процесс обработки ввода-вывода.

- **Принудительный сброс** (Forced flushing): происходит в случае резкого увеличения объема ввода-вывода, когда степень заполнения кэш-памяти достигает 100 %, что оказывает существенное влияние на время отклика при выполнении операций ввода-вывода. В данном режиме система отдает приоритет сбросу данных, выделяя для этого больше ресурсов.

Рис. 4.4. Типы сброса данных

Защита кэш-данных

Кэш является энергозависимой памятью, и сбой электропитания или любой другой сбой в работе кэш-памяти может привести к потере данных, еще не сброшенных на диск. Риск потери не переданных данных, находящихся в кэш-памяти, может быть уменьшен за счет использования зеркалирования кэша и аварийного сохранения данных из кэш-памяти.

- **Зеркальное кэширование** (Cache mirroring): каждая запись в кэш-память хранится в двух разных местах на двух независимых картах памяти. В случае сбоя в работе кэш-памяти данные для записи сохранятся в зеркально отображенном месте памяти и могут быть сброшены на диск. При чтении данные попадают с диска в кэш, поэтому при сбое кэш-памяти данные по-прежнему могут быть доступны на диске. Поскольку зеркалируются только записи, такой метод характеризуется более эффективным использованием доступной кэш-памяти.

  При использовании зеркального кэширования возникает проблема *согласованности кэш-памяти*. Согласованность кэш-памяти означает, что зеркалированные данные, помещенные в два разных места памяти, должны быть все время идентичны друг другу. За обеспечение согласованности отвечает операционная среда массива.

- **Аварийное сохранение данных кэш-памяти** (Cache vaulting): в случае сбоя электропитания кэш-память подвергается риску потери не

сброшенных на диск данных. Эта проблема может быть решена различными способами: питанием памяти от батареи до восстановления подачи сетевого питания или использованием энергии батареи для записи данных, находящихся в кэш-памяти, на диск. В случае длительного отсутствия электропитания использование батареи не является выходом из положения, поскольку в интеллектуальных системах хранения данных может появиться необходимость в сбросе больших объемов информации на многочисленные диски и батареи могут не обеспечить электропитание в течение времени, достаточного для записи каждого фрагмента данных на предназначенный для него диск. Поэтому поставщики систем хранения данных используют группу физических дисков для сброса на них содержимого кэш-памяти при сбое электропитания. Это называется *аварийным сохранением* данных из кэш-памяти, а диски называются *резервными накопителями*. Когда подача электроэнергии возобновляется, данные с этих дисков записываются обратно в кэш, а после этого сбрасываются на диски назначения.

ТЕХНОЛОГИЯ ФЛЕШ-КЭШИРОВАНИЯ СЕРВЕРА

Технология флеш-кэширования сервера использует на хост-машине программы интеллектуального кэширования и флеш-карту с интерфейсом PCI Express (PCIe). Благодаря уменьшению задержек и повышению пропускной способности производительность приложения существенно увеличивается. Технология флеш-кэширования сервера работает как в физической, так и в виртуальной среде и обеспечивает повышение производительности для рабочих нагрузок, связанных с интенсивным чтением данных. Этой технологией используются минимальные ресурсы центрального процессора и памяти сервера, поскольку управление флеш-памятью перекладывается на PCIe-карту.

Эта карта интеллектуально определяет, какие именно данные выгоднее всего разместить на сервере на PCIe флеш-памяти поближе к приложению. Тем самым можно избежать задержек, связанных с сетевым доступом ввода-вывода к массиву хранения данных. При этом вычислительные мощности, требующиеся для данных, к которым приложение обращается наиболее часто, тратятся не внутренним хранилищем, а PCIe-картой. Следовательно, массив хранения данных может выделить больше вычислительной мощности другим приложениям.

4.1.3. Внутренний интерфейс

Внутренний интерфейс обеспечивает взаимодействие кэш-памяти и физических дисков. Он состоит из двух компонентов: внутренних портов и внутренних контроллеров. Внутренний интерфейс управляет передачей данных между кэш-памятью и физическими дисками. Из кэш-памяти данные отправляются во внутренний интерфейс, а затем — на диск назначения. Физические диски подключены к портам внутреннего интерфейса. При выполнении

операций чтения и записи внутренний контроллер обменивается данными с дисками, а также предоставляет дополнительное, но ограниченное по объему временное хранилище данных. Алгоритмы, реализованные во внутренних контроллерах, обеспечивают наряду с RAID-функциональностью обнаружение и исправление ошибок.

Для достижения высоких уровней защиты и доступности данных системы хранения оснащаются двойными контроллерами с несколькими портами. Такая конфигурация обеспечивает альтернативный маршрут доступа к физическим дискам в случае сбоя контроллера или порта. Надежность повышается еще больше, если у дисков также имеются два порта. В этом случае каждый порт диска может быть подключен к отдельному контроллеру. Использование нескольких контроллеров также облегчает балансировку нагрузки.

4.1.4. Физический диск

Физические диски подключены к внутреннему контроллеру хранилища и обеспечивают постоянное хранение данных. Современные интеллектуальные системы хранения данных предоставляют поддержку разнообразных дисковых накопителей разных типов и скоростей, например FC, SATA, SAS и флеш-накопителей. Они также поддерживают использование в одном и том же массиве сочетания флеш-накопителей, FC или SATA.

4.2. Предоставление ресурсов хранения данных

Предоставление ресурсов хранения данных — это процесс назначения этих ресурсов хостам на основе требований работающих на хостах приложений по объему, доступности и производительности. Предоставление ресурсов хранения может выполняться двумя путями: традиционным и виртуальным. В *виртуальном предоставлении* для выделения приложениям ресурсов хранения используется технология виртуализации. В данном разделе подробно рассматривается как традиционное, так и виртуальное предоставление ресурсов хранения данных.

4.2.1. Традиционное предоставление ресурсов хранения данных

При традиционном предоставлении ресурсов хранения данных физические диски сводятся в логические группы, а для формирования набора применяется требуемый RAID-уровень, называемый RAID-набором. Доступность, объем и производительность RAID-набора определяются количеством имеющихся в нем накопителей. Чтобы обеспечить наиболее приемлемые объем, надежность и постоянство в производительности, настоятельно рекомендуется создавать RAID-набор из накопителей одного типа, равных по скорости и объему. К примеру, если в RAID-наборе смешать накопители разной

емкости, то для формирования общего объема RAID-набора от каждого диска будет использован объем, имеющийся у наименьшего из накопителей. Весь остальной объем более емких накопителей будет не задействован. Точно так же смешивание накопителей с более высокой скоростью вращения шпинделя с накопителями с меньшей скоростью, измеряемой в оборотах в минуту (RPM), приводит к снижению общей производительности RAID-набора.

Обычно RAID-наборы имеют весьма большую емкость, поскольку она является совокупностью емкостей входящих в них отдельных накопителей. Путем разбиения доступной емкости на устройства меньшего объема, то есть разбиения на разделы, из RAID-наборов создаются *логические устройства* (видимые как части RAID-набора). Затем эти устройства назначаются хосту на основе требований, предъявляемых к хранилищу данных.

Логические устройства охватывают все физические диски, принадлежащие набору. Каждому логическому устройству, созданному из RAID-набора, назначается уникальный идентификатор, который называется *номером логического устройства* — logical unit number (LUN). LUN скрывают от хостов организацию и структуру RAID-набора. Чтобы LUN, созданные методами традиционного предоставления ресурсов хранения данных, можно было отличить от LUN, созданных методами виртуального предоставления, их называют *толстыми LUN-устройствами* (thick LUNs).

На рис. 4.5 показан разбитый на разделы RAID-набор из пяти дисков, в результате чего получились два логических устройства с номерами LUN 0 и LUN 1. Эти устройства назначены в соответствии с требованиями по хранению данных хостам Host1 и Host 2.

Когда LUN-устройство создано и назначено невиртуализированному хосту, для идентификации LUN требуется сканирование шины. Это LUN-устройство операционная система видит как обычный диск. При подготовке к работе устройство форматируется под файловую систему, после чего эта система на него устанавливается.

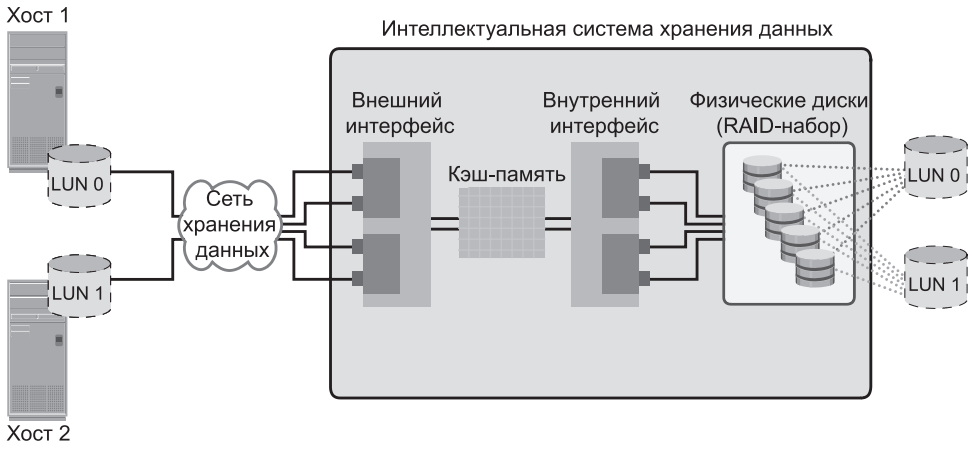


Рис. 4.5. RAID-набор и LUN-устройства

В среде виртуализированного хоста LUN-устройство назначается гипервизору, который распознает его как обычный диск. Этот диск настраивается файловой системой гипервизора, а затем на нем создаются виртуальные диски. *Виртуальные диски* представляют собой обычные файлы в файловой системе гипервизора. Затем виртуальные диски назначаются виртуальным машинам и отображаются для них в виде обычных дисков. Чтобы виртуальная машина смогла использовать виртуальный диск, предпринимаются такие же действия, как и в невиртуализированной среде. Пространство LUN-устройства может быть совместно использовано и одновременно доступно нескольким виртуальным машинам.

Виртуальные машины в системе хранения данных могут также получать доступ к LUN-устройствам напрямую. При использовании данного метода одной виртуальной машине выделяется все LUN-устройство. Хранение данных таким способом рекомендуется в том случае, когда приложения, выполняющиеся на виртуальной машине, требовательны ко времени отклика, а совместное использование устройства хранения данных с другими виртуальными машинами может негативно повлиять на этот параметр. Метод прямого доступа используется также при нахождении виртуальной машины в одном кластере с физической машиной. В этом случае виртуальной машине требуется доступ к LUN-устройству, которое было выделено физической машине.

LUN-расширение: MetaLUN

MetaLUN — это метод расширения LUN-устройств, требующих дополнительной емкости или производительности. MetaLUN может быть создан путем сочетания двух и более LUN-устройств. MetaLUN состоит из основного LUN-устройства и одного или более дополнительных LUN-устройств. MetaLUN-устройства могут быть либо *последовательно объединенными*, либо *чередующимися*.

Расширение с последовательным объединением является простым добавлением к основному LUN-устройству дополнительной емкости. При таком расширении от комплектующих LUN-устройств не требуется иметь такую же емкость, как у основного LUN-устройства. Все LUN-устройства, входящие в metaLUN-устройство с последовательным объединением, должны быть либо защищенными (контролем четности или зеркалированием), либо незащищенными (RAID 0). RAID-типы внутри metaLUN-устройства могут смешиваться. Например, LUN-устройство RAID 1/0 может быть последовательно объединено с LUN-устройством RAID 5. Но LUN-устройство RAID 0 может быть объединено только с другим LUN-устройством RAID 0.

Расширение с последовательным объединением обладает высоким быстродействием, но не дает выигрыша в производительности (рис. 4.6).

Расширение с чередованием организуется путем чередования данных основного LUN-устройства на нем самом и на дополнительных LUN-устройствах. При расширении с чередованием все LUN-устройства должны быть одинаковой емкости и одного и того же RAID-уровня. Расширение с чередованием дает повышенную производительность за счет увеличения количества дисков с чередующимися данными (рис. 4.7).

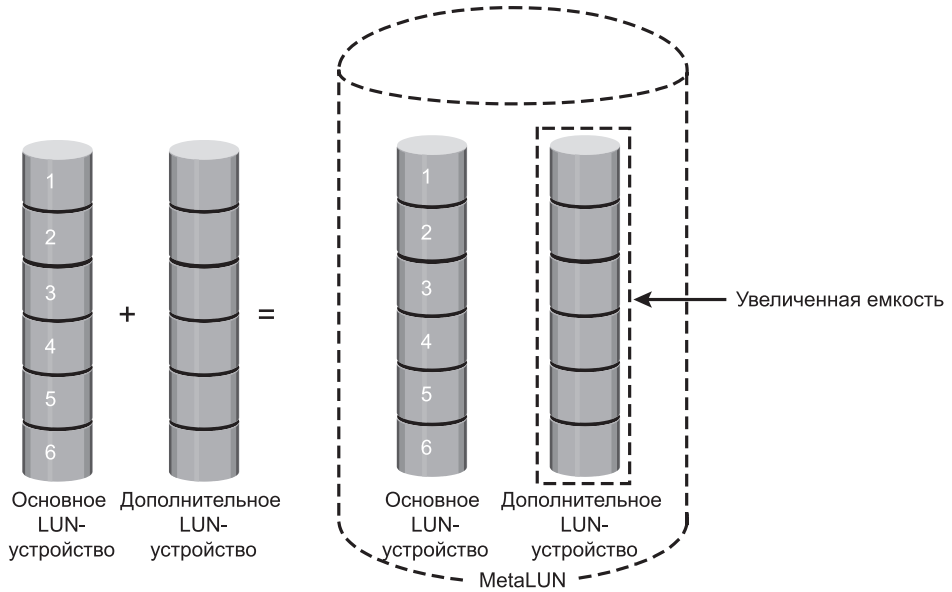


Рис. 4.6. MetaLUN-устройство с последовательным объединением

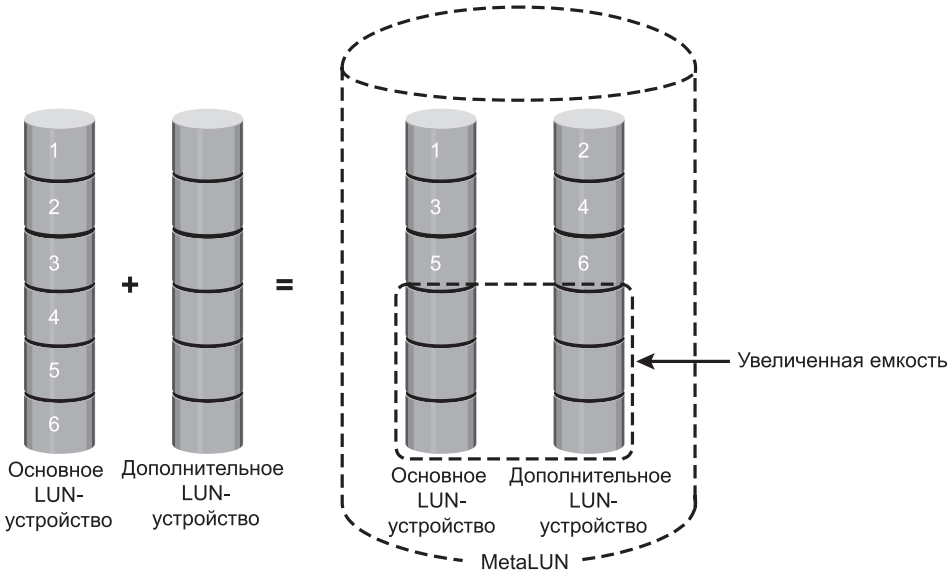


Рис. 4.7. MetaLUN-устройство с чередованием данных

Все LUN-устройства, как в расширении с последовательным объединением, так и в расширении с чередованием данных, должны находиться на одном и том же типе дисковых накопителей: все они должны быть либо Fibre Channel, либо ATA.

4.2.2. Виртуальное предоставление ресурсов хранения данных

Виртуальное предоставление позволяет создавать и передавать LUN-устройство с объемом, превосходящим тот объем, который был ему физически распределен в массиве хранения данных. Чтобы LUN-устройство, созданное с помощью виртуального предоставления ресурсов, можно было отличить от традиционного LUN-устройства, его называют *тонким LUN-устройством* (thin LUN).

Тонкие LUN-устройства на момент их создания и предоставления хосту не требуют полного распределения им физического хранилища данных. Это хранилище распределяется хосту по мере надобности из общего пула физической емкости. *Общий пул* состоит из физических дисков. В виртуальном предоставлении ресурсов хранения данных общий пул является аналогом RAID-группы, представляющей собой набор накопителей, на основе которого создаются LUN-устройства. Так же, как и RAID-группа, общий пул поддерживает единый уровень защиты RAID. Но в отличие от RAID-группы, общий пул может содержать большое количество накопителей. Общие пулы могут быть однородными (содержащими накопители одного типа) или неоднородными (содержащими смешанные типы накопителей, такие как флеш-, FC-, SAS- и SATA-накопители).

Виртуальное предоставление позволяет распределять хранилища данных хостам более эффективно. Оно также допускает превышение лимита емкости, при котором хостам предоставляется больше возможностей, чем фактически имеется в массивах хранения данных. И общий пул, и тонкое LUN-устройство могут быть расширены без ущерба для данных по мере роста потребностей хоста в объемах хранения данных. В массиве хранения данных может быть создано несколько общих пулов, и общий пул может совместно использоваться несколькими тонкими LUN-устройствами. Предоставление тонких LUN-устройств показано на рис. 4.8.

Сравнение виртуального и традиционного предоставления ресурсов хранения данных

Обычно администраторы распределяют емкость хранилища данных на основе предполагаемых потребностей. Как правило, это приводит к резервированию излишней емкости хранения данных, что увеличивает расходы и снижает коэффициент использования емкостей хранения. Часто причины предоставления таких излишних емкостей приложениям со стороны администраторов имеют разный характер, например чтобы избежать частого предоставления ресурсов в том случае, когда емкость LUN-устройства будет исчерпана, а также чтобы сократить для приложения перебои в доступности данных. Предоставление излишних емкостей хранения данных зачастую приводит к дополнительным расходам на приобретение и эксплуатацию средств хранения.

Эти проблемы решаются за счет виртуального предоставления ресурсов хранения данных. Виртуальное предоставление повышает коэффициент использования емкостей хранения данных и упрощает управление

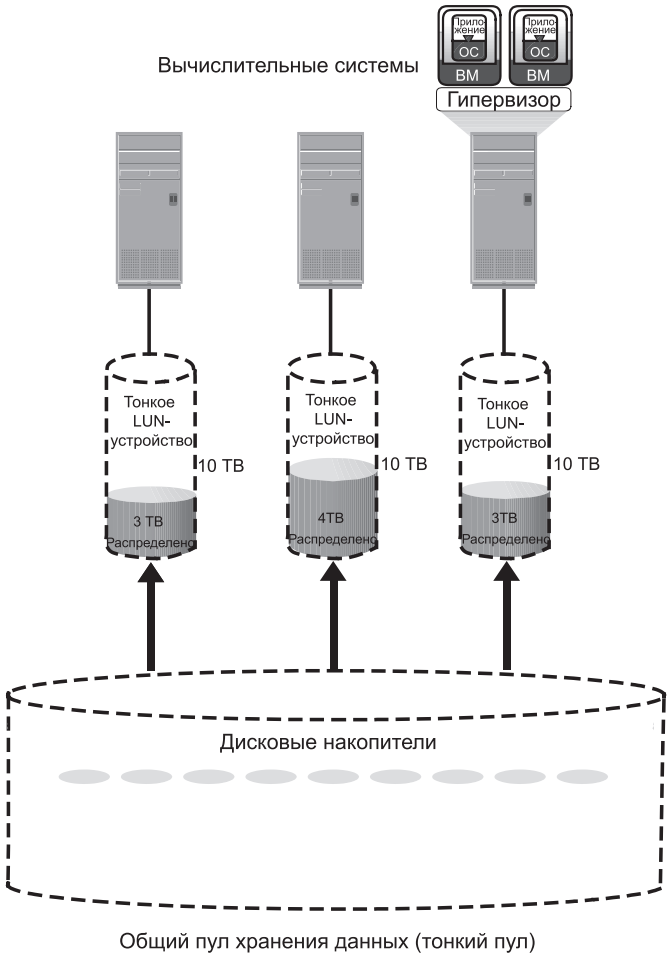


Рис. 4.8. Виртуальное предоставление ресурсов хранения данных

средствами хранения. На рис. 4.9 показан пример сравнения виртуального и традиционного предоставления ресурсов хранения данных.

При традиционном предоставлении одному или нескольким хостам создаются и предоставляются три LUN-устройства (см. рис. 4.9, *а*). Общая емкость системы хранения данных составляет 2 Тбайт. Распределенная емкость устройства LUN 1 составляет 500 Гбайт, из которых используются только 100 Гбайт, а остальные 400 Гбайт не используются. Емкость устройства LUN 2 составляет 550 Гбайт, из которых используются 50 Гбайт, а не используются 500 Гбайт. Емкость устройства LUN 3 составляет 800 Гбайт, из которых используются 200 Гбайт, а не используются 600 Гбайт. В целом в системе хранения имеется 350 Гбайт данных, 1,5 Тбайт распределенной, но неиспользуемой емкости и только 150 Гбайт оставшейся емкости, доступной другим приложениям.

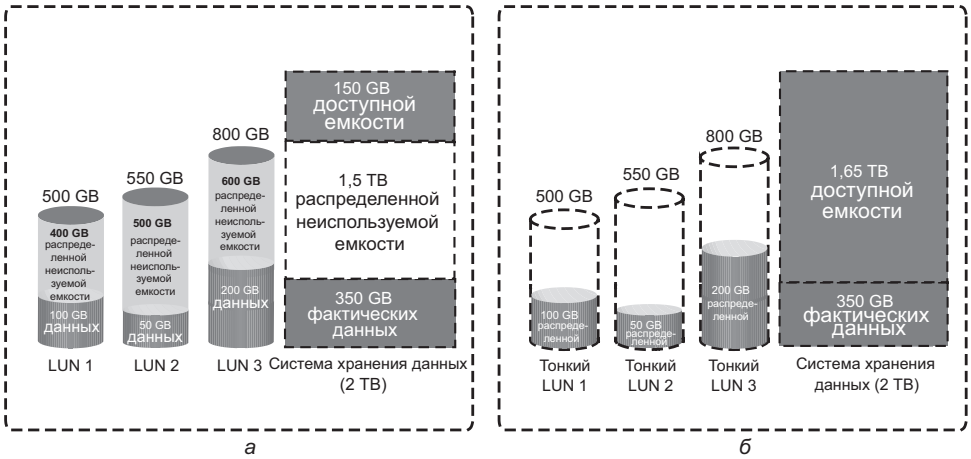


Рис. 4.9. Сравнение традиционного и виртуального предоставления ресурсов хранения данных: *а* — традиционное предоставление; *б* — виртуальное предоставление

Теперь рассмотрим аналогичную систему хранения данных емкостью 2 Тбайт с виртуальным предоставлением (см. рис. 4.9, *б*). В ней созданы три тонких LUN-устройства таких же емкостей. Но здесь нет распределенной неиспользуемой емкости. В целом система хранения данных с виртуальным предоставлением имеет те же 350 Гбайт данных, но при этом другим приложениям доступна емкость 1,65 Тбайт, в то время как в системе с традиционным предоставлением доступно только 150 Гбайт.

Варианты использования тонких и традиционных LUN-устройств

Виртуальное предоставление и тонкое LUN-устройство предлагают множество преимуществ, но бывает так, что для приложения больше подходит традиционное LUN-устройство. Тонкие LUN-устройства больше подходят тем приложениям, которые допускают изменения производительности. В ряде случаев при использовании тонких LUN-устройств ощущается прирост производительности, возникающий благодаря чередованию данных в большом количестве накопителей в пуле. Но при возникновении соперничества среди нескольких неполных LUN-устройств в борьбе за общие ресурсы хранения в заданном пуле и достижении самых высоких уровней загруженности производительность может падать. Тонкие LUN-устройства предоставляют наивысший коэффициент использования пространства хранения данных и хорошо подходят тем приложениям, для которых трудно предсказать объем потребляемого пространства хранения данных. Использование тонких LUN-устройств позволяет организациям экономить электроэнергию и уменьшать расходы на приобретение комплектующих при упрощенном управлении средствами хранения данных.

Традиционные LUN-устройства больше подходят тем приложениям, для которых требуется предсказуемая производительность. Традиционные LUN-устройства предоставляют полный контроль над точным размещением

данных и позволяют администратору создавать LUN-устройства в различных RAID-группах при наличии каких-либо разногласий, касающихся рабочей нагрузки. Организации, не слишком обремененные заботами об эффективном использовании пространства хранения данных, могут остановить свой выбор на использовании традиционных LUN-устройств.

В одном и том же массиве хранения данных могут сосуществовать как традиционные, так и тонкие LUN-устройства. Основываясь на предъявляемых требованиях, администратор может проводить миграцию данных между тонкими и традиционными LUN-устройствами.

4.2.3. Маскирование LUN

Маскированием LUN называется процесс предоставления управления доступом к данным путем определения того, к каким LUN-устройствам может обращаться хост. Функция маскирования LUN реализуется в массиве хранения данных. Тем самым обеспечивается соответствующее управление доступом хостов к томам, предотвращающее неавторизованное или случайное использование данных в совместно используемой среде.

Рассмотрим, к примеру, массив хранения данных с двумя LUN-устройствами, на которых хранятся данные отделов продаж и финансов. Без маскирования LUN оба отдела могли бы с легкостью просматривать данные друг друга и вносить в них изменения, создавая тем самым большую угрозу целостности и безопасности данных. При использовании LUN-маскирования LUN-устройства доступны только вполне определенным хостам.

4.3. Типы интеллектуальных систем хранения данных

Интеллектуальные системы хранения данных подразделяются на две категории:

- высокопроизводительные системы хранения данных;
- системы хранения среднего класса.

Традиционно высокопроизводительные системы хранения данных работают по схеме «активный — активный», в то время как системы хранения среднего класса работают по схеме «активный — пассивный». Различия между этими двумя реализациями постепенно стираются.

4.3.1. Высокопроизводительные системы хранения данных

Высокопроизводительные системы хранения данных, или так называемые массивы «активный — активный», предназначены, как правило, для использования с крупными корпоративными приложениями. Эти системы разрабатываются с большим количеством контроллеров и большим объемом кэш-памяти. При использовании массива типа «активный — активный» подразумевается, что хост может выполнять операции ввода-вывода на свои LUN-устройства через любые доступные контроллеры (рис. 4.10).

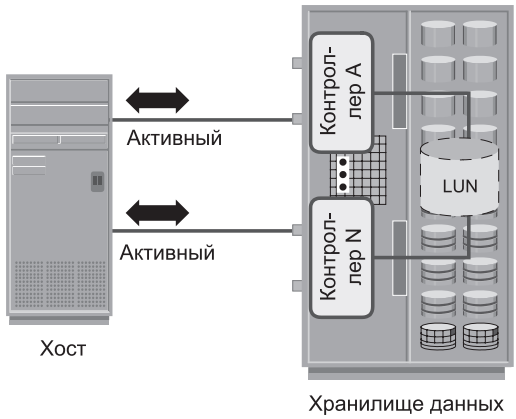


Рис. 4.10. Конфигурация «активный — активный»

Чтобы отвечать запросам корпоративного хранилища данных, эти массивы имеют следующие характеристики:

- большая емкость хранилища;
- большие объемы кэш-памяти для оптимального обслуживания проводимых хостом операций ввода-вывода;
- архитектура, позволяющая сохранять работоспособность при сбоях и повышающая доступность данных;
- возможность подключения к мэйнфреймам и хостам открытых систем;
- доступность нескольких внешних портов и протоколов интерфейса для обслуживания большого числа хостов;
- доступность нескольких внутренних контроллеров для управления дисками;
- масштабируемость для обеспечения растущих потребностей по возможностям подключения, производительности и емкости хранилища;
- возможность обрабатывать большой объем проводимых одновременно операций ввода-вывода от нескольких хостов и приложений;
- поддержка на базе массива локальной и удаленной репликации данных.

В дополнение к указанным характеристикам высокопроизводительные системы обладают рядом уникальных свойств и функций, которые требуются приложениям, выполняющим особо ответственные задачи.

4.3.2. Системы хранения данных среднего класса

Системы хранения данных среднего класса, называемые также массивами «активный — пассивный», оптимально подходят для приложений небольших и средних предприятий. Они также предоставляют оптимальные решения по хранению данных по низкой цене. В массиве «активный — пассивный» хост может выполнять операции ввода-вывода в отношении LUN-устройства

только через контроллер, владеющий этим устройством. Как показано на рис. 4.11, хост может осуществлять операции чтения или записи в отношении LUN-устройства только по пути к контроллеру A, поскольку устройством владеет именно этот контроллер. Путь к контроллеру B остается пассивным, и через него не производится никаких операций ввода-вывода.

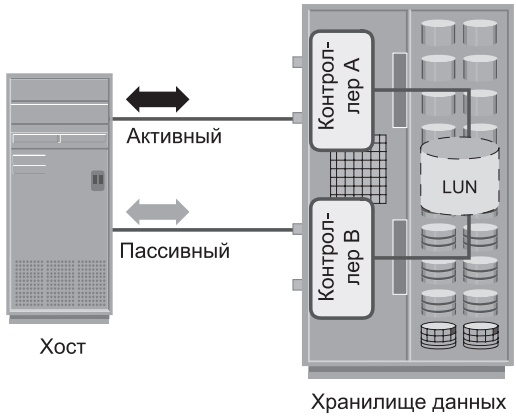


Рис. 4.11. Конфигурация «активный — пассивный»

Системы хранения среднего класса разрабатываются, как правило, с двумя контроллерами, каждый из которых содержит интерфейсы для связи с хостом, кэш-память, RAID-контроллеры и интерфейсы дисковых накопителей.

Массивы средней производительности призваны удовлетворять потребности приложений для небольших и средних предприятий, поэтому они имеют меньшую емкость хранилища и кэш-памяти по сравнению с массивами высокопроизводительных систем хранения данных. В них также меньше внешних портов для подключения к хостам. Тем не менее они обеспечивают высокий уровень избыточности и высокую производительность для приложений с предсказуемыми рабочими нагрузками. Они также поддерживают на базе массива локальную и удаленную репликацию данных.

4.4. Практическая реализация концепций: EMC Symmetrix и VNX

Чтобы проиллюстрировать концепции, о которых шла речь в данной главе, в настоящем разделе рассматривается реализация интеллектуальных массивов хранения данных, осуществленная компанией EMC.

Массив хранения данных EMC Symmetrix работает по схеме «активный — активный». Symmetrix представляет собой решение для потребителей, которым необходимы исключительно высокий уровень обслуживания и производительности, а также самое современное решение по обеспечению непрерывности бизнес-процессов для поддержки приложений с высокими

и непредсказуемыми по объему рабочими нагрузками. Symmetrix имеет также высочайший уровень встроенных функций безопасности и обеспечивает выполнение требований к хранению данных корпоративного уровня с наименьшими затратами на электропитание и охлаждение.

Массив хранения данных EMC VNX реализован по схеме «активный — пассивный». Он представляет собой предложение компании EMC из разряда систем хранения данных среднего класса, предоставляющее свойства и функциональные возможности корпоративного качества. EMC VNX представляет собой унифицированную платформу, предлагающую средства хранения данных на основе блоков, файлов и объектов в одном и том же массиве. Этот массив идеально подходит приложениям с предсказуемой рабочей нагрузкой, требующей обеспечения пропускной способности от среднего до высокого уровня. Характерные особенности унифицированного хранилища данных и EMC VNX рассматриваются в главе 8.

Самую свежую информацию по массивам хранения данных Symmetrix и VNX можно найти на сайте www.emc.com.

4.4.1. Массив хранения данных EMC Symmetrix

EMC Symmetrix устанавливает высочайшие стандарты производительности и функциональных возможностей систем хранения корпоративной информации и признан наиболее надежной платформой хранения данных. Чтобы отвечать требованиям непредсказуемости рабочей нагрузки по вводу-выводу, Symmetrix предлагает масштабируемость и производительность самого высокого уровня. Предложения EMC Symmetrix включают в себя серии Symmetrix Virtual Matrix (VMAX). Эти серии представляют собой инновационную платформу, выстроенную вокруг масштабируемой архитектуры виртуальной матрицы (Virtual Matrix architecture) и предназначенную для поддержки будущего роста потребностей в хранении данных, предъявляемых виртуальными IT-средами. Массив хранения данных Symmetrix VMAX показан на рис. 4.12. Он имеет следующие основные характеристики:

- наращиваемую масштабируемость до 2400 дисков;
- поддержку до 8 VMAX-узлов (каждый VMAX-узел содержит по два блока управления);
- поддержку флеш-накопителей, а также полностью автоматизированного многоуровневого хранения данных — fully automated storage tiering (FAST), виртуального предоставления ресурсов хранения данных и облачных вычислений;
- поддержку до 1 Тбайт глобальной кэш-памяти;
- поддержку подключений к хосту по протоколам FC, iSCSI, GigE и FICON;
- поддержку RAID уровней 1, 1 + 0, 5 и 6;
- поддержку на базе хранилища репликаций через EMC TimeFinder и EMC SRDF;

- высокую отказоустойчивость, допускающую проведение обновлений без сбоев функционирования, и полноценную избыточность на уровне компонентов, допускающую проведение замен оборудования без прерывания работы.

Рис. 4.12. EMC Symmetrix VMAX

4.4.2. Компоненты EMC Symmetrix VMAX

EMC Symmetrix VMAX содержит одну системную стойку и до десяти стоек с хранилищами данных. Стойка хранилища данных поддерживает до 16 полок с массивами накопителей — drive array enclosures (DAE), а каждая такая полка может содержать до 15 накопителей. Системная стойка содержит системные компоненты, включающие VMAX-узлы, полку матричной интерфейсной платы — Matrix Interface Board Enclosure (MIBE), модули резервного электропитания — standby power supply (SPS) и служебный процессор.

- **Узел VMAX** — состоит из двух блоков управления, содержащих четыре четырехъядерных процессора Intel, до 128 Гбайт памяти и до 16 внешних портов для доступа со стороны хостов или SRDF-каналов.
- **Полка матричной интерфейсной платы** — содержит два независимых матричных коммутатора, обеспечивающих одноранговый обмен данными между блоками управления. У каждого блока управления имеется два подключения к MIBE-полке VMAX. Поскольку у каждого блока управления есть два отдельных физических пути к каждому другому блоку управления, проходящих через виртуальную матрицу (Virtual Matrix), получается внутреннее соединение с высокой

степенью доступности без единой точки отказа. Такая конструкция исключает необходимость в отдельных внутренних соединениях для данных, управления, обмена сообщениями и проведения тестирования среды и системы. Единого внутреннего соединения с высокой степенью доступности вполне достаточно для всего обмена данными между блоками управления, что существенно упрощает конструкцию.

- **Служебный процессор** — используется для консоли настройки и управления. Обеспечивает также возможности выдачи уведомлений и поддержки, чтобы к системе можно было обращаться в локальном или удаленном режиме. Служебный процессор автоматически уведомляет имеющийся у поставщика центр поддержки клиентов об обнаружении отказов компонентов или нарушениях среды.
- **Symmetrix Enginuity** — операционная среда для EMC Symmetrix. Enginuity осуществляет функции управления и обеспечивает оптимальный поток и целостность информации при ее прохождении через различные компоненты оборудования в системе Symmetrix. Она управляет всеми операциями Symmetrix и системными ресурсами с целью интеллектуальной оптимизации производительности. Enginuity обеспечивает доступность системы за счет расширенного мониторинга отказов, возможностей их обнаружения и исправления и обеспечивает одновременное выполнение функций сопровождения и обслуживания. Она также предлагает основу для специальных программных средств, предназначенных для аварийного восстановления, обеспечения непрерывности бизнес-процессов и управления устройствами хранения данных.

4.4.3. Архитектура Symmetrix VMAX

Каждый VMAX-узел содержит часть глобальной памяти и два блока управления, которые способны одновременно управлять внешними, внутренними и удаленными подключениями. VMAX-узел подключен к виртуальной матрице (Virtual Matrix) и позволяет динамически обращаться ко всем системным ресурсам, включая центральный процессор, память, накопители и хост-порты, и совместно использовать эти ресурсы любым хостом. Кроме того, VMAX-узлы могут добавляться без нарушений работы массива хранения данных, позволяя тем самым эффективно наращивать системные ресурсы. Как показано на рис. 4.13, виртуальная матрица поддерживает в системе до восьми VMAX-узлов.

Резюме

В данной главе были рассмотрены характеристики и основные компоненты современных интеллектуальных систем хранения данных. Также были рассмотрены различные типы систем хранения данных, как высокопроизво-

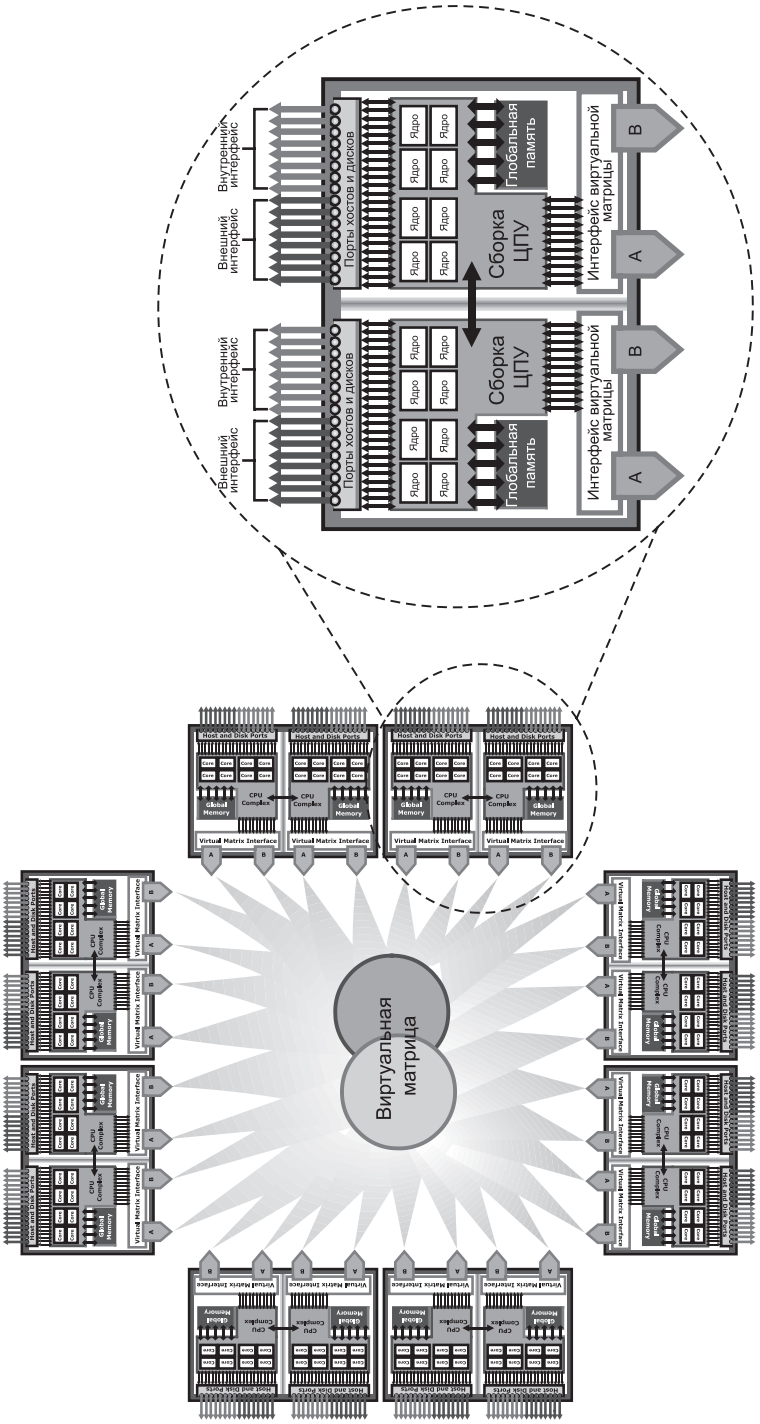


Рис. 4.13. Архитектура VMAX

дительные, так и среднего класса, а также разобраны их характеристики. Интеллектуальные системы хранения данных предоставляют организации следующие преимущества:

- увеличенную емкость хранилища данных;
- повышенную производительность ввода-вывода;
- упрощенное управление хранилищем данных;
- повышенную доступность данных;
- повышенную масштабируемость и гибкость;
- улучшенную систему обеспечения непрерывности бизнес-процессов;
- повышенную безопасность и улучшенное управление доступом.

Интеллектуальная система хранения данных является составной частью любого дата-центра. Большая емкость и высокая производительность, поддерживаемые интеллектуальной системой хранения данных, обусловливают необходимость совместного использования этой системы несколькими хостами. Интеллектуальные системы хранения данных позволяют предприятиям легко и безопасно осуществлять совместное использование данных.

Создание сетей хранения данных является гибкой информационно-ориентированной стратегией, позволяющей распространять возможности интеллектуальных систем хранения данных на все предприятие. Эта стратегия предоставляет универсальный способ управления фондами коммерческой информации, а также их совместного использования и защиты. Создание сетей хранения данных рассматривается в следующей части данной книги.

УПРАЖНЕНИЯ

1. Дайте характеристику механизмам обеспечения согласованности кэш-памяти и объясните их применение в среде с несколькими совместно используемыми блоками кэш-памяти.
2. Какой тип приложения выигрывает больше всего при записи в обход кэш-памяти? Обоснуйте свой ответ.
3. Дайте характеристику различным параметрам кэш-памяти: размеру страниц кэша, распределению кэш-памяти для чтения в сравнении с распределением ее для записи, объему упреждающей выборки и объему записи в обход кэш-памяти.
4. Для операций ввода-вывода в базе данных Oracle используется размер блока 4 Кбайт. Приложение, которое использует эту базу данных, сначала выполняет последовательную операцию чтения. Предложите и объясните подходящие значения для следующих параметров кэш-памяти: размера кэш-страницы, распределения кэш-памяти (для чтения и для записи), типа упреждающей выборки, объема записи в обход кэш-памяти.
5. Исследуйте архитектуру EMC VMAX и подготовьте презентацию по данной теме.

Раздел

II

Сетевые технологии хранения данных

В ЭТОМ РАЗДЕЛЕ

Глава 5. Оптоволоконные сети хранения данных
Глава 6. IP SAN и FCoE
Глава 7. Сетевые устройства хранения данных
Глава 8. Объектно-ориентированные и унифицированные хранилища данных

Глава 5
Оптоволоконные сети хранения данных

В настоящее время в организациях наблюдается взрывной рост объемов информации, нуждающейся в эффективном хранении, защите, оптимизации и контроле. Перед менеджерами дата-центров стоит сложная задача обеспечения низкозатратного и высокопродуктивного управления данными. Под эффективным управлением информацией подразумевается следующее.

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Архитектура Fibre Channel (FC)
- Стек протоколов Fibre Channel
- Порты Fibre Channel
- Адресация Fibre Channel
- Глобальные имена
- Зонирование
- Топологии Fibre Channel SAN
- Виртуализация хранилищ 1на уровне блоков
- Виртуальные сети хранения данных (SAN)

- **Своевременная передача информации бизнес-пользователям:** информация должна быть доступна бизнес-пользователям именно в тот момент, когда она им необходима. Лавинообразный рост объемов данных постоянной доступности, быстрое распространение новых серверов и приложений, распространение критически важных данных внутри предприятий и требование доступности данных 24 часа в сутки 7 дней в неделю — вот лишь часть задач, которые необходимо решить с целью обеспечения доступности информации в режиме реального времени.
- **Интеграция информационной инфраструктуры с бизнес-процессами:** инфраструктура хранения данных должна быть интегрирована с различными бизнес-процессами без угроз для безопасности и целостности.
- **Гибкая и отказоустойчивая архитектура хранилищ:** инфраструктура хранилищ должна обеспечить гибкость и отказоустойчивость с адаптацией под меняющиеся требования. Хранилища должны иметь

возможности масштабирования без снижения производительности приложений, и в то же время общая стоимость управления информацией не должна быть слишком высокой.

Систему хранения прямого подключения — Direct-attached storage (DAS) часто называют разрозненной средой хранения данных. Хосты «владеют» хранилищем, и на этих изолированных устройствах хранения трудно реализовать управление информацией и совместный доступ к ресурсам. Попытки организовать эти разрозненные данные привели к появлению *сетей хранения данных* (storage area network, SAN). SAN представляет собой высокоскоростную выделенную сеть серверов и совместно используемых устройств хранения данных. SAN обеспечивает объединение устройств хранения данных и упрощает централизованное управление ими. Она вполне адекватно отвечает требованиям, предъявляемым к хранению данных, предоставляя наиболее экономически выгодный вариант масштабирования, а также обеспечивает эффективное обслуживание и защиту данных. Виртуализированная SAN и виртуализация хранилищ данных на уровне блоков повышают степень использования хранилищ и улучшают совместную работу рассредоточенных ресурсов хранения данных. Реализация виртуализации в SAN приводит к повышению производительности, увеличению коэффициента использования ресурсов и открывает новые возможности по управлению ими.

Обычно SAN развертываются с использованием технологий Fibre Channel (FC) SAN и IP SAN. В Fibre Channel SAN для передачи данных, команд и информации о состоянии между серверами (или хостами) и устройствами хранения используется протокол Fibre Channel. В IP SAN для обмена данными используются протоколы на базе IP.

В настоящей главе подробно описывается суть технологии Fibre Channel (FC), на базе которой осуществляется развертывание сетей хранения данных, а также рассматриваются компоненты сети FC SAN, ее топология и виртуализация хранилищ на уровне блоков.

5.1. Fibre Channel: обзор

Основная конструкция инфраструктуры сетей хранения данных FC SAN формируется на базе архитектуры Fibre Channel (FC). Fibre Channel представляет собой высокоскоростную сетевую технологию на основе оптоволоконных кабелей с высокой пропускной способностью и медных кабелей, предназначенных для последовательной передачи данных. Технология FC была создана в ответ на требования повышения скоростей передачи данных между серверами и системами хранения данных. Хотя FC-сети впервые появились в 1988 году, процесс стандартизации FC начался лишь после создания в Американском национальном институте стандартизации (ANSI) рабочей группы по Fibre Channel (FCWG). К 1994 году был разработан новый стандарт высокоскоростного компьютерного соединения и была создана Ассоциация

разработчиков оптоволоконных каналов (FCA), учредителями которой стали 70 компаний. За разработку интерфейсов Fibre Channel отвечает технический комитет T11, являющийся комитетом внутри INCITS (Международного комитета по стандартам информационных технологий).

Важной особенностью сетевой технологии FC являются более высокие скорости передачи данных. Первоначальная реализация этой технологии обеспечивала пропускную способность 200 Мбайт/с (соответствует побитной скорости передачи данных 1 Гбит/с), что было выше скоростей, обеспечиваемых интерфейсом Ultra SCSI (20 Мбайт/с), который широко использовался в средах систем хранения данных прямого подключения. По сравнению с интерфейсом Ultra-SCSI, FC представляет собой существенный скачок в развитии технологии сетей хранения данных. Самая последняя реализация 16 GFC предлагает пропускную способность 3200 Мбайт/с (соответствует побитной скорости 16 Гбит/с), в то время как интерфейс Ultra640 SCSI обладает пропускной способностью 640 Мбайт/с. Архитектура FC легко масштабируется, и теоретически одна оптоволоконная сеть может обеспечить работу около 15 млн устройств.

5.2. Сеть хранения данных и ее эволюция

Сеть хранения данных (SAN) осуществляет перенос данных между серверами (или хостами) и устройствами хранения по сети FC (рис. 5.1). SAN позволяет объединять хранилища и обеспечивать их совместное использование сразу несколькими серверами. Тем самым по сравнению с архитектурой устройств прямого подключения повышается коэффициент использования ресурсов хранения данных и сокращается общее количество хранилищ, приобретаемых организацией и требующих от нее их содержания.

При объединении управление хранилищами становится централизованным и менее сложным, что приводит к дальнейшему сокращению расходов на управление информацией. Сеть хранения данных также позволяет организациям подключать географически удаленные серверы и хранилища.

В ранних реализациях FC SAN представляла собой простую группировку хостов и устройств хранения данных, подключенных к сети с помощью FC-концентратора, используемого в качестве устройства подключения. Эта конфигурация FC SAN известна как управляемая петля Fibre Channel (FC-AL). Использование концентраторов привело к появлению изолированных «островков» FC-AL SAN, так как концентраторы предоставляют ограниченные возможности по соединению и обеспечению пропускной способности.

Из-за своих ограничений концентраторы уступили дорогу высокопроизводительным FC-коммутаторам. Использование в SAN коммутаторов улучшило возможности соединения и повысило производительность, что позволило сетям FC SAN получить возможность широкомасштабного наращивания. Это повысило доступность данных для приложений в пределах предприятия. В настоящее время технология FC-AL в силу своих ограничений практически

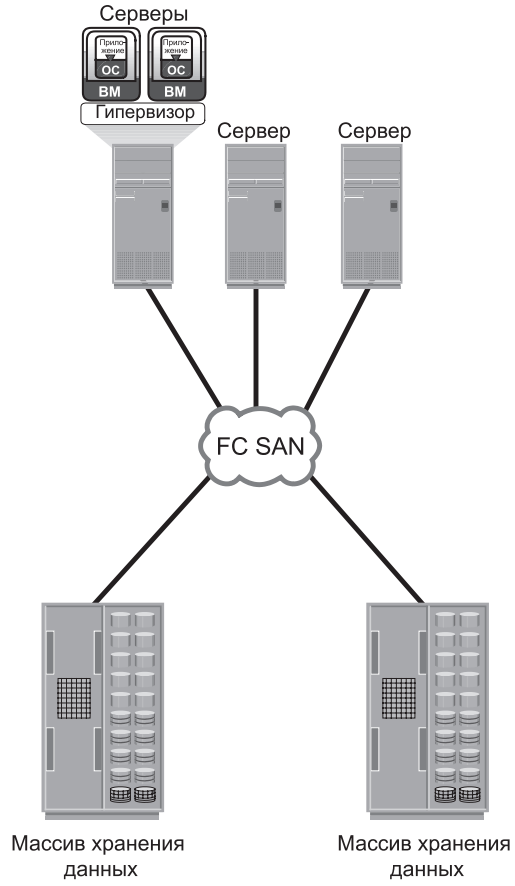


Рис. 5.1. Реализация FC SAN

вышла из употребления в сетях FC SAN, но все еще используется в качестве варианта внутренних интерфейсов для подключения к дисковым накопителям. На рис. 5.2 показана эволюция FC SAN от управляемой петли FC-AL до корпоративных сетей хранения данных.

5.3. Компоненты SAN

FC SAN представляет собой сеть серверов и совместно используемых устройств хранения данных. Серверы и хранилища в SAN являются оконечными точками или устройствами (так называемыми узлами). Инфраструктура FC SAN состоит из портов узлов, кабелей, разъемов и соединительных устройств (например, FC-коммутаторов или концентраторов), в нее также включаются программы управления сетью.

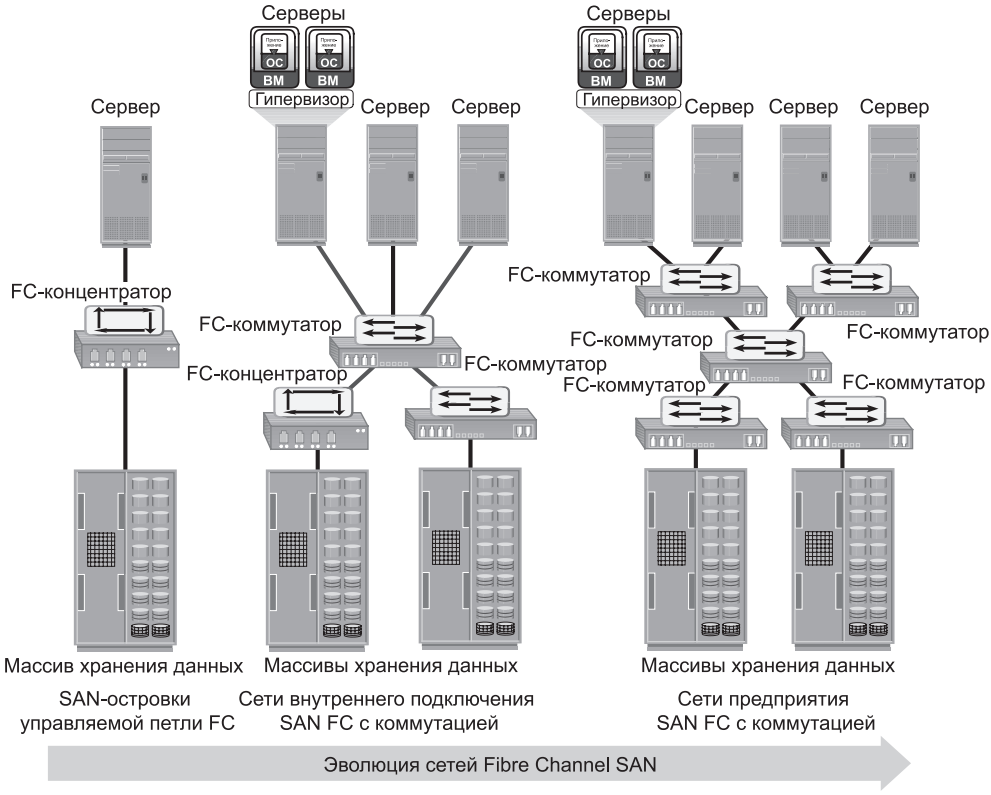


Рис. 5.2. Эволюция FC SAN

5.3.1. Порты узлов

В оптоволоконной сети такие оконечные устройства, как хосты, массивы хранения данных и библиотеки на магнитных лентах, называются узлами. Каждый узел является источником или пунктом назначения информации. Каждому узлу требуется один или более портов для обеспечения физического интерфейса для обмена данными с другими узлами. Эти порты являются неотъемлемыми компонентами адаптеров хоста, таких как адаптер главной шины (HBA), и контроллеров или адаптеров внешнего интерфейса устройства хранения данных. В FC-среде порт работает в полнодуплексном режиме передачи данных с передающим (Tx) и приемным (Rx) соединениями (см. рис. 5.3).

5.3.2. Кабели и разъемы

Сети хранения данных реализуются на базе оптоволоконных кабелей. Медный кабель может использоваться на малых расстояниях для подключений, относящихся к внутреннему интерфейсу, так как он обеспечивает приемлемое

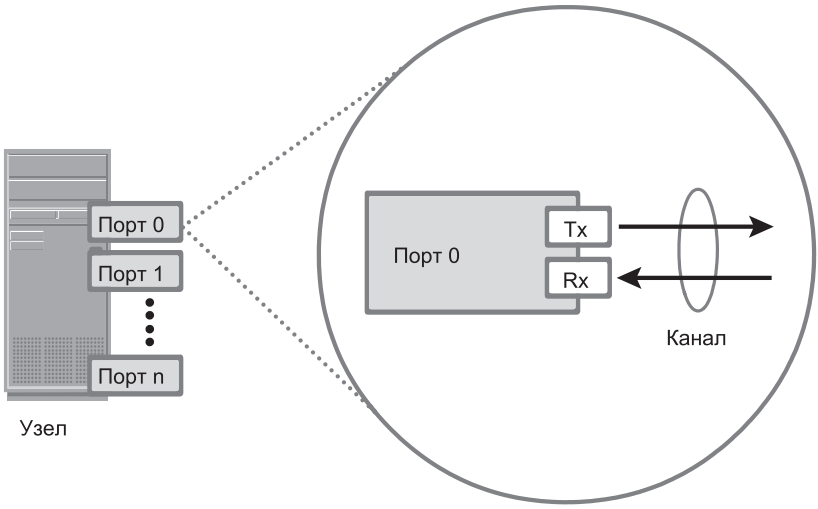


Рис. 5.3. Узлы, порты и каналы

соотношение «сигнал — шум» на расстояниях до 30 м. Оптоволоконный кабель передает данные в виде световых сигналов. Существуют два типа оптических кабелей: многомодовые и одномодовые.

Многомодовый кабель (MMF) передает множество световых лучей, одновременно проецируемых под разными углами на сердцевину кабеля (световод) (рис. 5.4, *а*). По ширине полосы пропускания многомодовые кабели подразделяются на OM1 (62,5 мкм), OM2 (50 мкм) и оптимизированный под лазер OM3 (50 мкм). Световые пучки, передаваемые по многомодовому кабелю, имеют тенденцию к рассеиванию и столкновению. Столкновения ослабляют сигнал после прохождения определенного расстояния, этот процесс известен как модовая дисперсия. Вследствие ухудшения качества (затухания) сигнала из-за модовой дисперсии многомодовый кабель используется, как правило, на небольших расстояниях.

Одномодовый оптический кабель (SMF) передает одиночный световой луч, проецируемый в центр жилы (рис. 5.4, *б*). Диаметр такого кабеля 7–11 мкм; наиболее широко применяемый диаметр — 9 мкм. Одиночный световой луч, передаваемый по одномодовому кабелю, проходит по прямой линии по центру сердцевины. Малый диаметр световода и одиночная световая волна позволяют ограничить модовую дисперсию. Среди всех типов оптоволоконного кабеля одномодовый кабель имеет минимальное затухание сигнала при максимальном расстоянии передачи (до 10 км). Одномодовый кабель используется для прокладки соединений на большие расстояния, и его возможности ограничены только мощностью лазера передатчика и чувствительностью приемника.

Как правило, многомодовые кабели используются в дата-центрах для прокладки линий связи на небольшие расстояния, в то время как одномодовые кабели используются для линий связи на больших расстояниях.

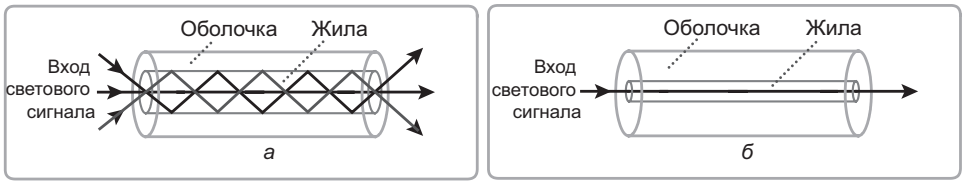


Рис. 5.4. Многомодовый и одномодовый оптический кабель

К концу кабеля присоединяется разъем, позволяющий быстро подключать и отключать кабель от порта. Для оптоволоконного кабеля наиболее широко используются два разъема: стандартный разъем (SC) (рис. 5.5, *а*) и разъем Lucent (LC) (рис. 5.5, *б*). Еще одним разъемом для оптоволоконного кабеля, часто используемым для патч-панелей, является прямой разъем (ST) (рис. 5.5, *в*).

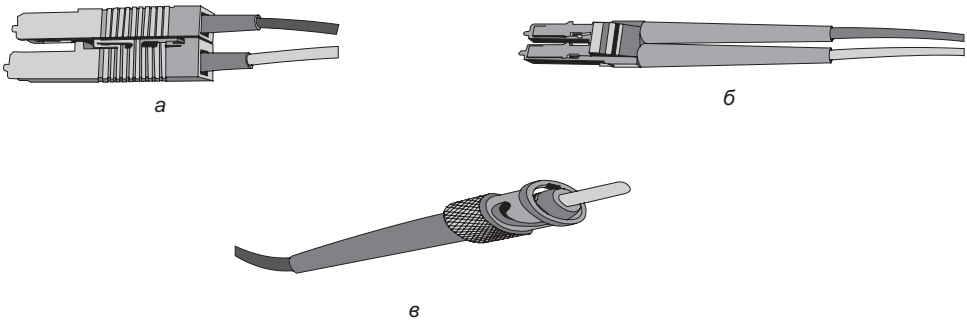


Рис. 5.5. Разъемы SC, LC и ST

5.3.3. Соединительные устройства

К соединительным устройствам, используемым в сетях FC SAN, можно отнести концентраторы, простые коммутаторы и коммутаторы класса Director.

Концентраторы применяются в качестве устройств связи в управляемой петле (FC-AL). Концентраторы физически соединяют узлы в логическую топологию «петля» или физическую топологию «звезда». Все узлы используют общую полосу пропускания, так как данные передаются через все точки подключения. В настоящее время из-за доступности дешевых и высокопроизводительных коммутаторов концентраторы в сетях FC SAN не используются.

Коммутаторы по сравнению с концентраторами представляют собой более интеллектуальные устройства и направляют данные непосредственно от одного физического порта к другому. Поэтому узлы не используют общую полосу пропускания. Вместо этого каждый узел имеет свой выделенный канал связи.

Коммутаторы класса Director являются устройствами высшего класса, имеющими большее количество портов и более высокую отказоустойчивость.

Коммутаторы бывают с фиксированным количеством портов или с модульной конструкцией. В модульных коммутаторах количество портов увеличивается путем установки в свободные слоты дополнительных карт портов. Коммутаторы класса Director всегда имеют модульную архитектуру, и количество портов в них увеличивается за счет вставки в корпус коммутатора дополнительных линейных карт или блейдов. Для обеспечения высокой доступности в коммутаторах высшего класса имеется избыточное число компонентов. Как у обычных коммутаторов, так и у коммутаторов класса Director для связи с серверами управления SAN имеются порты управления (последовательные или типа Ethernet).

Карта портов или блейд имеет несколько портов для подключения узлов и других FC-коммутаторов. Обычно в каждый слот порта устанавливается трансивер оптоволоконного канала (Fibre Channel transceiver), содержащий передающий (Tx) и приемный (Rx) каналы. В трансивере каналы Tx и Rx совместно используют общие цепи. Трансиверы внутри карты портов подключены к специализированной интегральной схеме, которую еще называют ASIC (application-specific integrated circuit) портов. Для обеспечения более высокой пропускной способности блейды в коммутаторах класса Director обычно содержат более одной ASIC.

5.3.4. Программы управления сетями хранения данных

Программы управления SAN берут под свой контроль интерфейсы между хостами, соединительными устройствами и массивами хранения данных. Они дают возможность обзора SAN-среды и позволяют управлять различными ресурсами с одной центральной консоли.

Программы позволяют выполнять ключевые функции управления, включая отображение устройств хранения, коммутаторов и серверов, мониторинг и выдачу оповещений об обнаруженных устройствах, а также зонирование (рассматриваемое в соответствующем разделе данной главы).

СРАВНЕНИЕ FC-КОММУТАТОРА И FC-КОНЦЕНТРАТОРА

Основная разница между коммутаторами и концентраторами — в возможностях масштабирования и достижения высокого уровня производительности. В коммутируемой топологии поддерживается более 15 млн устройств, в то время как управляемая петля FC-AL, реализованная с помощью концентраторов, поддерживает не более 126 узлов.

Коммутируемые матрицы обеспечивают максимальную ширину полосы пропускания между различными парами портов в матрице, что приводит к образованию масштабируемой архитектуры, поддерживающей одновременно множество связей.

Концентраторы единовременно поддерживают только одну связь. Они предлагают низкобюджетное решение по расширению возможностей подключения. Коммутаторы, напротив, могут использоваться для построения динамичных и высокопроизводительных коммутационных топологий, поддерживающих одновременную множественную связь. Коммутаторы обходятся дороже концентраторов.

5.4. Возможности соединений с применением FC

FC-архитектура поддерживает три основные топологии: «точка — точка», управляемая петля (FC-AL) и коммутируемое соединение.

5.4.1. «Точка — точка»

Самая простая FC-конфигурация — «точка — точка» — представляет собой два устройства, непосредственно подключенных друг к другу (рис. 5.6). Эта конфигурация обеспечивает выделенное соединение для передачи данных между узлами. Но она обладает ограниченными возможностями подключения, так как одновременно взаимодействовать друг с другом могут только два устройства. Более того, эта конфигурация не может наращиваться для включения большего числа узлов. Конфигурация «точка — точка» используется в стандартных системах хранения прямого подключения.

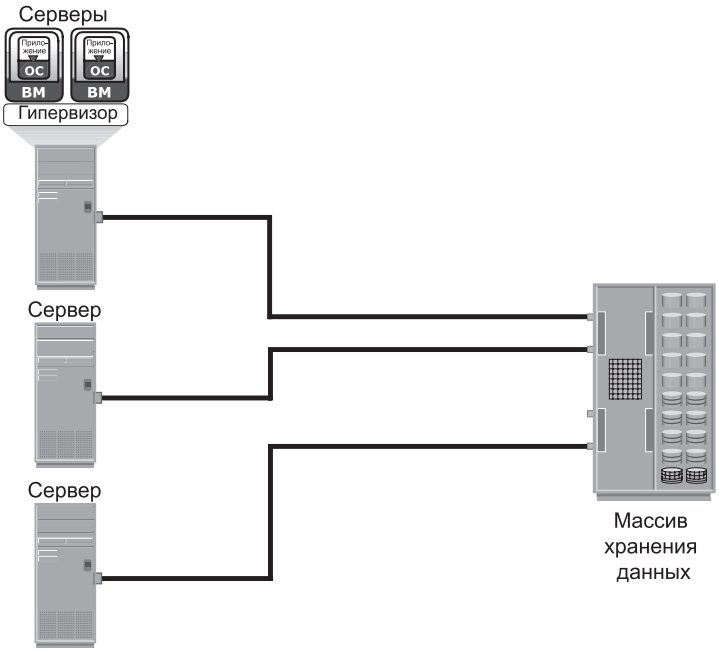


Рис. 5.6. Топология «точка — точка»

5.4.2. Управляемая петля Fibre Channel

В конфигурации типа «управляемая петля» (FC-AL) устройства подключаются к общей петле. FC-AL обладает характеристиками кольцевой топологии с эстафетным доступом и физической топологией «звезда». В сети FC-AL

каждое устройство соперничает с другими устройствами за возможность выполнения операций ввода-вывода. Чтобы получить контроль над петлей, устройства, подключенные к петле, должны выполнить «арбитраж». Единовременно операции ввода-вывода на петле может выполнять только одно устройство (рис. 5.7).

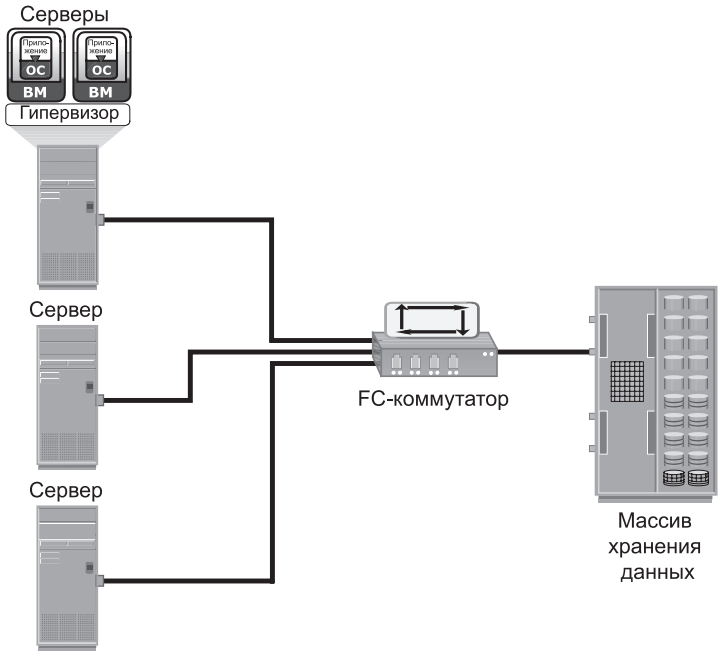


Рис. 5.7. Управляемая петля Fibre Channel

Являясь петлевой конфигурацией, FC-AL может быть реализована без соединительных устройств путем непосредственного подключения одного устройства к другому в кольцо посредством кабелей.

Тем не менее при реализации FC-AL могут также использоваться концентраторы, в этом случае управляемая петля физически соединяется в топологию «звезда».

Конфигурация FC-AL имеет следующие ограничения по наращиванию.

- FC-AL использует общую петлю, и единовременно операции ввода-вывода может выполнять только одно устройство. Так как каждое устройство в петле должно ожидать своей очереди для обработки запроса ввода-вывода, общая производительность в средах FC-AL невысока.
- FC-AL использует только 8-разрядную адресацию из 24 разрядов, допускаемых в технологии Fibre Channel (остальные 16 разрядов маскируются), и может назначать портам только 127 допустимых адресов.

Следовательно, в петле может поддерживаться до 127 устройств. Один адрес резервируется под необязательное подключение петли к порту FC-концентратора. Поэтому к петле может быть подключено до 126 узлов.

Добавление или удаление устройства приводит к повторной инициализации петли, что может вызвать кратковременную паузу в передаче данных в петле.

5.4.3. Коммутирующая матрица Fibre Channel

В отличие от петлевой конфигурации, коммутирующая матрица Fibre Channel switched fabric (FC-SW) обеспечивает выделенный информационный канал и возможность наращивания. Добавление или удаление устройства в коммутирующей матрице практически не прерывает ее работу и не влияет на выполняемую передачу данных между другими устройствами.

FC-SW также называют коммутируемой связной архитектурой (fabric connect). Она представляет собой логическое пространство, в котором все узлы взаимодействуют друг с другом в сети. Это виртуальное пространство может быть создано при помощи коммутатора или сети коммутаторов. Каждый коммутатор коммутируемой связной архитектуры содержит уникальный идентификатор домена, который является частью ее схемы адресации. В FC-SW узлы не используют общий контур, вместо этого данные передаются по выделенному информационному каналу между узлами. Каждый порт в связной архитектуре имеет для обмена данными уникальный 24-разрядный FC-адрес. Пример FC-SW показан на рис. 5.8.

В коммутируемой связной архитектуре соединение между двумя коммутаторами называется межкоммутаторной линией связи — Interswitch link (ISL). ISL-соединения позволяют подключать коммутаторы друг к другу для создания единого, более крупного коммутатора. ISL-соединения используются для передачи данных между хостом и хранилищем и трафика, управляющего коммутаторами от одного коммутатора к другому. Благодаря использованию ISL-соединений коммутируемая связная архитектура может быть расширена для соединения большего количества узлов.

Система коммутации может быть описана числом содержащихся в ней уровней. Количество уровней в этой системе берется из количества коммутаторов между двумя наиболее удаленными друг от друга узлами. Но это число основано на инфраструктуре, выстроенной системой коммутации, а не на том, как хранилище и сервер связаны через коммутаторы.

Когда число уровней в системе коммутации увеличивается, становится больше и то расстояние, которое должен проходить управляющий трафик системы, чтобы дойти до каждого коммутатора. Увеличение расстояния также увеличивает время на распространение и завершение события реконфигурации системы коммутации, например на распространение события добавления нового коммутатора или события настройки зоны. Двух- и трехуровневая архитектура системы коммутации показаны на рис. 5.9.

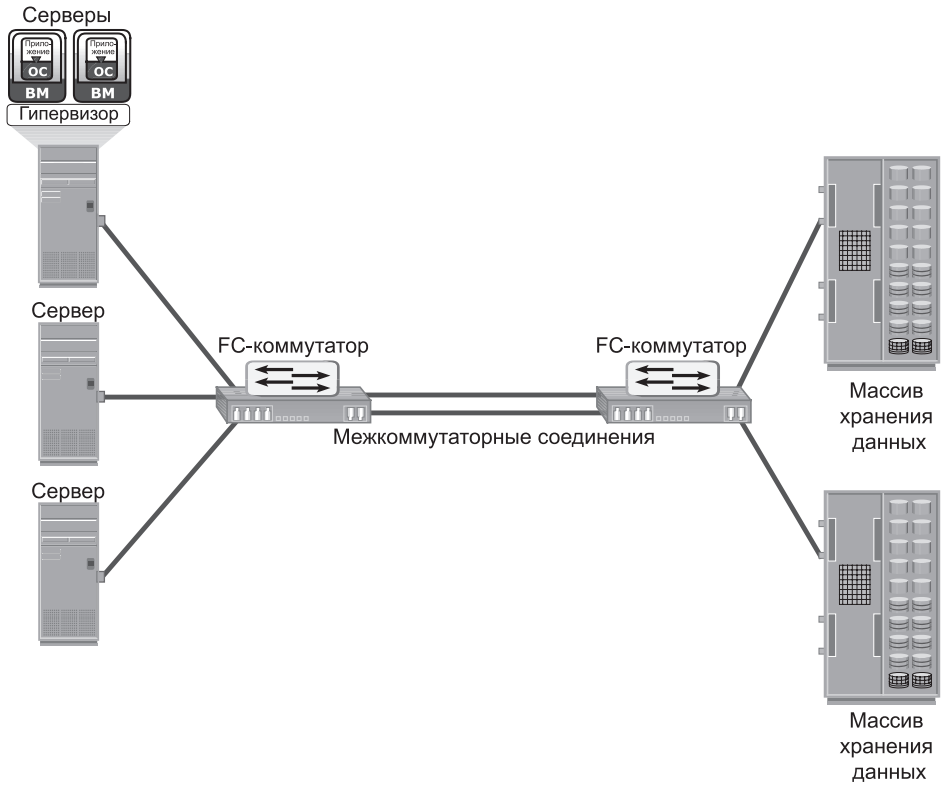


Рис. 5.8. Коммутирующая матрица Fibre Channel

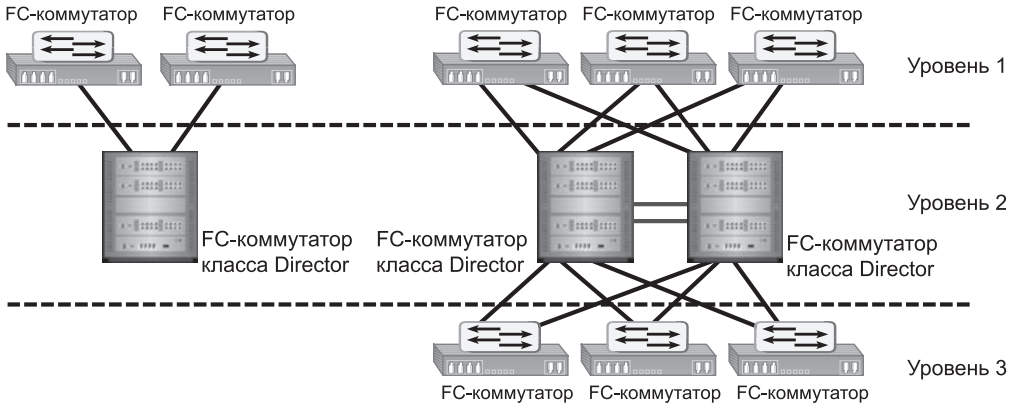


Рис. 5.9. Многоуровневая структура системы коммутации FC-SW

Передача данных в FC-SW

FC-SW использует коммутаторы, которые могут переключать поток данных между узлами непосредственно через порты коммутатора. Система коммутации задает маршруты кадрам между источником и приемником данных.

Как показано на рис. 5.10, если узел B намеревается осуществить обмен данными с узлом D, то сначала узлы должны осуществить индивидуальный вход в систему друг друга, а затем передать данные через коммутирующую матрицу FC-SW. Такая связь между инициатором и целевым устройством считается выделенной.

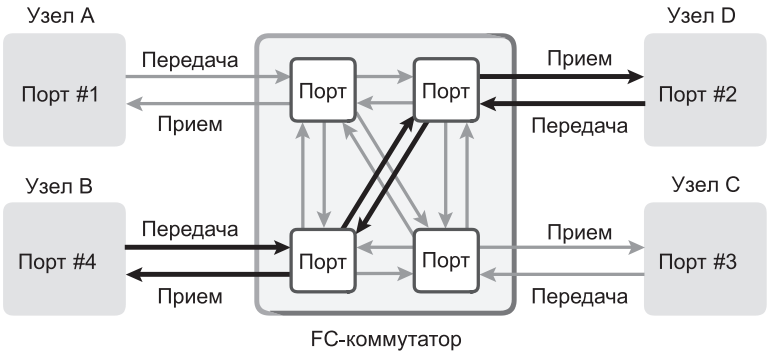


Рис. 5.10. Передача данных в системе коммутации FC-SW

5.5. Порты системы коммутации FC-SW

Порты в коммутируемой связной архитектуре FC-SW могут относиться к одному из следующих типов:

- **N_port** — оконечная точка в системе коммутации. Этот порт также известен как *узловой порт*. Как правило, это порт хоста (адаптера главной шины) или порт массива хранения данных, подключенный к коммутатору коммутирующей матрицы;
- **E_port** — это порт, формирующий соединение между двумя FC-коммутаторами. Он известен также как *порт расширения*. E_port FC-коммутатора соединяется с таким же портом другого FC-коммутатора той же системы коммутации посредством ISL-соединений;
- **F_port** — порт коммутатора, позволяющий подключить N_port. Он также известен как *порт коммутирующей матрицы*;
- **G_port** — универсальный порт, который может работать как E_port или F_port и определяет свои функциональные возможности автоматически во время инициализации.

Различные порты системы коммутации FC-SW показаны на рис. 5.11.

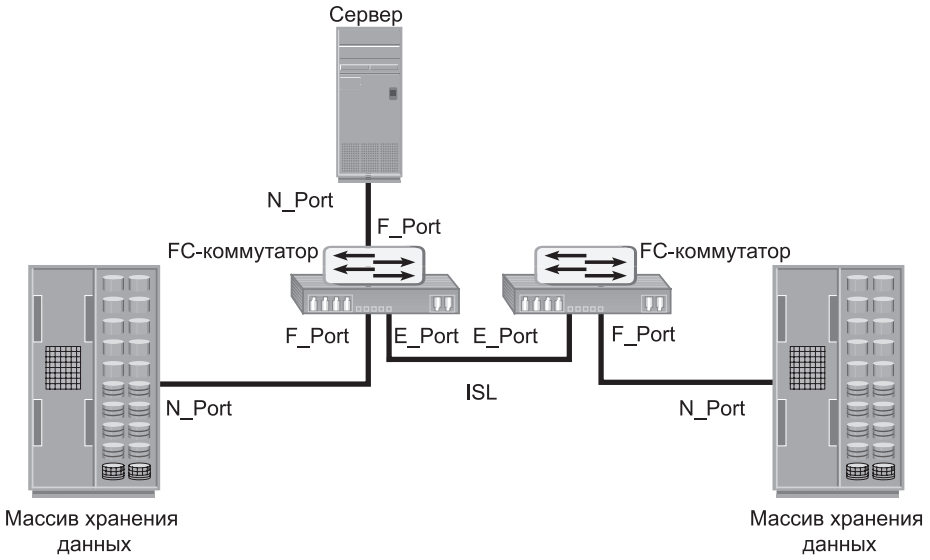


Рис. 5.11. Порты системы коммутации FC-SW

5.6. Архитектура Fibre Channel

Традиционно операционные системы хост-компьютера связывались с периферийными устройствами по таким каналам обмена данными, как ESCON и SCSI. Технологии каналов обеспечивают высокие уровни производительности при низких издержках протокола. Такая производительность достигается за счет статической природы каналов и высокого уровня интеграции аппаратных и программных средств, обусловленного технологиями каналов. Тем не менее эти технологии страдают от наследственных ограничений в плане количества устройств, которые могут быть подключены, и расстояния между этими устройствами.

В отличие от технологии каналов, сетевые технологии обладают большей гибкостью и предоставляют более широкие возможности по удаленности устройств.

Сетевые соединения обеспечивают лучшую масштабируемость и используют для обмена данными общую пропускную способность. Такая приспособляемость ведет к более высоким протокольным издержкам и снижает производительность.

FC-архитектура представляет собой настоящую интеграцию каналов и сетей и обладает рядом преимуществ как канальной, так и сетевой технологии. FC SAN использует протокол под названием *Fibre Channel Protocol* (FCP), предоставляющий скорость передачи данных, присущую каналам наряду с низкими издержками и масштабируемостью, характерными для сетевой технологии.

FCP формирует основную конструкцию инфраструктуры FC SAN. Fibre Channel предоставляет последовательный интерфейс передачи данных, работающий по медному проводу и оптоволокну. FCP является реализацией последовательного SCSI-протокола передачи данных по оптоволоконной сети. В FCP-архитектуре все внешние и удаленные устройства хранения данных, подключенные к SAN, видны в основной операционной системе как локальные устройства. У протокола FCP имеются следующие основные преимущества:

- устойчивая полоса пропускания для передачи данных на большие расстояния;
- поддержка в сети большого количества адресуемых устройств. Теоретически FC может поддерживать в сети более 15 млн адресов устройств;
- поддержка скорости до 16 Гбит/с (16 GFC).

5.6.1. Стек протоколов Fibre Channel

Понять коммуникационный протокол проще всего путем его представления в виде структуры независимых уровней. Протокол FCP определяет коммуникационный протокол на пяти уровнях, от FC-0 до FC-4 (за исключением уровня FC-3, который не реализован). В уровневой модели взаимодействия одноранговые уровни на каждом узле ведут диалог друг с другом через заданные протоколы. Стек протоколов FC показан на рис. 5.12.

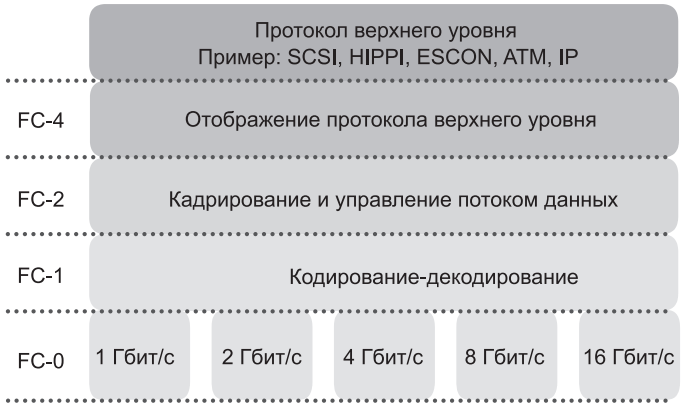


Рис. 5.12. Стек протоколов Fibre Channel

Уровень FC-4

FC-4 является самым верхним уровнем в стеке FCP. Он определяет интерфейсы приложений и способ, которым протоколы верхнего уровня (ULP) отображаются на более низкие FC-уровни. Стандарт FC определяет несколько протоколов, которые могут работать на уровне FC-4 (см. рис. 5.12). Некоторые

из этих протоколов включают SCSI, протокол High Performance Parallel Interface Framing Protocol (HIPPI), протокол Enterprise Storage Connectivity (ESCON), протокол Asynchronous Transfer Mode (ATM) и IP-протокол.

Уровень FC-2

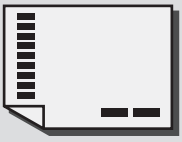
Уровень FC-2 обеспечивает адресацию, формирует структуру и осуществляет организацию данных (кадры, последовательности и обмены). Он также определяет службы системы коммутации, классы служб, порядок управления потоками и маршрутизацию.

Уровень FC-1

Уровень FC-1 определяет порядок кодирования данных до их передачи и их декодирование после приема. На передающем узле 8-разрядный символ кодируется в 10-разрядный символ передачи, который затем передается на приемный узел. На приемном узле 10-разрядный символ передается на уровень FC-1, который декодирует 10-разрядный символ в исходный 8-разрядный. В FC-соединениях со скоростью передачи данных 10 Гбит/с и выше используются 64–66-разрядные алгоритмы кодирования. Уровень FC-1 также определяет слова передачи, такие как разделители FC-кадра, идентифицирующие начало и конец кадра, и элементарные сигналы, служащие признаками событий в передающем порте. В дополнение ко всему этому уровень FC-1 выполняет инициализацию соединения и устранение ошибок.

Уровень FC-0

FC-0 является самым низким уровнем FCP-стека. Он определяет физический интерфейс, среду и порядок передачи битов. Спецификация FC-0 включает кабели, разъемы, а также оптические и электрические параметры для различных скоростей передачи данных. Передача в FC может проходить как в электрической, так и в оптической среде передачи данных.



В SAN-сетях мэйнфремов для высокоскоростного и быстродействующего подключения к контроллеру хранилища используется последовательный канал передачи данных Fibre Connectivity (FICON). Он был разработан в качестве замены Enterprise System Connection (ESCON) с целью поддержки подключаемых к мэйнфремам систем хранения данных.

5.6.2. Адресация в Fibre Channel

FC-адрес назначается динамически при регистрации порта узла в системе коммутации. FC-адрес имеет особый формат (рис. 5.13). Показанный здесь механизм адресации соответствует системе с коммутатором, используемым в качестве соединительного устройства.

Рис. 5.13. 24-разрядный FC-адрес порта типа N_Port

Первое поле FC-адреса содержит идентификатор домена коммутатора. Хотя это 8-разрядное поле, для идентификаторов домена доступны только 239 адресов, поскольку несколько адресов рассматриваются как специальные и оставлены для служб управления системой коммутации. Например, FFFFFC зарезервирован для сервера имен, а FFFFFE — для службы регистрации в системе. Идентификатор области используется для распознавания группы портов коммутатора, используемой для соединения узлов. Примером группы портов с общим идентификатором области может послужить установленная в коммутаторе карта портов. Последнее поле является идентификатором порта, с помощью которого распознается порт внутри группы.

Следовательно, максимально возможное число портов узлов в системе коммутации вычисляется таким образом:

239 доменов · 256 областей · 256 портов = 15 663 104.

ВИЗУАЛИЗАЦИЯ ИДЕНТИФИКАТОРА ПОРТА N_PORT (NPIV)

NPIV является конфигурацией Fibre Channel, которая позволяет нескольким идентификаторам порта N_port совместно использовать один общий физический N_port. Обычно **NPIV используется для предоставления SAN-хранилища виртуальным машинам в среде виртуализированного сервера.** С помощью NPIV несколько виртуальных машин хоста могут совместно использовать общий физический порт типа N_Port хоста, при этом у каждой виртуальной машины для этого физического порта узла будет использоваться собственный идентификатор N_Port_ID. Чтобы этот механизм работал, FC-коммутатор должен допускать использование NPIV.

5.6.3. Глобальные имена

Каждому устройству в среде FC назначается уникальный 64-разрядный идентификатор, называемый глобальным именем — World Wide Name (WWN). В среде Fibre Channel используются два типа WWN: глобальное имя узла (WWNN) и глобальное имя порта (WWPN). В отличие от FC-адреса, который назначается динамически, WWN является статическим именем каждого узла в FC-сети. Глобальные имена аналогичны адресам управления доступом к среде (MAC), используемым в IP-сетях. Глобальные имена прошиваются в аппаратных средствах или назначаются программным способом. WWN-имя используется в SAN в ряде конфигурационных определений для

идентификации устройств хранения данных и адаптеров главной шины (HBA). Связь глобальных имен с динамически создаваемыми FC-адресами для узлов сохраняется в сервере имен в FC-среде. На рис. 5.14 показаны примеры структуры WWN-имени для массива данных и адаптера главной шины.

Глобальное имя массива															
5	0	0	6	0	1	6	0	0	0	6	0	0	1	B	2
0101	0000	0000	0110	0000	0001	0110	0000	0000	0000	0110	0000	0000	0001	1011	0010
Тип формата	Идентификатор компании, 24 разряда						Порт	Исходный номер модели, 32 разряда							

Глобальное имя HBA															
1	0	0	0	0	0	0	0	c	9	2	0	d	c	4	0
Тип формата	Зарезервированные 12 разрядов			Идентификатор компании, 24 разряда						Данные, специфичные для компании, 24 разряда					

Рис. 5.14. Глобальные имена

5.6.4. FC-кадр

FC-кадр (рис. 5.15) состоит из следующих пяти частей: начала кадра (SOF), заголовка кадра, поля данных, контрольной суммы — cyclic redundancy check (CRC) и конца кадра (EOF).

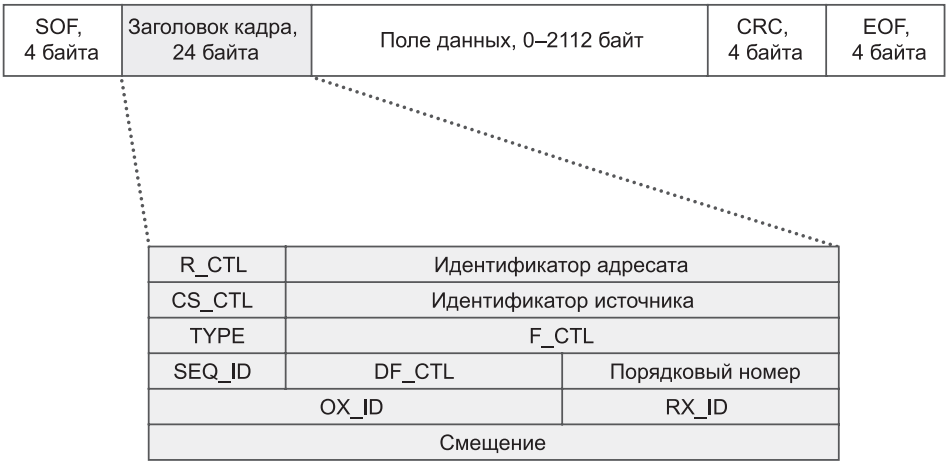


Рис. 5.15. Кадр FC

Начало кадра и конец кадра действуют в качестве разделителей. Кроме выполнения этой роли, начало кадра является также маркером, указывающим, является ли кадр первым в последовательности кадров.

Заголовок кадра состоит из 24 байтов и содержит информацию по адресации кадра. Он включает следующую информацию: идентификатор источника (S_ID), идентификатор адресата (D_ID), идентификатор последовательности (SEQ_ID), порядковый номер (SEQ_CNT), идентификатор вызывающего устройства (OX_ID), идентификатор отвечающего устройства (RX_ID), а также несколько полей управления.

Идентификатор источника (S_ID) и идентификатор адресата (D_ID) являются FC-адресами для порта-источника и порта-приемника соответственно. Идентификатор последовательности (SEQ_ID) и идентификатор вызывающего устройства (OX_ID) определяют кадр в качестве компонента определенной последовательности и обмена соответственно.

В заголовке кадра также определяются следующие поля:

- **управления маршрутизацией (R_CTL)** — это поле определяет принадлежность кадра. Он может быть либо кадром управления каналом связи, либо кадром данных. В кадрах управления каналом никакие пользовательские данные не переносятся. Эти кадры используются для настройки и обмена сообщениями. В отличие от них, в кадрах данных переносятся данные пользователя;
- **управления, относящегося к определенному классу (CS_CTL)** — это поле определяет скорость канала для передачи данных класса 1 и класса 4 (класс службы рассматривается в разделе 5.6.7 «Классы обслуживания»);
- **типа** — это поле описывает протокол верхнего уровня (ULP), передаваемый в кадре, если это кадр данных. Но если это кадр управления каналом, поле используется для подачи сигнала о событии, например: «Система коммутации занята». К примеру, если в поле ТИП значится 08 и кадр является кадром данных, это означает, что передача в FC будет вестись по протоколу SCSI;
- **управления полем данных (DF_CTL)** — однобайтовое поле, которое служит признаком присутствия в начале полезных данных каких-либо дополнительных заголовков. Данный механизм служит для расширения заголовочной информации за счет полезных данных;
- **управления кадром (F_CTL)** — трехбайтовое поле с информацией по управлению содержанием кадра. Например, один из битов в этом поле показывает, является ли этот кадр первой последовательностью обмена.

Поле данных в FC-кадре содержит полезную информацию объемом до 2112 байт собственно данных и ровно 36 байт служебных данных.

Контрольная сумма CRC способствует обнаружению ошибок в содержимом кадра. С ее помощью проверяется целостность данных, позволяющая убедиться в правильности полученного содержимого кадра. Контрольная сумма рассчитывается отправителем до кодировки на уровне FC-1. Таким же образом она рассчитывается получателем после декодирования на уровне FC-1.

5.6.5. Структура и организация FC-данных

Передача данных в FC-сети похожа на беседу двух людей, где кадр представляет слово, последовательность кадров — предложение, а обмен — сам разговор.

- **Обмен:** операция обмена позволяет двум портам узла идентифицировать набор информационных блоков и управлять ими. У каждого протокола верхнего уровня имеется своя, присущая ему информация, которая должна быть отправлена другому порту для выполнения конкретных операций. Такая информация называется информационным блоком. Структура информационных блоков определена на уровне FC-4. Блок отображается на последовательность. Обмен состоит из одной и более последовательностей.
- **Последовательность:** относится к непрерывному набору кадров, передаваемых от одного порта к другому. Последовательность соответствует информационному блоку, определяемому протоколом верхнего уровня (ULP).
- **Кадр:** является основным блоком данных, передаваемым на уровне 2. Каждый кадр может содержать до 2112 байт полезной информации.

5.6.6. Управление потоками

Управление потоками данных определяет скорость потока кадров данных во время передачи информации. Технология FC использует два механизма управления потоками: разрешение на передачу типа «буфер — буфер» (BB_Credit) и разрешение на сквозную передачу пакетов данных (EE_Credit).

BB_Credit

FC использует механизм BB_Credit для управления потоком. BB_Credit управляет максимальным количеством кадров, присутствующим в канале в любой момент времени. В системе коммутации управление типа BB_Credit может происходить между любыми двумя FC-портами. Передающий порт отслеживает количество свободных буферов приемника и продолжает отправлять кадры, если количество буферов больше нуля. Механизм BB_Credit использует элементарный сигнал «приемник готов» — Receiver Ready (R_RDY), свидетельствующий об освобождении буфера того порта, который передал этот сигнал.

EE_Credit

Функция разрешения сквозной передачи данных, известная как EE_Credit, аналогична механизму BB_Credit. Когда инициатор и целевое устройство учреждаются в качестве узлов, ведущих взаимный обмен данными, они обмениваются параметрами EE_Credit, что является частью регистрации порта. Механизм EE_Credit обеспечивает управление потоком только для трафика классов 1 и 2.

5.6.7. Классы обслуживания

С целью соответствия запросам широкого диапазона приложений стандартами FC определяются различные классы обслуживания. Эти классы и их характеристики приведены в табл. 5.1.

Таблица 5.1. Классы обслуживания FC

	КЛАСС 1	КЛАСС 2	КЛАСС 3
Вид связи	Выделенное соединение	Невыделенное соединение	Невыделенное соединение
Управление потоками	Разрешение сквозной передачи	Разрешение сквозной передачи Разрешение передачи «буфер — буфер»	Разрешение передачи «буфер — буфер»
Доставка кадра	Доставка по очереди	Очередность не гарантирована	Очередность не гарантирована
Подтверждение доставки кадра	С подтверждением	С подтверждением	Без подтверждения
Мультиплексирование	Нет	Да	Да
Коэффициент использования полосы пропускания	Низкий	Средний	Высокий

Еще один класс обслуживания — класс F — используется для управления системой коммутации. Класс F аналогичен классу 2 и обеспечивает уведомление о том, что кадры не доставлены.

5.7. Службы систем коммутации

Все FC-коммутаторы независимо от их производителя предоставляют общий набор служб, определенный в стандартах Fibre Channel. Эти службы доступны по конкретным, заранее заданным адресам. К числу этих служб, в частности, относятся сервер регистрации в системе коммутации — Fabric Login Server, контроллер этой системы — Fabric Controller, служба имен — Name Server и служба управления — Management Server (рис. 5.16).

Сервер регистрации в системе коммутации находится по предопределенному адресу FFFFFE и используется на начальной стадии процесса регистрации узлов системы.

Сервер имен (формально известный как *распределенный сервер имен* — Distributed Name Server) находится по предопределенному адресу FFFFFC и отвечает за регистрацию имен и управление портами узлов. Каждый коммутатор обменивается информацией своего сервера имен с другими коммутаторами системы коммутации с целью поддержки синхронизированной, распределенной службы имен.

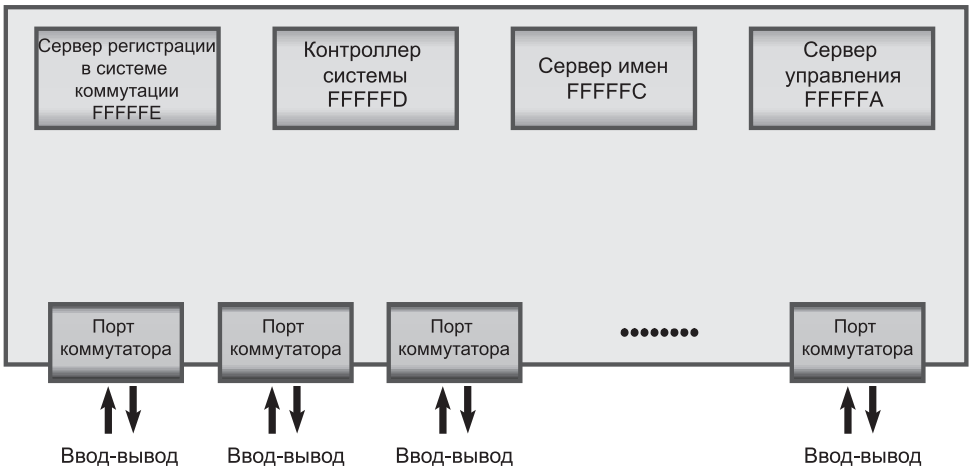


Рис. 5.16. Службы систем коммутации, предоставляемые FC-коммутаторами

У каждого коммутатора имеется *контроллер системы коммутации*, который находится по предопределенному адресу FFFFFD. Этот контроллер предоставляет службы как портам узлов, так и другим коммутаторам. Контроллер системы коммутации отвечает за управление и распределение уведомлений об изменении состояния — Registered State Change Notifications (RSCN-уведомлений) портам узлов, зарегистрированных с помощью этого контроллера. При наличии изменения в системе коммутации RSCN-уведомления рассылаются коммутатором по подключенным портам узлов. Контроллер системы коммутации также создает RSCN-уведомления коммутатора — Switch Registered State Change Notifications (SW-RSCN-уведомления), отправляемые каждому другому зарегистрированному домену (коммутатору) в системе коммутации. Эти RSCN-уведомления поддерживают актуальность имени сервера на всех коммутаторах системы.

FFFFFA является адресом Fibre Channel, предназначенным для сервера управления — Management Server. Этот сервер распределяется по каждому коммутатору в системе коммутации. Сервер управления позволяет программам управления FC SAN извлекать информацию и управлять системой коммутации.

5.8. Типы регистрации в системе коммутации

Службами системы коммутации определяются три типа регистрации.

- **Регистрация в системе коммутации — Fabric login (FLOGI)**. Выполняется между портом типа N_Port и портом типа F_Port. Для регистрации в системе коммутации узел отправляет FLOGI-кадр с WWNN- и WWPN-параметрами службе регистрации, находящейся по предопределенному FC-адресу FFFFFE (сервер регистрации

в системе коммутации). В свою очередь, коммутатор принимает регистрацию и возвращает кадр приемки — Accept (ACC) с назначенным FC-адресом для узла. Сразу же после FLOGI порт типа N_Port регистрируется с помощью локального сервера имен в коммутаторе, показывая свои WWNN, WWPN, тип порта, класс обслуживания, назначенный FC-адрес и т. д. После того как порт типа N_Port зарегистрировался, он может запросить у базы данных сервера имен информацию обо всех других зарегистрировавшихся портах.

- **Регистрация порта — Port login (PLOGI).** Выполняется между двумя портами типа N_Ports для установки сеанса. Порт-инициатор типа N_Port отправляет кадр PLOGI-запроса целевому порту типа N_Port, который его принимает. Целевой порт типа N_Port возвращает порту-инициатору того же типа кадр приемки ACC. Затем порты типа N_Ports обмениваются служебными параметрами, имеющими отношение к сеансу.
- **Регистрация процесса — Process login (PRLI).** Также выполняется между двумя портами типа N_Port. Эта регистрация относится к прикладным ULP-протоколам FC-4, например к SCSI. Если прикладным протоколом является SCSI, порты типа N_Ports обмениваются служебными параметрами, относящимися к SCSI-протоколу.

5.9. Зонирование

Зонирование — это функция FC-коммутатора, позволяющая объединять порты узлов внутри системы коммутации, получая при этом их логическую сегментацию в группы, внутри которых они могут связываться друг с другом (рис. 5.17).

Когда в базе данных сервера имен происходит изменение, контроллер системы коммутации отправляет уведомление об изменении состояния (RSCN) всем зарегистрированным узлам, которых оно касается. Если зонирование не настроено, контроллер системы коммутации отправляет RSCN-уведомление всем зарегистрированным узлам системы коммутации. Включение в число оповещаемых тех узлов, которых не касается это изменение, приводит в увеличению объема трафика, связанного с управлением системой коммутации. Для развитой системы коммутации объем FC-трафика, создаваемый этим процессом, может стать весьма существенным и оказать негативное влияние на трафик данных между хостом и хранилищем данных. Зонирование помогает ограничить количество RSCN-уведомлений в системе коммутации. При зонировании система коммутации отправляет RSCN-уведомление только тем узлам, в зоне которых произошло изменение.

Представители зон, зоны и наборы зон формируют иерархию, определяемую процессом зонирования (см. рис. 5.18). Набор зон состоит из группы зон, которая может быть активирована или дезактивирована в качестве единого

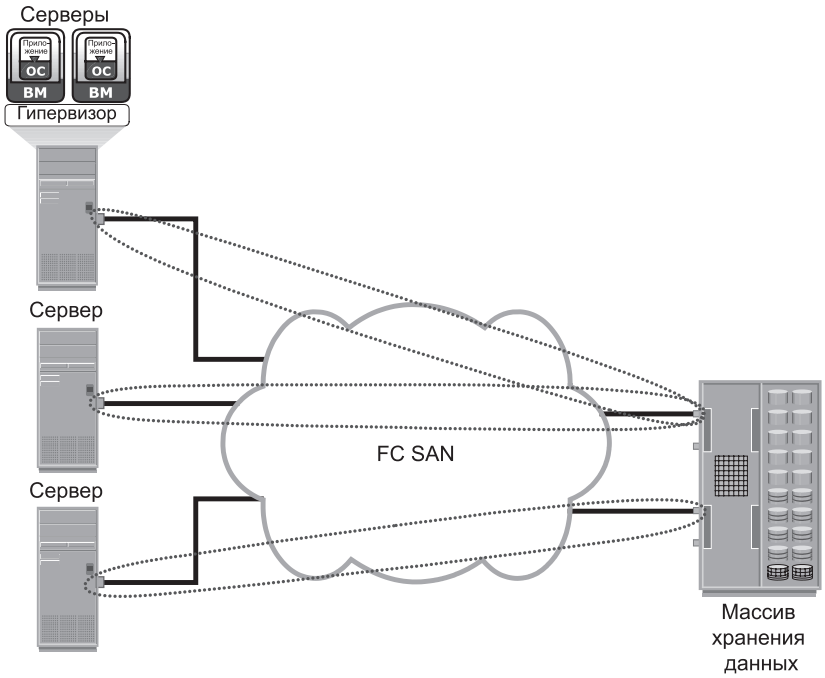


Рис. 5.17. Зонирование

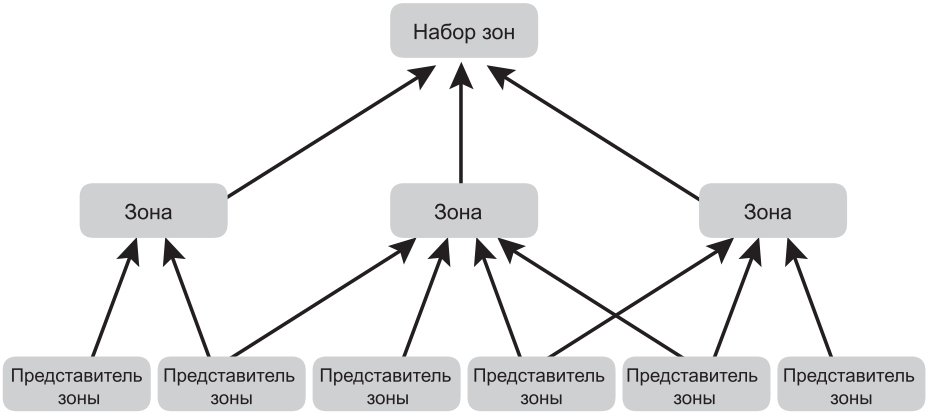


Рис. 5.18. Представители зон, зоны и наборы зон

целого в системе коммутации, где могут быть определены несколько наборов зон, но в любой момент времени может быть активирован только один такой набор. Представители зон являются узлами в SAN-сети, которые могут быть включены в зону. Представителями могут быть порты коммутаторов, HBA-порты и порты устройств хранения данных. Порт или узел может быть

представителем нескольких зон. Узлы, разбросанные по нескольким коммутаторам в системе коммутации, также могут быть сгруппированы в одну и ту же зону. Наборы зон называют также конфигурациями зон.

5.9.1. Типы зонирования

Зонирование можно разделить на следующие три типа.

- **Зонирование по портам.** Для определения зон используются физические FC-адреса портов коммутаторов. В зонировании по порту доступ к узлу определяется по физическому порту коммутатора, к которому подключен узел. Представителями зоны являются идентификатор порта (идентификатор домена коммутатора и номер порта), к которому подключен HBA-адаптер, и его целевые объекты (устройства хранения данных). Если узел перемещен на другой порт коммутатора в системе коммутации, зонирование должно быть изменено, чтобы позволить узлу в его новом порте быть представителем своей прежней зоны. Если порт HBA или устройство хранения данных выходит из строя, администратору просто нужно заменить отказавшее устройство, не меняя при этом конфигурацию зонирования.
- **Зонирование по глобальному имени (WWN-зонирование).** Для определения зон используются глобальные имена. Представителями зон являются уникальные WWN-адреса HBA и его целевые объекты (устройства хранения данных). Главным преимуществом WWN-зонирования является его гибкость. WWN-зонирование позволяет узлам перемещаться на другой порт коммутатора в системе коммутации и сохранять связь со своими партнерами по зоне без внесения изменений в конфигурацию зонирования. Такая возможность предоставляется в силу статичности WWN по отношению к порту узла.
- **Зонирование смешанного типа.** Объединяет достоинства WWN-зонирования и зонирования по портам. Использование зонирования смешанного типа позволяет привязать определенный порт узла к WWN-имени другого узла.

ЗОНИРОВАНИЕ ПО ОТДЕЛЬНОМУ HBA-АДАПТЕРУ

Зонирование по отдельному HBA-адаптеру (Single HBA zoning) считается наилучшим отработанным вариантом конфигурации набора зон. Зона отдельного HBA-адаптера состоит из одного HBA-порта и одного или нескольких портов устройств хранения данных. Зонирование по отдельному HBA-адаптеру исключает ненужные связи одного хоста с другим и сводит к минимуму количество RSCN-уведомлений. Применение данного типа зонирования в развитой системе коммутации приводит к настройке большого количества зон и увеличивает объем административных мероприятий. И тем не менее данный вариант повышает производительность FC SAN-сети и сокращает время выявления проблем, связанных с этой сетью.

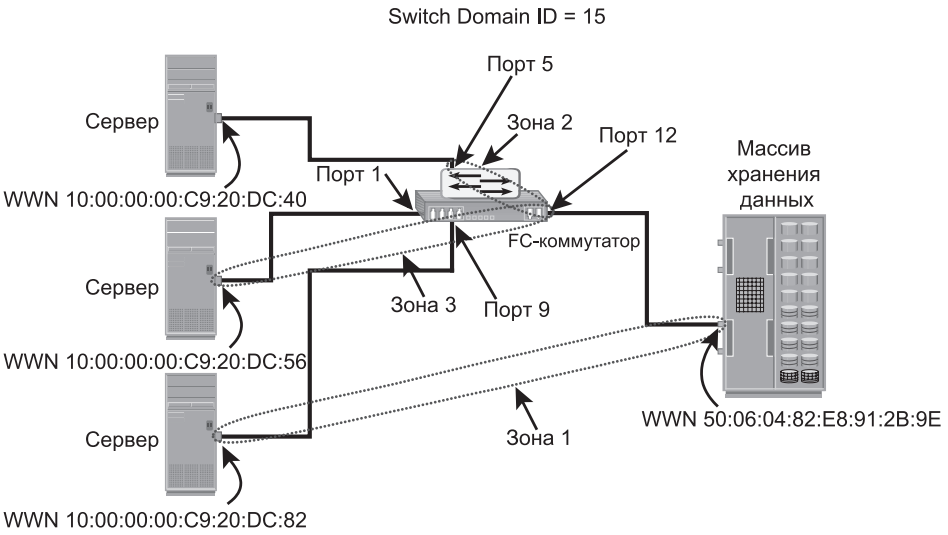


Рис. 5.19. Типы зонирования

Три типа зонирования в сети FC показаны на рис. 5.19.

Для управления доступом сервера к хранилищу зонирование используется совместно с LUN-маскированием. И тем не менее это два разных механизма. Зонирование происходит на уровне системы коммутации, а LUN-маскирование осуществляется на уровне массива.

5.10. Топологии FC SAN-сетей

При подключении устройств в системе коммутации применяются стандартные топологии. Одной из популярных топологий конструирования системы коммутации является топология «центр — периферия». Наиболее часто при реализациях FC SAN-сетей используются разновидности топологий систем коммутации типа «центр — периферия» и «решетка».

5.10.1. Топология типа «решетка»

Топология «решетка» может быть двух типов: полная или неполная. В полной решетке каждый коммутатор подключен к каждому другому коммутатору в топологии. Топология полной решетки подходит для тех случаев, когда число используемых коммутаторов невелико. При обычном развертывании используется до четырех обычных коммутаторов или коммутаторов класса Director, каждый из которых обслуживает строго локализованный трафик

от хоста к хранилищу данных. В топологии полной решетки для трафика от хоста к хранилищу данных требуется максимум одна межкоммутаторная линия связи (ISL) или транзитный участок. Но с увеличением количества коммутаторов увеличивается и количество портов коммутаторов, используемых для ISL. Тем самым уменьшается количество портов коммутаторов, доступных для обмена данными между узлами.

В топологии неполной решетки для достижения трафиком пункта назначения могут понадобиться несколько транзитных участков или межкоммутаторных линий связи. При топологии неполной решетки предлагаются более широкие возможности масштабирования, чем при топологии полной решетки. Но без надлежащего размещения хоста и устройств хранения данных управление трафиком в системе коммутации с топологией неполной решетки может быть усложнено, а ISL-линии из-за чрезмерного скопления трафика могут оказаться перегруженными. Топологии неполной и полной решетки показаны на рис. 5.20.

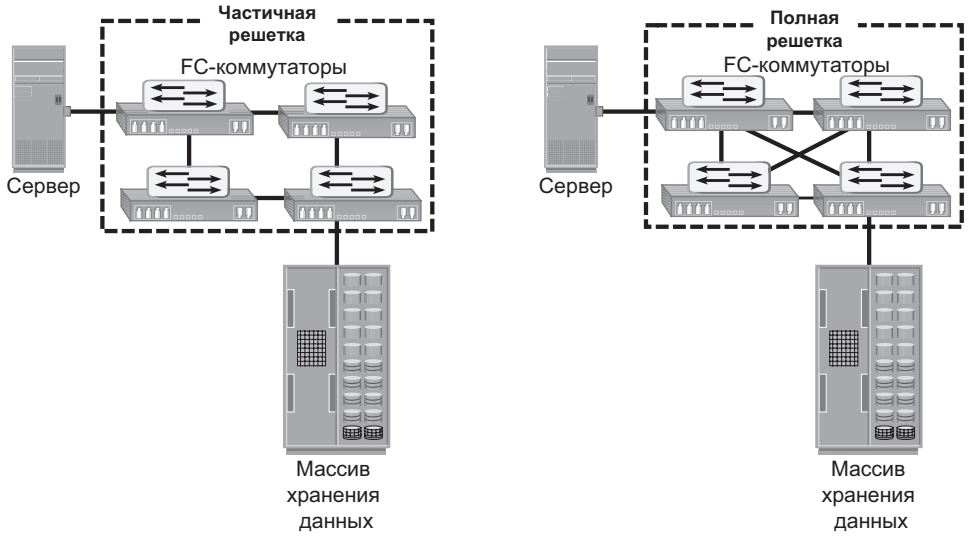


Рис. 5.20. Топологии неполной и полной решетки

ТОПОЛОГИЯ С ОДНИМ КОММУТАТОРОМ

Система коммутации с одним коммутатором состоит лишь из одного обычного коммутатора или одного коммутатора класса Director. Эта топология становится популярной, особенно в крупных дата-центрах, благодаря ее простоте. Увеличенное количество портов и модульная конструкция наряду с масштабируемой архитектурой обычных коммутаторов и коммутаторов класса Director позволяют конструкции SAN-сети начинаться со скромного масштаба и разрастаться по мере необходимости за счет добавления в коммутатор карт портов или блейдов вместо добавления новых коммутаторов.

5.10.2. Топология систем коммутации «центр — периферия»

В топологии системы коммутации «центр — периферия» существуют два типа коммутируемых уровней. Периферийный уровень обычно состоит из коммутаторов и предлагает недорогой способ добавления хостов к системе коммутации. Каждый коммутатор на периферийном уровне подключен через ISL-линии к коммутатору на центральном уровне.

Центральный уровень состоит, как правило, из коммутаторов класса Director, обеспечивающих высокий уровень доступности системы коммутации. Кроме этого, обычно весь трафик должен либо проходить через этот уровень, либо заканчиваться на нем. В этой конфигурации все устройства хранения данных подключены к центральному уровню, что позволяет трафику от хоста к хранилищу данных проходить только через одну ISL-линию. Хосты, требующие высокого уровня производительности, могут быть подключены непосредственно к центральному уровню, избегая при этом задержек на ISL-линии.

В топологии «центр — периферия» коммутаторы периферийного уровня не подключены друг к другу. Эта топология повышает возможности подключения внутри сети хранения данных при сохранении общего уровня загруженности портов. Если требуется расширение системы коммутации, к центру может быть подключен дополнительный периферийный коммутатор. Центральный уровень системы коммутации также расширяется при помощи добавления дополнительных обычных коммутаторов или коммутаторов класса Director. Исходя из количества коммутаторов центрального уровня, эта топология имеет разнообразные варианты исполнения, например топология с одним центром (рис. 5.21) или топология с двумя центрами (рис. 5.22). Чтобы превратить топологию с одним центром в топологию с двумя центрами, создаются новые ISL-линии для соединения каждого периферийного коммутатора с новым центральным коммутатором в системе коммутации.

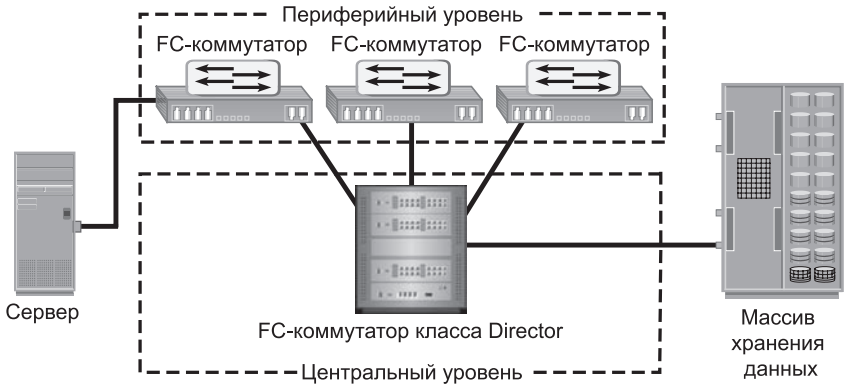


Рис. 5.21. Топология с одним центром

Преимущества и недостатки системы коммутации с топологией «центр — периферия»

Система коммутации с топологией «центр — периферия» обеспечивает максимум однотранзитный доступ ко всем устройствам хранения данных в системе. Поскольку трафик идет по обусловленной схеме (от периферии к центру и наоборот), топология «центр — периферия» облегчает вычисление загруженности межкоммутаторной линии связи (ISL) и режимов трафика. Так как в данной топологии порт коммутатора каждого уровня используется либо для хранилища данных, либо для хостов, можно легко определить, какие ресурсы приближаются к максимуму своих возможностей, что облегчает разработку набора правил для масштабирования и пропорционального распределения нагрузки.

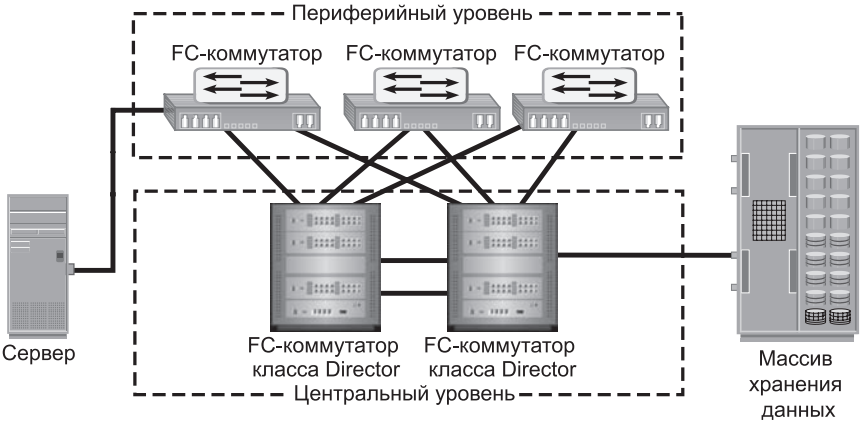


Рис. 5.22. Топология с двумя центрами

Системы коммутации с топологией «центр — периферия» наращиваются для получения более крупных сред путем добавления дополнительных центральных коммутаторов и их связывания друг с другом или путем добавления дополнительных периферийных коммутаторов. Этот метод позволяет расширять существующую простую модель «центр — периферия» или разворачивать систему коммутации в составную или более сложную модель «центр — периферия».

Тем не менее при использовании для системы коммутации топологии «центр — периферия» могут возникать проблемы, связанные с производительностью, поскольку масштабирование этой топологии подразумевает увеличение количества транзитных участков, то есть общего количества ISL-линий, через которые проходит пакет по пути от источника к пункту назначения. Согласно устоявшейся практике, количество транзитных участков на пути от хоста к хранилищу данных в топологии «центр — периферия» лучше сохранять неизменным, стремясь к использованию одного транзитного участка. Как правило, наличие большого количества транзитных участков означает высокий уровень задержек при передаче данных между источником и приемником.

При увеличении количества центров недопустимо сохранять ISL-линии от каждого центрального коммутатора к каждому периферийному. В таком случае конструкция системы коммутации меняется на составную или более сложную модель «центр — периферия» (рис. 5.23).

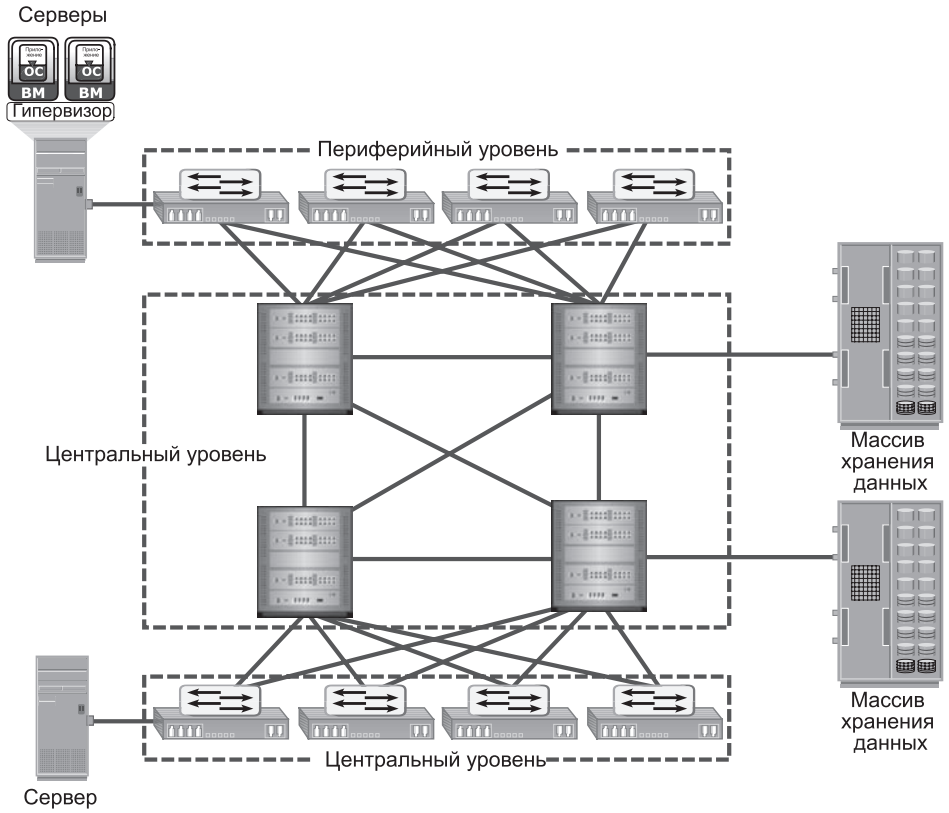


Рис. 5.23. Составная топология «центр — периферия»

РАЗВЕТВЛЕНИЕ (FAN-OUT) И ОБЪЕДИНЕНИЕ (FAN-IN)

Разветвление позволяет нескольким серверным портам обмениваться данными с одним портом хранилища. Подключение четырех серверов к одному порту хранилища приводит к коэффициенту разветвления 4. Этот коэффициент для порта хранилища зависит от возможностей системы хранения данных. Ключевым параметром, определяющим коэффициент разветвления порта хранилища, является возможность обработки данных, имеющаяся у внешнего интерфейса системы хранения данных. Обычно поставщик изделия указывает коэффициент разветвления системы хранения данных.

Объединение относится к количеству портов хранилища, используемых одним серверным портом. По аналогии с разветвлением ограничение по объединению основывается на возможности обработки данных, имеющейся у адаптера главной шины.

5.11. Виртуализация в SAN-среде

В данном разделе подробно рассматриваются две основанные на использовании сети технологии виртуализации в SAN-среде: виртуализация хранилища на уровне блоков и виртуальная SAN-сеть (VSAN).

5.11.1. Виртуализация хранилища на уровне блоков

За счет *виртуализации хранилища на уровне блоков* осуществляется объединение блочных устройств хранения данных (LUN-устройств) и появляется возможность предоставления виртуальных томов хранилищ независимо от физического хранилища данных, которое служит для них основой. Имеющийся в SAN уровень виртуализации позволяет абстрагироваться от особенностей физических средств хранения данных и создать пул хранилищ из разнородных устройств. Виртуальные тома создаются из пула хранилищ и назначаются хостам. Вместо направления на LUN-устройства в отдельных массивах хранения данных хосты направляются на виртуальные тома, предоставляемые уровнем виртуализации. Для хостов и массивов хранения данных уровень виртуализации показывается в виде целевого и инициирующего устройств соответственно. Уровень виртуализации отображает виртуальные тома на LUN-устройства на отдельно взятых массивах. Хосты остаются в неведении об операциях отображения и получают доступ к виртуальным томам, обращаясь к ним как к подключенным физическим хранилищам. Обычно уровнем виртуализации управляют через выделенное устройство виртуализации, к которому подключены хосты и массивы хранения данных.

На рис. 5.24 приведен пример виртуализированной среды с двумя физическими серверами, у каждого из которых имеется назначенный ему виртуальный том. Эти виртуальные тома используются серверами и отображены на LUN-устройства в массиве хранения данных. Когда к виртуальному тому отправляется запрос на ввод-вывод, он направляется через уровень виртуализации в сети хранения данных к LUN-устройствам, на которые выполнено отображение. В зависимости от возможностей устройства виртуализации архитектура может допускать более сложное отображение виртуальных томов на LUN-устройства массива.

Виртуализация хранилища на уровне блоков позволяет расширять тома хранилища при возрастающих потребностях приложения, не отключаясь при этом от системы. Она объединяет разнородные массивы хранения данных и позволяет получать прямой доступ к томам.

Виртуализация хранилища на уровне блоков обеспечивает также получение преимуществ миграции данных без нарушения режима работы. В традиционной SAN-среде миграция LUN-устройства с одного массива в другой производится в режиме автономной работы, поскольку хосты нуждаются в обновлении, чтобы в них отразилась новая конфигурация массива. В иных случаях для осуществления миграции данных из одного массива в другой, особенно в неоднородной среде, потребуются вычислительные ресурсы

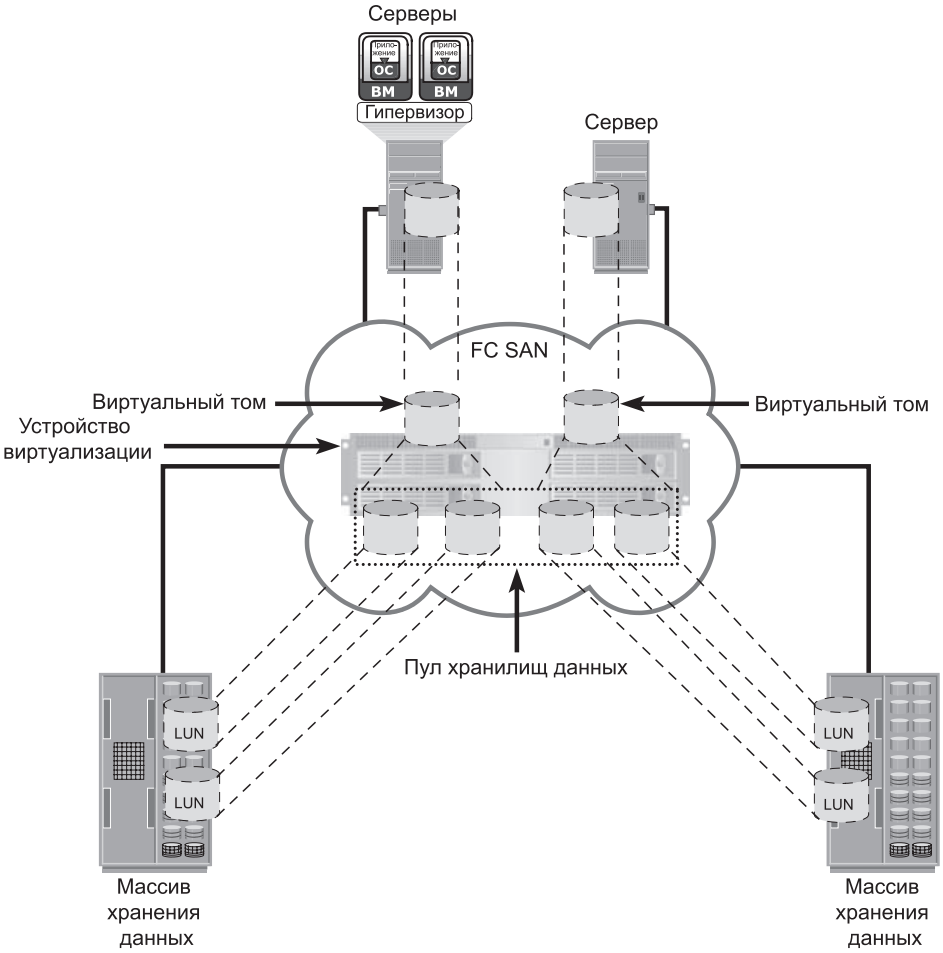


Рис. 5.24. Виртуализация хранилища на уровне блоков

центрального процессора хоста. При использовании виртуализации на уровне блоков внутренней миграцией данных занимается уровень виртуализации, позволяющий LUN-устройствам оставаться в подключенном состоянии и допускать обращения к себе в процессе миграции данных. Никаких физических изменений не требуется, поскольку хост по-прежнему указывает на те же самые виртуальные цели на уровне виртуализации. Но отображение информации на уровне виртуализации должно быть изменено. Эти изменения могут быть сделаны в динамическом режиме, невидимом конечному пользователю.

Ранее виртуализация хранилища на уровне блоков позволяла проводить миграцию данных без нарушения режима работы только в пределах дата-центра. На новой стадии развития этой технологии допускается не нарушающая режим работы миграция данных как внутри дата-центра, так и между

несколькими дата-центрами. При этом предоставляется возможность объединения уровней виртуализации нескольких дата-центров. Объединенные уровни виртуализации имеют централизованное управление и работают как единый уровень виртуализации, распространенный на несколько дата-центров (рис. 5.25). Тем самым допускается объединение ресурсов хранения на уровне блоков как внутри одного дата-центра, так и в рамках нескольких дата-центров. Виртуальные тома создаются из объединенных ресурсов хранения данных.

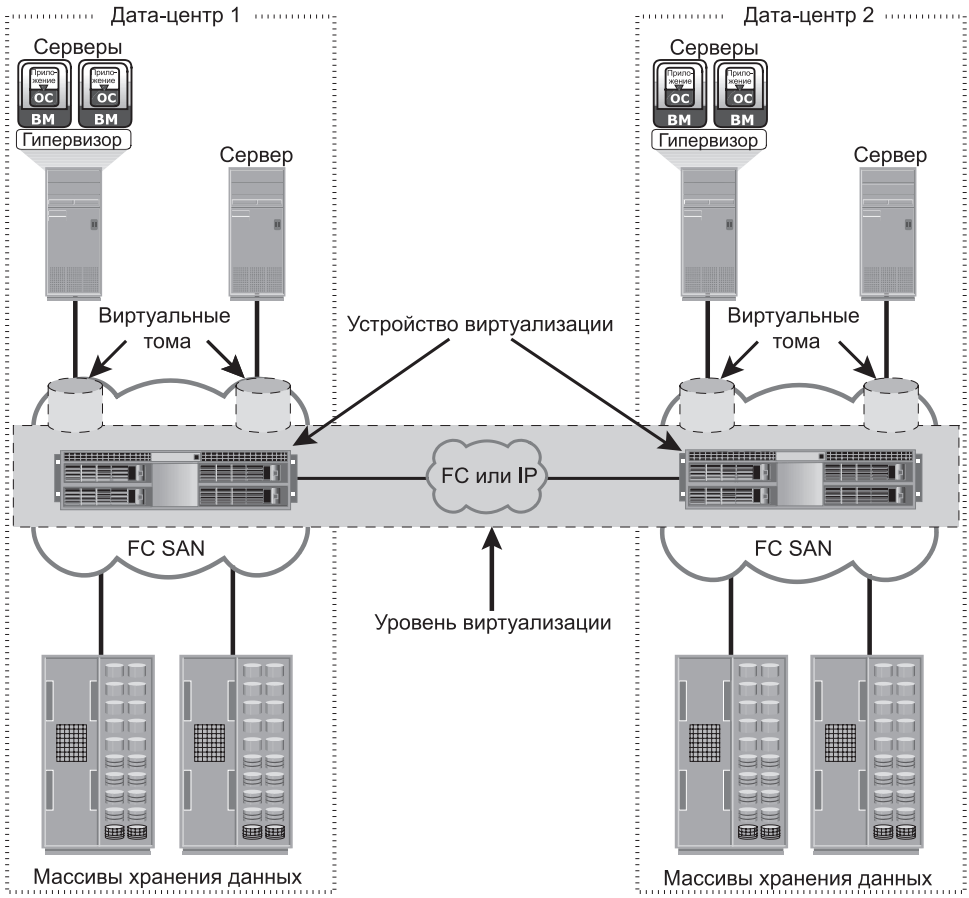


Рис. 5.25. Объединение хранилищ на уровне блоков в рамках нескольких дата-центров

5.11.2. Виртуальная SAN-сеть (VSAN)

Виртуальная SAN-сеть (называемая также виртуальной системой коммутации) является логической системой коммутации в FC SAN-сети, позволяющей производить обмен данными в группе узлов безотносительно их физического

расположения в системе коммутации. В VSAN группа хостов или портов хранилищ ведет обмен данными между собой с помощью виртуальной топологии, определяемой в физической SAN-сети. На одной физической SAN-сети может быть создано несколько VSAN-сетей. Каждая VSAN-сеть действует как независимая система коммутации со своим собственным набором служб такой системы, включая сервер имен и зонирование. Конфигурации, связанные с системой коммутации в одной VSAN-сети, не оказывают никакого влияния на трафик в других таких сетях.

VSAN-сети повышают безопасность, масштабируемость, доступность и управляемость SAN-сетей. VSAN-сети предоставляют более высокий уровень безопасности за счет изоляции конфиденциальных данных в VSAN-сети и ограничения доступа к ресурсам, размещенным внутри этой VSAN-сети. Один и тот же адрес Fibre Channel может быть назначен узлам в разных VSAN-сетях, повышая тем самым масштабируемость системы коммутации. События, препятствующие прохождению трафика в одной VSAN-сети, не выходят за пределы этой сети и не распространяются на другие VSAN-сети. VSAN-сети предлагают простой, гибкий и менее затратный способ управления сетями. Конфигурирование VSAN-сетей осуществляется проще и быстрее, чем создание отдельных физических FC SAN-сетей для различных групп узлов. Для перегруппировки узлов администратор просто изменяет настройки VSAN-сети, не перемещая узлы и не меняя кабельную разводку. Более подробно VSAN-сети рассматриваются в главе 14.

5.12. Практическая реализация концепций: EMC Connectrix и EMC VPLEX

Семейство EMC Connectrix предлагает в своей отрасли наиболее широкий выбор продукции, связанной с подключением сетевых хранилищ данных. Connectrix сочетает в себе высокоскоростное оптоволоконное соединение, весьма гибкие технологии коммутации, дополнительные средства для создания интеллектуальных сетевых хранилищ, основанных на IP-технологии, и возможности объединения по вводу-выводу с изделиями, поддерживающими технологию Fibre Channel через Ethernet.

EMC VPLEX является решением, относящимся к следующему поколению разработок для виртуализации на уровне блоков и мобильности данных внутри дата-центра, в пределах нескольких дата-центров и между дата-центрами. EMC VPLEX предоставляет объединение хранилищ путем создания группировки массивов хранения данных, которые могут находиться либо в одном и том же дата-центре, либо в нескольких дата-центрах. VPLEX также используется в качестве решения проблемы мобильности данных для сред облачных вычислений.

Самую последнюю информацию о связных изделиях Connectrix и VPLEX можно получить по адресу www.emc.com.

5.12.1. EMC Connectrix

Компания EMC предлагает под торговой маркой Connectrix следующие связные продукты (рис. 5.26):

- коммутаторы предприятия класса Director;
- коммутаторы отдела;
- универсальные коммутаторы.

Коммутатор предприятия класса Director

Коммутатор отдела

Универсальный коммутатор

Рис. 5.26. EMC Connectrix

Корпоративные коммутаторы типа Director идеально подходят для создания систем связи крупных предприятий. Они обеспечивают высокую плотность портов и избыточность компонентов. Коммутаторы класса Director, предназначенные для предприятия, развертываются в средах, обеспечивающих высокий уровень доступности информации, или в крупномасштабных средах и предлагают в каждом домене несколько сотен портов. Коммутаторы, предназначенные для отделов, больше всего подходят для рабочих групп и сред среднего уровня. Универсальные коммутаторы поддерживают различные протоколы, среди которых наряду с FC-протоколом можно назвать iSCSI, FCIP, FCoE и FICON. Кроме FC-портов коммутаторы Connectrix, как обычные, так и класса Director, имеют Ethernet-порты и последовательные порты для функций обмена данными и управления коммутацией. Программы управления Connectrix позволяют проводить конфигурирование и мониторинг коммутаторов Connectrix и управлять ими.

Коммутаторы Connectrix

Семейство коммутаторов Connectrix от компании EMC включает серии B и MDS. Эти коммутаторы разработаны в соответствии с требованиями, предъявляемыми к подобным изделиям, применяемым в масштабах рабочих групп, отделов и предприятий, и имеют архитектуру без применения блокировок, позволяющую работать в разнородных средах. Архитектура без

блокировок означает возможность для коммутатора одновременно обрабатывать независимые друг от друга пакеты, поскольку у коммутатора имеется достаточное количество внутренних ресурсов, чтобы справиться с максимальными скоростями обмена данными со всеми портами. Особенностями этих коммутаторов, обеспечивающими их высокую доступность, являются не нарушающее рабочий процесс обновление программного обеспечения и портов, а также наличие избыточных и допускающих горячую замену компонентов. Управлять этими коммутаторами можно посредством командной строки, а также HTTP и автономных средств с графическим интерфейсом пользователя.

Коммутаторы Connectrix класса Director

EMC предлагает семейство высокопроизводительных коммутаторов Connectrix класса Director. Их модульная архитектура предлагает высокий уровень масштабируемости, предоставляя более 500 портов. Эти коммутаторы пригодны для объединения серверов и хранилищ в масштабе предприятий. Коммутаторы этого класса обладают избыточностью компонентов для обеспечения высокой доступности и позволяют вести многопротокольный обмен данными как в среде мэйнфреймов, так и в среде открытых систем. Коммутаторы Connectrix класса Director предлагают высокоскоростной обмен данными (до 16 Гбит/с) и поддерживают объединение ISL-линий. Управлять коммутаторами Connectrix класса Director, как и обычными коммутаторами, можно посредством командной строки или других инструментальных средств, имеющих графический интерфейс пользователя.

Универсальные коммутаторы Connectrix

Универсальные коммутаторы обеспечивают поддержку сразу нескольких протоколов, таких как FC, FCIP, iSCSI, FCoE и FICON. Они выполняют преобразование протоколов и осуществляют маршрутизацию кадров между двумя разнородными сетями, например FC и IP. Возможность работы сразу по нескольким протоколам дает массу преимуществ, включая расширение SAN-сетей на большие расстояния, увеличение объемов совместно используемых ресурсов и упрощение управления. Универсальные коммутаторы Connectrix включают FCoE-коммутаторы, FCIP-маршрутизаторы, iSCSI-шлюзы и т. д.

Средства управления Connectrix

Отслеживание работы FC-коммутаторов в системе коммутации и управление ими проводится несколькими способами. Управление отдельными коммутаторами ведется с помощью командной строки или инструментальных средств в среде браузера.

Утилиты командной строки, такие как Telnet и Secure Shell (SSH), используются для регистрации в коммутаторе по IP-сети и выдаче команд через интерфейс командной строки. Основная цель использования интерфейса командной строки заключается в автоматизированном управлении большим

количеством обычных коммутаторов или коммутаторов класса Director с помощью сценариев. Средства на основе браузера предоставляют графический интерфейс пользователя. Они также отображают топологическую схему.

Управление всей системой коммутации и отслеживание ее работы осуществляются с помощью средств поставщика оборудования и стороннего программного обеспечения на основе простого протокола управления сетью — Simple Network Management Protocol (SNMP).

Единый интерфейс управления сетью хранения данных обеспечивается диспетчером EMC ControlCenter SAN Manager. Этот SAN-диспетчер позволяет администратору обнаруживать и отслеживать сложные разнородные SAN-среды, а также заниматься их настройкой и управлением ими. Он позволяет оптимизировать и централизовать операции по управлению SAN-сетью в условиях применения сетей хранения данных и устройств от разных производителей. Диспетчер позволяет администраторам хранилищ согласованно управлять SAN-зонами и LUN-маскированием в SAN-массивах и коммутаторах от различных производителей. EMC ControlCenter SAN Manager поддерживает также виртуальные среды, включая VMware, Symmetrix Virtual Provisioning и виртуальные сети хранения данных.

EMC ProSphere представляет собой недавно созданный инструмент с дополнительными функциональными возможностями, специально предназначенными для облачных сред. В ожидаемые выпуски EMC ProSphere будут включены все функциональные возможности EMC ControlCenter.

5.12.2. EMC VPLEX

EMC VPLEX предоставляет инфраструктуру виртуального хранилища, позволяющую объединять разнородные ресурсы хранения данных как внутри дата-центра, так и в рамках нескольких дата-центров. Устройства VPLEX размещаются между серверами и разнородными устройствами хранения данных. Они формируют пул распределяемых ресурсов хранилищ на уровне блоков и позволяют создавать из пула виртуальные тома хранилищ. Затем эти виртуальные тома выделяются серверам. Отображение виртуальных хранилищ на физические от серверов скрывается.

VPLEX предоставляет возможность перемещения данных по физическим устройствам хранения без нарушения режима работы с целью достижения сбалансированной нагрузки со стороны приложений и позволяет осуществлять как локальный, так и удаленный доступ к данным. Администратор может изменять отображение виртуальных томов на физические в динамическом режиме. Это позволяет виртуальным томам перемещаться по массивам хранения данных, не выпадая из производственного процесса.

В VPLEX используется уникальная архитектура кластеризации и согласованности распределенной кэш-памяти, что позволяет нескольким хостам в двух разных местах иметь доступ к одной и той же копии данных. Тем самым исключаются рабочие издержки и экономится время, необходимое на копирование и распространение данных по разным местам. VPLEX также

обеспечивает возможность зеркалирования данных виртуального тома как в одном и том же месте, так и в нескольких местах. Это позволяет хостам из разных дата-центров получать доступ к согласованным кэшированным копиям одного и того же виртуального тома. На практике эта возможность применяется для переноса данных, балансировки нагрузки и обеспечения высокой степени доступности данных между дата-центрами.

Чтобы избежать простоев приложений из-за отключения одного дата-центра, работа может быть быстро переброшена на другой дата-центр. Доступ приложений к тому же самому виртуальному тому при этом будет обеспечен и не прервется благодаря мобильности данных.

Семейство продуктов VPLEX

Семейство VPLEX состоит из трех продуктов: VPLEX Local, VPLEX Metro и VPLEX Geo.

EMC VPLEX Local производит локальное объединение, обеспечивающее упрощенное управление и мобильность данных без нарушения режима работы в разнородных массивах в пределах одного дата-центра. EMC VPLEX Metro производит распределенное объединение, обеспечивающее доступ к данным и их мобильность среди двух VPEX-кластеров в пределах одинаково проходимых по времени расстояний и при поддержке задержек в обоих направлениях, не превышающих 5 мс. EMC VPLEX Geo обеспечивает доступ к данным и их мобильность между двумя VPLEX-кластерами в пределах асинхронных дистанций, поддерживающих задержки в обоих направлениях, не превышающие 50 мс.

Резюме

Сеть хранения данных FC SAN позволяет объединять системы хранения и обеспечивает предприятиям существенную экономию средств при создании инфраструктуры хранения данных. Использование SAN-сети снижает общие эксплуатационные расходы и сокращает простои. Виртуализация хранилищ и сетей хранения данных еще больше упрощает управление ресурсами и снижает расходы. С падением цен на оборудование внедрение FC SAN-сетей расширилось, а с отработкой более четких стандартов на сети хранения данных спрос на SAN-сети увеличился.

В данной главе были подробно рассмотрены компоненты сетей хранения данных, топологии этих сетей и положенная в их основу FC-технология. Эта технология отвечает современным требованиям по надежности и высокой производительности приложений. В главе также рассмотрена виртуализация в SAN-среде.

По сравнению с ранними реализациями SAN-сетей наблюдается существенный рост возможностей взаимодействия FC-коммутаторов от разных производителей. Благодаря стандартам, опубликованным специальной

рабочей группой внутри T11, занимающейся FC SAN-маршрутизацией, а также появлению новых продуктов, предлагаемых поставщиками, сейчас происходят революционные изменения в способах развертывания и работе сетей хранения данных на основе оптоволоконных технологий.

Хотя FC SAN-сети исключили появление «островков» хранилищ данных, реализация этих сетей требует дополнительного оборудования и построения соответствующей инфраструктуры на предприятии. Появление технологий iSCSI и FCIP, рассматриваемых в главе 6, дало толчок к объединению FC SAN-сетей с IP-технологией, предоставляя при этом весьма экономичный метод использования уже существующей инфраструктуры на основе IP-технологии для создания сетей хранения данных.

УПРАЖНЕНИЯ

1. Что такое зонирование? Опишите сценарий, при котором:
   - WWN-зонирование будет предпочтительнее зонирования по портам;
   - зонирование по портам будет предпочтительнее WWN-зонирования.
2. Опишите процесс назначения FC-адреса узлу при первоначальной регистрации в сети.
3. В топологии полной решетки имеется подключение 17 коммутаторов с 16 портами в каждом. Сколько портов будет доступно для обеспечения связи хостов с хранилищами?
4. Для чего в системе коммутации с FC-коммутаторами предназначены сервер имен и контроллер системы коммутации?
5. Как работает управление потоками данных в FC-сети?
6. Объясните суть миграции хранилища данных с использованием виртуализированного хранилища на уровне блоков. Сравните эту миграцию с традиционными методами миграции.
7. Каким образом VSAN-сети улучшают управляемость FC SAN-сетей?

Глава 6
IP SAN и FCoE

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Протокол iSCSI
- Непосредственное и мостовое iSCSI-подключение
- Протокол FCIP
- Протокол FCoE

Традиционные SAN-сети позволяют осуществлять блочный ввод-вывод по оптоволоконному каналу, предоставляя при этом высокий уровень производительности и широкие возможности масштабирования. Получение этих преимуществ FC SAN-сетей сопряжено с дополнительными затратами на приобретение FC-компонентов, таких как FC HBA и коммутаторы. Обычно у организаций уже имеется инфраструктура на базе технологии Internet Protocol (IP), которая может быть задействована в сетях хранения данных. Развитие технологий позволило использовать IP для переноса блоков ввода-вывода по IP-сети. Эта технология называется IP SAN. IP является вполне устоявшейся технологией, и использование IP в качестве варианта построения сети хранения данных обеспечивает ряд преимуществ. Когда блок ввода-вывода переправляется по IP, может быть задействована уже существующая сетевая инфраструктура, что будет экономически выгоднее, чем вкладывание средств в новую SAN-инфраструктуру. Кроме того, сейчас в IP-сетях могут использоваться многие надежные и отработанные средства обеспечения безопасности. В сетях на основе IP-технологии уже давно используются решения географически распределенного аварийного восстановления — disaster recovery (DR). С появлением технологии IP SAN организации получили возможность расширения географического охвата своей инфраструктуры хранения данных.

В качестве механизма переноса данных IP усиливается двумя протоколами: Internet SCSI (iSCSI) и Fibre Channel over IP (FCIP). iSCSI представляет собой передаваемую по IP-каналу инкапсуляцию ввода-вывода SCSI. А FCIP представляет собой протокол, в котором для туннелирования системы коммутации FC через IP-сеть используется FCIP-элемент, например FCIP-шлюз. В протоколе FCIP FC-кадры инкапсулируются в полезную нагрузку IP.

Реализация FCIP позволяет объединить взаимосвязанные системы коммутации в единую систему. Зачастую связь через системы коммутации требуется только небольшому набору узлов с обоих концов. Поэтому в настоящее время в основном в реализациях FCIP используются для создания туннеля такие специфические коммутационные функции, как маршрутизация между виртуальными сетями хранения данных — Inter-VSAN Routing (IVR) или служба маршрутизации FC — Fibre Channel Routing Services (FCRS). Таким образом трафик может маршрутизироваться между конкретными узлами без фактического объединения систем коммутации.

В этой главе подробно рассматриваются протоколы, компоненты и топологии iSCSI и FCIP. В ней также рассматривается набирающий популярность протокол Fibre Channel over Ethernet (FCoE). Этот протокол сводит Ethernet- и FC-трафик в единую физическую связь, благодаря чему устраняет сложность управления двумя отдельными сетями в дата-центре.

6.1. Протокол iSCSI

iSCSI — это протокол на основе IP, устанавливающий соединения между хостами и хранилищами данных по IP и управляющий ими (рис. 6.1). Согласно протоколу iSCSI, SCSI-команды и данные инкапсулируются в IP-пакет и переносятся с использованием протокола TCP/IP. Технология iSCSI широко применяется для подключения серверов к хранилищу, поскольку она относительно недорога и проста в реализации, особенно в условиях отсутствия FC SAN-сетей.

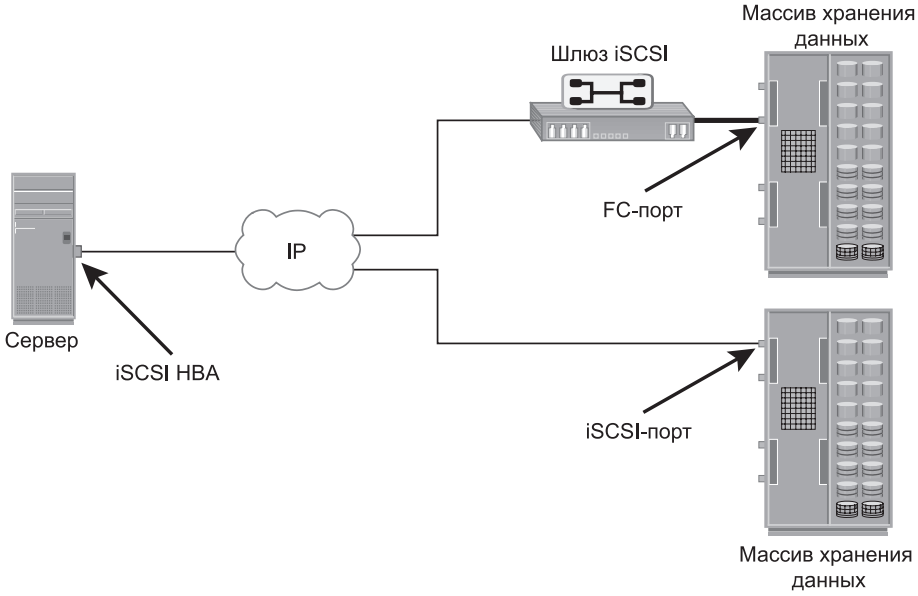


Рис. 6.1. Реализация iSCSI

6.1.1. Компоненты iSCSI

Основные компоненты iSCSI — это инициатор (хост), целевые устройства (хранилище или iSCSI-шлюз) и IP-сеть. Если развернут массив хранения данных, способный работать по протоколу iSCSI, то инициатор в виде хоста с iSCSI может осуществлять обмен данными с массивом хранения данных напрямую по IP-сети. Но в реализации, где для связи по протоколу iSCSI применяется существующий FC-массив, используется iSCSI-шлюз. Эти устройства выполняют преобразование IP-пакетов в FC-кадры и обратно, создавая таким образом мост, позволяющий связать среды IP и FC.

6.1.2. Варианты подключения iSCSI хоста

Существуют три варианта подключения по iSCSI: стандартный сетевой адаптер (NIC) с программным iSCSI-инициатором, встроенный NIC-адаптер с механизмом разгрузки центрального процессора — TCP offload engine (TOE) с программным iSCSI-инициатором и адаптер главной шины (HBA) iSCSI.

Самым простым и дешевым вариантом подключения является стандартный сетевой адаптер (NIC). Его легко задействовать, так как большинство серверов поставляются по крайней мере с одним, а во многих случаях и с двумя встроенными сетевыми адаптерами. Он требует всего лишь наличия программного инициатора для функций iSCSI. Поскольку NIC-адаптеры предоставляют только стандартную IP-функцию, инкапсуляция SCSI в IP-пакеты и декапсуляция выполняются центральным процессором хоста. Таким образом, центральный процессор хоста испытывает дополнительные нагрузки. Если стандартный NIC-адаптер используется во время большой загруженности при вводе-выводе, то использование центрального процессора хоста может стать узким местом в работе всей системы. TOE NIC помогает облегчить это бремя, забирая у хоста функции TCP-управления и оставляя за процессором хоста только функции, связанные с выполнением протокола iSCSI. Хост передает информацию iSCSI TOE-карте, которая отправляет информацию в пункт назначения, используя TCP/IP. Хотя это решение дает возможность повысить производительность, iSCSI-функции по-прежнему обрабатываются программным инициатором, потребляя ресурсы центрального процессора хоста.

Адаптер главной шины iSCSI (iSCSI HBA) может обеспечить выигрыш в производительности, так как он забирает у центрального процессора хоста всю обработку стека протоколов iSCSI и TCP/IP. Использование iSCSI HBA является также простейшим способом обеспечения начальной загрузки хоста из среды SAN через iSCSI. При отсутствии iSCSI HBA для начальной загрузки хоста с устройств хранения данных в основную операционную систему нужно вносить изменения, поскольку до загрузки операционной системы NIC-адаптер нуждается в получении IP-адреса. По функциональному назначению iSCSI HBA похож на FC HBA.

6.1.3. Топологии подключений iSCSI

Топологии, используемые для реализации iSCSI, можно разделить на два класса: непосредственные и мостовые.

Топологии непосредственного подключения не имеют FC-компонентов. Инициаторы могут быть подключены к исполнителям либо напрямую, либо через стандартную IP-сеть. Мостовая топология допускает сосуществование FC и IP, предоставляя функциональные возможности моста от iSCS к FC. Например, инициаторы могут существовать в IP-среде, в то время как хранилища данных будут оставаться в среде FC.

Непосредственное iSCSI-подключение

Если развернут массив, способный работать по протоколу iSCSI, FC-компоненты для связи по iSCSI-протоколу не нужны. На рис. 6.2, *а*, у массива имеется один или несколько iSCSI-портов, сконфигурированные с IP-адресом и подключенные к стандартному Ethernet-коммутатору. После регистрации инициатора сети он может обращаться к доступным LUN-устройствам в массиве хранения данных. Один порт массива может обслуживать несколько хостов или инициаторов до тех пор, пока массив способен обрабатывать объем трафика сохраняемых данных, генерируемый этими хостами.

Мостовое iSCSI-подключение

Реализация мостового iSCSI-подключения включает в свою конфигурацию компоненты оптоволоконного канала. Подключение iSCSI-хоста к FC-массиву хранения данных показано на рис. 6.2, *б*.

В данном случае у массива нет iSCSI-портов. Поэтому для соединения iSCSI-хоста и FC-хранилища необходимо использовать внешнее устройство, называемое шлюзом или многопротокольным маршрутизатором. Шлюз преобразует IP-пакеты в FC-кадры и наоборот. Для содействия обмену данными между FC- и Ethernet-средами мостовые устройства содержат как FC-, так и Ethernet-порты.

В мостовой iSCSI-реализации iSCSI-инициатор конфигурируется с IP-адресом шлюза как его целевым адресатом. С другой стороны, шлюз конфигурируется в качестве FC-инициатора по отношению к массиву хранения данных.

Сочетание FC- и непосредственного iSCSI-подключения

Наибольшее распространение получила топология, в которой сочетаются FC- и непосредственное iSCSI-подключение. Обычно массив хранения данных поставляется как с портами FC, так и с портами iSCSI, что позволяет создавать iSCSI- и FC-подключения в одной и той же среде (см. рис. 6.2, *в*).

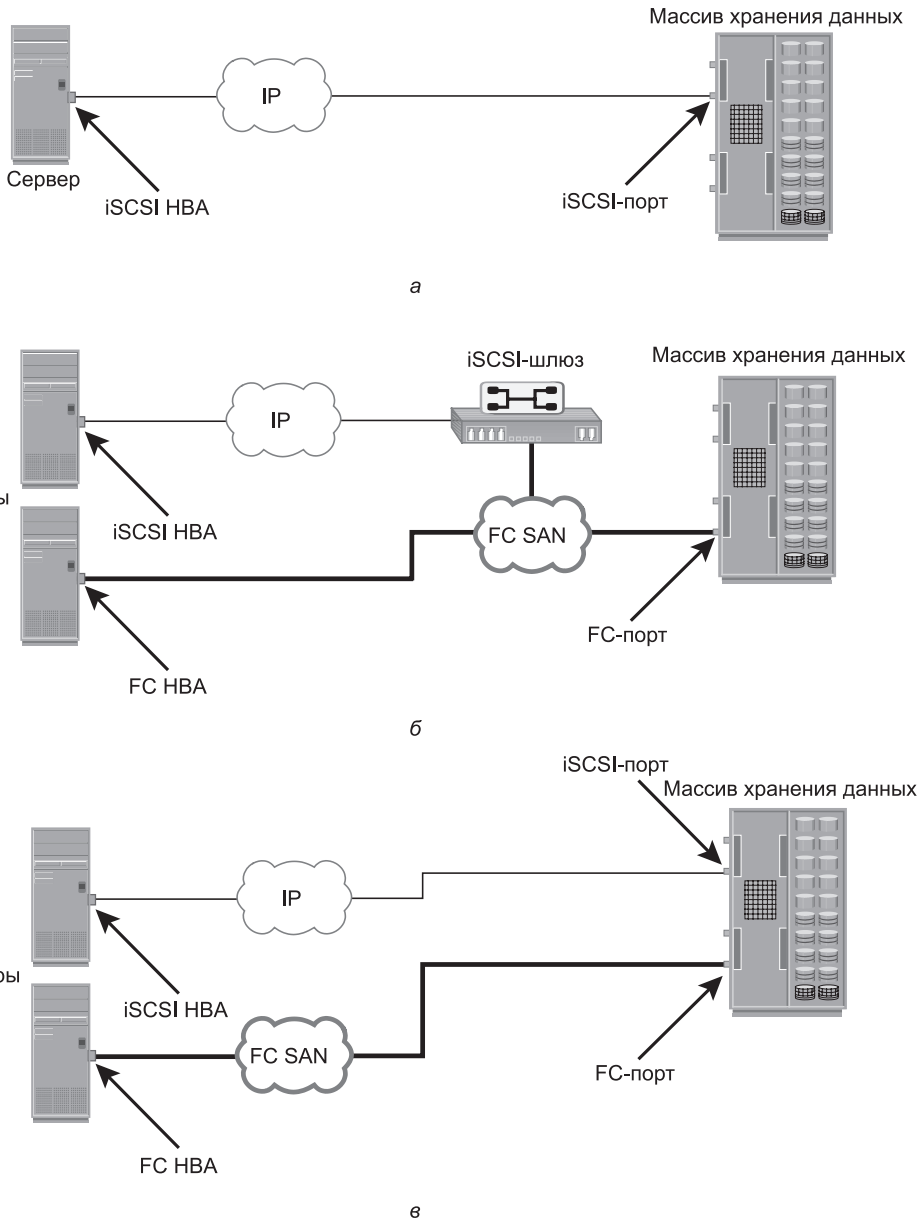


Рис. 6.2. Топологии iSCSI: *а* — непосредственное iSCSI-подключение; *б* — мостовое iSCSI-подключение; *в* — сочетание FC- и непосредственного iSCSI-подключения

6.1.4. Стек протоколов iSCSI

На рис. 6.3 показаны модель протокольных уровней iSCSI и порядок инкапсуляции SCSI-команд для их доставки по физическому носителю.

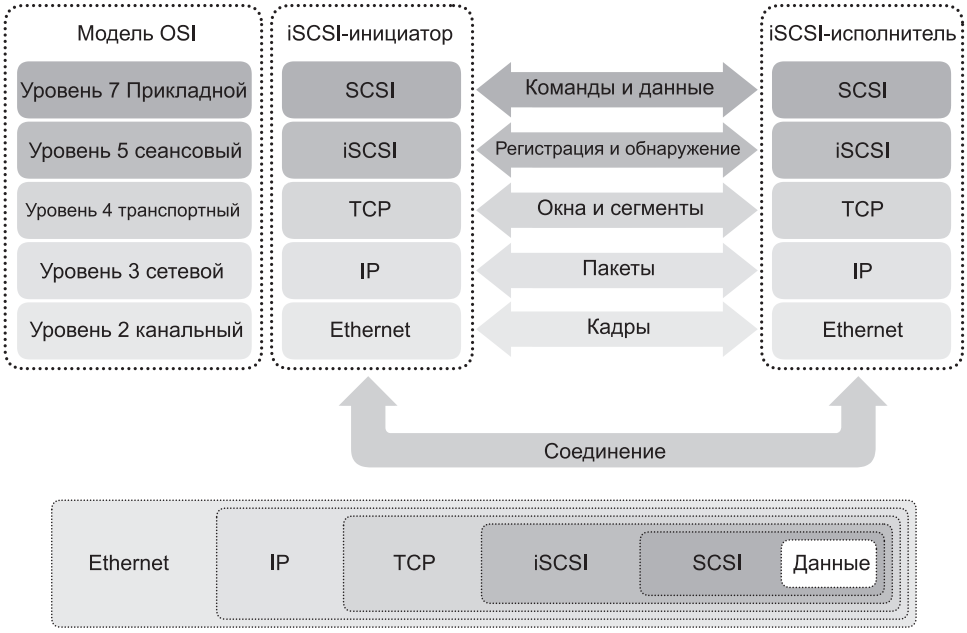


Рис. 6.3. Стек протоколов iSCSI

SCSI — это командный протокол, работающий на прикладном уровне модели взаимодействия открытых систем — Open System Interconnection (OSI). Для общения друг с другом инициаторы и исполнители применяют команды и ответы SCSI. Дескрипторы блоков команд SCSI, данные и сообщения о состоянии инкапсулируются в TCP/IP-пакеты и передаются по сети между инициаторами и исполнителями.

iSCSI — это протокол сеансового уровня, инициирующий надежный сеанс связи между устройствами, распознающими команды SCSI и TCP/IP. Интерфейс сеансового уровня iSCSI отвечает за обработку регистрации и аутентификации, обнаружение исполнителя и управление сеансом. TCP используется с iSCSI в качестве транспортного уровня для обеспечения надежной передачи данных.

TCP используется для управления потоком сообщений, составления окон и сегментов, устранения ошибок и повторной передачи данных. Обеспечение глобальной адресации и подключений осуществляется на сетевом уровне модели OSI. Имеющиеся в этой модели протоколы уровня 2, относящиеся к уровню канала передачи данных, допускают обмен данными между узлами по физической сети.

6.1.5. Протокольные блоки данных iSCSI

Протокольные блоки данных — protocol data unit (PDU) являются основной «информационной единицей» в iSCSI-среде. Инициаторы и исполнители iSCSI общаются с помощью PDU-блоков iSCSI. Такая связь включает в себя установление iSCSI-подключений и iSCSI-сеансов, выполнение iSCSI-обнаружения, отправку SCSI-команд и данных и получение SCSI-состояния. iSCSI PDU-блоки содержат один или несколько сегментов заголовков, за которыми либо не следуют никакие данные, либо следует один или несколько сегментов данных. Затем для упрощения транспортировки PDU-блок инкапсулируется в IP-пакет.

PDU-блок содержит компоненты, показанные на рис. 6.4. IP-заголовок дает информацию для маршрутизации пакета, использующуюся для перемещения пакета по сети. TCP-заголовок содержит информацию, необходимую для гарантированной доставки пакета исполнителю. iSCSI-заголовок (основной заголовочный сегмент) описывает исполнителю процесс извлечения команд и данных SCSI. Для обеспечения целостности датаграммы iSCSI добавляет контрольную сумму (CRC), известную как дайджест. Она идет в дополнение к контрольным суммам TCP и Ethernet. Дайджесты заголовка и данных используются в PDU-блоке дополнительно для проверки целостности и размещения данных.

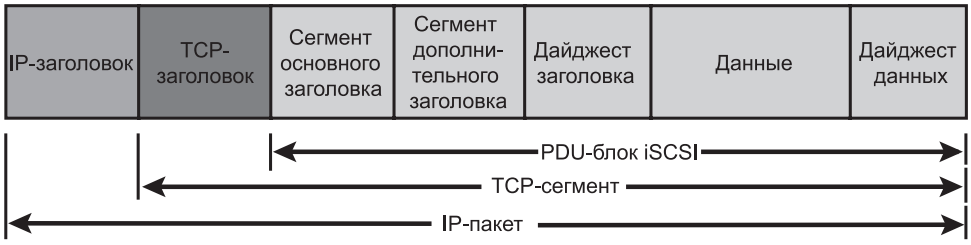
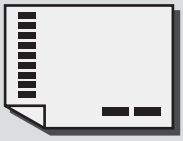


Рис. 6.4. PDU-блок iSCSI, инкапсулированный в пакет IP



Передаваемое по сети сообщение делится на совокупность пакетов. При необходимости каждый отдельно взятый пакет может быть отправлен по сети другим маршрутом. Пакеты могут прибывать в порядке, отличающемся от исходного. По IP-протоколу осуществляется только их доставка. Выстроить пакеты в нужной последовательности — задача TCP. SCSI-команды и данные исполнитель извлекает на основе информации, находящейся в заголовке iSCSI.

Как показано на рис. 6.5, соотношения один к одному между каждым отдельно взятым PDU-блоком iSCSI и IP-пакетом не соблюдается.

В зависимости от своего размера PDU-блок iSCSI может как распространяться на весь IP-пакет, так и находиться в нем вместе с другим PDU-блоком iSCSI. Чтобы получить соотношение один к одному между IP-пакетами и iSCSI PDU-блоками, изменяется максимальный размер передаваемого блока данных — maximum transmission unit (MTU) IP-пакета. Тем самым исключается фрагментация IP-пакета, что повышает эффективность передачи данных.

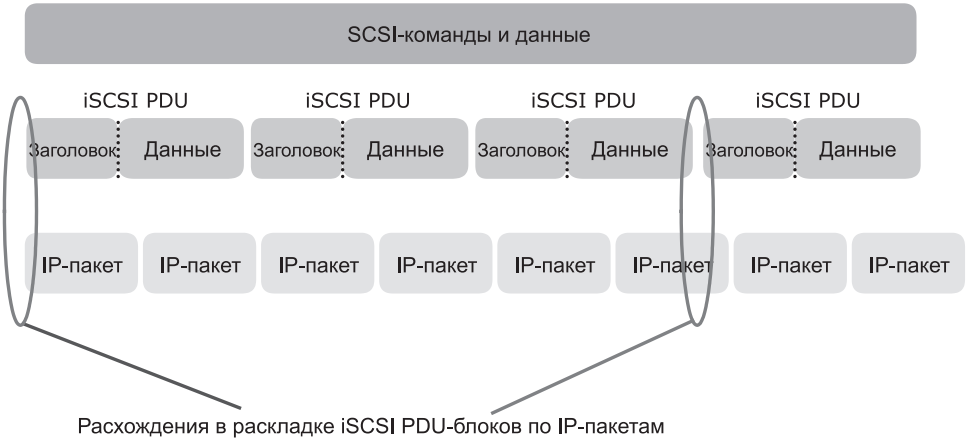


Рис. 6.5. Раскладка PDU-блоков iSCSI по IP-пакетам

6.1.6. Обнаружение устройств в iSCSI

Прежде чем установить сеанс связи, инициатор должен обнаружить местонахождение своих исполнителей в сети и узнать имена доступных ему исполнителей. Это можно сделать двумя способами: с помощью метода обнаружения SendTargets и с использованием службы имен хранилищ — internet Storage Name Service (iSNS).

При использовании метода SendTargets для установки сеанса обнаружения инициатору вручную задается сетевой портал-исполнитель. Инициатор подает команду SendTargets, и сетевой портал-исполнитель отправляет в ответ имена и адреса доступных хосту исполнителей.

Служба iSNS (рис. 6.6) позволяет автоматически обнаруживать iSCSI-устройства в IP-сети. Инициаторы и исполнители можно настроить на автоматическую регистрацию на iSNS-сервере. Когда инициатору потребуется узнать, к каким исполнителям он может получить доступ, он может запросить у iSNS-сервера список доступных исполнителей.

Обнаружение может быть осуществлено также с помощью протокола поиска службы — Service Location Protocol (SLP). Однако этот метод применяется реже, чем обнаружение с помощью SendTargets и iSNS.

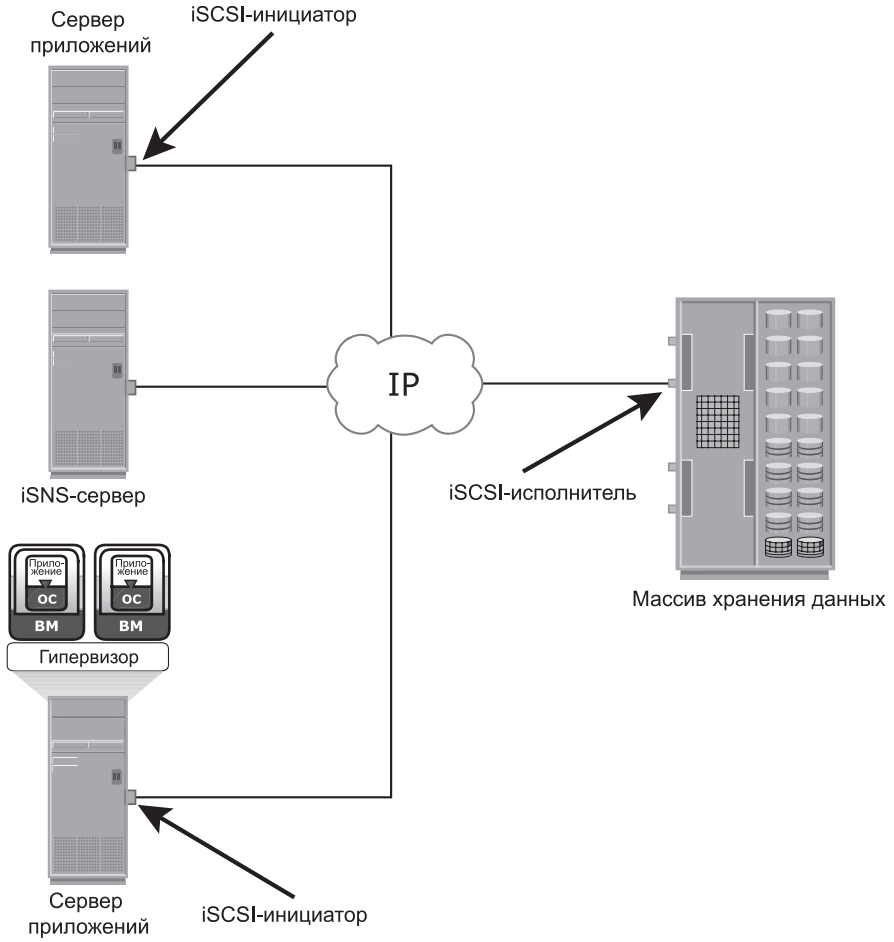


Рис. 6.6. Обнаружение с использованием iSNS

6.1.7. iSCSI-имена

С целью упрощения процесса передачи информации для идентификации инициаторов и исполнителей в iSCSI-сети используется уникальный глобальный iSCSI-идентификатор, известный как имя iSCSI. Уникальный идентификатор может быть сочетанием названия отдела, приложения или производителя, серийного номера, номера ресурса или любой другой маркировкой, которая используется для распознавания устройств и управления ими.

В большинстве случаев используются два типа iSCSI-имен:

- **полное iSCSI-имя — iSCSI Qualified Name (IQN).** Чтобы генерировать полные iSCSI-имена, в собственности организации должно быть зарегистрированное доменное имя. При этом активность доменного имени

или возможность его разрешения в адрес обязательными условиями не являются. Нужно лишь, чтобы это имя было зарезервировано и другие организации не могли воспользоваться для создания iSCSI-имен точно таким же доменным именем. Во избежание возможных конфликтов, вызванных пересылкой доменных имен, в имя включается дата. Пример полного имени типа IQN выглядит следующим образом: iqn.2008-02.com.example:optional_string.

Часть optional_string содержит серийный номер, номер ресурса или любой другой идентификатор устройства. Полное iSCSI-имя позволяет администраторам хранилищ данных присваивать iSCSI-устройствам содержательные имена, упрощая тем самым управление этими устройствами;

- **расширенный уникальный идентификатор — Extended Unique Identifier (EUI).** Это глобальный уникальный идентификатор на основе стандарта задания имен EUI-64 IEEE, который включает префикс eui, после которого следует 16-разрядное шестнадцатеричное число, задающее имя, например eui.0300732A32598D26.

В любом из этих двух форматов iSCSI-имен разрешенными специальными символами являются точки, тире и пробелы.

NETWORK ADDRESS AUTHORITY

Network Address Authority (NAA) является дополнительным типом имени iSCSI-узла, допускающим применение глобального формата задания имен, определяемого стандартом T11, принятым Международным комитетом стандартов информационных технологий — InterNational Committee for Information Technology Standards (INCITS). Это формат позволяет SCSI-устройствам хранения данных, имеющим как iSCSI-порты, так и SAS-порты, использовать одно и то же имя SCSI-устройства на основе NAA. Это формат определяется стандартом RFC 980, «T11 Network Address Authority (NAA) Naming Format for iSCSI Node Names».

6.1.8. Сеанс связи iSCSI

Сеанс связи iSCSI устанавливается между инициатором и исполнителем (рис. 6.7). Сеанс опознается с помощью идентификатора сеанса (SSID), включающего в себя идентификатор инициатора (ISID) и идентификатор исполнителя (TSID). Сеанс может предназначаться для одного из следующих действий:

- обнаружения доступных инициатору исполнителей и определения местонахождения конкретного исполнителя в сети;
- проведения обычной операции iSCSI (передачи данных между инициаторами и исполнителями).

Сеанс связи iSCSI устанавливается через процесс iSCSI-регистрации. Этот процесс начинается, когда инициатор устанавливает TCP-соединение с заданным исполнителем либо через общеизвестный порт 3260, либо через

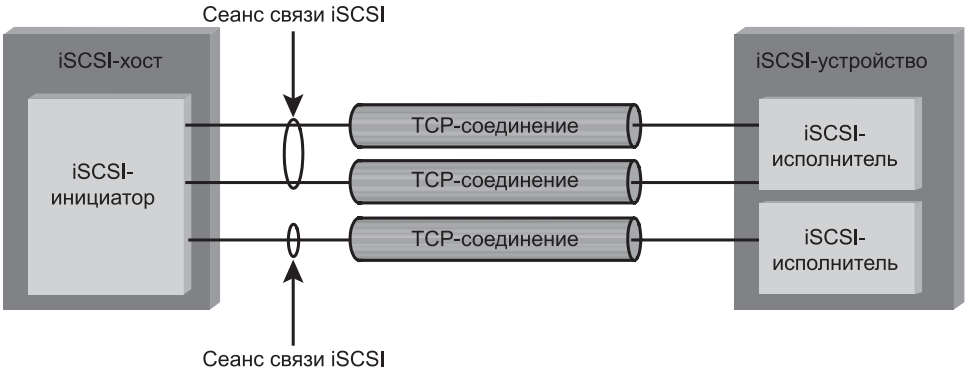


Рис. 6.7. Сеанс связи iSCSI

указанный порт исполнителя. На этапе регистрации инициатор и исполнитель проводят взаимную аутентификацию и согласуют различные параметры.

После успешного завершения этапа регистрации сеанс связи iSCSI входит в этап полноценной работы для осуществления обычных SCSI-транзакций. На этом этапе инициатор может отправлять команды и данные различным LUN-устройствам исполнителя, инкапсулируя их в PDU-блоках iSCSI, перемещаемых по установленному TCP-соединению.

Завершающим для сеанса связи iSCSI является этап разрыва соединения, который называется процедурой окончания сеанса. За начало процедуры окончания сеанса отвечает инициатор, но исполнитель также может побудить к разрыву соединения, отправив iSCSI-сообщение о возникновении условия внутренней ошибки. После того как запрос на окончание сеанса отправлен инициатором и принят исполнителем, по соединению не могут больше отправляться никакие запросы и ответы.

6.1.9. Выстраивание командных последовательностей iSCSI

Связь по протоколу iSCSI между инициаторами и исполнителями основывается на командных последовательностях типа «запрос — ответ». Из командной последовательности может быть создано несколько PDU-блоков. Для нумерации всех PDU-блоков, составляющих команду от инициатора к исполнителю, в ходе сеанса используется номер последовательности команд — command sequence number (CmdSN). Этот номер используется для того, чтобы обеспечить доставку каждой команды в том же порядке, в котором она была передана, независимо от TCP-соединения, переносящего команду в сеансе.

Командная последовательность начинается с первой команды регистраци, и с каждой последующей командой CmdSN увеличивается на единицу. За доставку команд SCSI-уровню по порядку их номеров CmdSN отвечает уровень iSCSI-исполнителя. Тем самым обеспечивается нужный порядок данных и команд, поступающих к исполнителю, даже при наличии между инициатором и исполнителем, использующим группы порталов, нескольких TCP-соединений.

По аналогии с нумерацией команд, для последовательной нумерации ответов о состоянии соединения используется номер последовательности состояния — status sequence number (StatSN) (рис. 6.8). Данные уникальные номера устанавливаются на уровне TCP-соединения.

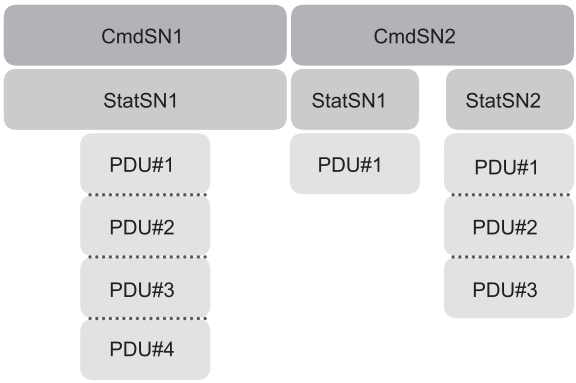


Рис. 6.8. Нумерация последовательности команд и состояний

Когда инициатор готов к приему данных, исполнитель отправляет ему PDU-блоки запроса на передачу — request-to-transfer (R2T). Для обеспечения доставки данных в составе одной и той же команды в нужном порядке используется номер последовательности данных — data sequence number (DataSN). Для выстраивания последовательности PDU-блоков данных и R2T-запросов используются DataSN и R2TSN соответственно. Каждый из этих порядковых номеров хранится локально как беззнаковый 32-разрядный целочисленный счетчик, определяемый iSCSI. Эти номера передаются между инициатором и исполнителем в соответствующих полях PDU-блоков iSCSI в ходе обмена командами, состояниями и данными.

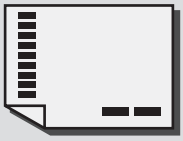
Для операций чтения номер DataSN начинается с нуля и увеличивается на единицу для каждого последующего PDU-блока в последовательности команд. Для операции записи первый предоставляемый без запроса PDU-блок данных или первый PDU-блок данных, предоставляемый в ответ на R2T-запрос начинается с нулевого номера DataSN и увеличивается на единицу для каждого последующего PDU-блока. Порядковый номер запроса на передачу (R2TSN) устанавливается равным нулю при инициации команды и увеличивается на единицу для каждого последующего запроса на передачу, отправляемого исполнителем для этой команды.

6.2. FCIP

FC SAN предоставляет высокопроизводительную инфраструктуру для локального перемещения данных. В настоящее время организации ищут новые пути передачи данных на большие расстояния между своими

разнородными территориально разнесенными SAN-сетями. Одним из лучших способов решения этой задачи является соединение территориально разнесенных SAN-сетей посредством надежных и высокоскоростных каналов связи. Данный подход предполагает транспортировку блоков данных FC по IP-инфраструктуре. Туннельным протоколом, позволяющим соединять разбросанные островки FC SAN-сетей по существующим сетям на основе IP-протокола, является FCIP.

Стандарт FCIP быстро завоевал признание как легкий в управлении и экономически эффективный способ совмещения всего лучшего из обеих областей: FC SAN-сетей и проверенной широко используемой IP-инфраструктуры. В результате его внедрения организации теперь располагают более эффективным способом защиты, хранения и перемещения своих данных, максимально используя при этом вложения в свою уже существующую IP-инфраструктуру. FCIP широко применяется в системах аварийного восстановления, данные для которых дублируются в хранилищах, расположенных в удаленном месте.



При репликации данных или их резервном копировании для FCIP может потребоваться высокая пропускная способность сети. FCIP не занимается регулированием трафика данных или управлением потоком, эти задачи берут на себя связанные друг с другом FC-коммутаторы и устройства в системе коммутации.

6.2.1. Стек протоколов FCIP

Стек протоколов FCIP показан на рис. 6.9. Приложения генерируют SCSI-команды и данные, обрабатываемые на разных уровнях стека протоколов.

Рис. 6.9. Стек протоколов FCIP

Протокол верхнего SCSI-уровня включает программу SCSI-драйвера, исполняющую команды чтения-записи. Ниже SCSI-уровня находится уровень FC-протокола (FCP), являющийся просто FC-кадром, чьей полезной нагрузкой служит SCSI. FCP-уровень является надстройкой над транспортным уровнем Fibre Channel. Это позволяет FC-кадрам перемещаться в среде системы коммутации SAN-сети естественным образом. Кроме того, FC-кадры могут быть инкапсулированы в IP-пакет и отправлены удаленной SAN-сети по IP. На FCIP-уровне кадры Fibre Channel инкапсулируются в полезную нагрузку IP и передаются на TCP-уровень (см. рис. 6.10). TCP и IP используются для транспортировки инкапсулированной информации по Ethernet, беспроводному или иному носителю, поддерживающему TCP/IP-трафик.

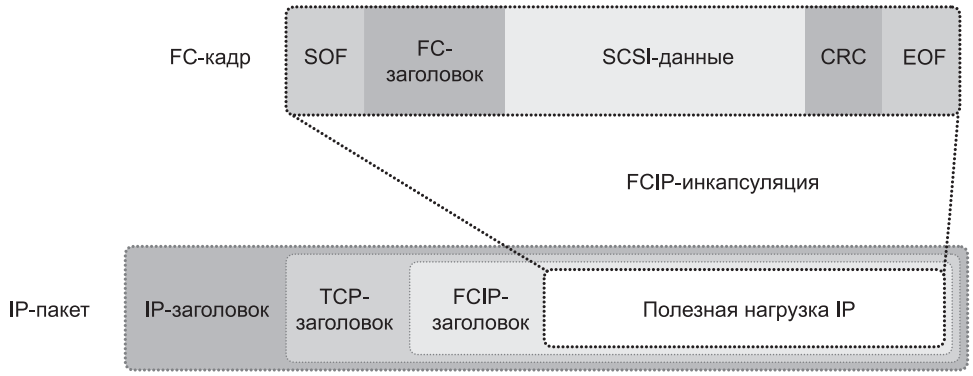


Рис. 6.10. FCIP-инкапсуляция

Когда канал данных не может поддерживать максимальный размер передаваемого блока (MTU) IP-пакета, инкапсуляция FC-кадра в IP-пакет может привести к фрагментации IP-пакета. Когда IP-пакет подвергается фрагментации, необходимые части заголовка должны быть скопированы во все фрагменты. При сегментировании TCP-пакета за получение и выстраивание данных перед их отправкой той части устройства, которая занимается FC-обработкой, отвечают обычные TCP-операции.

6.2.2. Топология FCIP

В FCIP-среде FCIP-шлюз подключен к каждой системе коммутации через стандартное FC-соединение (рис. 6.11). FCIP-шлюз с одного конца IP-сети инкапсулирует FC-кадры в IP-пакеты. Шлюз с другого конца удаляет IP-оболочку и отправляет FC-данные на уровень 2 системы коммутации. Система коммутации рассматривает такие шлюзы как коммутаторы системы уровня 2. Порту шлюза, подключенному к IP-сети, назначается IP-адрес. После установления IP-соединения узлы в двух независимых друг от друга системах коммутации могут обмениваться данными.

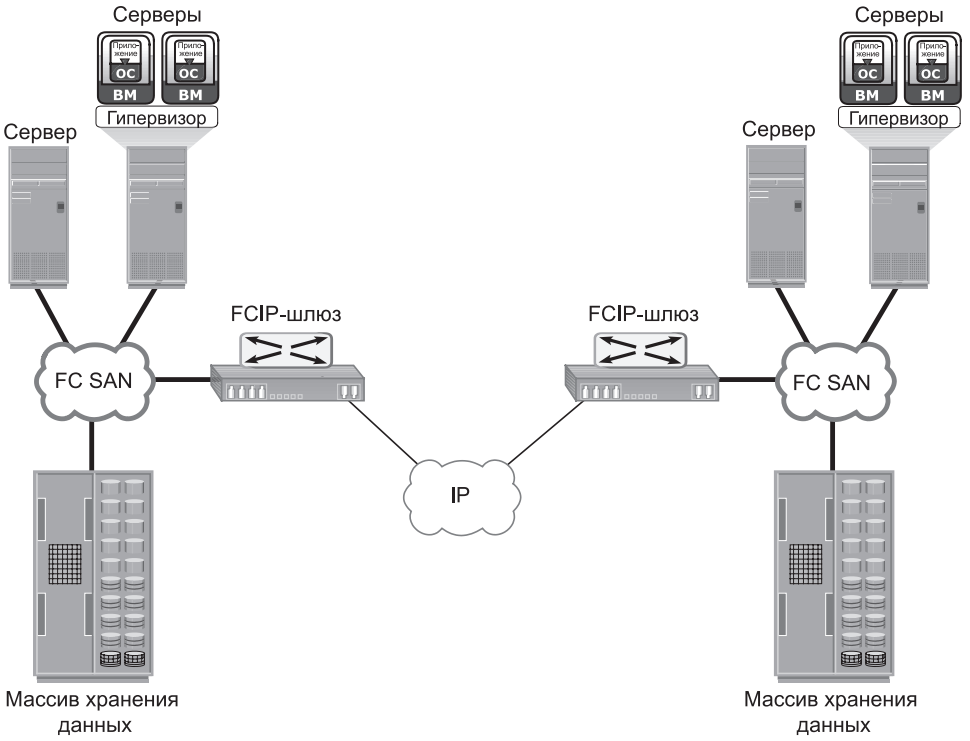


Рис. 6.11. Топология FCIP

6.2.3. Производительность и безопасность FCIP

При реализации решений, касающихся хранения данных, всегда нужно принимать во внимание показатели производительности, надежности и безопасности. То же самое относится и к реализации FCIP.

С точки зрения производительности настройка нескольких маршрутов между FCIP-шлюзами позволяет избавиться от единых точек отказов и обеспечивает повышенную пропускную способность. Если при удлинении каналов не будет обеспечена достаточная пропускная способность, IP-сеть может стать самым узким местом всей системы. Кроме того, поскольку FCIP создает объединенную систему коммутации, сбой в основной IP-сети может привести к нестабильной работе во всей SAN-сети. Такая нестабильность может проявляться в сегментации системы коммутации, слишком большом количестве уведомлений об изменениях состояний, поступающих в адрес устройств, зарегистрированных в RSCN-службе, и истечении времени ожидания события хостом.

Производители FC-коммутаторов признали недостатки, присущие FCIP, и разработали ряд мер по повышению стабильности, например добавили возможность выделять FCIP-трафик в отдельную виртуальную систему коммутации.

При внедрении FCIP-решения нужно также уделять внимание вопросам безопасности, поскольку данные передаются по общедоступным IP-каналам. Ряд вариантов обеспечения безопасности реализуется на основе поддержки этих вариантов со стороны маршрутизаторов. Одной из таких мер безопасности, реализуемой в FCIP-среде, является применение набора протоколов IPSec.

6.3. FCoE

Для работы с различными типами трафика ввода-вывода в дата-центрах обычно имеются несколько сетей, например Ethernet-сеть для обмена данными по протоколу TCP/IP и FC-сеть для связи по протоколу FC. TCP/IP-сети обычно используются для обмена данными по схеме «клиент — сервер», создания резервных копий данных, связи при управлении инфраструктурой и т. д. FC-сети обычно используются для перемещения данных на уровне блоков между хранилищем и серверами. Для поддержки нескольких сетей серверы в дата-центре оборудуются с избытком несколькими физическими сетевыми интерфейсами, например несколькими Ethernet- и FC-картами или адаптерами. Кроме того, для обеспечения обмена данными дата-центры комплектуются различными типами сетевых коммутаторов и физической кабельной инфраструктурой. Необходимость содержания двух разных типов физической сетевой инфраструктуры увеличивает общие затраты и усложняет работу дата-центра.

Объединить LAN- и SAN-трафик в единую инфраструктуру физического интерфейса позволяет протокол передачи данных в формате FC по сети Ethernet — Fibre Channel over Ethernet (FCoE). FCoE помогает организациям решать проблемы уже имеющихся в их распоряжении нескольких отдельных сетевых инфраструктур. Для отправки FC-кадров по Ethernet в FCoE используется связь по протоколу конвергентного улучшенного Ethernet — Converged Enhanced Ethernet (CEE) (10-гигабитный Ethernet).

6.3.1. Консолидация ввода-вывода с помощью FCoE

Основным преимуществом FCoE является консолидация ввода-вывода. На рис. 6.12 представлена инфраструктура до развертывания FCoE. В ней доступ к ресурсам хранения данных осуществляется с использованием HBA-адаптеров, а доступ к ресурсам IP-сети — с использованием серверами NIC-карт. Как правило, в дата-центрах у сервера имеется от 2 до 4 NIC-карт и избыточное количество HBA-карт. Если у дата-центра сотни серверов, понадобится большое количество адаптеров, кабелей и коммутаторов. Это приведет к созданию сложной, трудно управляемой и трудно расширяемой среды. Добавят проблем также расходы на электроэнергию, охлаждение аппаратуры и оплату занимаемых площадей.

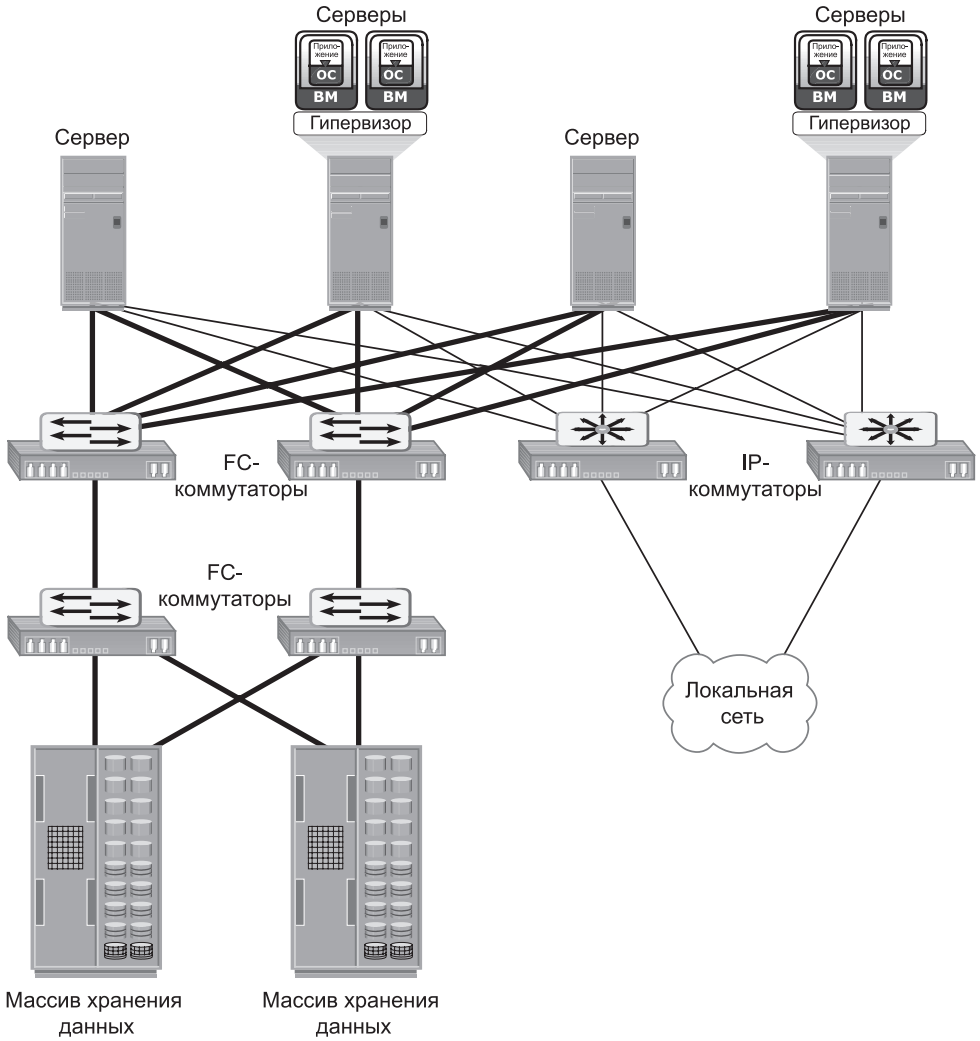


Рис. 6.12. Инфраструктура до использования

На рис. 6.13 показана консолидация ввода-вывода с помощью FCoE, при которой используются FCoE-коммутаторы и конвергентные сетевые адаптеры — Converged Network Adapters (CNA). CNA-адаптеры (рассматриваемые в разделе «Конвергентные сетевые адаптеры») заменяют в сервере как HBA-адаптеры, так и NIC-карты и консолидируют как IP-, так и FC-трафик. При этом серверу для подключения к различным сетям несколько сетевых адаптеров уже не нужны. В целом уменьшаются потребности в адаптерах, кабелях и коммутаторах. К тому же существенно сокращаются расходы на закупку оборудования и управление.

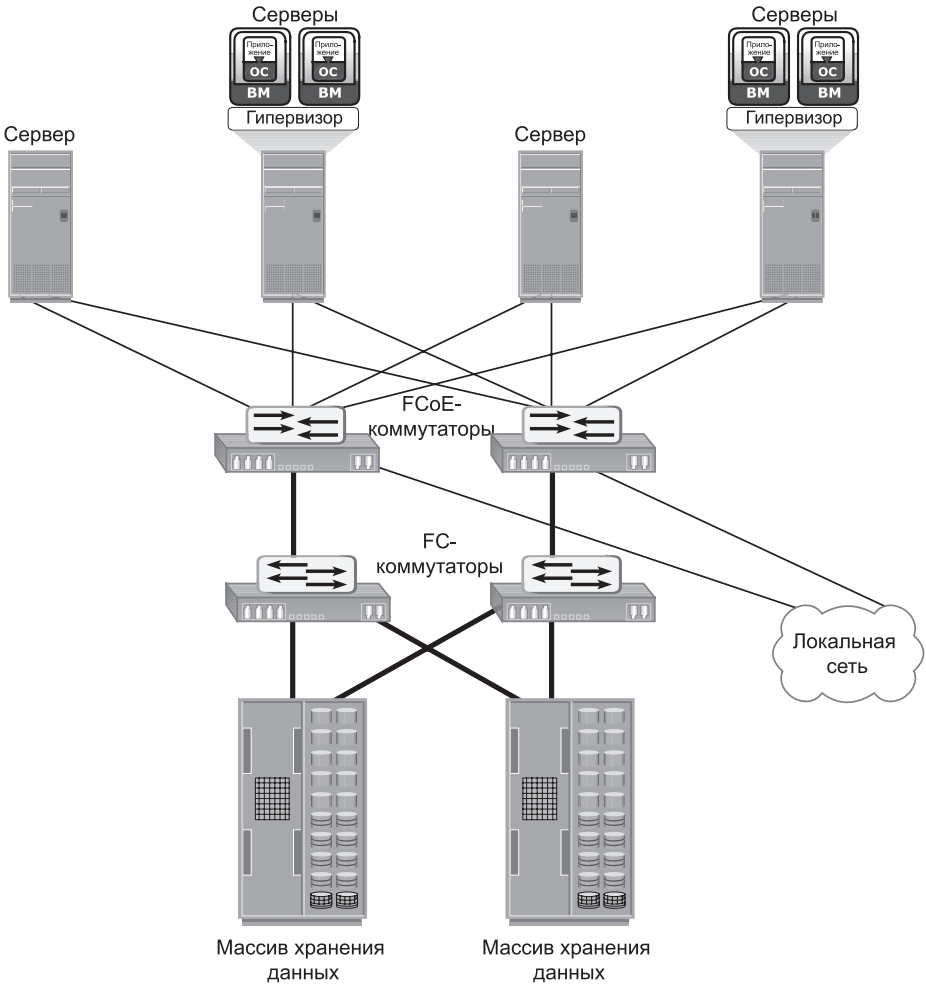


Рис. 6.13. Инфраструктура после внедрения FCoE

6.3.2. Компоненты FCoE-сети

В данном разделе рассматриваются основные физические компоненты, необходимые для реализации FCoE в дата-центре. К их числу относятся:

- конвергентный сетевой адаптер — Converged Network Adapter (CNA);
- кабели;
- FCoE-коммутаторы.

Конвергентный сетевой адаптер

CNA-адаптер предоставляет в одном адаптере функциональные возможности как стандартного NIC-адаптера, так и FC HBA-адаптера и консолидирует трафик обоих типов. CNA устраняет потребности в развертывании

отдельных адаптеров и кабелей для FC- и Ethernet-коммуникаций, сокращая тем самым количество серверных разъемов и портов коммутаторов. CNA разгружает сервер от выполнения задачи обработки FCoE-протокола, высвобождая тем самым на сервере ресурсы центрального процессора для решения прикладных задач. Как показано на рис. 6.14, в CNA содержатся отдельные модули специализированных интегральных схем (Application Specific Integrated Circuits, ASIC) для 10-гигабитного Ethernet (10GbE ASIC), Fibre Channel (FC ASIC) и FCoE (FCoE ASIC). В FCoE ASIC FC-кадры инкапсулируются в Ethernet-кадры. Для обеспечения возможности подключения сервера одна сторона ASIC подключена к 10-гигабитному Ethernet и специализированным интегральным схемам FC, а другая ее сторона обеспечивает подключение интерфейса 10-гигабитного Ethernet к FCoE-коммутатору.

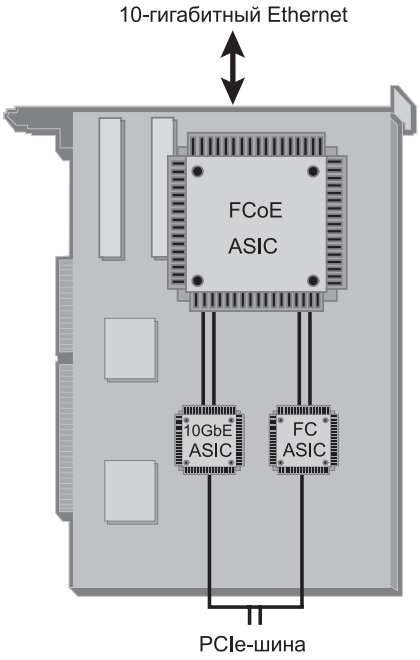
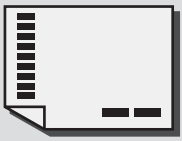


Рис. 6.14. Конвергентный сетевой адаптер

Кабели

В настоящее время доступны два варианта создания кабельных соединений FCoE: с использованием кабеля на медной основе Twinax и стандартных оптоволоконных кабелей. Кабель Twinax состоит из заключенных в экранированную оболочку двух пар медных кабелей. Этот кабель допускает передачу данных со скоростью 10 Гбит/с на расстояния, не превышающие 10 м. Кабели Twinax требуют меньших энергозатрат и стоят дешевле оптоволоконных кабелей. В основном для FCoE-подключений используется разъем типа Small

Form Factor Pluggable Plus (SFP+), который подходит как для оптических, так и для медных кабелей.



При типовом развертывании FCoE-оборудование находится в верхней части каждой серверной стойки, где размещаются два FCoE-коммутатора с избыточным числом компонентов. Как FC-, так и IP-подключение к каждому серверу выполняется с использованием недорогих Twinax-кабелей, идущих от сервера к верхней части стойки с FCoE-коммутаторами. Связь на таких коротких дистанциях хорошо поддерживается с помощью Twinax. Связь коммутаторов, находящихся в верхней части стойки с существующими опорными LAN- и SAN-инфраструктурами, то есть межстоечные соединения, обычно реализуются по оптическим каналам, поддерживающим более протяженные кабельные магистрали, которые могут потребоваться в данном случае.

FCoE-коммутаторы

FCoE-коммутаторы имеют функциональные возможности, присущие обоим коммутаторам, как Ethernet, так и Fibre Channel. Как показано на рис. 6.15, у FCoE-коммутатора есть блок выбора направления передачи Fibre Channel Forwarder (FCF), мост Ethernet Bridge и набор Ethernet-портов и дополнительных FC-портов. Задача FCF заключается в инкапсуляции FC-кадров, полу-

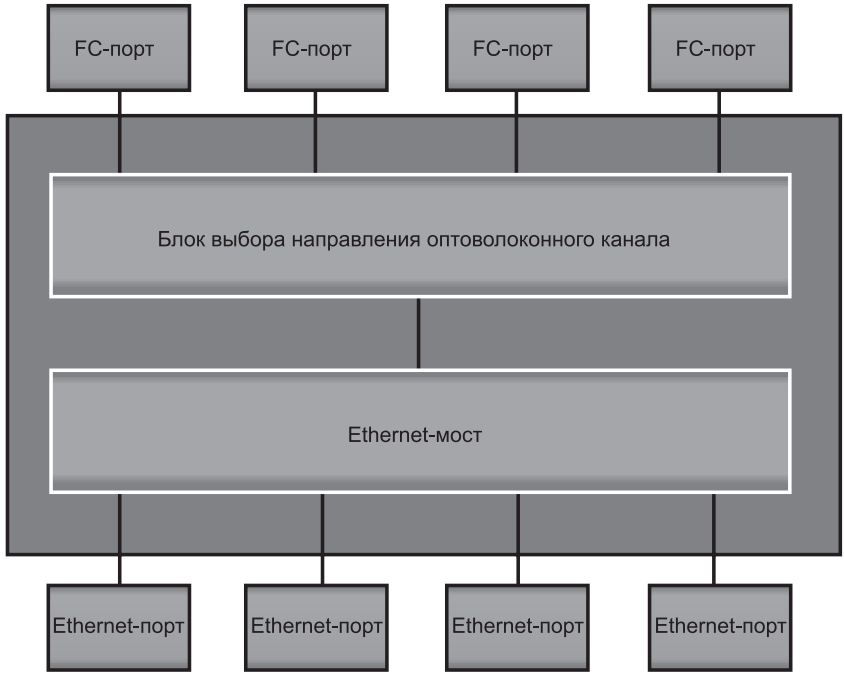


Рис. 6.15. Обобщенная схема архитектуры FCoE-коммутатора

ченных из FC-порта, в FCoE-кадры, а также в деинкапсуляции FCoE-кадров, полученных из моста Ethernet Bridge, в FC-кадры.

До получения входящего трафика FCoE-коммутатор проверяет Ethertype (тип Ethernet, используемый в качестве признака протокола, инкапсулированного в виде полезной нагрузки Ethernet-кадра) входящих кадров и использует его для определения пункта назначения. Если Ethertype кадра имеет значение FCoE, коммутатор понимает, что кадр содержит в качестве полезной нагрузки FC, и направляет его к FCF. Там FC извлекается из FCoE-кадра и передается в FC SAN по FC-портам. Если Ethertype не имеет значения FCoE, коммутатор обрабатывает данные как обычный Ethernet-трафик и направляет его по Ethernet-портам.

6.3.3. Структура кадра FCoE

FCoE-кадр представляет собой Ethernet-кадр, содержащий протокольный блок данных (PDU-блок) FCoE. Структура FCoE-кадра показана на рис. 6.16. Первые 48 разрядов кадра используются для указания MAC-адреса пункта назначения, а следующие 48 разрядов указывают на MAC-адрес источника. 32-разрядный указатель IEEE 802.1Q поддерживает в рамках одной физической инфраструктуры создание нескольких виртуальных сетей (VLAN). У FCoE имеется собственный Ethertype, обозначаемый следующими 16 разрядами, за которыми следует 4-разрядное поле версии. Следующие 100 разрядов зарезервированы, а за ними следуют 8-разрядный признак начала кадра — Start of Frame и сам FC-кадр. После 8-разрядного разделителя конца кадра — End of Frame следуют 24 зарезервированных разряда. Кадр оканчивается завершающими 32 разрядами, предназначенными для реализации функции контрольной последовательности кадров — Frame Check Sequence (FCS), обеспечивающей обнаружение ошибок для Ethernet-кадра.

Инкапсулированный кадр Fibre Channel состоит из исходного 24-разрядного FC-заголовка и передаваемых данных, включающих контрольную сумму Fibre Channel CRC. Структура FC-кадра поддерживается таким образом, что при подключении к коммутатору с FCoE-возможностями обычной FC SAN-сети FC-кадр деинкапсулируется из FCoE-кадра и без каких-либо дополнительных манипуляций перемещается в FC SAN-сеть. Эта возможность позволяет FCoE объединяться с существующими FC SAN-сетями, не испытывая при этом никакой потребности в шлюзе. Большое значение в FCoE имеет также размер кадра. В обычном кадре данных Fibre Channel имеется полезная нагрузка размером 2112 байт, 24-байтовый заголовок и FCS. У стандартного Ethernet-кадра имеется исходная полезная нагрузка объемом 1500 байт. Чтобы получить высокую производительность, в FCoE должны использоваться Jumbo-кадры (очень большие кадры), не допускающие разбиения кадра Fibre Channel на два Ethernet-кадра. Jumbo-кадры подробно рассматриваются в следующей главе. Для технологии FCoE требуется конвергентный улучшенный Ethernet — Converged Enhanced Ethernet, обеспечивающий отсутствие потерь и поддержку Jumbo-кадров.

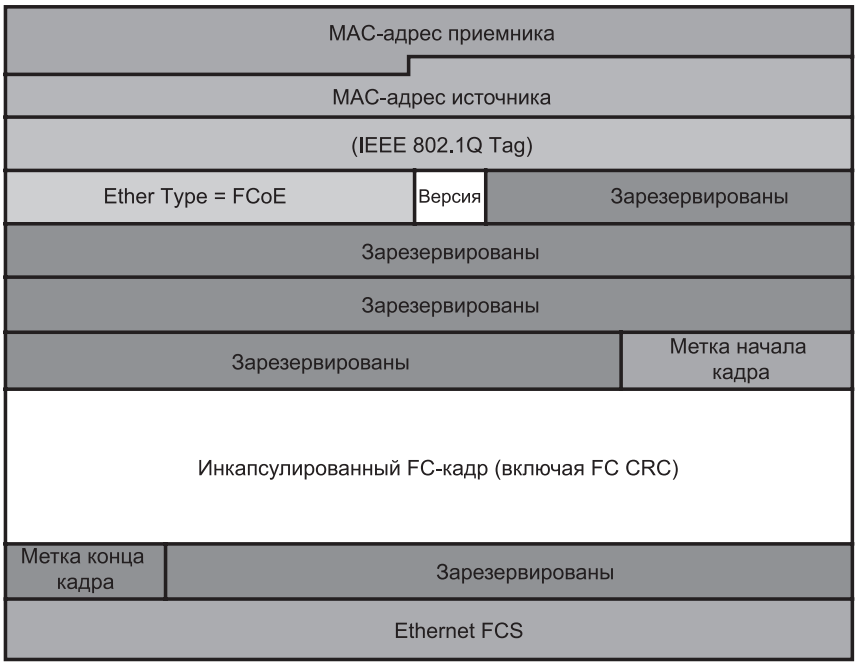


Рис. 6.16. Структура кадра FCoE

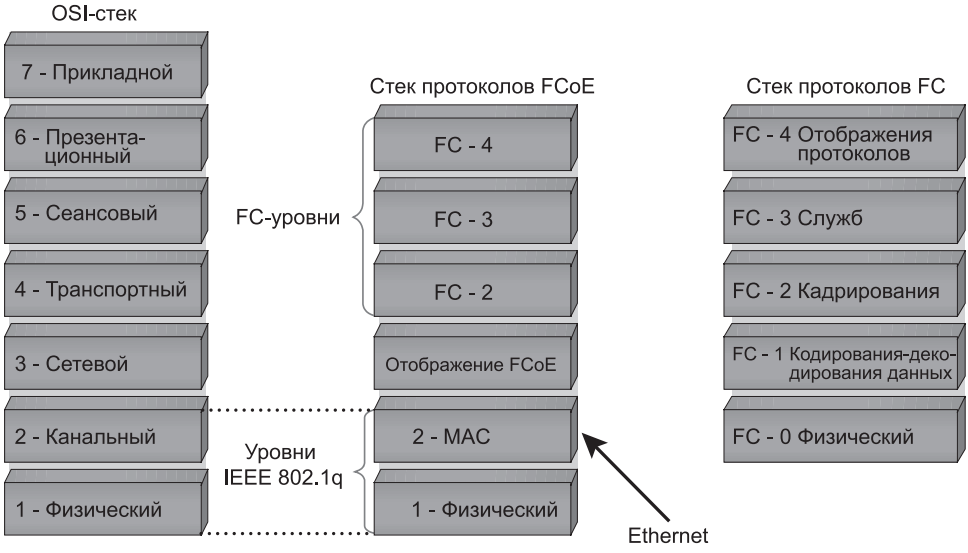


Рис. 6.17. Отображение FCoE-кадра

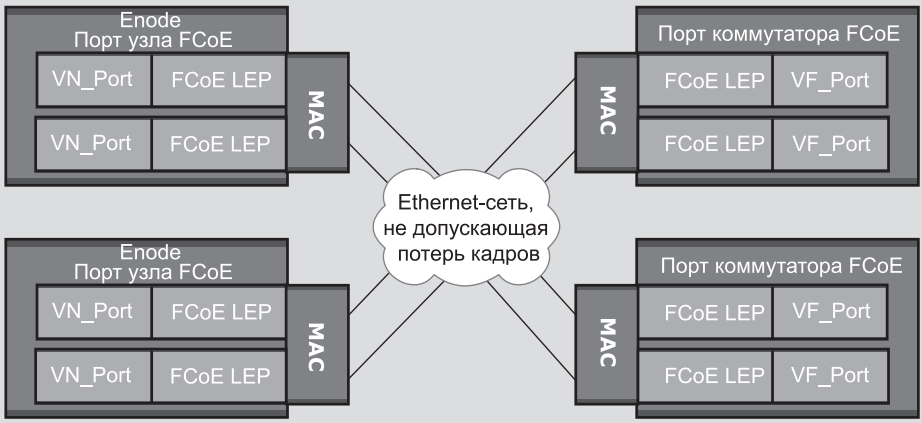
Отображение FCoE-кадра

Как показано на рис. 6.17, инкапсуляция кадров Fibre Channel осуществляется посредством их отображения на Ethernet. У Fibre Channel и у традиционных сетей имеются стеки уровней, где каждый уровень предоставляет набор определенных функциональных возможностей. FC-стек состоит из пяти уровней, от FC-0 до FC-4. Ethernet обычно рассматривается как набор протоколов, работающих на физических уровнях и уровнях канала передачи данных в стеке OSI-модели, состоящей из семи уровней. В спецификации протокола FCoE уровни FC-0 и FC-1 заменяются FC-стеком с Ethernet. Тем самым предоставляется возможность переноса уровней от FC-2 до FC-4 по Ethernet-уровню.

ПОРТЫ FCOE

Для перемещения FC-кадров FCoE-портам нужно эмулировать поведение FC-портов и превращаться в виртуальные FC-порты. Для определения различных портов в сети в FCoE используется такая же технология, как и в FC. У FCoE имеются следующие порты (см. следующий за перечнем рисунок):

- VN_Port (виртуальный N_Port) — порт в улучшенном Ethernet-узле — Enhanced Ethernet node (или Enode). Enode-узлы являются оконечными точками, такими как сервер, с конвергентными сетевыми адаптерами (CNA);
- VF_Port (виртуальный F_Port) — порт виртуальной системы коммутации в FCoE-коммутаторе;
- VE_Port (виртуальный E_Port) — виртуальный порт расширения в FCoE-коммутаторе для линий связи между коммутаторами (ISL-линий).



Между MAC и виртуальными портами находятся оконечные точки связи FCoE — Link End Points (LEP). LEP-точки отвечают за инкапсуляцию и деинкапсуляцию FC-кадров и передачу и получение инкапсулированных кадров через виртуальный порт.

6.3.4. Технологии, обеспечивающие работу FCoE

Обычному Ethernet свойственно терять данные, то есть пропускать или терять кадры при передаче. Конвергентный улучшенный Ethernet — Converged Enhanced Ethernet (CEE), или Ethernet без потерь, привносит в существующий стандарт Ethernet новую спецификацию, избавляющую Ethernet от его склонности к потере кадров. 10-гигабитный Ethernet становится вполне приемлемым вариантом для построения сетей хранения данных, подобных FC. Ethernet без потерь требует наличия определенных функциональных возможностей. Эти возможности определяются и составляются группой по выполнению задач сопряжения оборудования дата-центров — data center bridging (DCB) task group, которая является частью рабочей группы IEEE 802.1, и включают в себя:

- управление потоком на основе приоритетов;
- расширенный выбор передач;
- уведомление о перегрузке;
- протокол обмена при сопряжении оборудования дата-центра — Data center bridging exchange protocol.

Управление потоком на основе приоритетов — Priority-Based Flow Control (PFC)

В обычных FC-каналах с перегрузками справляются с помощью управления потоком на основе разрешений (credit-based flow control), гарантирующего отсутствие потерь кадров и находящегося на уровне канала связи. В обычном Ethernet в сочетании с TCP/IP используется механизм управления потоком с отбрасыванием пакетов. Этот механизм не гарантирует отсутствия потерь. С целью создания Ethernet без потерь эта проблема решается путем использования управляющего кадра IEEE 802.3x Ethernet PAUSE. Когда буфер приема заполнен, получатель может послать отправителю запрос на паузу (PAUSE). При получении PAUSE-кадра отправитель останавливает передачу кадров, что гарантирует отсутствие потерь. Недостаток использования Ethernet-кадра PAUSE заключается в том, что он работает со всем каналом, по которому могут передаваться сразу несколько потоков данных.

PFC предоставляет механизм управления потоком на уровне канала связи. PFC создает на одном физическом канале восемь отдельных виртуальных каналов и позволяет любому из этих каналов становиться на паузу и возобновлять передачу данных независимо от работы других каналов. PFC позволяет использовать механизм паузы на основе приоритетов пользователя или классов обслуживания. Использование паузы на основе приоритетов позволяет создавать для трафика, например для FCoE-трафика, каналы, не допускающие потерь. Этот PAUSE-механизм обычно реализуется для FCoE, в то время как обычный TCP/IP-трафик продолжает отбрасывать пакеты. На рис. 6.18 показано, как физический Ethernet-канал делится на восемь виртуальных каналов и допускает перевод в режим PAUSE отдельно взятого виртуального канала, не оказывая при этом никакого влияния на трафик других каналов.

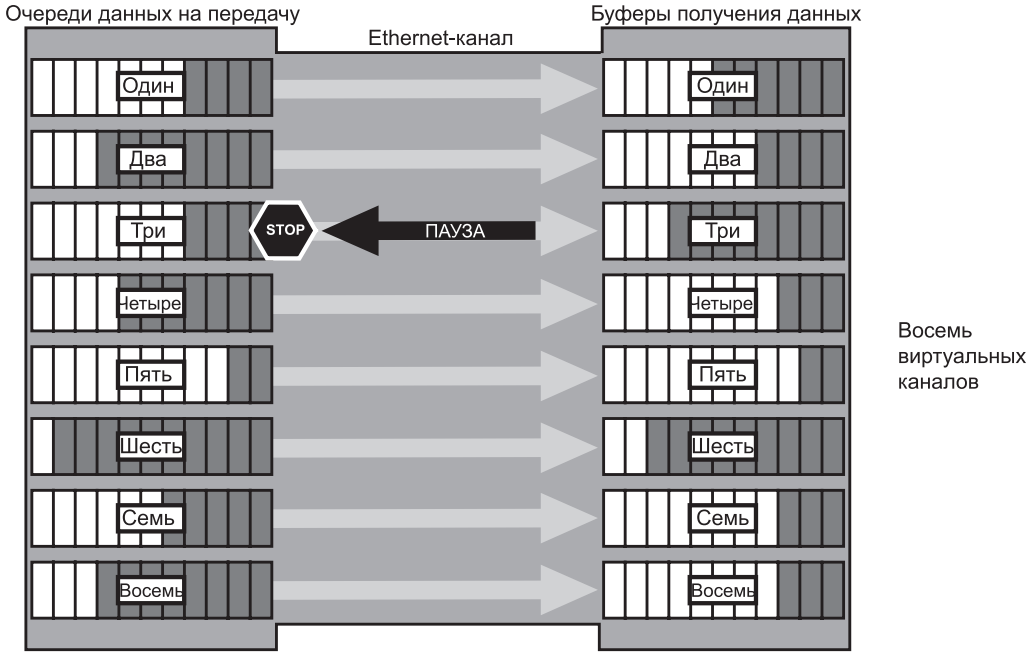


Рис. 6.18. Управление потоком на основе приоритетов

Расширенный выбор передач — Enhanced Transmission Selection (ETS)

Расширенный выбор передач представляет собой среду управления назначением полос пропускания различным классам трафика, таким как LAN и SAN, и обмену данными между процессами — Inter Process Communication (IPC). Когда конкретный класс трафика не использует выделенную ему полосу пропускания, ETS позволяет другим классам трафика использовать доступную полосу пропускания.

Уведомление о перегрузке — Congestion Notification (CN)

Уведомление о перегрузке обеспечивает сквозное управление перегрузками для таких протоколов, как FCoE, не имеющих встроенного механизма управления перегрузками. Уведомление о перегрузке на уровне канала передачи данных предоставляет механизм для определения состояния перегрузки и уведомления источника о том, что нужно удалить трафик из перегруженных каналов. Уведомление о перегрузке на уровне канала позволяет коммутатору отправлять сигнал другим портам, которые должны остановить свою работу или замедлить передачу данных. Процесс уведомления о перегрузке и управление, осуществляемое с его помощью, показаны на рис. 6.19, где представлена связь между узлами А (отправитель) и Б (получатель). Когда на приемной стороне возникает перегрузка, алгоритм, запускаемый в коммутаторе, генерирует сообщение о перегрузке в адрес узла-отправителя (узел А). В ответ на CN-сообщение отправитель ограничивает скорость передачи данных.

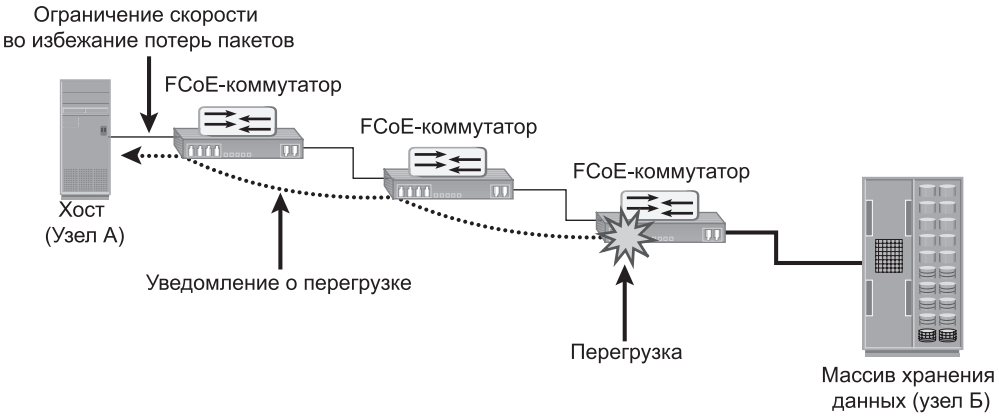


Рис. 6.19. Уведомление о перегрузке

Протокол обмена при сопряжении оборудования дата-центра — Data Center Bridging Exchange Protocol (DCBX)

DCBX-протокол предусматривает обмен обнаруженной информацией и возможностями и помогает устройствам, поддерживающим конвергентный улучшенный Ethernet — Converged Enhanced Ethernet (CEE), передавать и настраивать свои свойства во взаимодействии с другими подключенными к сети CEE-устройствами. DCBX-протокол используется для того, чтобы коммутаторы и адаптеры могли провести сопряжение своих возможностей, и позволяет коммутатору распространить значения настроек на все подключенные адаптеры. Это помогает обеспечить согласованную настройку всей сети.

Резюме

IP SAN позволила организациям сферы информационных технологий задействовать существующие инфраструктуры сетей хранения данных при вполне оправданных затратах. Теперь с помощью технологии IP SAN, повышающей коэффициент использования хранилищ данных на предприятиях, сети хранения данных могут получить необходимое географическое распределение. В качестве решения вопроса обеспечения непрерывности бизнес-процессов между дата-центрами был разработан протокол FCIP.

Поскольку работа IP SAN-сетей основана на стандартных IP-протоколах, сетевые администраторы уже знакомы с положенными в их основу концепциями, механизмами обеспечения безопасности и средствами управления. Это позволяет ускорить внедрение IP SAN в организациях. В этой главе были подробно рассмотрены две технологии IP SAN: iSCSI и FCIP. В ней также рассмотрена набирающая популярность технология FCoE, позволяющая

осуществлять передачу как LAN-, так и SAN-трафика по одной и той же физической сетевой инфраструктуре.

SAN-сеть предлагает высокопроизводительное решение по построению сетей хранения данных, но при этом она не допускает совместного использования данных несколькими хостами. Организациям может понадобиться совместное использование данных или файлов сразу несколькими разнородными клиентами с целью обеспечения их совместной работы.

В следующей главе подробно рассматриваются сетевые устройства хранения данных — network-attached storage (NAS), решения, предоставляющие среду совместного использования файлов разнородными клиентами. Поскольку NAS-устройства специально предназначены для совместного использования файлов, то по сравнению с традиционными файловыми серверами они обеспечивают более высокий уровень производительности.

УПРАЖНЕНИЯ

1. Как iSCSI справляется с процессом аутентификации? Исследуйте возможные варианты.
2. На сколько процентов можно снизить протокольные издержки в iSCSI с настройкой на использование крупных Jumbo-кадров при значении максимального размера передаваемого блока данных (MTU) 9000 байт по сравнению с передачей стандартного IP-пакета?
3. Почему в среде мостового iSCSI-подключения следует устанавливать значение MTU не менее 2500 байт?
4. Почему свойственная стандартному Ethernet потеря пакетов делает его неподходящим для многоуровневой реализации FCoE? Как с этой проблемой справляется конвергентный улучшенный Ethernet (CEE)?
5. Сравните различные протоколы дата-центра, использующие Ethernet в качестве физического посредника для передачи трафика хранения данных.

Глава 7
Сетевые устройства хранения данных

КЛЮЧЕВЫЕ ПОНЯТИЯ

- NAS-устройства
- Совместное сетевое использование файлов
- Унифицированные, шлюзовые и наращиваемые NAS-устройства
- Подключение к NAS и используемые протоколы
- Производительность NAS-устройств
- MTU и Jumbo-кадры
- TCP-окно и объединение каналов связи
- Виртуализация на уровне файлов

Совместное использование файлов, как следует из этого термина, дает возможность пользователям использовать файлы совместно с другими пользователями. Традиционные методы совместного использования файлов предполагают копирование файлов на портативные носители информации, такие как дискета, компакт-диск, DVD или накопители, подключаемые к порту USB, и доставку их другим пользователям, с которыми они будут совместно использоваться. Но корпоративной среде с большим количеством пользователей, находящихся в разных местах и нуждающихся в совместном использовании файлов, такой метод не подходит.

Гибкость в совместном использовании файлов множеством пользователей, находящихся на значительном удалении друг от друга, обеспечивает общее использование файлов по сети. Для этого на файловых серверах используется клиент-серверная технология. Чтобы справиться с огромным ростом объема файловых данных в корпоративной среде, организациям пришлось развертывать большое количество файловых серверов. Эти серверы подключаются либо к серверной дисковой памяти — direct-attached storage (DAS), либо к хранилищу данных, подключенному к сети хранения данных — storage area network (SAN). Это приводит к росту количества «островков», состоящих из перегруженных и недогруженных файловых серверов и хранилищ. Кроме того, подобные среды плохо наращиваются, требуют больших затрат на решение вопросов управления и представляют

собой весьма сложную структуру. Для решения этих проблем были созданы сетевые устройства хранения данных (NAS).

NAS является специально разработанным высокопроизводительным устройством совместного использования файлов и хранения данных. Оно позволяет своим клиентам совместно использовать файлы по IP-сети, предоставляя преимущества объединения серверов и исключая при этом необходимость в использовании нескольких файловых серверов. NAS также объединяет используемые клиентами хранилища данных в единую систему, упрощая тем самым управление системой хранения. Для обеспечения доступа к файловым данным в NAS используются сетевые протоколы и протоколы совместного использования файлов. В их числе протокол TCP/IP, используемый для передачи данных, а также общая межсетевая файловая система — Common Internet File System (CIFS) и сетевая файловая система — Network File System (NFS) для сетевого доступа к файлам. NAS позволяет совершенно прозрачно организовывать совместное использование файлов как пользователям UNIX, так и пользователям Microsoft Windows.

Для реализации специфических задач файлового обслуживания NAS-устройство использует собственную операционную систему и интегрированные аппаратные и программные компоненты. Операционная система NAS оптимизирована под файловый ввод-вывод и, следовательно, выполняет его лучше, чем сервер общего назначения. В результате NAS-устройство может обслуживать больше клиентов, чем традиционные файловые серверы, предоставляя при этом все преимущества объединения серверов.

Сетевая среда совместного использования файлов состоит из нескольких файловых серверов или NAS-устройств. По соображениям экономии или производительности может потребоваться перемещение файлов с одного устройства на другое. Решение, обеспечивающее мобильность файлов без нарушения режима работы, предоставляется в среде их совместного использования с помощью виртуализации на уровне файлов. Это решение позволяет перемещать файлы между NAS-устройствами, даже если к этим файлам осуществляется доступ.

В этой главе описаны компоненты NAS, различные типы NAS-реализаций и используемые в них протоколы совместного доступа к файлам. В ней также объясняются факторы, оказывающие влияние на производительность NAS-устройств, и рассматривается виртуализация на уровне файлов.

7.1. Сравнение NAS-устройств с серверами общего назначения

NAS-устройство оптимизировано под выполнение функций обслуживания файлов, то есть под хранение и извлечение файлов, а также обеспечение доступа к ним для приложений и клиентов. Как показано на рис. 7.1, сервер общего назначения может быть использован для размещения любого

приложения, поскольку он работает под управлением универсальной операционной системы.

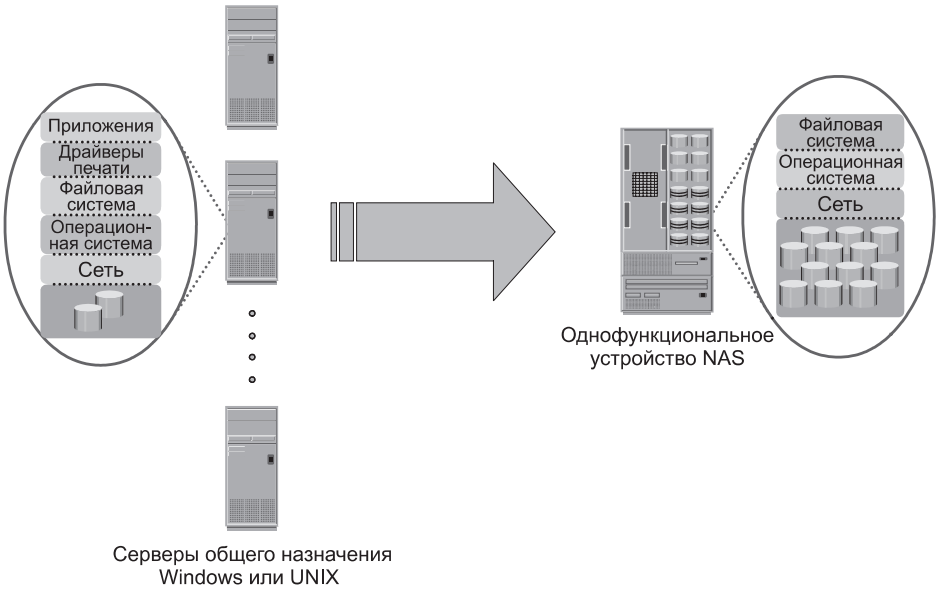


Рис. 7.1. Сравнение NAS-устройства и сервера общего назначения

В отличие от сервера общего назначения, NAS-устройство предназначено для обслуживания файлов. У него имеется специализированная операционная система, предназначенная для обслуживания файлов с использованием стандартных промышленных протоколов. Для обеспечения высокой доступности некоторые поставщики NAS-устройств поддерживают такое свойство, как изначально присущая такому устройству кластеризация.

7.2. Преимущества NAS

К предоставляемым NAS преимуществам относятся:

- **поддержка комплексного доступа к информации.** NAS обеспечивает эффективный совместный доступ к файлам и поддерживает конфигурации «многие к одному» и «один ко многим». Конфигурация «многие к одному» позволяет NAS-устройству одновременно обслуживать множество клиентов. Конфигурация «один ко многим» позволяет одному клиенту одновременно подключаться к множеству NAS-устройств;
- **повышенная эффективность.** По сравнению с файловым сервером общего назначения NAS-устройство обеспечивает более высокую производительность, поскольку в нем используется операционная система, специально предназначенная для обслуживания файлов;

- **повышенная гибкость.** С помощью применения стандартных промышленных протоколов NAS обеспечивает совместимость с клиентами как UNIX-, так и Windows-платформ. NAS — гибкое устройство и может обслуживать из одного и того же источника запросы от клиентов различных типов;
- **централизованное хранение.** NAS обеспечивает централизованное хранение данных, сводящее к минимуму их дублирование на клиентских рабочих станциях и гарантирующее более высокий уровень их защиты;
- **упрощенное управление.** NAS предоставляет централизованную консоль, позволяющую эффективно управлять файловыми системами;
- **масштабируемость.** Благодаря высокой производительности и быстрой реакции на запросы NAS обеспечивает хорошую масштабируемость под бизнес-приложения различных типов и профилей;
- **высокий уровень доступности данных.** NAS предоставляет высокоэффективные варианты репликации и восстановления данных, позволяющие гарантировать высокий уровень их доступности. В NAS используются избыточные компоненты, предоставляющие максимальное количество вариантов подключения. Для преодоления сбоев в работе в NAS-устройстве поддерживается технология кластеризации;
- **безопасность.** NAS обеспечивает безопасность, аутентификацию пользователей и блокировку файлов с помощью стандартных промышленных схем безопасности;
- **низкая стоимость.** NAS использует общедоступные и недорогие компоненты Ethernet;
- **простота развертывания.** Настройки клиента минимальны, так как клиенты для подключения к NAS используют встроенное программное обеспечение.

7.3. Файловые системы и совместный сетевой доступ к файлам

Файловая система — это структурированный способ хранения и организации файлов данных. Для упрощения поиска файлов и доступа к ним многие файловые системы поддерживают таблицу доступа к файлам.

7.3.1. Доступ к файловой системе

Прежде чем файловую систему можно будет использовать, она должна быть смонтирована. В большинстве случаев операционная система монтирует локальную файловую систему в процессе начальной загрузки. В ходе монтирования создается связь между файловой системой NAS-устройства

и операционной системой клиента. При монтировании файловой системы операционная система организует файлы и каталоги в древовидную структуру и предоставляет пользователю преимущественное право доступа к этой структуре. Корень древовидной структуры находится в точке монтирования, название которой соответствует соглашениям, принятым в той или иной операционной системе. Пользователи и приложения могут проходить по всему дереву от корня до концевых узлов, насколько это позволяет файловая система. Файлы находятся в концевых узлах, а каталоги и подкаталоги — в промежуточных. Доступ к файловой системе прекращается при ее демонтировании. Пример структуры каталогов, используемой в операционной системе UNIX, показан на рис. 7.2.

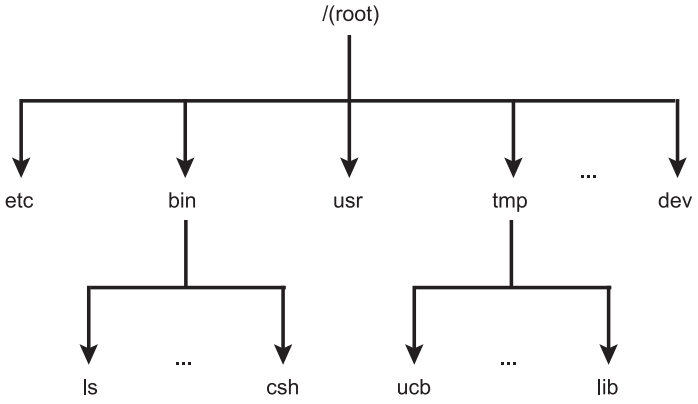


Рис. 7.2. Структура каталогов UNIX

7.3.2. Совместное сетевое использование файлов

К совместному сетевому использованию файлов относятся сохранение файлов и доступ к ним по сети. В среде совместного использования файлов пользователь, создающий файл (создатель или владелец файла), определяет тип доступа, предоставляемого другим пользователям (чтение, запись, исполнение, добавление и удаление), и контролирует изменения в файле. Если получить доступ к файлу одновременно пытаются сразу несколько пользователей, то для обеспечения целостности данных и в то же время для сохранения возможности совместного использования файла необходима схема блокировки доступа.

Примерами методов совместного использования файлов могут послужить протокол передачи файлов — file transfer protocol (FTP), распределенная файловая система — Distributed File System (DFS), клиент-серверные модели, использующие такие протоколы совместного использования файлов, как NFS и CIFS, а также пиринговая модель peer-to-peer (P2P).

FTP — это клиент-серверный протокол, позволяющий передавать данные по сети. FTP-сервер и FTP-клиент общаются друг с другом, используя

в качестве транспортного протокола TCP. FTP, как определено стандартом, не является безопасным методом передачи данных, поскольку он передает по сети незашифрованную информацию. Безопасность к исходной FTP-спецификации добавляется с помощью использования шифрованного соединения FTP через Secure Shell (SSH). Использование FTP через SSH называется Secure FTP (SFTP).

Распределенная файловая система — distributed file system (DFS) — это файловая система, распределенная между несколькими хостами. DFS может предоставить хостам прямой доступ ко всей файловой системе, обеспечивая при этом эффективное управление и защиту данных. Стандартные клиент-серверные протоколы совместного использования файлов — NFS и CIFS — позволяют владельцу файла устанавливать для конкретного пользователя или группы пользователей требуемый тип доступа, например только для чтения или для чтения и записи. Используя этот протокол, клиенты монтируют удаленные файловые системы, доступные на выделенных файловых серверах.

Служба имен, такая как система имен домена — Domain Name System (DNS), и такие службы каталогов, как Microsoft Active Directory и сетевая информационная служба — Network Information Services (NIS), помогают пользователям идентифицировать уникальные ресурсы сети и получить к ним доступ. *Протокол службы имен*, такой как облегченный протокол службы каталогов — Lightweight Directory Access Protocol (LDAP), создает пространство имен, содержащее уникальное имя каждого сетевого ресурса и помогающее распознавать ресурсы в сети.

Файлообменная модель peer-to-peer (P2P) использует пиринговую (одноранговую) сеть. P2P позволяет клиентским машинам делиться друг с другом файлами по сети напрямую. Клиенты пользуются файлообменными программами, которые ведут поиск других пиринговых клиентов. Эта модель отличается от клиент-серверной модели, в которой для хранения совместно используемых файлов применяются файловые серверы.

7.4. Компоненты NAS

У NAS-устройства имеются два основных компонента: NAS-надстройка и хранилище данных (рис. 7.3). В некоторых NAS-реализациях хранилище может быть внешним по отношению к NAS-устройству и использоваться совместно с другими хостами.

В NAS-надстройку входят следующие компоненты:

- центральный процессор и память;
- одна или несколько сетевых интерфейсных плат (NIC), обеспечивающих подключение к клиентской сети. В число сетевых протоколов, поддерживаемых NIC, входят Gigabit Ethernet, Fast Ethernet, ATM и Fiber Distributed Data Interface (FDDI);

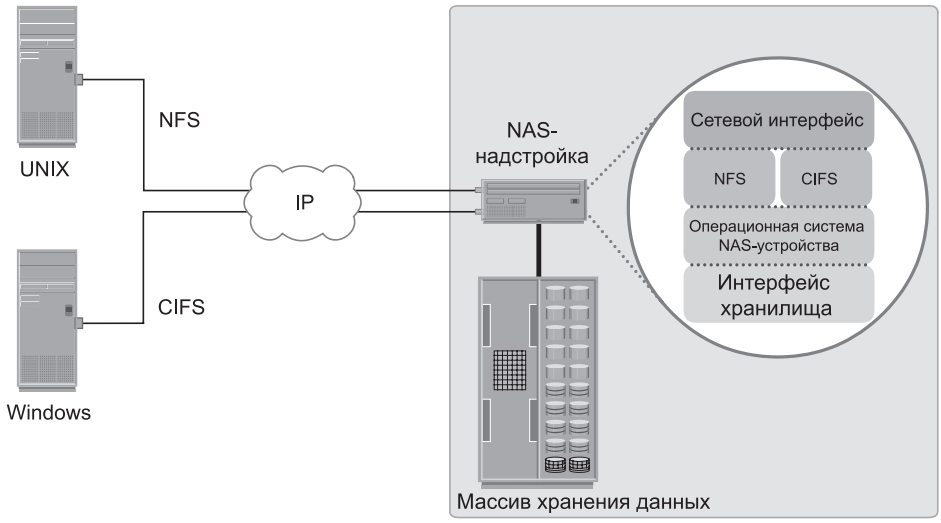


Рис. 7.3. Компоненты NAS

- оптимизированная операционная система для управления NAS-функциональностью. Она преобразует запросы на файловом уровне в блочные запросы к хранилищу, а затем преобразует данные, предоставленные на блочном уровне, в файловые данные;
- NFS, CIFS и другие протоколы совместного использования файлов;
- стандартные промышленные протоколы хранения данных и порты для подключения физических дисков и управления ими.

NAS-среда включает в себя клиентов, обращающихся к NAS-устройству по IP-сети с применением протоколов совместного использования файлов.

7.5. NAS-операции ввода-вывода

NAS предоставляет своим клиентам доступ к данным на уровне файлов. Файловый ввод-вывод инициируется запросом высокого уровня, определяющим файл, к которому осуществляется доступ. Например, клиент может запросить файл, указав его имя, место, где он находится или другие атрибуты. Операционная система NAS отслеживает размещение файлов на томе диска и для извлечения данных преобразует клиентский файловый ввод-вывод во ввод-вывод на уровне блоков. Процесс обработки запросов на ввод-вывод в NAS-среде проходит следующим образом.

- Запрашивающий (клиент) упаковывает запрос на ввод-вывод в TCP/IP-оболочку и отправляет его через сетевой стек. NAS-устройство получает этот запрос из сети.

- NAS-устройство конвертирует запрос на ввод-вывод в соответствующий запрос к физическому устройству хранения данных, являющийся запросом на ввод-вывод на уровне блоков, а затем выполняет операцию на физическом устройстве хранения данных.
- Когда NAS-устройство получает данные из хранилища, оно их обрабатывает и перепаковывает в ответ, соответствующий файловому протоколу.
- NAS-устройство упаковывает этот ответ опять в TCP/IP-контейнер и отправляет его клиенту по сети.

Этот процесс показан на рис. 7.4.

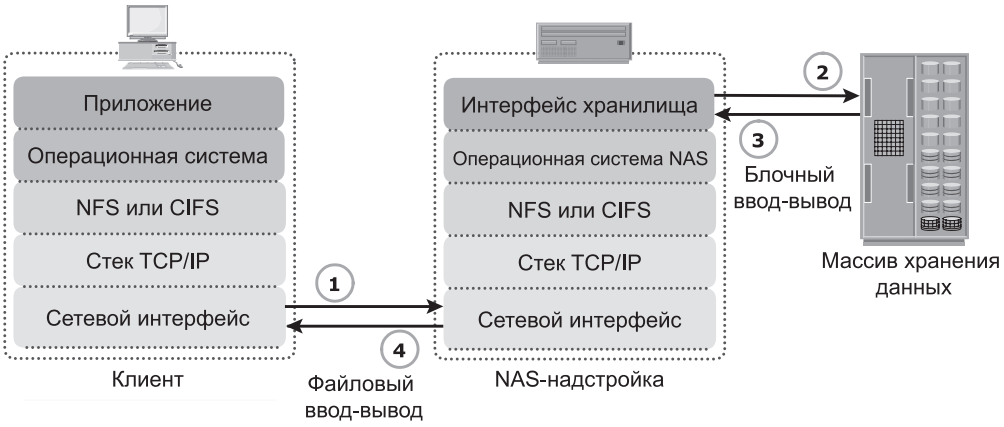


Рис. 7.4. NAS-операция ввода-вывода

7.6. Реализации NAS

К числу наиболее часто встречающихся NAS-реализаций относятся унифицированная, шлюзовая и наращиваемая. Унифицированная NAS-реализация объединяет обращения к данным на NAS- и SAN-основе в унифицированной платформе хранения данных и предоставляет унифицированный интерфейс управления обеими средами.

В шлюзовой реализации NAS-устройство использует для сохранения и извлечения данных внешнее хранилище, и в отличие от унифицированного хранилища в нем имеются отдельные административные задачи для NAS-устройства и для хранилища данных.

Наращиваемая NAS-реализация объединяет несколько узлов в общий кластерный пул. Узел может состоять либо из NAS-надстройки, либо из хранилища, либо из того и другого. В кластере NAS-операции выполняются как в едином устройстве.

7.6.1. Унифицированные NAS-устройства

Унифицированное NAS-устройство выполняет функции файлового обслуживания и сохранения файловых данных наряду с предоставлением доступа к данным на блочном уровне. Для доступа к файлам в нем поддерживается как CIFS-, так и NFS-протокол, а для доступа к данным на уровне блоков поддерживаются протоколы iSCSI и FC. Из-за объединения доступа к данным на основе NAS и SAN в единой платформе хранения данных унифицированное NAS-устройство позволяет сократить организационную инфраструктуру и снизить затраты на управление.

Унифицированное NAS-устройство состоит из одной или нескольких NAS-надстроек и хранилищ, находящихся в единой системе. NAS-надстройки подключены к контроллерам хранилищ — storage controllers (SC), предоставляющим доступ к хранилищам данных. Эти контроллеры также предоставляют возможности подключения к iSCSI- и FC-хостам. Для удовлетворения различных требований к рабочей нагрузке хранилище может состоять из накопителей разных типов, включая SAS, ATA, FC и флеш-накопители.

7.6.2. Возможности подключения унифицированных NAS-устройств

У каждой NAS-надстройки в унифицированном NAS-устройстве имеются внешние Ethernet-порты, подключенные к IP-сети. Внешние порты предоставляют возможности подключения клиентам и службам, выдающим запросы на файловый ввод-вывод. У каждого NAS-устройства имеются внутренние порты, предоставляющие возможности подключения к контроллерам хранилищ.

iSCSI- и FC-порты в контроллере хранилища позволяют хостам обращаться к хранилищу на блочном уровне напрямую или через сеть хранения данных. Пример подключений NAS-устройства показан на рис. 7.5.

7.6.3. Шлюзовые NAS-устройства

Шлюзовое NAS-устройство состоит из одной или нескольких NAS-надстроек и использует внешнее и независимо управляемое хранилище данных. Подобно унифицированному NAS-устройству, хранилище используется совместно с другими приложениями, применяющими блочный ввод-вывод. Функции управления в этом типе решения сложнее, чем в унифицированной NAS-среде, из-за наличия раздельных административных задач для NAS-надстройки и для хранилища. В шлюзовом решении для обращения к массивам хранения данных, подключенных к SAN-сети или имеющих непосредственное подключение, может использоваться такая FC-инфраструктура, как простые коммутаторы и коммутаторы класса Director.

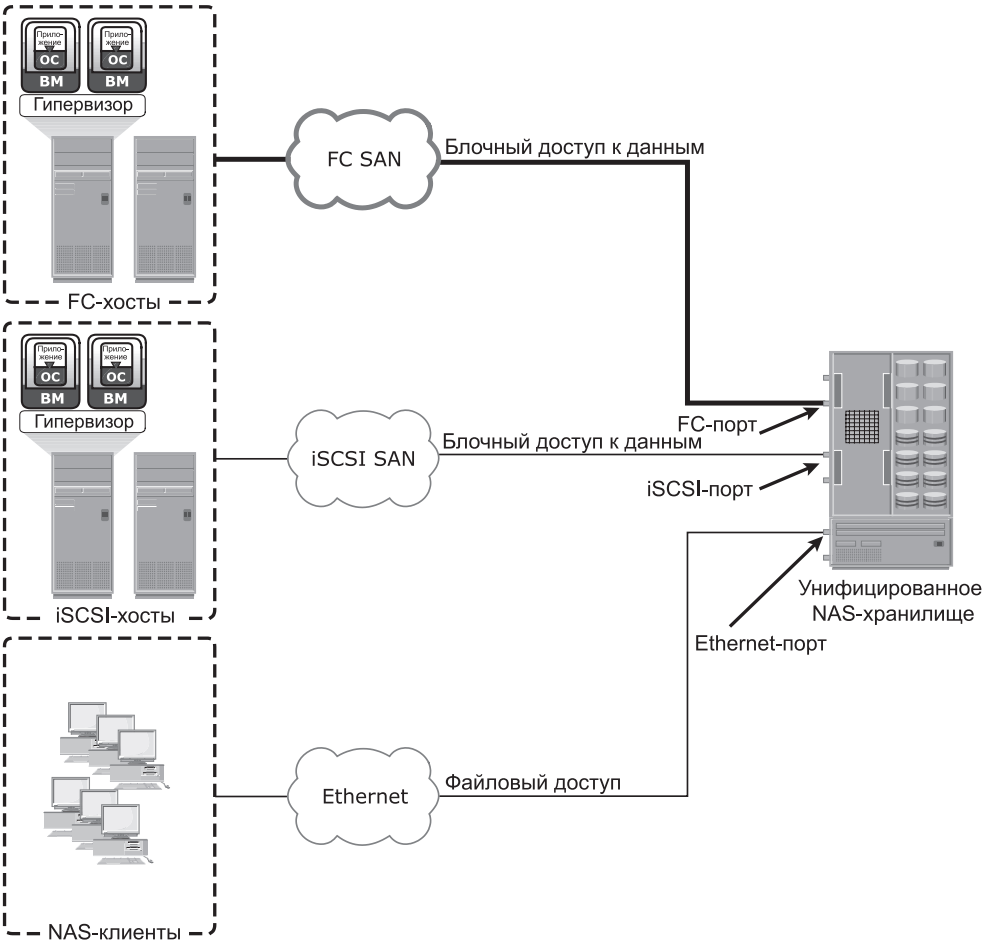


Рис. 7.5. Возможности подключения унифицированного NAS-устройства

По сравнению с унифицированным NAS-устройством шлюзовое NAS-устройство обладает более широкими возможностями по наращиванию, поскольку NAS-надстройки и массивы хранения данных могут по мере необходимости расширяться независимо друг от друга. Например, для повышения производительности NAS-устройства могут добавляться NAS-надстройки. А когда возникают ограничения по емкости хранилища, его можно расширить, добавляя емкость к SAN-сети независимо от NAS-надстройки. Так же, как и универсальное NAS-устройство, шлюзовое NAS-устройство позволяет добиться высокого коэффициента использования емкости хранилища за счет его совместной эксплуатации с SAN-средой.

7.6.4. Возможности подключения шлюзовых NAS-устройств

У шлюзового решения внешнее подключение такое же, как и у унифицированного решения. Связь между NAS-шлюзом и системой хранения данных в шлюзовом решении достигается за счет использования традиционной оптоволоконной сети хранения данных FC SAN. При развертывании шлюзового NAS-решения следует предусматривать наличие таких факторов, как множество путей для данных, избыточность системы коммутации, а также распределение нагрузки. Пример подключения шлюзового NAS-устройства показан на рис. 7.6.

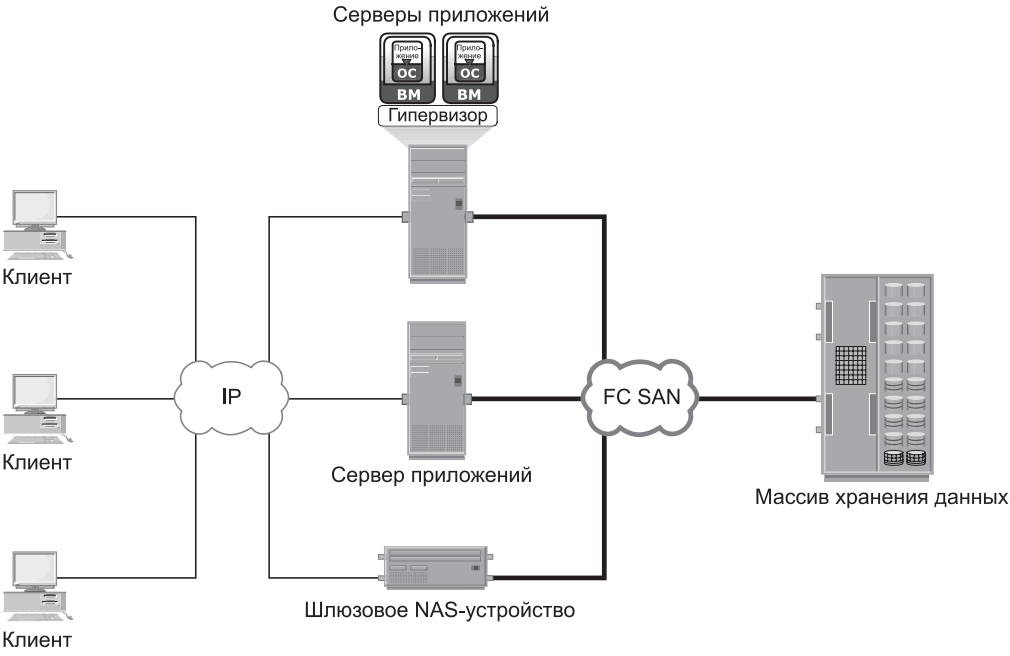


Рис. 7.6. Возможности подключения шлюзового NAS-устройства

Реализация как универсального, так и шлюзового NAS-решения требует анализа текущей SAN-среды. Этот анализ необходим для определения целесообразности объединения рабочей нагрузки NAS-устройства с существующей нагрузкой SAN-сети. Следует проанализировать SAN-сеть, чтобы определить, чем она в основном загружена — чтением или записью данных, и каков характер доступа к данным — произвольный или последовательный. Нужно также определить преобладающий размер ввода-вывода. Обычно рабочая нагрузка NAS-устройства носит произвольный характер при небольших размерах ввода-вывода. Включение рабочей нагрузки с произвольным доступом к данным в рабочую нагрузку с последовательным доступом может помешать последней. Поэтому рекомендуется разделять NAS- и SAN-диски. Кроме того, следует

определить, соответствует ли производительность при имеющейся рабочей нагрузке NAS-устройства настройкам кэша в системе хранения данных.

7.6.5. Наращиваемые NAS-устройства

Как унифицированные, так и шлюзовые NAS-реализации предоставляют возможности наращивания своих ресурсов при возрастании объема данных и повышении требований к производительности. Наращивание этих NAS-устройств предусматривает добавление к NAS-устройству центральных процессоров, памяти и накопителей. Наращивание ограничивается вместимостью помещения, в котором находится NAS-устройство, и возможностью использования дополнительных NAS-надстроек и объемов хранения данных.

Наращивание NAS-устройств позволяет создавать группу из нескольких узлов, выстраивая кластеризованную NAS-систему. Наращиваемое NAS-устройство предоставляет возможность наращивания своих ресурсов путем простого добавления узлов к кластеризованной NAS-архитектуре. Кластер работает как единое NAS-устройство и имеет централизованное управление. Узлы могут быть добавлены к кластеру при возникновении необходимости в более высоком уровне производительности или в дополнительной емкости хранилища, не создавая при этом причин для простоев оборудования. Наращиваемое NAS-устройство приспособлено под использование множества узлов с умеренными характеристиками производительности и доступности с целью создания общей системы, имеющей более высокие общие показатели производительности и доступности. Оно также характеризуется простотой использования, меньшим уровнем затрат и теоретически неограниченной наращиваемостью.

В наращиваемом NAS-устройстве создается единая файловая система, запускаемая на всех узлах кластера. Поэтому клиентам, подключившимся к любому узлу кластера, доступна вся информация, совместно используемая всеми узлами. В наращиваемом NAS-устройстве данные чередуются по всем узлам кластера, имея при этом защиту, выполненную путем зеркалирования или контроля четности. При пересылке данных от клиента к кластеру эти данные разбиваются на части и параллельно размещаются в разных узлах. Когда клиент отправляет запрос на чтение файла, наращиваемое NAS-устройство извлекает соответствующие блоки из нескольких узлов, составляя из блоков файл и предоставляя файл клиенту. По мере добавления узлов файловая система динамически разрастается и данные равномерно распределяются по всем узлам. Каждый добавленный к кластеру узел повышает общую емкость хранилища, объем памяти, возможности центрального процессора и сетевого обмена данными. Следовательно, повышается и производительность кластера.

Наращиваемые NAS-устройства хорошо подходят для решения задач обработки больших данных, с которыми сегодня сталкиваются предприятия и клиенты. Они предоставляют возможности хранения быстро возрастающих объемов данных и управления ими в одном месте наряду с возможностью соответствия широкому диапазону требований по производительности.

7.6.6. Возможности подключения масштабируемых NAS-устройств

Наращиваемые NAS-кластеры используют отдельные внутренние и внешние сети, работающие, соответственно, на внешние и внутренние подключения. Внутренняя сеть предоставляет возможности подключения для связи внутри кластера, а внешние сетевые подключения позволяют клиентам получать доступ к файловым данным и использовать их совместно с другими клиентами. Каждый узел в кластере подключен к внутренней сети. Внутренняя сеть характеризуется высокой пропускной способностью и малым временем задержки и использует высокоскоростные сетевые технологии, такие как InfiniBand или Gigabit Ethernet. Чтобы клиенты могли получить доступ к узлу, этот узел должен быть подключен к внешней Ethernet-сети. Для достижения высокого уровня доступности могут использоваться избыточные внутренние или внешние сети. Пример подключения наращиваемого NAS-устройства показан на рис. 7.7.

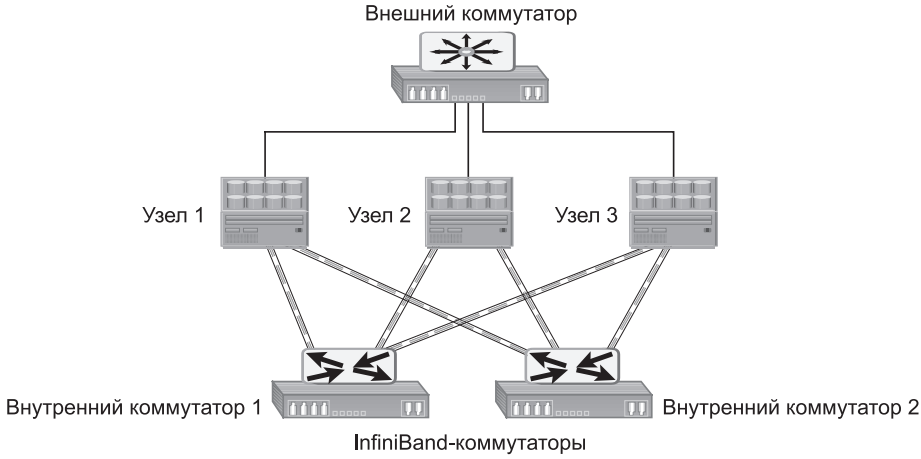


Рис. 7.7. Наращиваемое NAS-устройство с двойной внутренней и внешней сетью

INFINIBAND

InfiniBand представляет собой сетевую технологию, обеспечивающую канал связи между хостами и внешними устройствами с низким временем задержки и высокой пропускной способностью. Эта технология предоставляет последовательное соединение и часто используется в высокопроизводительной вычислительной среде для организации связи между серверами. InfiniBand допускает удаленный непосредственный доступ к памяти — remote direct memory access (RDMA), позволяющий устройству (хосту или периферийному оборудованию) обращаться к данным в памяти удаленного устройства напрямую. InfiniBand также

позволяет с помощью технологии мультиплексирования одновременно переносить по одному физическому каналу данные сразу нескольких каналов. Сетевая инфраструктура InfiniBand состоит из адаптеров каналов хоста — host channel adapters (HCA-адаптеров), адаптеров каналов конечного устройства — target channel adapters (TCA-адаптеров) и InfiniBand-коммутаторов. HCA-адаптеры находятся в составе хостов. Они предоставляют механизм подключения центральных процессоров и памяти хостов к InfiniBand-сети. Соответственно, TCA-адаптеры позволяют подключаться к InfiniBand-сети хранилищу и другим внешним устройствам. А InfiniBand-коммутаторы предоставляют возможность организации связи между HCA- и TCA-адаптерами.

7.7. Протоколы совместного использования файлов, применяемые в NAS

Для обработки запросов на файловый ввод-вывод, отправляемых удаленной файловой системе, большинством NAS-устройств поддерживаются несколько протоколов файловых служб. Как упоминалось ранее, стандартными протоколами для совместного доступа к файлам являются NFS и CIFS. NAS-устройства дают пользователям возможность совместного использования файловых данных в различных операционных средах и предоставляют средства беспрепятственной миграции из одной операционной системы в другую.

7.7.1. Сетевая файловая система

Сетевая файловая система (NFS) — это клиент-серверный протокол для совместного использования файлов, чаще всего применяемый в системах UNIX. Изначально NFS основывался на протоколе пользовательских датаграмм — User Datagram Protocol (UDP) без установления соединения. Для представления данных пользователя в нем используется машинно-независимая модель. В качестве метода межпроцессного взаимодействия между двумя компьютерами в нем используется также удаленный вызов процедур — Remote Procedure Call (RPC). Для получения доступа к удаленной файловой системе протокол NFS обеспечивает набор RPC-вызовов следующих операций:

- поиска файлов и каталогов;
- открытия файла, чтения из него данных и их записи, закрытия файла;
- изменения атрибутов файлов;
- изменения ссылок на файлы и каталоги.

Для переноса данных в NFS создается соединение между клиентом и удаленной системой. NFS (NFSv3 и более ранние версии) является протоколом *без поддержки состояния*, то есть в нем не ведутся никакие таблицы, хранящие информацию об открытых файлах и связанных с ними указателях.

Следовательно, в каждом вызове предоставляется полный набор аргументов для доступа к файлам на сервере. В число этих аргументов входят ссылка на описатель файла, конкретная позиция в файле для чтения или записи и версии NFS.

PNFS И MPFS

pNFS, являясь частью NFSv4.1, разбивает обработку протокола файловой системы на две части: обработку метаданных и обработку данных. Метаданные включают в себя информацию об объекте файловой системы, такую как его имя, местонахождение в пространстве имен, имя его владельца, список контроля доступа — access control list (ACL) и другие атрибуты. pNFS-сервер, называемый также сервером метаданных, осуществляет обработку метаданных и недоступен для передачи данных. Информацию о метаданных pNFS-серверу отправляют pNFS-клиенты. Эти клиенты обращаются к устройствам хранения данных напрямую с помощью нескольких параллельных маршрутов данных. Для выполнения операций ввода-вывода в отношении устройств хранения данных pNFS-клиент использует сетевой протокол этих устройств, например iSCSI или FC. pNFS-клиенты получают информацию об устройствах хранения данных от сервера метаданных. Поскольку pNFS-сервер освобожден от обработки данных и pNFS-клиенты могут обращаться к устройствам хранения данных напрямую, используя для этого параллельные маршруты, pNFS-механизм существенно повышает производительность pNFS-клиента.

Запатентованный компанией EMC протокол многомаршрутной файловой системы — Multi-Path File System (MPFS) работает схожим с pNFS образом. Программный MPFS-драйвер, установленный на NAS-клиенте (MPFS-сервер), отправляет метаданные файла NAS-устройству по IP-сети. MPFS-драйвер получает информацию о местонахождении данных от NAS-устройства по IP-сети. После того как станет известно местонахождение данных, MPFS-драйвер соединяется непосредственно с устройством хранения данных и позволяет NAS-клиентам обращаться к данным по SAN-сети. MPFS-архитектура, предоставляющая разные маршруты для передачи метаданных и данных файла, показана на следующем рисунке.

В настоящее время используются три версии NFS.

- **NFS версия 2 (NFSv2).** Для обеспечения сетевого соединения между клиентом и сервером без сохранения состояния в данной версии используется UDP-протокол. Такие функции, как блокировка, выполняются вне протокола.

- **NFS версия 3 (NFSv3).** Наиболее широко применяемая версия, в которой используется протокол UDP или TCP и в основу которой положен принцип работы без сохранения состояния. Для сокращения повторных выборок в версию включен ряд новых функций, таких как 64-битная разрядность, асинхронные записи и дополнительные атрибуты файла.
- **NFS версия 4 (NFSv4).** В данной версии применяется протокол TCP, в ее основу положен принцип работы с сохранением состояния. Эта версия обеспечивает повышенный уровень безопасности. Самая последняя версия с порядковым номером 4.1 является усовершенствованным вариантом NFS версии 4 с добавлением нескольких новых функций, таких как модель сеанса, параллельная NFS (pNFS) и сохранение данных.

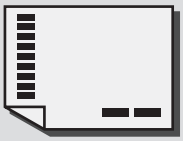
7.7.2. CIFS

CIFS представляет собой клиент-серверный протокол приложений, позволяющий клиентским программам запрашивать файлы и службы с удаленных компьютеров по сети TCP/IP. Он является общедоступным, или открытым, вариантом протокола — Server Message Block (SMB).

Протокол CIFS позволяет удаленным клиентам получать доступ к находящимся на сервере файлам. CIFS дает возможность совместного использования файлов с другими клиентами путем применения специальных блокировок. Имена файлов в CIFS кодируются символами Юникода. CIFS предоставляет следующие возможности обеспечения целостности данных:

- использование блокировки файла и записи для предотвращения переписывания пользователем работы другого пользователя в файле или записи;
- поддержка отказоустойчивости и автоматическое восстановление соединений с повторным открытием тех файлов, которые были открыты до возникновения сбоя. Воспользоваться функциями отказоустойчивости CIFS можно только в тех приложениях, которые написаны с расчетом на их использование. Кроме того, CIFS является протоколом с сохранением состояния, поскольку сервер CIFS содержит информацию о соединении, относящуюся к каждому подключенному клиенту. В случае сбоя сети или CIFS-сервера клиент получает уведомление о разрыве соединения. Если у приложения есть встроенная возможность восстановления соединения, длительность разрыва будет сведена к минимуму. Но при отсутствии такой возможности пользователю придется восстанавливать CIFS-соединение самостоятельно.

Для ссылки на удаленные файловые системы пользователи применяют простую схему задания имен файлов: \\сервер\общий_файл или \\имя_сервера.домен.суффикс\общий_файл.



В NFS-среде применяется следующая схема задания имен файлов: Сервер:/экспорт или Сервер.домен.суффикс:/экспорт.

7.8. Факторы, влияющие на производительность NAS

Так как в NAS используются IP-сети, на производительность NAS оказывают влияние связанные с IP проблемы ширины полосы пропускания и задержек. Одним из наиболее важных источников задержек (рис. 7.8) в среде NAS является перегрузка сети. Ниже приводится ряд других факторов, влияющих на производительность NAS на различных уровнях.

- **Большое количество транзитных участков.** Увеличение задержки по этой причине может произойти из-за того, что на каждом из таких участков требуется обработка IP, дополняющая задержку, возникающую в маршрутизаторе.
- **Аутентификация в службе каталогов, такой как Active Directory или NIS.** Служба аутентификации должна быть доступна в сети и иметь достаточный объем ресурсов, позволяющий выдержать связанную с аутентификацией нагрузку. В противном случае большое количество запросов на аутентификацию может увеличить задержку.
- **Повторная передача данных.** Ошибки соединения и переполнения буфера могут привести к повторной передаче данных. Это вынуждает заново отправлять те пакеты, которые не достигли указанного пункта назначения. Нужно уделить внимание согласованности настроек скоростей и дуплексных режимов на сетевых устройствах и NAS-надстройках. Неправильная настройка может привести к возникновению ошибок, повторной передаче данных и увеличению времени задержек.
- **Перегруженность маршрутизаторов и коммутаторов.** Время, необходимое перегруженному устройству в сети на отклик, всегда больше времени, которое затрачивается на это оптимально используемым или недогруженным устройством. Чтобы определить оптимальный режим использования коммутаторов и маршрутизаторов в сети, сетевые администраторы могут просмотреть статистику их использования. Если имеющиеся устройства перегружены, должны быть добавлены дополнительные устройства.
- **Поиск в файловой системе и запросы метаданных.** NAS-клиенты получают доступ к файлам на NAS-устройствах. Работа, необходимая для того, чтобы добраться до соответствующего файла или каталога,

может стать причиной задержки. Иногда задержка вызывается чрезмерной глубиной структуры каталогов и может быть уменьшена путем выравнивания этой структуры по глубине. Также могут снизить производительность непродуманная планировка файловой системы и перегруженная дисковая система.

- **Перегруженность NAS-устройств.** Клиенты, обращающиеся сразу к нескольким файлам, могут повысить загруженность NAS-устройства, что может быть обнаружено при просмотре статистики загруженности. Высокая загруженность памяти, центрального процессора или дисковой подсистемы может быть следствием непродуманной структуры файловой системы или вызываться дефицитом ресурсов в подсистеме хранения данных.
- **Перегруженность клиентов.** Клиент, осуществляющий доступ к данным по протоколу CIFS или в NFS-сети, также может быть перегружен работой. Такому клиенту требуется больше времени на составление запросов и обработку полученных ответов. В различных операционных системах имеются собственные средства отслеживания производительности, помогающие определить степень загруженности клиентских ресурсов.

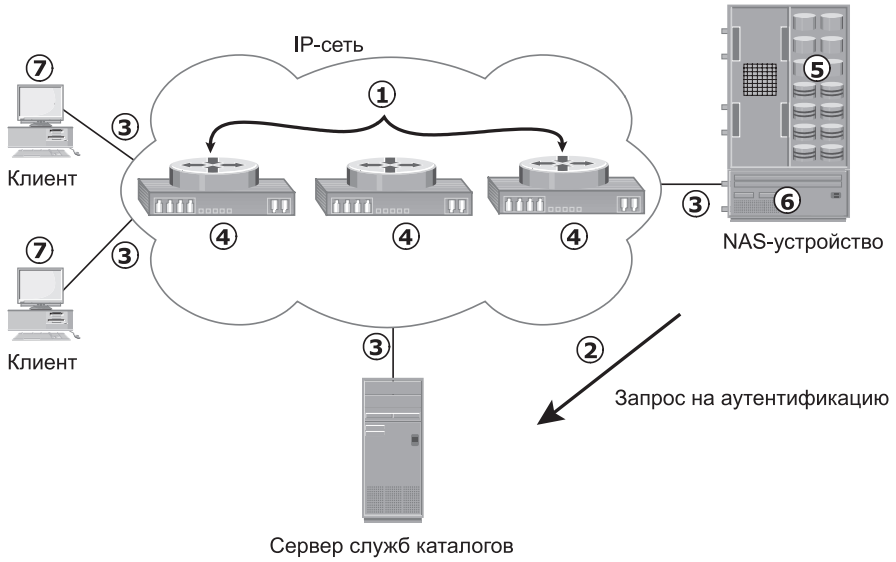


Рис. 7.8. Причины задержек

Производительность NAS-устройств может быть повышена за счет настройки *виртуальных локальных сетей* (VLAN) и установки подходящего максимального размера передаваемого блока данных (MTU) и размера окна TCP. Также высокий уровень доступности обеспечивается объединением

каналов и применением сетевых конфигураций с высокой степенью избыточности ресурсов.

Виртуальная локальная сеть (VLAN) является логическим сегментом коммутируемой сети или логической группировкой оконечных устройств, подключенных к различным физическим сетям. Оконечным устройством может быть клиент или NAS-устройство. Сегментация или группировка могут выполняться на основе потребностей бизнес-функций, команд, занимающихся разработкой проектов или приложений. VLAN является логической структурой второго уровня (уровня канала) и работает так же, как и физическая сеть. Сетевой коммутатор может быть логически поделен между несколькими VLAN-сетями, позволяя повысить эффективность использования коммутатора и сократить общие затраты на развертывание сетевой инфраструктуры.

Широковещательный трафик в отдельно взятой VLAN не передается за ее пределы, что существенно снижает издержки на его распространение, предоставляет приложениям нужную им полосу пропускания и снижает уязвимость сети от резких увеличений количества широковещательных пакетов.

VLAN-сети также обеспечивают повышенную безопасность путем ограничения пользовательского доступа, маркировки сетевых вторжений и управления размером и составом широковещательного домена. Размер наибольшего пакета, который может быть передан без фрагментации данных, определяется настройкой максимального размера передаваемого блока данных (MTU). *Определение максимального размера передаваемого блока данных для конкретного маршрута* представляет собой процесс, при котором устанавливается максимальный размер того пакета, который может быть отправлен по сети без фрагментации. Для карты Ethernet-интерфейса изначально устанавливается значение MTU, равное 1500 байт. Свойство, называемое поддержкой *Jumbo-кадров*, позволяет отправлять, получать или перемещать Ethernet-кадры с MTU более 1500 байт. Чаще всего в развертываемых сетях *Jumbo-кадр* имеет значение MTU 9000 байт. Но такой размер MTU для Jumbo-кадров устанавливается не всеми поставщиками оборудования. В условиях интенсивного сетевого трафика серверы лучше справляются с отправкой и приемом более крупных кадров. Jumbo-кадры гарантируют повышение эффективности работы сети, поскольку для одного и того же объема данных требуется меньшее количество кадров. При одной и той же полезной нагрузке пакеты большего размера расходуют меньшую долю полосы пропускания. Более крупные кадры помогают также сглаживать неожиданные всплески объемов ввода-вывода.

Размер окна TCP — это максимальный объем данных, отправляемых в любой момент времени для того или иного соединения. Например, если пара хостов обменивается данными через соединение TCP с размером окна 64 Кбайт, отправитель может послать только 64 Кбайт данных, после чего должен дождаться подтверждения от получателя. Если получатель подтвердит получение всех данных, отправитель вправе послать ему следующие 64 Кбайт данных. Если отправитель получает подтверждение от получателя о том, что

получены только первые 32 Кбайт данных, что может произойти только в том случае, если остальные 32 Кбайт данных еще находятся в пути или утрачены, отправитель может послать только следующие 32 Кбайт данных, поскольку в передаче может быть не более 64 Кбайт данных, в отношении которых ожидается подтверждение.

Теоретически размер окна TCP должен быть установлен, исходя из произведения доступной пропускной способности сети и времени, затрачиваемого на передачу и подтверждение приема данных, передаваемых по сети. Например, если пропускная способность сети 100 Мбит/с, а время, затрачиваемое на передачу и подтверждение приема, 5 мс, расчет размера окна TCP выглядит следующим образом:

100 Мбайт/с · 0,005 с = 524 288 бит, или 65 536 байт.

Размер окна TCP, указываемый в поле, контролирующем потоки данных, находится в пределах между 2 и 65 535 байт.

Объединение каналов — это процесс объединения двух или более сетевых интерфейсов в логический сетевой интерфейс, обеспечивающий более высокую пропускную способность, распределение или сбалансированность нагрузки, незаметный для пользователя обход сбойного канала и возможность масштабирования. Благодаря объединению каналов множество активных Ethernet-подключений к одному и тому же коммутатору представляется в виде одного канала. В случае отказа в таком объединении какого-либо соединения или порта весь сетевой трафик по данному каналу перераспределяется между оставшимися активными соединениями.

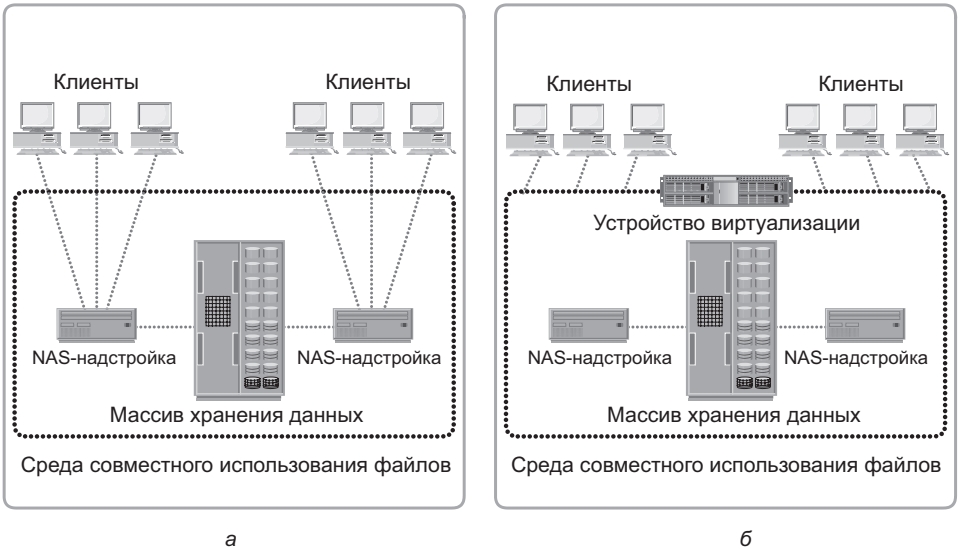
7.9. Виртуализация на уровне файлов

Виртуализация на уровне файлов устраняет зависимости между данными, доступными на уровне файлов, и местом, где файлы хранятся физически. Реализация виртуализации на уровне файлов получила в NAS-средах или в средах файловых серверов весьма широкое распространение. Виртуализация предоставляет неразрушающую мобильность файлов, позволяющую оптимизировать использование хранилища данных.

До виртуализации каждому хосту было точно известно, где расположены его файловые ресурсы. Поскольку файлы привязаны к конкретному устройству или файловому серверу, такой среде присущи проблемы неполного использования ресурсов хранилища и нехватки емкости. Из соображений производительности или при заполнении файлового сервера может возникнуть потребность в перемещении файлов с одного сервера на другой. Перемещение файлов внутри среды вызывает затруднения и приводит к их недоступности в ходе этой операции. Более того, хосты и приложения приходится перенастраивать на доступ к файлам на новом месте. Это усложняет для администраторов хранилищ задачу повышения эффективности их работы при обеспечении требуемого уровня обслуживания.

Виртуализация на уровне файлов упрощает решение вопросов мобильности файлов. Она предоставляет пользователю или приложению независимость от места хранения файла. Виртуализация на уровне файлов создает логический пул хранения данных, позволяя пользователям при обращении к файлам указывать не физические, а логические пути. Виртуализация на уровне файлов помогает перемещать файлы между подключенными к сети файловыми серверами или NAS-устройствами. Это означает, что в процессе перемещения файлов клиенты могут иметь к своим файлам доступ, не замечая этого перемещения. Клиенты могут также считывать свои файлы из прежних мест и записывать их обратно в новые места, не понимая, что их физическое размещение уже изменилось. Для отображения логического пути к файлу на имена физических путей используется глобальное пространство имен.

Среда обслуживания файлов до и после реализации виртуализации на уровне файлов показана на рис. 7.9.



а *б*

Рис. 7.9. Среда обслуживания файлов до и после виртуализации на уровне файлов: *а* — до виртуализации на уровне файлов; *б* — после виртуализации на уровне файлов

7.10. Практическая реализация концепций: EMC Isilon и EMC VNX Gateway

Устройство EMC Isilon является примером наращиваемого NAS-решения. Isilon предлагает широкие возможности масштабирования как для повышения производительности, так и для увеличения емкости хранилища. Это

устройство в состоянии справиться со всеми сложностями операций с большими данными.

Устройство VNX Gateway, входящее в семейство EMC VNX, является примером шлюзового NAS-решения. Оно предоставляет многопротокольный файловый доступ, динамическое расширение файловых систем, а также высокий уровень доступности и производительности.

Дополнительную информацию об устройствах EMC Isilon и VNX Gateway можно получить на сайте www.emc.com.

7.10.1. EMC Isilon

Устройство Isilon имеет специализированную операционную систему под названием OneFS, позволяющую наращивать NAS-архитектуру. OneFS объединяет три уровня традиционной архитектуры хранилищ: файловую систему, диспетчер томов и RAID — в один программный уровень, создавая при этом единую файловую систему, распространяющуюся на все узлы Isilon-кластера. OneFS позволяет осуществлять защиту данных и автоматическую балансировку нагрузки при их обработке. Ею также предоставляется возможность добавления хранилищ и других ресурсов без нарушения режима работы и простоев системы. При использовании OneFS возрастание или убывание пропускной способности при изменении количества узлов в кластере происходит в линейной зависимости.

С помощью дополнительной прикладной программы SmartPools операционная система OneFS позволяет смешивать в одном кластере узлы различных типов. SmartPools позволяет развертывать единую файловую систему, распространяющуюся на несколько узлов, имеющих различные характеристики производительности и емкости. Isilon предлагает узлы различных типов: X-Series, S-Series, NL-Series и Accelerator. Эти узлы различаются по цене, показателям производительности и емкости хранилищ данных. Каждый из этих типов оптимизирован под вполне определенный уровень рабочей нагрузки.

OneFS позволяет администратору системы хранения данных определять схему доступа (произвольный, параллельный или последовательный) на пофайловой или покаталожной основе. Эта уникальная возможность OneFS позволяет администратору системы хранения данных составлять решения по размещению данных, определять политики сохранения данных в кэш-памяти и политики предварительной выборки данных, добиваясь максимальной производительности для тех или иных рабочих процессов.

OneFS постоянно отслеживает общее состояние всех файлов и дисков в кластере, и в случае возникновения для компонентов какой-либо рисковой ситуации файловая система автоматически помечает проблемные компоненты для их замены или перемещения файлов, находящихся в зоне риска, на исправные компоненты без нарушения режима работы системы. В случае внезапных сбоев файловой системы в ходе ведения записи OneFS обеспечивает также целостность данных.

При добавлении нового узла хранения данных имеющаяся в OneFS функция Autobalance автоматически перемещает данные в этот новый узел по внутренней сети, построенной на основе технологии Infiniband. Эта автоматическая перебалансировка гарантирует, что новый узел не станет местом предпочтительного размещения новых данных. Функция Autobalance работает совершенно незаметно для клиентов и может настраиваться на оказание минимального влияния на систему при рабочих нагрузках, требующих высокой производительности.

Для обеспечения защиты данных в OneFS включена основная технология под названием FlexProtect. Эта технология предоставляет защиту от одновременного возникновения до четырех сбоев как в узлах, так и в отдельных накопителях в том или ином чередовании дисков. При возникновении сбоя FlexProtect гарантирует минимальную затрату времени на восстановление данных. FlexProtect дает возможность организовать защиту с учетом специфики файлов. Отдельным файлам, каталогам или частям файловой системы могут назначаться различные уровни защиты. Эти уровни выбираются на основе важности данных и характеристик рабочих процессов.

7.10.2. EMC VNX Gateway

Устройства VNX, входящие в серию Gateway, содержат одну или несколько NAS-надстроек, называемых X-Blades, которые обращаются по SAN-сети к таким внешним массивам хранения данных, как Symmetrix, VNX на основе блоков или CLARiiON. На X-Blades запускается операционная среда VNX, оптимизированная под доступ к высокопроизводительной многопротокольной сетевой файловой системе. У каждого устройства X-Blade имеются процессоры, избыточные маршруты данных, резервные источники электропитания, Gigabit Ethernet и 10-гигабитные оптические Ethernet-порты. Все X-Blades-устройства в шлюзовой системе VNX управляются контрольной станцией — Control Station, предоставляющей единую точку настройки VNX Gateway. Устройства VNX Gateway поддерживают как протокол pNFS, так и запатентованный компанией EMC протокол многомаршрутной файловой системы — Multi-Path File System (MPFS), что способствует дальнейшему повышению производительности VNX Gateway.

В VNX серии Gateway предлагаются две модели: VG2 и VG8. VG8 поддерживает до восьми устройств X-Blades, а VG2 поддерживает до двух таких устройств. X-Blades могут настраиваться на работу в режиме основных или в режиме резервных устройств. Основное X-Blade-устройство работает как NAS-надстройка, а резервное X-Blade-устройство задействуется при сбое основного X-Blade-устройства. Обработкой отказа X-Blade занимается Control Station. Кроме этого, Control Station предоставляет и другие меры обеспечения высокой доступности, такие как отслеживание сбоев, отправка уведомлений о сбоях и проведение удаленной диагностики оборудования.

Резюме

Решения, касающиеся выбора подходящей инфраструктуры хранения данных, должны быть основаны на достижении разумного баланса между стоимостью и производительностью. Организациям нужны производительность и масштабируемость сети хранения данных в сочетании с простотой использования и низкой общей стоимостью решений NAS. Как SAN-сети, так и NAS-решения продемонстрировали свои уникальные преимущества на предприятиях, а прогресс в технологии IP позволил масштабировать NAS-решения таким образом, чтобы можно было удовлетворять потребности приложений, предъявляющих высокие требования к производительности систем хранения данных. С развитием технологии сетевого хранения данных доступ к данным как на основе SAN-сетей, так и на основе NAS-решений сводится в единую платформу.

Хотя NAS-устройства неизменно имеют более высокий уровень протокольных издержек, их применение является, как правило, наиболее эффективным решением задач совместного использования файлов. С появлением протоколов MPFS и pNFS производительность NAS-устройств существенно возросла. В этих протоколах для предоставления доступа к файловым данным используются скоростные возможности SAN-сети. Они также разгружают NAS-устройства при обработке файловых данных. NAS-устройство может, помимо прочего, предоставить своим клиентам управление доступом на уровне файлов. Организации могут также развертывать NAS-решения для своих приложений, управляющих базами данных. Наращиваемые NAS-устройства удовлетворяют потребности в поддержании высокой производительности и большой емкости хранилищ при обработке больших данных. При использовании единой расширяемой файловой системы приложения, генерирующие большие данные, легче поддаются оптимизации и управлению. Виртуализация на уровне файлов позволяет добиться гибкости при перемещении файлов между NAS-устройствами, не прерывая доступ к файлам.

При преобразовании файлового ввода-вывода в блочный ввод-вывод и обратно NAS-устройства добавляют к клиентскому трафику дополнительные задержки. Кроме того, при структуре каталогов с глубокими вложениями и наличии управления разрешениями доступа к отдельным файлам и каталогам в NAS-устройствах возникают дополнительные издержки. По мере увеличения размеров файловой системы объем издержек в NAS-устройствах возрастает. Следовательно, NAS-клиенты испытывают на себе ограничения со стороны производительности NAS-устройств. Хотя использование протоколов pNFS и MPFS существенно повысило производительность NAS-устройств, эти протоколы могут усложнить решение вопросов обеспечения безопасности. Решить проблемы производительности и безопасности в среде обслуживания

файлов призвано объектно-ориентированное хранилище, подробно рассматриваемое в следующей главе. Унифицированное хранилище, которое также подробно рассматривается в этой главе, предоставляет единую платформу хранения для одновременного доступа к файлам, блокам и объектам. Унифицированное хранилище упрощает управление и исключает дополнительные затраты на развертывание отдельных систем хранения данных для хранения данных на основе файлов, блоков и объектов.

УПРАЖНЕНИЯ

1. SAN-сеть настроена на операции резервного копирования данных на диски, а устройство хранения данных после настройки имеет дополнительную доступную емкость. Может ли шлюзовая конфигурация NAS-устройства воспользоваться этим устройством хранения данных, подключенным к SAN-сети? Проанализируйте последствия совместного использования среды резервного копирования данных на диски, подключенные к SAN-сети, с NAS-устройством.
2. Объясните, какое влияние может быть оказано на производительность NAS, если размер окна TCP у отправителя и получателя не синхронизирован.
3. Как использование Jumbo-кадров влияет на производительность NAS-устройств?
4. Исследуйте свойства доступа к файлам и их совместного использования при применении протокола pNFS.
5. NAS-реализация сконфигурирована на Jumbo-кадры в NAS-надстройке с показателем MTU 9000. Но никакого роста производительности разработчики не получили и зафиксировали даже ее снижение. В чем может быть причина? Исследуйте требования к поддержке сквозных Jumbo-кадров в сети.
6. Каким образом виртуализация на уровне файлов гарантирует возможность перемещения файлов без нарушения режима работы?

Глава 8
Объектно-ориентированные и унифицированные хранилища данных

КЛЮЧЕВЫЕ ПОНЯТИЯ

Объектно-ориентированное хранилище данных

Контентно-адресуемое хранилище данных

Унифицированное хранилище данных

Последние исследования показали, что более 90 % созданных данных относятся к неструктурированным данным. Такой рост доли неструктурированных данных поставил перед IT-администраторами и диспетчерами хранилищ новые довольно непростые задачи. Из-за этого роста обычные NAS-устройства, являвшиеся доминирующими решениями в области хранения неструктурированных данных, стали малоэффективными. Увеличение объема данных добавило сетевым устройствам хранения данных (NAS) весьма существенные издержки, связанные с необходимостью управлять большим количеством прав доступа и вложенных каталогов. В корпоративной среде NAS-устройствам также приходится управлять большим объемом метаданных, которые создаются хостами, системами хранения данных и отдельными приложениями. Обычно эти метаданные сохраняются как часть файла и распространяются по всей среде. Это усложняет и замедляет поиск и извлечение файлов. Подобные затруднения потребовали применения вместо метаданных, которые относились к именам файлов, их расположению и т. д., более тонкого подхода к управлению неструктурированными данными на основе их содержимого. *Объектно-ориентированное хранилище* представляет собой способ хранить файловые данные в форме объектов, используя в качестве основы вместо имен и расположения содержимое и другие признаки этих файловых данных.

Из-за разных требований со стороны приложений организациям пришлось развертывать в своих дата-центрах сети хранения данных (SAN-сети), NAS-устройства и устройства объектно-ориентированного хранения данных (OSD-устройства). Развертывание столь разнородных решений хранения данных усложняет управление, увеличивает затраты и издержки среды. Идеальным было бы решение создать интегрированное хранилище данных,

которое поддерживало бы обращение к блокам, файлам и объектам. В качестве решения, объединяющего обращение к блокам, файлам и объектам в рамках единой унифицированной платформы, было создано унифицированное хранилище данных. Оно поддерживает несколько протоколов доступа к данным и может управляться с помощью единого интерфейса.

В этой главе подробно рассматриваются объектно-ориентированное хранилище данных, его компоненты и операции. Кроме того, в ней также в качестве специализированного типа OSD подробно рассматривается *контентно-адресуемое хранилище данных* — content addressed storage (CAS). Затем следует рассмотрение компонентов и методов обращения к данным в унифицированном хранилище.

8.1. Устройства объектно-ориентированного хранения данных

OSD является устройством упорядочения и хранения неструктурированных данных, например фильмов, офисных документов и графики, в виде объектов. Объектно-ориентированное хранилище обеспечивает масштабируемое, самоуправляемое, защищенное и совместное хранение данных. В OSD данные хранятся в форме *объектов*. Для хранения данных в OSD используется одноуровневое адресное пространство. Следовательно, в нем нет иерархии каталогов и файлов, в результате чего OSD-система позволяет сохранять громадное количество объектов (рис. 8.1).

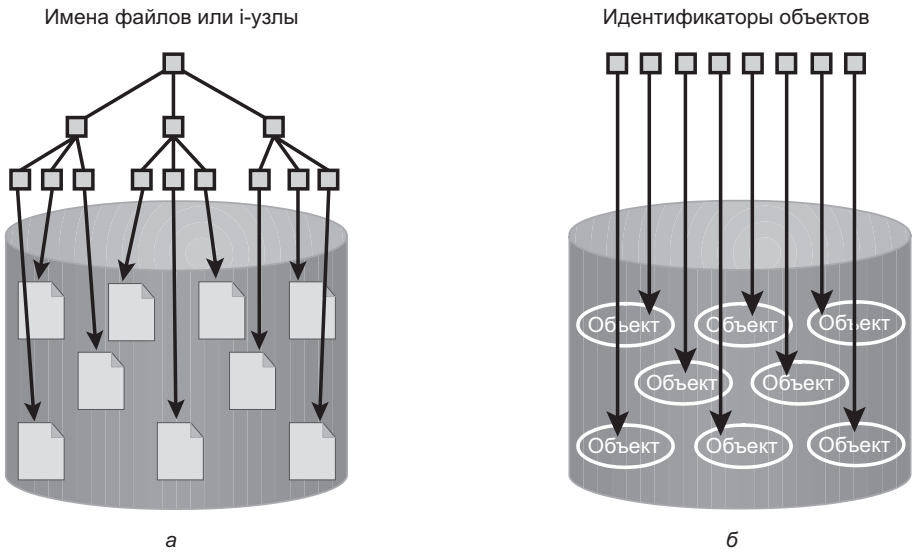


Рис. 8.1. Сравнение иерархической файловой системы и одноуровневого адресного пространства: *а* — иерархическая файловая система; *б* — одноуровневое адресное пространство

В объекте могут содержаться пользовательские данные, связанные с объектом метаданные (размер, дата, сведения о владельце и т. д.) и другие атрибуты данных (продолжительность использования, схема доступа и т. д.), показанные на рис. 8.2. Каждый сохраненный в системе объект распознается по уникальному идентификатору, называемому *идентификатором объекта* (ID объекта). Этот идентификатор создается с использованием специальных алгоритмов, например за счет применения хэш-функции к данным, и обеспечивает уникальную идентификацию каждого объекта.

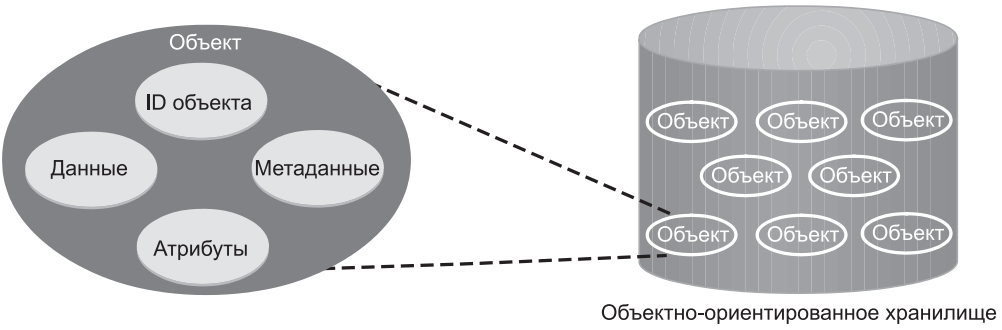


Рис. 8.2. Структура объекта

8.1.1. Архитектура объектно-ориентированного хранилища

При применении обычного метода блочного доступа ввод-вывод проходит на своем пути через различные уровни. Запрос на ввод-вывод, созданный приложением, проходит через файловую систему, канал или сеть и попадает в дисковый накопитель. Когда файловая система получает запрос на ввод-вывод от приложения, она отображает поступивший запрос на блоки диска. Для отправки данных ввода-вывода по каналу или сети к устройству хранения данных используется блочный интерфейс. Затем данные ввода-вывода записываются в блок, размещенный на дисковом накопителе. Доступ на уровне блоков показан на рис. 8.3, *а*.

У файловой системы есть два компонента: пользовательский компонент и компонент, относящийся к хранилищу. Пользовательский компонент файловой системы выполняет такие функции, как управление иерархией, присваивание имен и управление доступом со стороны пользователя. Компонент, относящийся к хранилищу, отображает файлы на их физическое размещение на дисковом накопителе.

Когда приложение обращается к данным, сохраненным в OSD-устройстве, запрос отправляется пользовательскому компоненту файловой системы. Пользовательский компонент общается с OSD-интерфейсом, который в свою очередь отправляет запрос к устройству хранения данных. У устройства хранения данных имеется компонент OSD-хранилища, отвечающий за управление доступом к объекту в устройстве хранения данных. Доступ на уровне

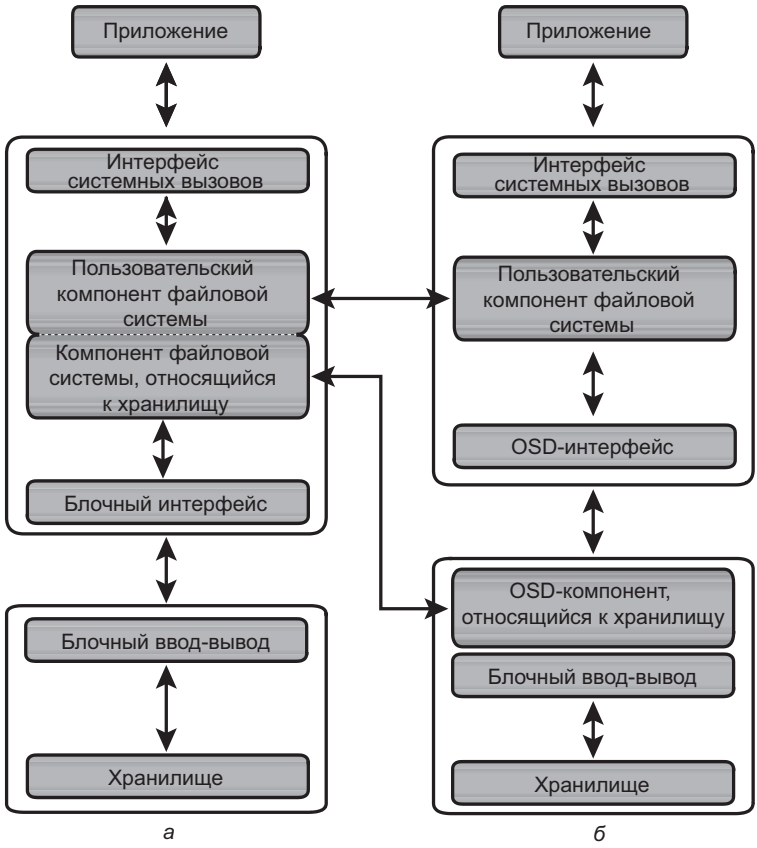


Рис. 8.3. Сравнение доступа на уровне блоков и доступа на уровне объектов: *а* — доступ на уровне блоков; *б* — доступ на уровне объектов

объектов показан на рис. 8.3, *б*. После сохранения объекта OSD отправляет серверу приложения подтверждение. Компонент OSD-хранилища управляет всеми требуемыми функциями, относящимися к низкоуровневому хранилищу и управлению его пространством. Он также управляет функциями обеспечения безопасности и контроля доступа в отношении объектов.

8.1.2. Компоненты OSD

OSD-система обычно состоит из трех основных компонентов: узлов, закрытой сети и хранилища (рис. 8.4).

OSD-система состоит из одного или нескольких *узлов*. Узел представляет собой сервер, на котором запущена операционная среда OSD и предоставляются службы для хранения и извлечения данных, имеющихся в системе, а также управления ими. У OSD-узла есть две основные службы: служба метаданных и служба хранения. Служба метаданных отвечает за создание

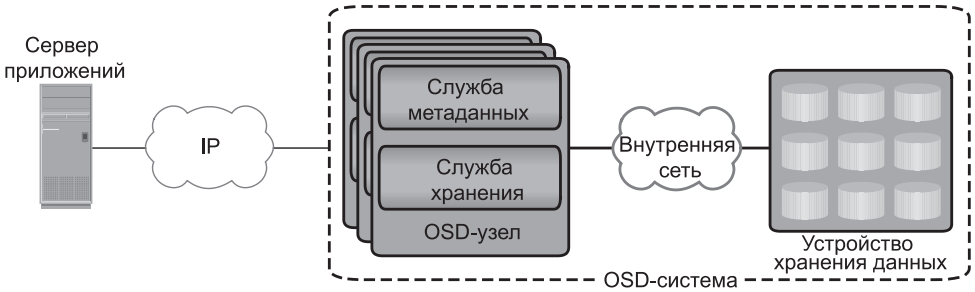


Рис. 8.4. Компоненты OSD

идентификатора объекта на основе содержимого файла (в состав содержимого могут включаться и другие атрибуты данных). Она также обслуживает отображение идентификаторов объектов на пространство имен файловой системы. Служба хранения управляет набором дисков, на которых хранятся пользовательские данные. OSD-узлы подключаются к хранилищу по внутренней сети. Эта сеть обеспечивает подключение узла к узлу и узла к хранилищу. Сервер приложений с целью сохранения и извлечения данных обращается к узлу по внешней сети. В некоторых реализациях, таких как CAS, служба метаданных может находиться на сервере приложений или на отдельном сервере.

Для хранения объектов в OSD обычно используются недорогие дисковые накопители, имеющие высокую плотность хранения данных. Если требуется дополнительный объем, система может наращиваться за счет добавления дополнительных дисковых накопителей.

8.1.3. Сохранение и извлечение объектов в OSD

Процесс сохранения объектов в OSD показан на рис. 8.5. Он осуществляется в следующем порядке.

1. Сервер приложений предоставляет предназначенный для сохранения файл OSD-узлу.
2. OSD-узел делит файл на две части: пользовательские данные и метаданные.
3. OSD-узел с помощью специального алгоритма создает ID объекта. Для получения идентификатора, уникального для этих данных, алгоритм применяется к содержимому пользовательских данных.
4. Для последующего доступа к объекту OSD-узел с помощью службы метаданных сохраняет метаданные и ID объекта.
5. OSD-узел с помощью службы хранения сохраняет пользовательские данные (объекты) в устройстве хранения данных.
6. Серверу приложений отправляется подтверждение о сохранении объекта.

Рис. 8.5. Сохранения объектов в OSD

После успешного сохранения объекта его можно будет извлечь. Пользователь обращается к данным, сохраненным в OSD, используя то же самое имя файла. Сервер приложений извлекает сохраненное содержимое с помощью ID объекта. Этот процесс проходит без участия пользователя.

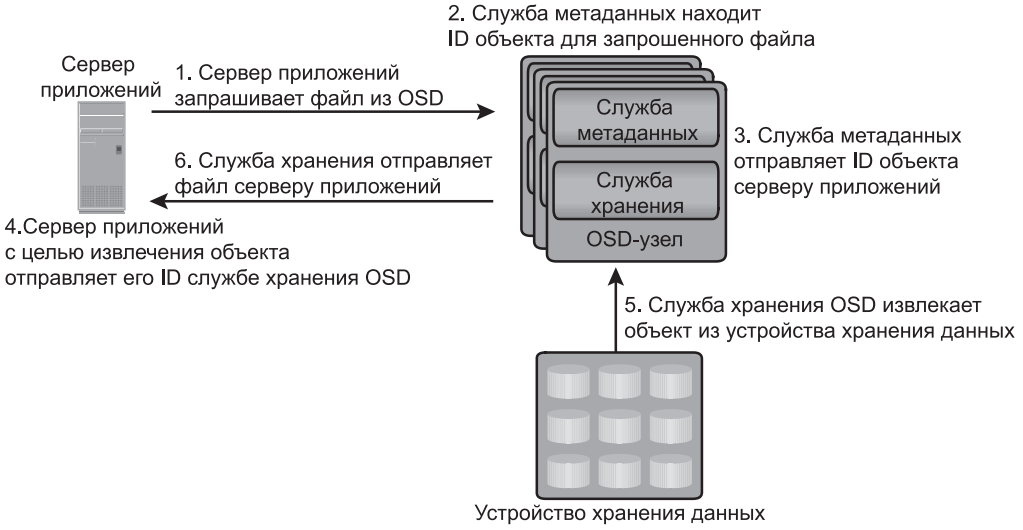


Рис. 8.6. Извлечение объекта из OSD-системы

Процесс извлечения объектов, хранящихся в OSD, показан на рис. 8.6. Он осуществляется в следующем порядке.

1. Сервер приложений отправляет OSD-системе запрос на чтение.
2. Служба метаданных извлекает ID объекта для запрашиваемого файла.
3. Служба метаданных отправляет ID объекта серверу приложений.
4. Сервер приложений для извлечения объекта отправляет его идентификатор (ID объекта) службе хранения OSD.
5. Служба хранения OSD извлекает объект из устройства хранения данных.
6. Служба хранения OSD отправляет файл серверу приложений.

8.1.4. Преимущества объектно-ориентированного хранилища

По сравнению с традиционными решениями в области хранения данных для неструктурированных данных объектно-ориентированное хранилище предоставляет ряд преимуществ. В идеале архитектура хранилища данных должна обеспечивать высокую производительность, масштабируемость, безопасность и совместное использование данных несколькими платформами. Традиционные решения в области хранения данных, такие как SAN и NAS, не предлагают всех этих полезных качеств в рамках единого решения. Объектно-ориентированное хранилище сочетает в себе преимущества обеих сфер. Оно предоставляет независимость от платформ и размещений и наряду с этим обеспечивает масштабируемость, безопасность и возможность совместного использования данных. Объектно-ориентированным хранилищам присущи следующие основные преимущества:

- **безопасность и надежность.** Основными свойствами устройств объектно-ориентированного хранения данных являются целостность данных и достоверность содержимого. Для создания объектов в OSD используются специальные алгоритмы, обеспечивающие возможность стойкого шифрования данных. Подлинность запроса проверяется в OSD не внешним механизмом аутентификации, а устройством хранения данных;
- **независимость от платформы.** Объекты являются абстрактными контейнерами данных, включающими метаданные и атрибуты. Эта особенность позволяет совместно использовать объекты в разнородных платформах в локальном или удаленном режиме доступа. Такая возможность обеспечения независимости от платформы превращает объектно-ориентированное хранилище в наиболее подходящего кандидата для сред облачных вычислений;
- **масштабируемость.** Благодаря использованию одноуровневого адресного пространства объектно-ориентированное хранилище может

справляться с большим объемом данных, не оказывая при этом отрицательного воздействия на производительность. В целях повышения производительности и емкости узлы хранилищ и OSD-узлы могут масштабироваться независимо друг от друга;

- **управляемость.** Для управления объектами и обеспечения их защиты объектно-ориентированное хранилище использует присущие ему интеллектуальные свойства. Для защиты и репликации объектов в нем используется возможность самовосстановления. Автоматически справляться с рутинными заданиями OSD помогает управление на основе избранной политики.

8.1.5. Наиболее распространенные примеры использования объектно-ориентированного хранилища

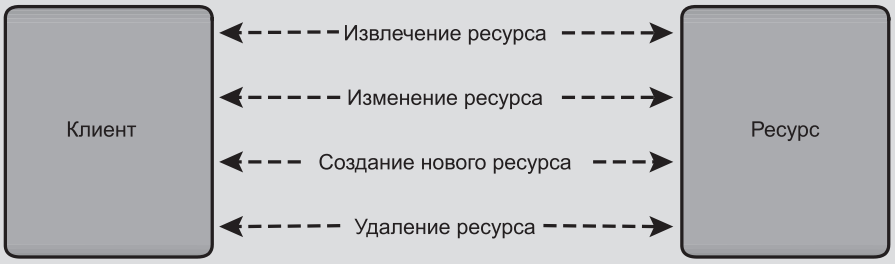
Многообещающим вариантом использования OSD является решение по архивации данных. Основными требованиями для любых решений по архивации данных являются целостность данных и их защита. Традиционными решениями по архивации данных, заключающимися в применении однократно записываемых компакт-дисков и DVD, масштабируемость и высокая производительность не обеспечиваются. В OSD данные хранятся в виде объектов, которые привязывают их к уникальному ID объекта и обеспечивают высокий уровень целостности данных. Наряду с обеспечением целостности предоставляются также масштабируемость и защита данных. Эти возможности превращают OSD в весьма приемлемый способ долговременного архивирования данных с неизменным содержимым. Специальной разновидностью объектно-ориентированного устройства, созданного для хранения фиксированного контента, является хранилище с контентным адресованием — content addressed storage (CAS). CAS-устройства рассматриваются в следующем разделе.

Еще одним примером использования OSD является хранилище на основе облака. Для обращения к ресурсам хранилища в OSD используется веб-интерфейс. OSD-хранилище предоставляет присущую ему безопасность, масштабируемость и автоматизированное управление данными. Оно также позволяет совместно использовать данные несколькими разнородными платформами или участниками, обеспечивая при этом целостность данных. Эти возможности превращают OSD в хороший вариант хранилища в системе облачных вычислений. Провайдеры облачных служб могут использовать OSD с целью предложения хранилищ в виде служб.

OSD поддерживает доступ к веб-службе через *передачу репрезентативного состояния* — representational state transfer (REST) и *простого протокола доступа к объектам* — simple object access protocol (SOAP). API-интерфейсы REST и SOAP могут быть весьма легко интегрированы с бизнес-приложениями, получающими доступ к OSD через Интернет.

REST и SOAP

REST представляет собой архитектурный стиль, разработанный для современных веб-приложений. REST предоставляет облегченные веб-службы для доступа к ресурсам (например, документам, блогам и т. д.), в которых может выполняться ряд основных операций, таких как извлечение, изменение, создание и удаление ресурсов. Веб-службы REST-стиля являются ресурсоориентированными службами. Ресурсы могут иметь уникальное расположение и распознаваться с помощью унифицированного идентификатора ресурса — Universal Resource Identifier (URI), а операции над этими ресурсами могут выполняться с помощью HTTP-спецификации. Например, если пользователь обращается к блогу с помощью REST посредством уникального идентификатора, запрос возвращает представление блога в определенном формате (XML или HTML).



а) REST

б) SOAP

SOAP является протоколом на основе XML, позволяющим устанавливать связь между веб-приложениями, запущенными под управлением различных операционных систем и основанными на применении различных языков программирования. SOAP предоставляет процессы кодирования HTTP-заголовков и XML-файлов, чтобы можно было допускать передачу информации между разными компьютерами.

8.2. Контентно-адресуемое хранилище

CAS представляет собой объектно-ориентированное хранилище, разработанное для получения безопасного хранилища постоянного доступа и извлечения фиксированного содержимого. Пользовательские данные и их атрибуты хранятся в CAS в виде объекта. Сохраненному объекту присваивается глобально-уникальный адрес, известный как *адрес контента* — content address (CA). Этот адрес берется из двоичного представления объекта. CAS предоставляет оптимизированные и централизованно-управляемые решения по хранению данных. Доступ к данным в CAS отличается от такового в других OSD-устройствах. В CAS сервер приложений обращается к CAS-устройству только через API-интерфейс CAS, запущенный на сервере приложений. Но способ хранения данных в CAS такой же, как и в других OSD-системах.

CAS предоставляет все свойства, необходимые для хранения фиксированного содержимого. CAS-хранилищу присущи следующие основные свойства:

- **подлинность содержимого.** Гарантируется подлинность хранящегося содержимого. Эта гарантия достигается за счет создания уникального контентного адреса для каждого объекта и подтверждения через определенные интервалы времени контентных адресов хранящихся объектов. Подлинность содержимого гарантируется тем, что адрес, присвоенный каждому объекту, уникален, как отпечаток пальца. При каждом считывании объекта в CAS используется алгоритм хэширования для нового вычисления контентного адреса объекта в качестве этапа подтверждения, а результат сравнивается с исходным контентным адресом этого объекта. Если объект не проходит проверку, CAS восстанавливает объект с использованием зеркальной копии или схемы защиты, использующей метод контроля четности;
- **целостность содержимого.** Предоставляется гарантия неизменности хранящегося содержимого. Для обеспечения подлинности и целостности содержимого в CAS используется алгоритм хэширования. Если фиксированное содержимое изменяется, то вместо переписывания исходного фиксированного содержимого в CAS для измененного контента создается новый адрес;
- **независимость местоположения.** Для извлечения данных в CAS используются не путевые имена каталогов или URL-адреса, а уникальный контентный адрес. Благодаря этому физическое расположение хранящихся данных не имеет никакого отношения к приложению, запрашивающему эти данные;
- **хранение в единственном экземпляре — Single-instance storage (SIS).** В CAS используется уникальный контентный адрес, гарантирующий хранение только одного экземпляра объекта. При записи нового объекта CAS-система проводит опрос, чтобы определить доступность объекта с точно таким же контентным адресом. Если такой объект

в системе доступен, то новый объект не сохраняется. Вместо этого создается только указатель на этот объект;

- **соблюдение политики сохранности.** Защита и сохранность объектов являются основными требованиями архивной системы хранения данных. После того как объект будет сохранен в CAS-системе и по отношению к нему будет определена политика сохранности, CAS не предоставит доступ к удалению объекта, пока не истечет срок действия политики;
- **защита данных.** CAS гарантирует доступность содержимого, сохраненного в CAS-системе даже при отказе диска или узла. CAS-система обеспечивает как локальную, так и удаленную защиту сохраненных в ней объектов данных. В варианте локальной защиты объекты данных либо зеркалируются, либо защищаются контролем четности. При зеркальной защите две копии объекта данных сохраняются в двух разных узлах одного и того же кластера. Это уменьшает общее доступное пространство на 50 %. При защите с помощью контроля четности объект данных разбивается на несколько частей, из которых генерируются данные для контроля. Каждая часть данных и ее информация, предназначенная для контроля четности, хранятся в разных узлах. На реализацию этого метода защиты сохраненных данных расходуется меньше памяти, но при этом восстановление данных в случае их повреждения происходит чуть дольше.

  В варианте удаленной репликации объекты данных копируются на вспомогательную CAS-систему, находящуюся в удалении от основной системы. В этом случае при отказе основной CAS-системы объекты остаются доступными со вспомогательной CAS-системы;
- **быстрое извлечение записей.** В CAS все объекты хранятся на дисках, обеспечивающих по сравнению с магнитными лентами и оптическими дисками ускоренный доступ к объектам;
- **сбалансированность нагрузки.** Для обеспечения максимальной пропускной способности и доступности данных объекты в CAS размещаются на нескольких узлах;
- **масштабируемость.** CAS позволяет добавлять к кластеру дополнительные узлы без остановки доступа к данным и при минимуме административных издержек;
- **уведомление о событиях.** В CAS ведется постоянное отслеживание состояния системы и выдается тревожное уведомление по поводу любого события, требующего внимания администратора. Для уведомления администратора о событиях используются технологии SNMP, SMTP или электронная почта;
- **самодиагностика и восстановление.** CAS автоматически определяет и восстанавливает поврежденные объекты и предупреждает администратора о потенциальной проблеме. CAS-системы могут быть

настроены на предупреждение удаленной команды специалистов, занимающихся поддержкой системы, которая может на расстоянии провести ее диагностику и восстановление;

- **контрольные записи.** В CAS отслеживаются управленческая активность, а также любой доступ к данным или факт их размещения. Требования по ведению контрольных записей являются обязательными.

8.3. Примеры использования CAS

CAS развертываются в организациях с целью решения ряда бизнес-задач. Описание таких решений дается в следующих двух разделах.

8.3.1. Решение в области здравоохранения: хранение результатов обследований пациентов

В крупных центрах охраны здоровья ежедневно проходят обследования сотни пациентов. При этом создаются большие объемы медицинских записей. Каждая запись может состоять из одного или нескольких изображений в диапазоне размеров от 15 Мбайт для стандартного цифрового рентгеновского снимка до 1 Гбайт для онкологических обследований. Медицинские записи пациентов определенный период времени хранятся в режиме постоянного доступа, чтобы лечащий врач без промедления мог воспользоваться ими. Даже если медицинская запись пациента больше не нужна, техническими требованиями может предусматриваться необходимость ее хранения в исходном формате в течение нескольких лет.

Провайдеры решений в области медицинских изображений предлагают клиникам возможность просмотра таких медицинских записей, как рентгеновские снимки, с приемлемым временем отклика и разрешением с тем, чтобы специалисты могли быстро оценить состояние здоровья пациентов. Использование CAS по этому сценарию показано на рис. 8.7. Записи пациентов остаются в основном хранилище в течение 60 дней, после чего переме-

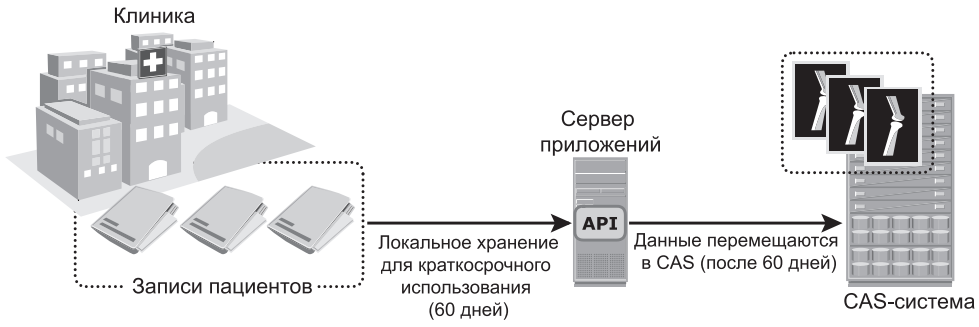


Рис. 8.7. Хранение результатов обследований пациентов в CAS-системе

щаются в CAS-систему. CAS обеспечивает долговременное хранение наряду с немедленным предоставлением доступа к данным в случае необходимости.

8.3.2. Финансовое решение: хранение финансовых записей

В типовом банковском сценарии изображения чеков, каждое из который имеет размер приблизительно 25 Кбайт, создаются и отправляются службам архивирования по IP-сети. Провайдер службы получения изображений чеков может обрабатывать приблизительно 90 млн чеков в месяц. Обычно изображения чеков активно обрабатываются в системах транзакций около 5 дней.

Следующие 60 дней изображения чеков могут быть запрошены банками или отдельными потребителями с целью проверки; по прошествии 60 дней требования к доступности резко падают. Использование CAS по этому сценарию показано на рис. 8.8. После 60 дней изображения чеков перемещаются из основного хранилища в CAS-систему и могут храниться там долгое время на основе политики сохранности. Работа с изображениями чеков является одним из примеров использования финансовой службы, которая предлагает более эффективное обслуживание с применением CAS. Транзакции клиентов, инициированные по электронной почте, контракты и записи безопасности транзакций могут содержатся в режиме постоянной доступности в течение 30 лет. В таких случаях именно CAS является тем решением, которому следует отдать предпочтение.

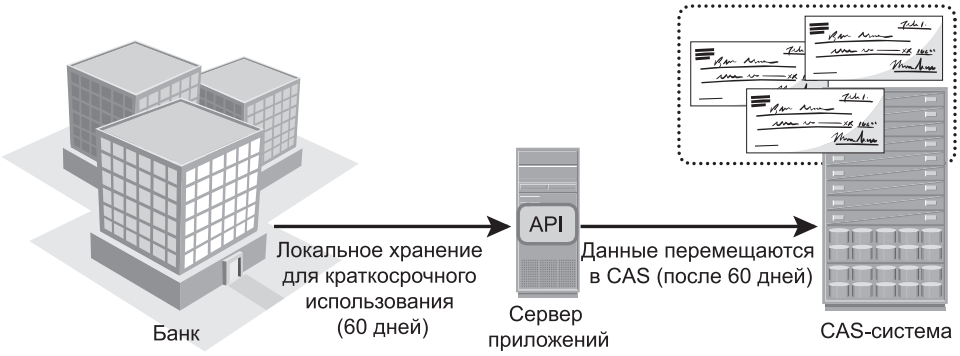


Рис. 8.8. Хранение финансовых записей в CAS-системе

8.4. Унифицированные хранилища

Унифицированное хранилище объединяет блочный, файловый и объектный доступ в единое решение по хранению данных. Оно поддерживает несколько протоколов, таких как CIFS, NFS, iSCSI, FC, FCoE, REST (передача репрезентативного состояния) и SOAP (простой протокол доступа к объектам).

8.4.1. Компоненты унифицированного хранилища

Унифицированная система хранения данных состоит из следующих основных компонентов: контроллера хранилища, NAS-надстройки, OSD-узла и хранилища. Блок-схема унифицированной платформы хранения данных показана на рис. 8.9.

Контроллер хранилища обеспечивает серверам приложений доступ на уровне блоков посредством использования протоколов iSCSI, FC или FCoE. Для непосредственного блочного доступа у него имеются интерфейсные

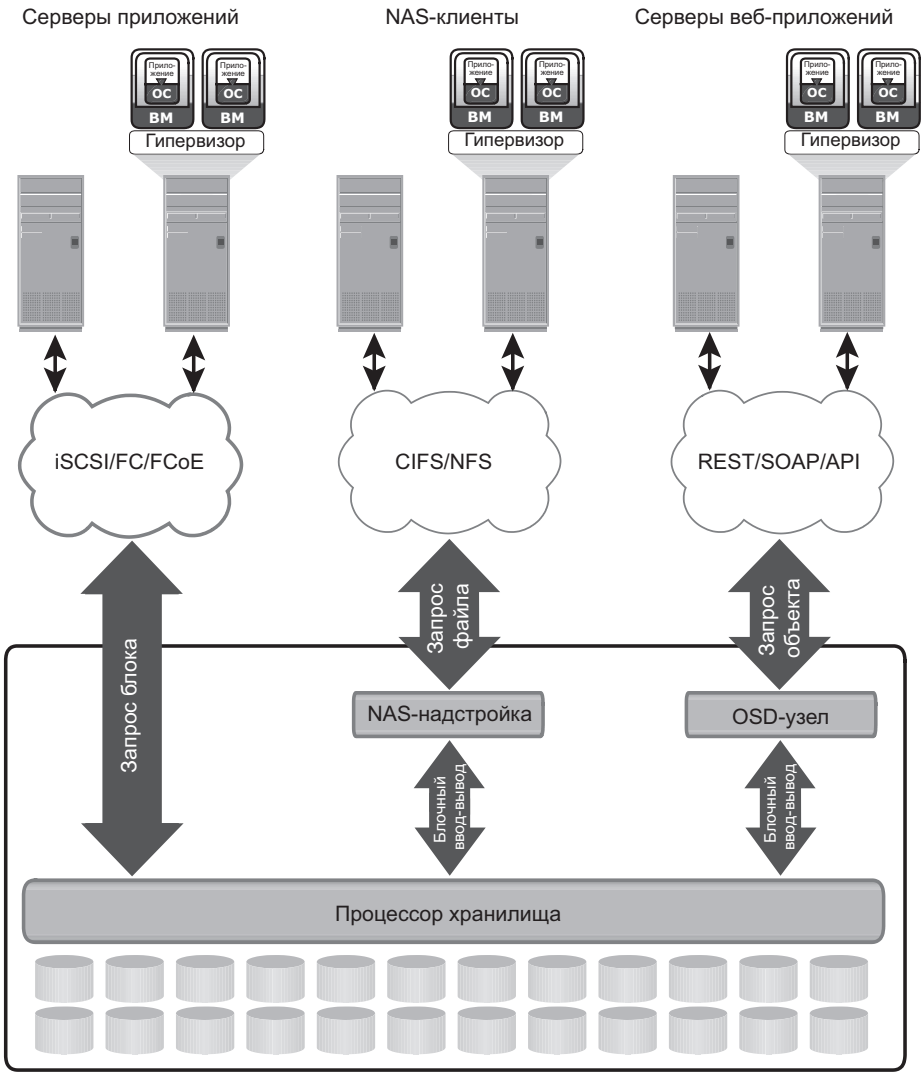


Рис. 8.9. Унифицированная платформа хранения данных

порты iSCSI, FC и FCoE. Контроллер хранилища также отвечает за управление внутренним пулом хранения данных в системе хранения. Контроллер настраивает LUN-устройства и предоставляет их серверам приложений, NAS-надстройкам и OSD-узлам. LUN-устройства, предоставленные серверу приложений, появляются в нем в виде локальных физических дисков. Файловая система настраивается на эти LUN-устройства, и они становятся доступными приложениям для хранения данных.

NAS-надстройка является специально выделенным файловым сервером, предоставляющим файловый доступ NAS-клиентам. NAS-надстройка подключается к хранилищу через контроллер хранилища обычно с помощью FC- или FCoE-подключения. Для обеспечения избыточности в системе обычно имеются две и более NAS-надстройки. LUN-устройства, предоставленные NAS-надстройке, появляются в виде физических дисков. NAS-надстройка настраивает файловые системы на эти диски, создавая NFS, CIFS или смешанную распределенную систему, и экспортирует это распределение NAS-клиентам.

OSD-узел получает доступ к хранилищу через контроллер хранилища, используя FC- или FCoE-подключение. LUN-устройства, предоставленные OSD-узлу, появляются в виде физических дисков. Эти диски настраиваются OSD-узлами, позволяя им хранить данные с серверов веб-приложений.

Доступ к данным из унифицированного хранилища

В унифицированной системе хранения данных направляемые в адрес хранилища запросы блоков, файлов и объектов проходят через различные пути ввода-вывода. Эти пути показаны на рис. 8.9.

- **Запрос на блочный ввод-вывод.** Серверы приложений подключаются к FC-, iSCSI- или FCoE-порту контроллера хранилища. Сервер отправляет блочный запрос по FC-, iSCSI- или FCoE-подключению. Процессор хранилища — storage processor (SP) обрабатывает ввод-вывод и отвечает серверу приложений.
- **Запрос на файловый ввод-вывод.** NAS-клиенты (там, где NAS-распределение установлено или отображено) отправляют файловый запрос NAS-надстройке, используя NFS- или CIFS-протокол. NAS-надстройка получает запрос, преобразует его в блочный запрос и направляет этот запрос контроллеру хранилища. По получении блочных данных от контроллера хранилища NAS-надстройка снова преобразует ответ на блочный запрос в ответ на файловый запрос и отправляет его клиентам.
- **Запрос на объектный ввод-вывод.** Серверы веб-приложений отправляют запрос на получение объекта OSD-узлу, используя для этого, как правило, REST- или SOAP-протоколы. OSD-узел получает запрос, преобразует его в блочный запрос и отправляет этот запрос диску через контроллер хранилища. Контроллер обрабатывает блочный запрос и возвращает ответ OSD-узлу, который в свою очередь предоставляет запрошенный объект серверу веб-приложений.

8.5. Практическая реализация концепций: EMC Atmos, EMC VNX и EMC Centera

EMC Atmos поддерживает объектно-ориентированное хранилище для неструктурированных данных, таких как изображения и видеоматериалы. Atmos сочетает в себе широкие возможности для расширения и специализированную интеллектуальность, нацеленные на снижение расходов и решение проблем распространения и управления, связанных с огромным объемом неструктурированных данных.

EMC VNX представляет собой унифицированную платформу хранения данных, объединяющую в одном решении блочный, файловый и объектный доступ. В этой платформе реализована модульная архитектура, объединяющая компоненты оборудования для блочного, файлового и объектного доступа. EMC VNX предоставляет функциональные возможности файлового доступа (NAS) посредством X-Blades (модулей Data Mover) и функциональные возможности блочного доступа посредством процессоров хранилищ. Дополнительно предлагается объектный доступ к хранилищу посредством использования EMC Atmos Virtual Edition (Atmos VE).

EMC Centera является простым, доступным по цене и безопасным хранилищем данных для архивирования информации. Хранилище EMC Centera сконструировано и оптимизировано специально под хранение и извлечение неизменного содержимого с учетом требований по производительности, совместимости и управляемости. По сравнению с традиционными архивными хранилищами EMC Centera предоставляет ускоренное извлечение записей, функцию хранения в единственном экземпляре — Single instance storage (SIS), гарантию подлинности содержимого, самовосстановление и поддержку множества промышленных и других регламентирующих стандартов.

Самую свежую информацию по EMC Atmos, EMC VNX и EMC Centera можно найти на сайте www.emc.com.

8.5.1. EMC Atmos

Система Atmos может быть развернута двумя способами: как специализированное оборудование или как программный комплекс в средах VMware, где Atmos VE может использовать имеющиеся серверы и хранилище.

Оборудование EMC Atmos показано на рис. 8.10. Оно состоит из серверов (узлов), подключенных к стандартным дисковым полкам.

Для обеспечения связи между узлами стойка включает в себя 24-портовый гигабитный Ethernet-коммутатор. В каждом узле установлено программное обеспечение Atmos.

Виртуальная версия Atmos VE позволяет пользователям применить все возможности Atmos в виртуальной среде. Она может быть развернута на виртуальной машине в VMware ESXi-хостах и сконфигурирована с сертифицированным компанией VMware внутренним хранилищем данных.

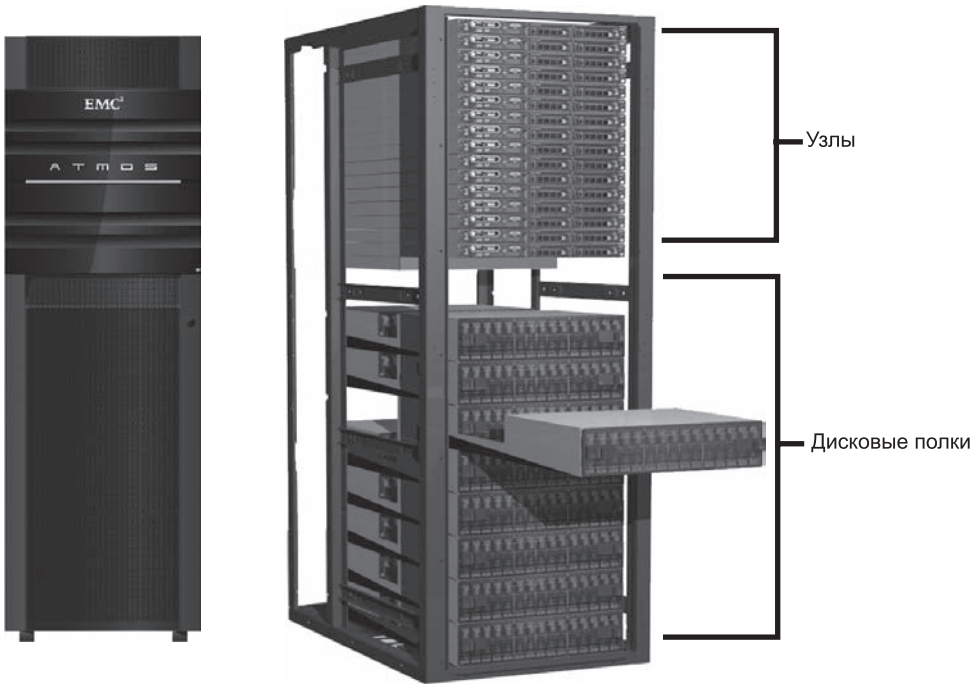


Рис. 8.10. Система хранения данных EMC Atmos

EMC Atmos обладает следующими основными характеристиками.

- **Управление на основе определенной политики.** EMC Atmos повышает эффективность работы путем автоматизированного распределения содержимого на основе определенной бизнес-политики. Политика, определяемая администратором, предписывает порядок, время и место размещения информации.
- **Защита.** Atmos предлагает два варианта защиты объектов, репликацию и распределенный контроль четности — Geo Parity:
  - *репликация* гарантирует доступность и высокую скорость доступа к содержимому путем создания избыточных копий объекта в намеченных резервных местах;
  - *Geo Parity* гарантирует доступность и высокую скорость доступа к содержимому путем деления объектов на несколько сегментов, создания в дополнение к этому сегментов контроля четности и распределения этих сегментов по одному или нескольким намеченным местам.
- **Наличие служб данных.** EMC Atmos включает такие службы данных, как сжатие и дедупликация. Эти свойства заранее заложены в Atmos, ими можно управлять через задание определенной политики.

- **Наличие веб-служб и унаследованных протоколов.** EMC Atmos предоставляет гибкий доступ к веб-службам (REST/SOAP) для веб-наращиваемых приложений и файлового доступа (API-интерфейс CIFS, или NFS, или устанавливаемой файловой системы, или Centera) для традиционных приложений.
- **Автоматизированная система управления.** EMC Atmos предоставляет возможности автоматической настройки, автоматического управления и автоматического восстановления, позволяет сократить время простоя и администрирования.
- **Многопользовательская среда.** EMC Atmos позволяет обслуживать в рамках одной и той же архитектуры сразу несколько приложений. Каждому приложению выделяется своя безопасная секция, и оно не может получить доступ к данным другого приложения. Принадлежность сразу нескольким приложениям является идеальным решением для провайдеров служб или крупных предприятий, желающих предоставить службы облачных вычислений нескольким потребителям или отделам, допуская при этом логическое и безопасное разделение в рамках единой инфраструктуры.
- **Гибкое администрирование.** EMC Atmos может управляться с использованием графического пользовательского интерфейса — graphical user interface (GUI) или интерфейса командной строки — command-line interface (CLI).

8.5.2. EMC VNX

VNX является предложением универсального хранилища от компании EMC. Массив хранения данных EMC VNX показан на рис. 8.11.

Системы хранения данных VNX включают следующие компоненты:

- *процессоры хранилищ* (SP-процессоры), которые поддерживают блочный ввод-вывод при доступе к хранилищу с использованием FC-, iSCSI- и FCoE-протоколов;
- *блейд-серверы X-Blades*, которые обращаются к данным по внутренней сети и обеспечивают доступ хостам с использованием NFS-, CIFS-, MPFS-, pNFS- и FTP-протоколов. Серверы X-Blades в каждом массиве могут масштабироваться и обеспечивают избыточность для гарантии отсутствия единой точки отказа;
- *станции управления*, которые предоставляют управляющие функции серверам X-Blades. Станции управления отвечают также за обработку отказа сервера X-Blade. Станция управления может быть дополнительно настроена на согласованную работу с вторичной станцией управления, обеспечивая тем самым избыточность системы управления в VNX-массиве;

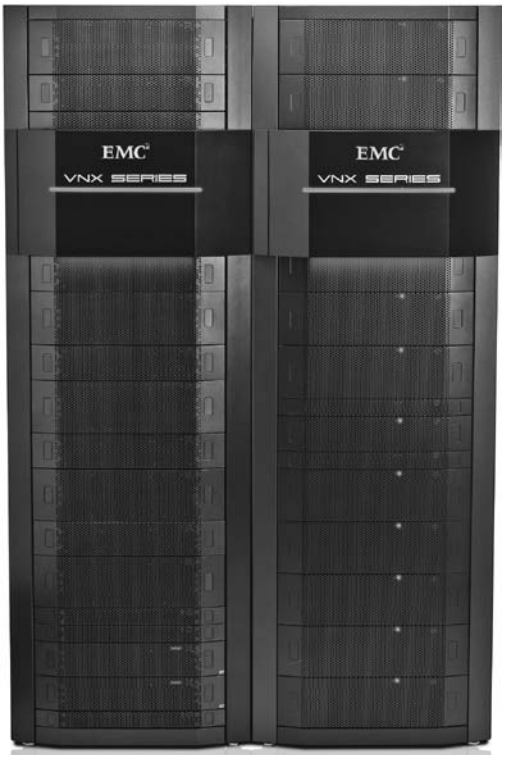


Рис. 8.11. Система хранения данных EMC VNX

- *резервное электропитание*, которое обеспечивает достаточное питание каждому процессору хранилища и первой полке дискового массива (DAE), чтобы гарантировать немедленное сохранение любых данных в области хранения в случае сбоя основного электропитания. Тем самым исключаются потери данных при записи;
- *полки дисковых массивов* — Disk-array enclosures (DAE-полки), в которых находятся накопители, используемые в массиве. Доступны полки разных размеров, каждая из которых может содержать 15, 25 или 60 накопителей. Для удовлетворения возрастающих потребностей могут добавляться дополнительные полки дисковых массивов.

8.5.3. EMC Centera

Система EMC Centera предлагается в трех разных моделях, отвечающих различным типам пользовательских запросов: EMC Centera Basic, EMC Centera Governance Edition и EMC Centera Compliance Edition Plus (CE+).

- **EMC Centera Basic** предоставляет все функции без обеспечения соблюдения сроков хранения данных.
- **EMC Centera Governance Edition** в добавление к функциям EMC Centera Basic предоставляет возможности сохранности, которые требуются организациям для управления цифровыми записями.
- **EMC Centera Compliance Edition Plus** предоставляет возможности, соответствующие широкому набору требований. Система CE+ предназначена для удовлетворения самых строгих требований бизнес-сред по электронным носителям данных, установленных регламентацией как Комиссии по ценным бумагам и биржам — Securities and Exchange Commission (SEC), так и других национальных или международных групп по выработке нормативов.

Архитектура EMC Centera

Centera-архитектура показана на рис. 8.12. Клиенты обращаются к Centera по локальной сети. Клиенты могут получать доступ к Centera только через сервер, на котором запущен Centera API (application programming interface — программный интерфейс приложения). Centera API отвечает за выполнение функций, позволяющих приложению сохранять и извлекать данные.

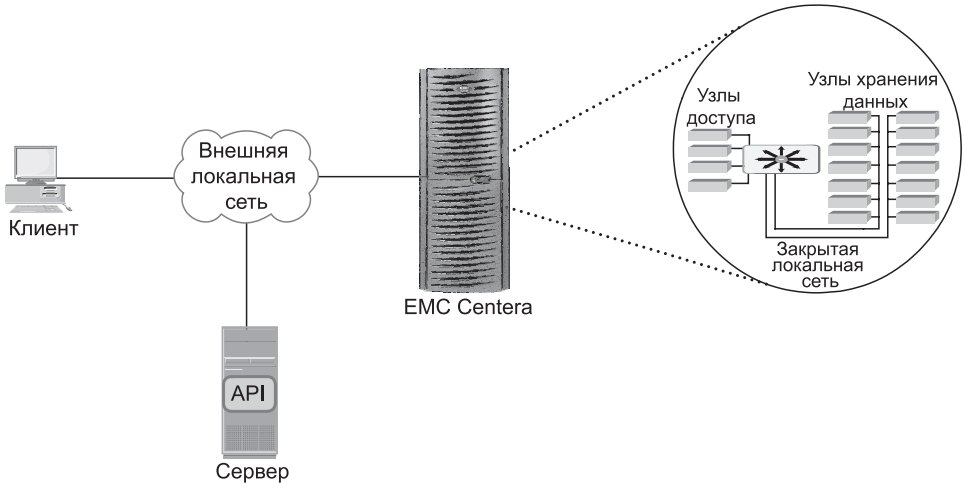


Рис. 8.12. Архитектура Centera

Архитектура Centera представляет собой *избыточный массив независимых узлов* — Redundant Array of Independent Nodes (RAIN). Этот массив содержит узлы хранения данных и узлы доступа, подключенные по сети в виде кластеров с использованием закрытой локальной сети. Внутренняя локальная сеть при обнаружении конфигурационных изменений, таких как добавление узлов хранения данных или узлов доступа, перенастраивается в автоматическом

режиме. Сервер приложений обращается к Centera по внешней локальной сети.

Узлы комплектуются недорогими дисковыми SATA-накопителями большой емкости. На этих узлах запускается CentraStar — операционная среда для Centera, предоставляющая свойства и функции, необходимые Centera-системе.

При установке узлов они настраиваются на «роль», определяющую предоставляемые узлом функциональные возможности. Узел может быть настроен в качестве узла хранения данных, узла доступа или узла с двойной ролью.

Узлы хранения данных хранят и защищают объекты данных. Иногда их называют *внутренними узлами*.

Узлы доступа предоставляют возможность подключения по внешней локальной сети. К имеющимся в кластере узлам хранения данных эти узлы устанавливают подключения по закрытой локальной сети. Количество узлов доступа определяется требуемой от кластера пропускной способностью. Если узел настроен исключительно как узел доступа, его дисковое пространство не может использоваться для хранения объектов данных. Запросы на сохранение и извлечение данных отправляются узлу доступа по внешней локальной сети.

Узлы с двойной ролью предоставляют возможности как узла хранения данных, так и узла доступа. Эта настройка встречается чаще, чем чистая настройка на роль узла доступа.

Резюме

Объектно-ориентированные системы хранения данных являются потенциальными решениями для хранения возрастающих объемов неструктурированных данных. Они также предоставляют решение для долговременного хранения данных с соблюдением определенных правил. Атрибуты объектов позволяют осуществлять автоматизированное управление данными на основе конкретной политики. Свойства OSD-устройств делают их также привлекательным решением для развертывания облачных систем. В этой главе были рассмотрены OSD-архитектура, ее компоненты, работа и контентно-адресуемое хранилище данных.

В этой главе также было рассмотрено универсальное хранилище данных, позволяющее осуществлять блочный, файловый и объектный доступ к данным в рамках единого решения. Это решение позволяет снизить затраты на приобретение оборудования, предоставляя при этом доступ к хранилищу различным приложениям. Были рассмотрены компоненты унифицированного хранилища и процессы доступа к данным из системы.

Современные системы хранения данных способны гарантировать высокую производительность, емкость и защищенность системы. Они имеют встроенную избыточность, исключающую любые повреждения при отказе одного из компонентов. Но ресурсы и данные по-прежнему уязвимы для

стихийных бедствий и других запланированных и внеплановых простоев, которые могут повлиять на доступность данных. В следующей главе рассматриваются вопросы обеспечения непрерывности бизнес-процессов и решения по аварийному восстановлению, гарантирующие высокий уровень доступности и непрерывности бизнес-операций.

УПРАЖНЕНИЯ

1. Опишите хранилище объектов и процесс их извлечения в OSD-системе.
2. Объясните процесс сохранения и извлечения данных для доступа на уровне блоков, файлов и объектов в унифицированной системе хранения.
3. Проведите исследование и подготовьте презентацию сценария, в котором объектно-ориентированное хранилище является более подходящим выбором, чем SAN и NAS.
4. Проведите исследование REST и SOAP и их реализаций.
5. При каких условиях унифицированное хранилище является подходящим вариантом для дата-центра? Обоснуйте свой ответ, сравнив преимущества, предлагаемые универсальным хранилищем, со свойствами традиционных решений для хранения данных.

Раздел

III

Резервное копирование, архивирование и репликация

В ЭТОМ РАЗДЕЛЕ

Глава 9. Введение в обеспечение непрерывности бизнес-процессов
Глава 10. Резервное копирование и архивирование
Глава 11. Локальная репликация
Глава 12. Удаленная репликация

Глава 9
Введение в обеспечение непрерывности бизнес-процессов

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Обеспечение непрерывности бизнес-процессов
- Доступность информации
- Аварийное восстановление
- Планирование обеспечения непрерывности бизнес-процессов
- Анализ факторов, влияющих на бизнес-процессы
- Программное обеспечение управления несколькими путями данных

В наши дни без непрерывного доступа к информации обеспечить должное проведение бизнес-операций невозможно. Цена недоступности информации возросла как никогда раньше, и даже час простоя в ключевых отраслях промышленности грозит миллионными убытками. Доступности информации могут угрожать, например, стихийные бедствия, внеплановые или запланированные события, результатом которых может стать отсутствие доступа к информации. Поэтому для ведения предпринимательской деятельности очень важно определить подходящую стратегию, позволяющую преодолеть неблагоприятные обстоятельства.

Определение и реализация таких стратегий и составляет основную суть обеспечения непрерывности бизнес-процессов.

Обеспечение непрерывности бизнес-процессов — business continuity (BC) представляет собой комплекс мероприятий в масштабе всей организации, включающий все виды деятельности (внутренние или внешние по отношению к информационным технологиям) и обязательный к выполнению предпринимателями в целях уменьшения влияния всех плановых и внеплановых простоев. Обеспечение непрерывности бизнес-процессов влечет за собой готовность к возникновению таких сбоев в работе системы, которые негативно сказываются на проведении бизнес-операций, соответствующее реагирование на них и восстановление работоспособности. Предполагается принятие профилактических мер, таких как анализ степени воздействия на бизнес-процессы, оценка рисков, развертывание оборудования, обеспечивающего

непрерывность бизнес-процессов (систем резервного копирования и репликации), а также проведение мер оперативного реагирования, таких как аварийное восстановление и перезапуск, которые должны предприниматься в случае сбоя. Цель обеспечения непрерывности бизнес-процессов состоит в том, чтобы добиться информационной доступности, необходимой для выполнения жизненно важных бизнес-операций.

В виртуализированной среде технологические решения в области обеспечения непрерывности бизнес-процессов нужны для защиты не только физических, но и виртуальных ресурсов. Виртуализация существенно упрощает реализацию BC-стратегий и претворение в жизнь соответствующих решений.

В данной главе описываются факторы, влияющие на доступность информации и последствия информационной недоступности. Также в ней рассматриваются ключевые параметры, влияющие на выработку любой BC-стратегии, и дорожная карта разработки эффективного плана обеспечения непрерывности бизнес-процессов.

9.1. Доступность информации

Под доступностью информации — Information availability (IA) понимается возможность инфраструктуры, занимающейся ее обработкой, функционировать в соответствии с возлагаемыми на нее бизнес-ожиданиями в течение времени, отведенного на ту или иную операцию. Доступность информации гарантирует, что люди (служащие, клиенты, поставщики, партнеры) смогут получить доступ к информации в любой необходимый момент времени. Доступность информации может быть описана в понятиях доступности, достоверности и своевременности.

- **Доступность.** Информация должна быть открыта для доступа в нужном месте и нужному пользователю.
- **Достоверность.** Информация должна быть достоверной и надлежащей во всех отношениях. Она должна быть именно такой, какой была сохранена, без каких-либо изменений или искажений.
- **Своевременность.** Определяет конкретный момент или период времени (конкретно указанные время суток, неделю, месяц и год), в течение которого информация должна быть доступна. Например, если интерактивный доступ к приложению требуется ежедневно с 8.00 до 22.00, то любые сбои вне этого временного интервала не должны считаться влияющими на своевременность предоставления данных.

9.1.1. Причины недоступности информации

Недоступность информации может стать результатом различных плановых и внеплановых событий. К запланированным простоям можно отнести установку, компоновку и обслуживание нового оборудования, обновления

программ или установку обновлений, создание резервных копий, восстановление приложений и данных, проведение работ на объекте (ремонт и строительство), а также обновление и перемещение программ из среды тестирования в эксплуатационную среду. К незапланированным простоям можно отнести сбои, вызванные ошибочными действиями персонала, повреждением баз данных, и сбои физических и виртуальных компонентов.

К событиям другого типа, которые могут стать причиной недоступности информации, можно отнести природные или техногенные катастрофы, такие как наводнения, пожары, землетрясения или химические заражения. Как показано на рис. 9.1, в основном простои бывают плановыми. Хотя они носят ожидаемый и запланированный характер, но так или иначе приводят к недоступности данных. По статистике, доля недоступности информации по причине непредвиденных обстоятельств составляет менее 1 %.

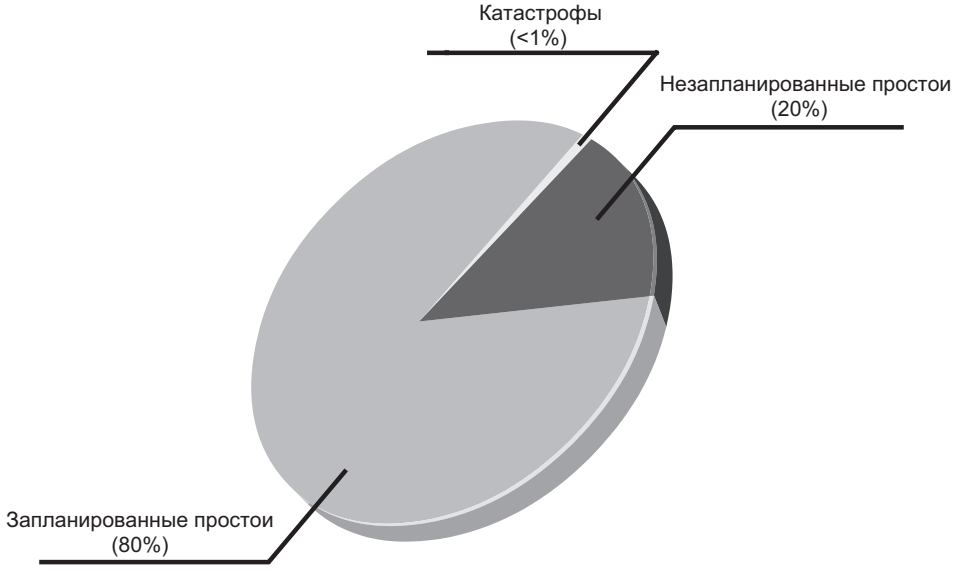


Рис. 9.1. Причины недоступности данных

9.1.2. Последствия вынужденного простоя

Недоступность данных, или вынужденный простой, может привести к потерям продуктивности и доходов, плохим финансовым показателям и нанесению урона репутации компании. Потери продуктивности уменьшают выработку на единицу труда, оборудования или капитала. Потери доходов включают в себя прямые потери, компенсационные выплаты, а также будущие потери доходов, оплату выставленных счетов за понесенные убытки и потерю инвестиций. Плохие финансовые показатели отрицательно влияют на признание дохода, движение наличности, процент скидки, гарантии

платежей, кредитные рейтинги и цены акций. Падение репутации может привести к потере доверия со стороны клиентов, поставщиков, финансовых рынков, банков и бизнес-партнеров. Другие возможные последствия простоя включают в себя стоимость аренды дополнительного оборудования, выплату за сверхурочную работу и оплату доставки оборудования.

Влияние простоя в деловом плане складывается из всех понесенных в результате данного нарушения режима работы потерь. Ключевую оценку при определении приемлемости BC-решений позволяет дать такой важный показатель, как средняя стоимость простоя в час. Она вычисляется следующим образом:

> Средняя стоимость простоя в час = средняя потеря продуктивности в час + средняя потеря дохода в час,

где *Потеря продуктивности в час* = Общая сумма зарплат и пособий всех работников за неделю/Среднее количество рабочих часов в неделю, *Средняя потеря дохода в час* = Общий доход организации за неделю/Среднее количество часов в неделю, когда организация открыта для бизнес-операций.

Средняя стоимость простоя в час может также включать в себя оценки прогнозируемых потерь дохода от других последствий, таких как ухудшение репутации и дополнительные затраты на восстановление работоспособности системы.

9.1.2. Оценка доступности информации

Доступность информации зависит от доступности физических и виртуальных компонентов дата-центра. Сбои в работе этих компонентов могут сделать информацию недоступной. Под сбоем понимается потеря компонентом возможности выполнения какой-либо обязательной функции. Эта возможность может быть восстановлена путем внешнего, устраняющего сбой воздействия, такого как ручной перезапуск, ремонт или замена отказавшего компонента (или компонентов). Ремонт включает в себя восстановление состояния компонента, позволяющего ему выполнять требуемую функцию. При упреждающем анализе рисков, выполняемом в качестве составной части процесса BC-планирования, оцениваются частота отказов компонентов и среднее время ремонта, которые определяются в понятиях среднего времени безотказной работы — mean time between failure (MTBF) и среднего времени восстановления — mean time to repair (MTTR).

- **Среднее время безотказной работы (MTBF)** — средний период времени, в течение которого система или компонент может выполнять свои функции без сбоев. Служит показателем надежности системы или компонента и чаще всего выражается в часах.
- **Среднее время восстановления (MTTR)** — средний период времени, в течение которого можно восстановить отказавший компонент. При вычислении MTTR предполагается, что неисправность, приведшая к сбою, обнаружена и необходимые детали и персонал доступны. Под

неисправностью понимается физический дефект на компонентном уровне, который может привести к недоступности данных. MTTR включает в себя время, необходимое для того, чтобы произвести следующие действия: обнаружить неисправность, привлечь ремонтную бригаду, провести диагностику неисправности, приобрести запасные части, провести ремонт, тестирование и восстановление данных. На рис. 9.2 проиллюстрированы различные показатели доступности информации, представляющие среднее время безотказной работы и время простоя.

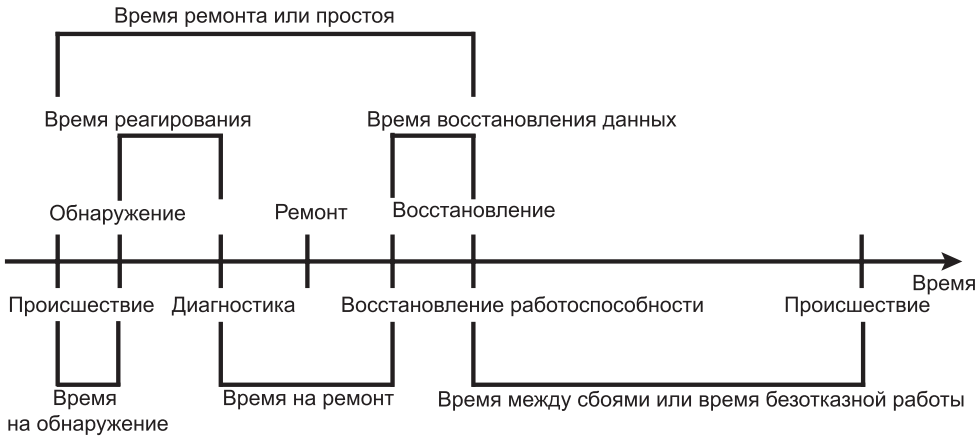


Рис. 9.2. Показатели доступности информации

Период доступности информации (IA) — это время, в течение которого система в состоянии выполнять возлагаемые на нее функции по предназначению. Его можно выразить в понятиях безотказной работы системы и ее простоя, оценить в процентном показателе безотказного периода работы системы:

IA = Период безотказной работы/(Период безотказной работы +
+ Период простоя),

где Период безотказной работы — это время, в течение которого система находится в доступном состоянии, а время недоступности системы называется периодом простоя.

В понятиях MTBF и MTTR период доступности информации может быть описан следующим образом:

IA = MTBF/(MTBF + MTTR).

Значение периода безотказной работы в течение года основано на конкретных требованиях к своевременности предоставления рассматриваемой службы. Его вычисление приводит к определению количества девяток, фигурирующих в показателях доступности. В табл. 9.1 приведены средние

значения периода простоя, допустимые для службы, при которых можно достичь конкретного уровня девяток в показателе доступности.

Например, если про службу говорят, что ее доступность на уровне пяти девяток, значит, она доступна в 99,999 % запланированного времени в год (24 · 365).

Таблица 9.1. Процент доступности и допустимые значения периода простоя

ВРЕМЯ БЕЗОТКАЗНОЙ РАБОТЫ, %	ВРЕМЯ ПРОСТОЯ, %	ВРЕМЯ ПРОСТОЯ ЗА ГОД	ВРЕМЯ ПРОСТОЯ ЗА НЕДЕЛЮ
98	2	7,3 дня	3 часа 22 минуты
99	1	3,65 дня	1 час 41 минута
99,8	0,2	17 часов 31 минута	20 минут 10 секунд
99,9	0,1	8 часов 45 минут	10 минут 5 секунд
99,99	0,01	52,5 минуты	1 минута
99,999	0,001	5,25 минуты	6 секунд
99,9999	0,0001	31,5 секунды	0,6 секунды

9.2. Терминология обеспечения непрерывности бизнес-процессов

В данном разделе вводятся и разъясняются общие понятия, связанные с операциями обеспечения непрерывности бизнес-процессов и использующиеся в следующих главах для объяснения более сложных понятий.

- **Аварийное восстановление (Disaster recovery)** — скоординированный процесс восстановления после аварии тех систем, данных и инфраструктуры, которые требуются для поддержки текущих бизнес-операций. В него входят восстановление предыдущей копии данных и применение к ней журнальных записей или других необходимых средств для приведения ее в известное состояние согласованности. После завершения всех восстановительных мероприятий данные проверяются для получения гарантии их приемлемости.
- **Аварийный перезапуск (Disaster restart)** — процесс перезапуска бизнес-операций с помощью зеркальных согласованных копий данных и приложений.
- **Целевая точка восстановления — Recovery-Point Objective (RPO)** — момент времени, к которому система и данные должны быть восстановлены после вынужденного простоя. Это понятие определяет и тот

объем данных, которое предприятие может себе позволить потерять. Высокий показатель RPO означает высокую устойчивость бизнеса к информационным потерям. На основании RPO в организации определяется периодичность создания резервных копий или точных копий данных. К примеру, если показатель RPO составляет 6 часов, то резервная или точная копия должна создаваться как минимум один раз в 6 часов. На рис. 9.2, *а* изображены различные показатели RPO и соответствующие наилучшие стратегии восстановления. На основе установленного показателя RPO организация может планировать внедрение соответствующей технологии обеспечения непрерывности бизнес-процессов. К примеру:

- **RPO порядка 24 часов** — в данном случае резервные копии создаются в сторонней библиотеке на магнитных лентах каждую полночь. Соответствующая стратегия восстановления заключается в восстановлении данных из последней резервной записи на магнитной ленте;
- **RPO порядка 1 часа** — записи журнала базы данных отправляются в удаленное хранилище каждый час. Соответствующая стратегия восстановления заключается в восстановлении базы на момент последней отправки журнальной записи;
- **RPO порядка нескольких минут** — ведется асинхронное зеркалирование данных в удаленном месте.
- **RPO, стремящийся к нулю**, — ведется синхронное зеркалирование данных в удаленном месте.

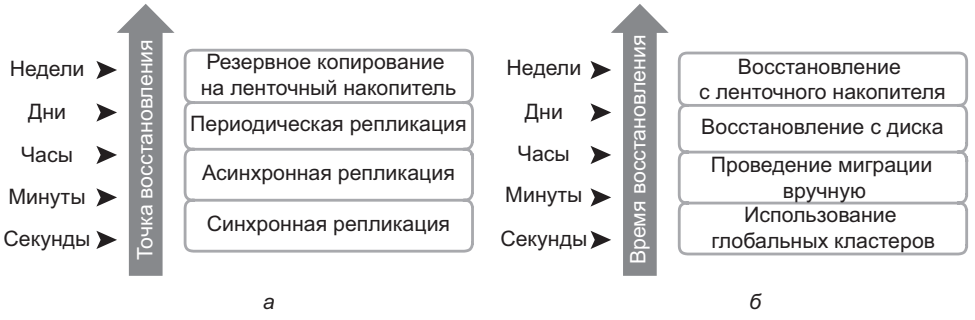


Рис. 9.3. Стратегии для достижения целей RPO и RTO: *а* — целевая точка восстановления; *б* — целевое время восстановления

- **Целевое время восстановления — Recovery-Time Objective (RTO)** — максимальное время, отводимое на восстановление систем и приложений после вынужденного простоя. Это понятие определяет тот срок, который организация может выдержать в состоянии вынужденного бездействия. Предприятия могут оптимизировать планы аварийного

восстановления после определения показателей RTO для каждой отдельно взятой системы. К примеру, если показатель RTO составляет 2 часа, то для резервного копирования следует использовать диски. А при RTO, равном одной неделе, для резервного копирования вполне могут сгодиться и накопители на магнитных лентах. Далее перечислены некоторые примеры RTO и соответствующие стратегии восстановления (см. также рис. 9.2, *б*):

- **RTO порядка 72 часов**: восстановление с ленточных накопителей, находящихся в «холодном» резервном состоянии;
- **RTO порядка 12 часов**: восстановление с ленточных накопителей, находящихся в горячем резервном состоянии;
- **RTO порядка нескольких часов**: использование хранилища модели data vault, находящегося в горячем резервном состоянии;
- **RTO порядка нескольких секунд**: использование промышленных кластерных серверов с двунаправленным зеркальным копированием, позволяющим приложениям одновременно работать в обоих местах.

- **Хранилище модели DataVault** — репозиторий, находящийся в удаленном месте, куда данные могут периодически или постоянно копироваться (либо на ленточные накопители, либо на диски), то есть речь идет о копии, которая всегда находится в другом месте.
- **Горячий резерв (Hot site)** — резервная площадка, куда можно перенести операции в случае возникновения аварийной ситуации. Там находятся все необходимое оборудование, операционная система, приложение и сетевая поддержка, позволяющая выполнять бизнес-операции, причем оборудование постоянно доступно и готово к работе.
- **Холодный резерв (Cold site)** — резервная площадка, куда можно перенести операции в случае возникновения аварийной ситуации, имеющая минимально допустимую IT-инфраструктуру и соответствующее оборудование, но в неактивном состоянии.
- **Кластерное объединение серверов** — группа серверов и других необходимых ресурсов, объединенных для работы в единой системе. Кластеры могут обеспечить высокую доступность данных и сбалансированность рабочей нагрузки. Как правило, в кластерах, обеспечивающих отказоустойчивость системы, на одном сервере работает приложение и проводится обновление данных, а другой сервер находится в режиме ожидания, чтобы полностью взять на себя всю работу, как только в этом возникнет необходимость. В более сложных кластерах на нескольких серверах могут содержаться данные, а один сервер, как правило, будет находиться в режиме ожидания. Кластерное объединение серверов обеспечивает сбалансированность рабочей нагрузки путем равномерного распределения нагрузки приложения между несколькими серверами кластера.

9.3. Жизненный цикл планирования обеспечения непрерывности бизнес-процессов

К планированию обеспечения непрерывности бизнес-процессов (BC), как и к любому другому виду планирования, нужно относиться серьезно. Сегодня на разработку и выполнение BC-планов организации выделяют специальные ресурсы. Для обеспечения непрерывности ведения бизнеса можно определить цикл мероприятий, проводимых от выработки концепции до реализации BC-плана. Жизненный цикл BC-планирования включает в себя пять этапов (см. рис. 9.4).

1. Определение целей.
2. Проведение анализа.
3. Проектирование и разработка.
4. Реализация.
5. Обучение персонала, тестирование, оценка и поддержка актуальности плана.

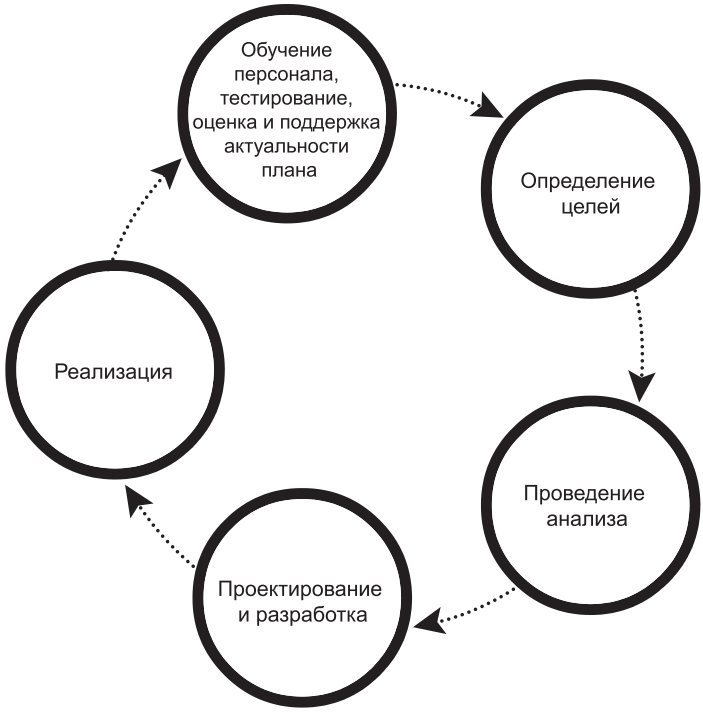


Рис. 9.4. Жизненный цикл BC-планирования

На каждом этапе жизненного цикла BC-планирования проводится ряд следующих основных мероприятий.

- Определение целей:
  - Определение требований к обеспечению BC.
  - Оценка масштабов работы и бюджета для достижения требуемых результатов.
  - Определение состава команды разработчиков BC-плана, в которую включаются как свои, так и сторонние специалисты всех профилей бизнеса.
  - Выработка политик обеспечения BC.
- Проведение анализа:
  - Сбор информации о совокупности параметров данных, бизнес-процессах, поддержке инфраструктуры, зависимостях и частоте использования бизнес-инфраструктуры.
  - Анализ факторов, оказывающих влияние на бизнес-процессы, — Business Impact Analysis (BIA).
  - Определение важнейших бизнес-процессов и назначение приоритетов при восстановлении данных.
  - Анализ рисков для наиболее важных функций и выработка стратегий смягчения последствий.
  - Проведение анализа стоимости и эффективности доступных решений на основе стратегии смягчения последствий.
  - Оценка возможных вариантов.
- Проектирование и разработка:
  - Определение структуры команд и распределение ролей и ответственности. Например, для выработки мер реагирования на чрезвычайные ситуации, для оценки ущерба и определения его характера нужна одна команда, а для выработки мер по восстановлению инфраструктуры и приложения — совершенно другая.
  - Выработка стратегий защиты данных и инфраструктуры.
  - Выработка решений на случаи возникновения непредвиденных ситуаций.
  - Выработка мер реагирования на возникновение чрезвычайных ситуаций.
  - Создание подробных описаний процедур восстановления и перезапуска.
- Реализация:
  - Реализация процедур риск-менеджмента и смягчения последствий, включающих резервное копирование, репликацию и управление ресурсами.

- Подготовка мест для восстановления при аварийных ситуациях, которые могли бы использоваться при аварии основного дата-центра.
- Реализация избыточности для каждого ресурса дата-центра, чтобы не было какого-либо одного компонента, приводящего к отказу всей системы.

■ Обучение персонала, тестирование, оценка и эксплуатация:

- Обучение персонала, ответственного за резервное копирование и репликацию особо важных для бизнеса данных, на регулярной основе или при внесении изменений в BC-план.
- Обучение персонала экстренным действиям в случае объявления чрезвычайных ситуаций.
- Обучение команды ликвидации последствий процедурам восстановления на основе сценариев действий в чрезвычайных ситуациях.
- Оценка ущерба и пересмотр планов восстановления.
- Регулярное тестирование BC-плана для оценки его эффективности и выявления недочетов.
- Оценка отчетов о производительности и выявление ограничивающих ее факторов.
- Обновление BC-планов и процедур восстановления и перезапуска с целью отражения в них изменений, регулярно происходящих в дата-центре.

9.4. Анализ сбоев

Анализ сбоев включает оценку компонентов как физической, так и виртуальной инфраструктуры с целью определения систем, уязвимых при наличии какого-либо одного компонента, приводящего к отказу всей системы, и недостаточной реализации механизмов отказоустойчивости.

9.4.1. Единая точка отказа

Единая точка отказа (single point of failure) представляет собой один из компонентов, сбой которого может привести к потере доступности всей системы или IT-службы. На рис. 9.5 изображена система, в которой приложение, запущенное на виртуальной машине, предоставляет интерфейс клиенту и выполняет операции ввода-вывода. Клиент подключен к серверу по IP-сети, а сервер подключен к массиву хранения данных по FC-каналу.

В конфигурации, где для обеспечения доступности данных каждый компонент должен работать должным образом, сбой одного из физических или

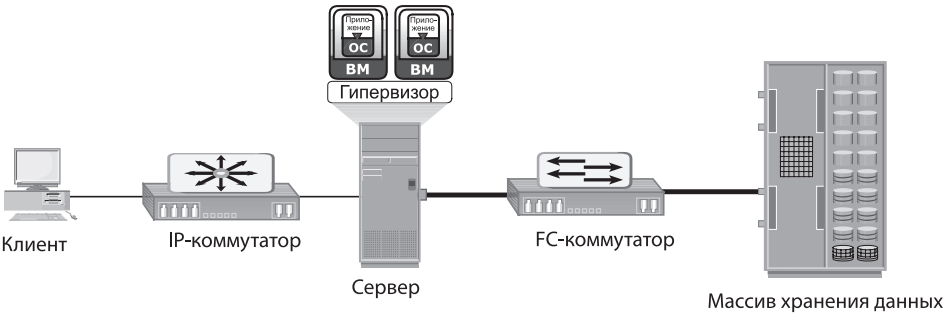


Рис. 9.5. Единая точка отказа

виртуальных компонентов приводит к недоступности приложения. Этот сбой приводит к нарушению бизнес-операций. Например, сбой гипервизора может повлиять на работу всех запущенных на его основе виртуальных машин и виртуальной сети. В конфигурации, показанной на рис. 9.5, можно обнаружить сразу несколько уязвимых мест. Виртуальная машина, гипервизор, адаптер главной шины или сетевая карта на сервере, физический сервер, IP-сеть, FC-коммутатор, порты массива хранения данных или даже сам массив хранения — все это может рассматриваться как потенциальные единые точки отказа.

9.4.2. Решение проблемы единых точек отказа

Для решения проблемы единых точек отказа системы проектируются с избыточностью компонентов, чтобы сбой всей системы мог произойти только при отказе всей группы компонентов. Тем самым гарантируется, что сбой одного компонента не повлияет на доступность данных. Для обеспечения отказоустойчивости с целью предоставления гарантии бесперебойного доступа к данным дата-центры придерживаются весьма строгих правил. Для устранения уязвимых мест проводится тщательный анализ системы. В примере на рис. 9.6 показаны все улучшения инфраструктуры, призванные решить проблему единых точек отказа.

- Включение в состав сервера избыточных HBA-адаптеров позволяет избежать общего отказа в случае сбоя одного из HBA-адаптеров.
- Включение в состав сервера группы NIC-адаптеров позволяет защититься от сбоя одного из сетевых адаптеров. Такая конфигурация позволяет составить группу из двух и более физических сетевых адаптеров и рассматривать ее как одно логическое устройство. При наличии группы NIC-адаптеров сбой одного входящего в нее физического адаптера или отсоединение от него кабеля приведет к перенаправлению трафика на другой сетевой адаптер группы. Таким образом, создание группы сетевых адаптеров приведет к устранению уязвимого места, в качестве которого рассматривался один адаптер.

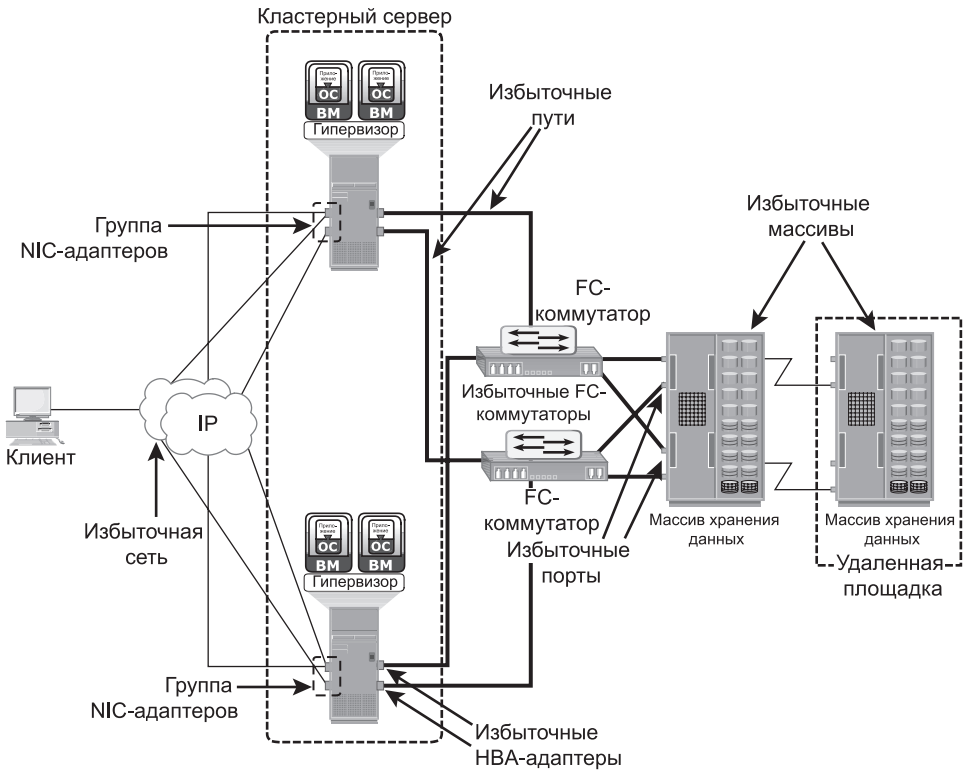


Рис. 9.6. Решение проблемы уязвимых мест

- Включение в общую конфигурацию избыточных коммутаторов на случай сбоя коммутатора.
- Включение в состав массива хранения данных нескольких портов для смягчения последствий отказа одного из них.
- Включение в общую конфигурацию RAID-массива и горячее резервирование оборудования для обеспечения непрерывности функционирования в случае отказа диска.
- Использование дополнительного массива хранения данных в удаленном месте с целью уменьшения негативных последствий в случае отказа локального оборудования.
- Создание в качестве механизма обеспечения отказоустойчивости серверного (или компьютерного) кластера, посредством которого на двух и более серверах кластера осуществляется доступ к одному и тому же набору томов данных. Серверы, входящие в кластер, обмениваются *сообщениями о состоянии работоспособности*, чтобы информировать друг друга о своем общем состоянии. При сбое одного из серверов

или гипервизоров всю рабочую нагрузку может взять на себя другой сервер или гипервизор.

- Создание механизма отказоустойчивости виртуальных машин, обеспечивающего непрерывность бизнес-процессов в случае отказа сервера. Согласно данной технологии на другом сервере создаются дубликаты каждой виртуальной машины, чтобы при обнаружении сбоя виртуальной машины для его обхода можно было воспользоваться дубликатом этой машины. Чтобы гарантировать успешный обход сбоя, две виртуальные машины поддерживаются в синхронизированном друг с другом состоянии.

9.4.3. Программное обеспечение управления несколькими путями

Конфигурация с использованием нескольких путей повышает доступность данных, так как позволяет обходить сбои. Если серверы настроены на один путь ввода-вывода, то в случае его недоступности обратиться к данным будет невозможно. Использование избыточных путей к данным исключает для пути вероятность стать единой точкой отказа. Наличие нескольких путей к данным также повышает производительность ввода-вывода за счет балансировки загруженности путей и повышения коэффициентов использования сервера, хранилища и путей данных.

Но на практике одной лишь настройкой серверов на несколько путей проблему не решить. Даже при наличии нескольких путей при отказе одного из них ввод-вывод не будет перенаправлен до тех пор, пока система не обнаружит наличие альтернативного пути. Обнаружить альтернативный путь ввода-вывода данных и воспользоваться им позволяет программное обеспечение управления несколькими путями. Эта программное обеспечение занимается также балансировкой нагрузки, направляя запросы на ввод-вывод по всем доступным и активным путям.

Программное обеспечение управления несколькими путями обладает интеллектом, позволяющим управлять путями обмена данными с устройством, отправляя запросы на ввод-вывод по оптимальному пути с учетом определенной для устройства сбалансированности нагрузки и политики обхода сбоев. Перед принятием решения относительно пути, по которому будут отправляться запросы на ввод-вывод, это программное обеспечение учитывает его загруженность и доступность. При сбое канала на пути доступа к устройству оно автоматически перенаправляет запросы на ввод-вывод по альтернативному пути.

В виртуальной среде организация нескольких путей возможна либо с использованием средств, встроенных в гипервизор, либо посредством запуска добавляемого к гипервизору программного модуля сторонних производителей.

9.5. Анализ факторов, влияющих на бизнес-процессы

Анализ факторов, влияющих на бизнес-процессы, — business impact analysis (BIA), определяет, какие составляющие бизнеса, операции и процессы играют важную роль в обеспечении его безотказного ведения. При этом оцениваются финансовые, оперативные и служебные последствия сбоя основных бизнес-процессов. Производится оценка отдельных функциональных областей на предмет определения устойчивости инфраструктуры при решении задач обеспечения доступности информации. В результате проведения BIA-анализа составляется отчет с подробным описанием возможных происшествий и их влияния на бизнес-процессы. Влияние может быть оценено либо в денежном выражении, либо в виде продолжительности вынужденного простоя. Учитывая потенциальные последствия простоя, предприятия могут определить приоритеты и принять контрмеры для снижения вероятности возникновения таких происшествий. Все это подробно описывается в плане обеспечения непрерывности бизнес-процессов. BIA-анализ включает в себя следующую совокупность задач.

- Определение бизнес-областей.
- Определение для каждой бизнес-области основных бизнес-процессов, крайне необходимых для ее функционирования.
- Определение свойств бизнес-процессов в понятиях приложений, баз данных и требований к оборудованию и программному обеспечению.
- Оценка убытков от сбоя для каждого бизнес-процесса.
- Вычисление максимально допустимого времени простоя и определение RTO- и RPO-показателей для каждого бизнес-процесса.
- Определение минимального количества ресурсов, необходимых для функционирования бизнес-процессов.
- Разработка стратегий восстановления и определение стоимости их реализации.
- Оптимизация стратегий резервного копирования и восстановления бизнеса в соответствии с бизнес-приоритетами.
- Анализ текущего состояния готовности системы обеспечения непрерывности бизнес-процессов и оптимизация будущего BC-планирования.

9.6. Технологические решения по обеспечению непрерывности бизнес-процессов

Следующим важным шагом после анализа влияния простоев на бизнес-процессы является разработка соответствующих решений для восстановления после сбоя. В любой из перечисленных ниже стратегий обеспечивается создание одной или нескольких копий данных, что позволяет при необходимости

восстановить данные или перезапустить бизнес-операции с использованием альтернативной копии.

- **Резервное копирование.** Является наиболее распространенным методом обеспечения доступности данных. Частота создания резервных копий определяется на основе показателей RPO, RTO и периодичности изменения данных.
- **Локальная репликация.** Репликация данных может проводиться в отдельном месте в том же самом массиве хранения данных. Создаваемая при этом точная копия данных независимо используется для других бизнес-операций. Она также может быть использована для операций восстановления при повреждении данных.
- **Удаленная репликация.** Репликация данных, находящихся в массиве хранения, может быть произведена в другой массив хранения данных, находящийся на удаленной площадке. Если в результате возникновения чрезвычайной ситуации массив хранения данных будет утрачен, бизнес-операции могут быть запущены с использованием удаленного массива хранения.

9.7. Практическая реализация концепции: EMC PowerPath

EMC PowerPath — это размещаемая на хост-машине программа направления данных по нескольким путям, обеспечивающая обход сбойного пути и осуществляющая балансировку нагрузки для SAN-среды. PowerPath располагается между операционной системой и драйверами устройств. Программа EMC PowerPath/VE позволяет оптимизировать работу виртуальной среды за счет применения возможностей PowerPath управления несколькими путями. Новые сведения об этой программе можно найти на сайте www.emc.com.

9.7.1. Свойства PowerPath

PowerPath обладает следующими свойствами.

- **Динамическая настройка путей и управление ими.** PowerPath предоставляет гибкие способы определения путей, ведущих к устройству, в качестве активных и в качестве резервных. Резервные пути используются в случае отказа всех активных путей к логическому устройству. Пути могут динамически добавляться и удаляться с установкой для них состояния «активный» и «резервный».
- **Динамическая балансировка нагрузки для нескольких путей.** PowerPath осуществляет интеллектуальное распределение запросов на ввод-вывод между всеми доступными путями к логическому устройству хранения данных. Тем самым сокращается количество узких мест и повышается производительность приложений.

- **Автоматический обход сбойных путей.** В случае отказа пути PowerPath мгновенно переключается на альтернативный путь, не прерывая работу приложений. PowerPath перераспределяет запросы на ввод-вывод по наиболее подходящим доступным путям, добиваясь оптимальной производительности хоста.
- **Упреждающее тестирование путей и их автоматическое восстановление.** Для упреждающего тестирования отказавших и восстановленных путей используются функции автозондирования и автовосстановления соответственно. Перед отправкой приложением запроса на ввод-вывод функция *автозондирования* периодически проверяет все пути на предмет наличия отказавших. Этот процесс позволяет программному обеспечению PowerPath заранее закрывать отказавшие пути, чтобы приложение не сталкивалось с истечением времени ожидания при отправке запроса на ввод-вывод по отказавшему пути. Функция *автовосстановления* запускается каждые пять минут и проверяет каждый сбойный или закрытый путь на предмет его восстановления.
- **Поддержка кластеров.** Развертывание PowerPath в кластере серверов позволяет избавиться от вызова процедуры обхода того сервера, отказ которого обусловлен сбоем пути.

9.7.2. Динамическая балансировка нагрузки

В тех средах, где рабочая нагрузка, связанная с вводом-выводом данных, не сбалансирована, PowerPath способствует существенному повышению производительности. Для каждого запроса на ввод-вывод драйвер фильтра PowerPath выбирает путь на основе политики сбалансированности нагрузки и настроек обхода сбоев, касающихся логического устройства хранения данных. Драйвер обнаруживает все доступные пути к устройству и выстраивает для устройств таблицу путей, называемую набором путей к томам. PowerPath придерживается конкретных политик сбалансированности нагрузки, определенных пользователем, к числу которых относятся:

- **политика применения алгоритма циклического перебора** — запросы на ввод-вывод направляются на каждый свободный путь по очереди;
- **политика наименьшего количества запросов** — запросы на ввод-вывод направляются по пути с наименьшим количеством запросов в очереди независимо от общего количества блоков ввода-вывода;
- **политика наименьшего количества блоков** — запросы на ввод-вывод направляются по пути с наименьшим количеством блоков ввода-вывода в очереди независимо от имеющегося числа запросов;
- **политика приоритетов** — запросы на ввод-вывод равномерно распределяются по нескольким путям на основе сочетания приоритетов операций чтения, записи, определенных пользователями устройств или приложений.

Работа системы ввода-вывода при отсутствии PowerPath

На рис. 9.7 показаны операции ввода-вывода в системе хранения данных при отсутствии PowerPath. Приложения, запущенные на хост-машине, располагают четырьмя путями, ведущими к массиву хранения данных. В данном примере показывается, как каналы связи, использующиеся для ввода-вывода, находятся в отсутствие PowerPath в разбалансированном состоянии. На два пути выпадает высокий уровень трафика ввода-вывода, и они сильно загружены, в то время как остальные два пути загружены меньше. В результате приложения не могут выйти на оптимальный уровень производительности.

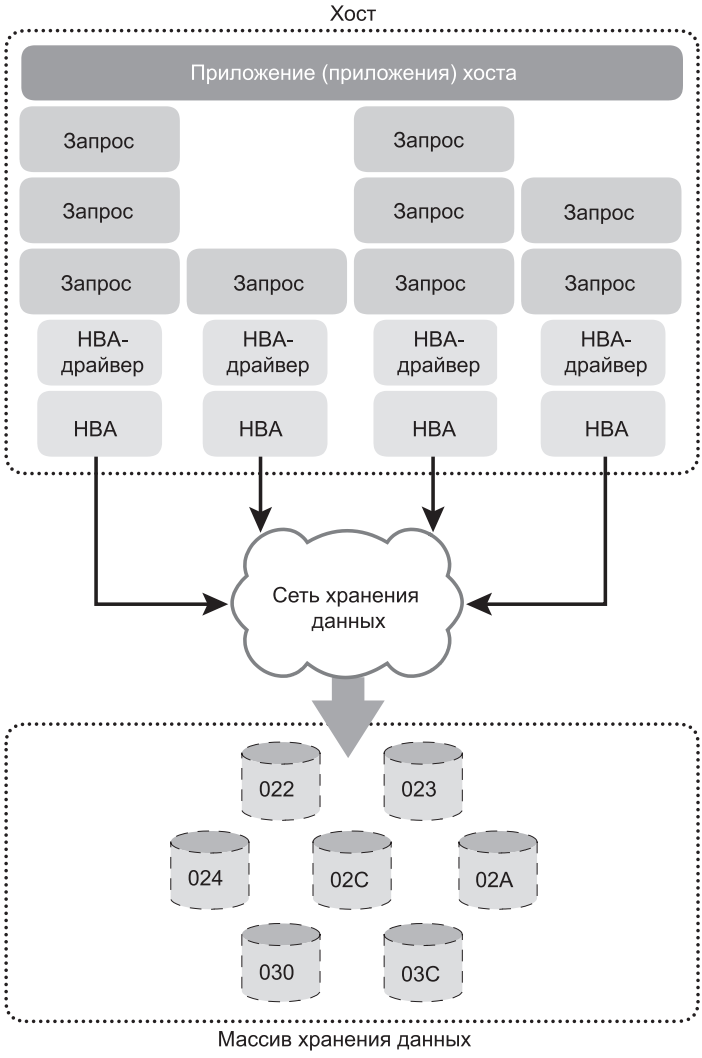


Рис. 9.7. Ввод-вывод при отсутствии PowerPath

Работа системы ввода-вывода при наличии PowerPath

На рис. 9.8 показана работа системы ввода-вывода в хранилище данных при наличии PowerPath. Программа PowerPath обеспечивает равномерное распределение запросов на ввод-вывод между всеми четырьмя путями в соответствии с выбранным алгоритмом балансировки. В результате приложения могут эффективнее использовать все пути, что приводит к повышению их производительности.

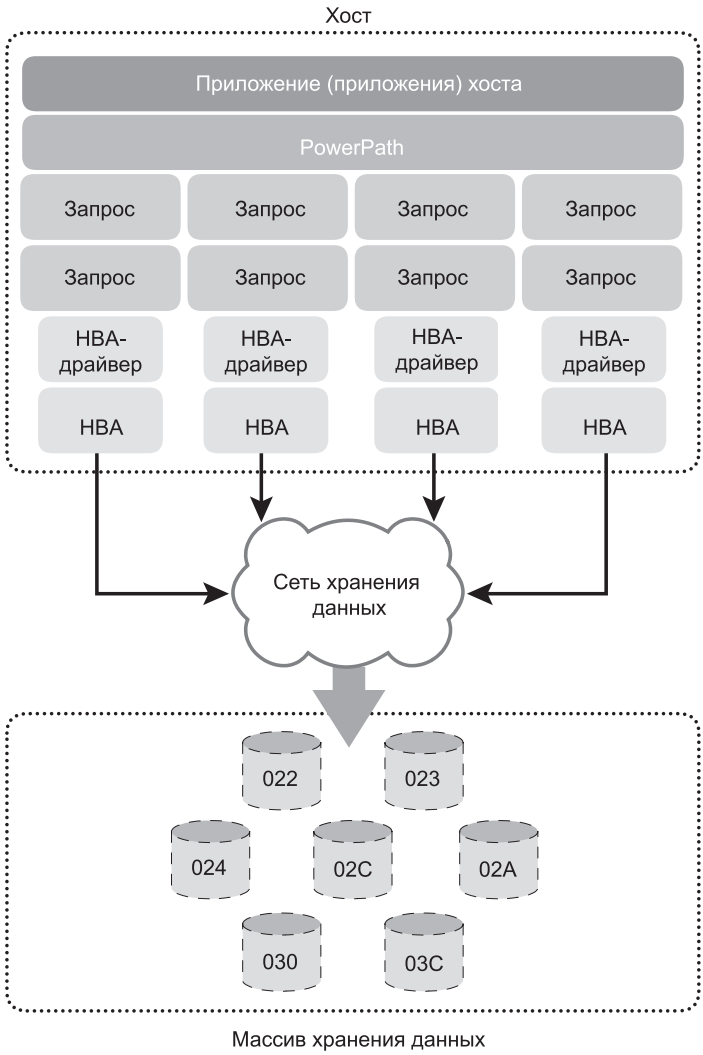


Рис. 9.8. Ввод-вывод при наличии PowerPath

9.7.3. Автоматический обход сбойных путей

В следующих двух примерах показывается, как PowerPath выполняет операции обхода пути в случае сбоя, произошедшего в конфигурациях массивов типа «активный — активный» и «активный — пассивный».

Отказ пути при отсутствии PowerPath

На рис. 9.9 показан сценарий, развивающийся при отсутствии PowerPath. Утрата пути (сбой на нем показан крестиком) из-за наличия уязвимого места, например при потере HBA-адаптера, утрате массивом хранения данных интерфейсного соединения, отказе порта коммутатора или разрыве кабеля, может привести к вынужденному простою одного или нескольких приложений, использующих данный путь.

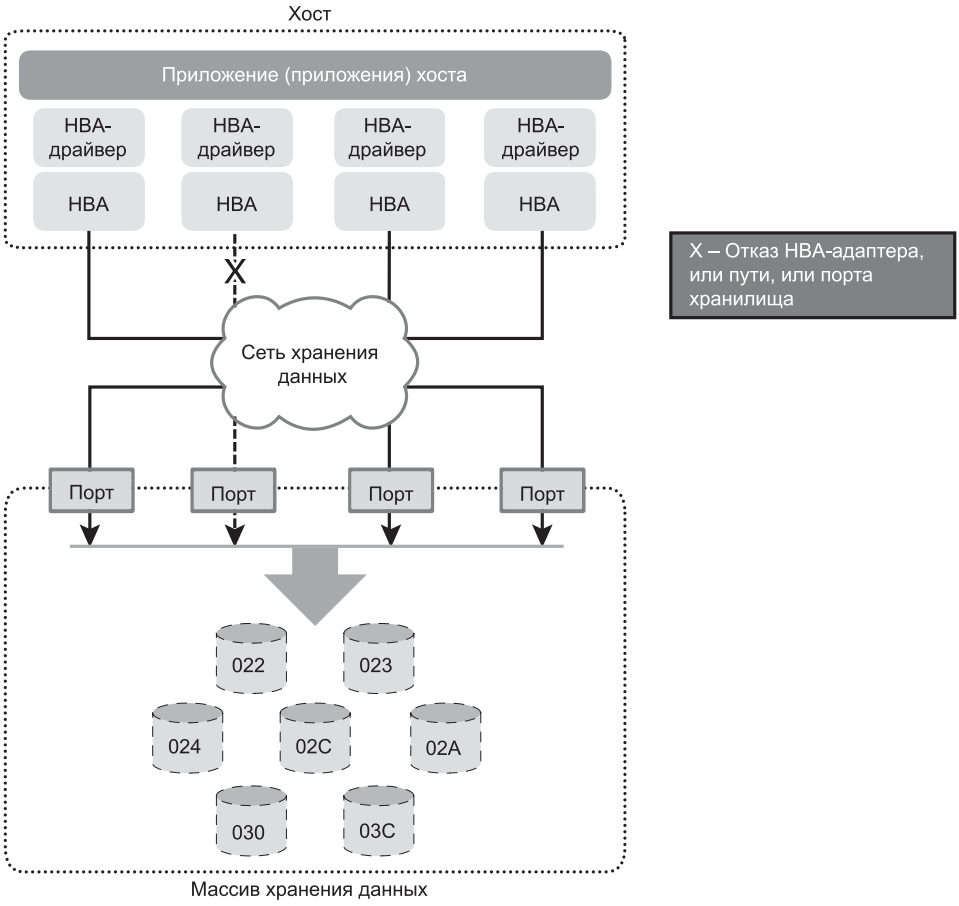


Рис. 9.9. Отказ пути при отсутствии PowerPath

Отказ пути при наличии PowerPath: конфигурация «активный — активный»

На рис. 9.10 показана среда системы хранения данных, в которой приложение для осуществления операций ввода-вывода использует PowerPath, имея конфигурацию хранилища типа «активный — активный». Если в массиве хранения данных с конфигурацией «активный — активный» есть несколько путей к логическому устройству, то все они находятся в активном состоянии и предоставляют доступ к этому устройству. Если путь к устройству дает сбой, PowerPath перенаправляет ввод-вывод приложения по альтернативному активному пути, предотвращая тем самым вынужденный простой приложения.

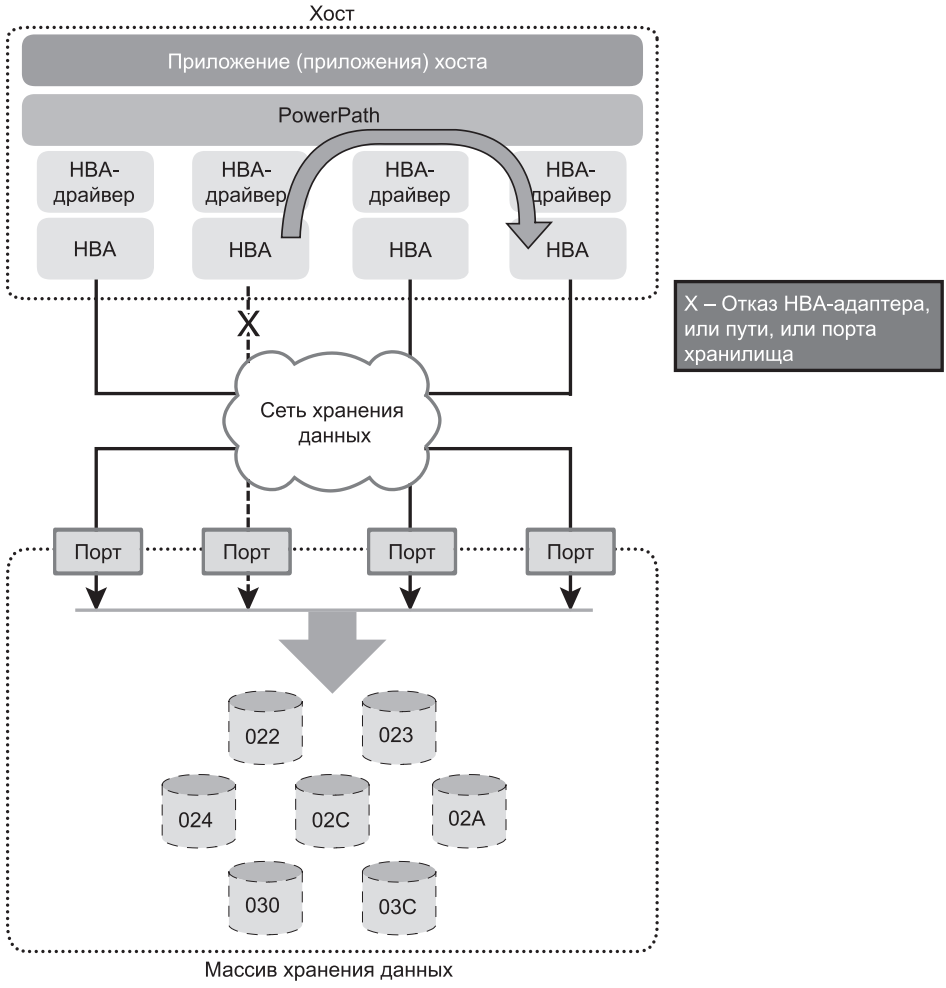


Рис. 9.10. Обход сбоя пути с использованием PowerPath для массива с конфигурацией «активный — активный»

Отказ пути при наличии PowerPath: конфигурация «активный — пассивный»

На рис. 9.11 показан сценарий, при котором логическое устройство назначено процессору хранилища B (SP B) и поэтому все запросы на ввод-вывод направляются к устройству по пути, проходящему через процессор SP B. Логическое устройство может также быть доступно через процессор SP A, но только после того как станет недоступен процессор SP B и устройство будет переназначено процессору SP A.

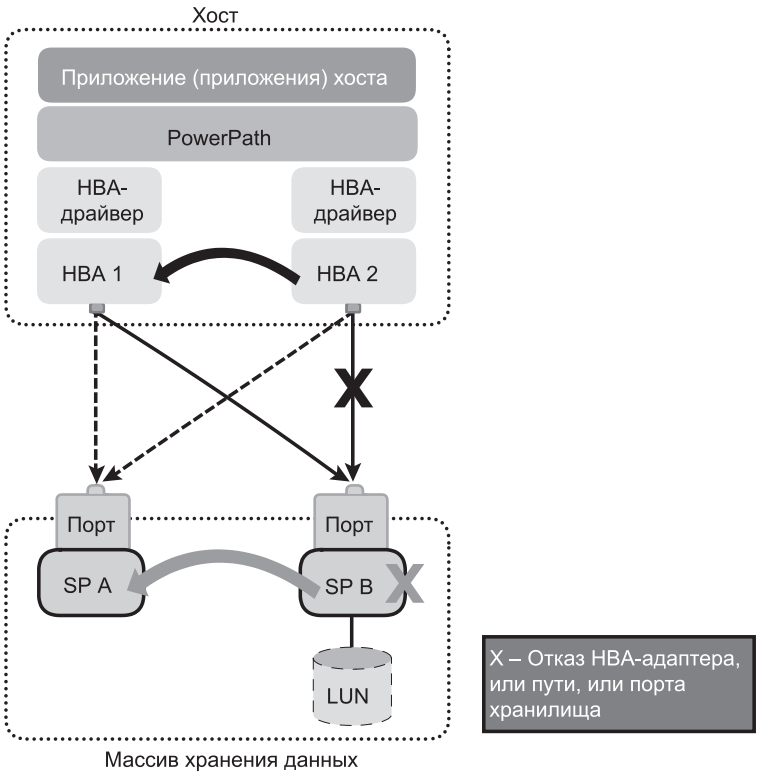


Рис. 9.11. Обход сбоя пути с использованием PowerPath для массива с конфигурацией «активный — пассивный»

Отказ пути может произойти в результате сбоя канала связи, HBA-адаптера или процессора хранилища (SP). В случае отказа пути PowerPath при конфигурации «активный — пассивный» выполняет операцию обхода сбоя следующим образом.

- Если путь ввода-вывода к процессору SP B, проходящий либо через HBA 2, либо через HBA 1, дает сбой, PowerPath использует для отправки запросов на ввод-вывод оставшийся доступный путь к SP B.

- Если отказывает процессор SP B, PowerPath прекращает отправку всех запросов на ввод-вывод к SP B и *переназначает* устройство процессору SP A. Все запросы на ввод-вывод отправляются по путям, ведущим к процессору SP A (по путям, которые ранее были в резерве, но теперь активированы для заданного LUN-устройства). Этот процесс называется *переназначением LUN-устройства*. Когда процессор будет возвращен в работоспособное состояние, программа PowerPath обнаружит его доступность и возобновит отправку запросов на ввод-вывод процессору SP B, но только после того как LUN-устройство будет опять переназначено процессору SP B.

Резюме

Технологические инновации сделали доступным широкий спектр устройств хранения данных и решений, отвечающих требованиям обеспечения непрерывности бизнес-процессов. Цель любого плана обеспечения непрерывности бизнес-процессов состоит в том, чтобы разработать и реализовать наиболее подходящие процедуры управления рисками и смягчения последствий, которые призваны защитить бизнес от возможных сбоев. Важнейшим является процесс анализа конфигурации оборудования и программного обеспечения на наличие единых точек отказа и определение степени их влияния на бизнес-операции. Анализ факторов, влияющих на бизнес-процессы, помогает организациям разработать план обеспечения непрерывности бизнес-процессов, который мог бы гарантировать, что инфраструктура и службы хранилища данных отвечают бизнес-требованиям. План обеспечения непрерывности бизнес-процессов дает организациям основу для создания и внедрения эффективных и низкозатратных процедур аварийного восстановления и перезапуска как в физической, так и в виртуальной среде. В постоянно меняющейся бизнес-среде обеспечение непрерывности бизнес-процессов может стать неукоснительным требованием.

В следующих трех главах рассматриваются конкретные технологические решения по обеспечению непрерывности бизнес-процессов, резервное копирование, а также локальная и удаленная репликация.

УПРАЖНЕНИЯ

1. У системы три компонента, и от всех трех требуется нахождение в рабочем состоянии 24 часа в сутки с понедельника по пятницу. Сбой компонента 1 происходит в следующей хронологии:
   - понедельник — без сбоев;
   - вторник — с 5.00 до 7.00;
   - среда — без сбоев;
   - четверг — с 16.00 до 18.00;
   - пятница — с 8.00 до 11.00.

Вычислите показатели MTBF и MTTR для компонента 1.

2. У системы три компонента, и от всех трех требуется нахождение в рабочем состоянии с 8.00 до 17.00, то есть в рабочее время, с понедельника по пятницу. Сбой компонента 2 происходит в следующей хронологии:
   - понедельник — с 8.00 до 11.00;
   - вторник — без сбоев;
   - среда — с 16.00 до 19.00;
   - четверг — с 15.00 до 20.00;
   - пятница — с 13.00 до 14.00.

   Вычислите показатель доступности компонента 2.
3. IT-отдел банка предоставляет клиентам доступ к таблице курсов валют с 9.00 до 16.00 с понедельника по пятницу. Отдел обновляет таблицу ежедневно в 8.00, причем данные берутся из системы мэйнфрейм. Процесс обновления занимает 35 минут. Во вторник из-за повреждения базы данных обновить таблицу курсов не представилось возможным. В 9.05 обнаружилось, что в таблице имеются ошибки. Было перезапущено обновление, и таблица была создана заново к 9.45. Проверка продолжалась 15 минут, после чего таблица курсов стала доступна отделениям банка. Какова была степень доступности таблицы курсов за ту неделю, в которой случилось это происшествие, при условии, что оно было единственным?
4. Исследуйте различные плановые и внеплановые случаи недоступности информации в контексте операций дата-центра.
5. Исследуйте технологию кластерного объединения серверов, используемую в дата-центре.

Глава 10
Резервное копирование и архивирование

Резервное копирование заключается в создании и сохранении дополнительной копии исключительно для восстановления утраченных или поврежденных данных. С развитием бизнеса и выработкой нормативных требований к хранению, а также обеспечению сохранности и доступности данных перед организациями встала задача резервного копирования постоянно растущего объема данных. С ростом объема информации в условиях сохранения прежних объемов бюджетов на развитие информационных технологий и сокращения сроков на проведение резервного копирования эта задача еще больше усложнилась. Более того, чтобы соответствовать соглашениям об уровнях бизнес-обслуживания — business service-level agreements (SLA), организациям нужно обеспечить быстрое восстановление данных из резервных копий.

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Факторы, определяющие порядок резервного копирования
- Архитектура резервного копирования
- Топология резервного копирования
- Виртуальная библиотека на ленточном накопителе
- Дедупликация данных
- Резервное копирование виртуальной машины
- Архивирование данных

Важным шагом к реализации успешных решений по созданию резервных копий и восстановлению данных является оценка различных методов резервного копирования с рассмотрением соответствующих им методов восстановления данных и требований к обеспечению их сохранности.

Организации создают и обслуживают большие объемы данных, основная масса которых представляет собой фиксированное содержимое, обращение к которому после определенного периода времени происходит довольно редко. И тем не менее для соблюдения нормативных требований эти данные должны храниться в течение нескольких лет. Накопление этих данных

в основном хранилище повышает для организации его общую стоимость. Более того, эти накапливающиеся объемы данных должны иметь резервные копии, что в свою очередь увеличивает время, необходимое на проведение резервного копирования.

Процесс архивирования данных заключается в перемещении тех данных, которые вышли из активного употребления, из основного хранилища в недорогое второстепенное хранилище. В целях соблюдения нормативных соглашений данные в таком хранилище находятся в течение довольно длительного срока. Перемещение данных из основного хранилища уменьшает объем данных, подлежащих резервному копированию, что позволяет сократить время, требующееся на эту операцию.

В данной главе подробно рассматриваются цели резервного копирования, особенности резервного копирования и восстановления, методы резервного копирования, архитектура, топологии и адресаты резервного копирования. Также рассматриваются вопросы оптимизации, проводимой с помощью дедупликации данных, и резервное копирование в виртуализированной среде. Затем в главе рассматриваются типы архивов данных и архитектура решений по созданию архивов.

10.1. Цели резервного копирования

Резервное копирование выполняется для достижения следующих целей: аварийного восстановления, оперативного восстановления и архивирования.

10.1.1. Аварийное восстановление

Одна из целей резервного копирования — обеспечение аварийного восстановления данных. Резервные копии используются для восстановления данных в другом месте, когда основное место утратило работоспособность по причине аварии. На основе целевой точки восстановления — recovery-point objective (RPO) и целевого времени восстановления — recovery-time objective (RTO) организации используют для аварийного восстановления различные стратегии защиты данных. Если для аварийного восстановления выбран вариант использования магнитной ленты, то эта лента с резервной копией хранится вне данного места, после чего такие ленты могут быть затребованы для проведения процедуры восстановления данных на месте возникновения аварии. Организации, придерживающиеся жестких требований по показателям RPO и RTO, используют технологию удаленной репликации, позволяющую получить в месте аварийного восстановления точную копию данных. Это дает организациям возможность в случае аварии ввести производственные системы в строй за относительно короткий период времени. Удаленная репликация подробно рассматривается в главе 12.

10.1.2. Оперативное восстановление

Данные в производственной среде изменяются с каждой бизнес-сделкой и каждой операцией. Резервные копии используются для восстановления данных при их утрате или логическом повреждении, произошедшем в ходе обычной обработки. В большинстве организаций основная часть запросов на восстановление относится именно к этой категории. Например, пользователям свойственно случайно удалять важные электронные письма, бывают и повреждения файлов, и все это может быть восстановлено с помощью данных, находящихся в резервной копии.

10.1.3. Архивирование

Резервные копии создаются также с целью архивирования данных. Хотя в качестве основного решения для создания архивов появились контентно-адресуемые хранилища (CAS), которые рассматриваются в главе 8, в целях долгосрочного хранения записей транзакций, сообщений электронной почты и других коммерческих документов, сохраняемых согласно нормативным требованиям, на малых и средних предприятиях до сих пор используются традиционные резервные копии.

ОКНО ДЛЯ РЕЗЕРВНОГО КОПИРОВАНИЯ

Окном для резервного копирования называется тот период времени, в течение которого источник доступен для выполнения резервного копирования данных. Для резервного копирования данных с источника иногда требуется приостановка производственных операций, поскольку данные, подвергаемые копированию, подпадают под исключительную блокировку, с тем чтобы они могли использоваться исключительно в процессе резервного копирования.

10.2. Факторы, определяющие порядок резервного копирования

При выборе и реализации определенной стратегии резервного копирования первое, чему следует уделить внимание, — это объемы утрачиваемых данных и периоды простоя, приемлемые для бизнеса и выраженные в показателях RPO и RTO. Показатель RPO относится к моменту обязательного восстановления данных и к тому моменту, когда нужно перезапустить бизнес-операции. На его основе определяется периодичность создания резервных копий. Иными словами, показатель RPO определяет частоту их создания. Например, если приложению требуется показатель RPO, указывающий на момент

времени, наступающий через один день, резервные копии данных должны создаваться по крайней мере раз в день. Еще одним фактором является срок хранения, определяющий длительность периода, в течение которого необходимо сохранять резервные копии в интересах бизнеса. Некоторые данные хранятся годами, а некоторые — всего несколько дней. Например, данные, скопированные для архива, сохраняются в течение большего периода времени, чем данные, сохраненные в резервной копии с целью оперативного восстановления.

Еще одним фактором, требующим рассмотрения, является тип носителя или целевая система для резервного копирования, выбираемая на основе показателя RPO и оказывающая влияние на время восстановления данных. На производительность резервного копирования с использованием носителя на магнитной ленте влияют отнимающие много времени операции запуска и остановки лентопротяжного механизма, это особенно заметно при резервном копировании большого количества файлов малого размера.

Организации должны также учитывать уровни объемов данных резервного копирования, речь о которых пойдет далее в соответствующем разделе. При выработке стратегии резервного копирования нужно также выбирать наиболее подходящее время для выполнения резервного копирования, чтобы минимизировать негативное влияние на режим производственных операций. Также следует учитывать местонахождение, размер, количество файлов и степень сжатия данных, поскольку они могут влиять на процесс резервного копирования. Местоположение является важным фактором для данных, подлежащих резервному копированию. У многих организаций имеются десятки разнородных платформ, поддерживающих их деловую активность как в локальном, так и в удаленном режиме. Рассмотрим среду хранения данных, использующую данные резервных копий из множества источников. В процессе резервного копирования нужно учитывать особенности данных источников, с тем чтобы обеспечить целостность транзакций и содержимого данных. Этот процесс должен быть согласован со всеми разнородными платформами из всех мест, где находятся данные.

Размеры файлов и их количество также влияют на процесс резервного копирования. Резервное копирование файлов большого размера (например 10 файлов по 1 Мбайт) может занять меньше времени, чем резервное копирование аналогичного объема данных, составленного из большого количества небольших по размеру файлов (например, 10 000 файлов по 1 Кбайт).

В среде резервного копирования широко используются сжатие данных и дедупликация, рассматриваемые далее в соответствующем разделе, поскольку эти технологии позволяют экономить место на носителе. Встроенная поддержка аппаратного сжатия данных имеется на многих устройствах резервного копирования. Некоторые данные, например двоичный код приложений, плохо поддаются сжатию, в то время как текстовые данные сжимаются весьма эффективно.

10.3. Уровни объемов данных резервного копирования

Уровни объемов данных резервного копирования зависят от бизнес-потребностей и установленных показателей RTO и RPO. По уровням объемов данных резервное копирование может быть полным — full, инкрементным — incremental или накопительным (дифференциальным) — cumulative (differential). Чтобы выполнять требования по резервному копированию и восстановлению данных, большинство организаций используют сочетание этих трех видов резервного копирования. На рис. 10.1 показаны различные уровни объемов данных резервного копирования.

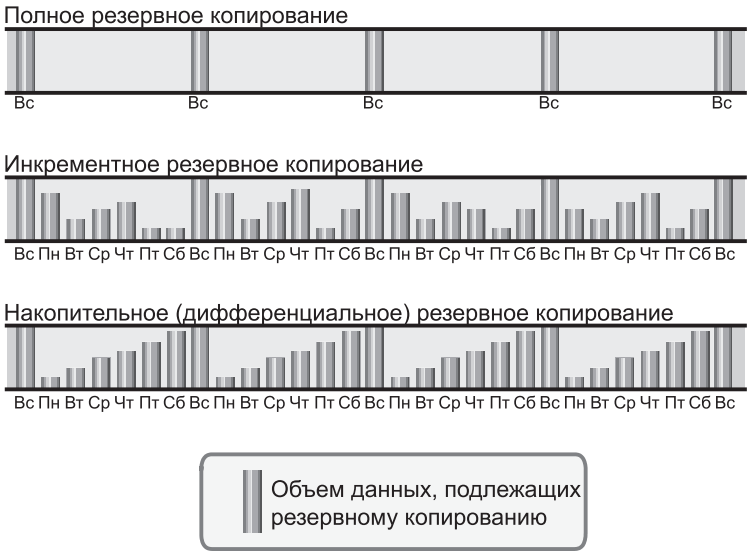


Рис. 10.1. Уровни гранулярности резервного копирования

Полное резервное копирование охватывает все данные производственных томов. Полная резервная копия создается путем копирования данных производственных томов на резервное устройство хранения. Инкрементное резервное копирование охватывает данные, изменившиеся с момента последнего полного или инкрементного резервного копирования. Оно проходит гораздо быстрее (поскольку охватываются только те данные, которые были изменены), но на восстановление данных уходит больше времени. Накопительное (дифференциальное) резервное копирование охватывает только те данные, которые изменились с момента последнего полного резервного копирования. Оно занимает больше времени, чем инкрементное копирование, зато восстановление данных проходит быстрее.

СИНТЕТИЧЕСКОЕ ПОЛНОЕ РЕЗЕРВНОЕ КОПИРОВАНИЕ

Есть еще один способ полного резервного копирования, который называется синтетическим или составным (synthetic or constructed). Он используется в том случае, когда ресурсы производственного тома не могут быть целиком отданы процессу резервного копирования на продолжительное время, требующееся для полного варианта копирования. При этом способе резервная копия составляется из самой последней полной резервной копии и всех инкрементных резервных копий, выполненных после нее. Такое резервное копирование называется синтетическим, потому что оно не касается напрямую производственных данных. Синтетическое полное резервное копирование позволяет создавать полную резервную копию автономно, не мешая проведению операции ввода-вывода на производственном томе. Кроме того, при резервном копировании не задействуются сетевые ресурсы, которые могут теперь использоваться для других производственных нужд.

Операции восстановления разнятся в зависимости от объема данных резервного копирования. При полном резервном копировании предоставляется отдельное хранилище данных, откуда их можно будет легко восстановить. Для восстановления после инкрементного резервного копирования требуются самая последняя полная резервная копия и все инкрементные резервные копии, доступные до момента восстановления. Для восстановления из накопительной резервной копии требуются самая последняя полная резервная копия, а также самая последняя накопительная копия.

На рис. 10.2 показан пример восстановления данных из инкрементной резервной копии.

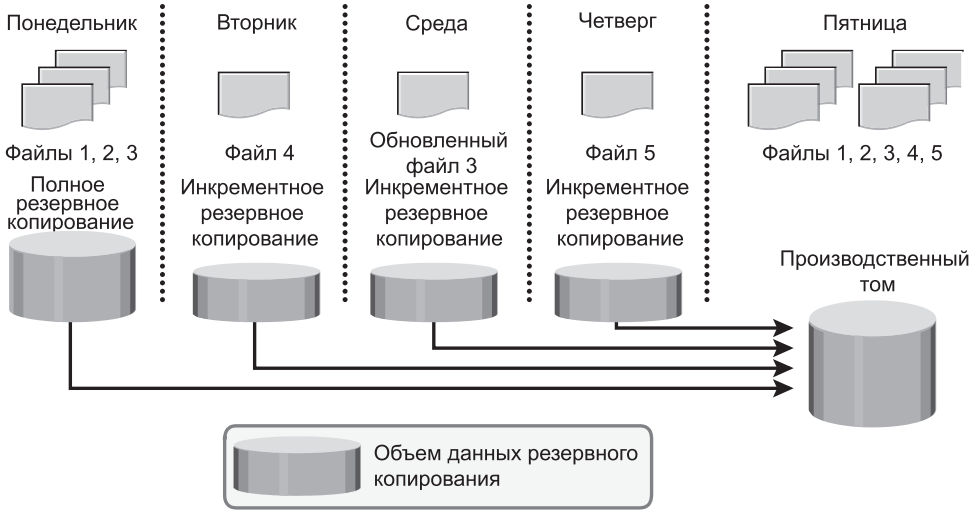


Рис. 10.2. Восстановление данных из инкрементной резервной копии

В данном примере полное резервное копирование производилось вечером в понедельник. Каждый день после этого выполнялось инкрементное резервное копирование. Во вторник добавился новый файл (файл *4* на рисунке), а другие файлы не изменились. Соответственно при инкрементном резервном копировании, выполненном во вторник вечером, был скопирован только файл *4*. В среду не было добавлено новых файлов, однако файл *3* был изменен. Соответственно, в среду вечером при инкрементном резервном копировании был скопирован только измененный файл *3*. Аналогично этому при инкрементном резервном копировании в четверг был скопирован только файл *5*. В пятницу утром произошло повреждение данных и потребовалось восстановление из резервной копии. Первым этапом восстановления данных стало восстановление всех данных из полной резервной копии, сделанной вечером в понедельник. Следующим этапом стало применение инкрементных резервных копий, сделанных во вторник, среду и четверг. Таким образом, данные можно успешно восстановить в их предыдущем состоянии, в том виде, в каком они были в четверг вечером. На рис. 10.3 показан пример восстановления данных из накопительной резервной копии.

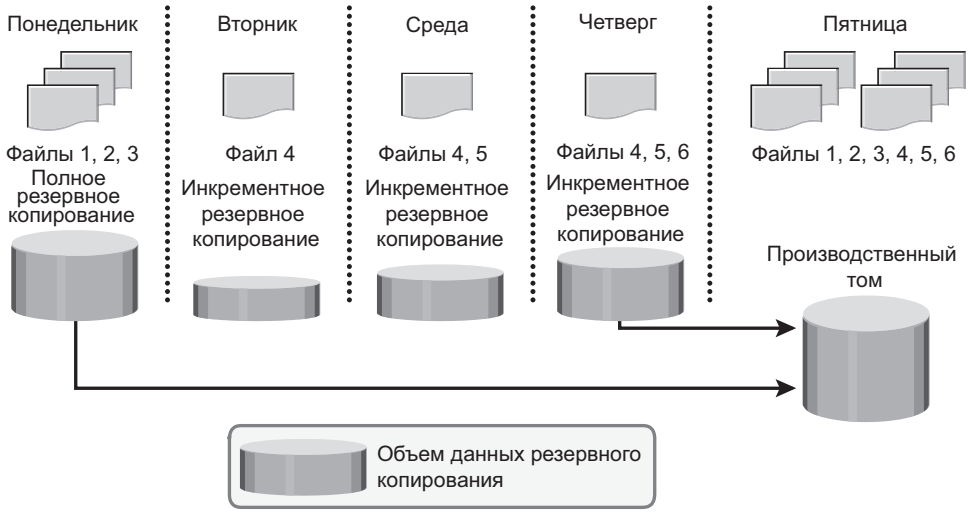


Рис. 10.3. Восстановление данных из накопительной резервной копии

В данном примере полное резервное копирование бизнес-данных выполнено вечером в понедельник. Каждый день после этого производилось накопительное резервное копирование. Во вторник добавился файл *4* и никакие другие данные после предварительного полного резервного копирования, выполненного вечером в понедельник, не менялись. Соответственно, при накопительном резервном копировании во вторник вечером был скопирован только файл *4*. В среду добавился файл *5*. При накопительном резервном

копировании, выполненном в среду вечером, были скопированы файлы *4* и *5*, поскольку после последнего полного резервного копирования они были добавлены или изменены. Аналогично в четверг добавился файл *6*. Поэтому при накопительном резервном копировании, вечером в четверг были скопированы все три файла: *4*, *5* и *6*.

Утром в пятницу произошло повреждение данных и потребовалось восстановление с использованием резервных копий. На первом этапе были восстановлены все данные из полной резервной копии, сделанной вечером в понедельник. На следующем этапе была применена только самая последняя накопительная резервная копия, сделанная в четверг вечером. Таким образом, данные производственного тома могут быть восстановлены быстрее, поскольку для этого потребуются только две копии данных — последняя полная резервная копия и самая последняя накопительная резервная копия.

10.4. Факторы, определяющие порядок восстановления данных

Ключевым фактором для восстановления данных является срок их хранения. Срок хранения резервной копии определяется на основе показателя RPO. Например, пользователи приложения могут потребовать восстановления его данных из их резервной копии, сделанной месяц назад. Исходя из этого определяется срок хранения резервной копии. Таким образом, минимальный срок хранения данных этого приложения составляет один месяц. Но в соответствии с внутренними установками или внешними факторами, например нормативными положениями, организация может выбрать и более длительный срок хранения резервных копий.

Если момент восстановления данных выбран за пределами срока хранения, то возможности восстановить все требующиеся данные на запрошенный момент времени может и не представиться. Для всех данных, нужных для заданного момента восстановления, могут быть определены длительные сроки хранения, позволяющие в течение определенного срока хранения соответствовать любому показателю RPO. Но для этого потребуется хранилище большой вместимости, что приведет к повышению затрат. Поэтому при определении срока хранения нужно проанализировать все прошлые запросы на восстановление и принять во внимание объем выделенных средств.

Показатель RTO зависит от времени, которое уходит на процесс восстановления. Чтобы соответствовать определенному показателю RTO, компания с целью минимизации времени восстановления может выбрать соответствующий уровень объема резервного копирования. В среде резервного копирования показатель RTO влияет и на выбор используемого типа носителя данных. Например, восстановление, производимое с магнитных лент, осуществляется дольше, чем восстановление с дисков.

10.5. Методы резервного копирования

В соответствии с состоянием приложения во время проведения резервного копирования существуют два метода его проведения: горячий и холодный. При *горячем резервном копировании* приложение запущено и работает, а пользователи имеют доступ к своим данным в течение всего процесса копирования. Этот метод также называют *резервным копированием в режиме онлайн*. При *холодном резервном копировании* приложение во время процесса копирования должно быть закрыто. Поэтому данный метод еще называют *автономным резервным копированием*.

Горячее резервное копирование производственных данных осуществляется сложнее, поскольку эти данные активно используются и изменяются. Если файл открыт, то он, как правило, в процессе создания резервной копии не копируется. В подобных ситуациях для резервного копирования открытых файлов нужен агент открытых файлов (open file agent). Такие агенты взаимодействуют непосредственно с операционной системой или приложением и позволяют создавать согласующиеся копии открытых файлов. В средах баз данных одного использования файловых агентов недостаточно, поскольку агенты должны также поддерживать согласованность резервных копий всех компонентов базы данных. Например, база данных состоит из множества файлов различных размеров, расположенных в нескольких файловых системах. Для обеспечения согласованного резервного копирования базы данных все файлы должны попасть в резервную копию в одинаковом состоянии. При этом совсем не обязательно, чтобы все файлы попали в резервную копию в одно и то же время, но они должны быть синхронизированы таким образом, чтобы база данных могла быть восстановлена в согласованном состоянии. Недостатком горячего резервного копирования является то, что агенты обычно оказывают негативное влияние на общую производительность приложения.

Согласованные резервные копии баз данных могут создаваться также в холодном режиме. Для этого требуется, чтобы базы данных в течение резервного копирования оставались неактивными. Конечно, такая недоступность для пользователей баз данных является минусом холодного копирования.

Метод создания мгновенных копий на заданный момент времени — point-in-time (PIT) используется в тех средах, где простой, связанный с холодным резервным копированием, или падение производительности по причине горячего резервного копирования являются неприемлемыми. PIT-копия создается с производственного тома и используется в качестве источника для резервного копирования. Это снижает отрицательное воздействие на производственный том. Более подробно эта технология рассматривается в главе 11.

Для обеспечения согласованности при восстановлении одного копирования производственных данных недостаточно. Необходимо также включать в копию прикрепленные к файлу свойства и атрибуты, такие как разрешения, сведения о владельце и прочие метаданные. Эти атрибуты так же важны,

как и сами данные, и должны быть включены в копию для обеспечения согласованности.

В среде восстановления после аварии к резервному копированию имеет отношение и так называемое восстановление на голое железо — bare-metal recovery (BMR), при котором для полного восстановления системы в резервную копию помещаются все метаданные, информация о системе и конфигурации приложений. При проведении BMR воссоздается базовая система, в которую входит разбиение на разделы, компоновка файловой системы, операционная система, приложения и все относящиеся к ним конфигурации. Перед началом восстановления файлов с данными при проведении BMR восстанавливается базовая система. Некоторые BMR-технологии, например с созданием резервной копии конфигурации сервера — server configuration backup (SCB), предусматривают восстановление сервера даже на оборудовании, отличающемся от исходного.

РЕЗЕРВНОЕ КОПИРОВАНИЕ КОНФИГУРАЦИИ СЕРВЕРА

Большинство организаций тратит много времени и денег, защищая данные своих приложений, но уделяет недостаточно внимания защите конфигурации своих серверов. Прежде чем в ходе аварийного восстановления приложение и данные станут доступны пользователю, нужно заново создать серверные конфигурации. Процесс восстановления системы включает в себя переустановку операционной системы, приложений и настроек сервера, а затем уже восстановление данных. В ходе обычной операции резервного копирования требуемые для восстановления серверные конфигурации не копируются. Резервное копирование конфигурации сервера — server configuration backup (SCB) предполагает создание и резервное копирование профилей серверной конфигурации на основе определенного пользователем расписания. Помещенные в резервную копию профили используются для настроек конфигурации восстанавливаемого сервера на случай отказа производственного сервера. SCB позволяет восстановить сервер даже на отличном от исходного оборудовании.

Процесс фиксирования мгновенного состояния конфигурации сервера приложений, проходящий в ходе создания резервной копии серверной конфигурации, называется профилированием. Данные профиля включают конфигурации операционной системы, конфигурации сети, конфигурации системы безопасности, настройки реестра, конфигурации приложения и т. д. Таким образом, профилирование позволяет восстановить конфигурацию отказавшей системы на новом сервере независимо от состава его оборудования.

Существует два типа профилей, создаваемых средой резервного копирования серверной конфигурации: основной профиль и расширенный профиль. В основном профиле содержатся ключевые элементы операционной системы, необходимые для восстановления сервера. Расширенный профиль обычно больше основного и содержит всю информацию, необходимую для воссоздания среды приложения.

10.6. Архитектура резервного копирования

В системе резервного копирования зачастую используется архитектура «клиент — сервер» с сервером резервного копирования и несколькими клиентами этой системы. Архитектура резервного копирования показана на рис. 10.4. Сервер резервного копирования управляет операциями резервного копирования и ведет его каталог, где содержатся информация о процессах резервного копирования и метаданные. В конфигурации резервного копирования содержится информация о том, когда запускать резервное копирование, какие именно данные клиента подлежат копированию и т. д., а в метаданных резервного копирования — информация о копируемых данных. Роль клиента резервного копирования сводится к сбору данных, подлежащих резервному копированию, и отправке их на узел хранения. Он также отправляет на сервер резервного копирования данные отслеживания процесса.

Рис. 10.4. Архитектура резервного копирования

Узел хранения отвечает за запись данных в устройство резервного копирования. (В среде резервного копирования *узел хранения* представляет собой хост, контролирующий устройства резервного копирования.) Узел хранения также отправляет информацию отслеживания процесса на сервер резервного копирования. Во многих случаях узел хранения объединен с сервером резервного копирования и оба они находятся на одной физической платформе. Устройство резервного копирования подключено напрямую или через сеть

к платформе, на которой находится узел хранения. В некоторых архитектурах резервного копирования узел хранения называется *медиасервером*, поскольку он управляет устройством хранения данных.

Программное обеспечение резервного копирования позволяет составлять отчеты на основе каталога резервного копирования и журнальных файлов. Эти отчеты могут содержать такую информацию, как общий объем скопированных данных, количество завершенных и незавершенных резервных копирований и типы ошибок, которые могут возникать в процессе копирования. Внешний вид отчета можно настраивать в зависимости от используемого специального программного обеспечения резервного копирования.

Важным аспектом резервного копирования является защита его метаданных. При утрате каталога резервного копирования восстановление данных может существенно усложниться. Поэтому все время отдельно от основного каталога следует вести его обновляемую копию.

10.7. Операции резервного копирования и восстановления

С момента начала процесса резервного копирования между различными компонентами его инфраструктуры происходит передача по сети больших объемов данных. Обычно операции резервного копирования инициируются сервером, но они могут быть инициированы и клиентом. Сервер резервного копирования инициирует процесс для различных клиентов на основе заданного для них расписания. Например, в расписании для процесса создания резервной копии для группы клиентов может быть задано время начала процесса ежедневно в 23:00.

Сервер резервного копирования координирует процесс копирования со всеми компонентами этой среды (рис. 10.5). Он ведет информацию о клиентах, у которых надлежит проводить операции резервного копирования, и об узлах хранения, используемых в этих операциях. Сервер резервного копирования извлекает из каталога информацию, связанную с этим копированием, и на ее основе дает команду узлу хранения на загрузку соответствующего носителя резервной копии в устройство создания этой копии. Одновременно он дает команду клиентам резервного копирования на сбор копируемых данных и отправку их по сети назначенному узлу хранения. После отправки данных резервного копирования на узел хранения клиент отправляет на сервер метаданные резервного копирования (количество файлов, имена файлов, подробные сведения об узле хранения и т. д.). Узел хранения получает данные от клиента, приводит их в порядок и отправляет на устройство хранения. Затем узел

хранения отправляет дополнительные метаданные (место, где находятся данные на устройстве резервного копирования, время создания резервной копии и т. д.) на сервер резервного копирования, который в свою очередь использует эту информацию для обновления каталога резервного копирования.

Рис. 10.5. Операция резервного копирования данных

После создания резервной копии данных при необходимости эти данные можно будет восстановить. Процесс восстановления должен быть запущен вручную от клиента. Некоторые пакеты программ резервного копирования имеют отдельные приложения для операций восстановления данных. Эти приложения обычно доступны только администраторам или операторам резервного копирования. Процесс восстановления данных показан на рис. 10.6.

После получения запроса на восстановление администратор открывает приложение восстановления данных для просмотра списка клиентов, для которых были созданы резервные копии. Кроме выбора клиента, для которого был сделан запрос на восстановление, администратору также необходимо определить клиента, который получит данные для восстановления. Данные могут быть восстановлены на машине того же клиента, для которого был сделан запрос на восстановление, или на машине любого другого клиента. Затем на основании показателя RPO администратор выбирает данные для восстановления и конкретный момент времени, к которому нужно будет восстановить данные. Поскольку вся эта информация поступает из каталога резервного копирования, приложение восстановления данных должно обмениваться информацией с сервером резервного копирования.

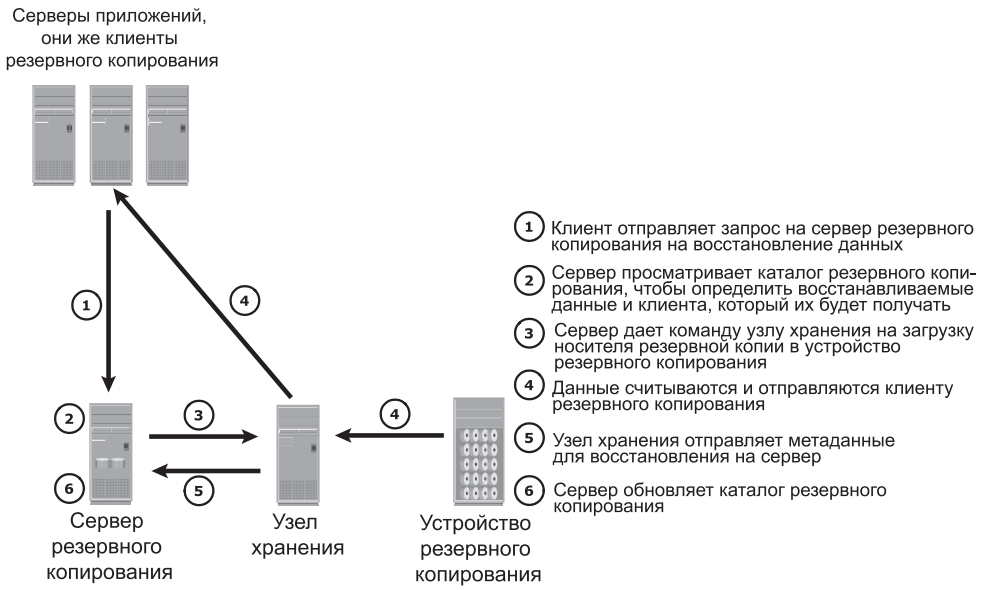


Рис. 10.6. Операция восстановления данных

Сервер резервного копирования дает команду соответствующему узлу хранения на установку определенного носителя резервной копии на устройство резервного копирования. Затем данные считываются и отправляются клиенту, который был определен в качестве получателя данных для восстановления.

Иногда восстановление происходит вполне успешно за счет использования только лишь запрошенных производственных данных. Например, процесс восстановления электронной таблицы завершается, как только будет восстановлен конкретный файл. А при восстановлениях баз данных наряду с производственными данными должны быть восстановлены и вспомогательные файлы, например файлы журналов. Тем самым обеспечивается согласованность восстанавливаемых данных. В таких случаях показатель RTO продлевается из-за необходимости выполнения в ходе операции восстановления данных дополнительных действий.

10.8. Топологии резервного копирования

В среде резервного копирования применяются три основные топологии: резервное копирование с использованием непосредственного подключения, резервное копирование с использованием локальной сети и резервное копирование в сети хранения данных. Также используется смешанная топология, сочетающая в себе топологию с использованием локальной сети и топологию с использованием сети хранения данных.

В случае резервного копирования с использованием непосредственного подключения устройство резервного копирования подключается непосредственно к клиенту. По локальной сети серверу резервного копирования отправляются только метаданные. Такая конфигурация освобождает локальную сеть от трафика резервного копирования. На рис. 10.7 показан пример непосредственного подключения устройства резервного копирования к тому клиенту, которому оно было выделено. По мере разрастания среды резервного копирования с целью оптимизации расходов возникает необходимость в централизованном управлении и совместном использовании устройств резервного копирования. Для подходящего в таком случае решения понадобится организовать совместное использование устройств резервного копирования несколькими серверами. Для решения, позволяющего оптимизировать использование устройств резервного копирования, нужно будет воспользоваться топологиями на основе сетей (локальных сетей и сетей хранения данных).

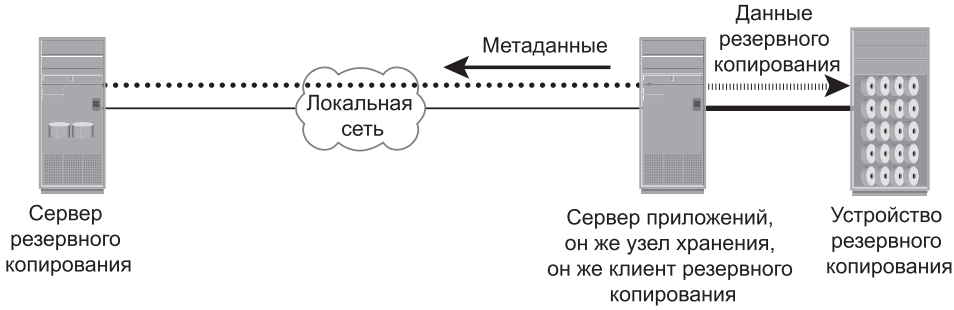


Рис. 10.7. Топология резервного копирования с непосредственным подключением

При резервном копировании с использованием локальной сети все клиенты, сервер резервного копирования, узел хранения и устройство резервного копирования подключены к этой сети (рис. 10.8). Данные, подлежащие резервному копированию, передаются от клиента резервного копирования (от источника) к устройству резервного копирования (к получателю) по локальной сети, что может отрицательно повлиять на ее производительность.

Отрицательное влияние может быть сведено к минимуму за счет принятия ряда таких мер, как включение в конфигурацию отдельных сетей для резервного копирования и установка специально выделенных узлов хранения для некоторых серверов приложений.

Резервное копирование с использованием сети хранения данных (SAN) также известно как резервное копирование без использования локальной сети. Топология этого вида копирования является наиболее подходящим решением, когда нужно предоставить устройство резервного копирования для совместного использования сразу нескольким клиентам. В таком случае устройство резервного копирования и клиенты подключаются к сети хранения данных. Резервное копирование с использованием сети хранения данных показано на рис. 10.9.

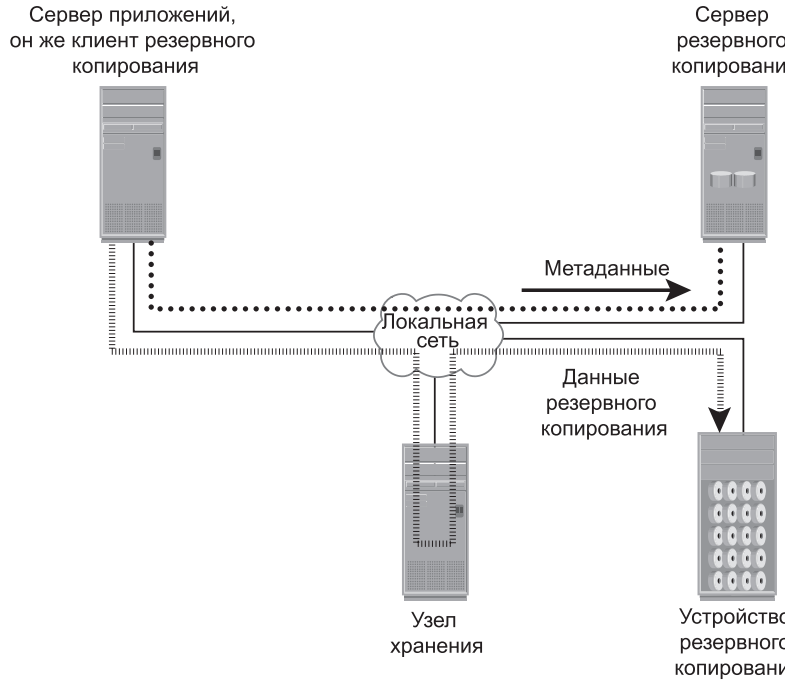


Рис. 10.8. Топология резервного копирования с использованием локальной сети

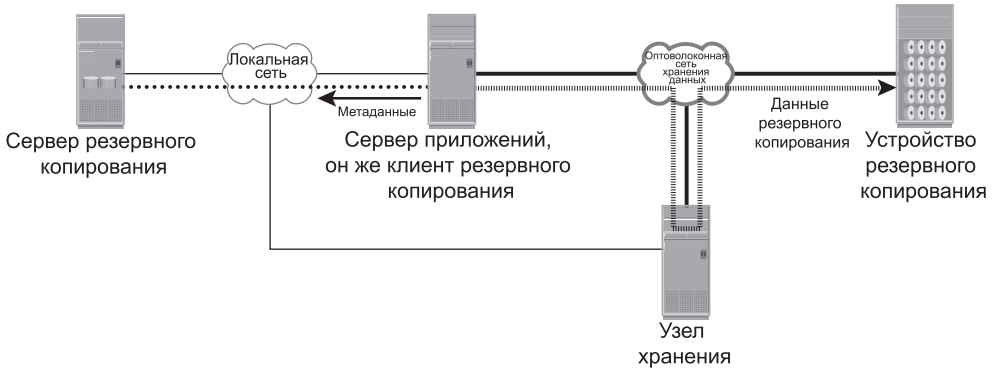


Рис. 10.9. Топология резервного копирования с использованием сети хранения данных

В этом примере клиент отправляет данные, предназначенные для резервного копирования, устройству резервного копирования по сети хранения данных. Поэтому трафик данных резервного копирования ограничивается сетью хранения данных (SAN), а по локальной сети отправляются только метаданные резервного копирования. По сравнению с производственными данными метаданные имеют весьма скромный объем, и в данной конфигурации производительность локальной сети от них не пострадает.

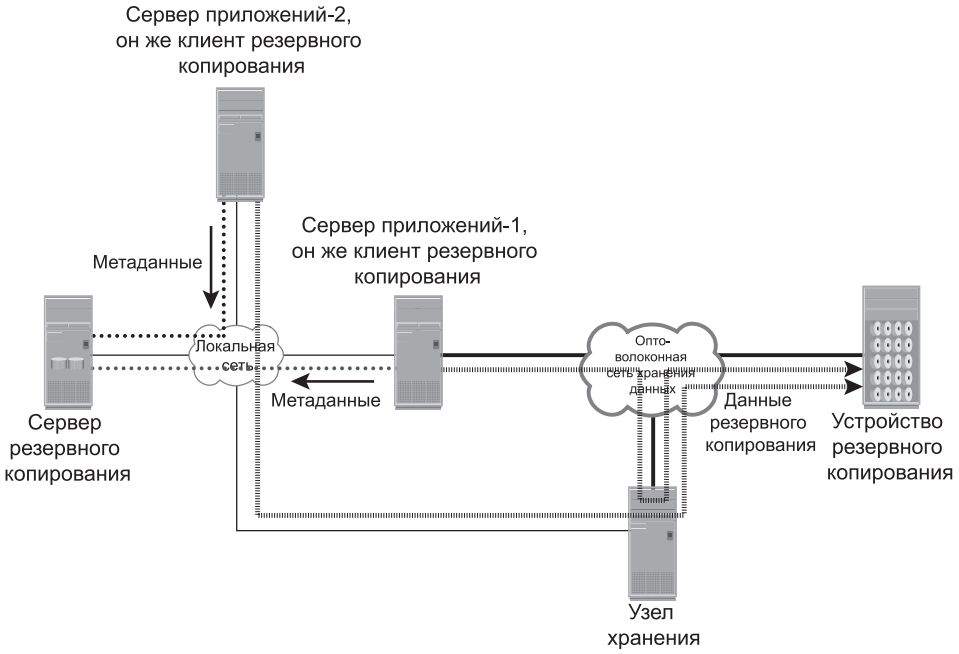


Рис. 10.10. Смешанная топология резервного копирования

Появление в качестве носителей резервных копий недорогих дисковых накопителей позволило подключать дисковые массивы к сети хранения данных и использовать их в качестве устройств резервного копирования. Для долговременного хранения и восстановления данных после возникновения чрезвычайных ситуаций, с этих резервных копий на дисках может быть сделана резервная копия на магнитной ленте, которую можно будет доставить в хранилище, находящееся в другом месте.

В *смешанной топологии* используются топологии как на основе использования локальных сетей, так и на основе использования сетей хранения данных (рис. 10.10). Причиной создания этой топологии может быть целый ряд обстоятельств, включая стремление к экономии средств, учет расположения серверов, необходимость сокращения административных издержек и решения проблем производительности.

10.9. Резервное копирование в средах сетевых устройств хранения данных (NAS)

Использование NAS-надстроек открывает новые перспективы для отработки стратегии резервного копирования и восстановления в NAS-среде. В NAS-надстройках используются собственная операционная система и структура файловой системы, которые поддерживают различные протоколы совместного использования файлов. В NAS-среде резервное копирование может

осуществляться различными способами: с использованием сервера, без использования сервера или с использованием сетевого протокола управления данными — Network Data Management Protocol (NDMP). Наиболее распространенными реализациями этого протокола являются протоколы NDMP 2-way и NDMP 3-way с двумя и тремя направлениями соответственно.

10.9.1. Резервное копирование с использованием и без использования сервера

При резервном копировании с использованием сервера NAS-надстройка извлекает данные из массива хранения данных по сети и передает их клиенту резервного копирования, запущенному на сервере приложений. Клиент резервного копирования отправляет эти данные на узел хранения, который в свою очередь записывает данные на устройство резервного копирования. Это приводит к перегрузке сети данными резервного копирования и использованию ресурсов сервера приложений для перемещения копируемых данных. Резервное копирование в NAS-среде с использованием сервера показано на рис. 10.11.

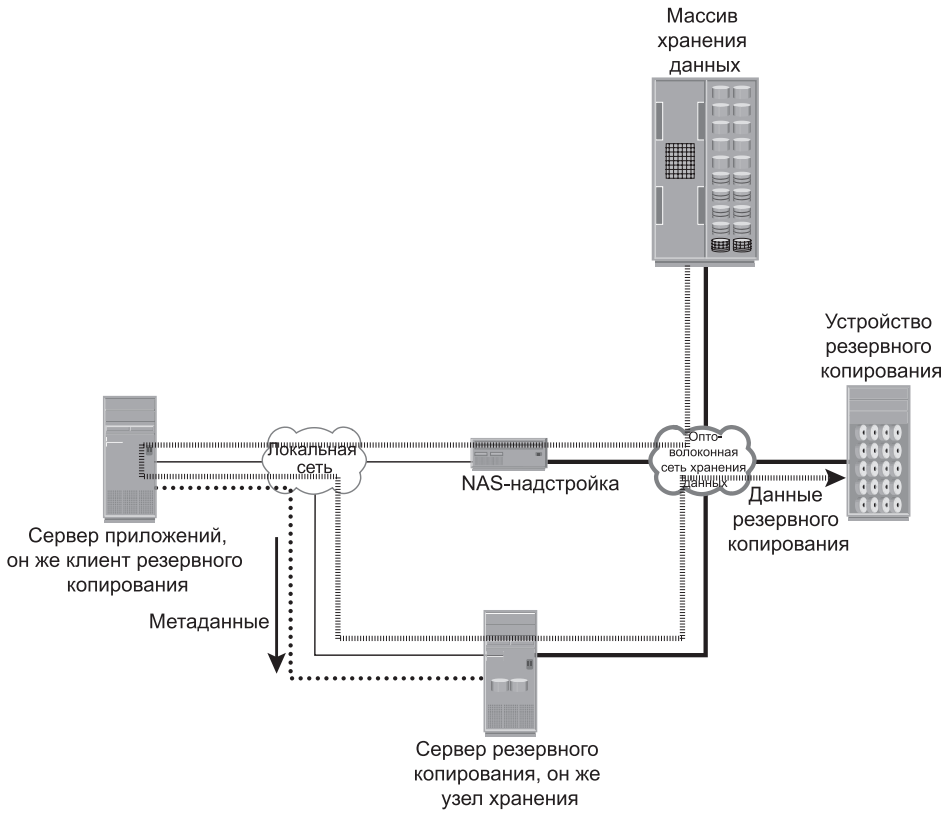


Рис. 10.11. Резервное копирование в NAS-среде с использованием сервера

При резервном копировании без использования сервера совместное использование сети организуется непосредственно на узле хранения. Это позволяет избегать перегрузок сети в ходе резервного копирования и исключает необходимость использования ресурсов сервера приложений. Резервное копирование в NAS-среде без использования сервера показано на рис. 10.12. В данном сценарии узел хранения, который также является и клиентом резервного копирования, считывает данные с NAS-надстройки и записывает их на устройство резервного копирования, не привлекая для этого сервер приложений. По сравнению с предыдущим решением это позволяет избавить сеть от одного транзитного участка.

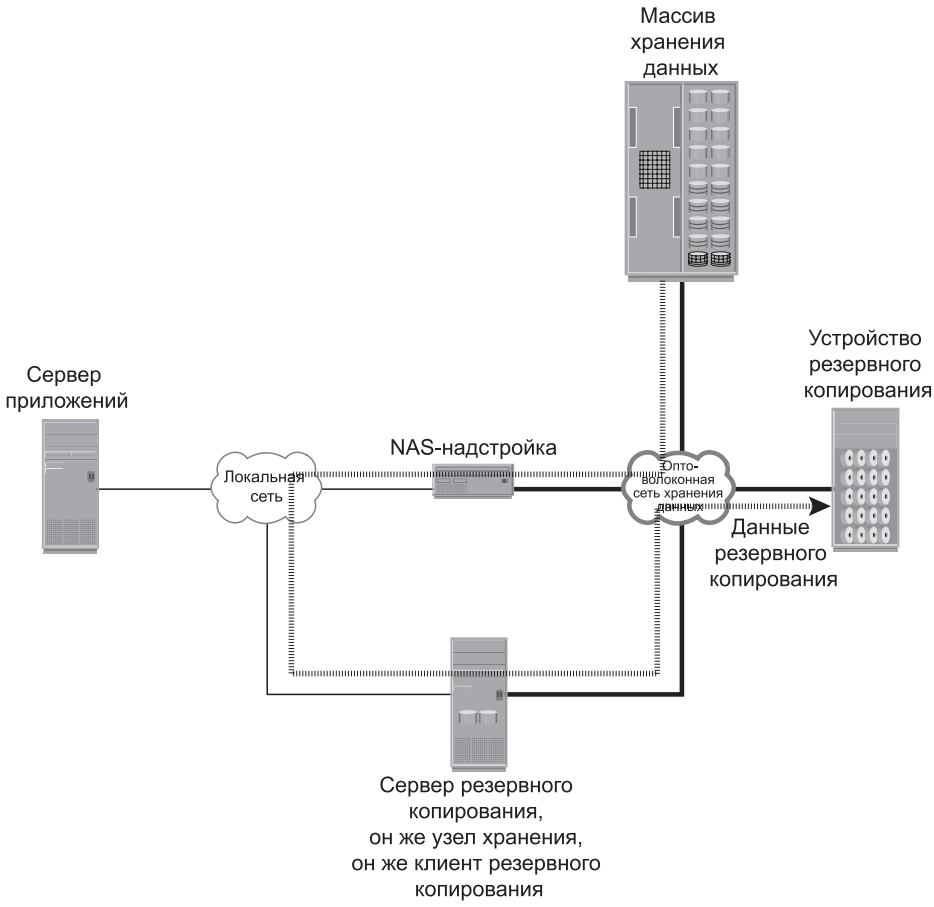


Рис. 10.12. Резервное копирование в NAS-среде без использования сервера

10.9.2. Резервное копирование с использованием NDMP-протокола

NDMP представляет собой принятый в промышленности протокол на основе TCP/IP-технологии, специально разработанный для проведения резервного копирования в NAS-среде. С его использованием организуется связь нескольких элементов в среде резервного копирования (NAS-надстройки, устройств резервного копирования, сервера резервного копирования и т. д.) с целью переноса данных. Также он позволяет поставщикам использовать общепринятый протокол для архитектуры резервного копирования. Данные могут быть помещены в резервную копию с помощью NDMP-протокола независимо от используемой операционной системы или платформы. Благодаря предоставляемой протоколом возможности проявления гибкости теперь нет необходимости переносить данные через сервер приложений, что снижает нагрузку на этот сервер и увеличивает скорость резервного копирования.

NDMP позволяет оптимизировать резервное копирование и восстановление путем использования высокоскоростного соединения устройств резервного копирования с NAS-надстройкой. При использовании протокола NDPM данные резервного копирования отправляются непосредственно из NAS-надстройки в устройство резервного копирования, а метаданные — на сервер резервного копирования. Пример резервного копирования в NAS-среде с использованием протокола с двумя направлениями — NDMP 2-way — показан на рис. 10.13. В этой модели сетевой трафик сведен к минимуму за счет того, что движение данных от NAS-надстройки к локально подключенному устройству резервного копирования происходит в изолированной области. По сети передаются только метаданные. Устройство резервного копирования выделено NAS-устройству, и, следовательно, этот метод не поддерживает централизованное управление всеми устройствами резервного копирования.

В методе, использующем протокол с тремя направлениями — NDMP 3-way, между всеми NAS-надстройками и NAS-надстройкой, подключенной к устройству резервного копирования, должна быть установлена отдельная закрытая сеть резервного копирования. Метаданные и данные управления NDMP по-прежнему передаются по открытой сети. Пример резервного копирования с использованием протокола NDMP 3-way показан на рис. 10.14.

Резервное копирование данных с использованием протокола NDMP 3-way применяется в том случае, когда устройства резервного копирования нужно использовать совместно с несколькими NAS-надстройками. Благодаря этому у NAS-надстройки появляется возможность управлять устройством резервного копирования и совместно использовать его с другими NAS-надстройками, получая данные резервного копирования с использованием NDMP-протокола.

Рис. 10.13. Использование в NAS-среде протокола NDMP 2-way

10.10. Адресаты резервного копирования

В настоящее время для адресатов резервного копирования имеется широкий выбор технологических решений. Чаще всего адресатами становятся библиотеки магнитных лент и дисков. В прошлом из-за своей низкой стоимости доминирующими адресатами резервного копирования служили магнитные ленты. Но ограничения по производительности и возможностям управления, присущие ленточным технологиям, а также доступность недорогих дисковых накопителей сделали диски вполне приемлемым адресатом резервного копирования. Одним из вариантов использования дисков в качестве носителей резервных копий является виртуальная ленточная библиотека — virtual tape library (VTL). В этой библиотеке эмулируются ленточные накопители и предоставляются улучшенные возможности для резервного копирования и восстановления данных.

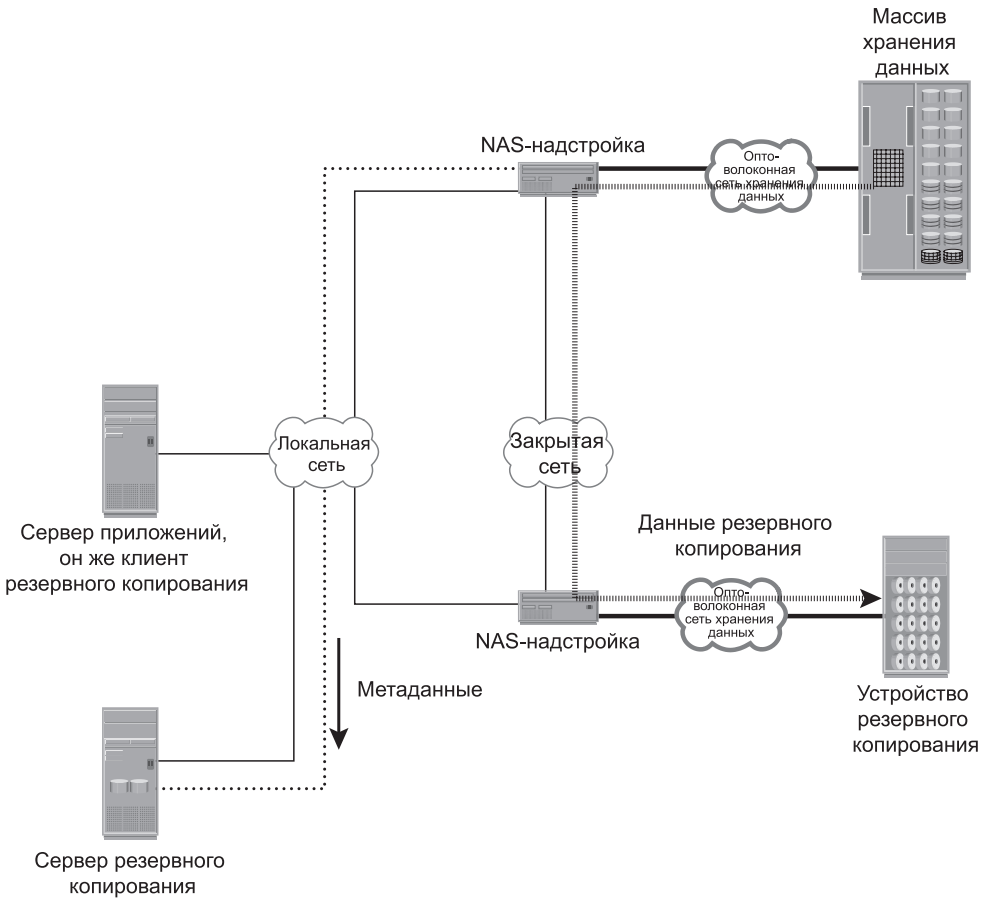


Рис. 10.14. Использование в NAS-среде протокола NDMP 3-way

10.10.1. Резервное копирование на ленту

Для резервного копирования благодаря своей относительной дешевизне широко используются магнитные ленты. Для чтения-записи данных используются ленточные накопители, в которых применяются картриджи (кассеты) с магнитной лентой. Такие накопители обычно называются устройствами с последовательным или линейным доступом, поскольку данные записываются или считываются последовательно. Ленточный картридж представляет собой магнитную ленту в пластмассовом корпусе. *Установкой ленты* называется процесс вставки ленточного картриджа в привод накопителя. Привод ленточного накопителя управляет двигателями, перемещающими магнитную ленту в разных направлениях, позволяя головке считывать или записывать данные.

Существует несколько типов ленточных картриджей. Они различаются по размеру, емкости, форме, плотности записи, длине ленты, толщине ленты, количеству дорожек на ленте и поддерживаемой скорости ее протяжки.

Физическая ленточная библиотека

В физической ленточной библиотеке размещаются и получают электропитание большое количество ленточных накопителей и картриджей, а также манипулятор или захватный механизм. В пакет программ резервного копирования заложен интеллект, достаточный для управления манипулятором и проведения всего процесса резервного копирования. Физическая ленточная библиотека показана на рис. 10.15.

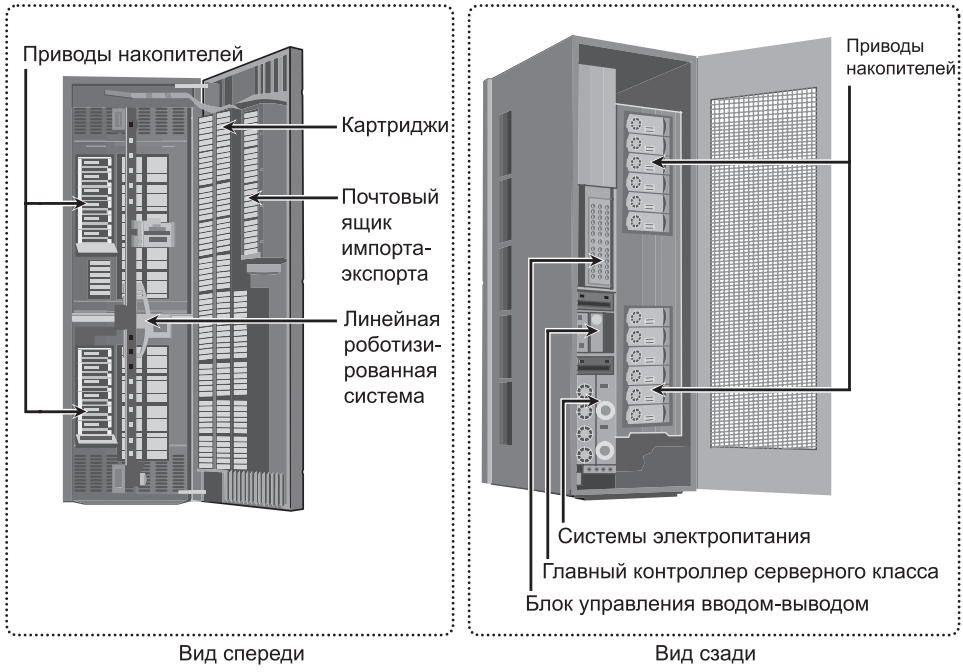


Рис. 10.15. Физическая ленточная библиотека

Ленточные накопители считывают и записывают данные, используя для этого магнитную ленту. Когда ленточные картриджи не используются накопителем, они помещаются в слоты. Для перемещения картриджей между слотами и приводами ленточных накопителей используются манипуляторы. Чтобы добавлять или извлекать ленты из библиотеки, не открывая панели доступа, используются почтовые, или импортно-экспортные, слоты (см. рис. 10.15, *вид спереди*).

При запуске процесса резервного копирования манипулятор получает команду на загрузку ленты в привод ленточного накопителя. Этот процесс

вызывает дополнительную задержку на величину, зависящую от типа используемого оборудования, но обычно установка ленты занимает от 5 до 10 секунд. После того как лента установлена, нужно еще время на позиционирование головок и проверку информации заголовка. Общее время, затрачиваемое на весь этот процесс, называется временем загрузки для подготовки к работе и может составлять от нескольких секунд до нескольких минут. Ленточный накопитель получает данные резервного копирования и сохраняет их в своем внутреннем буфере. Затем эти данные поблочно записываются на ленту. В ходе данного процесса желательно обеспечить бесперебойную занятость привода ленточного накопителя, чтобы не было разрывов между блоками. Это достигается путем буферизации данных на ленточных накопителях. Скорость приводов ленточных накопителей также может регулироваться, подстраиваясь под скорость передачи данных.

Чтобы выдерживать бесперебойную занятость привода ленточного накопителя, потоковые или многопотоковые ленточные накопители записывают на одну и ту же ленту данные сразу нескольких потоков. Как показано на рис. 10.16, работа сразу с несколькими потоками повышает производительность носителей, но имеет и недостаток. Данные резервного копирования получаются чередующимися, поскольку на ленту записываются данные сразу из нескольких потоков. Следовательно, время восстановления данных увеличивается, потому что при восстановлении одного потока должны считываться и отбрасываться все лишние данные других потоков.

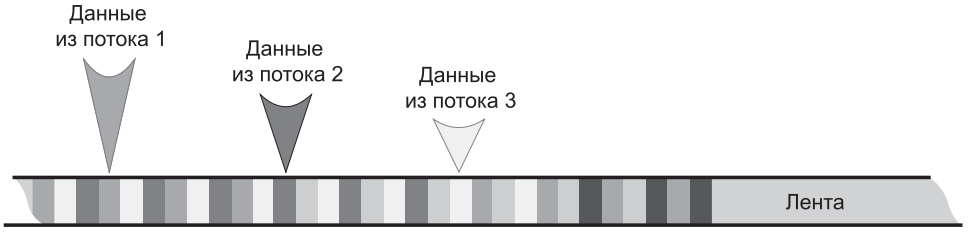


Рис. 10.16. Запись на ленточный носитель сразу нескольких потоков

Зачастую даже буферизация данных и настройка скорости протяжки ленты в приводе не могут предотвратить возникновения разрывов в записи, вызывающих так называемый эффект чистки обуви, или откат-разгон. Под ним подразумеваются повторяющиеся возвратно-поступательные движения привода ленточного накопителя, совершаемые им в местах прерывания потока данных резервного копирования. Например, если узел хранения отправляет данные медленнее, чем привод ленточного накопителя записывает их на ленту, привод периодически останавливается и ожидает догоняющие его работу данные. После того как привод определит, что данных достаточно для начала записи, он отматывает ленту до того места, где оканчивается последняя запись, и продолжает ведение записи. Эти повторяющиеся

возвратно-поступательные движения не только ухудшают качество обслуживания, но и приводят к повышенному износу и разрыву лент.

После завершения работы с лентой она перематывается на начало и извлекается из привода накопителя. Затем манипулятор получает команду на возвращение извлеченного из привода картриджа в его слот. Время перемотки может составлять от нескольких секунд до нескольких минут.

Когда инициируется операция восстановления, программа резервного копирования определяет нужную ленту. Манипулятор получает команду на перемещение картриджа с лентой из его слота в привод ленточного накопителя. Если нужная лента в библиотеке не найдена, программа выводит сообщение, в котором говорится, что оператор должен сам вставить требуемый картридж с лентой в стойку библиотеки. Когда требуется восстановить файл или группу файлов, то перед началом чтения лента должна быть перемотана к тому месту, где находится файл. Этот процесс может занять довольно много времени, особенно если требующиеся файлы записаны в конце ленты.

У современных ленточных устройств имеется механизм индексирования, который допускает быструю перемотку ленты вперед к месту, близкому к запрошенным данным. Затем привод ленточного накопителя точно выставляет позицию на ленте для получения данных. Но прежде чем принять решение, предусматривающее применение этого механизма, нужно взвесить все преимущества, получаемые от высокой производительности считывания потока данных, и сравнить их с издержками, связанными с записью индекса.

Ограничения, связанные с применением ленточных накопителей

Ленточные накопители благодаря своей низкой стоимости используются преимущественно для долговременного хранения данных на удаленной площадке.

Чтобы гарантировать сохранность носителей и предотвратить порчу данных, ленты следует хранить в местах с благоприятной для них контролируемой внешней средой. Ленты позволяют получать только последовательный доступ к данным, что может замедлить операции резервного копирования и восстановления данных. Физический перенос лент в какое-либо другое место приводит к дополнительным административным издержкам и повышает вероятность утраты лент в ходе их транспортировки.

10.10.2. Резервное копирование на диск

Благодаря своей растущей доступности и более высокой производительности сегодня в качестве основного устройства хранения резервных копий на смену лентам приходят недорогие диски. Системы резервного копирования на диски предлагают простоту реализации, более низкое значение показателя TCO и более высокое качество обслуживания. Кроме преимуществ производительности, выражающихся в скорости передачи данных, диски по сравнению с лентами предлагают также более быстрое восстановление данных.

Резервное копирование на системы дисковых устройств хранения данных предлагает вполне очевидные преимущества, обусловленные присущими этим системам произвольному доступу к данным и возможностям защиты данных с помощью RAID-технологий. В большинстве сред резервного копирования диски используются как подготовительная область, куда данные копируются на временной основе перед их переносом или размещением на лентах. Благодаря этому повышается производительность резервного копирования. Некоторые пакеты резервного копирования позволяют создаваемым образам резервных копий какое-то время оставаться на дисках даже после размещения этих копий на лентах. Это может существенно ускорить восстановление данных. На рис. 10.17 показан сценарий восстановления, в котором сравнивается работа с лентой и с диском в среде Microsoft Exchange, поддерживающей 800 пользователей с размерами почтовых ящиков 75 Мбайт и базой данных 60 Гбайт. Как показано на рисунке, для одного и того же окружения восстановление с диска занимает 24 минуты, а с ленты — 108 минут.

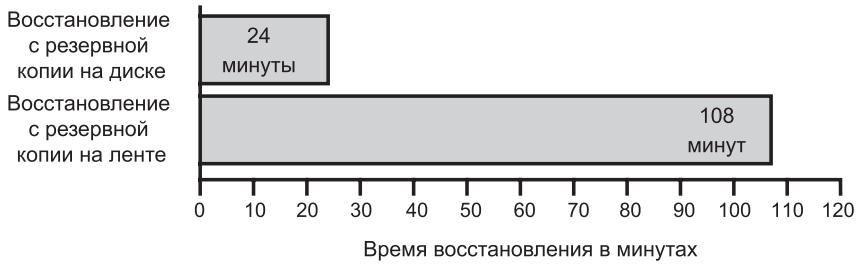


Рис. 10.17. Сравнение восстановления с диска и с ленты

Восстановление из полной резервной копии на диске, находящемся на основном месте производства, предлагает самое быстрое решение по восстановлению данных. Использование диска позволяет чаще делать полные резервные копии, что в свою очередь улучшает показатели RPO и RTO.

Проведение резервного копирования на диск не предлагает каких-либо присущих ему сторонних возможностей и зависит от применения других технологий, таких как локальная или удаленная репликация. Кроме того, некоторые пакеты резервного копирования для поддержки резервного копирования на диск требуют дополнительных модулей и лицензий, что может потребовать также дополнительных действий по созданию конфигурации, включая создание RAID-групп и настройку файловой системы. Эти действия администраторами резервного копирования обычно не выполняются.

10.10.3. Резервное копирование на виртуальную ленту

Виртуальные ленты — это не что иное, как дисковые накопители, эмулирующие ленты и представляющиеся в виде лент программам резервного копирования. Основным преимуществом использования виртуальной ленты

является отсутствие требований по установке каких-либо дополнительных модулей, изменению конфигурации или внесению каких-либо других изменений в унаследованное программное обеспечение резервного копирования, что позволяет не вкладывать средства в его замену.

Виртуальная ленточная библиотека

Виртуальная ленточная библиотека — virtual tape library (VTL) имеет те же компоненты, что и физическая ленточная библиотека, за исключением того, что большинство компонентов представляют собой виртуальные ресурсы. Для программ резервного копирования нет никакой разницы между физической и виртуальной ленточными библиотеками. Виртуальная ленточная библиотека показана на рис. 10.18.

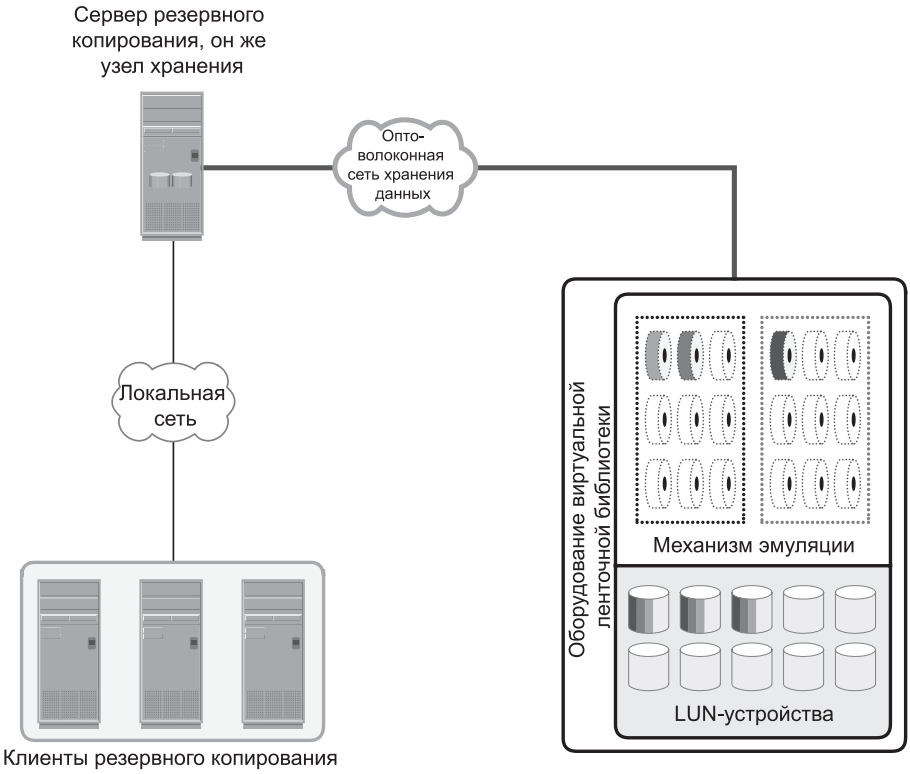


Рис. 10.18. Виртуальная ленточная библиотека

В качестве носителей резервного копирования в виртуальных ленточных библиотеках используются диски. У программы эмуляции есть своя база данных со списком виртуальных лент, и каждой виртуальной ленте выделяется пространство на LUN-устройстве. Если нужно, виртуальная лента может

распространяться на несколько LUN-устройств. При этом при резервном копировании не нужны никакие сведения о файловой системе, поскольку в решениях, использующих виртуальную ленту, обычно используются только сами устройства.

Как и в физической ленточной библиотеке, при запуске процесса резервного копирования в виртуальной ленточной библиотеке выполняется виртуальная роботизированная установка носителя данных. Но в отличие от физической ленточной библиотеки, где этот процесс влечет за собой механические задержки, в виртуальной ленточной библиотеке он проходит практически мгновенно. В ней даже время загрузки для подготовки к работе существенно меньше, чем в физической ленточной библиотеке.

После установки виртуальной ленты и позиционирования виртуального привода ленточного накопителя виртуальная лента готова к использованию и на нее можно записывать данные резервного копирования. В отличие от физической ленточной библиотеки, у виртуальной ленточной библиотеки отсутствуют ограничения, связанные с последовательным доступом и эффектом чистки обуви. Когда операция завершается, программа резервного копирования выдает команду на обратную перемотку. Эта «перемотка» также происходит мгновенно. Затем виртуальная лента удаляется и виртуальный манипулятор получает команду на ее возвращение в виртуальный слот.

Этапы восстановления данных аналогичны этапам в физической ленточной библиотеке, но операция восстановления происходит очень быстро. Несмотря на то что виртуальные ленты созданы на основе дисков, предоставляющих произвольный доступ к данным, в них все равно эмулируется поведение настоящей ленты.

Оборудование виртуальной ленточной библиотеки предоставляет ряд свойств, которые недоступны при использовании физических ленточных библиотек. Некоторые виртуальные ленточные библиотеки предлагают несколько механизмов эмуляции, сведенных в активную кластерную конфигурацию. Механизм эмуляции представляет собой выделенный сервер со специализированной операционной системой, который предъявляет приложению резервного копирования физические диски в виде лент VTL-библиотеки. Обладая этим свойством, один механизм эмуляции может забрать виртуальные ресурсы у другого механизма эмуляции в случае какого-либо сбоя и позволить клиентам продолжить использование выделенных им виртуальных ресурсов, не оповещая их об этом.

Большинство устройств виртуальных ленточных библиотек позволяют проводить репликацию данных по IP-сетям. Это позволяет создавать точные копии виртуальных лент в удаленном месте с использованием недорогих IP-сетей. В результате организации могут выполнить требования по хранению резервных копий в удаленном месте. Подключение механизмов эмуляции из состава оборудования виртуальной ленточной библиотеки к физической ленточной библиотеке позволяет копировать виртуальные ленты на физические, которые затем могут быть отправлены в защищенное

помещение или доставлены в удаленное место за пределами основного предприятия.

Использование виртуальных лент дает ряд преимуществ по сравнению с использованием как физических лент, так и дисков. Если сравнивать с физическими лентами, виртуальные ленты предлагают более высокую однопотоковую производительность, более высокую надежность и произвольный дисковый доступ к данным. Операции резервного копирования и восстановления данных выигрывают от произвольного доступа к данным, поскольку аппаратура всегда находится в состоянии полной готовности и предоставляет возможность провести резервное копирование и восстановление гораздо быстрее. Накопителю на виртуальной ленте не требуется проводить обычное техническое обслуживание, связанное с накопителем на физической ленте, например, периодически производить чистку и калибровку привода. Если проводить сравнение с устройствами резервного копирования на диски, то виртуальная ленточная библиотека позволяет упростить установку и управление, поскольку она уже сконфигурирована производителем. Но виртуальные ленточные библиотеки используются, как правило, только для целей резервного копирования. А в среде устройств резервного копирования на диски дисковая система используется как в производственных целях, так и для резервного копирования данных.

Сравнительные характеристики различных адресатов резервного копирования показаны в табл. 10.1.

Таблица 10.1. Сравнительные характеристики адресатов резервного копирования

ХАРАКТЕРИСТИКИ	ЛЕНТА	ДИСК	ВИРТУАЛЬНАЯ ЛЕНТА
Возможности репликации за пределы основного производства	Нет	Да	Да
Надежность	Отсутствие каких-либо присущих этой технологии методов защиты	Да	Да
Производительность	Нужно учитывать продолжительность механических операций и время загрузки	Более быстрая работа с одним потоком	Более быстрая работа с одним потоком
Использование	Только для резервного копирования	Разнообразное (резервное копирование, производство)	Только для резервного копирования

10.11. Дедупликация данных при резервном копировании

Традиционные решения по проведению резервного копирования не обеспечивают встроенной функциональности для предотвращения резервного копирования дублированных данных. В условиях постоянного роста объемов информации и выдвижения требований по ее постоянной доступности (24×7) окно резервного копирования сильно сужается. В ходе традиционных процессов резервного копирования в копию попадает слишком много дубликатов данных, что существенно повышает требования к величине окна резервного копирования и приводит к неоправданному расходу ресурсов, в частности, пространства хранения данных и полосы пропускания сети.

Избавлением от дубликатов данных называется процесс обнаружения и устранения избыточных данных. Когда в ходе резервного копирования обнаруживаются дубликаты данных, они отбрасываются, а на экземпляр уже попавших в резервную копию таких же данных создается указатель со ссылкой. Дедупликация данных помогает снизить требования к объему хранилища для резервного копирования, сузить окно резервного копирования, а также позволяет избежать высоких нагрузок на сеть. Кроме того, она помогает сохранять больше резервных копий на диске и хранить их намного дольше.

10.11.1. Методы дедупликации данных

Существуют два метода дедупликации: на уровне файлов и на уровне субфайлов. При определении уникальности данных каждый из этих двух методов имеет свои преимущества, но результаты их применения могут быть разными. Разница заключается в объеме сокращаемых данных и во времени, которое каждый из них затрачивает на определение уникальности содержимого.

Дедупликация на уровне файлов (также называется *сохранением единственного экземпляра*) позволяет находить и удалять лишние копии одинаковых файлов. Этот метод допускает сохранение только одной копии файла, а все последующие копии заменяются указателями на оригинальный файл. Дедупликация на уровне файлов проходит проще и быстрее, чем при использовании второго метода, но не решает проблемы дубликатов содержимого внутри файлов. Например, две PowerPoint-презентации по 10 Мбайт, различающиеся только титульными листами, не рассматриваются в качестве файлов-дубликатов и будут сохранены обе.

Дедупликация на уровне субфайлов позволяет разбивать файл на мелкие части, после чего для определения избыточных данных как внутри одного файла, так и во всех файлах использует специальный алгоритм. В результате дедупликации на уровне субфайлов удаляются дубликаты данных в файлах. Существует два вида субфайловой дедупликации: с блоками фиксированной длины и с сегментами переменной длины. При *избавлении от дубликатов*

с блоками фиксированной длины файлы разбиваются на блоки одинаковой длины и для поиска дубликатов применяется хэш-алгоритм. Эта разновидность при всей своей простоте может пропускать многие возможности обнаружения избыточных данных, поскольку границы блоков у одинаковых данных могут быть совершенно разными. Например, при добавлении чьего-то имени к заголовку документа произойдет смещение текста всего документа и получится, что у всех блоков будут изменения, при которых метод дедупликации не сможет найти совпадений. При *дедупликации с сегментами переменной длины* корректировка границы происходит только у того сегмента, в котором произошли изменения, а все остальные сегменты остаются в прежних границах. По сравнению с методом, в котором используются фиксированные блоки, этот метод существенно повышает возможности обнаружения дубликатов сегментов данных.

10.11.2. Реализация методов дедупликации данных

Дедупликация при резервном копировании происходит либо на источнике, либо на приемнике (адресате) данных.

Дедупликация на источнике данных

При *дедупликации на источнике данных* лишние данные удаляются на источнике еще до того, как они передаются на устройство резервного копирования. Это позволяет существенно сократить объем данных резервного копирования, отправляемых по сети в ходе такого копирования. Преимущества заключаются в сужении окна резервного копирования и в меньшей требующейся полосе пропускания сети. Кроме того, существенно сокращается пространство хранилища, необходимое для хранения образов резервных копий. Дедупликация на источнике данных показана на рис. 10.19.

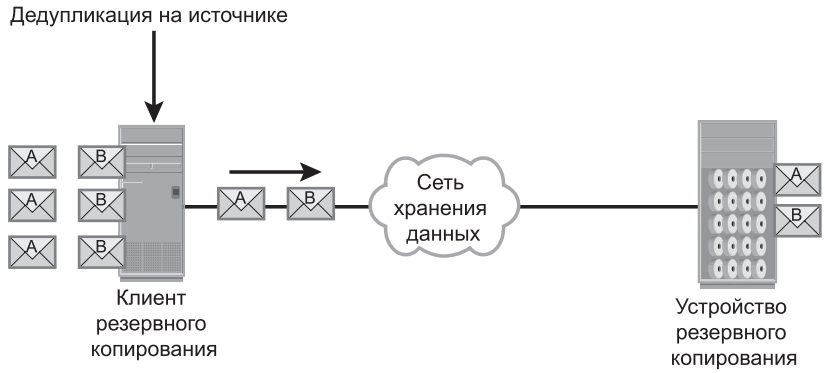


Рис. 10.19. Дедупликация на источнике данных

Дедупликация на источнике данных повышает непроизводственные расходы клиента резервного копирования, снижающие производительность

резервного копирования и приложения, запущенного на клиентской машине. Дедупликация на источнике данных может также потребовать внесения изменений в программный компонент резервного копирования, если эта функция в нем не поддерживается.

Дедупликация на приемнике данных

Альтернативой дедупликации на источнике данных служит *дедупликация на приемнике данных*. Она происходит на устройстве резервного копирования, снимая тем самым соответствующую нагрузку с клиента резервного копирования. Дедупликация на приемнике данных показана на рис. 10.20.

В данном случае клиент резервного копирования отправляет предназначенные для копирования данные устройству резервного копирования, на котором и осуществляется избавление от дубликатов либо сразу же (на входе), либо в то время, которое определено распорядком работы (после процесса сохранения). Поскольку дедупликация происходит на адресате резервного копирования, все данные резервного копирования приходится передавать по сети, что повышает требования к ее полосе пропускания. Дедупликация на приемнике данных не требует внесения изменений в существующий программный компонент резервного копирования.

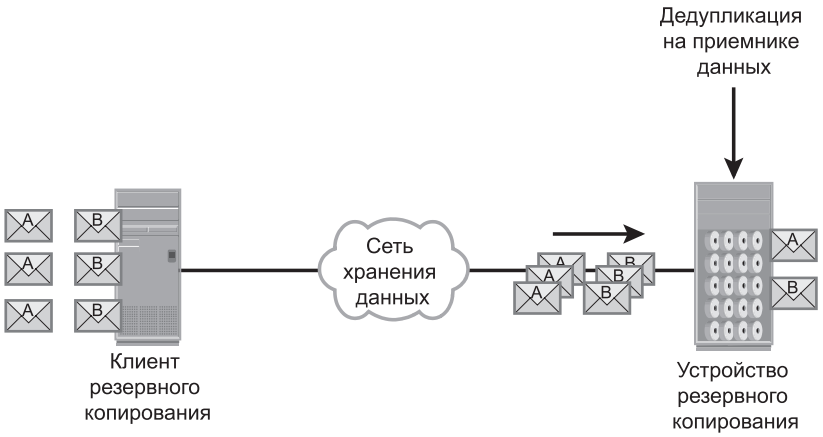


Рис. 10.20. Дедупликация на приемнике данных

Дедупликация на входе происходит перед сохранением данных на устройстве резервного копирования. Это позволяет сократить пространство хранилища, необходимое для резервной копии. Дедупликация на входе влечет за собой накладные расходы в виде времени, которое нужно затратить на обнаружение и удаление дубликатов данных. Поэтому данный метод больше всего подходит для тех сред, которые имеют большие окна резервного копирования.

Дедупликация после сохранения данных позволяет сначала сохранять или записывать данные на устройство резервного копирования, а процесс

дедупликации проводить позже. Этот метод больше подходит для тех ситуаций, когда нужно воспользоваться более узким окном резервного копирования. Но дедупликация после сохранения данных требует от хранилища большего пространства для сохранения образов резервных копий до удаления из них дубликатов данных.

РЕЗЕРВНОЕ КОПИРОВАНИЕ (В ТОМ ЧИСЛЕ АВТОМАТИЗИРОВАННОЕ) В УДАЛЕННЫХ ОФИСАХ ИЛИ ОТДЕЛЕНИЯХ ОРГАНИЗАЦИИ

Сегодня у организаций зачастую имеются удаленные офисы или отделения, разбросанные по нескольким местам. Обычно удаленные офисы располагают собственной локальной IT-инфраструктурой, куда включаются файловые серверы, серверы печати, веб-серверы или серверы электронной почты, рабочие станции и настольные компьютеры. Кроме того, у них могут быть установлены приложения и базы данных. Эти системы являются для удаленных офисов основой для поддержки региональных бизнес-функций, например обработки заказов, управления запасами и сбытом продукции.

Довольно часто данные, имеющиеся в удаленных офисах и представляющие для ведения бизнеса особую важность, оказываются недостаточно защищенными, что подвергает бизнес риску утраты данных и потери продуктивности. Поэтому защита данных в отделениях организации или удаленных офисах, разбросанных по нескольким местам, играет весьма важную роль в ведении бизнеса. Обычно резервное копирование данных удаленных офисов делается вручную с использованием магнитных лент, которые затем перевозятся в другие места, чтобы данные можно было восстановить в случае возникновения чрезвычайных ситуаций. Но такой подход сопряжен с рядом трудностей:

- недостатком высококвалифицированного технического персонала для создания резервных копий на местах;
- риском перевозки лент в другое место с возможностью утраты или кражи важных данных.

Резервное копирование данных из удаленных офисов в главный дата-центр было связано с существенными затратами времени и средств на передачу большого объема данных по глобальной сети. Поэтому организациям нужно было найти эффективное решение проблем, связанных с резервным копированием и восстановлением данных удаленных офисов и отделений.

Справиться с проблемами централизованного хранения резервных копий данных удаленных офисов помогут решения, основанные на применении дисков, и дедупликация на источнике данных. Дедупликация существенно снижает требования к ширине полосы пропускания сети и позволяет выполнять резервное копирование данных удаленных офисов с помощью уже имеющейся сетевой инфраструктуры. Это также позволяет организации централизованно управлять резервным копированием данных удаленных офисов и делать это в автоматическом режиме, снижая требования к размеру окна резервного копирования.

10.12. Резервное копирование в виртуализированных средах

В виртуализированной среде необходимость в резервном копировании данных виртуальной машины (операционной системы, данных приложений и конфигурации) связана с необходимостью предупреждения их утраты или повреждения из-за человеческих или технических ошибок. В виртуализированной системе применяются два подхода к резервному копированию: традиционное резервное копирование и резервное копирование на основе создания образов резервных копий.

При *традиционном подходе к резервному копированию* на виртуальную машину (VM) или на гипервизор устанавливается агент резервного копирования. Традиционный подход к резервному копированию данных виртуальной машины показан на рис. 10.21. Если агент резервного копирования установлен на виртуальную машину, эта машина представляется ему в виде физического сервера. Агент, установленный на виртуальной машине, осуществляет резервное копирование ее данных на устройство резервного копирования. Агент не собирает файлы самой виртуальной машины, такие как файл виртуальной BIOS-системы, своп-файл виртуальной машины, файлы журналов и конфигурации. Поэтому для восстановления виртуальной машины пользователю нужно вручную воссоздать эту виртуальную машину, а затем восстановить на ней данные.

Рис. 10.21. Традиционный метод резервного копирования данных виртуальной машины

Если агент резервного копирования установлен на гипервизор, виртуальные машины представляются этому агенту в виде набора файлов. Поэтому файлы виртуальной машины могут попасть в резервную копию путем выполнения резервного копирования файловой системы из гипервизора. Считается, что этот подход проще предыдущего, поскольку требует наличия агента только на гипервизоре, а не на всех виртуальных машинах. Реализация традиционного метода резервного копирования может сильно нагрузить центральный процессор сервера, выполняющего эту задачу.

При традиционном подходе резервное копирование должно выполняться в то время, когда ресурсы сервера свободны, или в период низкой активности сети. Когда в среде имеется большое количество виртуальных машин, нужно также предусмотреть выделение достаточного количества ресурсов, позволяющих справиться с резервным копированием на каждом сервере.

Резервное копирование на основе создания образов выполняется на уровне гипервизора и, по сути, является созданием моментального среза данных виртуальной машины. При выполнении образа создается копия гостевой операционной системы и всех связанных с ней данных (моментальный срез дисковых файлов виртуальной машины), включая состояние виртуальной машины и конфигурации приложений. Резервная копия сохраняется в виде единого файла, называемого *образом*, и этот образ монтируется на отдельной физической машине, выполняющей роль прокси-сервера, которая работает в качестве клиента резервного копирования. Затем программа резервного копирования создает резервные копии этих файлов-образов обычным путем (рис. 10.22). Это существенно разгружает гипервизор при резервном копировании и перекладывает основную нагрузку на прокси-сервер, уменьшая тем самым отрицательное влияние на работу виртуальных машин, запущенных на гипервизоре. Резервное копирование на основе создания образов позволяет проводить быстрое восстановление виртуальной машины.

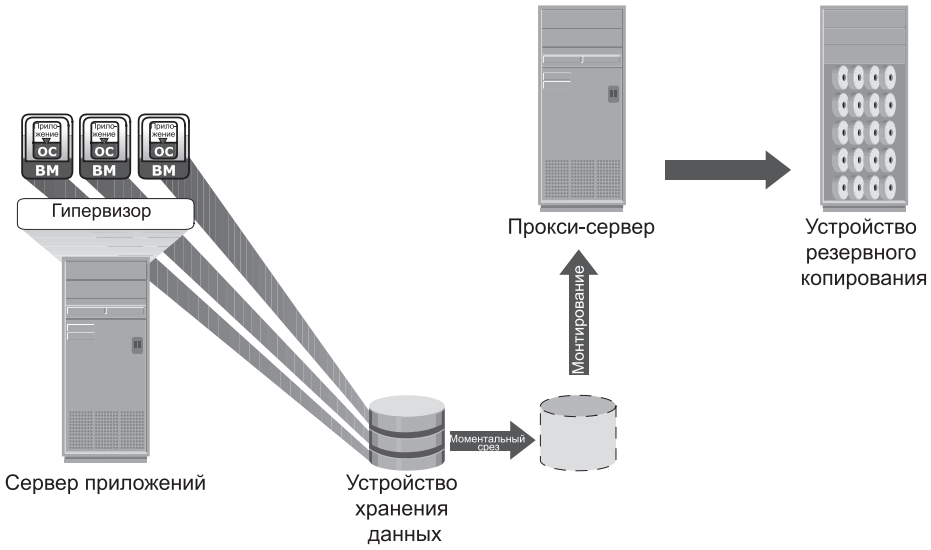


Рис. 10.22. Резервное копирование на основе создания образов

Использование технологии дедупликации данных существенно сокращает объем данных, подлежащих резервному копированию в виртуализированной среде. Эффективность дедупликации проявляется, когда в дата-центре развернуты виртуальные машины с одинаковой конфигурацией.

В виртуализированной среде используются такие же виды и методы дедупликации, как и в физической среде.

10.13. Архивирование данных

В жизненном цикле информации происходит создание данных, обращение к ним и их изменение в активном режиме. По мере того как данные утрачивают свою новизну, вероятность внесения в них изменений снижается, и в конечном итоге они становятся фиксированными, но все еще продолжают оставаться востребованными со стороны приложений и пользователей. Такие данные называются фиксированным контентом. Их примерами могут послужить рентгеновские снимки, сообщения электронной почты и мультимедийные файлы. Примеры фиксированного контента показаны на рис. 10.23.

Рис. 10.23. Примеры данных, относящихся к фиксированному контенту

У всех организаций могут возникать потребности в сохранении данных на длительные сроки из-за требований со стороны властей или в соответствии с юридическими и договорными обязательствами. Кроме того,

фиксированный контент используется организациями для выработки новых стратегий извлечения дохода и улучшения уровня обслуживания. Хранилища, куда помещается фиксированный контент, называются архивами.

Архивные решения могут быть реализованы с постоянным доступом (online), в готовности к доступу (nearline) или с отключенным доступом (offline).

- **Архив с постоянным доступом.** Устройство хранения данных подключено напрямую к хосту, который позволяет получать моментальный доступ к данным.
- **Архив в готовности к доступу.** Устройство хранения данных подключено к хосту, но для доступа к данным само устройство, на котором они хранятся, должно быть смонтировано или установлено в привод.
- **Архив с отключенным доступом.** Устройство хранения данных не готово к использованию. Для его подключения, монтирования или установки носителя в привод устройства требуется ручное вмешательство, и только после этого данные станут доступны.

Традиционно для архивов использовались оптические и ленточные носители данных. Оптические носители выбираются, как правило, с возможностью однократной записи и многократного чтения — write once read many (WORM), что защищает исходный файл от перезаписи. Эту же функцию предоставляют и некоторые ленточные носители, на которых реализованы возможности блокировки файлов. При невысокой стоимости этих устройств их использование влечет за собой накладные расходы, связанные с эксплуатацией, управлением и обслуживанием. Традиционный процесс архивирования, в котором используются оптические диски и ленты, не оптимизирован под распознавание содержимого, поэтому один и тот же контент может попасть в архив несколько раз.

На содержание хранилища носителей данных, находящееся на удаленной площадке, и управление носителями приходится нести дополнительные расходы. Ленты и оптические носители подвержены износу и механическим повреждениям. Частые изменения технологии устройств приводят к дополнительным расходам на перевод носителей в новые форматы, позволяющие получить доступ к данным и извлечь информацию. Правительственные учреждения и организации, выпускающие нормативные акты, создают новые законы и правила для усиления защиты архивов от несанкционированного уничтожения и изменения. Эти правила и стандарты устанавливают новые требования для сохранения целостности информации в архивах. Эти требования выявили недостатки традиционных решений по созданию архивов на основе использования лент и оптических носителей данных.

В качестве альтернативы решениям с использованием лент и оптических носителей появились *контентно-адресуемые хранилища* — content addressed storage (CAS) на магнитных дисках. CAS-хранилища отвечают требованиям

по повышению доступности, а также защите и ликвидации данных и обеспечивают выполнение условий, прописанных в соглашениях об уровне обслуживания — service-level agreements (SLA) для архивных данных. Более подробно CAS-хранилища рассматривались в главе 8.

10.14. Архитектура решений, связанных с архивированием данных

Архитектура решений, связанных с архивацией данных, содержит три основных компонента: агент архивирования, сервер архивирования и устройство хранения архивных данных (рис. 10.24).

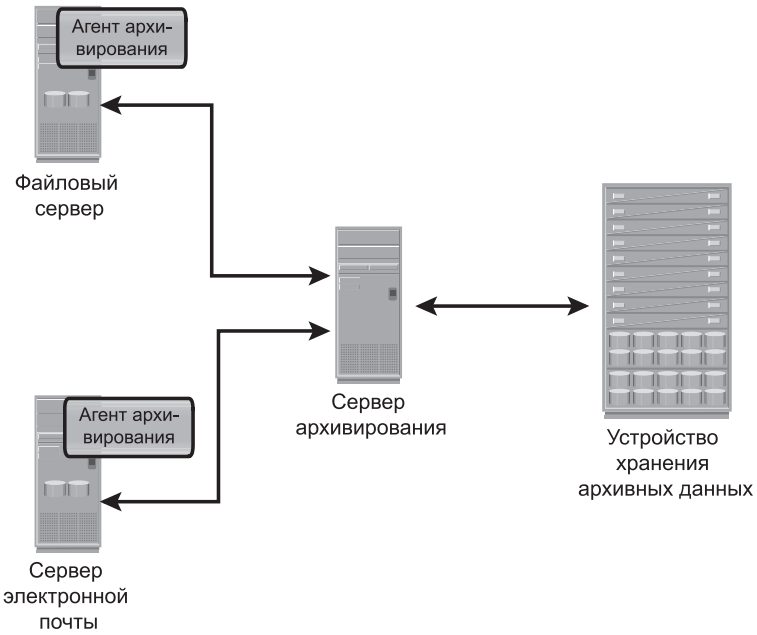


Рис. 10.24. Архитектура решений, связанных с архивированием данных

Агент архивирования представляет собой программу, установленную на сервере приложений. Он отвечает за сканирование данных, которые могут быть помещены в архив на основе политики, определенной на сервере архивирования. После того как данные будут идентифицированы пригодными для архивирования, агент отправляет их на сервер архивирования. Затем исходные данные на сервере приложений заменяются файлом-заглушкой. Этот файл содержит адрес данных, помещенных в архив. У этого файла небольшой размер, что дает существенную экономию места на основном

устройстве хранения данных. Этот файл-заглушка используется для извлечения настоящего файла из устройства хранения архивных данных.

Сервер архивирования представляет собой программу, установленную на хост, которая позволяет администраторам настраивать политики архивирования данных. Политики могут быть определены на основе размера файла, его типа или времени создания, времени изменения или времени обращения к файлу. Сервер архивирования получает от агента данные, подлежащие архивированию, и отправляет их на устройство хранения архивных данных.

Устройство хранения архивных данных хранит фиксированный контент. Для архивирования доступны несколько вариантов различных типов носителей сохраняемых данных, например оптические, ленточные и дешевые дисковые накопители.

10.14.1. Сценарий использования: архивирование электронной почты

Электронная почта может послужить примером приложения, которое получает наибольшие выгоды от архивирования. Обычно системный администратор настраивает на работу небольшие почтовые ящики, в которых хранится ограниченное количество сообщений. Дело в том, что большие ящики с большим количеством сообщений могут усложнить управление, повысить стоимость основного устройства хранения данных и снизить производительность системы. Когда почтовый сервер настроен на большое количество почтовых ящиков, системный администратор обычно определяет квоту каждого почтового ящика, ограничивающую его размер на данном сервере. Настройка фиксированных квот для почтового ящика ущемляет интересы конечных пользователей. Фиксированная квота для почтового ящика заставляет пользователей удалять сообщения при подходе к границе квоты. Конечным пользователям зачастую требуется обращаться к сообщениям, которые пришли несколько недель, месяцев или даже лет назад.

Архивирование сообщений электронной почты предоставляет отличное решение, позволяющее преодолеть ранее обозначенные трудности. В процессе архивирования те сообщения электронной почты, которые были определены в качестве кандидатов на архивирование, перемещаются из основного хранилища данных в устройство хранения архивных данных, причем делается это на основе предопределенной политики, например: «архивированию подлежат сообщения электронной почты, поступившие более чем 90 дней назад». В соответствии с политиками сохранения данных, сообщения электронной почты после архивирования могут храниться годами. Тем самым существенно экономится пространство основного хранилища данных, а организации получают возможность выполнять нормативные требования. Реализация решений архивирования дает в распоряжение пользователей виртуально неограниченное пространство почтового ящика.

10.14.2. Сценарий использования: архивирование файлов

Среда совместного использования файлов также получает преимущества от внедрения решений, позволяющих архивировать данные. Обычно пользователи помещают в общее хранилище весьма большое количество файлов. Большинство этих файлов являются устаревшими, и обращение к ним происходит крайне редко. Администраторы устанавливают квоты на совместно используемые файлы, которые заставляют пользователей удалять устаревшие файлы. Это ущемляет интересы пользователей, поскольку им может потребоваться доступ к файлам, которые попали в хранилище несколько месяцев или даже лет назад. В ряде случаев пользователь может попросить увеличить объем квоты на совместно используемые файлы. А это в свою очередь увеличивает стоимость основного хранилища данных. Решение по архивированию файлов позволяет помещать файлы в архив на основе политики, связанной с временем создания файлов, их размером и т. д. Архивирование позволяет существенно снизить требования к емкости основного хранилища данных, и, кроме того, оно позволяет пользователям долгое время хранить файлы в архиве.

АРХИВИРОВАНИЕ ДАННЫХ В ОБЛАЧНОМ ХРАНИЛИЩЕ

Сегодня для архивирования своих данных организации используют облачные хранилища. Облачное хранилище данных не требует от организаций никаких первоначальных капиталовложений — capital expenditure (CAPEX), таких как закупка аппаратных и программных компонентов для архивирования данных. Организациям нужно платить только за потребленные облачные ресурсы.

Облачные вычисления предоставляют организациям в виде услуги бесконечно масштабируемое хранилище, что позволяет им расширять свои хранилища по мере надобности. Чтобы использовать для архивирования облачное хранилище данных, приложение архивирования должно поддерживать API-интерфейс этого хранилища.

10.15. Практическая реализация концепций: EMC NetWorker, EMC Avamar и EMC Data Domain

Портфолио компании EMC по средствам резервного копирования, восстановления и дедупликации состоит из широкого ассортимента продуктов для постоянно растущих объемов данных, подлежащих резервному копированию. В данном разделе будут кратко представлены такие продукты, как EMC NetWorker, EMC Avamar и EMC Data Domain. Самую свежую информацию о них можно найти на сайте www.emc.com.

10.15.1. EMC NetWorker

Применение программного комплекса EMC NetWorker позволяет проводить централизованное, автоматизированное и ускоренное резервное копирование и восстановление данных в масштабах предприятия. EMC NetWorker обладает следующими основными свойствами:

- поддержка разнородных платформ, таких как Windows, UNIX и Linux, а также поддержкой виртуальных сред;
- поддержка технологий кластеризации и резервного копирования открытых файлов;
- поддержка различных адресатов резервного копирования — лент, дисков и виртуальных лент;
- поддержка мультиплексирования данных (или многопоточности);
- предоставление возможности дедупликации данных с использованием как источника, так и приемника данных за счет объединения с такими продуктами, как EMC Avamar и EMC Data Domain соответственно;
- использование 256-разрядного усовершенствованного стандарта шифрования — AES (advanced encryption standard), обеспечивающего безопасность данных, подвергаемых резервному копированию. Хост-машины NetWorker проходят строгую аутентификацию на основе протокола защищенных сокетов — Secure Sockets Layer (SSL);
- наличие возможности облачного резервного копирования, позволяющего NetWorker создавать резервные копии как в закрытых, так и в публичных облачных конфигурациях.

NetWorker предоставляет централизованное управление средой резервного копирования с использованием графического интерфейса пользователя. Это управление имеет настраиваемую под запросы клиентов систему составления отчетов и контролируемую программой-мастером систему задания конфигурации. При использовании консоли управления — NetWorker Management Console (NMC) резервное копирование легко поддается управлению с любого хоста, имеющего поддержку веб-браузера. NetWorker также предоставляет возможность использования множества утилит командной строки. Для облегчения выполнения административных задач доступность некоторых отчетов обеспечивается через свойство получения отчетов NMC-консоли. Данные, хранящиеся в базе данных NMC-сервера и собираемые с любого или со всех NetWorker-серверов, используются для подготовки отчетов по статистике и состоянию, событиям, хостам, пользователям и устройствам.

10.15.2. EMC Avamar

Продукт EMC Avamar представляет собой решение по резервному копированию и восстановлению данных на основе использования дисков со встроенной системой дедупликации на стороне источника данных. Имеющаяся в данном продукте уникальная функция глобальной дедупликации данных выгодно отличает Avamar от традиционных решений по резервному копированию

и восстановлению данных тем, что в данном решении определяются и сохраняются только уникальные субфайловые объекты данных. Поиск избыточных данных ведется на источнике, за счет чего существенно сокращается объем данных, передаваемых по сети, а также уменьшаются требования к емкости хранилища резервных копий. В Avamar-системе имеются три основных компонента: Avamar-сервер, клиенты резервного копирования Avamar и Avamar-администратор. На Avamar-сервере хранятся резервные копии клиентов, и этим сервером обеспечиваются проведение всех основных процессов и предоставление услуг, требующихся для доступа со стороны клиентов и удаленного управления системой. Программа Avamar-клиента запускается на каждом компьютере или сервере сети, на котором проводится резервное копирование. Avamar-администратор является пользовательским консольным приложением для управления, используемым для удаленного администрирования Avamar-системы. Существуют три версии Avamar-сервера:

- **Software only.** Версия Avamar Software является решением, основанным на использовании исключительно программного обеспечения. Серверная программа устанавливается на аппаратные платформы, предоставляемые клиентом и прошедшие квалификационные испытания с использованием Avamar;
- **Avamar Data Store.** Версия Avamar Data Store включает как аппаратный, так и программный компонент от компании EMC, предназначенный для Avamar-сервера;
- **Avamar Virtual Edition.** Версия Avamar Virtual Edition для VMware является программой Avamar-сервера, который развертывается в виде виртуального устройства.

Продукт EMC Avamar характеризуется:

- **дедупликацией данных**. Это свойство гарантирует, что данные в среде резервного копирования попадают в копию только один раз;
- **системной отказоустойчивостью.** Это свойство обусловливается использованием RAID, RAIN, контрольных точек и репликации данных, предоставляющих гарантии целостности данных и возможности их восстановления после чрезвычайных ситуаций;
- **использованием стандартной IP-сети.** Это свойство позволяет оптимизировать использование сети в целях резервного копирования; специально выделенные сети резервного копирования не требуются. Ежедневное полное резервное копирование возможно при существующих сетях и инфраструктуре;
- **масштабируемой серверной архитектурой.** Это свойство предполагает добавление дополнительных узлов хранения данных без нарушения режима работы к многоузловому Avamar-серверу, чтобы приспособиться к возрастающим требованиям к хранилищу данных резервного копирования;

- **централизованным управлением.** Это свойство позволяет осуществлять удаленное управление Avamar-серверами из единого центра и посредством использования интерфейсов Avamar Enterprise Manager и Avamar Administrator.

10.15.3. EMC Data Domain

Система хранения данных с дедупликацией EMC Data Domain является решением, реализуемым на стороне приемника данных. При использовании высокоскоростной технологии дедупликации на лету система Data Domain обеспечивает результат дедупликации, который в среднем намного меньше исходного набора данных. Система поддерживает различные приложения резервного копирования и корпоративные приложения в средах баз данных, электронной почты, управления содержимым и в виртуальных средах. Системы Data Domain могут масштабироваться, начиная с оборудования небольших удаленных офисов и заканчивая системами крупных дата-центров. Эти системы доступны в виде встроенных устройств или шлюзов, использующих внешнее хранилище данных.

Системы хранения данных с дедупликацией предоставляют следующие уникальные преимущества:

- **архитектуру с неуязвимостью данных.** Эта архитектура предоставляет беспрецедентные уровни целостности данных, проверку достоверности данных и возможности самовосстановления, обеспечиваемые, например, применением массива RAID 6. Постоянное отслеживание сбоев, возможность исправления поврежденных данных и проверка успешного завершения их записи гарантируют точное сохранение резервной копии, доступность данных и возможность их восстановления;
- **архитектуру масштабирования Data Domain SISL (Stream-Informed Segment Layout).** Эта архитектура позволяет наращивать количество центральных процессоров с целью добавления системе непосредственных преимуществ от масштабируемости;
- **технологию поддержки внутренней репликации.** Эта технология позволяет проводить автоматический безопасный перенос сжатых данных по глобальной сети — wide area network (WAN) с минимальными требованиями к ширине полосы пропускания;
- **глобальное сжатие.** В системах применяются технологии высокоэффективной дедупликации и сжатия данных, радикально улучшившие экономические показатели сохранения данных.

EMC Data Domain Archiver является решением для долгосрочного хранения данных резервного копирования и архивирования. Этот продукт разработан с внутренним многоуровневым подходом к обеспечению экономически эффективного долгосрочного хранения данных на диске путем реализации технологии дедупликации данных.

Резюме

Доступность информации является одним из самых важных требований для информационно-ориентированных предприятий. Резервные копии защищают предприятия от потери данных, а также помогают им отвечать требованиям нормативов и соглашений.

Архивирование данных позволяет IT-организациям дополнительно сэкономить средства и повысить эксплуатационную эффективность. Архивирование данных позволяет соответствовать требованиям законодательства, что помогает организациям избежать штрафов и осложнений, связанных с проблемами их выполнения.

В данной главе были подробно рассмотрены факторы, определяющие порядок резервного копирования данных, методы, технологии и реализации резервного копирования в среде хранилищ, доступных по сети. Были также детально рассмотрены различные топологии резервного копирования, используемые архитектуры, дедупликация данных и резервное копирование в виртуализированных средах. Кроме того, в главе подробно рассмотрена архитектура решений архивирования данных. Хотя вопросы выбора конкретного носителя резервных копий решаются путем определения показателей RTO и RPO, резервное копирование на основе использования дисков имеет явные преимущества над резервным копированием с использованием ленточных носителей, выражаемые в показателях производительности, доступности, скорости восстановления данных и простоты управления. Эти преимущества еще больше усиливаются при использовании технологий репликации данных с целью достижения самого высокого уровня обслуживания и выполнения требований по доступности данных. Более подробно технологии репликации рассматриваются в следующих двух главах.

УПРАЖНЕНИЯ

1. Клиент выполняет полное резервное копирование в первое воскресенье месяца, за которым следуют накопительные резервные копирования в другие воскресные дни. Также ежедневно с понедельника по субботу выполняется инкрементное резервное копирование. С целью получения возможности восстановления данных после возникновения чрезвычайной ситуации каждое утро в 10:00 ленты отправляются в другое место. В среду третьей недели в 15:00 у клиента произошел системный сбой, потребовавший восстановления системы. За сколько дней нужно будет взять ленты для восстановления данных?
2. В NAS-среде совместного использования файлов имеется ограниченное количество устройств резервного копирования. Предложите подходящую реализацию резервного копирования, с помощью которой можно свести к минимуму сетевой трафик, избежать любого затора данных и в то же время не помешать проведению производственных операций. Обоснуйте ответ.

3. Какие бизнес- и технологические факторы следует рассмотреть для реализации решений по выполнению резервного копирования и как эти рассматриваемые факторы влияют на выбор решения по выполнению резервного копирования и варианта его реализации?
4. Перечислите факторы, оправдывающие применение магнитной ленты в технологии резервного копирования, и объясните свой выбор. С какими трудностями приходится сталкиваться в такой среде?
5. Опишите преимущества использования виртуальной ленточной библиотеки по сравнению с использованием физической ленточной библиотеки.
6. Исследуйте преимущества и проблемные вопросы использования облачного хранилища данных для архивирования и подготовьте презентацию на эту тему.

Глава 11
Локальная репликация

В современной бизнес-среде организациям необходимо защищать важные для ведения бизнеса данные и сводить к минимуму риски нарушения бизнес-процесса. Если происходит локальный сбой или стихийное бедствие, то для обеспечения непрерывности бизнес-процессов важнейшим вопросом становится быстрое восстановление данных и перезапуск бизнес-операций. Одним из способов обеспечения непрерывности бизнес-процессов является репликация данных, представляющая собой процесс создания точной копии (реплики) данных. Точные копии используются для операций восстановления данных и перезапуска бизнес-операций после потери данных. Эти точные копии могут быть также предназначены для других хостов с целью выполнения различных бизнес-операций, например создания резервной копии, составления отчета и проведения тестов.

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Согласованность данных
- Локальная репликация на основе использования хоста
- Локальная репликация на основе использования массива хранения данных
- Копирование при первом обращении (CoFA)
- Копирование при первой записи (CoFW)
- Локальная репликация на основе использования сети
- Особенности восстановления и перезапуска
- Репликация виртуальной машины

Репликации можно разбить на две основных категории: локальные и удаленные. Локальные репликации относятся к репликациям данных, проводимым в одном и том же массиве данных или в одном и том же дата-центре. Удаленные репликации относятся к репликациям данных в удаленном месте. В данной главе будут подробно рассмотрены различные технологии локальной репликации, а также факторы, определяющие порядок проведения восстановления данных и перезапуска бизнес-операций. Кроме того, в ней будут подробно рассмотрены вопросы репликации в виртуальной среде. Удаленные репликации рассматриваются в главе 12.

11.1. Терминология репликаций

Для представления различных объектов и операций в среде проведения репликаций используются следующие общепринятые термины:

- **источник.** Хост, получающий доступ к производственным данным с одного или нескольких LUN-устройств в массиве хранения данных, называется *производственным хостом*, а такие LUN-устройства называются LUN-источниками (ими могут быть устройства или дисковые тома), производственными LUN-устройствами или просто *источниками*;
- **приемник.** Одно или несколько LUN-устройств, на которые копируются данные, называются целевым LUN-устройством, или просто *приемником*, или репликой;
- **моментальная (Point-in-Time, PIT) и непрерывная реплика.** Реплики могут быть либо моментальными (PIT-реплики), либо непрерывными. PIT-реплика является точным образом источника на какой-то конкретный момент времени. Например, если реплика файловой системы создана в 16:00 в понедельник, то это реплика является PIT-копией на 16:00 понедельника. Непрерывная реплика все время находится в синхронизированном состоянии с производственными данными;
- **пригодность для восстановления данных и перезапуска бизнес-операций.** Пригодность для восстановления данных позволяет восстанавливать данные из реплики на их источнике при потере данных или их повреждении. Пригодность для перезапуска бизнес-операций позволяет перезапускать бизнес-операции с использованием реплик. Чтобы реплику можно было применять как для восстановления данных, так и для перезапуска бизнес-операций, она должна быть согласована с источником. Согласованность реплик подробно рассматривается в одном из следующих разделов.

СРАВНЕНИЕ РЕПЛИКИ С РЕЗЕРВНОЙ КОПИЕЙ

Приложения имеют к репликам немедленный доступ, а для того чтобы приложениям стала доступна резервная копия, программа резервного копирования должна сначала провести восстановление данных из этой копии. Резервные копии делаются всегда на определенный момент времени, то есть они являются PIT-копиями, а реплики могут как быть PIT-копиями, так и делаться непрерывно. Резервное копирование обычно используется для оперативного восстановления или восстановления данных после чрезвычайного происшествия, а реплики могут использоваться для восстановления данных и перезапуска бизнес-операций, а также для проведения других бизнес-операций, например таких как резервное копирование, составление отчетов и тестирование. По сравнению с восстановлением данных из резервной копии реплики обычно обеспечивают более низкий показатель RTO.

11.2. Использование локальных реплик

Создание одной или нескольких локальных реплик на основе данных источника может проводиться с различными целями, в том числе:

- **для создания альтернативного источника резервного копирования.** При обычных операциях резервного копирования данные считываются с производственных томов и записываются на устройство резервного копирования. Это создает дополнительную нагрузку на производственную инфраструктуру, поскольку производственные LUN-устройства одновременно занимаются как производственными операциями, так и обслуживанием данных для операций резервного копирования. В локальной реплике содержится точная PIT-копия источника данных, поэтому ее можно использовать в качестве источника для проведения резервного копирования. Этим можно уменьшить нагрузку на производственные тома, связанную с операциями ввода-вывода, необходимыми для резервного копирования. Еще одно преимущество использования реплик для резервного копирования заключается в том, что *окно резервного копирования* сводится к нулю;
- **быстрого восстановления данных.** Локальную реплику можно использовать для восстановления данных источника в случае их потери или повреждения. Если произойдет полный отказ источника, некоторые технические решения по проведению репликации данных позволяют использовать реплику для восстановления данных на другом наборе исходных устройств или же перезапускать производство на самой реплике. В любом случае, по сравнению с традиционным восстановлением с ленточных резервных копий данный метод обеспечивает более быстрое восстановление данных и минимальный показатель RTO. Во многих случаях бизнес-операции могут быть запущены с использованием устройства, послужившего источником данных, еще до того, как данные будут полностью скопированы с реплики;
- **выполнения действий, помогающих принимать правильные решения, например составления отчетов или переноса данных в хранилище.** Составление отчетов с использованием данных реплик существенно снижает нагрузку на производственное устройство, связанную с вводом-выводом данных. Локальные реплики также используются приложениями, переносящими данные в хранилища, которые могут получать эти данные из реплики, не вмешиваясь в работу производственной среды;
- **в качестве платформы для тестирования.** Локальные реплики могут использоваться для тестирования новых или обновленных приложений. Например, организация может использовать реплику для тестирования обновления производственного приложения. Если тест

будет пройден успешно, обновлением можно будет воспользоваться в производственной среде;

- **для миграции данных.** Локальная реплика может использоваться также для миграции данных, выполняемой по различным причинам, например для переноса данных из LUN-устройств меньшей емкости в более емкие LUN-устройства с целью использования самых последних версий приложения.

11.3. Согласованность реплик

В большинстве файловых систем и баз данных до записи на диск данные помещаются в буфер хоста. Создание согласованной реплики дает гарантию, что данные, находящиеся в буфере хоста, были записаны на диск. Перед созданием реплики те данные, которые накопились в кэше и еще не были записаны на диск, должны быть сброшены на этот диск. До инициирования операции создания реплики среда выполнения операций в массиве хранения данных сбрасывает данные из своей кэш-памяти на диск. Обеспечение согласованности гарантирует пригодность реплики и является главным требованием для всех технологий репликации данных.

11.3.1. Согласованность реплицированных файловых систем

Для сокращения времени отклика приложений файловые системы осуществляют буферизацию данных в памяти хоста. Данные, попавшие в буфер, периодически записываются на диск. В операционных системах семейства UNIX сбросом данных из буферов на диск через определенные интервалы времени занимается *демон синхронизации*. В некоторых случаях реплика создается между установленными интервалами, что может привести к созданию ее несогласованного экземпляра. Поэтому для обеспечения согласованности данных реплики буферы памяти хоста должны перед ее созданием сбрасываться на диск. На рис. 11.1 показано, как перед репликацией данных буфер файловой системы сбрасывается на дисковое устройство источника. Если буферы памяти хоста не сброшены, в данных реплики не будет содержаться информация, которая была занесена в буфер хоста. Если файловая система размонтируется до создания реплики, буферы автоматически будут сброшены на диск и реплика будет содержать согласованные данные.

Если репликация данных происходит на смонтированной файловой системе, то на ней нужно провести некоторые действия по ее восстановлению, например запустить *программу поиска и исправления ошибок — fsck* или запустить *воспроизведение действий по журналу операций — log replay*. После завершения процесса репликации и проверки файловой системы реплицированная файловая система может быть смонтирована для ее оперативного использования.

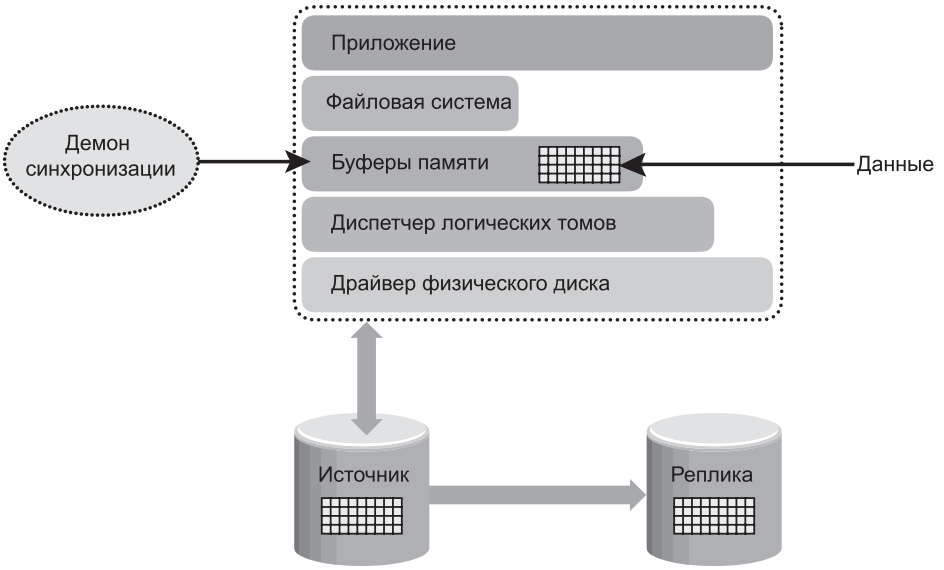


Рис. 11.1. Сброс на диск буфера файловой системы

11.3.2. Согласованность реплицированных баз данных

База данных может быть рассредоточена по множеству различных файлов, файловых систем и устройств. Все они должны быть согласованным образом реплицированы, чтобы можно было гарантировать пригодность реплики для восстановления данных и перезапуска бизнес-операций. Репликация базы данных может быть выполнена в момент ее нахождения как в подключенном, так и в отключенном режиме. Если при создании реплики база данных находится в отключенном режиме, то для операций ввода-вывода она не доступна. Поскольку в источнике не происходит никаких обновлений, реплика получается согласованной.

Если же база данных находится в подключенном режиме, она доступна для проведения операций ввода-вывода и проводимые в ней транзакции постоянно ее обновляют. При репликации базы данных, находящейся в подключенном режиме, все изменения, сделанные в базе данных в ходе репликации, должны быть применены к реплике, иначе она будет несогласованной. Согласованная реплика подключенной базы данных создается путем использования принципа зависимой записи при операциях ввода-вывода или путем мгновенного проведения операций ввода-вывода в отношении источника перед созданием реплики.

Принцип применения *зависимой записи при проведении операций ввода-вывода* с целью обеспечения согласованности свойственен многим приложениям и системам управления базами данных — database management systems (DBMS). Согласно этому принципу команда на запись при проведении

операции ввода-вывода не выдается приложением до тех пор, пока не будет завершена предыдущая связанная с ней запись, проводимая в ходе операции ввода-вывода. Например, запись данных зависит от успешного завершения предыдущей записи в журнал транзакций.

Чтобы транзакция считалась завершенной, необходимо, чтобы в базе данных в конкретной последовательности была произведена целая серия записей. Эти записи будут вестись на различные устройства или в различных файловых системах. Процесс сброса данных, находящихся в буфере, с хоста на источник показан на рис. 11.2. Чтобы данная транзакция считалась завершенной, должны завершиться операции ввода-вывода с *1-й* по *4-ю*. Операция ввода-вывода *4* зависит от операции ввода-вывода *3* и осуществляется только при завершении этой операции. Операция ввода-вывода *3* зависит от операции ввода-вывода *2*, которая, в свою очередь, зависит от операции ввода-вывода *1*. Каждая операция ввода-вывода завершается только после завершения предыдущей операции или операций ввода-вывода.

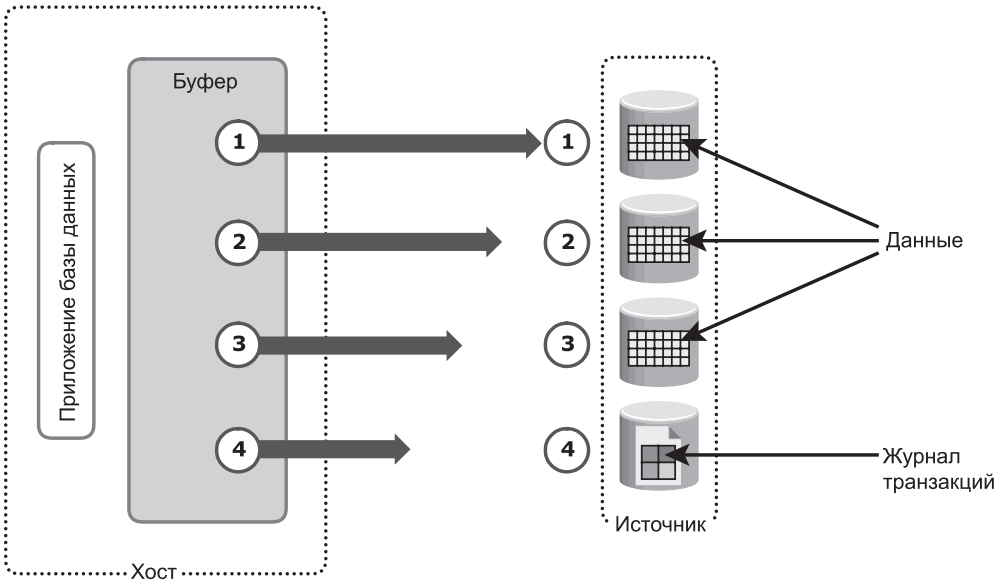


Рис. 11.2. Последовательность зависимых записей на источнике

Чтобы обеспечить согласованность данных, в момент создания реплики все записи на устройствах-источниках должны попасть на устройства реплики. Процесс репликации данных из источника в реплику показан на рис. 11.3. Чтобы в реплику попали согласованные данные, в отношении этих данных должны быть выполнены транзакции ввода-вывода с *1-й* по *4-ю*.

Может случиться так, что транзакции ввода-вывода *3* и *4* были скопированы на устройства реплики, а транзакции ввода-вывода *1* и *2* не были скопированы. Данная ситуация показана на рис. 11.4. В таком случае данные

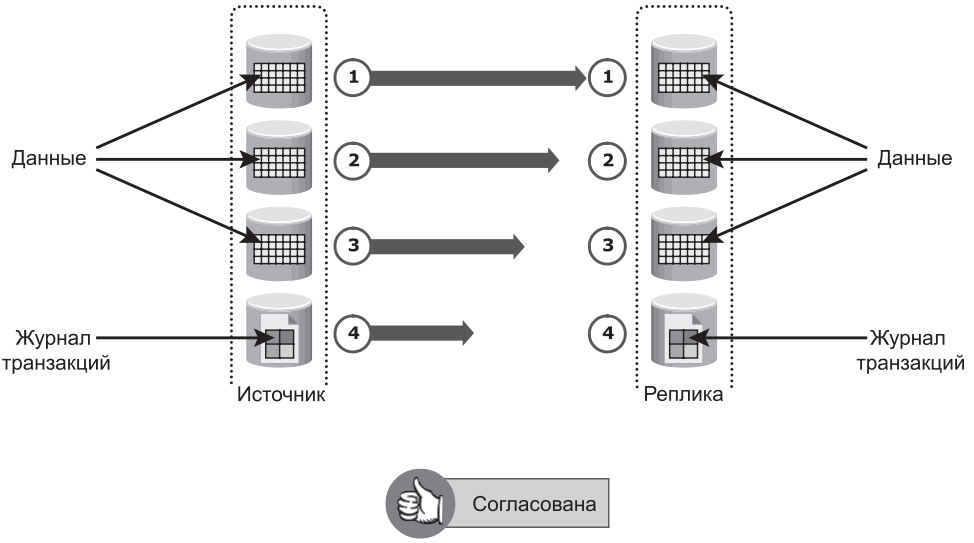


Рис. 11.3. Согласованность реплики, обеспечиваемая зависимой записью

в реплике не будут согласованы с данными в источнике. Если перезапуск бизнес-операций должен выполняться на устройствах, где находится реплика, наличие транзакции ввода-вывода *4*, которая доступна в реплике, может свидетельствовать о том, что соответствующая транзакция была завершена, но все данные, связанные с этой транзакцией, будут в реплике недоступны, что сделает ее несогласованной.

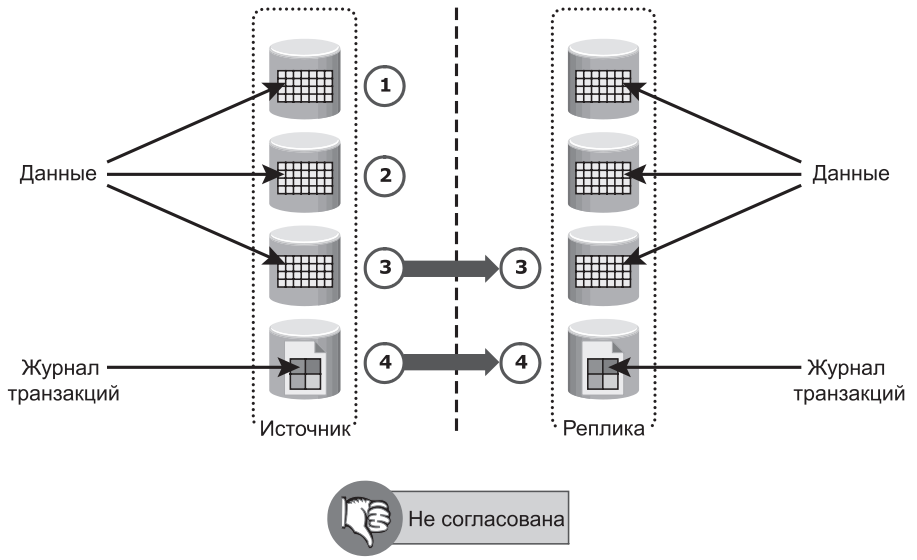


Рис. 11.4. Несогласованная реплика базы данных

Еще один способ обеспечения согласованности заключается в обеспечении гарантии того, что операции записи в рамках процедур ввода-вывода на все устройства источника на время создания реплики приостанавливаются. Благодаря этому создается согласованный образ реплики. Но если остановка будет длиться слишком долго, то у баз данных и приложений может закончиться время ожидания отклика.

11.4. Технологии локальных репликаций

Основными технологиями локальных репликаций являются репликации, основанные на использовании хоста, массива хранения данных и сети. Примерами локальных репликаций на основе использования хоста являются репликации файловых систем и репликации с использованием диспетчера логических томов (LVM). Репликация на основе использования массива хранения данных может применяться при реализации отдельных решений, в частности, зеркального копирования всего тома, репликации всего тома на основе использования указателей и виртуальной репликации на основе использования указателей. А примером репликации на основе использования сети может послужить непрерывная защита данных — continuous data protection (CDP), рассматриваемая в одном из следующих разделов.

11.4.1. Локальная репликация на основе использования хоста

При репликации на основе использования хоста самыми распространенными методами являются репликация с использованием диспетчера логических томов (LVM) и создание моментального снимка файловой системы.

Репликация с использованием LVM

При репликации на основе использования LVM за создание логическими томами на уровне хоста и управление ими отвечает диспетчер логических томов. У LVM имеются три компонента: физические тома (физический диск), группы томов и логические тома. *Группа томов* создается за счет объединения одного или нескольких физических томов. *Логические тома* создаются в заданной группе томов. У группы томов может быть несколько логических томов.

При репликации на основе использования LVM каждый *логический блок* в логическом томе отображается на два физических блока в двух различных физических томах (рис. 11.5). Когда приложение ведет запись в логический раздел, драйвер устройства LVM записывает данные сразу в два физических раздела. Эта особенность также известна как *LVM-зеркалирование*. Зеркала могут быть разделены, и к содержащимся в них данным может осуществляться независимый доступ.

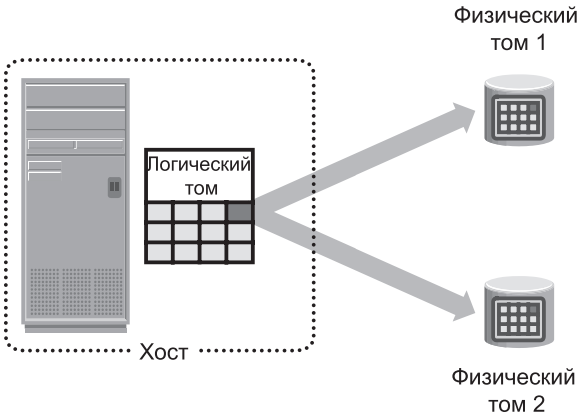


Рис. 11.5. Зеркалирование на основе использования LVM

Преимущества репликации на основе использования LVM

Технология репликации на основе использования LVM не зависит от особенностей системы хранения данных, задаваемых тем или иным поставщиком. Обычно LVM являются частью операционной системы и для LVM-зеркалирования дополнительной лицензии не требуется.

Ограничения, накладываемые на репликацию на основе использования LVM

Любая запись, осуществляемая приложением, превращается в две записи на диск, тем самым увеличивая нагрузку на центральный процессор хоста, что может снизить производительность приложения. Предоставить локальную реплику, созданную с использованием LVM, другому хосту обычно не представляется возможным, поскольку реплика по-прежнему будет частью группы томов, доступ к которой в любой отдельно взятый момент времени имеет, как правило, только один хост.

Отслеживание изменений в зеркалах и выполнение операций инкрементной ресинхронизации также являются проблемными вопросами, поскольку инкрементная ресинхронизация не поддерживается ни одним LVM-диспетчером. Если устройства уже защищены каким-либо из RAID-уровней, реализованным в массиве, то надобность в дополнительной защите путем LMV-зеркалирования отпадает. Это решение не предусматривает возможности масштабирования и поэтому не позволяет создавать реплики объединенных баз данных и приложений. И реплика, и источник хранятся в одной и той же группе томов. Следовательно, при возникновении ошибки в группе томов реплика может стать недоступной. А при отказе сервера, пока он не будет возвращен в строй, будут недоступны как источник, так и реплика.

Объединенная база данных представляет собой коллекцию баз данных, работающую как единое целое. Каждая отдельно взятая база данных в объединенной базе является самодостаточной и полнофункциональной. При получении запроса объединенная база данных направляет его к той базе данных, в которой содержится запрашиваемая информация. Приложение видит объединенную базу данных как единое целое. Тем самым исключается надобность в отправке запросов к нескольким базам данных и объединении полученных результатов.

Моментальный снимок файловой системы

Моментальный снимок файловой системы (FS) — это реплика на основе использования указателей, для которой требуется часть пространства, используемого для производственной файловой системы. Этот моментальный снимок может использоваться либо самой файловой системой, либо LVM-диспетчером. При создании моментального снимка применяется принцип копирования при первой записи (CoFW).

Когда делается моментальный снимок файловой системы (Snap FS), в его метаданных создаются битовый и блочный массивы. Битовый массив используется для отслеживания блоков, претерпевших изменения в производственной FS после создания моментального снимка. Блочный массив используется для указания точного адреса, с которого нужно считывать данные, когда они доступны из моментального снимка файловой систем (Snap FS). Сразу после создания моментального снимка файловой системы все считывания из этого снимка на самом деле обслуживаются за счет чтения производственной файловой системы. Когда используется механизм CoFW, то при выдаче первого после создания моментального снимка запроса на запись в рамках операции ввода-вывода в отношении производственной файловой системы эта операция удерживается и в моментальный снимок перемещаются исходные данные производственной файловой системы, соответствующие месту предполагаемой записи. Затем разрешается сделать запись в производственную файловую систему. В соответствии с этим обновляются битовый и блочный массивы. Последующие записи в то же самое место активности механизма CoFW не вызывают. При чтении из Snap FS происходит обращение к битовому массиву. Если соответствующий бит в нем установлен в нуль, запрос на чтение направляется к производственной файловой системе. Если бит установлен в 1, то адрес блока извлекается из блочного массива и данные считываются с полученного адреса в Snap FS. Запросы на чтение из производственной файловой системы обрабатываются в обычном режиме.

Операция записи в производственную файловую систему показана на рис. 11.6. В приводимом на рисунке примере в блок 3 производственной файловой системы, в котором в данный момент содержатся данные *c*, осуществляется запись данных *C*. Приложение, создающее моментальный снимок,

удерживает операцию ввода-вывода в производственную файловую систему и сначала копирует старые данные *c* в доступный блок в Snap FS. В метаданных моментального снимка изменяются значения битового и блочного массива для блока *3* производственной файловой системы. Бит, соответствующий блоку *3*, устанавливается в 1, показывая, что этот блок в производственной файловой системе претерпел изменения. В блочном массиве запись, соответствующая блоку *3*, изменяется и показывает номер блока в Snap FS, куда были записаны данные (в данном случае это блок *2*). После выполнения этих действий операции ввода-вывода в производственную файловую систему разрешается завершить свою работу. Любые последующие записи в блок *3* производственной базы данных происходят в обычном режиме, без вызова операции CoFW. Точно так же, если операция ввода-вывода запрошена в отношении блока *4* производственной файловой системы с целью замены значения данных *d* значением *D*, приложение, создающее моментальный снимок, удерживает операцию ввода-вывода в производственную файловую систему и копирует старые данные в доступный блок в Snap FS. Затем бит, соответствующий блоку *4* в битовом массиве, устанавливается в 1, показывая, что данные блока в производственной файловой системе изменились. В блоке блочного массива, соответствующем блоку *4*, показывается номер того блока, в котором данные могут быть найдены в Snap FS (в данном случае это блок *1*). После того как это будет сделано, операции ввода-вывода в производственную файловую систему будет разрешено завершиться.

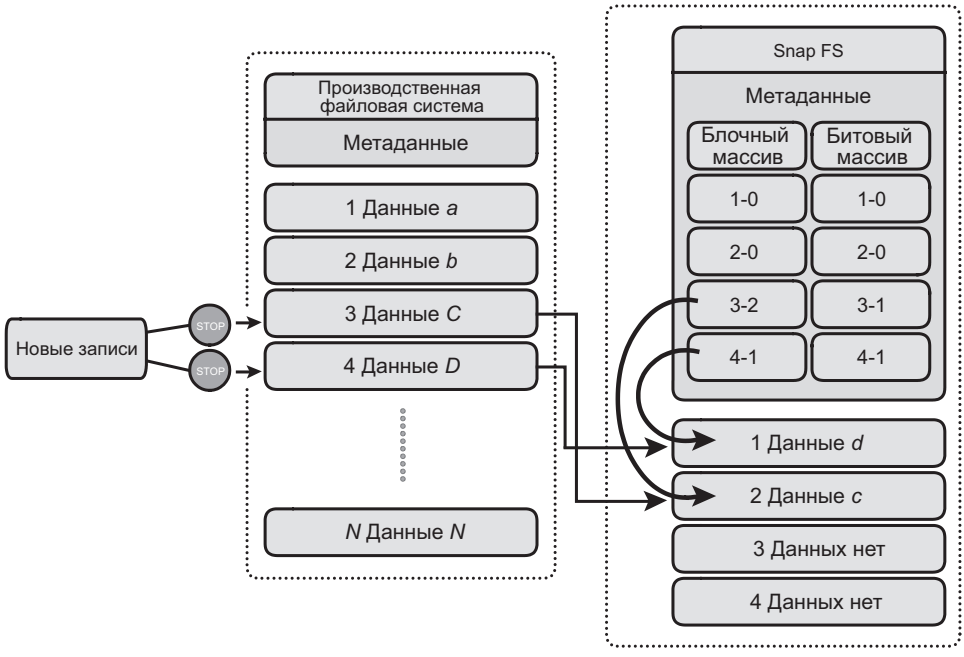


Рис. 11.6. Запись в производственную файловую систему

11.4.2. Репликация на основе использования массива хранения данных

При *локальной репликации на основе использования массива хранения данных* процесс локальной репликации выполняет операционная среда массива. Ресурсы хоста, центральный процессор и память в процессе репликации не используются. Следовательно, хост операциями репликации не загружен. Для проведения других бизнес-операций к реплике можно обратиться с другого хоста.

При проведении данной репликации нужно в том же массиве выбрать требуемое количество устройств репликации, после чего реплицировать данные между парами источник-реплика. На рис. 11.7 показана локальная репликация на основе использования массива хранения данных, где источник и приемник находятся в одном и том же массиве и к ним осуществляется доступ с разных хостов.

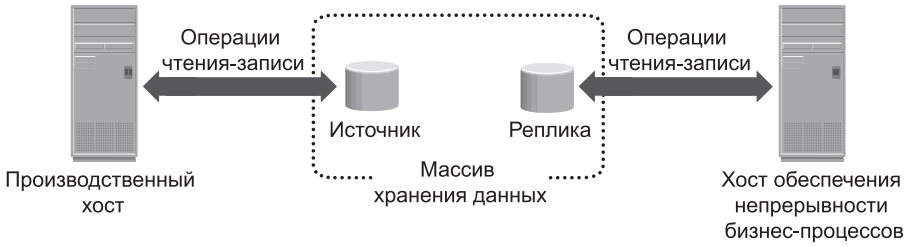


Рис. 11.7. Репликация на основе использования массива хранения данных

Локальная репликация на основе использования массива хранения данных обычно реализуется тремя способами: зеркальным копированием всего тома, репликацией всего тома на основе использования указателей и виртуальной репликацией на основе использования указателей. Устройства репликации, которые также называются приемными устройствами, доступны другим хостам.

Зеркальное копирование всего тома

При создании *зеркальной копии всего тома* приемник подключается к источнику и на нем создается зеркало источника (рис. 11.8, *а*). Данные источника копируются на приемник. При обновлении данных источника обновляются и соответствующие данные приемника. После копирования всех данных и возникновения ситуации, при которой на источнике и приемнике содержатся одинаковые данные, приемник может считаться зеркалом источника.

Пока приемник подключен к источнику, он остается недоступным всем остальным хостам. Но производственный хост продолжает иметь к нему доступ.

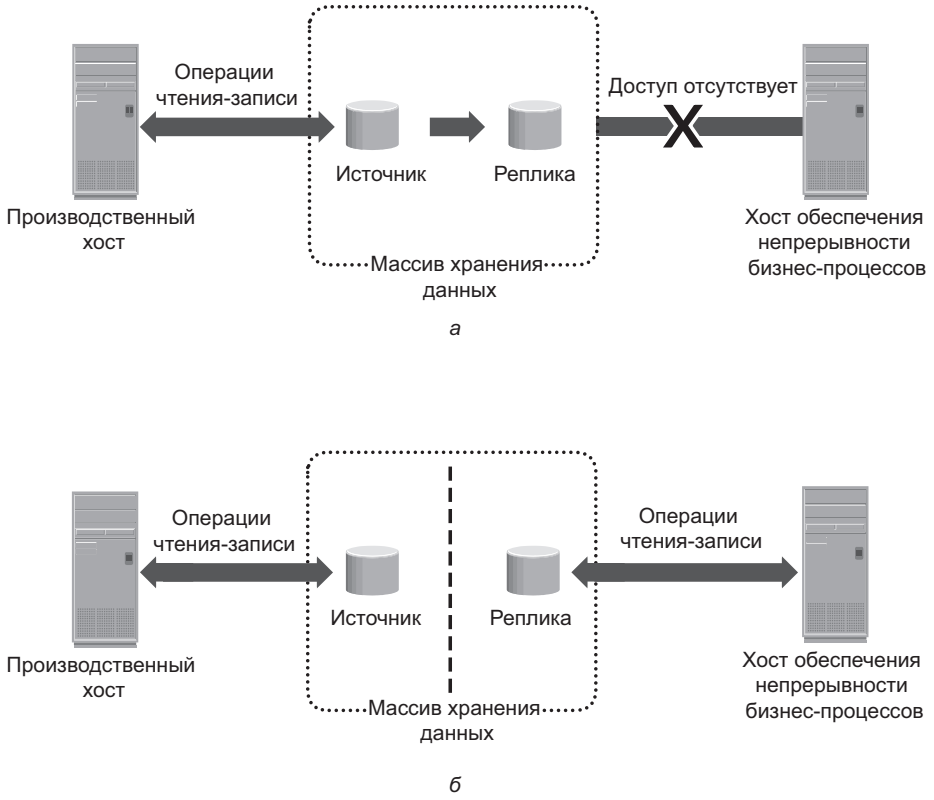


Рис. 11.8. Создание зеркальной копии всего тома: *а* — зеркалирование всего тома с источником, подключенным к реплике; *б* — зеркалирование всего тома с источником, отключенным от реплики

После завершения синхронизации приемник может быть отключен от источника и стать доступным для проведения других бизнес-операций. На рис. 11.8, *б*, показано зеркалирование всего тома с приемником, отключенным от источника. Доступ для проведения операций чтения и записи открыт как к источнику, так и к приемнику для производственного хоста и хоста обеспечения непрерывности бизнес-процессов соответственно.

После отключения от источника приемник становится PIT-репликой источника. Момент создания реплики определяется временем отключения источника от приемника. Например, если время отключения 16:00, то PIT для приемника имеет значение 16:00.

После отключения источника от приемника изменения, вносимые в источник и реплику, могут быть отслежены поблочно с использованием заранее определенного размера блоков. Это позволяет провести инкрементную ресинхронизацию (с источника на приемник) или инкрементное восстановление (с приемника на источник). Степень детализации отслеживания изменений

в данных может варьироваться в зависимости от размера блоков, который может быть от 512 байт до 64 Кбайт и более.

Репликация всего тома на основе использования указателей

Еще одним методом создания локальной реплики на основе использования массива данных является *репликация всего тома на основе использования указателей*. Как и зеркалирование всего тома, эта технология может обеспечить создание в приемниках полных копий данных источника. В отличие от зеркалирования всего тома, приемник становится доступен хосту обеспечения непрерывности бизнес-процессов сразу же после активации сеанса репликации. Поэтому для обращения к реплике не нужно ждать синхронизации и отключения приемника. Момент создания реплики (PIT) определяется временем активации сеанса репликации.

Репликация всего тома на основе использования указателей может быть активирована либо в режиме копирования при первом обращении — Copy on First Access (CoFA), либо в режиме полного копирования. В любом случае во время активации для всех данных на устройствах-источниках создается битовый массив защиты. С помощью этого массива отслеживаются все изменения, происходящие на устройстве-источнике. А чтобы создать отображение на соответствующие блоки данных на источнике, на приемнике инициализируются указатели. Затем на основе режима активации данные копируются с источника на приемник. В режиме копирования при первом обращении (CoFA) после запуска сеанса репликации данные копируются с источника на приемник только при возникновении следующих условий:

- выдача первого запроса на проведение операции ввода-вывода, связанной с записью данных по конкретному адресу источника;
- выдача первого запроса на проведение операции ввода-вывода, связанной с чтением или записью данных по конкретному адресу приемника.

Когда после активации сеанса репликации выдается первый запрос записи на источник, исходные данные, находящиеся по адресу запроса, копируются в приемник, а запись новых данных в источник происходит уже после этой операции. Тем самым гарантируется, что в приемнике сохраняются исходные данные, имевшиеся на момент активации или на PIT-момент (рис. 11.9).

Когда после активации сеанса репликации выдается самый первый запрос на чтение с источника, исходные данные копируются с источника на приемник и становятся доступны хосту обеспечения непрерывности бизнес-процессов (рис. 11.10).

Когда после активации сеанса репликации в приемник поступает первый запрос на запись, на приемник с источника копируются исходные данные, после чего в приемник на место исходных записываются новые данные. (рис. 11.11).

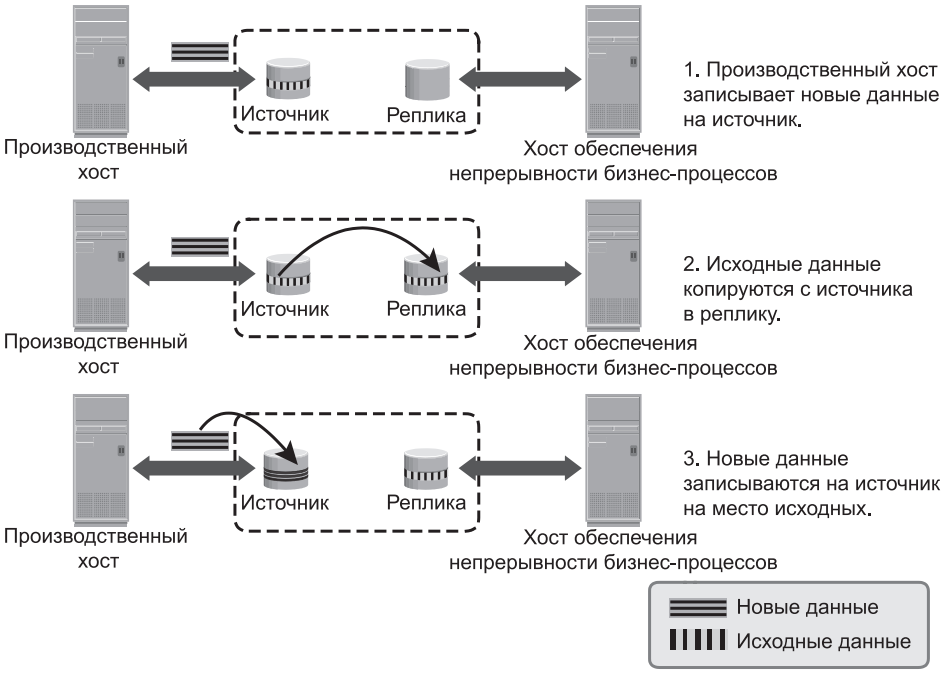


Рис. 11.9. Копирование при первом обращении (CoFA) — запись в источник

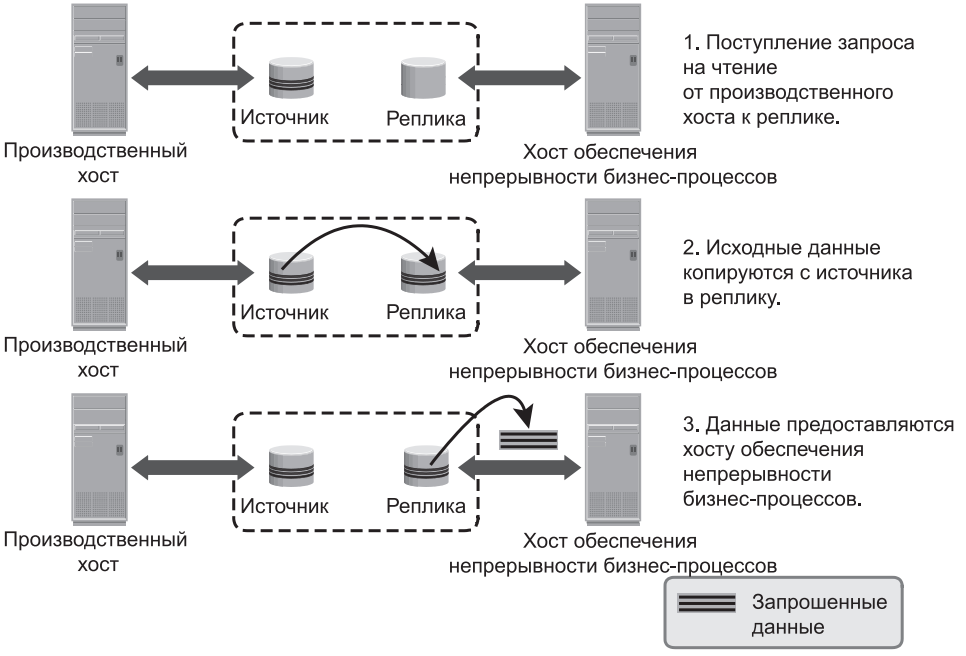


Рис. 11.10. Копирование при первом обращении (CoFA) — чтение из приемника

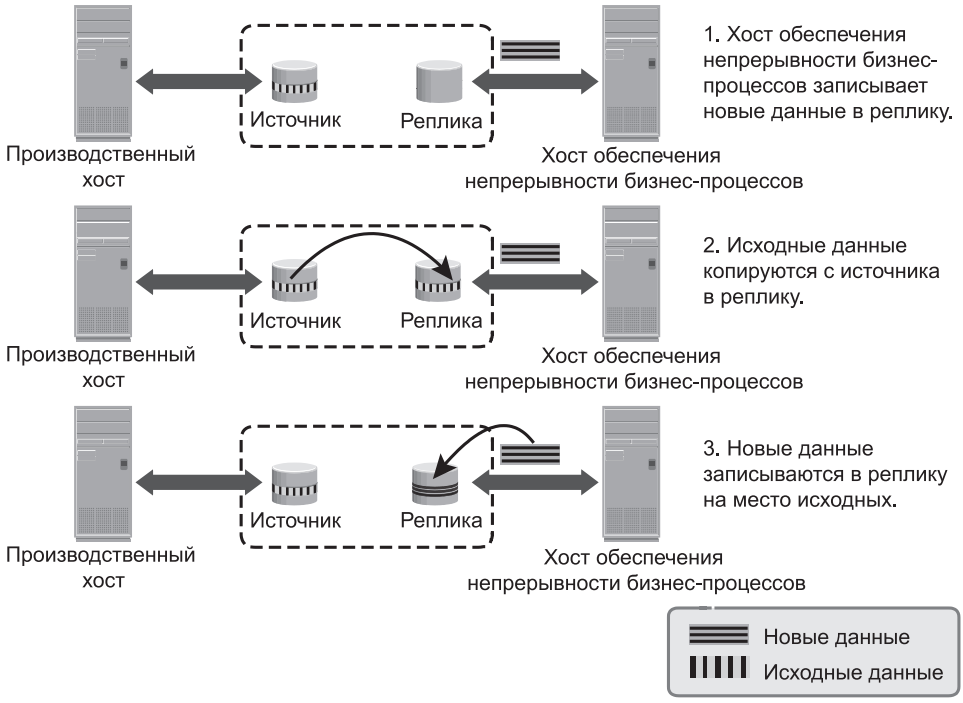


Рис. 11.11. Копирование при первом обращении (CoFA) — запись в приемник

Во всех случаях для блока данных в источнике устанавливается бит защиты, показывающий, что исходные данные были скопированы в приемник. Теперь указатель на данные источника может быть сброшен. Последующие записи в тот же блок данных на источнике, а также проведение операций чтения или записи в те же блоки данных на приемнике не станут основанием для проведения операции копирования, поэтому данный метод и называется «Копирование при первом обращении».

Если сеанс репликации прекращается, на устройстве-приемнике остаются только те данные, к которым было обращение до прекращения сеанса, а не полное содержимое источника на момент создания реплики (PIT-момент). В таком случае данные на приемнике не могут использоваться для восстановления, потому что они не являются полной репликой источника.

В режиме создания полной копии все данные из источника копируются в приемник в фоновом режиме. Данные копируются независимо от обращения к ним. Если нужно обратиться к блоку, который еще не был скопирован в приемник, то сначала этот блок копируется в приемник. При отработке полного цикла в режиме создания полной копии на приемник копируются все данные с источника. Если сеанс репликации будет прекращен уже после полной отработки цикла, на приемнике будут находиться все исходные данные, которые были на источнике на момент активации (PIT-момент).

Поэтому приемник станет вполне пригодной копией для восстановления данных или проведения других операций по обеспечению непрерывности бизнес-процессов.

Главное отличие режима полного копирования на основе использования указателей от создания зеркальной копии всего тома заключается в том, что в режиме полного копирования приемник становится доступным сразу же после активации сеанса репликации. Обе технологии, и создания зеркальной копии всего тома, и репликации всего тома на основе использования указателей, требуют того, чтобы устройства-приемники были как минимум более емкими, чем устройства-источники. Следует также добавить, что при создании зеркальной копии всего тома и репликации всего тома на основе использования указателей в режиме полного копирования системе доступны возможности проведения инкрементной ресинхронизации и восстановления данных.

Виртуальная репликация на основе использования указателей

При *виртуальной репликации на основе использования указателей* во время активации сессии в приемнике содержатся указатели на размещение данных в источнике. При этом приемник не содержит данных, поэтому его называют *виртуальной репликой*. Так же, как и при репликации всего тома на основе использования указателей, приемник становится доступным сразу же после активации сеанса репликации. Для всех блоков данных устройства-источника создается защитный битовый массив. Степень разбиения на блоки может варьироваться, при этом блоки могут иметь размер от 512 байт до более чем 64 Кбайт.

В виртуальной репликации на основе использования указателей применяется технология копирования при первой записи — CoFW. При поступлении первого после активации сеанса репликации запроса на запись в источник исходные данные с указанного адреса копируются в предопределенную область массива данных. Эта область известна как *место хранения*. Указатель на приемнике обновляется, чтобы содержать ссылку на эти данные в месте сохранения. И только после этого в источник делается новая запись. Этот процесс показан на рис. 11.12.

Когда после активации сеанса репликации будет выдан первый запрос на запись в приемник, данные копируются из источника в место хранения и значение указателя обновляется, позволяя ему ссылаться на эти данные в месте хранения. Еще одна копия исходных данных создается в месте хранения перед тем, как в это место будет сделана новая запись. Последующие записи в тот же самый блок данных в источнике или приемнике никаких операций копирования уже не вызывают. Этот процесс показан на рис. 11.13.

Когда выдается запрос на чтение из приемника, не изменившиеся с момента активации сеанса репликации блоки данных считываются из источника, в то время как блоки данных, претерпевшие изменения, считываются из места хранения.

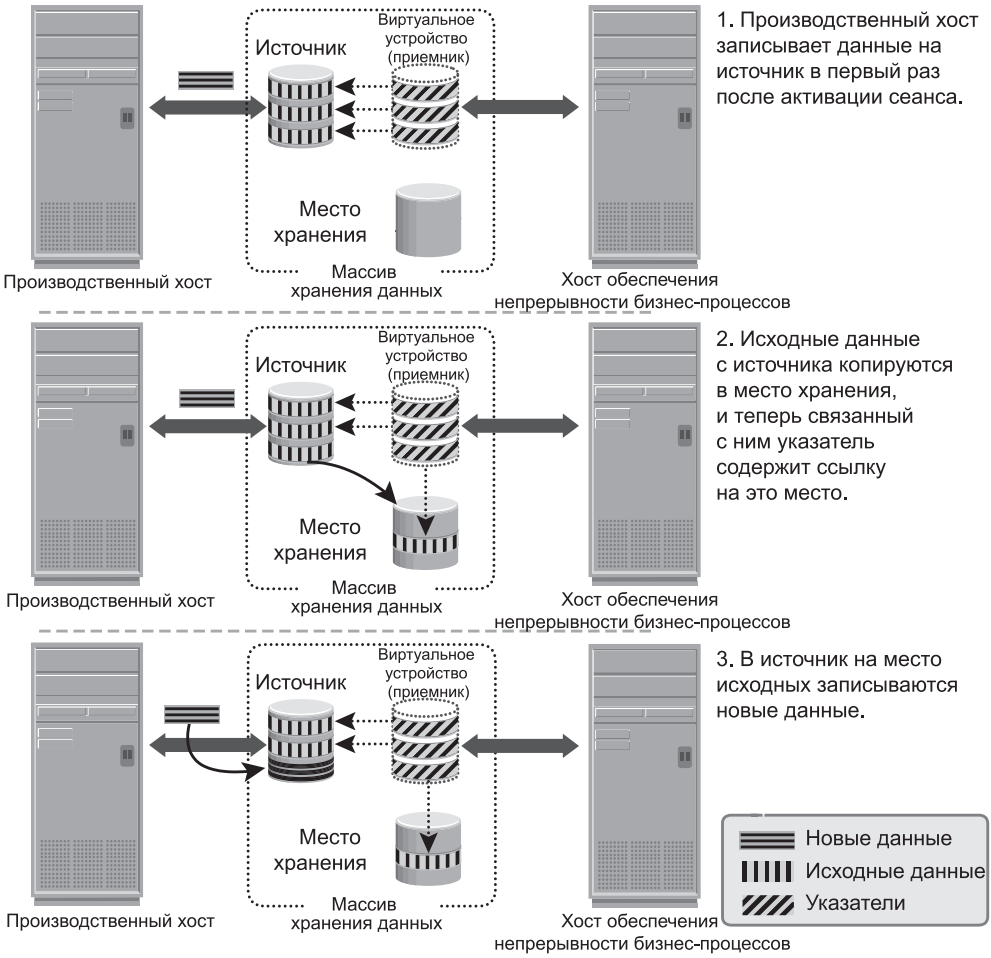


Рис. 11.12. Виртуальная репликация на основе использования указателей

В данных приемника объединяются представление не изменившихся данных в источнике и данные в месте хранения. Недоступность устройства-источника делает бесполезными те данные, которые имеются в приемнике. В приемнике содержатся только указатели на данные, поэтому физическая емкость, требующаяся для приемника, составляет лишь малую долю емкости источника. Емкость, требующаяся для места хранения, зависит от ожидаемой степени изменения данных.

11.4.3. Локальная репликация на основе использования сети

В репликации на основе использования сети сама репликация происходит на сетевом уровне между хостами и массивами хранения данных. Репликация на основе использования сети сочетает в себе преимущества репликации на

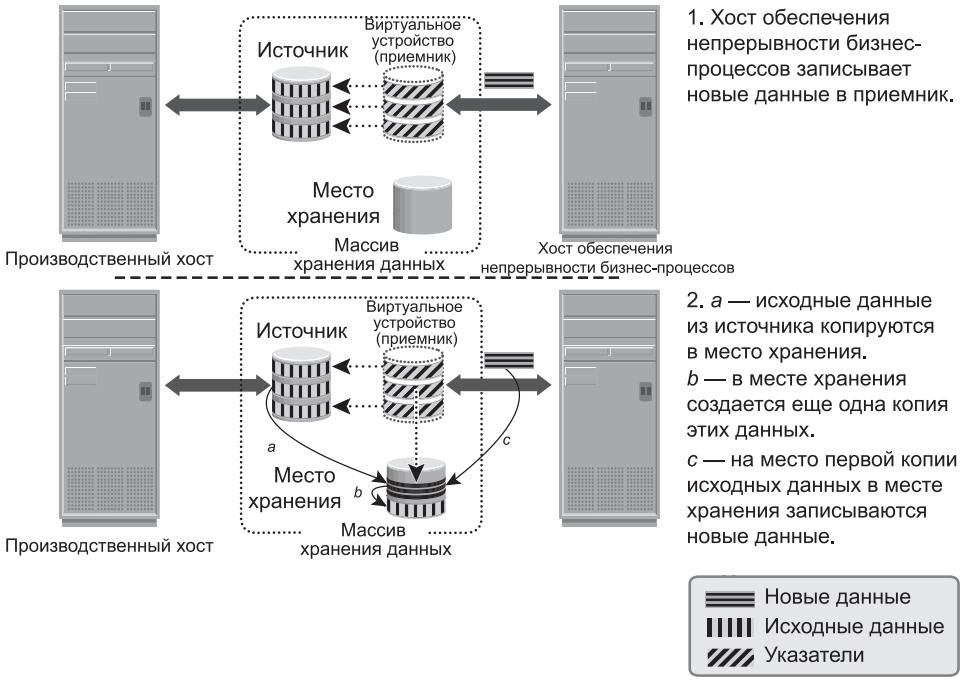


Рис. 11.13. Виртуальная репликация на основе использования указателей — запись в приемник

основе использования массива данных и репликации на основе использования хоста. Благодаря снятию нагрузки с серверов и массивов хранения данных репликация на основе использования сети может работать на большом количестве серверных платформ и массивов хранения данных, что делает ее идеальным решением для сред, характеризующихся высокой степенью разнородности оборудования. Для локальных и удаленных репликаций на основе использования сети используется технология непрерывной защиты данных — continuous data protection (CDP). Технология CDP для удаленных репликаций подробно рассматривается в главе 12.

Непрерывная защита данных

В среде дата-центра приложения, играющие особо важную роль в бизнес-процессах, зачастую требуют мгновенных и ничем не ограниченных точек восстановления данных. Традиционные технологии защиты данных предлагают точки восстановления с рядом ограничений. При потере данных можно выполнить откат системы только к последней доступной точке восстановления. Создание зеркальных копий предлагает постоянную репликацию, но если в производственных данных произойдет какое-либо логическое нарушение, ошибка может распространиться и на зеркальную копию, что сделает реплику непригодной к применению. В штатном режиме работы CDP позволяет

восстановить данные по состоянию на любой предыдущий PIT-момент. Такая возможность предоставляется благодаря отслеживанию всех изменений на производственных устройствах и поддержке согласованных PIT-образов.

В CDP все изменения данных постоянно перехватываются и сохраняются в отдельном от основного хранилища месте. Более того, показатель RPO имеет произвольную величину и не нуждается в предварительном определении. При использовании CDP восстановление после повреждения данных не создает никаких проблем, потому что позволяет вернуться к PIT-образу, предшествующему моменту повреждения данных. В CDP для хранения всех изменений в основном хранилище используется журнальный том. В этом томе содержатся все данные, претерпевшие изменения со времени начала сеанса репликации. Объем пространства, выделенного под журнал, определяет, насколько глубоко в прошлое могут уходить точки восстановления данных. CDP обычно осуществляется с использованием CDP-устройства и разветвителей (splitters). CDP может осуществляться также на основе использования хоста, в таком случае CDP-программа устанавливается на отдельной хост-машине.

CDP-устройство представляет собой интеллектуальную аппаратную платформу, на которой запускается CDP-программа и осуществляется управление локальной и удаленной репликациями. Разветвители записи перехватывают записи, производимые на производственный том с хоста, и разделяют каждую запись на две копии. Разветвление записи может выполняться на хосте, системе коммутации или массиве хранения данных.

Локальная репликация с использованием CDP-операций

Локальная репликация с использованием CDP изображена на рис. 11.14. При применении данного метода до начала репликации реплика синхронизируется с источником и только после этого стартует процесс репликации. После начала репликации все записи на источник разветвляются на две копии. Одна копия отправляется на CDP-устройство, а другая — на производственный том. Когда CDP-устройство получает копию записи, эта копия вместе с меткой времени записывается в журнальный том. На следующем шаге данные из журнального тома отправляются через заранее заданные интервалы реплике.

При восстановлении данных в источнике CDP-устройство проводит это восстановление из реплики и применяет записи журнала, сделанные до момента времени, выбранного для восстановления.

11.5. Отслеживание изменений в источнике и реплике

Обновление может произойти на устройстве-источнике после создания локальной PIT-реплики. Если главной целью локальной репликации является получение пригодной для восстановления данных или проведения операций возобновления бизнес-процессов PIT-копии, устройства репликации обновляться не должны. Изменения на устройстве репликации могут

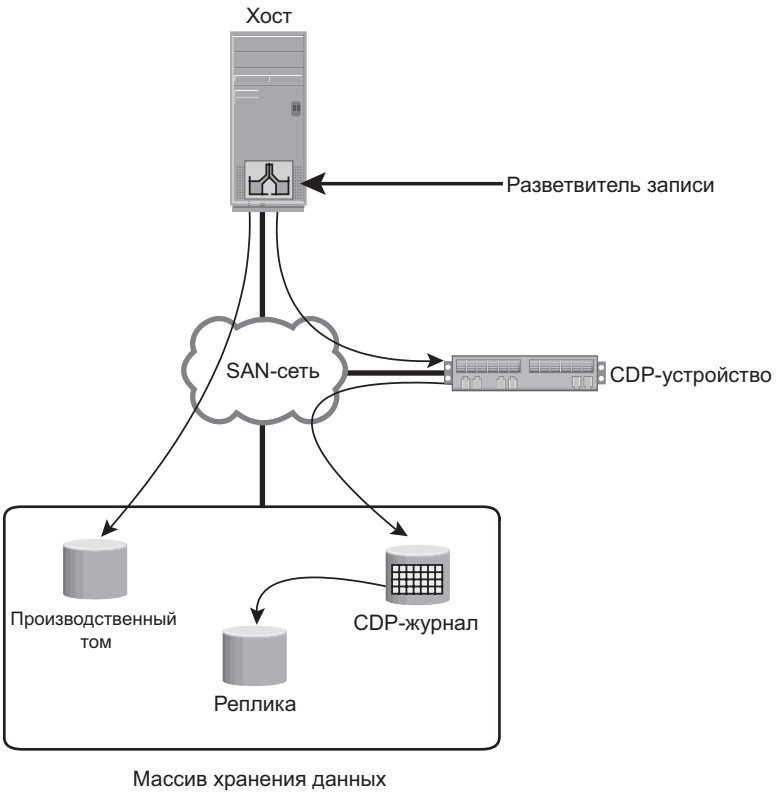


Рис. 11.14. Непрерывная защита данных — локальная репликация

произойти в том случае, если оно используется для проведения других бизнес-операций. Чтобы получить возможность проведения инкрементной ресинхронизации или операций возобновления работы, изменения как в источнике, так и в устройствах, репликации, происходящие после PIT-реплики, должны отслеживаться. Обычно это делается с помощью битовых массивов, где каждый бит представляет отдельно взятый блок данных. Размеры блоков данных могут варьироваться от 512 байт до более чем 64 Кбайт. Например, если размер блока составляет 32 Кбайт, то для устройств емкостью 1 Гбайт понадобится 32 768 бит (1 Гбайт, поделенный на 32 Кбайт). Тогда размер битового массива будет 4 Кбайт. Если данные в любом 32-килобайтном блоке изменились, соответствующий бит в битовом массиве будет помечен (установлен в единицу). Если в целях удобства отслеживания размер блока будет уменьшен, то, соответственно, возрастет и размер битового массива.

При создании реплики все биты в битовых массивах источника и приемника устанавливаются в нуль. Затем любые изменения в источнике или реплике помечаются путем установки соответствующих битов в битовом массиве в единицу. Когда потребуется ресинхронизация или восстановление данных, над битовыми массивами источника и приемника совершается

операция логического ИЛИ. Битовый массив, получившийся в результате этой операции, дает ссылки на все блоки, изменившиеся либо в источнике, либо в приемнике (рис. 11.15). Это позволяет оптимизировать ресинхронизацию или операцию восстановления, поскольку исключает необходимость копирования всех блоков между источником и репликой. Направление перемещения данных зависит от того, что именно проводится, ресинхронизация или операция восстановления.

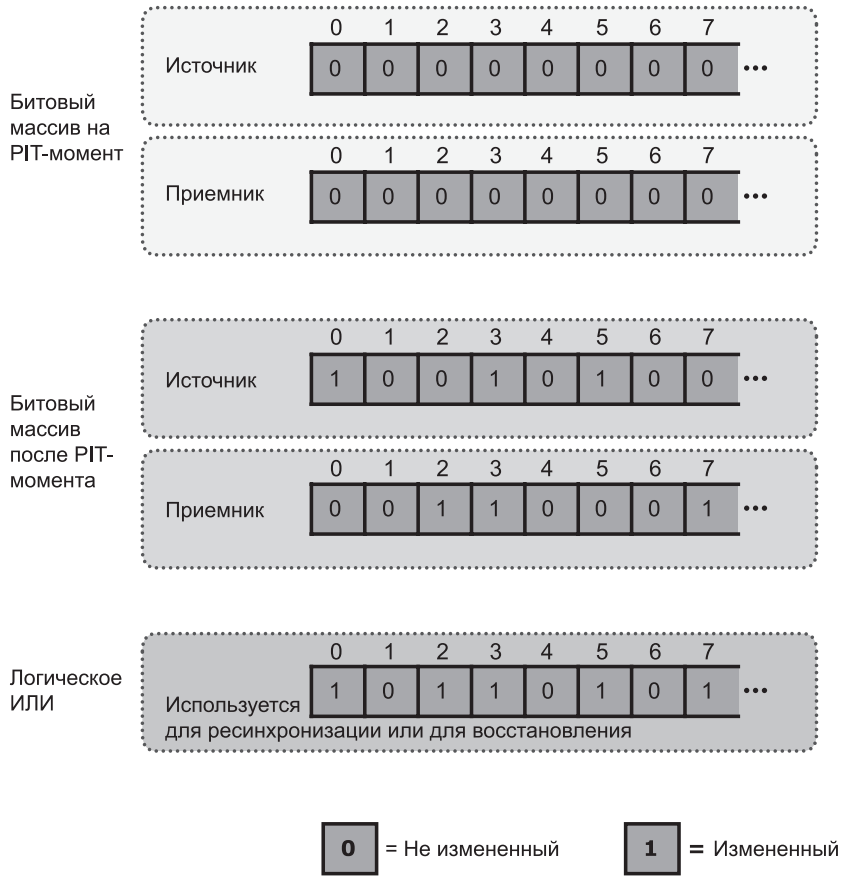


Рис. 11.15. Отслеживание изменений

Если нужно провести ресинхронизацию, изменения в реплике перезаписываются содержимым соответствующих блоков, взятых из источника. В показанном примере это будут блоки реплики с метками *2*, *3* и *7*.

Если же нужно выполнить восстановление данных, изменения в источнике перезаписываются содержимым соответствующих блоков из реплики.

В показанном примере это будут блоки источника с метками *0*, *3* и *5*. В любом случае изменения в источнике и приемнике не могут сохраняться одновременно.

11.6. Особенности восстановления и перезапуска

Локальные реплики используются для восстановления данных на производственных устройствах. Кроме того, согласованные PIT-реплики могут быть использованы для перезапуска бизнес-процессов.

Реплики используются для восстановления данных на производственных устройствах в случае логического повреждения данных на этих устройствах, то есть в тех случаях, когда устройства доступны, но в данных утрачена логическая последовательность. Примерами логического повреждения являются случайное уничтожение информации (таблиц или записей в базе данных), неправильный ввод данных и ненадлежащее обновление существующей информации. Операции восстановления с реплики являются инкрементными и имеют весьма небольшой показатель RTO. В некоторых случаях работа приложений на производственных устройствах может быть возобновлена до завершения копирования данных. Доступ к производственным устройствам и устройствам репликации должен быть прекращен до операции восстановления.

Производственные устройства могут также стать недоступными из-за физических сбоев, например, из-за сбоя производственного сервера или физического накопителя. В этом случае приложения могут быть перезапущены с использованием данных наиболее поздней реплики. В качестве защиты от дальнейших сбоев должна быть создана так называемая золотая копия (еще одна копия устройства репликации) для защиты копии данных от сбоя или повреждения устройств репликации.

Реплики всего тома (созданные как путем зеркалирования всего тома, так и путем использования указателей в режиме полного копирования) могут стать основой для восстановления данных на исходных устройствах-источниках или на новом наборе устройств-источников. Восстановление данных на исходных устройствах-источниках может быть инкрементным, но восстановление на новой группе устройств является операцией копирования всего тома.

При виртуальной репликации на основе использования указателей и репликации всего тома на основе использования указателей в режиме CoFA доступ к данным в реплике зависит от общего состояния и доступности томов источника. Если том источника по какой-либо причине недоступен, эти реплики становятся непригодными для операций восстановления или перезапуска.

Сравнительный анализ различных технологий репликации на основе использования массивов хранения данных приведен в табл. 11.1.

Таблица 11.1. Сравнение технологий локальной репликации

ФАКТОР	ЗЕРКАЛО ВСЕГО ТОМА	РЕПЛИКАЦИЯ ВСЕГО ТОМА НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ УКАЗАТЕЛЕЙ	ВИРТУАЛЬНАЯ РЕПЛИКАЦИЯ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ УКАЗАТЕЛЕЙ
Снижение производительности источника вследствие проведения репликации	Отсутствует	В режиме CoFA — незначительное, в режиме полной копии — отсутствует	Значительное
Размер приемника	Как минимум такой же, как и у источника	Как минимум такой же, как и у источника	Небольшая доля емкости источника
Доступность источника для восстановления данных	Не требуется	В режиме CoFA — требуется, в режиме полной копии — не требуется	Требуется
Доступность приемника	Только после синхронизации и отключения от источника	Доступен сразу же	Доступен сразу же

11.7. Создание нескольких реплик

Большинство технологий репликации на основе использования массива хранения данных позволяют устройствам-источникам поддерживать отношения, связанные с репликацией данных, с множеством приемников. Изменения, внесенные в источник и в каждый приемник, могут быть отслежены. Это позволяет производить инкрементную ресинхронизацию приемников. Каждая PIT-копия может быть использована для различных видов обеспечения непрерывности бизнес-процессов, а также в качестве точки восстановления.

На рис. 11.16 показан пример создания копии из одного и того же источника каждые шесть часов.

Если источник поврежден, данные могут быть восстановлены с самой последней PIT-копии. Максимальный показатель RPO в примере на рис. 11.16 — шесть часов. Более частое копирование уменьшает значение показателя RPO.

Технологии локальной репликации на основе использования массива хранения данных также позволяют создавать несколько *одномоментных* PIT-реплик. В этом случае все реплики будут содержать идентичные данные. Одна или несколько реплик могут быть отложены специально для проведения операций восстановления, а другие реплики могут быть использованы для проведения мероприятий по поддержке принятия решений.

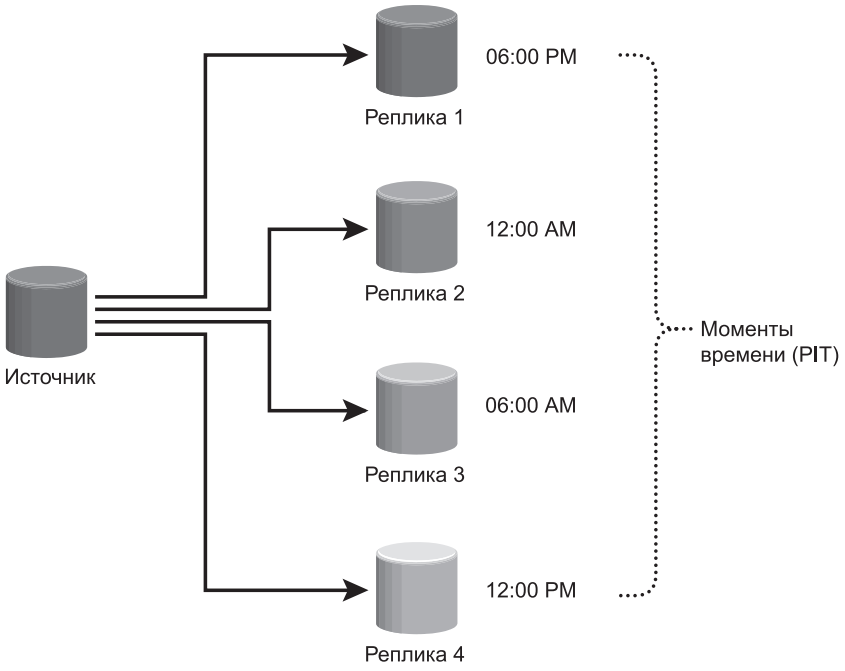


Рис. 11.16. Несколько реплик, созданных в различные моменты времени

11.8. Локальная репликация в виртуальной среде

До сих пор речь шла только о локальной репликации в среде физической инфраструктуры. В виртуальной среде наряду с репликацией томов хранилищ данных востребована также репликация виртуальной машины (VM). Обычно локальная репликация виртуальных машин выполняется на уровне вычислительной системы гипервизором. Но она может быть выполнена также на уровне хранилища данных с помощью локальной репликации на основе использования массива хранения данных, как это делается в физической среде. При применении метода на основе использования массива хранения данных LUN-устройство, на котором находятся виртуальные машины, реплицируется на другое LUN-устройство того же массива. Что же касается локальной репликации на основе использования гипервизора, то тут есть два варианта: моментальный снимок виртуальной машины — VM Snapshot и клонирование виртуальной машины — VM Clone.

VM Snapshot представляет собой снимок состояния и данных работающей виртуальной машины на конкретный момент времени. Состояние виртуальной машины включает в себя файлы виртуальной машины, такие как файл BIOS и файл конфигурации сети, и состояние ее функционирования (работает, не работает или находится в приостановленном состоянии). В данные виртуальной машины входят все файлы, из которых она составлена, включая

виртуальные диски и память. Для записи всех изменений на виртуальном диске с момента активации моментального снимка в технологии VM Snapshot используется отдельный дельта-файл. Моментальные снимки применяются, когда виртуальную машину нужно вернуть в предыдущее состояние в случае каких-либо логических повреждений. Возвращение виртуальной машины к предыдущему состоянию приводит к тому, что к PIT-моменту, в который был сделан снимок, возвращаются все настройки конфигурации гостевой операционной системы. С применением технологии VM Snapshot связан ряд проблем. В том случае, когда виртуальная машина обращается к данным, используя для этого обычные диски, репликация данных в этой технологии не поддерживается. Кроме того, использование для создания моментальных снимков гипервизора увеличивает нагрузку на вычислительную систему и негативно сказывается на ее производительности.

Другим рассматриваемым способом является клонирование виртуальной машины — VM Clone, при котором создается ее точная копия. Когда операция клонирования будет завершена, клон станет отдельной виртуальной машиной, не имеющей никакого отношения к породившей его виртуальной машине. Клон имеет свой собственный MAC-адрес, и изменения, вносимые в клон, не оказывают никакого воздействия на родительскую виртуальную машину. Точно так же изменения, вносимые в родительскую виртуальную машину, не появляются в клоне. Метод VM Clone хорошо подходит при необходимости развертывания множества идентичных виртуальных машин. Установка гостевой операционной системы и приложений на несколько виртуальных машин занимает много времени, а технология VM Clone помогает существенно упростить этот процесс.

11.9. Практическая реализация концепций: EMC TimeFinder, EMC SnapView и EMC RecoverPoint

Компания EMC предлагает широкий ассортимент решений по выполнению локальной репликации на основе использования массива хранения данных, который подходит для массивов разных производителей. При работе с массивами Symmetrix для репликации всего тома и локальной репликации на основе использования указателей предлагается воспользоваться семейством продуктов EMC TimeFinder. При работе с массивами EMC VNX предлагается воспользоваться решением EMC SnapView. А продукт EMC RecoverPoint представляет собой решение, позволяющее проводить репликации на основе использования сети. Самую свежую информацию об этих продуктах можно найти на сайте www.emc.com.

11.9.1. EMC TimeFinder

Семейство продуктов TimeFinder состоит из двух основных и четырех дополнительных решений. Основными решениями являются TimeFinder/Clone

и TimeFinder/Snap. К дополнительным относятся решения TimeFinder/Clone Emulation, TimeFinder/Consistency Groups, TimeFinder/Exchange Integration Module и TimeFinder/SQL Integration Module.

TimeFinder доступен как для открытых систем, так и для мэйнфреймов. Базовые решения поддерживают различные технологии локальной репликации на основе использования массива хранения данных, рассмотренные в данной главе. Дополнительные решения используются для приспособления реплик под специфические приложения и среды баз данных.

TimeFinder/Clone

TimeFinder/Clone создает PIT-копию исходного тома, которая может использоваться для создания резервных копий, содействия принятию решений или поддержки любых других процессов, требующих параллельного доступа к производственным данным. В TimeFinder/Clone используется технология репликации всего тома на основе использования указателей. TimeFinder/Clone позволяет создавать из одного производственного устройства до 16 активных клонов, и все клоны сразу же становятся доступными для проведения операций чтения и записи.

TimeFinder/Snap

TimeFinder/Snap создает экономящие пространство логические PIT-образы, которые называются моментальными снимками (snapshots). Эти снимки не являются полными копиями, а содержат указатели на данные источника. Устройство-приемник, используемое TimeFinder/Snap, называется виртуальным устройством — virtual device (VDEV). В нем содержатся указатели на устройство-источник или устройства хранения (SAVE-устройства), в которых находятся PIT-данные, подвергшиеся изменениям в источнике после запуска сеанса репликации. TimeFinder/Snap позволяет создавать на основе данных одного устройства-источника до 128 моментальных снимков.

11.9.2. EMC SnapView

SnapView представляет собой программу для массива EMC VNX, которая выполняет локальную репликацию на основе использования массива хранения данных и создает виртуальную копию, использующую указатели, и зеркальную копию всего тома источника, для чего применяются SnapView Snapshot и SnapView Clone соответственно.

SnapView Snapshot

Моментальный снимок SnapView не является полной копией производственного тома. Это логическое представление производственного тома, основанное на времени создания моментального снимка. Создание моментальных снимков занимает секунды, а неиспользуемые снимки можно просто удалить. Имеющаяся в Snapshot функция отката обеспечивает мгновенное

восстановление данных на томе источника. В SnapView Snapshot используются следующие основные понятия:

- **сеанс SnapView** — механизм SnapView Snapshot активируется при запуске сеанса и деактивируется при его остановке. В ходе сеанса моментальный снимок находится в отключенном режиме. В сеанс могут быть включены несколько моментальных снимков;
- **пул зарезервированных LUN-устройств** — это закрытая область, также называемая областью хранения, которая используется для хранения данных копии при выполнении первой записи (CoFW). Слово «зарезервированных» в этом названии отражает тот факт, что эти LUN-устройства зарезервированы и поэтому не могут выделяться хосту.

SnapView Clone

Клоны SnapView представляют собой копии всего тома, для которых требуется столько же места на диске, сколько занимает источник. Эти PIT-копии могут использоваться другими бизнес-операциями, например, резервного копирования и тестирования. SnapView Clone позволяет проводить инкрементную ресинхронизацию между источником и репликой. Процесс разобщения клона и его источника называется отрывом клона. После отрыва клон становится PIT-копией и может быть доступен другим бизнес-операциям.

11.9.3. EMC RecoverPoint

RecoverPoint представляет собой высокопроизводительное экономичное решение, обеспечивающее локальную и удаленную защиту данных как для физической, так и для виртуальной среды. Оно предоставляет возможность быстрого восстановления данных и создания неограниченного количества точек восстановления. RecoverPoint обеспечивает непрерывную защиту данных и выполняет репликацию между LUN-устройствами, расположенными в одном или нескольких массивах хранения данных в одном и том же месте. В RecoverPoint используется ресурсосберегающая технология, применяемая либо на сервере приложений, либо в системе коммутации, либо в массивах хранения данных с целью создания зеркальной копии записи в RecoverPoint-устройстве. Семейство продуктов RecoverPoint включает в себя версии RecoverPoint/CL, RecoverPoint/EX и RecoverPoint/SE.

RecoverPoint/CL является продуктом для создания реплик в разнородных средах серверов и хранилищ. Он поддерживает массивы хранения данных как компании EMC, так и других производителей. Этот продукт поддерживает разветвители записи, находящиеся как на хост-машине, так и в системе коммутации или массиве хранения данных. RecoverPoint/EX поддерживает создание реплик между массивами хранения данных компании EMC и допускает использование разветвителей записи только в массиве хранения данных. RecoverPoint/SE представляет собой версию RecoverPoint, предназначенную для массивов хранения данных серии VNX и рассчитанную на работу только

на хостах под управлением Windows с использованием разветвителей, находящихся в массиве хранения данных.

Резюме

Локальная репликация позволяет проводить быстрое восстановление данных, обеспечивая тем самым защиту от их повреждения в ходе интенсивного обновления информации на источнике данных. Эта технология стала неотъемлемой частью повседневных операций дата-центра.

В данной главе были рассмотрены процессы проведения локальных репликаций и примеры использования реплик. Локальные репликации могут выполняться с использованием различных технологий: на основе использования хоста, на основе использования массива хранения данных и на основе использования сети. Также в главе были рассмотрены особенности восстановления данных и перезапуска бизнес-процессов для локальной репликации на основе использования массива хранения данных и создания нескольких реплик. Кроме того, была рассмотрена локальная репликация виртуальных машин и виртуальных дисков.

Наряду с тем, что создание дубликатов данных с помощью локальной реплики обеспечивает высокий уровень их доступности, рассредоточение дубликатов по разным площадкам открывает возможность обеспечения непрерывной работы дата-центров в случае возникновения стихийных бедствий, которые могут вывести из строя весь производственный объект. Создание в результате репликации точных копий данных на удаленных площадках уже приобрело статус вполне состоявшейся технологии. Подробности удаленной репликации рассматриваются в следующей главе.

УПРАЖНЕНИЯ

1. Проведите исследование различных технологий, используемых для обеспечения согласованности локальной реплики.
2. Приведите примеры использования локальной реплики в различных бизнес-операциях.
3. Исследуйте факторы, определяющие требования к емкости хранилища, необходимой для создания места хранения при виртуальной репликации на основе использования указателей.
4. Проведите исследование технологии непрерывной защиты данных и ее преимуществ над технологиями репликаций на основе использования массивов хранения данных.
5. Администратор настроил создание шести виртуальных реплик исходного LUN-устройства на основе использования указателей и создание восьми реплик всего тома для того же самого LUN-устройства. Затем администратор создал четыре реплики на основе использования указателей для каждой созданной реплики всего тома. Сколько теперь доступно пригодных к использованию реплик?

Глава 12
Удаленная репликация

Удаленная репликация — это процесс создания реплик информационных активов на удаленных площадках. Удаленные реплики позволяют организациям снижать риски, связанные с региональными перебоями в снабжении электроэнергией, возникающими из-за катаклизмов, вызываемых как природными явлениями, так и действиями человека. На время перебоев вся работа может быть перемещена на удаленную площадку, что позволит обеспечить непрерывность бизнес-процессов. Так же, как и локальные, удаленные реплики можно использовать для проведения других бизнес-операций.

КЛЮЧЕВЫЕ ПОНЯТИЯ

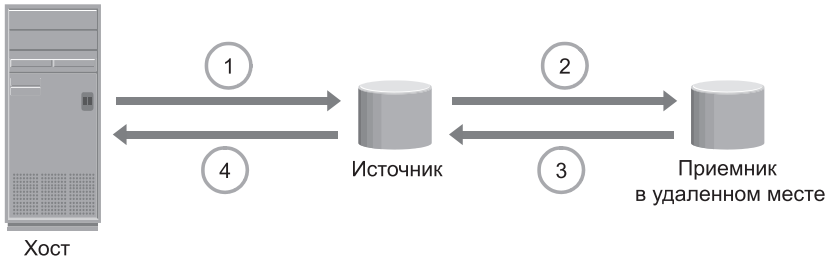
- Синхронная и асинхронная репликация
- Репликация на основе использования диспетчера логических томов (LVM)
- Доставка журналов на основе использования хоста
- Репликация в режиме буферизации диска
- Трехсторонняя репликация
- Миграция виртуальной машины

В данной главе рассматриваются различные технологии удаленной репликации, а также трехсторонняя репликация и приложения, осуществляющие миграцию данных. Кроме того, в данной главе рассматриваются удаленная репликация в виртуальной среде и миграция виртуальных машин.

12.1. Режимы удаленной репликации

Удаленная репликация может проводиться в двух основных режимах — синхронном и асинхронном. При *синхронной удаленной репликации* записи должны производиться на источник и в удаленную реплику (приемник), пока хост не получит подтверждение «запись завершена» (рис. 12.1). До завершения и подтверждения предыдущей записи следующая запись на

источник производиться не может. Тем самым гарантируется постоянная идентичность источника и реплики. Более того, записи передаются на удаленную площадку точно в том же порядке, в котором их получает источник. Таким образом поддерживается порядок следования записей. Если на месте источника случится сбой, синхронная удаленная репликация обеспечит нулевой или близкий к нулевому показатель целевой точки восстановления (RPO).



1. Хост записывает данные в источник.
2. Данные из источника реплицируются на приемник на удаленной площадке.
3. Приемник возвращает источнику подтверждение.
4. Источник отправляет хосту подтверждение о завершении операции записи.

Рис. 12.1. Синхронная репликация

Но при любой синхронной удаленной репликации увеличивается время отклика приложения, поскольку до отправки на хост подтверждения «запись завершена» эта запись должна быть произведена как на источник, так и на приемник. Степень влияния на время отклика зависит в основном от расстояния между площадками, ширины полосы пропускания и качества предоставляемых услуг — quality of service (QoS), обеспечиваемого сетевой инфраструктурой связи. Требования к ширине полосы пропускания сети при проведении синхронной репликации представлены на рис. 12.2. Если ширина полосы пропускания, предоставляемая для синхронной удаленной репликации, меньше максимальной нагрузки при записи, то в течение дня будут возникать ситуации, при которых время отклика окажется слишком долгим и лимит ожидания со стороны приложений будет превышен. Расстояния, на которые можно развертывать синхронную репликацию, зависят от устойчивости приложений к увеличению времени отклика. Обычно системы синхронной репликации развертываются, когда расстояние между двумя площадками не превышает 200 км.

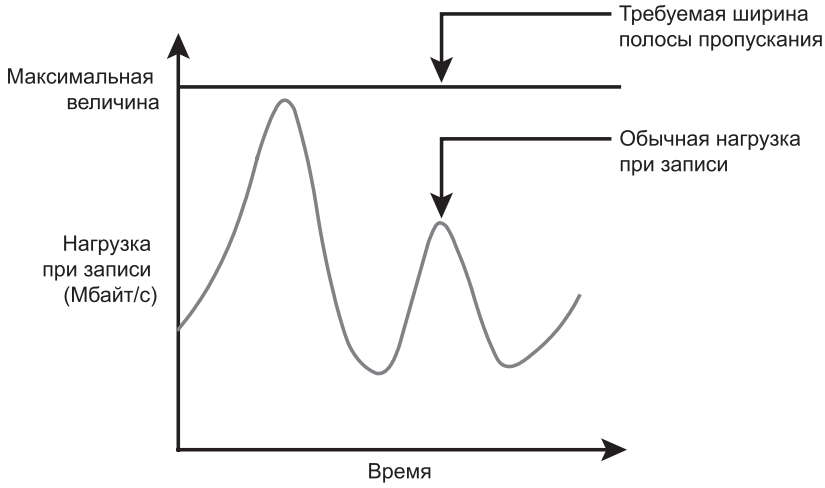
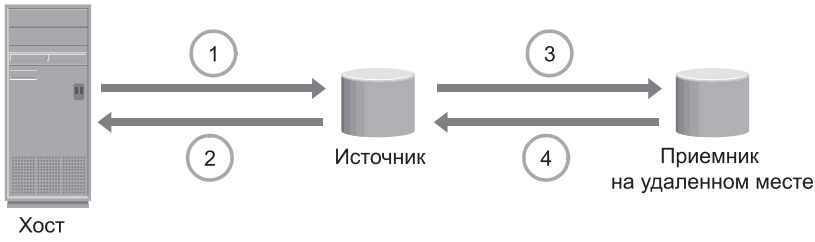


Рис. 12.2. Требования, предъявляемые к ширине полосы пропускания при синхронной репликации

При *асинхронной удаленной репликации* хост получает подтверждение сразу же после завершения записи данных на источник. В этом режиме данные заносятся в буфер источника, а затем уже передаются на удаленную площадку (рис. 12.3).



1. Хост записывает данные в источник.
2. Подтверждение о записи тут же отправляется на хост.
3. Спустя некоторое время данные передаются на приемник на удаленном месте.
4. Приемник отправляет источнику подтверждение о завершении операции записи.

Рис. 12.3. Асинхронная репликация

Асинхронная репликация исключает влияние на время отклика приложений, поскольку подтверждение о завершении записи поступает на хост источника незамедлительно. Это позволяет развертывать системы асинхронной репликации, когда расстояние между основной и удаленной площадками составляет от нескольких сотен до нескольких тысяч километров. Требуемая ширина полосы пропускания сети для проведения асинхронной репликации показана на рис. 12.4. В данном случае нужно, чтобы эта ширина была не ниже средней нагрузки при записи. Когда ширины полосы пропускания не хватает, данные могут заноситься в буфер и передаваться в удаленное место позже. Поэтому следует обеспечить достаточную емкость буфера.

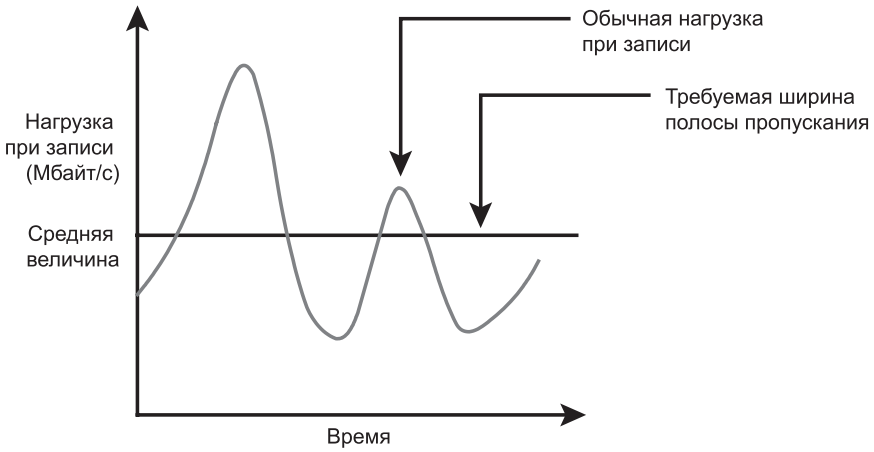


Рис. 12.4. Требования к ширине полосы пропускания для асинхронной репликации

При асинхронной репликации объем данных на удаленной площадке отстает от их объема на источнике по крайней мере на размер буфера. Поэтому при асинхронной удаленной репликации имеется вполне определенный (ненулевой) показатель RPO, учитываемый при реализации решений по восстановлению данных после возникновения аварийных ситуаций. Показатель RPO зависит от размера буфера, доступной ширины полосы пропускания сети и нагрузки на источник при записи.

При асинхронной репликации можно воспользоваться тем, что *в ссылке указывается место записи* (наличием повторяющихся записей на одно и то же место). Если до передачи данных на удаленную площадку выяснится, что на одно и то же место на устройстве хранения запись велась несколько раз, передана будет только последняя версия данных. Это свойство позволяет сэкономить ресурс полосы пропускания канала.

Как при синхронном, так и при асинхронном режиме реплицируются только записи на источник, а операции чтения по-прежнему обслуживаются с источника.

12.2. Технологии удаленной репликации

Удаленная репликация данных может управляться хостами или массивами хранения данных. В числе других вариантов можно назвать специализированные устройства на основе сети, предназначенные для репликации данных по локальной сети (LAN) или сети хранения данных (SAN). Усовершенствованный вариант репликации, так называемая трехсторонняя репликация, рассматривается в следующем разделе.

12.2.1. Удаленная репликация на основе использования хоста

При удаленной репликации на основе использования хоста для выполнения операций репликации и управления ими используются ресурсы хоста. При реализации этой разновидности репликации используются два основных подхода: репликация на основе использования диспетчера логических томов (LVM) и репликация базы данных, осуществляемая путем доставки журнала записей.

Удаленная репликация на основе использования LVM

Удаленная репликация на основе использования LVM выполняется и управляется на уровне группы томов. Записи на тома источника передаются на удаленный хост LVM-диспетчером. Такой же диспетчер на удаленном хосте получает записи и фиксирует их на удаленной группе томов.

Началу репликации предшествует создание как на месте источника, так и на месте приемника абсолютно идентичных групп томов, логических томов и файловых систем. Затем выполняется начальная синхронизация данных между источником и репликой. Одним из методов выполнения этой синхронизации является создание резервной копии данных источника и восстановление данных на удаленной реплике. Ее можно выполнить также путем репликации данных по IP-сети. До завершения начальной синхронизации производственная работа на томах источника, как правило, останавливается. После завершения начальной синхронизации она может быть запущена на томах источника и репликация данных может выполняться по существующей стандартной IP-сети (рис. 12.5).

При удаленной репликации на основе использования LVM поддерживается как синхронный, так и асинхронный режим работы. При сбое на стороне источника приложения могут быть перезапущены на удаленном хосте с использованием данных удаленных реплик.

Удаленная репликация на основе использования LVM не зависит от массивов хранения данных и поэтому поддерживает репликацию между разнородными массивами. Большинство операционных систем поставляются с диспетчерами логических томов (LVM), поэтому дополнительных лицензий и специализированного оборудования, как правило, не требуется.

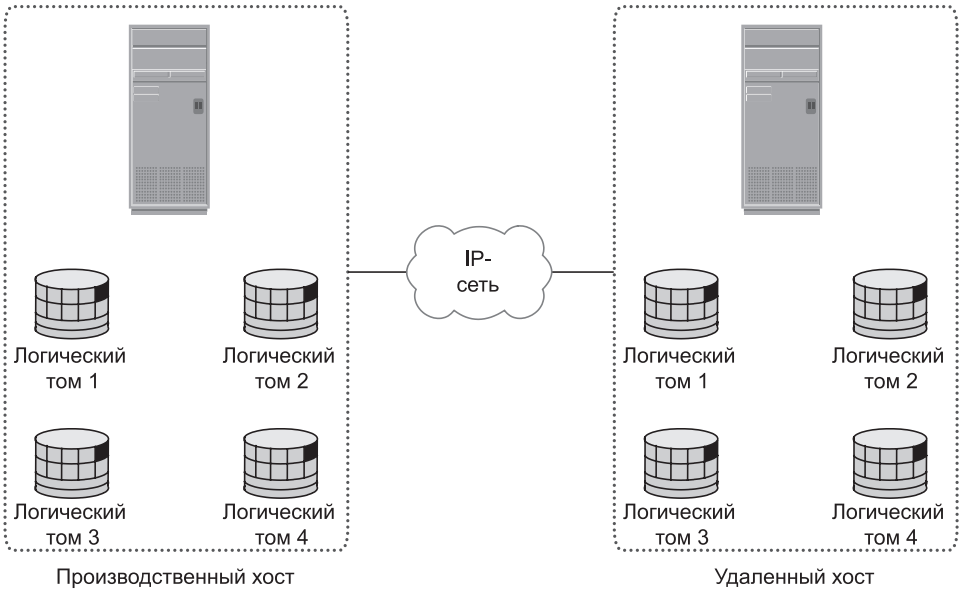


Рис. 12.5. Удаленная репликация на основе использования LVM

Процесс репликации требует дополнительных расходов ресурсов центральных процессоров хоста. Эти ресурсы на хосте источника делятся между задачами выполнения репликации и задачами приложений, что может стать причиной снижения производительности приложений, запущенных на хосте.

Поскольку в процесс репликации вовлекается также удаленный хост, он должен быть постоянно включен и доступен.

Доставка журналов на основе использования хоста

Репликация баз данных, осуществляемая посредством доставки журналов, представляет собой технологию репликации на основе использования хоста, поддерживаемую большинством баз данных. Транзакции на базе данных источника заносятся в журналы, которые периодически передаются хостом источника на удаленный хост (рис. 12.6). Удаленный хост получает журналы и применяет их записи к удаленной базе данных.

До начала производственной работы и репликации журнальных файлов все важные компоненты базы данных источника реплицируются на удаленную площадку. Все это происходит при закрытой базе данных источника.

После этого в базе данных источника начинается производственная работа. Удаленная база данных запускается в режиме ожидания, в котором она, как правило, недоступна для проведения транзакций.

Все системы управления базами данных через заранее определенные интервалы времени или при заполнении файла журнала проводят переключение с одного файла журнала на другой. Во время переключения с одного

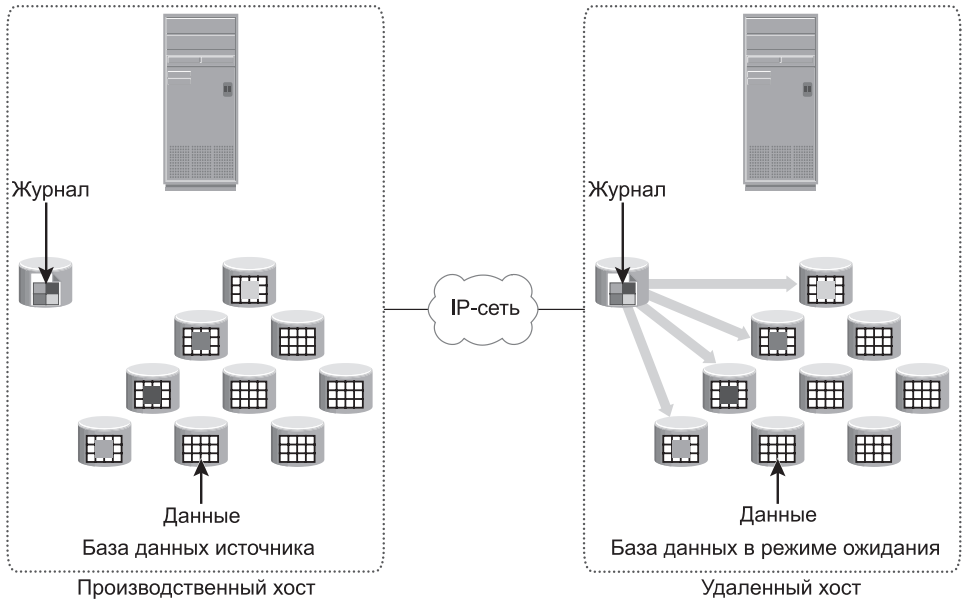


Рис. 12.6. Доставка журналов на основе использования хоста

журнала на другой текущий файл журнала закрывается и открывается новый файл журнала. После этого переключения закрывшийся файл журнала пересылается хостом источника удаленному хосту. Удаленный хост получает журнал и проводит обновление базы данных, находящейся в режиме ожидания.

Данный процесс обеспечивает согласованность базы данных, находящейся в режиме ожидания, с основной базой вплоть до последней зафиксированной в журнале транзакции. Показатель RPO на удаленной площадке имеет вполне определенное значение и зависит от размера журналов и частоты переключений с одного журнала на другой. При определении оптимального размера файла журнала следует учитывать доступную ширину полосы пропускания сети, время ожидания отклика, интенсивность обновлений базы данных источника, а также частоту переключений с одного журнала на другой.

Так же, как при удаленной репликации на основе использования LVM, для репликации файлов журналов может использоваться существующая стандартная IP-сеть. Доставка журналов на основе использования хоста не требует широкой полосы пропускания, поскольку при этой доставке через определенные интервалы времени передаются лишь файлы журналов.

12.2.2. Удаленная репликация на основе использования массивов хранения данных

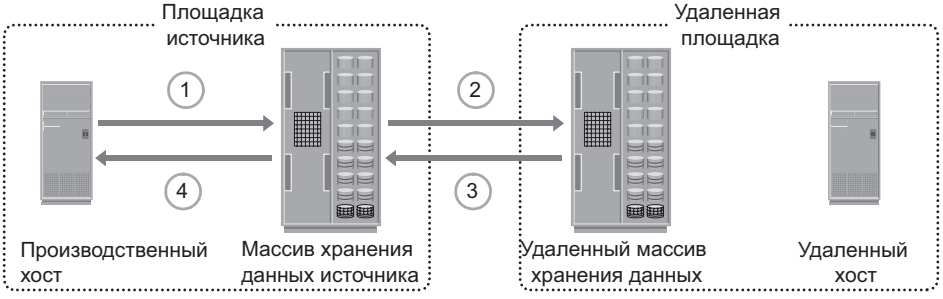
При удаленной репликации на основе использования массивов хранения данных репликация выполняется и управляется за счет использования

операционной среды и ресурсов массива. Тем самым снижается нагрузка на центральные процессоры хоста, ресурсы которых предпочтительнее использовать для работы приложений, запущенных на хосте. Устройства источника и его реплики находятся в разных массивах хранения данных. Данные из массива хранения источника в массив хранения приемника могут передаваться по совместно используемой или выделенной сети.

Репликации между массивами могут выполняться в синхронном, асинхронном режиме или в режиме буферизации диска.

Режим синхронной репликации

При синхронной удаленной репликации на основе использования массива запись должна быть направлена на источник и на приемник и не считается завершенной вплоть до отправки на производственный хост подтверждения «запись завершена». Следующие записи на этот источник не могут производиться до завершения и подтверждения каждой предшествующей им записи. Процесс синхронной репликации на основе использования массива данных показан на рис. 12.7.



1. Запись с производственного хоста принимается массивом хранения данных источника.
2. Затем запись передается на удаленный массив хранения данных.
3. Удаленный массив хранения данных отправляет подтверждение массиву хранения данных источника.
4. Массив хранения данных источника сообщает производственному хосту о завершении операции записи.

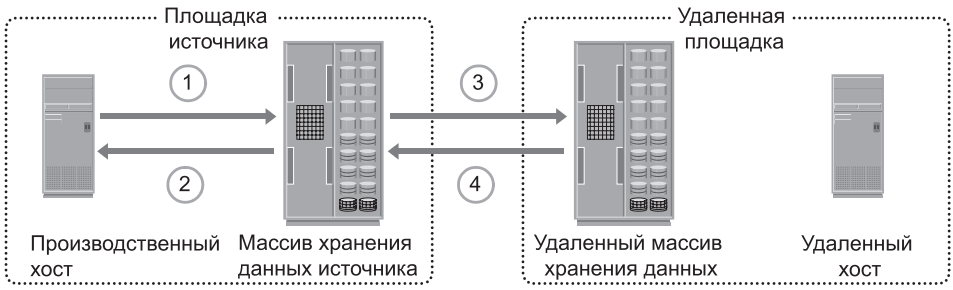
Рис. 12.7. Синхронная удаленная репликация на основе использования массива хранения данных

Чтобы при синхронной репликации можно было оптимизировать процесс репликации и свести к минимуму влияние на время отклика приложения, данные записи помещаются в кэш-память обоих массивов хранения данных. «Интеллектуальные» массивы хранения откладывают стадию записи этих данных на соответствующие диски на более поздний срок.

Если сетевой канал дает сбой, репликация приостанавливается, но производственная работа с использованием массива хранения данных источника может вестись непрерывно. Операционная среда массива может отслеживать записи, не переданные в удаленный массив хранения данных. Когда сетевой канал будет восстановлен, накопленные данные будут переданы в удаленный массив хранения данных. Если за время неисправности сетевого канала произойдет сбой на стороне источника, часть данных будет утрачена, а показатель RPO приемника будет иметь ненулевое значение.

Режим асинхронной репликации

При режиме *асинхронной удаленной репликации* на основе использования массива хранения данных (рис. 12.8) запись производится на источник, о чем тут же проходит оповещение хоста. Данные записи попадают в буфер источника, после чего передаются на удаленную площадку. Устройства источника и приемника не могут все время содержать одинаковые данные, поскольку обновление данных на приемнике постоянно запаздывает, соответственно, показатель RPO приемника не может быть нулевым.



1. Производственный хост ведет запись в массив хранения данных источника.
2. Массив хранения данных тут же уведомляет производственный хост о завершении записи.
3. Затем данные записи отправляются в массив хранения данных приемника.
4. Приемник после получения данных записи имеющимся у него массивом хранения данных отправляет подтверждение массиву хранения данных источника.

Рис. 12.8. Асинхронная удаленная репликация на основе использования массива хранения данных

Так же, как и при синхронной репликации, записи асинхронной репликации сначала размещаются в кэш-памяти обоих массивов, а затем уже сохраняются на соответствующих дисках.

В некоторых реализациях асинхронной удаленной репликации поддерживается порядок следования записей. К каждой записи при ее получении источником прикрепляются метка времени и порядковый номер. Затем

записи передаются на удаленный массив, где попадают в удаленную реплику точно в таком же порядке, в каком они поступали в буфер источника. Тем самым косвенно гарантируется согласованность данных на удаленных репликах. В других реализациях согласованность данных гарантируется использованием принципа зависимых записей, присущего большинству систем управления базами данных. При асинхронной удаленной репликации записи сохраняются в буфере в течение заранее определенного срока. По истечении данного срока буфер закрывается и для последующих записей открывается новый буфер. Все записи из закрытого буфера передаются и записываются в удаленную реплику единым пакетом.

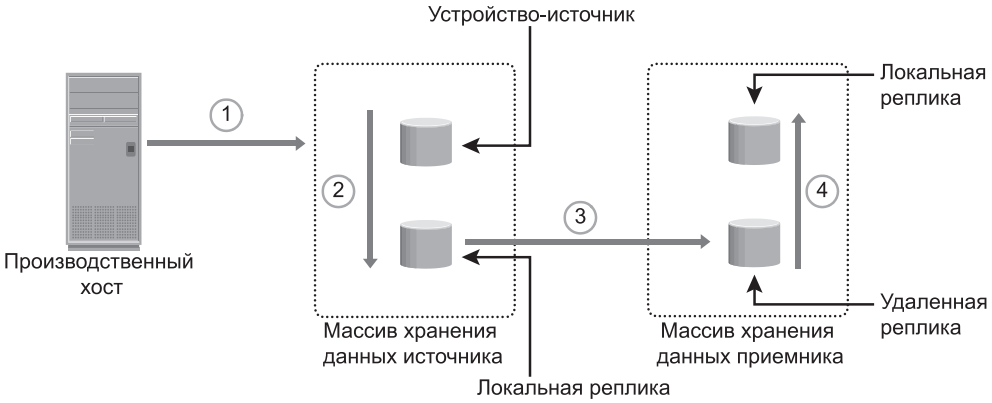
Асинхронная удаленная репликация позволяет сэкономить средства на ширине полосы пропускания сети, поскольку требует меньшей ширины, чем при пиковых нагрузках, возникающих при проведении записей. В те моменты времени, когда рабочая нагрузка при выполнении записей превышает среднее значение ширины полосы пропускания, на массиве хранения данных должен быть выделен буфер соответствующей емкости, позволяющий сохранять данные этих записей.

Репликация в режиме буферизации диска

Репликация в режиме буферизации диска представляет собой сочетание технологий локальной и удаленной репликации. Сначала создается согласованная локальная PIT-реплика устройства-источника. Затем она реплицируется на удаленную реплику массива хранения данных приемника.

Последовательность операций при удаленной репликации в режиме буферизации диска показана на рис. 12.9. В начале каждого цикла работа сетевых каналов между двумя массивами приостанавливается и передача данных не производится. При работе производственного приложения на устройстве источника создается согласованная локальная PIT-реплика устройства-источника. Сетевые каналы возобновляют работу, и данные локальной реплики массива хранения данных источника передаются их удаленной реплике в массиве хранения данных приемника. После синхронизации этой пары работа сетевых каналов приостанавливается и создается следующая локальная реплика источника. Дополнительно в массиве хранения данных приемника может быть создана локальная PIT-реплика удаленного устройства. Частота этого цикла операций зависит от доступной ширины канала и интенсивности изменения данных на устройстве-источнике. Поскольку при удаленной репликации в режиме буферизации диска используется локальная репликация, изменения, вносимые в источник и его реплику, можно отследить. Следовательно все операции ресинхронизации между источником и приемником могут проводиться в инкрементном режиме. По сравнению с синхронной и асинхронной репликацией для удаленной репликации, проводимой в режиме буферизации диска, требуется меньшая ширина полосы пропускания.

При удаленной репликации, проводимой в режиме буферизации диска, показатель RPO в удаленном месте составляет порядка нескольких часов.



1. Производственный хост записывает данные на устройство-источник.
2. Создается согласованная локальная PIT-реплика устройства-источника.
3. Данные из локальной реплики в массиве хранения данных источника передаются на их удаленную реплику в массиве хранения данных приемника.
4. Дополнительно в массиве хранения данных приемника создается локальная PIT-реплика удаленного устройства.

Рис. 12.9. Удаленная репликация в режиме буферизации диска

Например, локальная реплика устройства-источника создана в 10:00 и ее данные передаются на удаленную реплику, для чего требуется один час. Изменения, внесенные в источник после 10:00, отслеживаются. Еще одна реплика устройства-источника создается в 11:00 путем применения отслеженных изменений между источником и локальной репликой (копией на 10:00). В течение следующего цикла передачи (данных, имевшихся на 11:00) данные источника полностью попадают в приемник к 12:00. Пока в удаленную реплику не будут успешно переданы данные на 11:00, на локальной реплике удаленного массива находятся данные на 10:00. Если до завершения передачи произойдет сбой на месте источника, то в самом худшем случае показатель RPO будет равен двум часам, потому что в удаленном месте данные будут на 10:00.

12.2.3. Удаленная репликация на основе использования сети

При удаленной репликации на основе использования сети сама репликация происходит на сетевом уровне между хостом и массивом хранения данных. Решения для проведения удаленной репликации на основе использования сети предлагаются также технологией непрерывной защиты данных — continuous data protection (CDP), рассматриваемой в предыдущей главе.

Удаленная репликация на основе использования CDP

В условиях обычной эксплуатации удаленная репликация на основе использования CDP предоставляет возможность восстановления данных на любой

момент времени (any-point-in-time), что позволяет LUN-устройствам приемника производить откат к любому предыдущему моменту времени. Так же, как и при локальной репликации на основе использования CDP, при удаленной репликации на этой же основе обычно используются журнальный том, CDP-устройство или CDP-программа, установленная на отдельном хосте (при CDP-защите, основанной на использовании хоста), и разветвитель записи для выполнения репликации между площадками. CDP-устройства содержатся как на стороне источника, так и на стороне приемника.

Процесс проведения удаленной репликации на основе использования CDP показан на рис. 12.10. При использовании данного метода реплика синхронизируется с источником, после чего начинается процесс репликации. Как только он начнется, все записи с хоста на источник разветвляются на две копии. Одна копия отправляется локальному CDP-устройству на площадке источника, а вторая — на производственный том. После получения записи устройство на стороне источника отправляет ее устройству на удаленной площадке. Для проведения асинхронных операций записи на CDP-устройстве источника накапливаются, а лишние блоки удаляются. Затем записи выстраиваются по порядку следования и сохраняются с соответствующими метками времени. После чего данные сжимаются и вычисляется их контрольная сумма. Затем они включаются в план доставки по IP- или FC-сети в удаленное CDP-устройство. После получения данных удаленное устройство проверяет их целостность, используя для этого контрольную сумму. Затем данные распаковываются и записываются в удаленный журнальный том. На следующем этапе данные из журнального тома отправляются через заранее определенные интервалы времени реплике.

В асинхронном режиме локальное CDP-устройство отправляет подтверждение о записи сразу же после получения данных этой записи. В синхронном

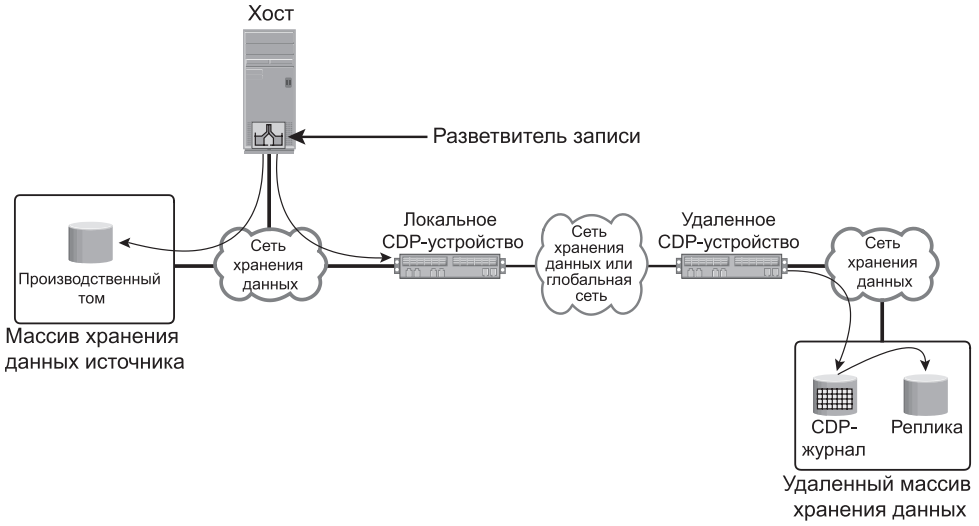


Рис. 12.10. Удаленная репликация на основе использования CDP

режиме репликации хост с приложениями, перед тем как инициировать следующую запись, ждет подтверждения от CDP-устройства, находящегося на удаленной площадке. Режим синхронной репликации при высокой загруженности записями оказывает отрицательное влияние на производительность приложений.

Для удаленной репликации на большие расстояния развертывается оборудование, работающее по технологиям оптоволоконной сети, например оборудование со спектральным уплотнением высокой плотности — dense wavelength division multiplexing (DWDM), оборудование с разреженным спектральным уплотнением — coarse wavelength division multiplexing (CWDM) и оборудование синхронной оптической сети — synchronous optical network (SONET). Дополнительные сведения об этих технологиях приводятся в приложении Д.

12.3. Трехсторонняя репликация

При синхронной репликации площадку, где находится источник, и площадку, где находится приемник, разделяет сравнительно небольшое расстояние. Поэтому, если в районе их размещения произойдет какой-нибудь катаклизм, обе эти площадки могут стать недоступными. Это может привести к увеличению показателей RPO и RTO, поскольку самая последняя пригодная копия данных должна поступить из другого источника, например из находящейся на другом площадке ленточной библиотеки.

Но при асинхронной репликации местный катаклизм, вероятнее всего, не затронет площадку, где находится приемник, потому что она обычно находится на расстоянии нескольких сотен, а то и тысяч километров. Если на площадке источника возникнет аварийная ситуация, производство может быть перемещено на площадку приемника, но до тех пор, пока авария не устранена, удаленной защиты данных не будет.

Снизить риски, связанные с двусторонней репликацией, призвана *трехсторонняя репликация*, при которой данные с места источника реплицируются сразу на две удаленные площадки. В отношении одной из двух площадок репликация может быть синхронной, при этом создается решение с практически нулевым показателем RPO, а в отношении другой она может быть асинхронной или использующей буферизацию диска, тем самым обеспечивается вполне определенный отличный от нуля показатель RPO. Трехсторонняя удаленная репликация может быть реализована в виде каскадного (с несколькими транзитными участками) или треугольного (с несколькими приемниками) решения.

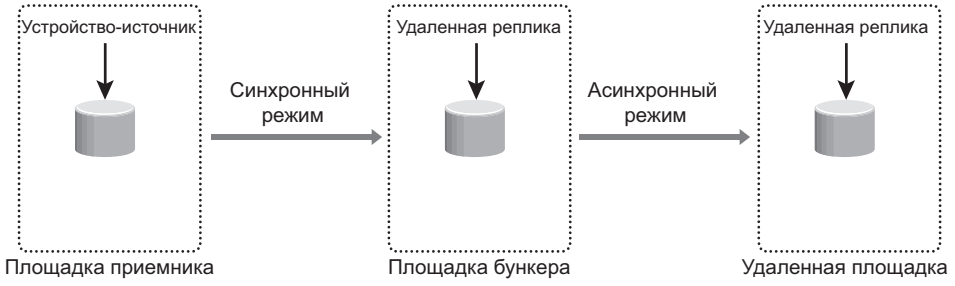
12.3.1. Трехсторонняя репликация — каскадное решение (с несколькими транзитными участками)

При *каскадной (с несколькими транзитными участками)* трехсторонней репликации на первом транзитном участке поток данных направляется от источника к промежуточному массиву хранения данных — так называемому

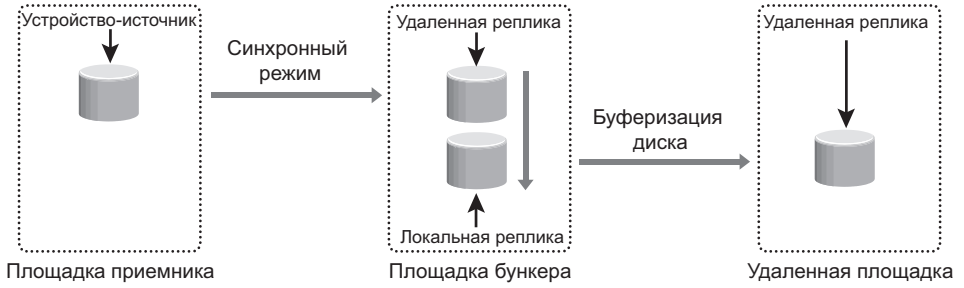
бункеру, а затем на втором транзитном участке он направляется из бункера в массив хранения данных на удаленной площадке. Репликация между источником и удаленной площадкой может быть проведена двумя способами: синхронным + асинхронным или синхронным + с буферизацией диска. Репликация между источником и бункером осуществляется в синхронном режиме, а репликация между бункером и удаленной площадкой может быть проведена либо в режиме буферизации диска, либо в асинхронном режиме.

Метод, использующий синхронный режим + асинхронный режим

В данном методе используется сочетание синхронной и асинхронной технологий удаленной репликации. Синхронная репликация осуществляется между источником и бункером, а асинхронная — между бункером и удаленной площадкой. Удаленная реплика в бункере служит для асинхронной репликации источником для создания удаленной реплики на удаленной площадке. Метод, использующий синхронный режим + асинхронный режим, показан на рис. 12.11, *а*.



а



б

Рис. 12.11. Трехсторонняя удаленная репликация с каскадным решением (с несколькими транзитными участками): *а* — метод, использующий синхронный режим + асинхронный режим; *б* — метод, использующий синхронный режим + режим буферизации диска

Для данной реализации показатель RPO на удаленной площадке обычно составляет порядка нескольких минут. Для этого метода требуется как минимум три устройства хранения данных, включая источник. Другими двумя устройствами являются устройства, содержащие синхронную реплику в бункере и асинхронную реплику на удаленной площадке.

При возникновении аварийной ситуации на источнике производственные операции с нулевой или почти нулевой потерей данных переносятся в то место, где находится бункер. Но в отличие от ситуации с синхронной двусторонней репликацией, при этом остается еще одна удаленная защита на третьей площадке. Показатель RPO между бункером и третьей площадкой может составлять порядка нескольких минут.

Если аварийная ситуация возникнет в том месте, где находится бункер, или будет нарушен сетевой канал между площадкой источника и площадкой бункера, работа на исходной площадке будет продолжена в обычном режиме, но уже без удаленной репликации. Сложится точно такая же ситуация, как и при аварии на удаленной площадке при двусторонней репликации. Из-за аварии в месте бункера провести обновление на удаленной площадке будет невозможно. Поэтому обновление данных на удаленной площадке будет по-прежнему отставать от источника, но преимущество такого решения заключается в том, что теперь при сбое на площадке источника работа может быть возобновлена на удаленной площадке. Показатель RPO на удаленной площадке зависит от разницы во времени между аварией в месте бункера и аварией на площадке источника.

Региональный катаклизм при трехсторонней удаленной репликации с каскадным решением (с несколькими транзитными участками) похож по своему воздействию на аварию на площадке источника при двусторонней асинхронной репликации. Работа перемещается на удаленную площадку с показателем RPO порядка нескольких минут. Удаленная защита данных при этом отсутствует вплоть до устранения последствий регионального катаклизма. Пока эти последствия не будут устранены, на удаленной площадке могут быть использованы технологии локальной репликации.

Если катаклизм произойдет на удаленной площадке или будет нарушен сетевой канал между бункером и удаленной площадкой, работа на площадке источника продолжится в обычном режиме, при этом защита от возникновения чрезвычайных ситуаций будет предоставлена в том месте, где находится бункер.

Метод, использующий синхронный режим + режим буферизации диска

В данном методе используется сочетание технологий локальной и удаленной репликации. Синхронная репликация проводится между источником и бункером, при этом в бункере создается согласованная локальная PIT-реплика. Данные из локальной реплики в бункере передаются удаленной реплике на удаленной площадке. Кроме того, после получения данных из бункера локальная реплика может быть создана и на удаленной площадке. Метод, использующий синхронный режим + режим буферизации диска, показан на рис. 12.11, *б*.

При реализации данного метода для репликации одного устройства хранения данных потребуется минимум четыре устройства хранения, включая источник. Остальные три устройства являются синхронной удаленной репликой в бункере, согласованной локальной PIT-репликой в бункере и репликой на удаленной площадке. Для данной реализации показатель RPO на удаленной площадке обычно составляет порядка нескольких часов.

Процесс создания согласованной PIT-копии в бункере и инкрементное обновление удаленной реплики происходят постоянно в циклическом режиме.

12.3.2. Трехсторонняя репликация — треугольное решение (с несколькими приемниками)

При *трехсторонней репликации с треугольным решением (с несколькими приемниками)* данные в массиве хранения данных источника используются параллельно для репликации в два разных массива хранения данных на двух разных площадках (рис. 12.12). Репликация с источника на площадку бункера

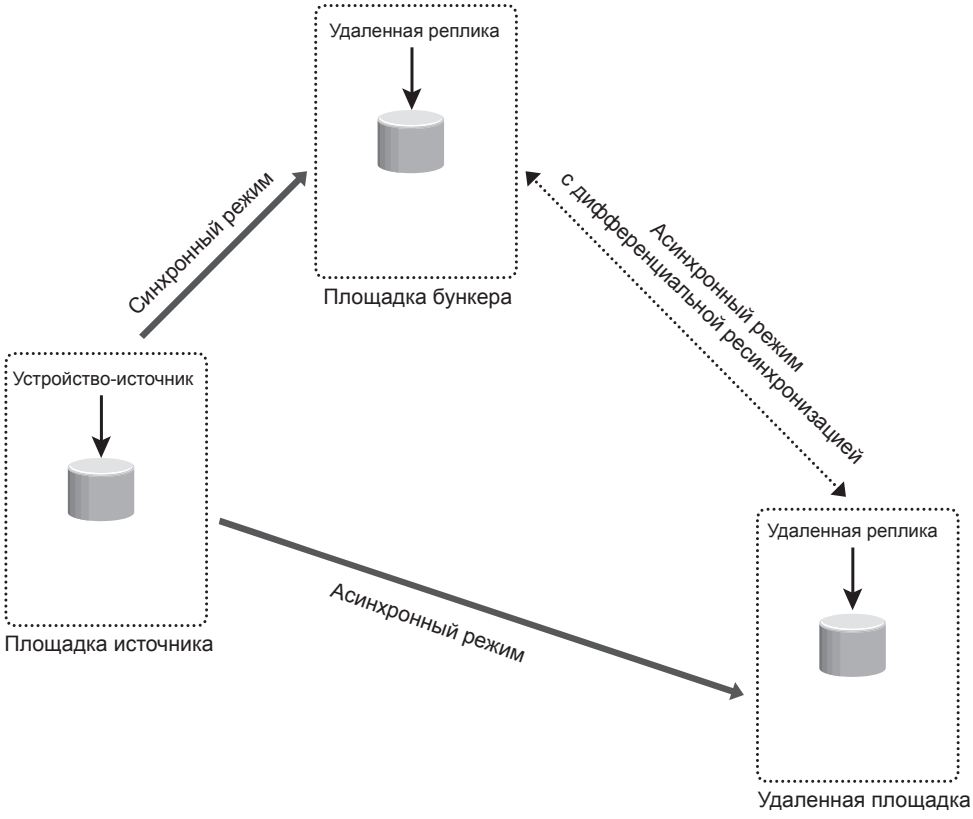


Рис. 12.12. Трехсторонняя удаленная репликация с треугольным решением (с несколькими приемниками)

(приемник *1*) проводится в синхронном режиме с показателем RPO, близким к нулю. Репликация с источника на удаленную площадку (приемник *2*) проводится в асинхронном режиме с показателем RPO, составляющим порядка нескольких минут. Расстояние между источником и удаленной площадкой может составлять несколько тысяч километров. При этой реализации обновление данных на удаленной площадке не зависит от того места, где находится бункер, поскольку данные копируются в асинхронном режиме непосредственно с источника. Треугольная конфигурация (с несколькими приемниками) в отличие от каскадных решений (с несколькими транзитными участками), для которых авария на площадке бункера приводит к отставанию в обновлении данных и увеличению показателя RPO, обеспечивает вполне удовлетворительные значения RPO.

Основное преимущество трехсторонней репликации с треугольным решением (с несколькими приемниками) заключается в возможности в случае аварии на площадке источника переместить работу в любое из двух удаленных мест с сохранением при этом асинхронной защиты от всевозможных катаклизмов как в месте бункера, так и в удаленном месте. Ресинхронизация между двумя не пострадавшими местами-приемниками носит инкрементный характер. При аварии в любом из этих трех мест защита от катаклизмов все равно остается доступной.

В ходе обычной работы все три площадки доступны, и с производственной нагрузкой справляется оборудование в месте источника. В любой отдельно взятый момент времени данные в бункере и на источнике являются идентичными. Обновление данных на удаленной площадке отстает от их обновления на источнике и в бункере. Сетевые каналы репликации между площадкой бункера и удаленной площадкой хотя и будут готовыми к работе, но задействованы не будут. Таким образом, в ходе обычной работы никакого перемещения данных между массивами хранения бункера и удаленной площадки производиться не будет. Разница в данных между площадкой бункера и удаленной площадкой отслеживается, поэтому в случае аварии на площадке источника работа может быть возобновлена на площадке бункера или на удаленной площадке с инкрементной ресинхронизацией между этими двумя местами.

Региональный катаклизм при трехсторонней удаленной репликации с треугольным решением (с несколькими приемниками) по своим последствиям похож на аварию на площадке источника при двусторонней асинхронной репликации. При возникновении аварийной ситуации работа переносится на удаленную площадку, при этом показатель RPO составляет порядка нескольких минут. Удаленная защита утрачивается до момента устранения последствий регионального катаклизма. Пока эти последствия не будут устранены, на удаленной площадке могут быть использованы технологии локальной репликации.

Аварии в бункере или на удаленной площадке не относятся к разряду серьезных, поскольку работа может непрерывно вестись в месте источника без потери удаленной защиты от всевозможных катаклизмов. Нарушение сетевого канала либо между источником и бункером, либо между источником

и удаленной площадкой не повлияет на работу на площадке источника, и при этом будет сохранена удаленная защита от всевозможных катаклизмов с того места, которое осталось доступным.

12.4. Решения по осуществлению миграции данных

Решение, позволяющее осуществить миграцию или обеспечить мобильность данных, относится к специализированной репликационной технологии, позволяющей создавать удаленные копии на конкретный момент времени (PIT-копии). Эти копии могут использоваться для обеспечения мобильности данных, их миграции, распространения определенного содержимого и восстановления после аварий. Рассматриваемое решение позволяет перемещать данные между разнородными массивами хранения. Данные перемещаются из одного массива в другой по SAN- или WAN-сети. Эта технология не зависит от применяемых приложений, конструкций серверных платформ и типов операционных систем, поскольку все связанные с репликацией операции выполняются одним из массивов хранения данных.

Под *мобильностью данных* понимается возможность их переноса между разнородными массивами хранения данных из соображений сокращения затрат на приобретение оборудования, повышения производительности или по каким-либо другим причинам. Это помогает создавать многоуровневую стратегию хранения данных. *Миграцией данных* называется перемещение данных из одного массива хранения данных в другие разнородные массивы для проведения технического переоснащения, объединения данных или решения других задач. Массив, выполняющий операции репликации, называется *управляющим массивом*. Данные могут быть перемещены между устройствами в обе стороны, устройствами в самом управляющем массиве и устройствами в удаленном массиве хранения данных. Устройства в управляющем массиве, задействованные в сеансе репликации, называются *управляющими устройствами*. У каждого управляющего устройства имеется свой партнер с противоположной стороны — *удаленное устройство* в *удаленном массиве хранения данных*. Понятия «управляющий» или «удаленный» не служат признаком направления потока данных, они лишь указывают на тот массив, который управляет выполнением операций репликации. Направление перемещения данных определяется сутью самой операции.

Интерфейсные порты управляющего массива должны быть зонированы с интерфейсными портами удаленного массива. С целью разрешения доступа удаленных устройств к интерфейсному порту управляющего массива в удаленном массиве должно быть выполнено маскирование LUN-устройств. В результате выполнения этих условий интерфейсные порты управляющего массива будут работать как адаптер главной шины (HBA), инициируя перенос данных в удаленное устройство и из него — в управляющее устройство.

В решениях, связанных с миграцией данных, для перемещения этих данных выполняются операции проталкивания (push) и выталкивания (pull). Эти

понятия определены с точки зрения управляющего массива. При *операции проталкивания*, или *push-операции*, данные перемещаются из управляющего массива в удаленный массив. Поэтому управляющее устройство действует как источник, а удаленное устройство играет роль приемника.

При *операции выталкивания*, или *pull-операции*, данные перемещаются из удаленного массива в управляющий массив. Удаленное устройство становится источником, а управляющее — приемником.

При инициировании push- или pull-операции для отслеживания процесса репликации в управляющем массиве создается защитный битовый массив. В каждом бите этого массива представлена часть данных, имеющихся на управляющем устройстве. Размер этой части варьируется в зависимости от способа реализации данной технологии. При инициировании операции репликации все биты устанавливаются в единицу, чем показывается, что копированию на устройство-приемник подлежит все содержимое устройства-источника. По мере копирования данных в ходе процесса репликации биты сбрасываются в нуль, показывая тем самым, что та или иная часть данных уже была скопирована. В конце процесса репликации все биты становятся нулевыми.

В ходе push- и pull-операций доступ хоста к удаленному устройству запрещается, потому что управляющий массив не контролирует удаленный массив и не в состоянии отследить какие-либо изменения, которые могут произойти в удаленном устройстве. Если в удаленное устройство в ходе push- и pull-операций вносятся какие-либо изменения, то целостность данных не может быть гарантирована. Push- и pull-операции могут совершаться либо в горячем, либо в холодном режиме. Эти понятия применяются только к управляющим устройствам. При проведении *холодной операции* доступ хоста к управляющему устройству в ходе репликации закрыт. Холодные операции гарантируют согласованность данных, поскольку и управляющее, и удаленное устройства находятся в режиме автономной работы. При проведении горячей операции управляющее устройство открыто для операций с хостом. В ходе горячих push- и pull-операций на управляющем устройстве могут происходить изменения, поскольку управляющий массив может их отслеживать, гарантируя тем самым целостность данных. При инициировании горячей push-операции приложения на устройствах управления могут быть в работоспособном состоянии. При создании защитного битового массива ввод-вывод на управляющих устройствах приостанавливается. Тем самым гарантируется согласованность PIT-образа с данными. Обращение к защитному битовому массиву происходит при каждой записи в управляющие устройства. Если бит сброшен, запись разрешается. Если бит установлен, процесс репликации удерживает входящую запись, копируя соответствующую часть данных в удаленное устройство, а затем разрешает завершить запись.

При проведении горячей pull-операции хосты могут обращаться к управляющим устройствам после того, как эта операция начнется. Обращение к защитному битовому массиву происходит при каждом чтении или записи. Если бит сброшен, чтение или запись проходят беспрепятственно. А если он установлен, чтение или запись удерживаются и процесс репликации копирует востребованную часть данных с удаленного устройства. Когда эта

часть будет скопирована на управляющее устройство, завершение чтения или записи будет разрешено. Управляющие устройства доступны для производственных операций практически сразу же после инициирования pull-операции и создания защитного битового массива.

Управляющий массив хранения данных может отслеживать изменения, вносимые в управляющие устройства, позволяя проводить push-операции в инкрементном режиме. Создается второй битовый массив, который называется ресинхронизационным. Как показано на рис. 12.13, *а*, при инициализации push-операции все биты в этом массиве устанавливаются в нуль (сбрасываются). При внесении изменении в управляющее устройство биты перебрасываются из нуля в единицу, показывая тем самым, что изменения действительно имели место (рис. 12.13, *б*). Когда требуется провести ресинхронизацию, происходит повторная инициализация push-операции и, как показано на рис. 12.13, *в*, битовый массив ресинхронизации становится новым защитным битовым массивом и удаленному устройству передаются только изменившиеся части данных. Проведение инкрементной pull-операции невозможно, потому что отслеживание изменений в удаленном устройстве не ведется.

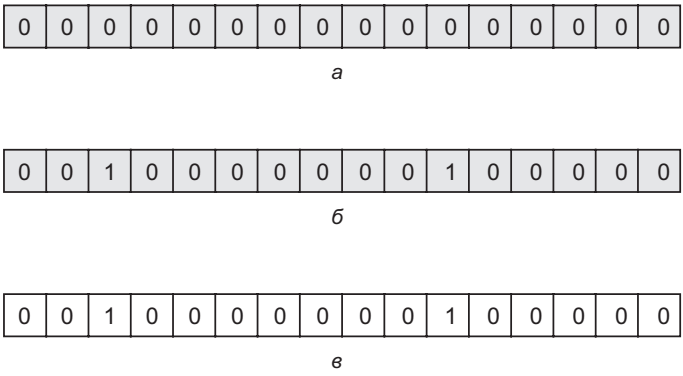


Рис. 12.13. Состояние битового массива в ходе выполнения push-операции:
а — состояние ресинхронизационного битового массива при инициализации push-операции;
б — состояние ресинхронизационного битового массива после обновления частей данных;
в — превращение ресинхронизационного битового массива в защитный битовый массив

12.5. Удаленная репликация и миграция в виртуализированной среде

В виртуализированной среде все данные виртуальной машины и файлы ее конфигурации, находящиеся в массиве хранения данных на основном месте производства, реплицируются в массив хранения данных, находящийся на удаленной площадке. Этот процесс на работе самих виртуальных машин не отражается. LUN-устройства реплицируются между двумя площадками с применением технологии репликации на основе использования массива хранения данных. Этот процесс репликации может быть либо синхронным

(при небольших расстояниях и с практически нулевым показателем RPO), либо асинхронным (при больших расстояниях и с показателем RPO больше нуля).

Еще одной технологией обеспечения непрерывности бизнес-процессов, применяемой в случае сбоя гипервизора или проведения планового технического обслуживания, является миграция виртуальной машины. Под этой миграцией понимается процесс перемещения виртуальных машин из одного гипервизора в другой без выключения самих виртуальных машин. Миграция виртуальных машин также помогает сбалансировать нагрузку, когда на одном гипервизоре запущены сразу несколько виртуальных машин, ведущих борьбу за выделяемые им ресурсы. Двумя наиболее востребованными технологиями миграции виртуальных машин являются миграции из гипервизора в гипервизор и из массива в массив.

При миграции виртуальных машин из гипервизора в гипервизор между этими гипервизорами переносится все активное состояние виртуальной машины. Эта миграция показана на рис. 12.14. При применении данного метода производится копирование содержимого памяти виртуальной машины из гипервизора-источника в гипервизор-приемник, а затем уже гипервизору-приемнику передается управление файлами виртуальных дисков. Поскольку виртуальные диски виртуальных машин не мигрируют, для данной технологии требуется доступ к одному и тому же хранилищу данных как со стороны гипервизора-источника, так и со стороны гипервизора-приемника.

При проведении миграции виртуальной машины из массива в массив виртуальные диски перемещаются из массива-источника в удаленный массив. Этот подход позволяет администратору перемещать виртуальные машины между не похожими друг на друга массивами хранения данных. Миграция виртуальной машины из массива в массив показана на рис. 12.15. Миграция из массива в массив начинается с копирования метаданных, характеризую-

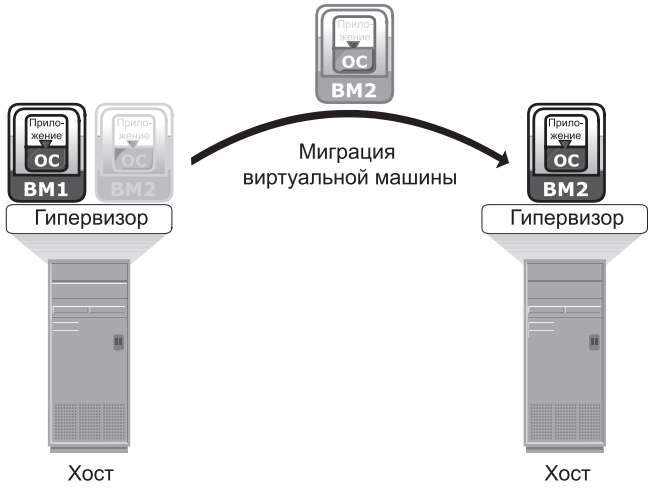


Рис. 12.14. Миграция виртуальной машины из гипервизора в гипервизор

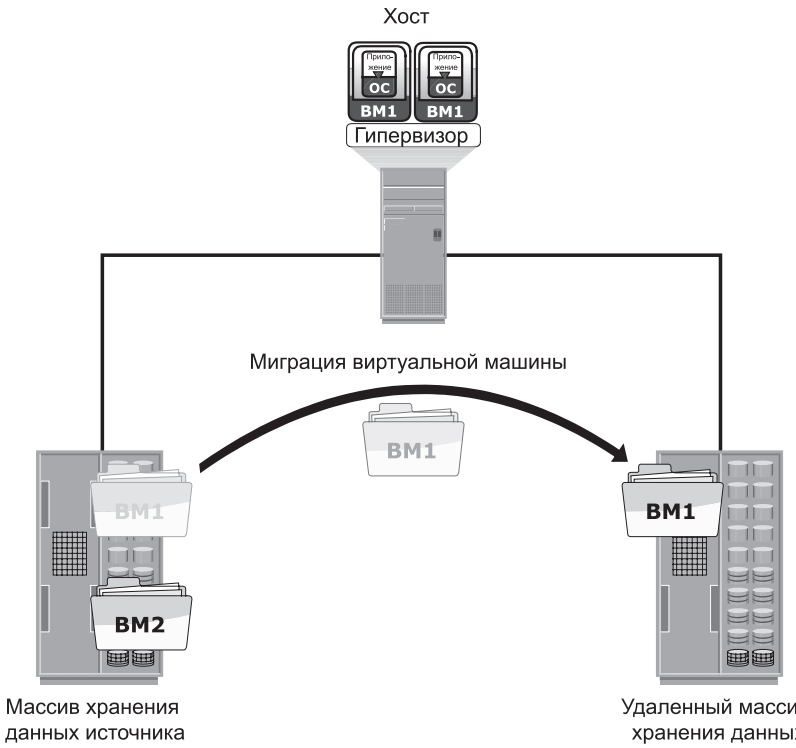


Рис. 12.15. Миграция виртуальной машины от массива к массиву

щих виртуальную машину, из массива-источника в приемник. Метаданные состоят главным образом из файлов конфигурации, подкачки и журналов. После копирования метаданных в новое место реплицируется дисковый файл виртуальной машины. В ходе репликации возможно обновление источника, поэтому для поддержки целостности данных возникает необходимость в отслеживании изменений в источнике. По завершении репликации на новое место реплицируются и те блоки, которые претерпели изменения со времени запуска данной репликации. Миграция виртуальных машин из массива в массив повышает производительность и способствует сбалансированности емкостей хранения данных путем перераспределения виртуальных дисков другим устройствам хранения.

12.6. Практическая реализация концепций: EMC SRDF, EMC MirrorView и EMC RecoverPoint

В этом разделе рассматривается продукция компании EMC, предназначенная для проведения удаленных репликаций. Два первых продукта, EMC Symmetrix Remote Data Facility (SRDF) и EMC MirrorView, представляют

собой программные средства для удаленных массивов хранения данных, поддерживаемых EMC Symmetrix и VNX соответственно. Третий продукт, EMC RecoverPoint, является решением, позволяющим проводить репликацию на основе использования сети. Всю самую последнюю информацию об этих продуктах можно найти на сайте www.emc.com.

12.6.1. EMC SRDF

Продукт SRDF предлагает целое семейство технологических решений для реализации удаленных репликаций на основе использования массивов хранения данных. К семейству программных продуктов SRDF относятся следующие средства:

- **SRDF/Synchronous (SRDF/S)** — решение, позволяющее проводить удаленные репликации, создавая синхронные реплики на одном или нескольких Symmetrix-приемниках, расположенных в пределах границ предприятия, мегаполиса или региона. Продукт SRDF/S предоставляет решение с практически нулевым показателем RPO, не допускающее потерю данных при возникновении локальных аварий;
- **SRDF/Asynchronous (SRDF/A)** — решение, позволяющее источнику проводить удаленные репликации в асинхронном режиме. В нем имеется встроенный набор дельта-технологий, позволяющих выстраивать последовательность записей путем использования механизма буферизации. При возникновении региональных катаклизмов SRDF/A обеспечивает минимально возможную потерю данных;
- **SRDF/DM** — решение, позволяющее проводить миграцию данных на большие расстояния из источника в том-приемник;
- **SRDF/Automated Replication (SRDF/AR)** — решение, позволяющее проводить удаленную репликацию. В этом решении для реализации технологии репликации с буферизацией диска используется не только SRDF, но TimeFinder/Mirror. Этот продукт предлагается как с одним транзитным участком — SRDF/AR Single-hop для двусторонней репликации, так и с несколькими транзитными участками — SRDF/AR Multihop для трехсторонней каскадной репликации. Продукт SRDF/AR предлагает решение для работы на больших расстояниях с показателем RPO, составляющим порядка нескольких часов;
- **SRDF/Star** — решение, позволяющее проводить трехстороннюю удаленную репликацию с несколькими приемниками. Предусматривает наличие трех площадок: основной (производственной), второстепенной (бункер) и третьестепенной (удаленной). Репликация между основной и второстепенной площадками проводится в синхронном режиме, а репликация между основной и третьестепенной — в асинхронном режиме. Если выйдет из строя оборудование, находящееся на основной площадке, EMC-решение SRDF/Star позволит организациям быстро переместить работу в одно из удаленных мест и восстановить удаленную репликацию между оставшимися двумя площадками.

12.6.2. EMC MirrorView

Программный продукт MirrorView позволяет устройству хранения данных EMC VNX проводить удаленную репликацию на основе использования массива хранения данных. Содержимое основного тома реплицируется на второстепенный том, который находится в другой системе хранения VNX. В семейство MirrorView входят решения MirrorView/Synchronous (MirrorView/S) и MirrorView/Asynchronous (MirrorView/A).

12.6.3. EMC RecoverPoint

Продукт EMC RecoverPoint Continuous Remote Replication (CRR) представляет собой решение, обеспечивающее полную защиту данных и предоставляющее двунаправленную синхронную и асинхронную репликацию. В обычном режиме работы RecoverPoint CRR позволяет пользователям удаленно восстанавливать данные на любой момент времени. RecoverPoint динамически переключается между синхронной и асинхронной репликацией на основе политики, определяемой в отношении производительности и времени отклика.

Резюме

В данной главе были подробно рассмотрены технологии удаленной репликации. Удаленная репликация предоставляет решения, позволяющие проводить аварийное восстановление работоспособности и аварийный перезапуск бизнес-процессов. Она позволяет при выходе оборудования из строя проводить быстрый перезапуск бизнес-операций на удаленной площадке с приемлемыми потерями данных.

Удаленная реплика используется и для проведения других бизнес-операций, например резервного копирования, составления отчетов и тестирования. Разделение бизнес-операций между источником и приемником защищает источник от возникновения обстоятельств, снижающих его производительность, обеспечивая повышение производительности основного приложения.

Удаленная репликация помогает также выполнять миграцию в дата-центрах и снижает уровень отрицательного воздействия на производственные операции, поскольку не оказывает никакого влияния на обращение приложений к данным источника.

В главе были рассмотрены также различные типы решений, позволяющих проводить удаленную репликацию. При решении о том, какую технологию удаленной репликации следует развертывать, основным фактором выступает расстояние между основной и удаленной площадками. Асинхронная репликация позволяет этим площадкам находиться на большем удалении друг от друга при вполне приемлемых показателях RPO и RTO. Трехсторонняя репликация снижает риски потери работоспособности, возникающие в случае региональных катаклизмов, при двусторонней репликации. А непрерывная

защита данных является усовершенствованным решением на основе использования сети, позволяющим проводить как локальную, так и удаленную репликацию с неограниченным количеством точек восстановления. В данной главе рассматривались также удаленная репликация и миграция виртуальных машин в виртуализированной среде.

Чтобы в современной, задающей высокие темпы, требующей постоянной доступности и крайне взаимосвязанной мировой экономике организации могли сохранять высокий уровень конкурентоспособности, они должны проявлять гибкость, приспособляемость и способность быстро реагировать на изменения рыночной ситуации. Облако, являющееся следующим поколением способа производства вычислений, обеспечивает возможность широкого масштабирования и получения гибко настраиваемых вычислительных ресурсов, предоставляемых по мере надобности. Облачной инфраструктуре и облачным услугам и будет посвящена следующая глава.

УПРАЖНЕНИЯ

1. Какие факторы следует учесть при определении порядка реализации синхронной удаленной репликации?
2. Расскажите, какого показателя RPO можно достичь при удаленной репликации в синхронном и асинхронном режимах, а также в режиме с буферизацией диска.
3. Объясните, в какой степени сбой бункера в трехсторонней репликации окажет влияние на следующие реализации:
   - с несколькими транзитными участками — с методом, использующим синхронный режим + режим буферизации диска;
   - с несколькими транзитными участками — с методом, использующим синхронный режим + асинхронный режим;
   - с несколькими приемниками.
4. Объясните, в какой степени сбой источника в трехсторонней репликации окажет влияние на следующие реализации, и назовите доступные варианты восстановления работоспособности предприятия:
   - с несколькими транзитными участками — с методом, использующим синхронный режим + режим буферизации диска;
   - с несколькими транзитными участками — с методом, использующим синхронный режим + асинхронный режим;
   - с несколькими приемниками.
5. База данных хранится в 10 LUN-устройствах, представляющих собой 10 массивов RAID 1 емкостью по 9 Гбайт. Для восстановления работоспособности после серьезной аварии было выбрано решение, позволяющее проводить трехстороннюю каскадную удаленную репликацию в синхронном режиме и режиме буферизации диска. Все задействованные в решении LUN-устройства обладают защитой, обеспечиваемой массивом RAID 1. Вычислите общую исходную емкость этого массива, которая потребуется для реализации данного решения.

Раздел

IV

Облачные вычисления

В ЭТОМ РАЗДЕЛЕ

Глава 13. Облачные вычисления

Глава 13
Облачные вычисления

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Основные характеристики облачных вычислений
- Облачные услуги и модели развертывания
- Инфраструктура облачных вычислений
- Особенности внедрения облачных вычислений

В современной конкурентной среде организации находятся под постоянно возрастающим давлением и вынуждены повышать эффективность своей деятельности, а также вносить изменения в IT-процессы, с тем чтобы добиваться большего с меньшими затратами. Чтобы адекватно реагировать на постоянно меняющиеся условия ведения бизнеса и ускоренные темпы инноваций, предприятия должны сокращать сроки вывода своей продукции на рынок, быть проворнее, повышать свою доступность и сокращать расходы. В таких условиях ведения бизнеса IT-отделам приходится решать целый ряд непростых задач, основными из которых являются круглосуточное и повсеместное обслуживание клиентов, быстрое обновление технологий и ускоренное предоставление IT-ресурсов, и все это с сокращением затрат.

Все эти извечные проблемы решаются с появлением нового стиля так называемых облачных вычислений, позволяющих организациям и индивидуальным предпринимателям получать и предоставлять IT-ресурсы в виде услуг. Используя облачные вычисления, пользователи могут просматривать и выбирать через портал нужные им облачные услуги, предоставляющие вычислительные мощности, программы, хранилища данных или сочетание этих ресурсов. Облачные вычисления автоматизируют поставку пользователям выбранных облачных услуг. Они помогают организациям и индивидуальным предпринимателям развертывать IT-ресурсы с сокращением совокупной стоимости владения, более быстрым предоставлением услуг и соблюдением соответствующих норм.

Общепринятое определение облачных вычислений, предложенное Национальным институтом стандартов и технологий США — U.S. National Institute of Standards and Technology (NIST Special Publication 800-145), гласит:

> *Облачные вычисления — это модель, позволяющая легко осуществлять сетевой доступ по требованию к единому пулу настраиваемых вычислительных ресурсов (например, сетей, серверов, систем хранения данных, приложений и услуг), которые быстро выделяются для использования и затем так же быстро высвобождаются. При этом затрачивается минимум усилий на администрирование и взаимодействие с поставщиком услуг.*

В данной главе рассматриваются высокоэффективные технологии, основные характеристики, преимущества, услуги, модели развертывания и инфраструктура облачных вычислений. В главу также включены проблемные вопросы и факторы, определяющие порядок внедрения облачных вычислений в повседневную практику.

13.1. Высокоэффективные облачные технологии

К высокоэффективным облачным технологиями относятся grid-вычисления, utility-вычисления, виртуализация и сервис-ориентированная архитектура.

- **Grid-вычисления** представляют собой разновидность распределенного вычисления, позволяющую предоставлять ресурсы многочисленных разнородных компьютеров, находящихся в сети, для совместной работы по одновременному решению единой задачи. Grid-вычисления позволяют производить параллельные вычисления и наилучшим образом подходят для решения задач с большими объемами работ.
- **Utility-вычисления** представляют собой модель предоставления услуг, в которой поставщик каждой отдельно взятой услуги предоставляет клиентам доступ к вычислительным ресурсам по запросу и ведет учет предоставляемых услуг по факту их потребления. Здесь прослеживается аналогия с предоставлением жилищно-коммунальных услуг, например с обеспечением электроэнергией, где учет ведется на основе фактического потребления.
- **Виртуализация** представляет собой технологию, позволяющую пользователям IT-ресурсов абстрагироваться от их физических характеристик. Она дает возможность просматривать ресурсы и управлять ими в составе пула, а также позволяет пользователям создавать из пула виртуальные ресурсы. Виртуализация дает возможность проявлять более высокий уровень гибкости в предоставлении IT-ресурсов по сравнению с выполнением этой же задачи в невиртуализированной среде.

Она помогает оптимизировать использование ресурсов и повысить эффективность их предоставления.

- **Сервис-ориентированная архитектура** — Service Oriented Architecture (SOA) — предоставляет набор служб, способных взаимодействовать друг с другом. Эти службы ведут совместную работу для выполнения определенных действий или просто обмениваются данными друг с другом.

13.2. Характеристики облачных вычислений

Вычислительная инфраструктура, используемая для предоставления облачных услуг, должна обладать определенными возможностями или характеристиками. В соответствии с положениями, выработанными NIST, облачная инфраструктура должна иметь пять основных характерных особенностей.

- **Самообслуживание по требованию.** Потребитель в инициативном порядке может быть автоматически снабжен вычислительными ресурсами, такими как серверное время и сетевое хранилище, по факту потребности, без участия представителей поставщиков услуг.

  Поставщик облачных услуг публикует перечень услуг, содержащий информацию обо всех облачных услугах, доступных потребителям. Перечень услуг включает информацию о характере услуг, ценах и способах их запросов. Потребители просматривают перечень услуг с помощью пользовательского веб-интерфейса и используют его для запроса услуги. Потребители могут либо воспользоваться теми услугами, которые уже готовы к использованию (ready-to-use), либо изменить некоторые параметры обслуживания, чтобы настроить услуги под свои запросы.
- **Широкополосный сетевой доступ.** Возможности, доступные по сети, обращение к которым осуществляется с использованием стандартных механизмов, способствующих их использованию разнородными, как полными, так и неполными клиентскими платформами (например, мобильными телефонами, планшетными компьютерами, ноутбуками и рабочими станциями).
- **Создание пула ресурсов.** Вычислительные ресурсы поставщика объединяются в пул для обслуживания нескольких потребителей с использованием модели, допускающей наличие множества арендаторов различных физических и виртуальных ресурсов, динамически выделяемых и высвобождаемых в соответствии с потребительскими

нуждами. У клиента создается чувство независимости от местонахождения за счет того, что клиент, как правило, не контролирует точное местонахождение предоставляемых ресурсов или даже не знает о нем, но при этом может указать местонахождение на более высоком уровне абстракции (например, страну, штат или дата-центр). В качестве примеров ресурсов могут послужить хранилища данных, средства обработки данных, память и полоса пропускания сети.

- **Оперативность и гибкость.** С целью быстрого масштабирования, соразмерного с внешним и внутренним спросом, ресурсы могут быть гибко предоставлены и высвобождены, причем в ряде случаев в автоматическом режиме. Потребителю доступные для предоставления ресурсы зачастую видятся неограниченными и предоставляемыми в любом объеме и в любое время.

  При резких колебаниях потребностей в IT-ресурсах потребители могут воспользоваться преимуществами оперативности и гибкости облачной среды. Например, организации в течение определенного периода времени для выполнения определенной задачи может понадобиться двойное количество веб-серверов и серверов приложений. В оставшийся период они с целью экономии средств могут освободиться от неиспользуемых серверных ресурсов. Облако позволяет потребителям увеличивать и сокращать количество ресурсов динамически.
- **Оплата услуг по факту потребления.** Облачные системы автоматически контролируют и оптимизируют использование ресурсов, применяя возможности измерения на уровне абстракции, соответствующем типу услуги (например, по факту хранения, обработки, предоставления полосы пропускания и по времени активности учетных записей пользователей). Использование ресурсов может отслеживаться, контролироваться и заноситься в отчет, обеспечивая прозрачность и поставщику, и потребителю используемой услуги.

МНОГОПОЛЬЗОВАТЕЛЬСКИЙ РЕЖИМ

Многопользовательский режим относится к архитектуре, в которой несколько независимых потребителей (арендаторов) обслуживаются с использованием одного и того же набора ресурсов. Это снижает стоимость услуг для потребителей. Виртуализация позволяет объединять ресурсы в пул и обеспечивать в облаке возможность обслуживания сразу нескольких арендаторов. Например, несколько виртуальных машин разных потребителей могут работать одновременно на одном и том же физическом сервере с запущенным гипервизором.

13.3. Преимущества, получаемые от облачных вычислений

Облачные вычисления предлагают следующие основные преимущества.

- **Уменьшение стоимости IT-услуг.** Облачные услуги могут приобретаться на основе оплаты фактически потребленных объемов или по цене их абонирования. Тем самым снижаются или исключаются капитальные расходы (CAPEX) потребителей на IT-инфраструктуру.
- **Адаптивность бизнеса.** Облачные вычисления дают возможность быстро выделять и масштабировать вычислительные мощности. Они способны сократить время, требующееся на предоставление и развертывание новых приложений, с месяцев до минут. Это позволяет предприятию намного быстрее реагировать на рыночные перемены и сокращать срок вывода продукции на рынок.
- **Гибкое масштабирование.** Облачные вычисления позволяют потребителям легко наращивать и сокращать вычислительные мощности, расширять и сужать границы их применения в соответствии с потребностями в компьютерных ресурсах.

  Потребители могут в одностороннем порядке или автоматически масштабировать вычислительные ресурсы без взаимодействия с поставщиками облачных услуг. Присущая облачным вычислениям возможность гибкого предоставления услуг зачастую формирует у потребителей облачных вычислений ощущение возможности неограниченного масштабирования.
- **Высокая доступность.** Облачные вычисления могут гарантировать доступность ресурсов на различных уровнях в зависимости от политики и приоритетов потребителей. Отказоустойчивость развернутых облачных систем гарантируется избыточной инфраструктурой компонентов (серверов, сетевых маршрутов и оборудования для хранения данных наряду с программами кластеризации). Эти технологии могут охватывать несколько дата-центров, находящихся в различных географических районах, что предотвращает недоступность данных из-за сбоев в каком-нибудь из районов.

13.4. Модели облачного обслуживания

В соответствии с положениями NIST предложения облачного обслуживания в основном сводятся к трем моделям: инфраструктура как услуга — Infrastructure-as-a-Service (IaaS), платформа как услуга — Platform-as-a-Service (PaaS) и программное обеспечение как услуга — Software-as-a-Service (SaaS).

13.4.1. Инфраструктура как услуга

Потребителю предоставляется возможность использования вычислительных мощностей, хранилища данных, сетей и других основных физических ресурсов, на которых потребитель может развертывать и запускать какое угодно программное обеспечение, включая операционные системы и приложения. Потребители не занимаются управлением или контролем основной облачной инфраструктуры, но им доступен контроль над операционными системами и развернутыми приложениями и, возможно, ограниченный контроль над избранными сетевыми компонентами (например, над межсетевым экраном хоста).

Основным уровнем стека облачных услуг является IaaS (рис. 13.1, *а*). Он служит фундаментом для двух других уровней, SaaS и PaaS.

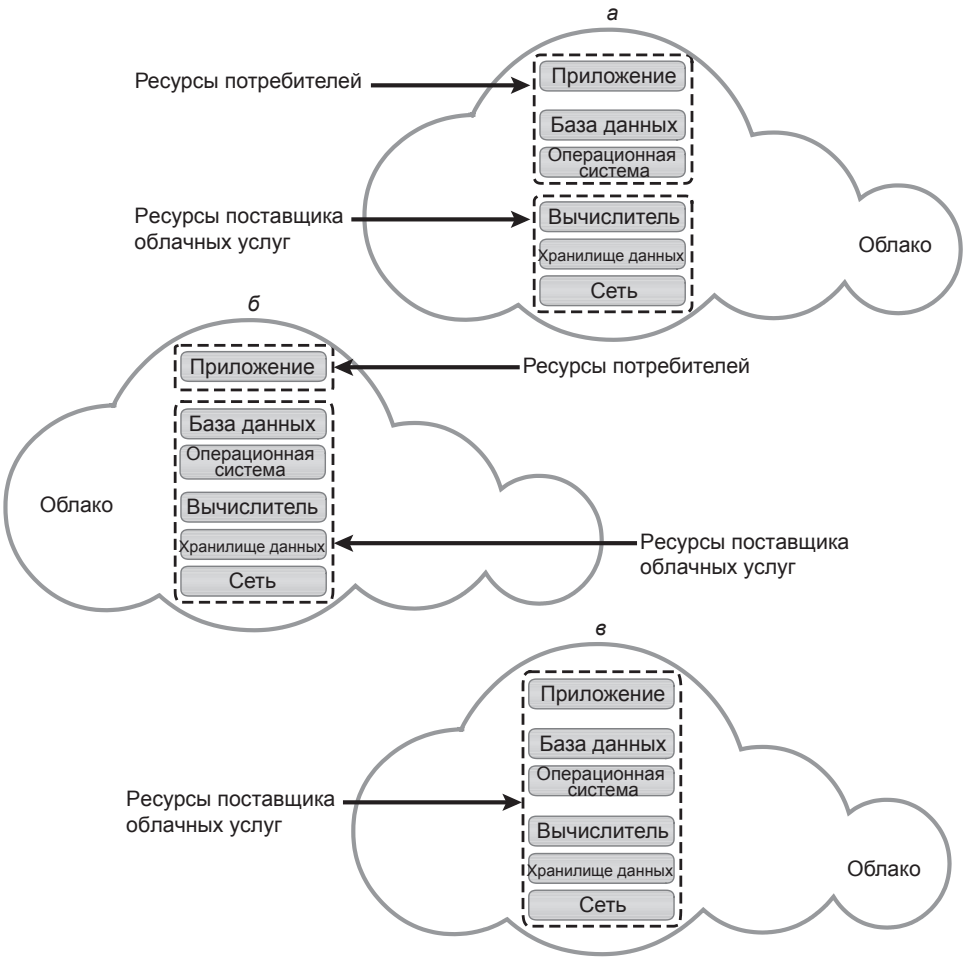


Рис. 13.1. Модели: *а* — IaaS; *б* — PaaS; *в* — SaaS

Примером IaaS-модели, предоставляющей в облаке масштабируемые по запросу вычислительные мощности, является облако Amazon Elastic Compute Cloud (Amazon EC2). Оно позволяет потребителям воспользоваться крупной вычислительной инфраструктурой компании Amazon без первоначальных капиталовложений.

13.4.2. Платформа как услуга

Потребителю предоставляется возможность развернуть в облачной инфраструктуре созданные им самим или купленные приложения, при создании которых использовались языки программирования, библиотеки, службы и инструментальные средства, поддерживаемые поставщиком облачных вычислений. Потребитель не имеет возможности управлять основной облачной инфраструктурой, включая сеть, серверы, операционные системы или хранилища данных, или контролировать ее, но может контролировать развернутые приложения и, возможно, настройки конфигурации той среды, в которой развертываются приложения (см. рис. 13.1, *б*).

PaaS-модель также используется как среда развертывания приложения, при этом поставщик облачных вычислений предлагает ее в качестве услуги. Потребитель может воспользоваться этими платформами для создания кода своих приложений с последующим развертыванием приложений в облаке. Поскольку рабочая нагрузка, требующаяся для развертывания приложений, изменяется, вычислительной платформой обычно гарантируется незаметная для потребителя масштабируемость вычислительных ресурсов. Примерами реализации PaaS-модели могут послужить Google App Engine и Microsoft Windows Azure Platform.

13.4.3. Программное обеспечение как услуга

Потребителю предоставляется возможность использовать приложения поставщика, запущенные в облачной инфраструктуре. Приложения доступны на различных устройствах клиента либо через неполную клиентскую инфраструктуру, такую как веб-браузер (в качестве примера можно привести электронную почту на основе веб-технологий) или программный интерфейс. Потребителю не дается возможность управления или контроля над основной облачной инфраструктурой, включая сеть, серверы, операционные системы, хранилища данных или даже отдельные возможности приложения, за исключением, может быть, ограниченных настроек конфигурации приложений для конкретного круга пользователей (см. рис. 13.1, *в*).

В SaaS-модели поставщики облачных услуг предлагают такие приложения, как управление взаимоотношениями с клиентами — customer relationship management (CRM), электронная почта и программа мгновенного обмена сообщениями — instant messaging (IM). Для поддержки своих услуг нужной для этого компьютерной инфраструктурой управляют исключительно поставщики облачных услуг. Потребителям может быть разрешено вносить

изменения в некоторые настройки конфигурации для подстраивания приложений под свои нужды.

Примером SaaS-модели является EMC Mozy. Потребители могут без особого труда воспользоваться консолью Mozy для осуществления автоматизированного безопасного онлайнового резервного копирования и восстановления своих данных. А компания Salesforce.com является поставщиком таких CRM-приложений на основе SaaS-модели, как Sales Cloud и Service Cloud.

13.5. Модели развертывания облака

В соответствии с положениями NIST в облачных вычислениях выделяются четыре модели развертывания: публичная (public), частная (private), общественная (community) и гибридная (hybrid), — которые закладывают основы порядка построения и потребления возможностей облачной инфраструктуры.

13.5.1. Публичное облако

В модели *публичного облака* облачная инфраструктура подготовлена для открытого использования широким кругом лиц. Этой инфраструктурой могут владеть, управлять и распоряжаться деловые, академические или правительственные организации или определенные их сочетания. Она находится на территории поставщика облачных услуг.

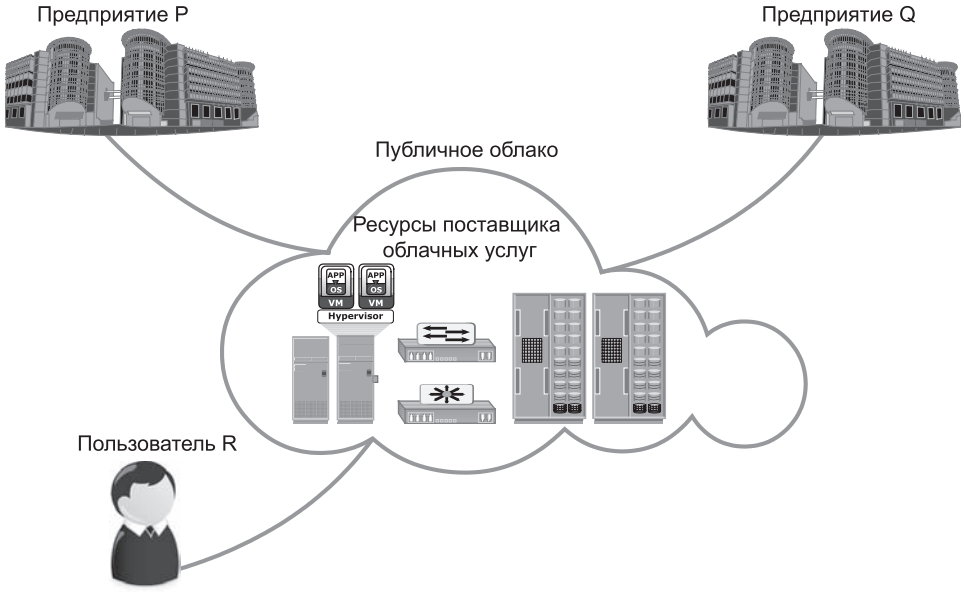


Рис. 13.2. Публичное облако

Потребители пользуются облачными услугами, предоставляемыми поставщиками через Интернет, и рассчитываются за них по зафиксированному факту использования или путем внесения абонентской платы. Преимуществом публичного облака являются низкие капитальные вложения наряду с широкими возможностями для масштабирования. Но для потребителей эти преимущества сопряжены с определенными рисками, связанными с отсутствием контроля над имеющимися в облаке ресурсами, недостаточным уровнем обеспечения безопасности конфиденциальных данных, проблемами производительности сети и вопросами обеспечения совместимости программных продуктов. Популярными поставщиками публичных облачных услуг являются компании Amazon, Google и Salesforce.com. На рис. 13.2 показано публичное облако, предоставляющее облачные услуги организациям и частным лицам.

13.5.2. Частное облако

В модели *частного облака* облачная инфраструктура подготовлена для эксклюзивного использования одной организацией, включающей нескольких потребителей (в качестве которых могут выступать ее подразделения). Этой инфраструктурой может владеть, управлять и распоряжаться организация, третья сторона или ряд их сочетаний, и она может развертываться как на собственном, так и на стороннем оборудовании. Приведем два примера модели частного облака.

- **Частное облако на собственном оборудовании.** Известно также как внутреннее облако и содержится организацией в ее собственных датацентрах (рис. 13.3, *а*). Эта модель позволяет организациям привести процессы управления и обеспечения безопасности к определенным стандартам, хотя у нее есть ограничения по размерам и масштабируемости ресурсов. Организации также могут быть вынуждены нести капитальные и операционные расходы на физические ресурсы. Больше всего такая модель подходит тем организациям, которым требуется полный контроль над их приложениями, настройками инфраструктуры и механизмами обеспечения безопасности.
- **Частное облако, развернутое за пределами территории предприятия.** Этот тип частного облака находится за пределами организации (см. рис. 13.3, *б*) и управляется сторонней организацией, обслуживающей эксклюзивную облачную среду для конкретной организации с полной гарантией закрытости и конфиденциальности.

13.5.3. Общественное облако

В модели *общественного облака* облачная инфраструктура подготовлена для эксклюзивного использования определенным сообществом потребителей из

организаций, имеющих общие интересы (например, общие задачи, требования по обеспечению безопасности, политики и требования по обеспечению совместимости). Этой инфраструктурой могут владеть управлять и распоряжаться одна или несколько организаций сообщества, третья сторона или определенные их сочетания, и она может создаваться как на собственном, так и на стороннем оборудовании (рис. 13.4).

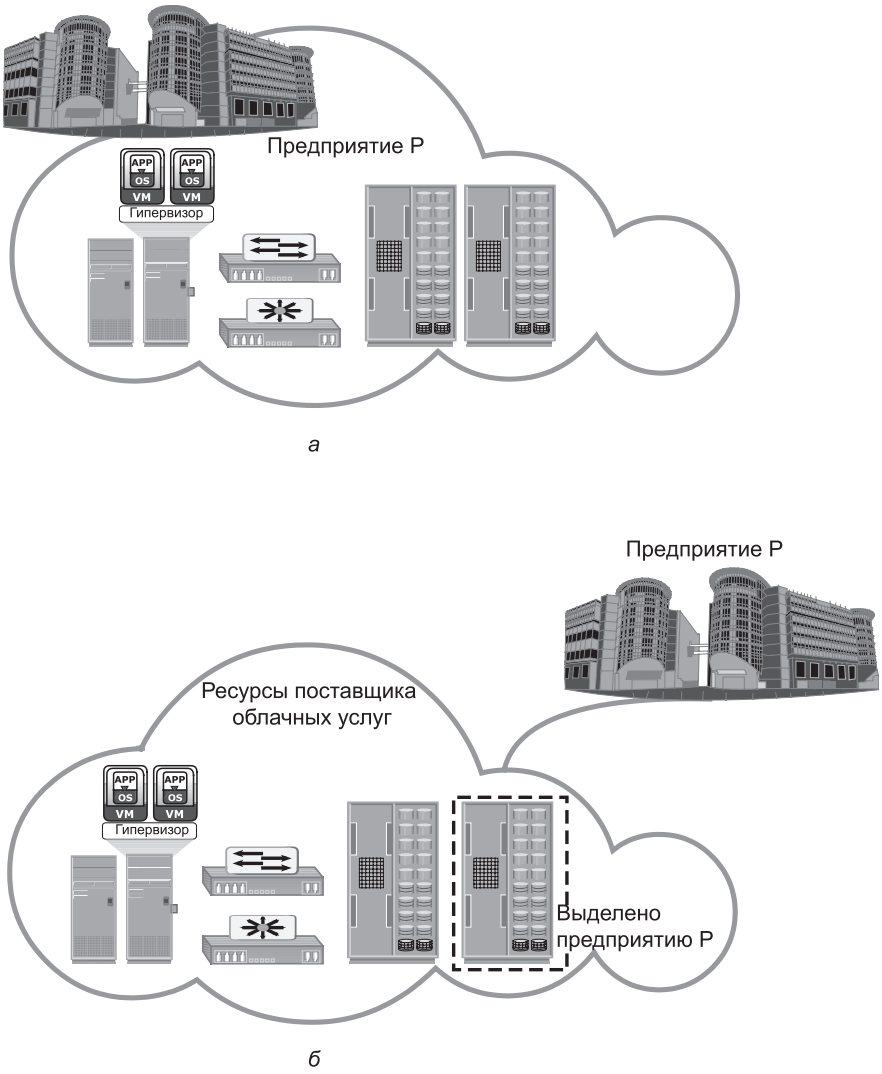


Рис. 13.3. Частные облака: *а* — на собственном оборудовании; *б* — закрытое облако, развернутое за пределами территории предприятия

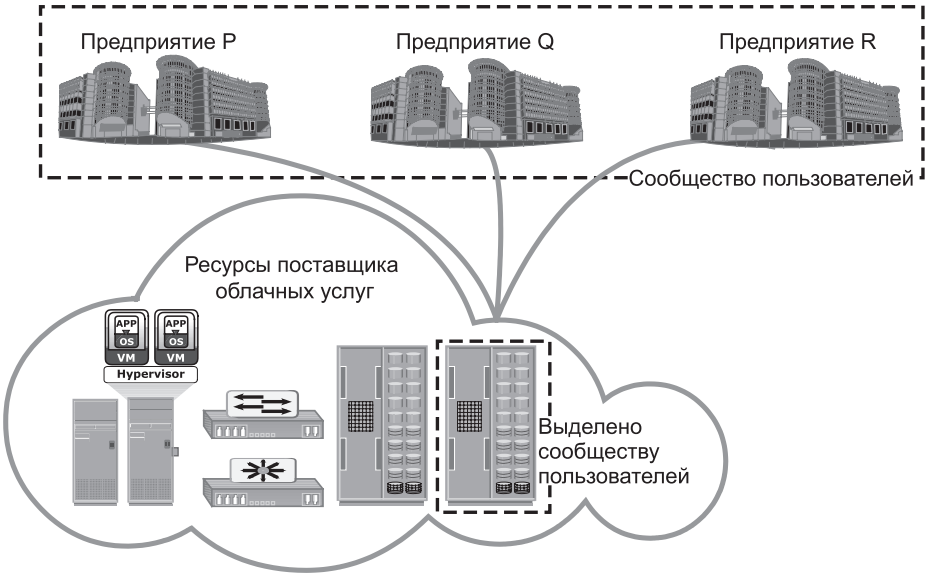


Рис. 13.4. Общественное облако

В общественном облаке затраты распределяются среди меньшего количества потребителей, чем в публичном облаке. Поэтому данный вариант обходится дороже, но может предоставить более высокий уровень секретности, безопасности и совместимости. По сравнению с публичным облаком общественное облако предлагает организациям доступ к более широкому пулу ресурсов. К примеру, общественное облако может быть востребовано правительственными учреждениями. Если различные учреждения в правительстве руководствуются одними и теми же нормативными положениями, то все они могут совместно использовать одну и ту же инфраструктуру и сократить тем самым свои индивидуальные вложения.

13.5.4. Гибридное облако

В модели *гибридного облака* облачная инфраструктура представляет собой сочетание из двух и более отдельных облачных инфраструктур (закрытых, общественных или публичных), которые по-прежнему остаются уникальными объектами, но связаны едиными технологическими стандартами или формами владения, позволяющими осуществлять перенос данных и приложений (например, разрывать облако на части с целью балансировки нагрузки между облаками).

Гибридная модель позволяет организациям развертывать менее важные приложения и данные в публичном облаке, используя возможности масштабирования и экономии средств, предоставляемые этим облаком. А приложения и данные, важные для деятельности организации, остаются при этом

в закрытом облаке, которое предоставляет повышенные меры безопасности. Пример гибридного облака показан на рис. 13.5.

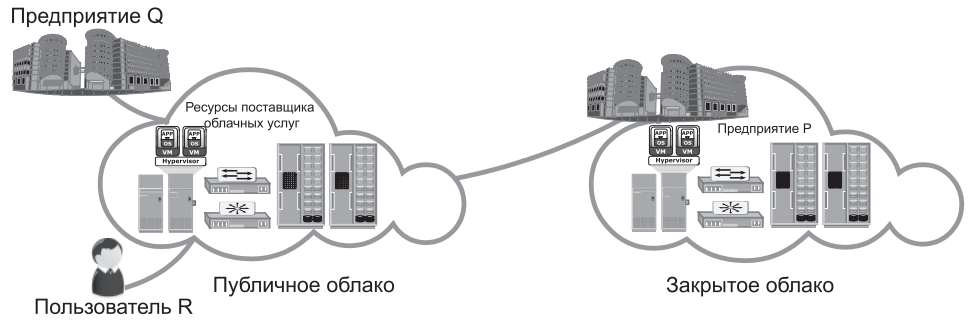


Рис. 13.5. Гибридное облако

13.6. Инфраструктура облачных вычислений

Инфраструктура облачных вычислений представляет собой совокупность оборудования и программного обеспечения, которая отвечает пяти наиболее важным характеристикам облачных вычислений. Инфраструктура облачных вычислений обычно состоит из следующих уровней:

- физической инфраструктуры;
- виртуальной инфраструктуры;
- приложений и программного обеспечения платформы;
- программы управления облаком и инструментов для создания услуг.

Для предоставления облачных услуг потребителям ресурсы этих уровней объединяются, а их работа координируется (рис. 13.6).

13.6.1. Физическая инфраструктура

Физическая инфраструктура состоит из физических вычислительных ресурсов, куда включаются физические серверы, системы хранения данных и сети. Физические серверы соединены друг с другом, подключены к системам хранения данных и клиентам посредством таких сетей, как IP, FC SAN, IP SAN или FCoE.

Поставщики облачных услуг могут для их предоставления использовать физические вычислительные ресурсы из одного или нескольких дата-центров. Если вычислительные ресурсы распределены по нескольким дата-центрам, то должна быть обеспечена возможность их взаимодействия, позволяющая дата-центрам, находящимся в разных местах, работать в качестве единого большого дата-центра. Тем самым будет обеспечена возможность миграции бизнес-приложений и данных между дата-центрами и подготовка облачных услуг, использующих ресурсы из нескольких дата-центров.

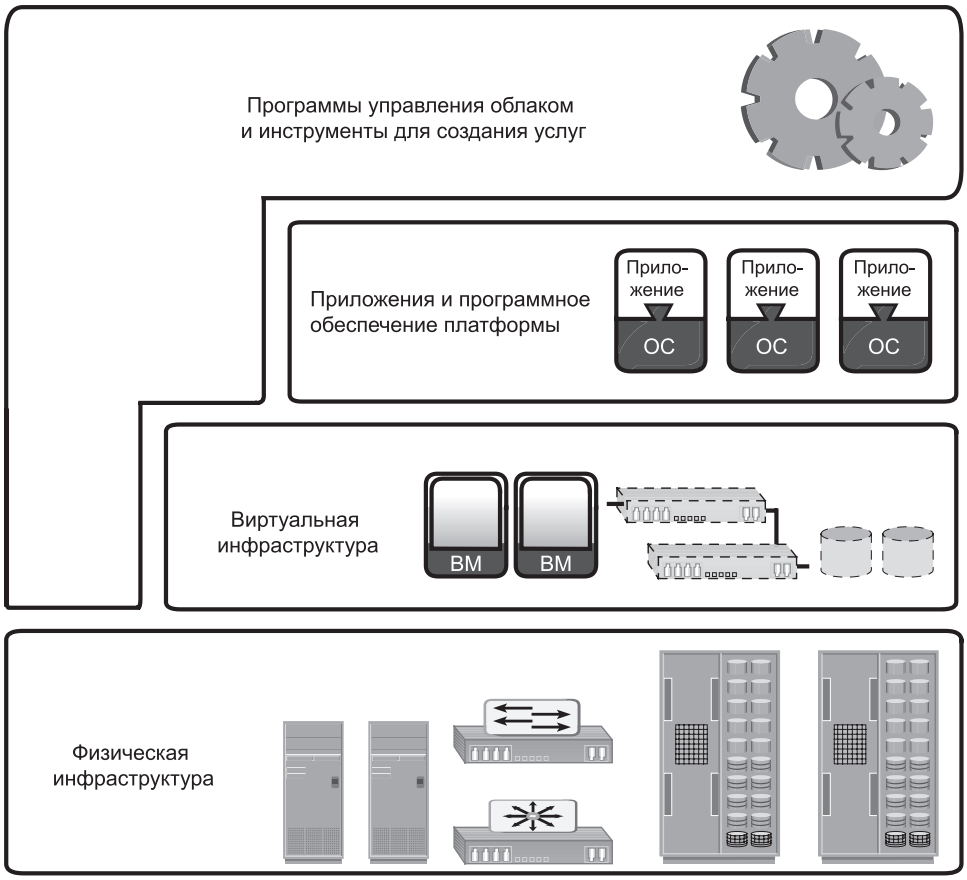


Рис. 13.6. Уровни облачной инфраструктуры

13.6.2. Виртуальная инфраструктура

Поставщики облачных услуг используют технологии виртуализации для создания уровня виртуальной инфраструктуры в виде надстройки над физической инфраструктурой. Виртуализация позволяет обеспечивать соответствие некоторым облачным характеристикам, таким как создание пула и быстрая адаптация к изменению условий использования. Она также позволяет снизить затраты на предоставление облачных услуг.

У некоторых поставщиков облачных услуг принадлежащая им физическая инфраструктура может быть пока еще не полностью виртуализирована, но они внедряют виртуализацию для повышения эффективности и оптимизации работы облака.

Виртуализация позволяет абстрагироваться от физических вычислительных ресурсов и создавать общее представление о ресурсном потенциале. Например, пул ресурсов может объединять в кластер центральные

процессоры физических серверов. Мощность пула ресурсов складывается из мощности всех доступных в кластере центральных процессоров (например, 10 000 МГц). В дополнение к пулу центральных процессоров виртуальная инфраструктура включает другие типы пулов ресурсов, такие как пул памяти, сетевой пул и пул устройств хранения данных. Кроме пулов ресурсов виртуальная инфраструктура также включает пулы идентификации, такие как пулы VLAN ID и VSAN ID. Количество пулов каждого типа и мощность пула зависят от требований к поставщику облачных услуг по созданию различных облачных услуг.

Виртуальная инфраструктура также включает виртуальные вычислительные ресурсы, такие как виртуальные машины, виртуальные тома хранилищ данных и виртуальные сети. Эти ресурсы получают из пулов ресурсов мощности центральных процессоров, объемы памяти, полосы пропускания сети и пространства хранилищ данных. Эти мощности на основе требований по обслуживанию потребителя легко и гибко распределяются виртуальным вычислительным ресурсам. Виртуальные сети создаются с использованием сетевых идентификаторов, таких как VLAN ID и VSAN ID, из соответствующих пулов идентификации. Виртуальные вычислительные ресурсы используются для создания услуг, предоставляемых облачной инфраструктурой.

13.6.3. Приложения и программное обеспечение платформы

На этом уровне находится набор бизнес-приложений и программного обеспечения платформы, такого как операционная система и база данных. Программное обеспечение платформы предоставляет среду, в которой запускаются бизнес-приложения. Для создания SaaS и PaaS приложения и программное обеспечение платформы помещаются на виртуальные машины. Для SaaS-модели поставщиком облачных услуг предоставляются как приложение, так и программное обеспечение платформы. Что же касается PaaS-модели, здесь поставщики облачных услуг предоставляют только программное обеспечение платформы, а потребители экспортируют свои приложения в облако.

13.6.4. Программы управления облаком и инструменты для создания услуг

Уровень программ управления облаком и инструментов для создания услуг включает программы трех типов:

- программы управления физической и виртуальной инфраструктурой;
- единую программу управления;
- программы управления пользовательским доступом.

Эта классификация основана на различных функциях программ. С целью предоставления облачных услуг эти программы взаимодействуют друг с другом.

Программы управления физической и виртуальной инфраструктурой предлагаются как поставщиками различных ресурсов инфраструктуры, так и сторонними организациями. Например, у массива хранения данных имеется собственная программа управления. Сети и физические серверы также имеют независимое управление с использованием соответствующих программ управления сетью и программ управления компьютерами. Для создания виртуальной инфраструктуры из имеющейся в распоряжении физической инфраструктуры этими программами предоставляются специальные интерфейсы.

Единая программа управления взаимодействует со всеми отдельными программами управления физической и виртуальной инфраструктуры. Она собирает информацию о существующих настройках физической и виртуальной инфраструктуры, об имеющихся там подключениях и о степени их использования. Единая программа управления создает из этой информации сводные данные и дает общее представление о ресурсах инфраструктуры, которые могут быть рассредоточены по одному или нескольким дата-центрам. Это позволяет администратору централизованно отслеживать показатели производительности, емкости и доступности физических и виртуальных ресурсов. Единая программа управления предоставляет также единый интерфейс управления для настройки физической и виртуальной инфраструктуры и объединения вычислительных пулов (имеются в виду центральные процессоры и память), а также пулов сетей и пулов устройств хранения данных. Объединение позволяет сводить в группу вычислительные пулы для использования пулов сетей и пулов устройств хранения данных для переноса и хранения данных соответственно. Единая программа управления передает команды настройки соответствующим программам управления физической и виртуальной инфраструктурой, которые выполняют эти команды. Это позволяет избавиться от отдельного администрирования ресурсов производства вычислений, хранения данных и сети, использующих собственные программы управления.

Основной функцией единой программы управления является автоматизация создания облачных услуг. Она позволяет администраторам определять такие показатели услуг, как мощность центрального процессора, емкость памяти, полоса пропускания сети, емкость хранилища данных, имена и описания приложений и программ, обеспечивающих работу платформы, расположение ресурсов и политика создания резервных копий. Когда единая программа управления получает запросы потребителей на предоставление облачных услуг, она создает услугу на основе предопределенных для нее показателей.

Программа управления пользовательским доступом предоставляет потребителям пользовательский интерфейс на основе веб-технологий. Потребители могут использовать интерфейс для просмотра перечня облачных услуг и их заказа. Перед тем как переправить запросы пользователей единой программе управления, программа управления пользовательским доступом проводит аутентификацию пользователей. Она также следит за выделением или использованием ресурсов, связанных с экземплярами облачных услуг. На основе

выделения или использования ресурсов программа генерирует отчет о суммах оплаты услуг. Этот отчет могут увидеть потребители, и к тому же он обеспечивает прозрачность отношений между поставщиками и потребителями.

ХРАНИЛИЩЕ ДАННЫХ, ОПТИМИЗИРОВАННОЕ ПОД ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ

Рост количества приложений с большим объемом содержащейся информации в сочетании с ростом объема создаваемых пользователем неструктурированных данных усложнили управление в рамках традиционного подхода к хранению данных. Такое сочетание значительного роста объемов, появления новых видов информации и необходимости обслуживания пользователей по всему миру привело к выработке всеобщих требований к хранению информации и управлению данными. Решением, отвечающим этим требованиям, является хранилище данных, оптимизированное под облачные вычисления. Оно обеспечивает масштабируемую и гибкую архитектуру, гарантирующую быструю адаптацию к потребностям в ее использовании, глобальный доступ к данным и объем хранения данных, выделяемый по фактической потребности. Оно также устраняет ограничения во взаимодействии хранилища данных с потребителем, связанные с жесткой привязкой к точке монтирования, предоставляя единую точку доступа ко всей инфраструктуре хранилища.

В хранилище данных, оптимизированном под облачные вычисления, используются встроенная многопользовательская модель и возможность самообслуживания, а также полноценно замеряемый объем доступа к ресурсам хранилища, что позволяет обеспечивать предоставление хранилища в виде услуги (storage-as-a-service) в совместно используемой архитектуре. В оптимизированном под облачные вычисления хранилище обычно применяется технология хранения данных на основе объектов, в которой для управления политиками размещения в хранилище, защиты и жизненного цикла используются настраиваемые, ориентированные на ценность информации метаданные. Хранилища, оптимизированные под облачные вычисления, обладают следующими основными характеристиками:

- широкой масштабируемостью инфраструктуры, поддерживающей большое количество объектов в глобально распределенной инфраструктуре;
- наличием унифицированного пространства имен, исключающим наряду с другими ограничениями файловой системы ограничения по емкости и размещению;
- наличием возможностей управления, оптимизирующих защиту данных, их доступность и производство вычислений на основе уровней обслуживания, настраиваемых по метаданным и другой информации, задающей ту или иную политику;
- безопасной многопользовательской средой, позволяющей нескольким приложениям получать безопасное обслуживание от одной и той же инфраструктуры. Каждое приложение надежно отделяется от других приложений, и данные не только не смешиваются, но и не становятся доступными другим арендаторам;
- предоставлением доступа через API-интерфейсы веб-служб на основе REST- и SOAP-технологий и доступа на файловой основе с использованием большого разнообразия клиентских устройств.

13.7. Основные проблемы облачных вычислений

Несмотря на то что облачные вычисления получают все большее признание, их поставщикам и потребителям все же приходится сталкиваться с рядом проблем.

13.7.1. Основные проблемы потребителей

Данные, имеющие для ведения бизнеса особо большое значение, требуют защиты и непрерывного отслеживания доступа к ним. Если данные перемещаются в облачную модель, отличную от частной модели, основанной на использовании собственного оборудования, потребители могут утратить абсолютный контроль над своими особо важными данными. Хотя большинство поставщиков облачных услуг предлагают усовершенствованную защиту данных, потребители могут отказаться от передачи облаку контроля за особо важными для своего бизнеса данными.

Поставщики облачных услуг могут для их предоставления использовать несколько дата-центров, расположенных в разных странах. С целью обеспечения высокого уровня доступности и равномерного распределения нагрузки они могут реплицировать или перемещать данные между этими дата-центрами. Потребители могут знать, а могут и не знать, в какой именно стране хранятся их данные. Некоторые поставщики облачных услуг позволяют потребителям выбирать место хранения данных. Требования, касающиеся неприкосновенности частной жизни, и такие нормативные требования, как «EU Data Protection Directive» и «U.S. Safe Harbor program», создают препятствия для потребителей по внедрению облачных вычислений в свою повседневную практику.

Облачные услуги могут быть доступны по сети в любом месте. Но если облачная инфраструктура слишком удалена от точки доступа, задержки сети увеличиваются. Большие сетевые задержки могут либо увеличить время отклика приложения, либо вызвать превышение лимита времени ожидания приложения. Это проблема может быть решена путем реализации обязательных к исполнению соглашений об уровне обслуживания — Service Level Agreements (SLA), заключаемых с поставщиками облачных услуг.

Суть другой проблемы состоит в том, что службами облачной платформы могут не поддерживаться нужные пользователю приложения. Например, поставщик услуг может быть не в состоянии поддерживать узкоспециализированные или фирменные среды, такие как совместимые с приложением операционные системы и предпочтительные языки программирования, требующиеся для разработки и запуска приложений потребителей. Кроме того, на миграцию виртуальных машин внутри облака или между облаками может отрицательно повлиять несогласованность гипервизоров.

Следующая проблема заключается в привязке к поставщику, создающая трудности для потребителей при смене их поставщика облачных

услуг. Отсутствие взаимодействия между API-интерфейсами разных поставщиков облачных услуг также может усложнить миграцию и повысить цены на ее проведение при переходе от одного поставщика облачных услуг к другому.

13.7.2. Основные проблемы поставщиков

Обычно поставщики облачных услуг публикуют соглашение об уровне обслуживания — service-level agreement (SLA), чтобы их потребители были в курсе всех возможностей обслуживания, размеров компенсаций за вынужденное бездействие и нормативно-правовых положений. Как вариант, такие соглашения могут подписывать поставщик и потребитель облачных услуг с учетом специфики клиента. В SLA-соглашениях обычно упоминается сумма штрафа на случай отказа поставщиков услуг от предоставления оговоренного уровня обслуживания. Таким образом, поставщики облачных услуг должны гарантировать наличие у них соответствующих ресурсов для предоставления требуемых уровней обслуживания. Из-за рассредоточенности облачных ресурсов и изменений потребностей в обслуживании поставщикам облачных услуг очень трудно предоставить физические ресурсы при пиковых запросах у всех потребителей и оценить текущую стоимость предоставления услуг.

У многих поставщиков программного обеспечения отсутствует модель лицензирования программ на готовность к работе в облачной среде. Некоторые поставщики программных продуктов предлагают стандартные облачные лицензии по более высокой цене, чем при применении обычных моделей лицензирования. Сложности, возникшие с лицензированием облачного программного обеспечения, вызваны проблемами с развертыванием программ поставщиков в облаке. С такими же сложностями сталкиваются и потребители. Поставщики облачных услуг обычно предлагают для доступа к облаку собственные API-интерфейсы. Но у потребителей может появиться желание воспользоваться API-интерфейсами с открытым кодом или же стандартными API-интерфейсами, чтобы стать арендаторами нескольких облаков. Это создаст проблему для поставщиков облачных услуг, поскольку потребует заключения соглашений между ними.

13.8. Особенности внедрения облачных вычислений

Организации, решившиеся на внедрение облачных вычислений, задаются вопросом: «Как облако впишется в среду организации?». Большинство организаций не готово к тому, чтобы отказаться от уже сделанных вкладов в IT-системы и разом переместить все свои бизнес-процессы в облако. Перед этим им нужно рассмотреть множество различных факторов. Даже индивидуальные предприниматели, присматривающиеся к использованию

облачных услуг, должны обратить внимание на особенности их внедрения в практику. Ряд основных факторов, которые нужно учитывать при решении о внедрении облачных вычислений в повседневную практику, выглядит следующим образом.

- **Выбор модели развертывания.** Основным фактором, определяющим принятие решения по внедрению в свою практику облачных вычислений, является разумное соотношение риска и удобств. Этим фактором также закладывается основа для выбора правильной модели развертывания облака. Индивидуальным предпринимателям и только что открывшимся фирмам обычно предпочтительнее выбирать публичное облако. Для них сниженные цены, предлагаемые при использовании публичного облака, перевешивают риски безопасности или доступности данных, находящихся в облаке. У малых и средних предприятий — small- and medium-sized businesses (SMB) имеется умеренная база клиентов, и на их бизнес могут отрицательно повлиять любые недочеты в базе клиентов и уровне обслуживания. Поэтому их руководителям вряд ли захочется развертывать свое главное приложение, например, проводящее в сети обработку транзакций — Online Transaction Processing (OLTP), в публичном облаке. В таком случае более подходящим вариантом будет модель гибридного облака. Самые важные приложения должны будут запускаться в частном облаке, а менее важные, такие как программы резервного копирования, архивирования и тестирования, могут быть развернуты в публичном облаке. У крупных предприятий обычно имеется база серьезных клиентов по всему миру. На этих предприятиях, как правило, практикуется весьма строгая политика безопасности, позволяющая защитить важные сведения о клиентах. Исходя из своих финансовых возможностей, они могут отдать предпочтение созданию собственных частных облаков.
- **Приемлемость для приложений.** Для развертывания в публичном облаке подходят далеко не все приложения. Причиной может стать несовместимость программного обеспечения облачной платформы с приложениями клиента, или же то, что организация спланировала размещение в облаке устаревшего приложения. Ключевыми и особенно важными являются те приложения, которые находятся в собственности организации и определяют основы ведения бизнеса. Обычно они разрабатываются, развертываются и обслуживаются собственными силами организации. Такие приложения зачастую закладывают основу для конкурентных преимуществ. Из-за высокой степени риска организации вряд ли развернут такие приложения в публичном облаке. Скорее всего, такие приложения станут подходящими кандидатами для развертывания в частном облаке, созданном на собственном оборудовании. А приложения сторонних разработчиков, которые не оказывают влияния на основы ведения бизнеса, вполне подойдут

для развертывания в публичном облаке. Если рабочая нагрузка приложения предполагает наличие интенсивного сетевого трафика, его производительность может быть неоптимальной для развертывания в публичном облаке. Если приложение связано с другими ресурсами или приложениями дата-центра, то при этом также могут появиться проблемы с его производительностью.

- **Финансовые преимущества.** Тщательный анализ финансовых преимуществ дает четкую картину экономии при внедрении облачных вычислений в повседневную практику. При проведении анализа нужно сравнить показатели совокупной стоимости владения — Total Cost of Ownership (TCO) и рентабельности инвестиций — Return on Investment (ROI) при использовании облака и без него, выявив при этом потенциальные финансовые преимущества. При вычислении показателей TCO и ROI организации и индивидуальные предприниматели должны принять во внимание затраты на развертывание и обслуживание собственных инфраструктур в сравнении с затратами на внедрение в свою повседневную практику облачных вычислений. При вычислении затрат на приобретение ресурсов инфраструктуры организации должны включить в них капитальные — capital expenditure (CAPEX) и операционные расходы — operation expenditure (OPEX). В CAPEX включается стоимость серверов, хранилищ данных, операционных систем, приложений, сетевого оборудования, недвижимости и т. д. В OPEX включается стоимость расходов на электроэнергию, охлаждение, персонал, обслуживание, резервное копирование и т. д. Эти расходы нужно сравнить с операционными расходами на внедрение в практику облачных вычислений. В затраты на внедрение включаются стоимость миграции данных в облако, обеспечения совместимости и безопасности, а также внесение платы за использование или абонирование. Перемещение приложений в облако сокращает CAPEX-расходы, если только облако не было создано на собственном оборудовании.
- **Выбор поставщика облачных услуг.** Выбор поставщика особенно актуален для публичного облака. Потребителям нужно узнать, как долго и насколько успешно поставщик предоставляет свои услуги. Нужно также определить, насколько легко у этого поставщика можно будет добавить облачные услуги и прекратить их использование. Потребитель должен узнать, насколько легко можно будет перейти к другому поставщику, если появится такая потребность. Нужно оценить, как поставщик справляется с выполнением требований безопасности, нормативно-правовых положений и сохранности информации частного характера. Нужно также проверить, насколько хорошо поставщик справляется с обслуживанием клиентов.
- **Соглашение о предоставляемых услугах — service-level agreement (SLA).** Обычно поставщики наряду с перечнем облачных услуг

упоминают о показателях качества обслуживания — quality of service (QoS), в частности, пропускной способности и времени безотказной работы. QoS-показатели обычно входят в SLA-соглашение, являющееся договором на предоставление услуг между поставщиком и потребителями. SLA-соглашение служит основой для ожидаемого уровня обслуживания, предоставляемого поставщиком потребителю. Перед тем как внедрять облачные вычисления в свою повседневную практику, потребители должны проверить, соответствуют ли QoS-показатели всем их требованиям.

13.9. Практическая реализация концепций: Vblock

Vblock является предложением полностью интегрированной облачной инфраструктуры, включающей средства вычисления, хранения данных, сеть и технологические средства виртуализации, предоставляемые компаниями EMC, VMware и Cisco, сформировавшими коалицию по поставке решений Vblock.

Vblock позволяет организациям создавать виртуализированные дата-центры и облачные инфраструктуры. Решение Vblock уже имеет предопределенную архитектуру, предварительные настройки, прошло предварительное тестирование и имеет вполне определенные показатели производительности и доступности. В качестве альтернативы приобретению и сборке клиентами отдельных компонентов облачной инфраструктуры Vblock предоставляет уже проверенное решение облачной инфраструктуры, находящееся в состоянии заводской готовности к развертыванию и вводу в эксплуатацию. Это позволит существенно сэкономить средства и время на развертывание.

Унифицированным управляющим решением для Vblock является единый диспетчер EMC Unified Infrastructure Manager (UIM). UIM предоставляет единый центр управления Vblock-инфраструктурами и способен управлять сразу несколькими их экземплярами. С использованием UIM-диспетчера услуги облачной инфраструктуры могут предоставляться в автоматическом режиме с учетом всего лучшего, что было создано в практике такого предоставления.

Дополнительные сведения о Vblock можно найти на сайте www.emc.com.

Резюме

Несмотря на то что облачные вычисления находятся в фазе развития, они набирают все большую популярность, поскольку потребители видят в них потенциальную возможность экономии средств, а поставщики — возможности предоставления новых услуг. Облачные вычисления позволили организациям и индивидуальным предпринимателям воспользоваться такими преимуществами, как автоматизированное и быстрое предоставление

ресурсов, гибкость, высокая доступность и меньшее время выхода на рынок при сокращении общих затрат на приобретение оборудования. Несмотря на имеющиеся опасения и проблемы, преимущества облачных вычислений являются довольно убедительными для их внедрения в повседневную практику.

Для организаций, имеющих в собственности обычные дата-центры, внедрение облачных вычислений сродни некому путешествию, начинающемуся с объединения вычислительных ресурсов, включающих вычислительные системы, устройства хранения данных и сети с помощью технологий виртуализации. За виртуализацией ресурсов организациям нужно сделать следующий шаг по реализации унифицированных средств управления облачной инфраструктурой и составлению перечня услуг. Ключевым фактором приведения предоставляемых облачных услуг к уровню ожиданий, возлагаемых на них предприятиями и потребителями, является реализация надлежащих процессов управления услугами.

В данной главе были подробно рассмотрены характеристики облачных вычислений, их преимущества, предоставляемые услуги, модели развертывания и инфраструктура. В ней также были рассмотрены проблемы облачных вычислений и факторы, определяющие порядок их внедрения в повседневную практику. В следующей главе основное внимание будет уделено безопасности, обеспечиваемой инфраструктурой хранения данных, куда также будут включены вопросы о факторах, определяющих основы безопасности хранилищ в виртуализированных и облачных средах.

УПРАЖНЕНИЯ

1. Назовите наиболее важные характеристики облачных вычислений.
2. Какой вклад облачные вычисления привносят в адаптивность бизнеса?
3. Проведите исследование сервис-ориентированной архитектуры и ее применения в облачных вычислениях.
4. Проведите исследование скоординированности облачных вычислений.
5. Проведите исследование различных факторов, определяющих выбор поставщика публичных облачных услуг.
6. Что нужно брать в расчет при финансовой оценке преимуществ облачных вычислений?

Раздел

V

Обеспечение безопасности и управление инфраструктурой хранения данных

В ЭТОМ РАЗДЕЛЕ

Глава 14. Обеспечение безопасности инфраструктуры хранения данных
Глава 15. Управление инфраструктурой хранения данных

Глава 14
Обеспечение безопасности инфраструктуры хранения данных

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Концепция безопасности хранения данных
- Триада рисков
- Отказ в обслуживании
- Домены безопасности
- Управление правами на информацию
- Контроль доступа

В массивах хранения данных, доступных по сети, постоянно ведутся обработка и хранение ценной информации, включая интеллектуальную собственность, личные данные и сведения о финансовых операциях. Поэтому в настоящее время хранилища данных наиболее подвержены различным угрозам безопасности, потенциально способным повредить важные для ведения бизнеса данные и воспрепятствовать предоставлению основных услуг. Защита инфраструктуры хранилища данных стала неотъемлемой частью процесса управления хранилищем данных как в обычных, так и в виртуализированных дата-центрах. Интенсивная работа над ее совершенствованием стала постоянной составляющей решения задач по управлению жизненно важной информацией и ее защите.

Безопасность хранилища данных в среде публичного облака обеспечивается намного сложнее, поскольку контроль над общей IT-инфраструктурой и мерами безопасности со стороны организаций весьма ограничен. Более того, многопользовательский режим предоставления услуг в облачной среде допускает совместное использование ресурсов, включая хранилище данных, несколькими потребителями. При этом создается угроза перемешивания данных, принадлежащих разным арендаторам.

В данной главе рассматривается концепция обеспечения информационной безопасности, призванная снизить уровень потенциальных угроз и вести борьбу с вредоносными атаками на инфраструктуру хранения данных. Кроме того, в главе дается описание основных вопросов реализации мер безопасности хранилищ данных, включая архитектуру системы безопасности и механизмы защиты в сетях FC-SAN и IP-SAN, а также в NAS-устройствах. Далее

в главе дается описание дополнительных факторов, определяющих порядок обеспечения безопасности в виртуализированных и облачных средах.

14.1. Концепция информационной безопасности

Базовая концепция информационной безопасности разрабатывается для решения четырех задач: обеспечения конфиденциальности, целостности и доступности данных — confidentiality, integrity, availability (CIA), а также обеспечения возможности ведения учета всех операций, проводимых с данными. В этой концепции объединяются все стандарты, процедуры и средства управления безопасностью, необходимые для снижения уровня угроз в среде инфраструктуры хранения данных.

- **Конфиденциальность.** Предоставляет требуемую скрытность информации и обеспечивает возможность доступа к данным только авторизованным пользователям. Для этого требуется, чтобы пользователи, нуждающиеся в получении информации, проходили аутентификацию. Данные, находящиеся в процессе перемещения (передаваемые по кабелям), и данные, находящиеся в состоянии покоя (размещенные в основном хранилище, на носителях резервных копий или в архивах), для обеспечения конфиденциальности могут быть зашифрованы. Кроме отстранения от доступа к информации неавторизованных пользователей конфиденциальность требует в качестве составной части протокола безопасности принятия мер по защите трафика. Эти меры обычно включают сокрытие адресов источника и приемника, частоты отправки тех или иных данных и объема отправленных данных.
- **Целостность.** Гарантирует отсутствие каких-либо изменений информации. Обеспечение целостности требует выявления попыток несанкционированного изменения или удаления информации и защиты от них. Обеспечение целостности предусматривает также такие меры, как обнаружение ошибок и внесение исправлений как в данные, так и в системы.
- **Доступность.** Гарантирует надежный и своевременный доступ к системам, данным и приложениям, находящимся на этих системах для авторизованных пользователей. Доступность требует защиты от несанкционированного удаления данных и отказа от обслуживания (эти вопросы рассматриваются в разделе «14.2.2. Угрозы»). Доступность также подразумевает, что для предоставления услуг имеются достаточные объемы ресурсов.
- **Предоставление возможности ведения учета.** Эта цель имеет отношение к учету всех событий и операций, имевших место в инфраструктуре дата-центра. Службой учета ведется журнал событий, который может быть впоследствии проверен или просмотрен в целях обеспечения безопасности.

14.2. Триада рисков

Триада рисков определяет риск с точки зрения угроз, активов и уязвимостей. Риск возникает, когда агент угрозы (взломщик) пользуется имеющейся уязвимостью для взлома служб безопасности. Например, если документ особой важности передается без какой-либо защиты по небезопасному каналу, взломщик может получить неавторизованный доступ к документу и нарушить его конфиденциальность или целостность. Это, в свою очередь, может отрицательно сказаться на бизнесе организации. При развитии данного сценария возникает потенциальный риск бизнес-потерь, поскольку взломщик пользуется уязвимостью незащищенных каналов связи для доступа к документу и проведения с ним различных манипуляций.

Для управления рисками организации в первую очередь сосредоточивают внимание на уязвимостях, поскольку они не в состоянии избавиться от агентов угроз, появляющихся в разнообразных формах и источниках и угрожающих их активам. Организации могут усилить контрмеры, стремясь тем самым сократить вероятность осуществления атак и серьезность их последствий.

Первым шагом на пути определения степени потенциальных угроз и рисков в IT-инфраструктуре является оценка рисков. В ходе этого процесса определяется степень рисков и создаются предпосылки для определения соответствующих средств управления, позволяющих снизить или вовсе устранить риски. Основываясь на ценности активов, оценка рисков помогает наметить приоритетные направления приложения сил и подготовки мер безопасности. Чтобы определить вероятность возникновения неблагоприятных событий, угрозы в отношении IT-систем должны быть проанализированы на фоне потенциальных уязвимостей и существующей системы управления безопасностью.

Опасность неблагоприятных событий оценивается по степени их влияния на важные бизнес-операции. На основе этого анализа IT-активам и ресурсам должны присваиваться относительные показатели их важности и уязвимости. Например, если взлом конкретного компонента IT-систем может стать причиной полного прекращения предоставления жизненно важных услуг, ему может быть присвоено значение высокой важности.

В следующих разделах будет представлено подробное исследование всех ключевых элементов триады рисков: активов, угроз и уязвимостей — с точки зрения выявления рисков и анализа возможностей управления ситуацией.

14.2.1. Активы

Одним из наиболее важных активов любой организации является информация. К прочим активам можно отнести оборудование, программное обеспечение и другие компоненты инфраструктуры, необходимые для доступа к информации. Для защиты этих активов организации должны разработать

набор параметров, позволяющих гарантировать доступность ресурсов для авторизованных пользователей и высоконадежных сетей. Эти параметры применяются к ресурсам хранения данных, сетевой инфраструктуре и политикам организации.

Методы обеспечения безопасности преследуют две цели. Первая — обеспечить беспрепятственный доступ к сети авторизованным пользователям. Ее достижение также предполагает надежную и стабильную работу в любых условиях окружающей обстановки при любой степени загруженности оборудования. Вторая цель — максимально затруднить потенциальным злоумышленникам доступ к системе и снизить до минимума вероятность ее взлома.

Методы обеспечения безопасности должны предлагать принятие адекватных мер против неавторизованного доступа, проникновения вирусов, червей, троянов и других вредоносных программ. Меры безопасности должны также включать возможность шифрования важных данных и отключения неиспользуемых служб с целью сведения к минимуму количества потенциальных уязвимостей в системе безопасности. Методы обеспечения безопасности должны гарантировать регулярную установку обновлений операционной системы и других программных средств. В то же время они должны предоставлять соответствующую избыточность в виде реплик и зеркальных копий производственных данных для предотвращения катастрофических потерь данных в случае неожиданного взлома системы. Чтобы работа системы безопасности шла без сбоев, все пользователи должны быть уведомлены о политике, регулирующей порядок использования сети.

Эффективность методов обеспечения безопасности хранилищ данных может быть оценена по двум ключевым критериям. Согласно первому затраты на реализацию системы должны составлять лишь часть от оценочной стоимости защищаемых данных. Согласно второму возможный взлом этой системы должен весьма дорого обходиться потенциальному взломщику как по средствам, так и по прилагаемым усилиям и затраченному времени.

14.2.2. Угрозы

Под *угрозами* понимаются потенциально возможные атаки, которые могут предприниматься в отношении IT-инфраструктуры. Эти атаки можно разделить на активные и пассивные. *Пассивные атаки* предпринимаются для получения неавторизованного доступа внутри системы. Они угрожают конфиденциальности информации. *Активные атаки* включают в себя атаки, нацеленные на изменение данных, отказ в обслуживании (DoS) и отрицание или отсутствие фиксации действий (repudiation). Они угрожают целостности данных, их доступности и возможности ведения учета событий и операций.

При атаках, направленных на изменение данных, неавторизованные пользователи пытаются модифицировать информацию, стараясь причинить вред

организации. Такие атаки могут быть нацелены как на хранящиеся, так и на передаваемые данные, и угрожают целостности этих данных.

Атаки, вызывающие *отказ в обслуживании* — denial of service (DoS), не дают легальным пользователям получить доступ к ресурсам и услугам. Как правило, такие атаки не касаются доступа к информации и не нацелены на ее изменение. Они угрожают доступности данных. Одним из примеров DoS-атак может послужить преднамеренное лавинное забрасывание сети или веб-сайта запросами с целью воспрепятствования законному доступу авторизованных пользователей к данным.

Атака, нацеленная на *отрицание или отсутствие фиксации действий*, ведется, чтобы воспрепятствовать возможности учета событий и операций. При ее проведении предпринимается попытка предоставления ложной информации либо путем выдачи себя за другое лицо, либо путем отрицания того, что событие или транзакция действительно имели место. Например, атака отрицания или отсутствия фиксации действий может касаться выполнения какого-либо действия и уничтожения любых следов, которые могли бы подтвердить идентичность пользователя (взломщика), предпринявшего это действие. Атаки отрицания или отсутствия фиксации действий заключаются в попытках обхода ведения журнала событий, влияющих на безопасность, или в попытках проведения с этим журналом различных манипуляций с целью сокрытия личности злоумышленника.

ПРИМЕРЫ ПАССИВНЫХ АТАК

- Подслушивание. Когда кто-нибудь подслушивает разговор, то такой несанкционированный доступ к информации называется подслушиванием.
- Перехват данных. Под ним подразумевается получение доступа к новым пользовательским данным несанкционированным путем. В общем смысле перехват и подслушивание являются синонимами.

Злоумышленники зачастую используют такие технологии и средства, как программы перехвата действий пользователя (key loggers), позволяющие отслеживать нажатия клавиш и добывать информацию о паролях и регистрационных именах (логинах), или программы перехвата сообщений электронной почты и другой закрытой информации и пересылаемых данных. Иногда в организациях проводится вполне легальный перехват информации своих работников с целью отследить род их занятий на служебных компьютерах и цели использования ими Интернета.

14.2.3. Уязвимость

Зачастую уязвимостью для потенциальных атак страдают пути, по которым предоставляется информация. На каждом из путей могут быть разнообразные точки доступа, предоставляющие различные уровни доступа к ресурсам

хранилища данных. В точках доступа на пути передачи данных важно реализовать соответствующие средства контроля безопасности. Реализация средств контроля безопасности в каждой точке доступа каждого пути передачи данных известна как *глубоко эшелонированная оборона*.

При реализации такой обороны рекомендуется использовать сразу несколько мер для снижения риска угроз безопасности в случае взлома одного из компонентов защиты. Такой прием известен также как многоуровневый подход к организации безопасности. Благодаря принятию нескольких мер безопасности на различных уровнях глубоко эшелонированная оборона дает дополнительное время на обнаружение атаки и подготовку ответных мер. При прорыве системы безопасности это может сузить область или степень отрицательного воздействия.

При оценке степени уязвимости среды от угроз безопасности следует учитывать три фактора: *поверхность атаки*, *вектор атаки* и *фактор трудозатрат*. *Поверхность атаки* относится к различным точкам входа, которыми взломщик может воспользоваться для перехода в атаку. Источником потенциальной уязвимости может послужить каждый компонент сети хранения данных. Чтобы предпринять какую-либо атаку, взломщик может воспользоваться всеми внешними интерфейсами, поддерживаемыми тем или иным компонентом, например аппаратным и управляющим интерфейсом. Из этих интерфейсов и формируется поверхность атаки для взломщика. Стать частью поверхности атаки в случае пребывания в подключенном состоянии могут даже неиспользуемые службы.

Под *вектором атаки* понимается действие или серия действий, необходимых для успешного завершения атаки. Например, взломщик может воспользоваться уязвимостью в интерфейсе управления и развить атаку перехвата, посредством которой может изменить конфигурацию устройства хранения данных, открыв доступ к трафику с одного и более хостов. Этот перенаправленный трафик может использоваться для перехвата передаваемых данных.

Под *фактором трудозатрат* обычно понимается тот объем времени и прикладываемых усилий, которые должны быть затрачены на то, чтобы можно было воспользоваться вектором атаки. Например, если взломщики пытаются извлечь важную информацию, они берут в расчет время и усилия, затрачиваемые на то, чтобы предпринять атаку на базу данных. В этот расчет могут включаться определение привилегированных учетных записей, определение схемы базы данных и написание SQL-запросов. Вместо этого на основе оценки фактора трудозатрат взломщики могут рассматривать менее затратные способы, позволяющие воспользоваться массивом хранения данных путем атаки, непосредственно направленной на этот массив и позволяющей осуществлять чтение обычных дисковых блоков.

Оценив степень уязвимости среды, организаторы могут развернуть специализированные средства контроля. Любые контрольные меры должны

касаться всех трех аспектов инфраструктуры: людей, процессов и технологий, а также связей между ними. Для того чтобы обезопасить людей, в первую очередь нужно организовать и обеспечить их идентичность. На основе их идентичности может быть реализован избирательный контроль над их доступом к данным и ресурсам. Эффективность любых мер безопасности определяется главным образом процессами и проводимыми политиками. Процессы должны основываться на глубоком понимании имеющихся в среде рисков и устанавливать относительную степень конфиденциальности различных типов данных и потребности в доступе к этим данным различных заинтересованных лиц. Без эффективных процессов развертывание технологий не будет обладать ни нужной рентабельностью, ни соответствием приоритетам организаций. В конечном итоге, для того чтобы развернутые технологии или средства контроля работали эффективно, они должны быть согласованы с процессами, применяемыми политиками и интересами людей. Технологии обеспечения безопасности направлены на уменьшение уязвимости путем сведения к минимуму поверхностей атак и максимально возможного увеличения трудозатрат для взломщиков. Эти средства контроля могут быть как техническими, так и не имеющими к технике никакого отношения. Технические средства контроля обычно реализуются посредством использования компьютерных систем, а нетехнические средства — посредством административного и физического контроля. Административные меры включают политику безопасности и кадровые или стандартные процедуры, направленные на безопасное выполнение различных операций. Физические средства контроля включают создание различных физических преград в виде действий подразделений охраны, установки заборов или дверных замков.

На основе своей роли средства контроля делятся на профилактические, выявляющие и корректирующие. Профилактические средства контроля призваны помешать атаке, выявляющие средства контроля предназначены для обнаружения ведущейся атаки. А когда атака уже обнаружена, вступают в силу корректирующие средства контроля. *Профилактические средства контроля* предотвращают использование уязвимостей и препятствуют атаке или же снижают степень ее отрицательного воздействия. *Корректирующие средства контроля* снижают эффективность атаки, а *выявляющие средства контроля* обнаруживают атаки и переключают усилия с профилактических мер на корректирующие. Например, система обнаружения внедрения в инфраструктуру и его предотвращения — Intrusion Detection/Intrusion Prevention System (IDS/IPS) является выявляющим средством контроля, определяющим, ведется ли атака, после чего этим средством предпринимается попытка остановить эту атаку путем разрыва сетевого подключения или вызова одного из средств контроля, принадлежащих межсетевому экрану, для блокировки трафика.

14.3. Домены безопасности хранилища данных

Устройства хранения данных, подключенные к сети, повышают степень риска, и, кроме того, они более открыты угрозам безопасности со стороны сетей. К тому же с повышением степени использования сетевых технологий в среде хранения данных устройства хранения стали слишком уязвимы для угроз безопасности, исходящих от различных источников. Для защиты среды сетевого хранения данных должны быть реализованы специальные средства контроля. Это требует более пристального внимания к обеспечению безопасности сетевого хранения данных и четкого знания путей доступа, ведущих к ресурсам хранилищ. Если какой-то конкретный путь не прошел авторизацию и должен быть запрещен техническими средствами контроля, нужно убедиться в том, что эти средства не были взломаны. Если каждый компонент в сети хранения данных рассматривается в качестве потенциальной точки доступа, область атаки всех этих точек доступа должна быть проанализирована с целью выявления связанных с ними уязвимостей.

Для обнаружения угроз, касающихся сети хранения данных, пути доступа к хранилищу должны быть отнесены к трем различным доменам безопасности: *доступа со стороны приложений*, *доступа для управления* и *доступа для создания резервных копий, реплик и архивов*. Эти три домена безопасности системы хранения данных показаны на рис. 14.1.

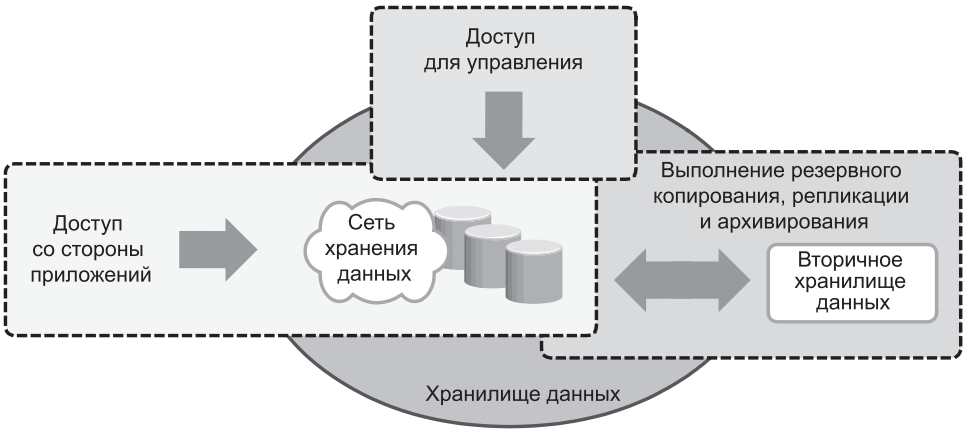


Рис. 14.1. Домены безопасности хранения данных

Первый домен безопасности относится к доступу со стороны приложений, осуществляемому с целью хранения данных в сети хранения. Второй домен безопасности относится к доступу для управления к устройствам хранения,

устройствам системы организации связи и к данным, размещенным на этих устройствах. К этому домену в основном обращаются администраторы хранилищ данных, занимающиеся настройкой среды и управлением этой средой. Третий домен относится к процессам обращения к данным при выполнении резервного копирования, репликации данных и создании архивов. Наряду с точками доступа в этом домене нужно обезопасить и носители резервных копий.

Для обеспечения безопасности сети хранения данных нужно определить существующие угрозы внутри каждого домена безопасности и классифицировать угрозы на основе каждого аспекта безопасности — обеспечения доступности, конфиденциальности, целостности данных и предоставления возможности ведения учета.

Следующими шагами на пути обеспечения безопасности станут выбор и реализация различных средств контроля в качестве мер, препятствующих возникновению угроз.

14.3.1. Обеспечение безопасности домена доступа со стороны приложений

Домен доступа со стороны приложений может охватывать только те приложения, которые обращаются к данным через файловую систему или через интерфейс базы данных.

Важным шагом на пути обеспечения безопасности домена доступа со стороны приложений является определение возможных угроз в среде хранения данных и соответствующих противопоставляемых им мер контроля. Реализация физических мер безопасности также является важным вопросом, требующим рассмотрения в плане предотвращения хищений носителей данных.

Доступ со стороны приложений в среде сетевого хранения данных показан на рис. 14.2. Хост *А* может обращаться ко всем томам *Т1*, хост *Б* может обращаться ко всем томам *Т2*. Эти тома классифицируются в соответствии с уровнем доступа, например по принадлежности к конфиденциальному, ограниченному или открытому доступу. Среди возможных угроз в данном сценарии может быть подмена идентификатора хоста *А* или повышение уровня привилегий хоста *Б* для получения доступа к ресурсам хоста *Б*. Еще одной угрозой может быть получение доступа к сети со стороны хоста, не прошедшего авторизацию: взломщик, работающий с этого хоста, может попытаться подменить идентификатор другого хоста и повредить данные, перехватить сеть или предпринять DoS-атаку. К взлому системы безопасности может привести также хищение носителей данных. Все эти угрозы могут создать ряд серьезных проблем в обеспечении безопасности сети, поэтому они требуют пристального рассмотрения.

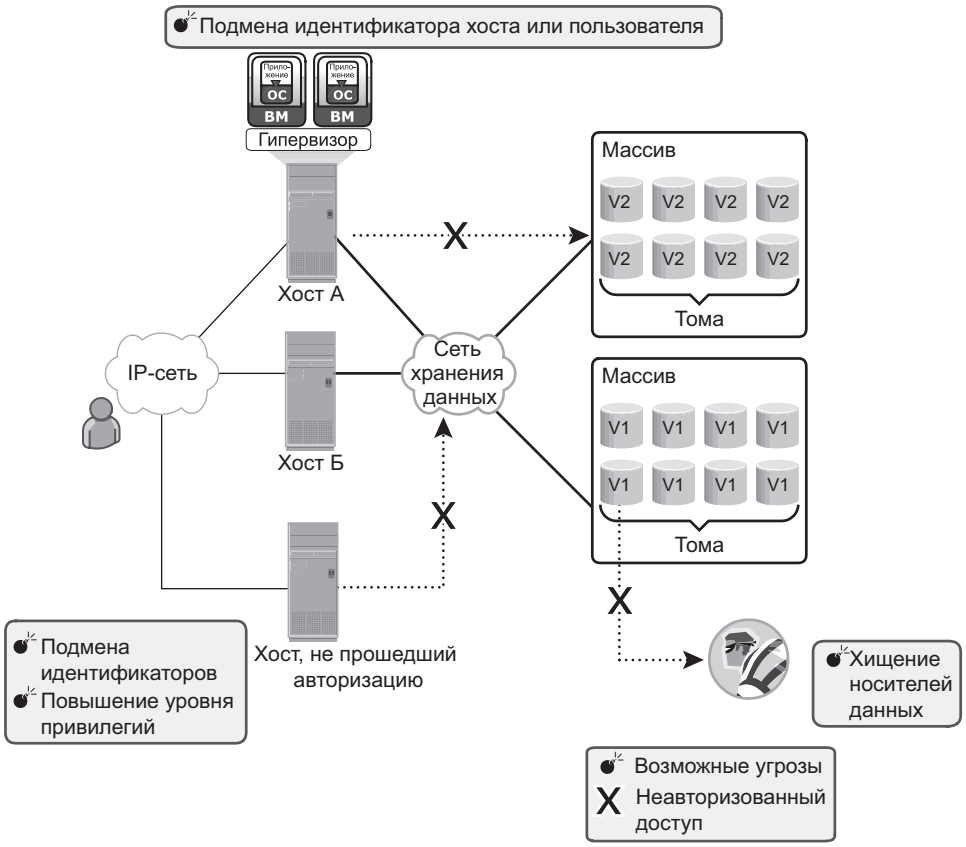


Рис. 14.2. Угрозы обеспечению безопасности в домене доступа к данным со стороны приложения

Управление доступом к данным со стороны пользователей

Доступ пользователей к данным регламентируется службами контроля доступа. Эти службы снижают степень угроз подмены идентификаторов хоста и повышения уровня его привилегий. Обе эти угрозы могут повлиять на целостность данных и их конфиденциальность.

Механизмы контроля доступа, используемые в домене доступа со стороны приложений, заключаются в аутентификации пользователя и хоста (техническом контроле) и авторизации (административном контроле). Эти механизмы могут выходить за границы сети хранения данных и требуют применения различных систем для взаимодействия с другими системами управления идентификацией, имеющимися на предприятии, и системами аутентификации, например системами, обеспечивающими строгую аутентификацию

и авторизацию для защиты пользовательских идентификаторов от подмены. NAS-устройства поддерживают создание *списков контроля доступа* (access control lists), регулирующих пользовательский доступ к конкретно указанным файлам. Приложение управления контентом предприятия — Enterprise Content Management осуществляет доступ к данным путем использования системы управления правами на информацию — Information Rights Management (IRM), которая определяет, какие пользователи имеют права на тот или иной документ. Ограничение доступа на уровне хоста начинается с аутентификации узла в ходе его попытки подключиться к сети.

В различных технологиях сетевого хранения данных, например в хранилищах, построенных на основе iSCSI, FC и IP, для аутентификации доступа со стороны хоста используются различные механизмы — Challenge-Handshake Authentication Protocol (CHAP), Fibre Channel Security Protocol (FC-SP) и IPSec соответственно.

После того как хост пройдет аутентификацию, следующим шагом будет определение механизмов контроля безопасности для таких ресурсов хранения данных, как порты, тома или пулы хранилищ, для которых хост авторизовался с целью получения доступа. Механизмом контроля на коммутаторах является *зонирование*, сегментирующее сеть на определенных маршрутах, которые будут использоваться для трафика данных; *маскирование LUN-устройств*, определяющее, какие хосты к каким устройствам хранения данных могут получить доступ. Некоторые устройства поддерживают отображения глобального имени хоста (WWN) на конкретный FC-порт, а от него и на конкретное LUN-устройство. Эта привязка WWN-имени к физическому порту является наиболее безопасным решением.

И наконец, нужно обязательно обеспечить реализацию таких средств административного контроля, как определение политик и стандартов безопасности. Для обеспечения правильной работы средств административного контроля требуется их регулярный аудит. Его проведение возможно путем регистрации в журнале всех важных событий на всех участвующих в работе устройствах. Записи в журнале событий также должны быть защищены от неавторизованного доступа, поскольку если содержимое журнала открыто для взломщиков и позволяет им вносить в него изменения, то журналы уже не могут выполнить отводимую им роль.

Защита инфраструктуры хранения данных

К защите инфраструктуры хранения данных от неавторизованного доступа относится защита всех элементов этой инфраструктуры. Средства контроля безопасности для защиты инфраструктуры хранения данных реагируют на угрозы несанкционированной фальсификации данных в процессе их передачи, приводящей к потере их целостности, отказу от обслуживания, подрыву доступности данных и сетевому перехвату данных, который может привести к потере их конфиденциальности.

Средства безопасности, предназначенные для защиты сети, можно разбить на две основные категории: *средства поддержания целостности сетевой*

инфраструктуры и *средства шифрования данных в сети хранения*. К средствам обеспечения целостности сетевой инфраструктуры относится функция системы коммутации, гарантирующая целостность этой системы. Эта гарантия достигается за счет предотвращения добавления хоста к системе коммутации SAN-сети без надлежащей авторизации. К методам шифрования данных сети хранения относятся использование IPSec для защиты сетей хранения данных, построенной на основе IP-сети, и FC-SP для защиты FC-сетей.

В безопасной среде хранения данных права администратора, или так называемые root-права, на конкретное устройство каждому пользователю не предоставляются, а для присваивания пользователям нужных привилегий развертывается *ролевой контроль доступа* — role-based access control (RBAC), позволяющий им выполнять свои роли. Роль может представлять собой какую-либо рабочую функцию, например администратора. Привилегии связаны с ролями, и пользователи получают эти привилегии на основе исполняемых ими ролей.

При определении процедур, проводимых в дата-центре, также целесообразно обратить внимание на административные средства контроля, например на разделение обязанностей. Четкое разделение обязанностей гарантирует, что ни один человек не может одновременно определять возможность того или иного действия и выполнять это действие. Например, человек, позволяющий создавать учетные записи администратора, не должен быть в категории лиц, пользующихся такой учетной записью. Обеспечение безопасности при доступе к данным для управления подробно рассматривается в следующем разделе.

Сети управления для систем хранения данных должны быть логически отделены от других сетей предприятия. Эта сегментация очень важна для упрощения задач управления и повышения уровня безопасности за счет предоставления доступа только тем компонентам системы, которые находятся в том же сегменте. Например, сегментация IP-сети обеспечивается за счет развертывания на третьем уровне (Layer 3) фильтров с использованием маршрутизаторов и межсетевых экранов и за счет использования на втором уровне (Layer 2) VLAN-сетей и системы безопасности на уровне портов в Ethernet-коммутаторах.

И наконец, необходимо контролировать физический доступ к консоли устройства и кабелям Fibre Channel коммутаторов, гарантируя тем самым защиту инфраструктуры хранилища данных. Если не прошедший авторизацию пользователь получит физический доступ к устройству, то все остальные устанавливаемые меры безопасности станут бесполезными, поскольку это позволяет покушаться на достоверность устройства.

Шифрование данных

Наиболее важным аспектом обеспечения безопасности данных является защита тех данных, которые хранятся внутри массивов. Угрозы на этом уровне включают фальсификацию данных, нарушающую их целостность,

и хищение носителей данных, подвергающее риску доступность данных и их конфиденциальность. Для защиты от этих угроз нужно шифровать данные, находящиеся на носителях хранилища, или шифровать данные до их переноса на диск. Важно также решить, каким методом следует воспользоваться, чтобы удаляемые за ненадобностью данные были полностью стерты с дисков и не могли быть восстановлены для реализации преступных замыслов.

Данные должны шифроваться как можно ближе к месту своего происхождения. Если нет возможности зашифровать их на хост-машине, средства шифрования должны быть использованы на точке входа в сеть хранения данных. Устройства шифрования должны быть реализованы в системе коммутации, которая шифрует данные между хостом и носителем данных хранилища. Эти механизмы могут защитить как те данные, которые находятся в покое на устройстве-приемнике, так и те данные, которые находятся в пути.

На NAS-устройствах улучшить обеспечение целостности данных можно за счет антивирусной проверки и управления расширениями файлов. В случае использования контентно-адресуемых хранилищ (CAS) гарантировать целостность данных за счет обнаружения любых изменений в битовых комбинациях контента может использование криптографических алгоритмов MD5 или SHA-256. Кроме того, служба уничтожения данных обеспечивает их полную перезапись битовой последовательностью еще до того, как диск будет выведен из эксплуатации. Необходимость затирания данных диска перед его выводом из эксплуатации и степень серьезности данной операции, согласно нормативным требованиям, определяются политикой классификации данных организации.

14.3.2. Обеспечение безопасности домена доступа для управления

Доступ к данным для управления, будь то мониторинг, подготовка к работе или управление ресурсами хранилища, связан с каждым устройством, имеющимся в сети хранения данных. Большинством программных систем управления поддерживаются те или иные формы интерфейса командной строки (CLI) системы консоли управления или интерфейсы на основе использования веб-технологий. Реализация надлежащих средств контроля обеспечения безопасности приложений управления хранилищами данных играет весьма важную роль, поскольку степень ущерба, который может быть нанесен при использовании данных приложений, может быть очень значительным.

На рис. 14.3 изображена среда сетевого хранения данных, в которой производственные хосты подключены к системе SAN-коммутации и обращаются к производственному массиву хранения данных *А*, подключенному с целью проведения репликации к удаленному массиву хранения данных *Б*. В этой конфигурации имеется также платформа управления хранилищами,

находящаяся на хосте *А*. Возможной угрозой в данной среде может быть подмена на хосте, не прошедшем авторизацию, идентификаторов пользователя или хоста с целью управления массивами хранения данных или сетью. Например, хост, не прошедший авторизацию, может получить доступ к управлению удаленным массивом *Б*.

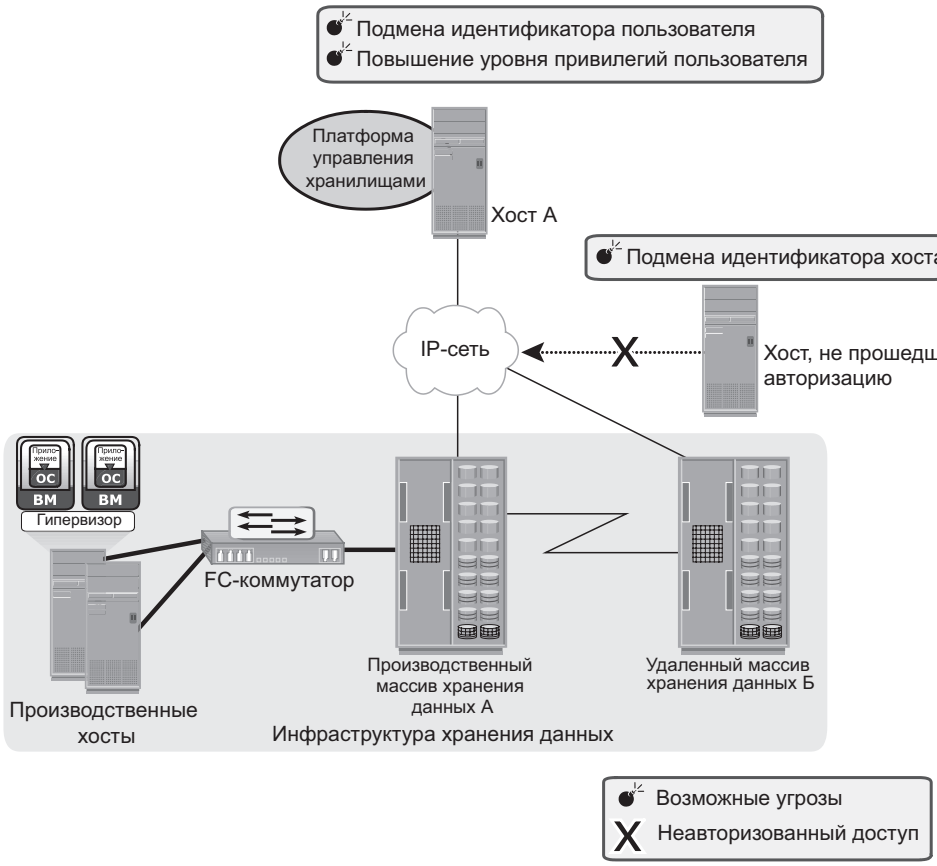


Рис. 14.3. Угрозы обеспечению безопасности в домене доступа к данным со стороны администрации

Предоставление доступа к данным для управления через внешнюю сеть повышает потенциальную возможность подключения к этой сети через не прошедшие авторизацию хост или коммутатор. В данных обстоятельствах реализация соответствующих мер безопасности не допускает осуществления конкретных типов удаленного подключения. Эффективная защита от этих угроз обеспечивается за счет использования безопасных каналов связи, таких как Secure Shell (SSH) или Secure Sockets Layer (SSL)/Transport Layer Security

(TLS). Обнаружить неавторизованный доступ и несанкционированное внесение изменений в инфраструктуру помогает мониторинг журнала событий. Журналы событий должны размещаться за пределами систем хранения данных общего пользования, откуда они могут быть извлечены в случае взлома хранилища.

Следует проверить платформу управления хранилищами на наличие доступных средств контроля безопасности и обеспечить соответствие этих средств контроля требованиям соблюдения мер безопасности всей среды хранения данных. Идентичность администратора и его роль должны быть защищены от любых попыток получения доступа обманным путем, то есть так называемого спуфинга, чтобы взломщик не смог манипулировать всем массивом хранения данных и вызвать недопустимые потери данных путем переформатирования имеющихся в хранилище носителей данных или пресечения доступа к ресурсам данных.

Управление доступом, осуществляемым с правами администратора

Управление доступом к хранилищу, осуществляемое с правами администратора, направлено на защиту от угроз, связанных с попытками подмены взломщиками идентификаторов, принадлежащих администраторам, или повышения уровня привилегий с целью получения доступа с правами администратора. Обе эти угрозы нацелены на нарушение целостности данных и устройств. Для защиты от этих угроз используются нормы предоставления административного доступа и различные технологии аудита, направленные на повышение подотчетности пользователей и процессов. Контроль доступа должен быть обеспечен в отношении каждого компонента системы хранения данных.

В некоторых средах хранения данных может появиться необходимость объединения устройств хранения данных в каталоги аутентификации, создаваемые с помощью средств сторонних разработчиков, таких как облегченный протокол службы каталогов — Lightweight Directory Access Protocol (LDAP) или Active Directory.

Передовой опыт управления безопасностью подсказывает, что никакой отдельно взятый пользователь не должен иметь полный контроль над всеми аспектами системы. Если есть необходимость наделения пользователя правами администратора, то перечень его действий, требующих прав администратора, должен быть сведен к минимуму. А еще лучше наделять различными административными функциями путем использования RBAC, то есть управления доступом на основе ролей — Role Based Access Control. Весьма важной мерой контроля, предпринимаемой для отслеживания действий администратора, является аудит событий, зарегистрированных в соответствующем журнале. Но доступ к файлам журнала, касающегося действий администрации, и к их содержимому должен быть защищен. Еще

одним важным требованием, обеспечивающим согласованное отслеживание действий между системами, является развертывание на каждой системе надежного сетевого протокола времени — Network Time Protocol, который должен быть синхронизирован с единым временем. Кроме того, проведение эффективного анализа файлов журнала событий поддерживается наличием такого средства, как программа управления информационной безопасностью — Security Information Management (SIM).

Защита инфраструктуры управления

Механизмы защиты сетевой инфраструктуры управления включают шифрование трафика управления, введение контроля доступа с правами администратора и применение передового опыта обеспечения безопасности IP-сетей. Этот передовой опыт подсказывает, что для ограничения трафика областью конкретных устройств нужно применять IP-маршрутизаторы и Ethernet-коммутаторы. Ограничение сетевых действий и доступ к ограниченному набору хостов позволяют свести к минимуму угрозу взлома со стороны подключенного к сети устройства, не прошедшего авторизацию, с целью получения доступа к интерфейсам управления. Средства управления доступом должны применяться на уровне массивов хранения данных, с тем чтобы определить, какой из хостов к какому из массивов имеет доступ с правами администратора. Некоторые устройства хранения данных и коммутаторы могут ограничивать доступ с правами администратора конкретным хостам и ограничивать перечень команд, которые могут быть выданы с каждого хоста.

Для трафика управления настоятельно рекомендуется иметь отдельную закрытую сеть управления. По возможности трафик управления не должен смешиваться ни с трафиком производственных данных, ни с каким-либо другим трафиком LAN-сети, используемым на предприятии. Неиспользуемые сетевые службы должны быть отключены на всех устройствах сети хранения данных. Это сократит для устройства поверхность атак, поскольку количество интерфейсов, через которые может быть осуществлен доступ к устройству, будет сведено к минимуму. Подводя итоги, следует заметить, что усилия по обеспечению безопасности должны быть сосредоточены на каналах связи управления между устройствами, обеспечении конфиденциальности и целостности данных системы управления и на доступности сетей управления и устройств.

14.3.3. Обеспечение безопасности инфраструктур резервного копирования, репликации и архивирования

Резервное копирование, репликация и архивирование представляют тот самый третий домен, который нуждается в защите от взлома. Как говорилось в главе 10, резервное копирование связано с копированием данных из

массива хранения данных на носители резервных копий, например на ленты или диски. Безопасное резервное копирование представляет собой непростую задачу и основано на применении программ резервного копирования, обращающихся к массивам хранения данных. Безопасность в данном вопросе также зависит от конфигурации среды хранения данных на основной и второстепенной площадке, особенно если решения, связанные с удаленным резервным копированием, выполняются непосредственно на удаленном ленточном устройстве или с помощью удаленной репликации, проводимой на основе использования массива хранения данных.

Организации должны убедиться в том, что на площадке, предназначенной для восстановления работоспособности предприятия после возникновения чрезвычайных обстоятельств на площадке основного производства, то есть на так называемом DR-месте (disaster recovery site), для данных, попавших в резервную копию, применяется такой же уровень безопасности. Нужно, чтобы защита инфраструктур резервного копирования, репликации и архивирования могла реагировать сразу на несколько угроз, включая подмену допустимых идентификаторов, предназначенных для DR-места, фальсификацию данных, сетевой перехват, DoS-атаки и хищение носителей данных. Эти угрозы потенциально способны нарушить целостность, конфиденциальность и доступность данных. На рис. 14.4 показана обычная конструкция, предназначенная для удаленного резервного копирования, где данные, находящиеся в массиве хранения, реплицируются по DR-сети на вторичное устройство хранения, находящееся в DR-месте. В решении, использующемся для проведения резервного копирования, где компоненты хранения данных разделены сетью, должны быть учтены и угрозы на уровне передачи данных. В противном случае взломщик сможет подменить идентификатор сервера резервного копирования и запросить у хоста отправку его данных. Не прошедший авторизацию хост, утверждающий, что он и является сервером резервного копирования, может сделать так, что удаленное резервное копирование будет проводиться в то место, которое не проходило авторизацию и о котором ничего не известно. Кроме того, взломщики могут воспользоваться подключением к DR-сети для фальсификации данных, перехвата сети и перехода в DoS-атаку на устройства хранения данных.

Еще одной разновидностью угроз являются физические угрозы, связанные с утратой, хищением или отправкой по неверному адресу лент с резервными копиями, особенно если на этих лентах содержится совершенно секретная информация. То, что приложения для создания резервных копий на магнитных лентах не занимаются шифрованием данных в ходе копирования, чревато весьма серьезными последствиями.

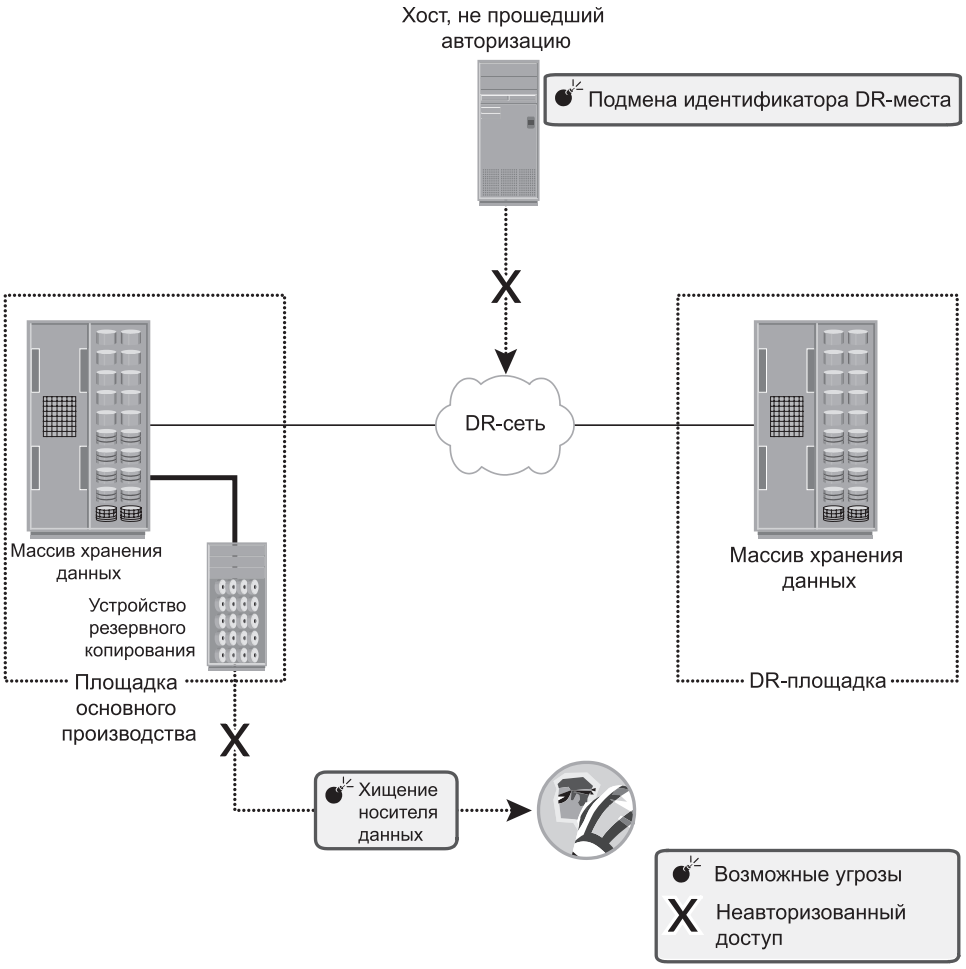


Рис. 14.4. Угрозы безопасности в средах резервного копирования, репликации и архивирования

14.4. Реализация мер безопасности в сети хранения данных

В следующих разделах будут рассмотрены подробности некоторых основных реализаций мер безопасности в средах FC-SAN-сетей, NAS-устройств и IP-SAN-сетей.

14.4.1. Реализация мер безопасности в FC-SAN-сетях

Традиционные FC-SAN-сети пользуются преимуществами систем безопасности, унаследованными у сетей, создаваемых на основе IP-технологий. Конфигурация FC-SAN-сети создается как изолированная закрытая среда с количеством узлов меньшим, чем у IP-сети. Следовательно, FC-SAN-сети меньше подвержены угрозам безопасности. Но с появлением конвергентных сетей и объединенных хранилищ данных, создаваемых под влиянием быстро развивающихся и широко востребованных разработок для крупных и сложных SAN-сетей, распространяющихся на несколько географических точек в рамках одного предприятия, этот сценарий изменился. Сегодня для FC-SAN-сетей нет единого комплексного решения, направленного на обеспечение безопасности. Многие механизмы безопасности FC-SAN-сети берут начало от своих аналогов в IP-сети, и в практику внедряются уже состоявшиеся решения в области обеспечения безопасности.

В стандартах (T11 standards) *протокола безопасности оптоволоконных сетей* — Fibre Channel Security Protocol (FC-SP), опубликованных в 2006 году, были согласованы механизмы и алгоритмы обеспечения безопасности между соединениями IP- и FC-сетей. В этих стандартах дается описание протоколов реализации мер безопасности в системе коммутации FC-сетей, а также элементов этой системы и входящих в нее портов типа N_Port. Туда также было включено руководство по аутентификации в FC-сетях, установке ключей сеанса, оговорены параметры, требующиеся для обеспечения покадровой целостности и конфиденциальности, а также создания и распространения политик в системе коммутации FC-сети.

Архитектура обеспечения безопасности FC-SAN-сети

Среда сетевого хранения данных в силу своей разбросанности и сложности является потенциальной мишенью для попыток неавторизованного доступа, хищения данных и манипулирования ими. Поэтому стратегии обеспечения безопасности основываются на концепции *глубоко эшелонированной обороны*, согласно которой рекомендуется создавать несколько объединенных в одно целое уровней обеспечения безопасности. Тем самым гарантируется, что сбой одного из средств контроля не приведет к взлому защищаемых активов. Уровни (зоны) среды сетевого хранения данных, нуждающиеся в защите и соблюдении обязательных к развертыванию мер безопасности, показаны на рис. 14.5.

FC-SAN-сети не только страдают от собственных конкретных рисков и уязвимостей, но и разделяют общие проблемы обеспечения безопасности, связанные с физической защитой и удаленным административным доступом. Вдобавок к реализации мер безопасности, специфичных для SAN-сетей, организации должны одновременно с ними использовать другие

реализации мер безопасности, принятые на предприятии. Полный перечень стратегий защиты, которые должны быть реализованы в различных зонах обеспечения безопасности, приведен в табл. 14.1. Некоторые из механизмов обеспечения безопасности, перечисленных в табл. 14.1, не относятся только к SAN-сетям, а являются технологиями, широко применяемыми во всем дата-центре. Например, широкое применение имеет двухфакторная аутентификация, для которой в простой обязательной реализации используется пара, состоящая из имени пользователя и пароля, а также дополнительные компоненты безопасности, например используемая для аутентификации смарт-карта.

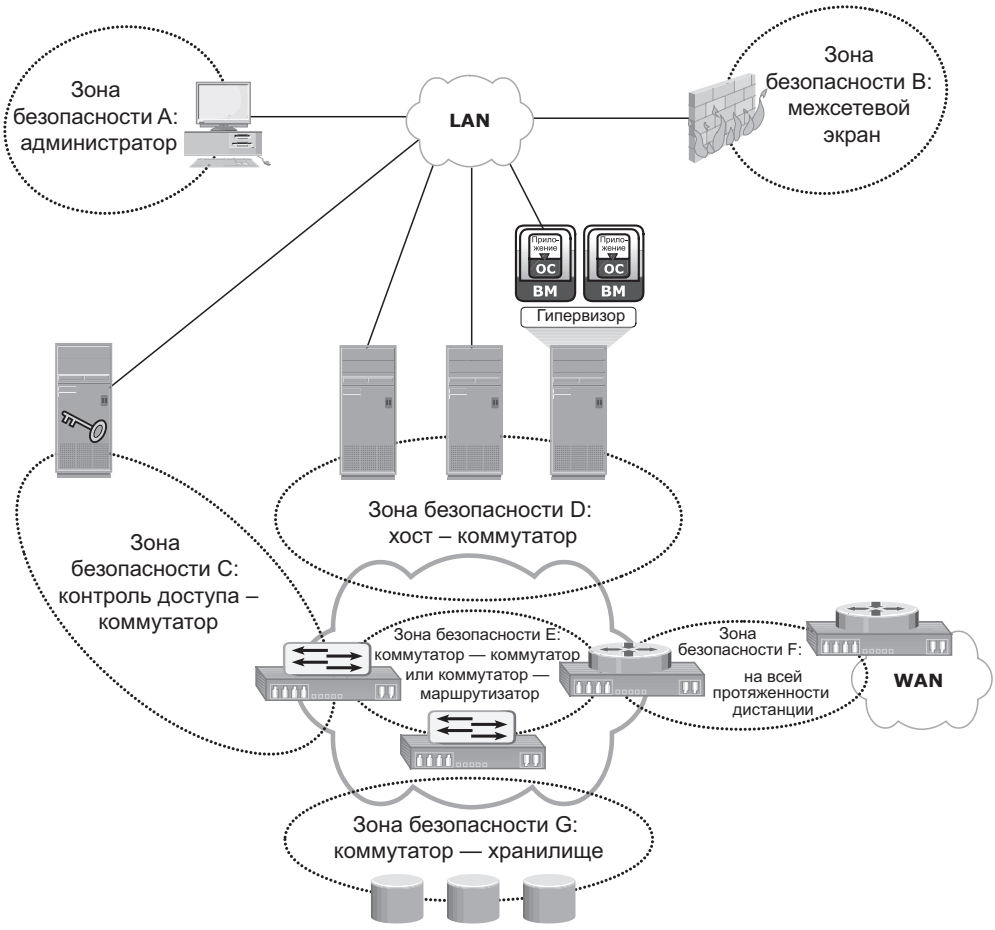


Рис. 14.5. Архитектура обеспечения безопасности FC-SAN-сети

Таблица 14.1. Зоны обеспечения безопасности и стратегии защиты

ЗОНЫ БЕЗОПАСНОСТИ	СТРАТЕГИИ ЗАЩИТЫ
Зона A (аутентификация на консоли управления)	а) Ограничить доступ к локальной сети управления, предоставив его только авторизованным пользователям (ограничить по MAC-адресам); б) реализовать VPN-туннелирование для безопасного удаленного доступа к локальной сети управления; в) использовать для сетевого доступа двухфакторную аутентификацию
Зона B (межсетевой экран)	Блокировать неприемлемый трафик путем: а) фильтрации адресов, не разрешенных в вашей локальной сети; б) экранирования допустимых протоколов, блокирования неиспользуемых портов
Зона C (контроль доступа — коммутатор)	Проводить аутентификацию пользователей и администраторов FC-коммутаторов с использованием Remote Authentication Dial In User Service (RADIUS), DH-CHAP (Diffie-Hellman Challenge Handshake Authentication Protocol) и т. д.
Зона D (от хоста до коммутатора)	Ограничить доступ к системе коммутации легальными хостами путем: а) реализации списков доступа (ACL) — известные HBA-адаптеры могут подключаться только к конкретным портам коммутатора; б) реализации метода создания зон безопасности, например зонирования портов (которое известно также как жесткое зонирование)
Зона E (от коммутатора до коммутатора или от коммутатора до маршрутизатора)	Защитить трафик в системе коммутации путем: а) использования аутентификации портов типа E_Port; б) шифрования трафика в пути; в) реализации средств контроля FC-коммутаторов и средств контроля портов
Зона F (на всей протяженности дистанции)	Применить шифрование для данных, находящихся в процессе передачи по сети: а) FC-SP для FC-сетей большой протяженности; б) IPSec для дистанций SAN-сетей, реализованных с использованием технологии FCIP
Зона G (от коммутатора до хранилища данных)	Защитить массивы хранения данных на своей SAN-сети путем: а) маскирования LUN-устройств на основе применения WWPN-имен; б) применения S_ID-блокировки, то есть маскирования на основе исходных FC-адресов

Основные механизмы обеспечения безопасности SAN-сети

К наиболее часто используемым методам обеспечения безопасности SAN-сети относятся маскирование и зонирование LUN-устройств, контроль доступа на всех коммутаторах и системах коммутации, RBAC и логическое разбиение системы коммутации (создание виртуальных SAN-сетей).

Маскирование и зонирование LUN-устройств

Маскирование и зонирование LUN-устройств относится к основным механизмам обеспечения безопасности SAN-сетей, используемым для защиты от неавторизованного доступа к хранилищу. Маскирование и зонирование LUN-устройств подробно рассматриваются в главах 4 и 5 соответственно. При стандартной реализации маскирования LUN-устройств в массивах хранения данных маска накладывается на LUN-устройства, представляемые интерфейсному порту хранилища, на основе WWPN-имен исходных HBA-адаптеров. Иногда может предлагаться более строгий вариант маскирования LUN-устройств, где маскирование может производиться на основе исходных FC-адресов. При этом предлагается механизм сведения FC-адреса конкретного порта узла к его WWN-имени. В средах, требующих особых мер безопасности, наиболее подходящим выбором будет *WWPN-зонирование*.

Обеспечение безопасности портов коммутаторов

Кроме зонирования и маскирования LUN-устройств есть еще дополнительные механизмы обеспечения безопасности, которые могут применяться в отношении портов коммутатора, такие как привязка порта, закрытие порта, блокировка порта и постоянное отключение порта. *Привязка порта* ограничивает количество устройств, которые могут быть подключены к конкретному порту коммутатора, и позволяет подключаться к узлу для доступа к системе коммутации только соответствующему порту коммутатора. Привязка портов уменьшает угрозу подмены WWPN-имен, но не исключает ее возможности. *Закрытие порта* и *блокировка порта* ограничивают тип инициализации порта коммутатора. Обычные варианты блокировки порта не дают ему возможности работать в роли порта типа E_Port и не позволяют использовать его для создания межкоммутаторных линий связи (ISL), имитируя его принадлежность к неисправному коммутатору. Некоторые варианты гарантируют сведение роли порта лишь к функциям портов типа FL_Port, F_Port, E_Port или их сочетаний. *Постоянное отключение порта* препятствует включению порта коммутатора даже после перезагрузки устройства.

Контроль доступа в масштабах коммутатора и всей системы коммутации

По мере наращивания организациями своих SAN-сетей как в локальном масштабе, так и на большом удалении от основного места производства они

все больше нуждаются в эффективном управлении системой безопасности сетей. Сетевая система безопасности может быть сконфигурирована на FC-коммутаторе с использованием *списков контроля доступа* — access control lists (ACL) и на всей системе коммутации путем привязки коммутаторов (допуска к подключению к сети только тех коммутаторов, которые указаны в списке разрешенных).

ACL объединяют в себе политики контроля подключения устройств и контроля подключения коммутаторов. Политика контроля подключения устройств определяет, какие именно HBA-адаптеры и порты хранилища могут быть частью системы коммутации, предотвращая тем самым доступ к себе со стороны неавторизованных устройств. По аналогии с этим политика контроля подключения коммутаторов определяет, каким именно коммутаторам разрешено быть частью системы коммутации, предотвращая тем самым подключение к системе неавторизованных коммутаторов.

Привязка системы коммутации предотвращает подключение к имеющимся в ней коммутаторам других коммутаторов, не прошедших авторизацию. Она гарантирует наличие на каждом коммутаторе авторизованных данных о принадлежности к системе, и любые попытки подключить к любому коммутатору системы какой-нибудь другой коммутатор с помощью ISL-линии приводят к сегментации системы коммутации.

Контроль доступа на основе ролей придает SAN-сети дополнительную гарантию безопасности путем предотвращения неавторизованных операций управления в системе коммутации. Это средство позволяет администратору системы безопасности назначать пользователям определенные роли с конкретным указанием привилегий или прав доступа после регистрации в системе коммутации. Например, роль администратора зон позволяет изменять зоны, определенные в системе коммутации, а обычный пользователь может лишь просматривать любую информацию, связанную с системой коммутации, например, о типах портов и зарегистрированных в системе узлах.

Логическое разбиение системы коммутации: виртуальные SAN-сети

VSAN-сети позволяют на основе обычной физической SAN-сети создавать несколько логических SAN-сетей. Они предоставляют возможность выстраивать более крупные консолидированные системы коммутации и при этом по-прежнему обеспечивать требуемый уровень безопасности и изолированности сетей друг от друга. Логическое разбиение в VSAN-сети показано на рис. 14.6.

Администратор SAN-сети может создавать обособленные VSAN-сети путем заполнения каждой из них с помощью портов коммутатора. В показанном на рисунке примере порты коммутатора распределены по двум VSAN-сетям: 10 и 20, соответственно, для инженерного отдела и отдела кадров. Несмотря на то что физическое коммутационное оборудование используется

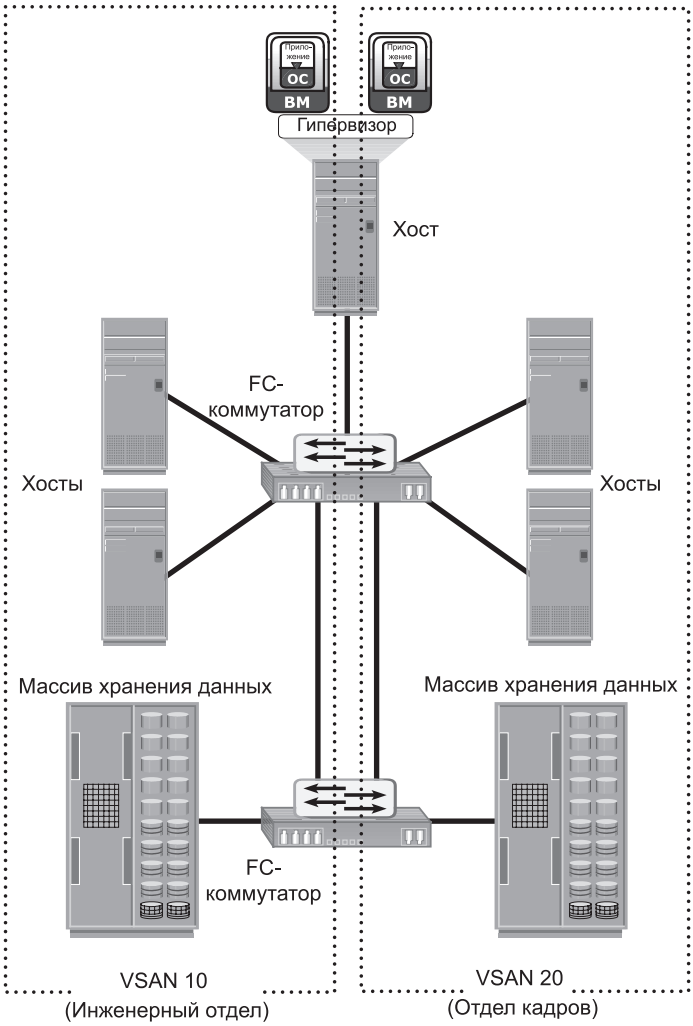


Рис. 14.6. Обеспечение безопасности SAN-сети с помощью создания VSAN-сетей

ими совместно с другими отделами, управление может осуществляться по отдельности, как в автономных системах коммутации. Для защиты всей физической SAN-сети для каждой VSAN-сети должно применяться зонирование. Каждая управляемая VSAN-сеть одновременно может иметь только один набор активных зон.

VSAN-сети сводят к минимуму влияние общесистемных, применительно к системе коммутации, разрушительных событий, поскольку управление и контроль трафика в SAN-сети, куда могут быть включены RSCN-уведомления, а также события приведения в активное состояние наборов зон и многое другое не выходят за границы VSAN-сети. Поэтому

VSAN-сети являются вполне рентабельной альтернативой созданию изолированных физических систем коммутации. Они способствуют доступности информации и обеспечению безопасности путем изолирования событий системы коммутации и предоставления контроля авторизации в пределах одной системы коммутации.

14.4.2. NAS-устройства

NAS-устройства открыты для множества вредоносных программ, включая вирусы и черви, а также для неавторизованного доступа, подмены идентификаторов и фальсификации данных. Для защиты данных и инфраструктуры их сетевого хранения на NAS-устройствах реализованы различные механизмы обеспечения безопасности.

Права доступа и ACL составляют первый уровень защиты ресурсов NAS-устройств, действующий путем ограничения их доступности и возможности совместного использования. Права доступа будут играть роль надстройки над исходными характеристиками и атрибутами, связанными с файлами и папками. Кроме того, для проверки идентичности пользователей сети и определения имеющихся у них привилегий реализуются другие механизмы аутентификации и авторизации, такие как Kerberos и службы каталогов. Кроме того, защита инфраструктуры хранения данных от неавторизованного доступа и вредоносных атак осуществляется с помощью межсетевых экранов.

Совместное использование файлов в NAS-устройствах: ACL ОС Windows

В Windows поддерживаются два типа ACL: *списки разграничительного контроля доступа* — discretionary access control lists (DACL) и *системные списки контроля доступа* (SACL). DACL обычно называют просто ACL, определяющим контроль доступа. SACL определяет, какой доступ нужно подвергать аудиту, если функция проведения аудита включена.

Кроме этих ACL в Windows также поддерживается концепция владения объектами. У владельца объекта имеются жестко закодированные права на этот объект, и эти права не нуждаются в явном предоставлении в SACL. Сведения о владельце, SACL и DACL статически закрепляются за каждым объектом. Windows также предлагает функциональную возможность наследования прав доступа, что позволяет дочерним объектам, существующим внутри родительского объекта, автоматически наследовать ACL родительского объекта.

ACL применимы также к объектам каталога, известным как идентификаторы безопасности — security identifiers (SID). Они автоматически создаются Windows-сервером или доменом при создании пользователя или группы и абстрагируются от пользователя. Иначе говоря, хотя пользователь может определить идентификатор своего имени пользователя (login ID)

как User1, это будет всего лишь текстовое представление реального SID-идентификатора, используемого той операционной системой, под управлением которой ведется работа. Внутренние процессы, идущие в Windows, при предоставлении доступа к объекту ссылаются на SID учетной записи, а не на используемое в ней имя пользователя или группы. ACL устанавливаются путем использования стандартного GUI-интерфейса Windows Explorer, но могут настраиваться также с помощью команд интерфейса командной строки (CLI) или других инструментальных средств, созданных сторонними разработчиками.

Совместное использование файлов в NAS-устройствах: права доступа в системе UNIX

Для операционной системы UNIX *пользователь* является абстракцией, означающей логическую единицу для назначения прав на владение и привилегий при работе в системе. Пользователем может быть либо человек, либо системная операция. Система UNIX понимает только привилегии пользователя на выполнение определенных операций в системе и распознает каждого пользователя по его пользовательскому идентификатору — user ID (UID) и имени пользователя независимо от того, человек это, системная операция или устройство.

В UNIX пользователи могут объединяться в одну или несколько групп. Понятие группы служит для назначения набора привилегий для заданного ресурса и распространения их среди множества пользователей, нуждающихся в данных привилегиях. Например, группа людей, работающих над одним и тем же проектом, может нуждаться в одних и тех же правах доступа к определенному набору файлов.

Права доступа в UNIX определяют операции, которые могут выполняться любым пользователем, имеющим какое-либо отношение к владению файлом. Проще говоря, эти права определяют, что может сделать владелец, что может сделать группа владельцев и что могут сделать с файлом все остальные пользователи. Для любого заданного отношения к владению при определении прав доступа используются три бита. Первый бит означает доступ для чтения — read (r), второй бит означает доступ для записи — write (w), и третий бит означает доступ для запуска на выполнение — execute (x). Поскольку в UNIX определяются три вида отношений к владению (владелец — Owner, группа — Group и все остальные — All), для каждого отношения к владению нужна структура из трех элементов (определяющая права доступа), в результате чего получается девять битов. Каждый бит может быть либо установлен, либо сброшен. Когда информация выводится на экран, набор битов маркируется буквой соответствующей операции (r, w или x), при этом сброшенный бит показывается с дефисом (-), и все биты помещаются в одной строке, например, rwxr-xr-x. В данном примере пользователь может применять к файлу весь набор операций, а вот группа владельцев и все остальные пользователи могут только читать файл или запускать его на выполнение. При

выводе на экран перед этим шаблоном из девяти битов должен стоять символ, обозначающий образ действий файла. Например, если файл представляет собой каталог, он обозначается буквой «d», а если он является ссылкой, в обозначении фигурирует буква «l».

Совместное использование файлов в NAS-устройствах: аутентификация и авторизация

В среде совместного использования файлов в NAS-устройствах применяются стандартные протоколы такого использования, NFS и CIFS. Поэтому аутентификация и авторизация реализованы и поддерживаются на NAS-устройствах точно так же, как в среде совместного использования файлов в UNIX или в Windows.

Аутентификация требует проверки идентичности пользователя сети и поэтому включает в себя в среде UNIX поиск учетных данных регистрации в сервере сетевой информационной системы — Network Information System (NIS). По аналогии с этим клиент Windows проходит аутентификацию с помощью имеющегося в Active Directory контроллера домена Windows. В Active Directory для доступа к информации об объектах сети, имеющихся в каталоге, используется LDAP, а для обеспечения безопасности сети используется технология Kerberos. В NAS-устройствах для подтверждения учетных данных пользователя сети используются такие же технологии аутентификации. Процесс аутентификации в NAS-среде показан на рис. 14.7.

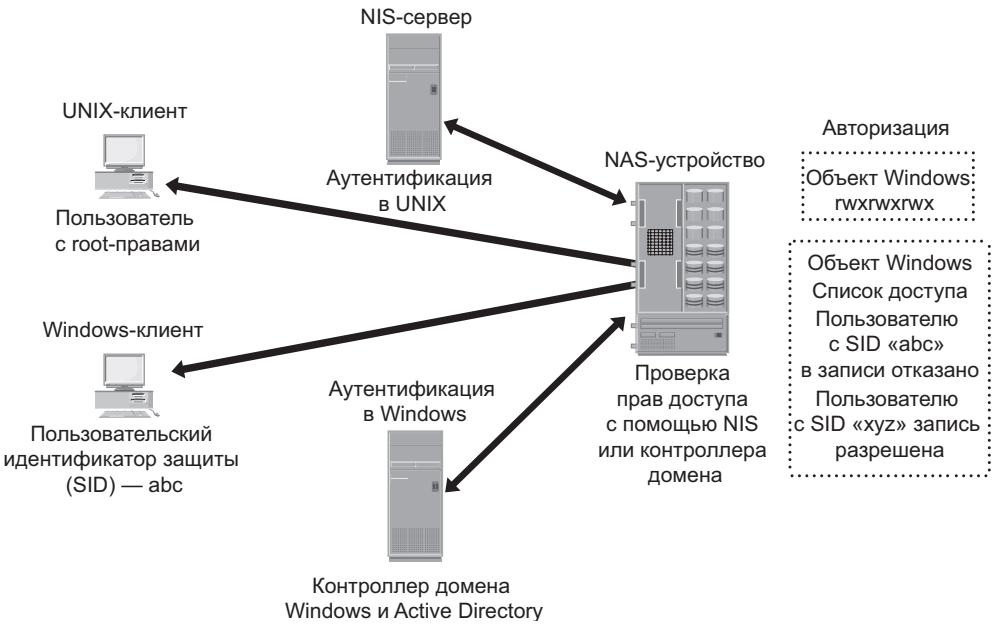


Рис. 14.7. Обеспечение безопасного пользовательского доступа в NAS-среде

Авторизация определяет права пользователя в сети. Технологии авторизации для UNIX-пользователей и Windows-пользователей совершенно не похожи друг на друга. В файлах UNIX для определения прав доступа, предоставленных владельцу, группе или всем остальным пользователям, используются биты режима работы, а в Windows для предоставления конкретному пользователю в отношении конкретного файла тех или иных прав или для отказа в их предоставлении используется ACL.

Хотя в NAS-устройствах для пользователей UNIX и Windows поддерживаются обе эти методологии, когда UNIX- и Windows-пользователи обращаются к одним и тем же файлам и используют их совместно, ситуация усложняется. Если в NAS-устройстве поддерживаются несколько протоколов, должна быть сохранена целостность методологий проверки прав доступа. Поставщики NAS-устройств предоставляют метод отображения прав доступа, определяемых в UNIX, на права доступа, определяемые в Windows, и наоборот, поэтому может быть поддержана среда с использованием сразу нескольких протоколов. Но все эти сложности, связанные с использованием нескольких протоколов, должны быть учтены при проектировании NAS-решений. Вместе с тем нужно проверить возможность подключения и ширину полосы пропускания контроллера домена и NIS-сервера. Если требуется доступ с использованием нескольких протоколов, нужно будет обратить внимание на конкретную реализацию политики доступа к файлам, предлагаемую поставщиком.

Kerberos

Kerberos представляет собой сетевой протокол аутентификации, разработанный с целью обеспечения строгой аутентификации для клиент-серверных приложений путем использования шифрования с секретным ключом. Благодаря применению криптографии клиент и сервер могут предоставлять друг другу свои идентификаторы по небезопасному сетевому соединению. После того как клиент и сервер подтвердят свою идентичность, они могут сделать выбор в пользу шифрования всего объема своего трафика, гарантируя тем самым закрытость и целостность этих данных.

В Kerberos аутентификация происходит между клиентами и серверами. Клиент получает билет на обслуживание, а сервер расшифровывает его, используя свой секретный ключ. Любой получивший билет на обслуживание в службе Kerberos, будь то пользователь или хост, называется *Kerberos-клиентом*. Понятие *Kerberos-сервер* обычно означает центр распределения ключей — Key Distribution Center (KDC). В KDC реализуются служба аутентификации — Authentication Service (AS) и служба предоставления ключей — Ticket Granting Service (TGS). В KDC имеется копия каждого пароля, связанного с каждым принципалом (пользователем или процессом, имеющим учетную запись), поэтому очень важно, чтобы KDC не подвергался никаким опасностям. Пользователи и серверы, для которых секретный ключ хранится в базе данных KDC, известны как *принципалы*.

В NAS-среде Kerberos используется в основном при аутентификации в домене Microsoft Active Directory, хотя может быть применен и при

выполнении функций обеспечения безопасности в средах UNIX. Процесс Kerberos-аутентификации (рис. 14.8) включает в себя следующие этапы.

1. Пользователь регистрируется на рабочей станции в домене (или в лесу) Active Directory с помощью своего идентификатора и пароля. Компьютер клиента отправляет запрос на получение Kerberos-билета службе аутентификации (AS), запущенной в KDC-центре. KDC проверяет информацию об имени пользователя на основе данных Active Directory. (Этот этап на рис. 14.8 в явном виде не показан.)
2. KDC отвечает на запрос зашифрованным билетом на получение билета — Ticket Granting Ticket (TGT) и зашифрованным ключом сеанса. У TGT ограниченный срок действия. TGT может быть зашифрован только KDC-центром, а клиент может расшифровать только ключ сеанса.
3. Когда клиент нуждается в обслуживании со стороны сервера, он отправляет KDC-центру запрос на обслуживание, состоящий из ранее сгенерированного TGT-билета, зашифрованного ключом сеанса, а также из информации о ресурсе.
4. KDC-центр проверяет права доступа в Active Directory и убеждается в том, что пользователь имеет право на получение данной услуги.
5. KDC-центр возвращает клиенту билет на обслуживание. В этом билете имеются поля, адресуемые клиенту и серверу, осуществляющему обслуживание.

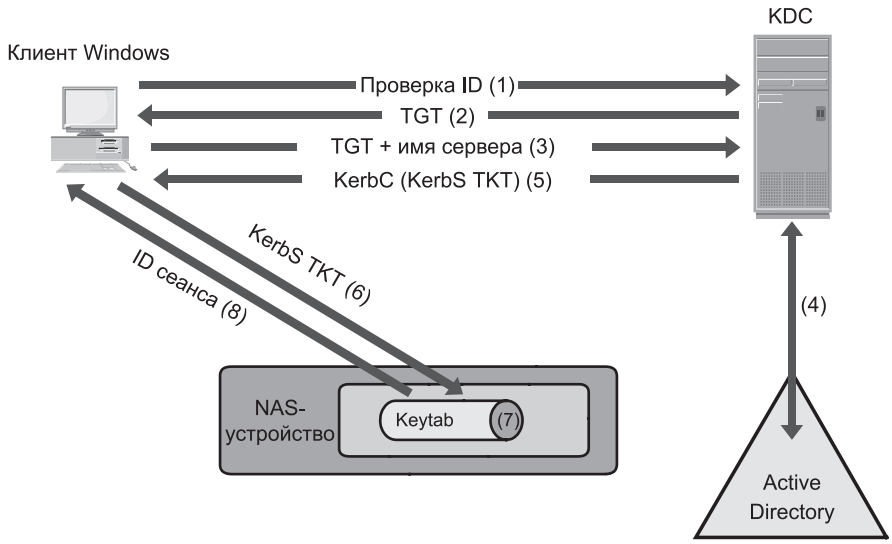


Рис. 14.8. Авторизация по Kerberos-протоколу

6. Клиент отправляет билет на обслуживание серверу, на котором находятся нужные ему ресурсы.
7. Сервер, в данном случае NAS-устройство, расшифровывает предназначенную для сервера часть билета и сохраняет информацию в файле ключа (keytab-файле). При условии действительности билета Kerberos-клиента этот процесс авторизации в повторении не нуждается. Сервер автоматически разрешает клиенту доступ к соответствующим ресурсам.
8. Теперь устанавливается сеанс клиента с сервером. Сервер возвращает клиенту идентификатор сеанса, с помощью которого, при условии активности сеанса, отслеживается деятельность клиента, например установка им блокировки на файл.

Межсетевые экраны, работающие на сетевом уровне

Поскольку в NAS-устройствах используется протокол IP-стека, они страдают уязвимостью для разнообразных атак, инициируемых по общедоступной IP-сети. Для защиты NAS-устройств от этих угроз безопасности в NAS-среде применяются межсетевые экраны, работающие на сетевом уровне. Эти межсетевые экраны могут исследовать сетевые пакеты на предмет их соответствия установленным правилам безопасности. Пакеты, запрещенные этими правилами, отбрасываются, и их доставка к месту назначения прекращается. Правила могут устанавливаться на основе адреса источника (сети или хоста), адреса приемника (сети или хоста), номера порта или сочетания этих факторов (IP источника, IP приемника и номера порта). Эффективность межсетевого экрана зависит от строгости и области распространения правил безопасности. Нечетко определенный набор правил может повысить вероятность взлома системы безопасности.

Типичная реализация межсетевого экрана показана на рис. 14.9. В сетевой среде обычно используется демилитаризованная зона — demilitarized zone (DMZ). DMZ предоставляет средства, обеспечивающие безопасность внутренних активов наряду с разрешением доступа к различным ресурсам на основе интернет-протоколов. В DMZ-среде серверы, к которым нужно обратиться через Интернет, помещаются между двумя наборами межсетевых экранов. Пройти через межсетевой экран к серверу в DMZ можно только к портам, определенным приложением, таким как HTTP или FTP. А трафику, не относящемуся к Интернету, разрешено проходить через второй набор межсетевых экранов и получать доступ к внутренней сети.

Серверам в DMZ может разрешаться, а может и не разрешаться связь с внутренними ресурсами. При такой настройке сервер, находящийся в DMZ, является веб-приложением с выходом в Интернет и доступом к данным, хранящимся на NAS-устройстве, которое может находиться во внутренней закрытой сети. По плану обеспечения безопасности через DMZ будут

обслуживаться только данные, направляющиеся в адрес внутреннего и внешнего приложений.

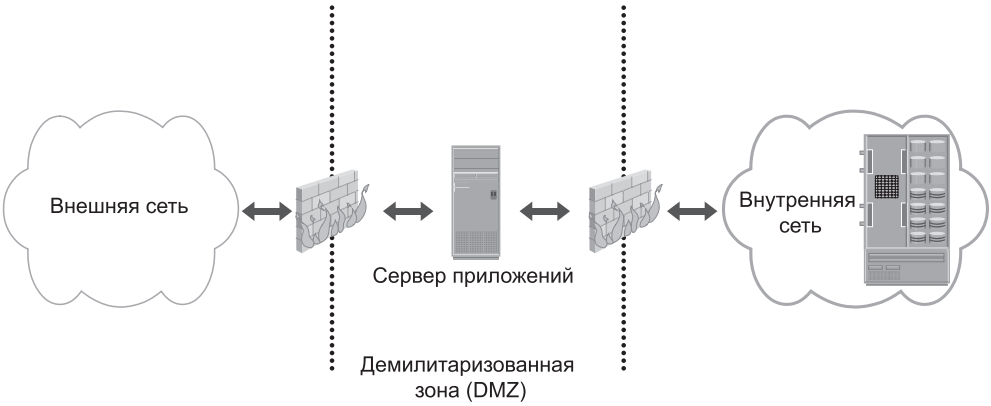


Рис. 14.9. Обеспечение безопасности NAS-среды с помощью межсетевого экрана, работающего на сетевом уровне

МЕЖСЕТЕВЫЕ ЭКРАНЫ, РАБОТАЮЩИЕ НА УРОВНЕ ПРИЛОЖЕНИЙ, И XML-МЕЖСЕТЕВЫЕ ЭКРАНЫ

Межсетевые экраны, работающие на уровне приложений, и XML-межсетевые экраны относятся к межсетевым экранам третьего поколения, контролирующим доступ к приложению путем фильтрации трафика, не отвечающего настроенной политике межсетевого экрана. В отличие от межсетевых экранов, работающих на сетевом уровне и сканирующих пакеты на основе адреса источника, адреса приемника и т. д., межсетевые экраны, работающие на уровне приложений, проводят более детальное сканирование содержимого пакетов. XML-межсетевой экран является специализированным средством, работающим на уровне приложений и защищающим приложения, доступные через интерфейсы на основе XML. XML-межсетевые экраны обычно развертываются в DMZ-среде организации, проверяют XML-трафик, фильтруя XML-содержимое, и контролируют доступ к ресурсам, предоставляемым на основе XML.

14.4.3. Обеспечение безопасности сетей IP-SAN

В этом разделе рассматриваются некоторые из основных механизмов обеспечения безопасности, используемые в среде сетей IP-SAN. Основным общепринятым сетевыми устройствами и хостами механизмом проведения

аутентификации является *протокол аутентификации по методу «вызов-приветствие»* — Challenge-Handshake Authentication Protocol (CHAP). CHAP-протокол предоставляет для инициаторов и адресатов метод, позволяющий им идентифицировать друг друга путем использования секретного кода или пароля. Секретные коды CHAP обычно представляют собой случайные секретные последовательности длиной от 12 до 128 символов. Обмен секретными кодами никогда не осуществляется по каналу связи напрямую, вместо этого секретные коды превращаются односторонней хэш-функцией в хэш-значения, которыми затем и осуществляется обмен. В хэш-функции используется алгоритм MD5, который преобразует данные таким образом, что получается уникальное значение, которому не может быть возвращен его прежний вид. Процесс CHAP-аутентификации показан на рис. 14.10.

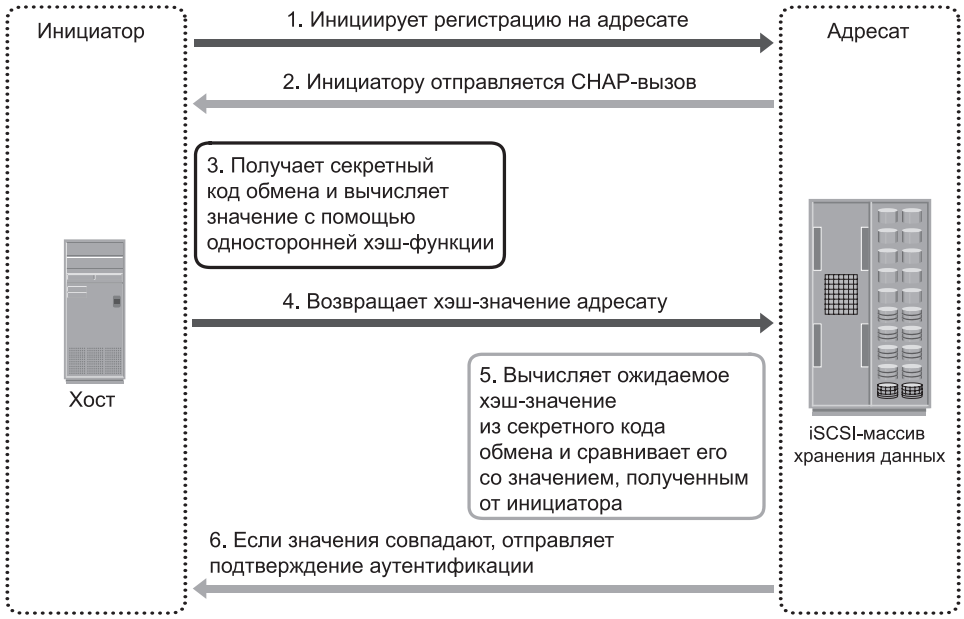


Рис. 14.10. Обеспечение безопасности сети IP-SAN с помощью CHAP-аутентификации

Если инициатору нужна обратная CHAP-аутентификация, инициатор аутентифицирует адресата с помощью такой же процедуры. Секретный код CHAP должен быть настроен на инициаторе и на адресате. CHAP-запись, состоящая из имени узла и секретного кода, связанного с этим узлом, хранится адресатом и инициатором. Аналогичные этапы выполняются при сценарии двусторонней CHAP-аутентификации. Когда эти этапы будут завершены, инициатор проводит аутентификацию адресата. Если оба цикла

аутентификации пройдут успешно, доступ к данным будет разрешен. CHAP-аутентификация используется довольно часто благодаря очень простому для реализации протоколу и возможности ее реализации на множестве различных систем.

Домены обнаружения iSNS функционируют так же, как и зоны Fibre Channel. Домены обнаружения предоставляют возможность функционального создания групп устройств в IP-SAN. Чтобы устройства могли вести обмен данными друг с другом, они должны быть настроены в одном и том же домене обнаружения. Уведомления об изменении состояния — state change notifications (SCN-уведомления) оповещают iSNS-сервер о добавлении устройств в домен обнаружения или об их удалении из домена. Домены обнаружения в iSNS показаны на рис. 14.11.

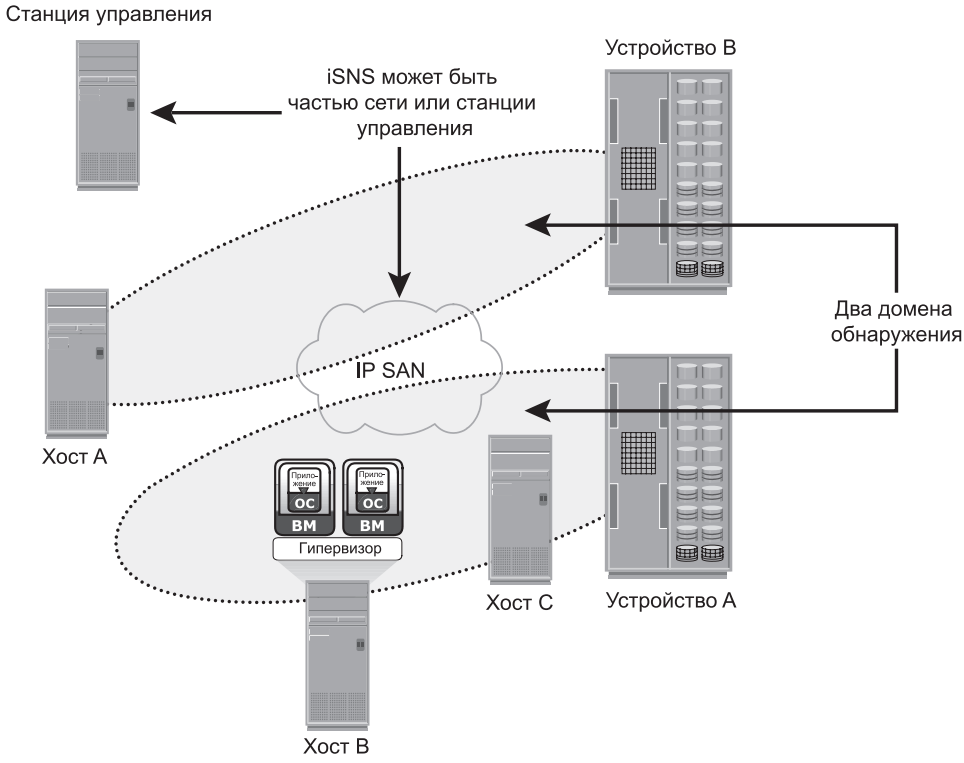


Рис. 14.11. Обеспечение безопасности IP-SAN с помощью доменов обнаружения iSNS

14.5. Обеспечение безопасности инфраструктуры хранения данных в виртуализированных и облачных средах

До сих пор речь в главе шла только об угрозах и мерах безопасности, касающихся традиционного дата-центра. Эти же угрозы и меры применимы и к хранилищам информации в виртуализированной и облачной среде. Но среды виртуализированных и облачных вычислений, работающих в многопользовательском режиме, из-за отсутствия контроля над облачными ресурсами создают для данных, принадлежащих организации, дополнительные угрозы. Общественное облако вызывает больше опасений, чем частное, и требует применения дополнительных контрмер. Дело в том, что в общественном облаке у пользователей (потребителей) обычно весьма ограниченные возможности контроля над ресурсами, вследствие чего обеспечить потребителей механизмами безопасности довольно трудно.

С точки зрения безопасности, факторы, определяющие порядок ее обеспечения, имеются как у потребителей, так и у поставщиков облачных услуг — cloud service providers (CSP): и тем, и другим приходится сталкиваться с многочисленными угрозами. Проблемы обеспечения безопасности и предпринимаемые меры рассматриваются в следующих разделах.

14.5.1. Проблемы обеспечения безопасности

Темпы внедрения виртуализации и облачных вычислений в повседневную деятельность организаций стремительно растут, но при этом приходится сталкиваться с целым рядом проблем обеспечения безопасности. Основными проблемами являются многопользовательский режим работы, стремительность развития атаки, проблема выдачи гарантий на предоставление информации и проблема обеспечения конфиденциальности данных.

Проблема многопользовательского режима, возникающая в результате внедрения виртуализации, заключается в разрешении множеству независимых друг от друга арендаторов получать услуги с использованием одного и того же набора ресурсов хранилища. Несмотря на преимущества, предлагаемые возможностью обслуживания сразу нескольких арендаторов, этот фактор по-прежнему является наиболее проблемным как для пользователей, так и для поставщиков услуг. Сосуществование нескольких виртуальных машин на одном сервере и совместное использование ими ряда ресурсов расширяет поверхность атак. Может случиться, что важные бизнес-данные одного арендатора станут доступны другим конкурирующим с ним арендаторам, запускающим приложения с использованием тех же самым ресурсов.

Проблема стремительности развития атаки касается той ситуации, при которой любая существующая угроза безопасности распространяется в облаке намного быстрее и оказывает более серьезное воздействие, чем в средах традиционных дата-центров. *Гарантии предоставления информации*

должны обеспечивать пользователям конфиденциальность, целостность и доступность данных, находящихся в облаке. Кроме того, пользователям облака нужны гарантии, что все проводимые ими в облаке операции не будут подменены другими операциями и доступ к данным будет предоставлен только при наличии законных прав и соблюдении границ.

Проблема закрытости данных стоит также в виртуализированной и облачной среде в числе главных. Поставщик облачных услуг должен гарантировать, что личная информация — Personally Identifiable Information (PII), касающаяся его клиентов, юридически защищена от любого несанкционированного раскрытия.

14.5.2. Меры обеспечения безопасности

Меры безопасности могут применяться на уровне вычислительной системы, сети и хранилища. Реализованные на трех уровнях, эти меры снижают риски возникновения угроз в виртуализированной и облачной среде.

Обеспечение безопасности на уровне вычислительной системы

Обеспечение безопасности вычислительной инфраструктуры включает в себя выполнение мер безопасности в отношении физического сервера, гипервизора, виртуальной машины и гостевой операционной системы (операционной системы, запущенной на виртуальной машине).

Обеспечение безопасности физического сервера заключается в реализации механизмов аутентификации и авторизации пользователя. Эти механизмы идентифицируют пользователей и предоставляют им права доступа к серверу. Чтобы свести к минимуму поверхность возможных атак против сервера, неиспользуемые компоненты оборудования, такие как сетевые адаптеры, USB-порты или приводы накопителей, должны быть удалены или отключены.

Гипервизор является единственной точкой взлома системы безопасности сразу для всех запущенных на нем виртуальных машин. Руткиты и вредоносные программы, установленные на гипервизоре, затрудняют антивирусным программам, установленным на гостевой операционной системе, обнаружение вирусов. Чтобы защититься от взломов, необходимо регулярно устанавливать обновления гипервизоров, играющие важную роль в обеспечении безопасности. Кроме того, должна быть защищена имеющаяся в гипервизоре система управления. Взломы и внедрения в систему управления могут повлиять на работу всех существующих виртуальных машин и позволить взломщикам создавать новые виртуальные машины. Доступ к системе управления должен предоставляться только авторизованным администраторам. Более того, заниматься этим должен отдельный межсетевой экран, установленный между системой управления и всеми остальными устройствами сети.

Изоляция и усиление защиты виртуальных машин является одним из наиболее часто применяемых механизмов защиты, позволяющих эффективно охранять виртуальные машины от атак. Изоляция виртуальной машины помогает предотвратить влияние взломанной гостевой операционной системы на другие гостевые операционные системы. Эта изоляция реализуется на уровне гипервизора. Кроме изоляции нужно повысить устойчивость защиты виртуальной машины к угрозам безопасности. Усиление защиты представляет собой процесс изменения исходной конфигурации для достижения более высокого уровня безопасности.

Кроме мер обеспечения безопасности гипервизора и виртуальных машин виртуальные и облачные среды нуждаются в дополнительных мерах защиты на уровне гостевой операционной системы и приложений.

ВЫСОКОНАДЕЖНОЕ СЕТЕВОЕ СОЕДИНЕНИЕ — TRUSTED NETWORK CONNECT (TNC)

TNC является протоколом, основанным на принципе трех «А» (authentication — аутентификации, authorization — авторизации и accounting — учете), с возможностью проведения авторизации клиентов сети на основе конфигурации оборудования, BIOS, версий ядра, установленных обновлений операционной системы и антивирусных программ, а также других сведений. Этот протокол был разработан некоммерческой организацией Trusted Computing Group (TCG), занимающейся выработкой промышленных стандартов. TCG создает спецификации на базе концепции аппаратных оснований для обеспечения доверия в отношении широкого круга устройств, приложений и услуг.

Обеспечение безопасности на уровне сети

Основными мерами безопасности, сводящими к минимуму уязвимости на сетевом уровне, являются межсетевой экран, обнаружение несанкционированных проникновений, демилитаризованная зона — demilitarized zone (DMZ) и шифрование данных на лету.

Межсетевой экран защищает сети от неавторизованного доступа, разрешая иметь доступ только в рамках законного обмена данными. В виртуализированной и облачной средах межсетевой экран может также защищать гипервизоры и виртуальные машины. Например, если гипервизор доступен для удаленного администрирования, доступ ко всему удаленному администрированию должен быть ограничен межсетевым экраном. Безопасность трафика между виртуальными машинами также обеспечивается применением межсетевого экрана. В таком случае услуги межсетевого экрана могут предоставляться *виртуальным* межсетевым экраном — Virtual Firewall (VF), представляющим собой службу межсетевого экрана, работающую исключительно

на гипервизоре. VF проводит фильтрацию пакетов и отслеживает трафик между виртуальными машинами, позволяя визуально наблюдать и контролировать трафик виртуальных машин и применять политики на уровне виртуальных машин.

Обнаружение несанкционированных проникновений — Intrusion Detection (ID) представляет собой процесс обнаружения событий, способных нанести вред конфиденциальности, целостности или доступности ресурсов. ID-система (IDS) проводит автоматический анализ событий, проверяя событие или последовательности событий на соответствие известным схемам аномальных действий или на их статистическое отличие от большинства других событий в системе. При обнаружении чего-либо необычного IDS выдает предупреждение. Также в качестве мер безопасности в виртуализированной и облачной среде развертываются DMZ и используется шифрование данных. Но процессы развертывания ничем не отличаются от аналогичных процессов в обычном дата-центре.

Обеспечение безопасности на уровне хранилища

Основные угрозы системам хранения данных в виртуализированной и облачной средах возникают из-за взломов вычислительного устройства, сети и системы безопасности на физических уровнях. Причина в том, что доступ к системам хранения данных происходит через инфраструктуры вычислительных устройств и сети. Поэтому для получения гарантий безопасности хранилища соответствующие меры безопасности должны быть предприняты на уровнях вычислительных устройств и сети. К числу наиболее часто используемых механизмов обеспечения безопасности, защищающих хранилище данных, можно отнести:

- методы контроля доступа, регулирующие состав пользователей и процессов, имеющих доступ к данным в системах хранения;
- зонирование и маскирование LUN-устройств;
- шифрование хранящихся (в системе хранения) данных и тех данных, которые находятся в процессе передачи. Шифрование данных должно включать шифрование резервных копий и раздельное содержание ключей к шифрам и данных;
- стирание данных, позволяющее убирать все следы удаленных данных.

Наряду с применением всех этих механизмов повысить безопасность систем хранения данных можно путем изоляции различных видов трафика с помощью применения VSAN-сетей. Для хранилищ, используемых гипервизором, требуется предпринимать дополнительные меры безопасности. Хранилищам для гипервизоров, использующим кластерные файловые системы, которые поддерживают сразу несколько виртуальных машин, могут понадобиться отдельные LUN-устройства для компонентов виртуальных машин и их данных.

14.6. Практическая реализация концепций: продукты RSA и VMware Security

RSA является подразделением, работающим в компании EMC над решением вопросов обеспечения безопасности и являющимся основным поставщиком решений в области безопасности, снижения рисков и обеспечения совместимости, которые помогают организациям справляться с наиболее сложными и важными проблемами обеспечения безопасности данных.

Компания VMware предлагает безопасные и надежные решения в области виртуализации для виртуализированных и облачных сред. В данном разделе будет дан краткий обзор таких продуктов, как RSA SecureID, RSA Identity and Access Management, RSA Data Protection Manager и VMware vShield.

14.6.1. RSA SecurID

RSA SecurID является средством двухфакторной идентификации, предоставляющим дополнительный уровень безопасности, гарантирующий доступ к системам и данным только допустимым пользователям. Работа RSA SecurID основана на какой-либо информации, известной пользователю (на пароле или PIN-коде), и на чем-нибудь, что есть у пользователя (устройстве проверки подлинности). Это средство предоставляет гораздо более надежный уровень пользовательской аутентификации, чем многократно используемые пароли. Новый одноразовый код пароля генерируется этим средством каждые 60 секунд, усложняя для всех остальных, кроме настоящего пользователя, ввод правильного токен-кода в любой момент времени. Чтобы получить доступ к своим ресурсам, пользователи в нужный момент объединяют свой секретный персональный идентификационный номер — Personal Identification Number (PIN) с токен-кодом, появляющимся на дисплее их устройства SecurID-аутентификации. В результате получается уникальный одноразовый пароль подтверждения личности пользователя.

14.6.2. RSA Identity and Access Management

RSA Identity and Access Management является продуктом, предоставляющим возможность управления идентификацией, обеспечением безопасности и контролем доступа для физических, виртуальных и облачных сред на основе управления доступом. Это позволяет доверенным лицам свободно и безопасно взаимодействовать с системами и пользоваться доступом к ним. В семействе RSA Identity and Access Management имеется два продукта: *RSA Access Manager* и *RSA Federated Identity Manager*. RSA Access Manager позволяет организациям централизованно управлять политиками аутентификации и авторизации для большого количества пользователей, веб-порталов и ресурсов приложений. Access Manager предоставляет

возможность беспрепятственного пользовательского доступа по принципу однократной регистрации — single sign-on (SSO) и для большей безопасности сохраняет содержимое личных данных. RSA Federated Identity Manager позволяет конечным пользователям после однократной идентификации и входа в систему сотрудничать с бизнес-партнерами, внештатными поставщиками услуг и партнерами по цепочке поставок или работать в различных офисах или агентствах.

14.6.3. RSA Data Protection Manager

RSA Data Protection Manager позволяет развернуть простое и вполне рентабельное управление шифрованием и обеспечением предприятия токенами и ключами. Семейство RSA Data Protection Manager состоит из двух продуктов: *Application Encryption and Tokenization* и *Enterprise Key Management*.

- **Application Encryption and Tokenization** из семейства RSA Data Protection Manager помогает добиться согласованности с правилами, связанными с личной идентификационной информацией (PII), путем быстрого внедрения шифрования и токенизации конфиденциальных данных и содействия предотвращению утраты этих данных. Это средство работает в месте создания данных, гарантируя тем самым, что данные при их передаче и сохранении будут находиться в зашифрованном виде.
- **Enterprise Key Management** является простым в использовании средством для ключей шифрования, которые применяются на уровне баз данных, файловых серверов и хранилищ. Оно разработано для упрощения развертывания шифрования на предприятии. Оно также помогает обеспечить достаточный уровень безопасности и полную доступность данных по мере их надобности на любой стадии их жизненного цикла.

14.6.4. VMware vShield

Семейство VMware vShield включает три продукта: *vShield App*, *vShield Edge* и *vShield Endpoint*.

VMware vShield App является межсетевым экраном на основе гипервизора, способным учитывать особенности приложений. Это средство защищает приложения в виртуализированной среде от сетевых угроз, предоставляя видимость сетевых коммуникаций и применяя дозированную политику доступа с использованием групп безопасности. VMware vShield App следит за активностью использования сети виртуальными машинами для определения и улучшения политик межсетевого экрана и обеспечения безопасности бизнес-процессов путем составления подробных отчетов о трафике приложений.

VMware vShield Edge предоставляет для виртуализированной среды комплексное обеспечение безопасности по периметру сети. Это средство развертывается в виде виртуального устройства и служит в качестве сетевого шлюза безопасности для всех хостов виртуализированной среды. Оно предоставляет множество услуг, включая службы межсетевого экрана, виртуальной закрытой сети (VPN) и протокола динамической конфигурации хоста — Dynamic Host Configuration Protocol (DHCP).

VMware vShield Endpoint состоит из особо защищенной специальной виртуальной машины, предназначенной для обеспечения безопасности, с установленной на ней антивирусной программой стороннего производителя. VMware vShield Endpoint упрощает и ускоряет развертывание программ, предназначенных для борьбы с вирусами и вредоносным кодом, поскольку обновление файлов самого антивируса и используемых им сигнатур производится только внутри особо защищенной специальной виртуальной машины. VMware vShield Endpoint повышает производительность виртуальных машин за счет того, что освобождает их от сканирования файлов и выполнения других антивирусных задач, поручая все это особо защищенной виртуальной машине. Данное средство предотвращает возникновение пиковых нагрузок и узких мест, связанных с одновременным сканированием файлов и обновлением своего кода и данных сразу несколькими антивирусами и программами для борьбы с вредоносным кодом. Оно также отвечает требованиям по аудиту, подробно регистрируя в журнале всю деятельность программ для борьбы с вирусами и вредоносным кодом.

Резюме

Продолжающееся расширение сетей хранения данных открывает все новые уязвимости ресурсов дата-центров и инфраструктуры хранения данных. Сетевое хранение данных на основе использования IP-сетей открывает для ресурсов хранилищ традиционные уязвимости этих сетей. Объединение данных также является фактором, повышающим степень возможного отрицательного воздействия на данные при взломе системы безопасности. В дополнение к этим проблемам обеспечения безопасности расширяются и усложняются различные нормативные требования. Администрации дата-центров приходится заниматься устранением угроз взлома системы безопасности как в самой организации, так и за ее пределами.

Создавая свои новые IT-модели, организации внедряют в повседневную практику виртуализацию и облачные вычисления. Но основной проблемой, препятствующей их ускоренному внедрению, является обеспечение безопасности данных. В облаке, по сравнению с обычным или виртуализированным дата-центром, имеется намного больше уязвимостей. Причина в том, что облачные ресурсы совместно используются множеством потребителей.

К тому же у потребителей весьма ограниченные возможности контроля облачных ресурсов. Поставщики и потребители облачных услуг сталкиваются в облачной среде с угрозами взлома системы безопасности.

В данной главе была подробно рассмотрена структура обеспечения безопасности хранилищ данных и предложены методы ослабления угроз, которые могут быть применены для обнаруженных в среде сетевого хранения данных. Кроме того, подробно рассмотрены архитектура системы безопасности и механизмы защиты SAN-, NAS- и IP-SAN-сред. Затем в ней были рассмотрены проблемы безопасности и меры, принимаемые для ее обеспечения в виртуализированной и облачной средах.

Решение вопросов безопасности стало неотъемлемым компонентом управления хранилищем данных и основным параметром, отслеживаемым в отношении всех компонентов дата-центра. В следующей главе основное внимание будет уделено вопросам управления инфраструктурой хранения данных.

УПРАЖНЕНИЯ

1. Исследуйте следующие механизмы обеспечения безопасности и объясните порядок их использования:
   - алгоритм MD-5;
   - алгоритм SHA-256;
   - RADIUS;
   - DH-CHAP.
2. Массив хранения данных автоматически обращается в центр поддержки при каждом обнаружении ошибки. Представитель производителя в центре поддержки может войти в систему процессора обслуживания в массиве хранения данных через Интернет, чтобы провести диагностику и устранить неполадки. Проанализируйте все факторы, угрожающие безопасности хранилища в данной среде, и предложите методы обеспечения безопасности, которые могут быть реализованы для ослабления любых вредоносных атак, проводимых через данный шлюз.
3. Разработайте перечень мероприятий по аудиту системы безопасности среды хранения данных, в которой используются технологии SAN, NAS и iSCSI. Объясните, как вы будете осуществлять аудит. Исследуйте возможные уязвимости в системе безопасности. Составьте их список и предложите механизмы контроля, которые должны быть реализованы для их устранения.
4. Расскажите о проблемах безопасности и мерах, предпринимаемых для их устранения в виртуализированной и облачной среде.
5. Проведите исследование технологии обеспечения безопасности путем применения многофакторной аутентификации и подготовьте по ней презентацию.

Глава 15
Управление инфраструктурой хранения данных

КЛЮЧЕВЫЕ ПОНЯТИЯ

- Мониторинг и предупреждения
- Стандарты платформ управления
- Стоимость ресурсов для начисления платежей
- Управление жизненным циклом информации
- Многоуровневое хранение данных

Беспрецедентный рост объема информации, резкое увеличение количества приложений, повышение сложности бизнес-процессов и выставление требований о круглосуточной доступности информации без учета выходных и праздничных дней (24×7) предъявили более высокие требования к инфраструктуре хранения данных.

Эффективное управление инфраструктурой является ключевым условием, позволяющим организациям справиться с этими задачами и обеспечить непрерывность бизнес-процессов.

Комплексное управление инфраструктурой хранения данных требует создания интеллектуального набора инструментальных средств и запуска надежных процессов, удовлетворяющих необходимому уровню обслуживания. Инструменты должны справляться с настройками производительности, контролем доступа, проведением централизованного аудита и соответствовать требованиям совместимости с используемым оборудованием. Они также должны обеспечивать слаженную работу и более эффективное использование имеющихся ресурсов, снижая тем самым потребности в дополнительных текущих капиталовложениях в инфраструктуру. Процесс управления определяет процедуры для эффективного подхода к различным операциям, например при устранении последствий чрезвычайных ситуаций, решении возникающих проблем и реагировании на изменение требований. Необходимо управлять не только отдельными компонентами, но и всей инфраструктурой в целом, поскольку работа всех компонентов носит взаимосвязанный характер.

Управление инфраструктурой хранения данных также придерживается таких стратегий, как стратегия *управления жизненным циклом*

информации — Information Lifecycle Management (ILM), которая помогает сэкономить средства, затрачиваемые на оборудование хранилища данных при соблюдении соответствующих требований к уровню предоставляемых услуг. ILM помогает управлять информацией на основе ее ценности для ведения бизнеса.

Управление инфраструктурой хранения данных требует выполнения различных действий, включая обеспечение доступности, достаточного объема ресурсов, приемлемой производительности и безопасности хранилища данных. Все эти действия взаимосвязаны и должны быть направлены на обеспечение максимальной отдачи от внесенных капиталовложений. С появлением технологий виртуализации в парадигму управления инфраструктурой хранения данных были внесены весьма существенные изменения.

В этой главе подробно рассматривается деятельность, связанная с мониторингом и управлением инфраструктурой хранения данных. В ней также дается описание общепринятых стандартов, используемых при разработке инструментов управления ресурсами хранилища. Кроме того, в главе подробно рассматривается управление жизненным циклом информации — Information Lifecycle Management (ILM), раскрываются преимущества такого подхода и рассматривается многоуровневое хранение данных.

15.1. Мониторинг инфраструктуры хранения данных

Мониторинг относится к одному из наиболее важных аспектов, влияющих на формирование основ управления ресурсами инфраструктуры хранения данных. Мониторинг предоставляет сведения о состоянии производительности и доступности различных компонентов. Он также позволяет администраторам выполнять важные управленческие действия. Кроме этого, мониторинг помогает проводить анализ использования и потребления различных ресурсов инфраструктуры хранения данных. Этот анализ обеспечивает планирование выделения объемов ресурсов, а также помогает прогнозировать выделение ресурсов и организовывать их оптимальное использование. Мониторингу подлежат также такие параметры среды инфраструктуры, как температурный режим оборудования и состояние его электропитания.

15.1.1. Отслеживаемые параметры

Компоненты инфраструктуры хранения данных должны отслеживаться на доступность, объем, производительность и безопасность. Под *доступностью* понимается способность компонента выполнять требуемую операцию в течение определенного периода времени. Под мониторингом доступности компонентов оборудования (например, порта, HBA-шины или дискового накопителя) или программных средств (например, базы данных) подразумевается проверка их состояния доступности путем просмотра оповещений,

генерируемых системой. Например, сбой порта может привести к выдаче целой серии предупреждений о его недоступности.

В инфраструктуре хранения данных используются избыточные компоненты, позволяющие избежать появления единых точек отказа. Сбой компонента может привести к простою, который, в свою очередь, может повлиять на доступность приложения или вызвать снижение производительности даже при сохранении доступности. Постоянный мониторинг ожидаемой доступности каждого компонента и отправка отчета о любых отклонениях помогают администратору выявить отказавшие компоненты и спланировать корректирующие действия для обеспечения требований соглашения об уровне предоставляемых услуг (SLA).

Под *объемом* подразумевается объем доступных ресурсов в инфраструктуре хранения данных. В качестве примеров мониторинга объемов можно привести определение свободного пространства в файловой системе или в RAID-группе, определение количества квот на объем почтового ящика, выделенных пользователю, или определение количества портов, доступных на коммутаторе. Несоответствующий объем приводит к снижению производительности или даже недоступности приложения или услуги. *Мониторинг объемов* обеспечивает непрерывную доступность данных и дает возможность увеличить объем с целью предотвращения простоев. Например, если в конкретной системе коммутации SAN-сети используется 90 % портов, это может стать показателем необходимости установки еще одного коммутатора, если в той же системе коммутации нужно установить большее количество массивов хранения данных и серверов. Для проведения анализа текущих тенденций при мониторинге объемов обычно используются аналитические инструменты. Выявление этих тенденций помогает понять объем будущих потребностей в ресурсах и установить сроки развертывания дополнительных объемов.

Мониторинг производительности позволяет оценивать эффективность работы различных компонентов инфраструктуры и помогает выявлять помехи. При мониторинге производительности работа оценивается и анализируется по таким показателям, как время отклика или возможность работать на конкретном, заранее определенном уровне. Кроме того, оценивается нагрузка на ресурсы, оказывающая влияние на их работу и время отклика. Оценка производительности представляет собой довольно сложную задачу, включающую оценку различных компонентов по нескольким взаимосвязанным параметрам. Примерами отслеживаемых параметров оценки производительности могут послужить количество операций ввода-вывода, выполняемых диском, время отклика приложения, нагрузка на сеть и нагрузка на центральный процессор сервера.

Мониторинг безопасности инфраструктуры хранения данных помогает отслеживать и предотвращать любой неавторизованный доступ, как случайный, так и злонамеренный. *Мониторинг безопасности* помогает отслеживать несанкционированные изменения конфигурации ресурсов инфраструктуры хранения данных, например отслеживать исходную конфигурацию

зонирования и оповещать обо всех ее изменениях в дальнейшем. Мониторинг безопасности также способен обнаруживать недоступность информации авторизованным пользователям, возникшую из-за взлома системы безопасности. Кроме того, постоянному мониторингу должна подвергаться физическая безопасность инфраструктуры хранения данных, для чего используются устройства чтения идентификационных карточек, биометрические сканеры или видеокамеры.

15.1.2. Отслеживаемые компоненты

Мониторингу на доступность, объем, производительность и безопасность в среде хранения данных должны подвергаться хосты, сети и хранилища данных. Эти компоненты могут быть как физическими, так и виртуальными.

Хосты

Доступность хоста зависит от состояния доступности компонентов оборудования и запущенных на нем программных процессов. Например, сбой сетевого адаптера может привести к недоступности хоста его пользователям. Механизмом, обеспечивающим высокую доступность в случае сбоя сервера, может стать кластеризация серверов.

Мониторинг степени использования пространства файловой системы хоста важен для обеспечения доступности достаточного для приложений объема хранения данных. Дефицит пространства файловой системы вредит доступности приложения. Мониторинг позволяет оценить скорость роста занятости файловой системы и предсказать момент, когда эта занятость достигнет 100 %. В соответствии с этим, чтобы избежать простоя приложения, администратор может заблаговременно расширить пространство файловой системы (вручную или автоматически). Использование технологий виртуального предоставления ресурсов позволяет эффективно управлять потребностями объема хранилища, но это управление сильно зависит от мониторинга объемов.

При мониторинге производительности хоста в основном используется проверка состояния использования различных ресурсов сервера, таких как центральный процессор и память. Например, если сервер, запустивший приложение, постоянно находится под 80 %-ной нагрузкой, это означает, что у сервера может не хватить вычислительной мощности, а это, в свою очередь, может привести к снижению производительности и увеличению времени отклика. Для устранения этой проблемы администраторы должны выполнить ряд действий, например, обновить процессоры, или увеличить их количество, или передать нагрузку другим серверам. Чтобы удовлетворить потребность в повышении производительности в виртуализированной среде, виртуальным машинам могут быть в динамическом режиме выделены из общего пула дополнительные центральные процессоры и дополнительные объемы памяти.

При мониторинге системы безопасности на серверах используется отслеживание неудавшихся попыток входов в систему и выполнения неавторизованных приложений или программных процессов. Профилактические меры против неавторизованного доступа к серверам основаны на выявлении различных угроз. Например, администратор может заблокировать пользовательский доступ, если зафиксирует сразу нескольких неудачных попыток входа в систему.

Сеть хранения данных

Сети хранения данных нуждаются в мониторинге с целью обеспечения непрерывной связи между сервером и массивом хранения данных. Непрерывность доступа к данным по сети хранения зависит от доступности физических и логических компонентов сети хранения данных. Физические компоненты этой сети включают коммутаторы, порты и кабели. Логические компоненты включают такие построения, как зоны. Любые сбои физических или логических компонентов приводят к недоступности данных. Например, ошибки в зонировании, такие как указание неверного WWN-имени порта, приводят к сбою доступа к этому порту, который потенциально может препятствовать доступу хоста к его хранилищу.

При мониторинге объемов в сетях хранения данных используется отслеживание количества доступных в системе коммутации портов, загруженности каналов между коммутаторами или загруженности отдельных портов и каждого имеющегося в системе коммутации соединительного устройства. Мониторинг объемов предоставляет все необходимые сведения для перспективного планирования и оптимизации использования ресурсов системы коммутации.

Мониторинг производительности сети хранения данных позволяет оценить производительность отдельных компонентов и способствует выявлению в сети узких мест. Например, мониторинг производительности порта включает оценку метрик использования ссылок на получение или передачу данных, из которой можно сделать вывод о загруженности порта коммутатора. Интенсивно используемые порты могут вызвать скопление в очереди на сервере операций ввода-вывода, что отрицательно скажется на производительности.

Что касается IP-сетей, мониторинг производительности включает отслеживание сетевых задержек, потерь пакетов, степени использования ширины полосы пропускания для ввода-вывода, ошибок сети, уровня повторной передачи пакетов и возникновения конфликтных ситуаций.

Мониторинг безопасности сети предоставляет информацию о любом несанкционированном изменении конфигурации системы коммутации, например изменении политик зонирования, что может повлиять на безопасность данных. В журнал постоянно должны вноситься записи и на их основе постоянно отслеживаться все случаи неудачного входа в систему и неавторизованного доступа к коммутаторам с целью внесения каких-либо административных изменений.

Хранилище данных

Доступность массива хранения данных должна отслеживаться по его аппаратным компонентам и различным процессам. Конфигурация массива хранения данных настраивается в основном на использование избыточного количества компонентов, и поэтому сбой одного из компонентов обычно не влияет на доступность всего массива. Но сбой любого из процессов в массиве хранения данных может нарушить ход выполнения бизнес-операций или поставить их выполнение под угрозу. Например, сбой задачи репликации данных оказывает отрицательное влияние на возможность восстановления работы после серьезной аварии. У некоторых массивов хранения данных в случае сбоя процесса имеется возможность отправки сообщений в центр поддержки производителя массива, известный как *звонок домой*.

Мониторинг объемов массива хранения данных позволяет администраторам заранее реагировать на возникновение потребностей в объеме хранилища на основе тенденций роста использования и потребления пространства хранения данных. Информация о ненастроенном и нераспределенном пространстве хранения данных позволяет администратору принимать решения о выделении пространства хранения данных еще одному серверу.

Массив хранения данных может отслеживаться с помощью ряда показателей производительности, таких как степень использования различных компонентов массива хранения данных, время отклика на запросы ввода-вывода и степень использования кэша. Перегруженный компонент массива хранения данных может стать причиной снижения производительности.

Массив хранения данных обычно является общим ресурсом, который может подвергаться различным угрозам безопасности. Мониторинг безопасности позволяет отслеживать попытки несанкционированного изменения конфигурации массива хранения данных и обеспечивает возможность допуска к массиву только авторизованных пользователей.

15.1.3. Примеры мониторинга

Для активного отслеживания всех параметров своих компонентов инфраструктура хранения данных требует реализации сквозных решений. Обеспечение бесперебойного обслуживания критически важных активов невозможно без раннего обнаружения и упреждающего оповещения. Кроме того, инструментарий мониторинга должен анализировать влияние сбоев и выявлять основную причину тех или иных неблагоприятных симптомов.

Мониторинг доступности

Сбой любого из компонентов может отрицательно повлиять на доступность одного или нескольких компонентов по причине их взаимосвязанности и зависимости друг от друга. Рассмотрим реализацию инфраструктуры хранения данных, в которой имеется три сервера: *H1*, *H2* и *H3*. Как показано

на рис. 15.1, в конфигурацию каждого из серверов входят два HBA-адаптера, каждый из которых подключен к производственному массиву хранения данных через два коммутатора, *SW1* и *SW2*. На массиве хранения данных всеми серверами совместно используются два порта хранилища данных, и на всех серверах установлено программное средство, допускающее ввод-вывод по нескольким маршрутам.

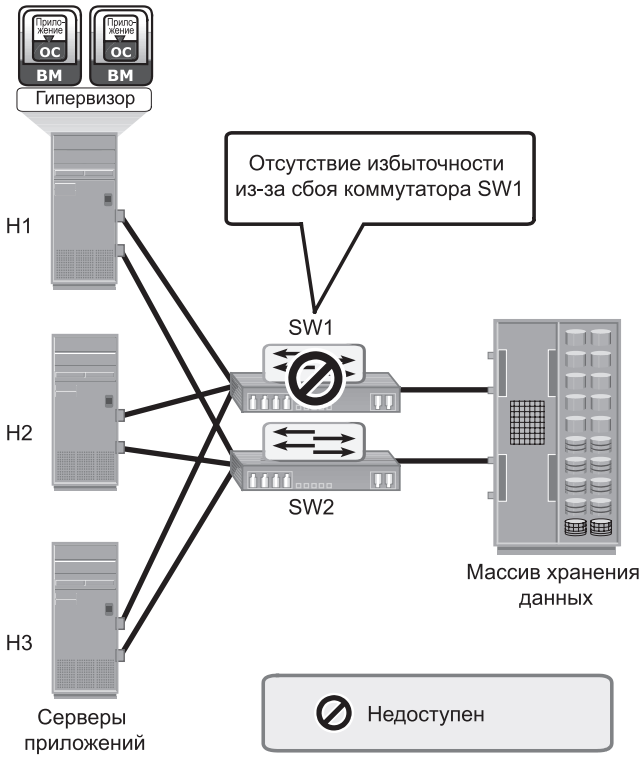


Рис. 15.1. Сбой коммутатора в инфраструктуре хранения данных

Если один из коммутаторов (*SW1*) откажет, программа направления данных по нескольким маршрутам инициирует обход отказавшего маршрута и все серверы продолжат получать доступ к данным через другой коммутатор, *SW2*. Но из-за отсутствия избыточного коммутатора отказ второго коммутатора может повлечь за собой недоступность массива хранения данных. Мониторинг доступности позволяет обнаружить сбой коммутатора и помогает администраторам провести мероприятия по устранению сбоя до возникновения второго отказа.

В большинстве случаев администратор получает оповещения о тревожных симптомах в отношении сбойного компонента и может принять меры до того, как на нем произойдет какой-нибудь отказ.

Мониторинг объемов

В сценарии, показанном на рис. 15.2, серверы *H1*, *H2* и *H3* подключены к производственному массиву хранения данных через два коммутатора, *SW1* и *SW2*. Каждому из этих серверов выделено хранилище из состава массива хранения данных. Когда в этой конфигурации развертывается новый сервер, работающие на нем приложения нуждаются в выделении объемов в хранилище данных, входящем в производственный массив хранения данных. Мониторинг имеющихся в массиве доступных объемов (нераспределенных и допускающих конфигурирование) способствует оперативному принятию решения относительно того, может ли массив предоставить требуемые объемы новому серверу. Кроме того, мониторинг доступного количества портов на *SW1* и *SW2* способствует принятию решения о возможности подключения нового сервера к коммутаторам.

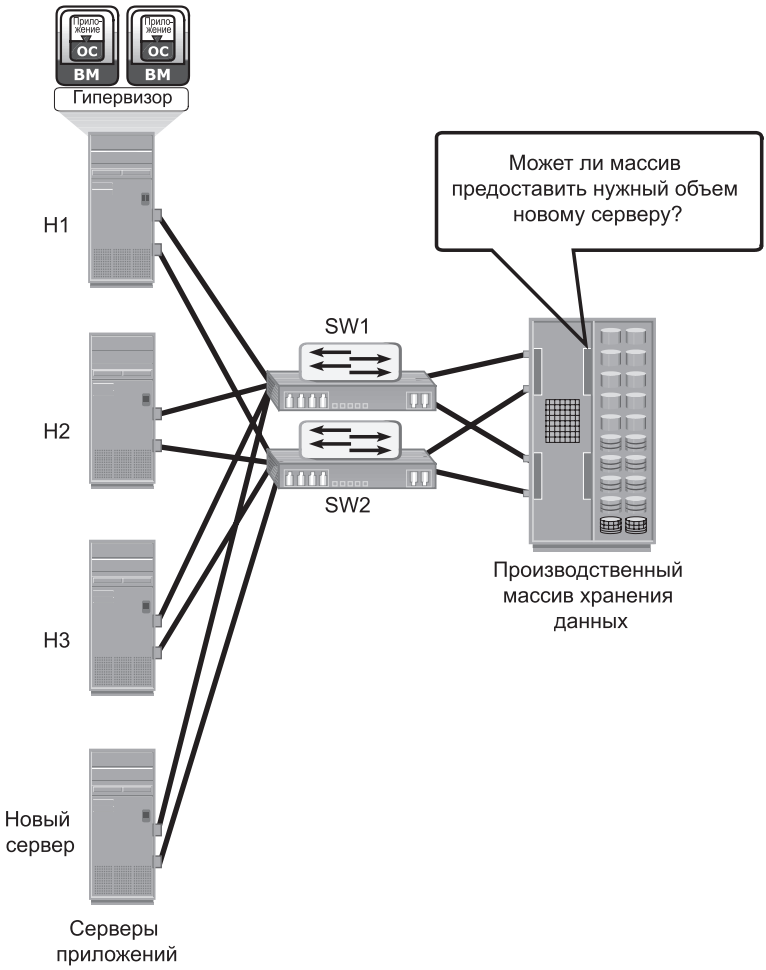


Рис. 15.2. Мониторинг объемов массива хранения данных

В следующем примере показано, насколько важен мониторинг объемов файловой системы на файловых серверах. На рис. 15.3, *а*, показана среда файловой системы при ее предельном заполнении, приводящем к простоям приложения, в случае если мониторинг объемов не ведется. Мониторинг может быть настроен на выдачу сообщения в тот момент, когда в расходе объемов файловой системы были пройдены определенные пороговые значения. Например, когда файловая система достигнет 66 % своего объема, выдается предупредительное сообщение, а при достижении 80 % — тревожный сигнал (рис. 15.3, *б*). Это позволяет администратору принять соответствующие меры по расширению файловой системы еще до исчерпания всего объема. Интенсивный мониторинг файловой системы может воспрепятствовать простоям приложения из-за нехватки объемов файловой системы.

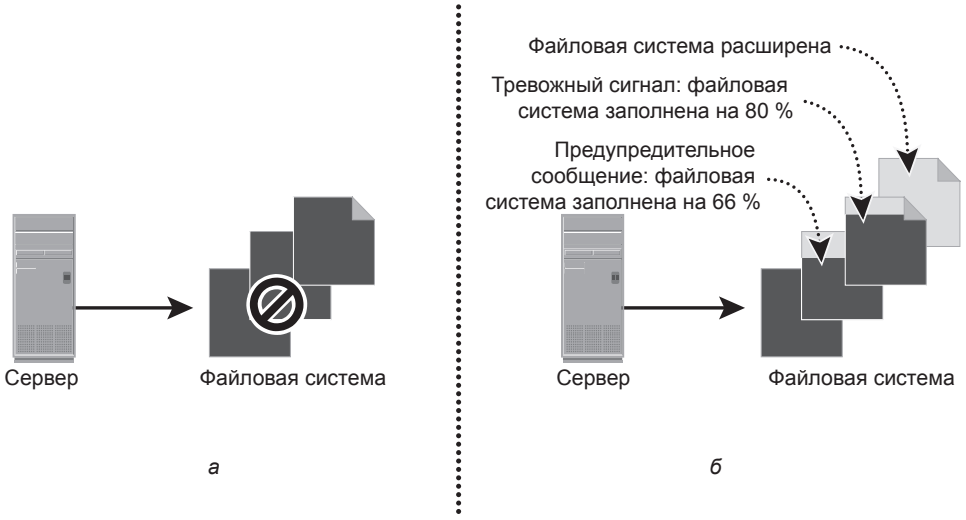


Рис. 15.3. Мониторинг объемов файловой системы сервера: *а* — мониторинг не ведется; *б* — ведется мониторинг файловой системы

Мониторинг производительности

В примере на рис. 15.4 показано, насколько важно вести мониторинг производительности массива хранения данных. В этом примере серверы *H1*, *H2* и *H3* (у каждого из которых по два HBA-адаптера) подключены к массиву хранения данных через коммутаторы *SW1* и *SW2*. Для доступа к LUN-устройствам все три сервера совместно используют одни и те же порты массива хранения данных. Для совместного использования тех же самых портов массива хранения данных, которые уже используются серверами *H1*, *H2* и *H3*, развертывается новый сервер, выполняющий приложение с высокой рабочей нагрузкой.

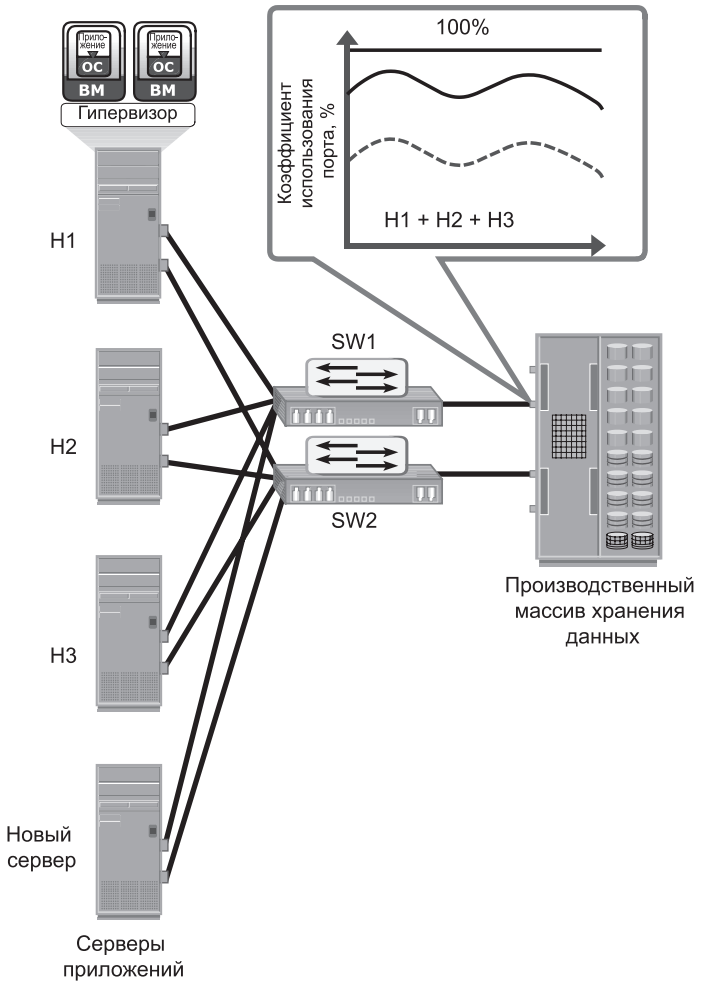


Рис. 15.4. Мониторинг степени использования порта массива данных

Мониторинг степени использования портов массива дает гарантии того, что новый сервер не окажет негативного влияния на производительность других серверов. В данном примере коэффициент использования общего порта хранилища показан на графике сплошной и пунктирной линиями. Если коэффициент использования порта до развертывания нового сервера был близок к 100 %, то развертывать этот сервер не рекомендуется, поскольку это может отрицательно повлиять на производительность остальных серверов. Но если коэффициент использования порта до развертывания нового сервера близок к тому, что показан пунктирной линией, у порта еще есть запас ширины полосы пропускания для добавления нового сервера.

На большинстве серверов предлагаются инструментальные средства, позволяющие проводить мониторинг коэффициента использования (загруженности) центрального процессора. Например, коэффициент загруженности центрального процессора и памяти можно увидеть в Диспетчере задач Windows (рис. 15.5). Но для мониторинга сотен серверов в среде дата-центра подобные средства не подходят. Там нужны интеллектуальные средства мониторинга производительности, способные отслеживать сразу множество серверов.

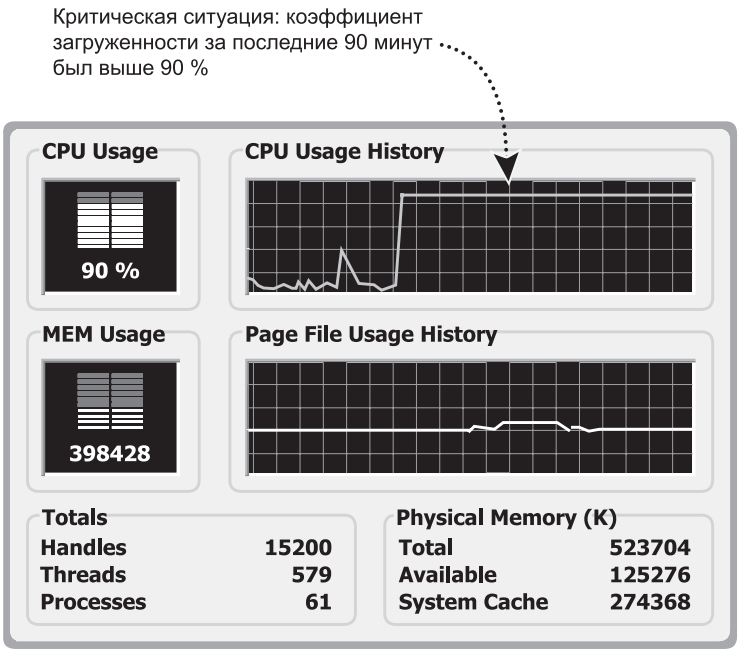


Рис. 15.5. Мониторинг коэффициента загруженности центрального процессора и памяти сервера

Мониторинг безопасности

В примере на рис. 15.6 показано, насколько важно вести мониторинг безопасности в массиве хранения данных.

В данном примере массив хранения данных совместно используется двумя рабочими группами, *WG1* и *WG2*. Данные рабочей группы *WG1* не должны быть доступны рабочей группе *WG2* и наоборот. Пользователь из группы *WG1* может попытаться создать локальную реплику данных, принадлежащих группе *WG2*. Если это действие не отслеживается или не записывается, то заметить подобную уязвимость информационной безопасности довольно трудно. Если же это действие отслеживается, может быть отправлено

предупредительное сообщение с предложением провести корректирующее действие или, по крайней мере, может быть предоставлена возможность обнаружения этого действия в ходе проведения очередной операции аудита.

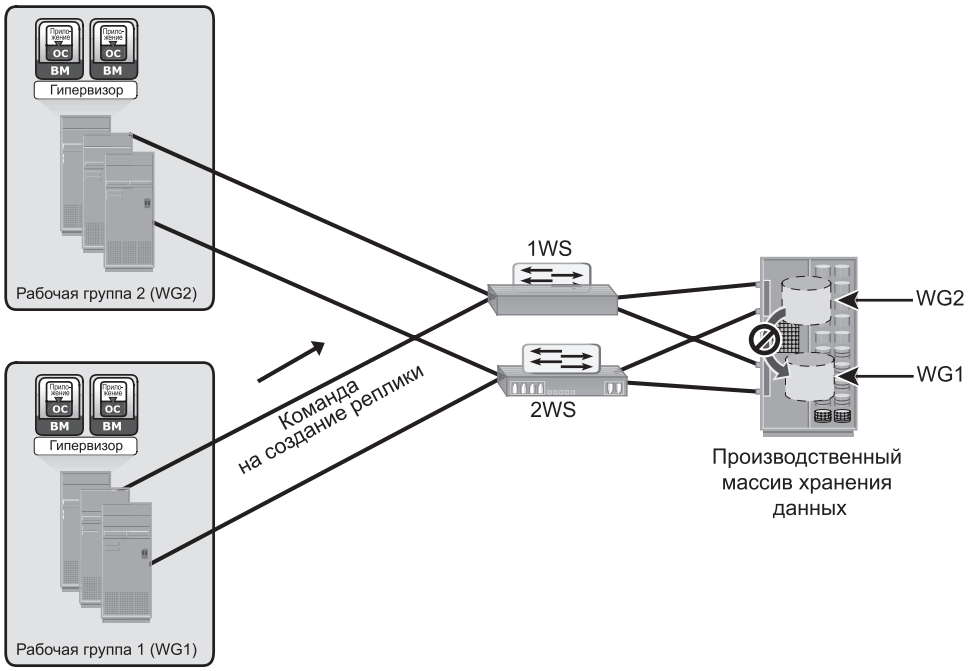


Рис. 15.6. Мониторинг безопасности в массиве хранения данных

В примере мониторинга безопасности хоста отслеживаются попытки зарегистрироваться на хосте. Имя пользователя проходит авторизацию, если верны введенные при входе идентификатор имени пользователя и пароль, в противном случае попытка зарегистрироваться терпит неудачу. Такие неудачи могут быть случайными (из-за какой-нибудь опечатки) или умышленными, допущенными при попытке получения доступа к серверу. Многие серверы позволяют, как правило, предпринимать определенное число последовательных неудачных попыток регистрации, запрещая все остальные попытки при исчерпании установленного лимита. В среде, где ведется мониторинг, регистрационная информация записывается в файл системного журнала, и при трех последовательных неудачных попытках выдается сообщение, предупреждающее о возможной угрозе безопасности.

15.1.4. Предупреждения

Предупреждения о событиях являются неотъемлемой частью мониторинга. Предупреждения держат администратора в курсе состояния различных

компонентов и процессов, например, они выдаются при возникновении сбоев электропитания, дисков, памяти или коммутаторов, способных повлиять на доступность услуг и потребовать неотложного внимания администратора. При возникновении других условий, например при достижении файловой системой порога заполнения или возникновении несущественной ошибки носителя на дисковом накопителе, предусматривается вывод предупреждений, которые также должны привлечь внимание администратора.

Средства проведения мониторинга позволяют администраторам назначать различные степени опасности на основе серьезности негативного воздействия на работу системы тех условий, при которых выдается предупреждение. Когда создаются условия с определенной степенью серьезности, администратору отправляется предупреждение, запускается сценарий или выдается заявка с описанием проблемы (incident ticket), побуждающая к разрешению ситуации. Предупреждения можно классифицировать в диапазоне от информационных сообщений до срочных сигналов о серьезных сбоях. *Информационные предупреждения* предоставляют полезную информацию, но при этом не требуют вмешательства администратора. Примерами информационных предупреждений могут быть сообщения о создании зон или LUN-устройств. *Тревожные предупреждения* требуют внимания администратора, в них содержится настораживающая информация о событии, не влияющем на доступность данных. Например, если предупреждение показывает, что число незначительных ошибок на носителе данных дискового накопителя подошло к определенному порогу, администратор может принять решение о замене диска. *Срочные сигналы о серьезных сбоях* требуют незамедлительной реакции, поскольку создавшиеся условия могут отрицательно повлиять на общую производительность, безопасность или доступность массива хранения данных. Например, если произошел отказ диска, администратор должен обеспечить его быструю замену.

Постоянный мониторинг с автоматической выдачей предупреждений позволяет администраторам оперативно реагировать на возникающие сбои. Предупреждения предоставляют информацию, помогающую администраторам определить приоритетность своих реакций на происходящие события.

15.2. Действия по управлению инфраструктурой хранения данных

Темпы роста объемов информации, резкое увеличение количества приложений, наличие разнородной инфраструктуры и обязательных для выполнения требований, касающихся уровней обслуживания, существенно усложнили управление инфраструктурами хранения данных. Но с появлением виртуализации хранилищ и других технологий, таких как избавление от дубликатов данных и сжатие информации, виртуальное предоставление ресурсов, обеспечение интегрированного доступа к хранилищам данных и многоуровневая

организация памяти позволили администраторам повысить эффективность управления ресурсами хранения данных.

Основная деятельность по управлению инфраструктурой хранения данных, проводимая в дата-центрах, может быть в целом разделена на управление доступностью, управление объемами, управление производительностью, управление безопасностью и составление отчетов.

15.2.1. Управление доступностью

Важнейшей задачей управления доступностью является выбор верного направления на основе определенных уровней обеспечения доступности. *Управление доступностью* включает решение всех вопросов, связанных с доступностью компонентов или служб, с целью обеспечения требуемых уровней обслуживания. Основной деятельностью при управлении доступностью является обеспечение избыточности на всех уровнях, включая компоненты, данные и даже производственные объекты. Например, когда сервер развернут с целью поддержки жизненно важных бизнес-функций, он требует высокого уровня доступности. Как правило, выполнение этого требования обеспечивается развертыванием двух и более HBA-адаптеров, установкой программ, обеспечивающих работу по нескольким маршрутам, и созданием кластера серверов. Сервер должен быть подключен к массиву хранения данных с помощью как минимум двух независимых систем коммутации и коммутаторов, имеющих встроенную избыточность. Кроме того, у массивов хранения данных также должна быть встроенная избыточность различных компонентов, которые должны поддерживать как локальную, так и удаленную репликацию.

15.2.2. Управление объемами

Целью *управления объемами* является обеспечение соответствующей доступности ресурсов на основе требований, выдвигаемых в отношении уровня обслуживания. Управление объемами касается также оптимизации объемов, основанной на стоимости оборудования и будущих потребностях. Управление объемами предоставляет анализ объемов, который постоянно сравнивает выделенный объем памяти хранилища с прогнозируемыми потребностями в его выделении. При управлении объемами проводится также анализ тенденций на основе скорости роста потребления объемов, который должен помочь в рациональном определении сроков приобретения и развертывания дополнительных объемов хранения данных. Примером управления объемами является подготовка объемов хранения данных к работе. Под этой подготовкой понимается деятельность по созданию наборов RAID-массивов и LUN-устройств и их распределению между хостами. Еще одним примером управления объемами является соблюдение квот объемов для пользователей. Предоставление фиксированного количества пользовательских квот удерживает пользователей от превышения выделенных объемов.

Технологии избавления от дубликатов и сжатия данных уменьшают объем данных, предназначенных для резервного копирования, и сокращают тем самым объемы хранения, подлежащие управлению.

15.2.3. Управление производительностью

Управление производительностью обеспечивает оптимальную эффективность работы всех компонентов. Анализ производительности играет важную роль в определении производительности компонентов инфраструктуры хранения данных. Этот анализ предоставляет информацию о том, соответствует ли компонент ожидаемым уровням производительности.

Развертывание в существующей инфраструктуре хранения данных нового приложения или сервера требует выполнения ряда действий, касающихся управления производительностью. Каждый компонент должен быть проверен на соответствие своих возможностей по производительности определенным уровням обслуживания. Например, для получения ожидаемых уровней производительности нужна тонкая настройка на сервере таких параметров, как конфигурация томов, схема базы данных или приложения, конфигурация нескольких HBA-адаптеров и интеллектуальной программы, допускающей использование нескольких маршрутов. К числу задач управления производительностью SAN-сети относятся проектирование и реализация достаточного количества линий связи между коммутаторами (ISL-каналов) в системе коммутации с достаточной шириной пропускания для поддержки требуемых уровней производительности. Задачи настройки конфигурации массивов хранения данных при решении проблем сквозной производительности включают выбор соответствующего RAID-типа, схемы LUN-устройств, интерфейсных портов, внутренних портов и конфигурации кэш-памяти.

15.2.4. Управление безопасностью

Основной целью *управления безопасностью* является обеспечение конфиденциальности, целостности и доступности информации как в виртуализированной, так и в невиртуализированной среде. Управление безопасностью предотвращает неавторизованный доступ и внесение изменений в конфигурацию компонентов инфраструктуры хранения данных. Например, при развертывании приложения или сервера задачами управления безопасностью являются управление пользовательскими учетными записями и политиками доступа, проводящими авторизацию пользователей на те или иные действия на основе отведенных этим пользователям ролей. Задачи управления безопасностью в SAN-среде включают конфигурацию зонирования для ограничения доступа к определенным портам массива хранения данных со стороны неавторизованных HBA-адаптеров. По аналогии с этим задачи управления безопасностью в массиве хранения данных включают LUN-маскирование, разрешающее доступ хостов только к тем LUN-устройствам, которые для них предназначены.

15.2.5. Составление отчетов

Составление отчетов об инфраструктуре хранилища данных заключается в отслеживании состояния и сборе информации о различных компонентах и процессах. Эта информация накапливается для составления отчетов, позволяющих проводить анализ тенденций, вести планирование наращивания объемов, оценивать стоимость ресурсов для начисления платежей и производительность. Отчеты для планирования наращивания объемов содержат текущую и историческую информацию о степени использования хранилища, файловой системы, табличного пространства базы данных, портов и т. д. В отчеты об управлении конфигурацией и активами включаются подробности, касающиеся распределения устройств, создания локальных или удаленных реплик и конфигурации системы коммутации. В этом отчете также дается перечень всего оборудования, где указываются дата приобретения, состояние аренды и обслуживания. Отчеты о стоимости ресурсов для начисления платежей содержат информацию о выделении или коэффициенте использования компонентов инфраструктуры хранения данных различными подразделениями или группами пользователей. Отчеты о производительности содержат подробности, касающиеся производительности различных компонентов инфраструктуры хранения данных.

15.2.6. Управление инфраструктурой хранения данных в виртуализированной среде

Технология виртуализации существенно повлияла на сложность управления инфраструктурой хранения данных. Фактически, виртуализация на всех уровнях IT-инфраструктуры была введена в практику в немалой степени под впечатлением от возможностей осуществления гибкого и простого управления.

Виртуализация хранилищ данных открыла возможности динамической миграции данных и расширения томов хранилищ. Благодаря возможности динамического расширения тома хранилища, чтобы отвечать требованиям по увеличению объема и повышению производительности, способны расширяться незаметно для пользователя. Поскольку виртуализация разрушила связь между томами хранилища, представляемыми хостам, и их физическим расположением, данные могут мигрировать как внутри дата-центра, так и за его пределы, не вызывая при этом никаких простоев. Это существенно облегчило стоящие перед администраторами задачи по реконфигурации физической среды.

Еще одним средством, удешевившим и упростившим управление инфраструктурой, стало выделение виртуального хранилища. Когда выделение идет обычным порядком, в выделяемый объем хранилища авансом закладывается возможный будущий рост потребностей. Поскольку рост происходит неравномерно, некоторые пользователи или приложения могут испытывать дефицит объема, а у других в то же время может быть его избыток, остающийся невостребованным. С этой проблемой может справиться виртуальное

выделение объемов, которое уменьшает груз проблем управления объемами. При виртуальном выделении хранилище назначается хостам из общего пула по мере надобности. Это повышает коэффициент использования объема и упрощает управление объемами.

МНОГОПОЛЬЗОВАТЕЛЬСКАЯ СРЕДА ХРАНЕНИЯ ДАННЫХ

Когда некоторые ресурсы, предоставляемые одним и тем же арендодателем (поставщиком ресурсов), могут совместно использоваться сразу несколькими арендаторами, это называется многопользовательской средой хранения данных, или мультитенантностью (multitenancy). Двумя самыми распространенными примерами являются использование одного и того же серверного оборудования сразу несколькими виртуальными машинами благодаря применению запущенного на сервере гипервизора, а также использование одной и той же платформы хранения данных сразу несколькими пользовательскими приложениями. Мультитенантность — понятие не новое, но ставшее благодаря росту популярности развертывания облачных вычислений темой множества дискуссий, поскольку совместно используемая инфраструктура является основным компонентом любой облачной стратегии.

Как и при любом совместном обслуживании, основными проблемами в мультитенантной среде хранения данных являются гарантии предоставления соответствующих уровней безопасности и обслуживания. Безопасная мультитенантность означает, что никакой арендатор не может воспользоваться данными другого арендатора. Для ее достижения при развертывании хранилища данных следует придерживаться четырех основных принципов мультитенантности.

- Безопасное разделение. Оно позволяет разделить маршруты данных разных арендаторов в мультитенантной среде. На уровне хранилища эта основа может быть поделена на четыре базовых требования: разделение данных, находящихся в состоянии покоя, разделение адресного пространства, разделение аутентификации и службы имен и разделение доступа к данным.
- Предоставление гарантий обслуживания. Предоставление постоянных и надежных уровней обслуживания является неотъемлемой составляющей мультитенантности хранилища. Гарантии обслуживания играют весьма важную роль в предоставлении уровней обслуживания, которые могут быть уникальными для каждого арендатора.
- Доступность. Высокая доступность гарантируется устойчивой архитектурой, предоставляющей высокий уровень отказоустойчивости и избыточности. В условиях совместного использования архитектуры хранения данных сразу несколькими арендаторами роль доступности возрастает еще больше, поскольку сфера влияния любого простоя существенно расширяется.
- Управление. Здесь берутся в расчет условия, позволяющие арендодателю управлять основной инфраструктурой, делегируя при этом арендаторам ответственность за управление теми ресурсами, с которыми они ежедневно работают. Такая концепция известна как соблюдение баланса возможностей внутреннего контроля со стороны поставщика (арендодателя) с возможностями внутреннего контроля со стороны арендатора.

Виртуализация также положительно влияет на эффективность управления сетью. VSAN- и VLAN-сети упрощают работу администратора путем логической изоляции различных сетей с помощью инструментальных средств управления, а не путем их физического разделения. В одной и той же физической сети могут быть созданы совершенно разные виртуальные сети, а переконфигурация узлов может быть произведена быстро и без каких-либо физических изменений. Заодно решается и ряд проблем безопасности, с которыми можно столкнуться в обычной сетевой среде.

На стороне хоста виртуализация вычислительного устройства сделала развертывание хоста, его реконфигурацию и миграцию проще, чем в физической среде. Виртуализация вычислительных устройств, приложений и памяти не только улучшила систему их выделения, но и внесла значительный вклад в доступность ресурсов.

15.2.7. Примеры управления хранилищами данных

В данном разделе приводятся примеры различной деятельности по управлению хранилищем данных.

Пример 1. Выделение хранилища новому серверу или хосту

Рассмотрим развертывание нового RDBMS-сервера в существующей инфраструктуре хранения данных, не подвергавшейся виртуализации. Первое, что должен сделать администратор при управлении хранилищем данных, это установить на сервере HBA-адаптеры и драйверы устройств и настроить конфигурацию этих адаптеров, и только после этого можно будет заняться физическим подключением к SAN-сети. Дополнительно на сервере может быть установлено программное обеспечение, позволяющее осуществлять ввод-вывод по нескольким путям, для которого могут понадобиться дополнительные настройки. Кроме того, к SAN-сети должны быть подключены порты массива хранения данных.

На следующем этапе администратору нужно выполнить зонирование на SAN-коммутаторах, чтобы позволить новому серверу обращаться к портам массива хранения данных через свои HBA-адаптеры. Чтобы обеспечить избыточные маршруты между сервером и массивом хранения данных, HBA-адаптеры нового сервера должны быть подключены к разным коммутаторам и зонированы с разными портами массива.

Кроме того, администратору нужно настроить LUN-устройства массива и назначить эти LUN-устройства интерфейсным портам массива хранения данных. В дополнение к этому на массиве хранения данных выполняется настройка маскирования LUN-устройств, разрешающая доступ к LUN-устройствам только со стороны конкретного сервера.

Затем сервер обнаруживает выделенные ему LUN-устройства либо с помощью процесса повторного сканирования шины, либо в иных случаях в зависимости от типа установленной на нем операционной системы путем перезагрузки сервера. Для настройки конфигурации локальных томов

и файловых систем на хост-машине может использоваться диспетчер томов. Количество создаваемых локальных томов или файловых систем зависит от ожидаемого способа использования базой данных или приложением хранилища данных. Задачей администратора также является установка базы данных или приложения на созданные логические тома или файловые системы.

На заключительном этапе базе данных или приложению дается возможность использования пространства новой файловой системы. Действия, выполняемые на сервере, в SAN-сети и в массиве данных для выделения хранилища новому серверу, показаны на рис. 15.7.

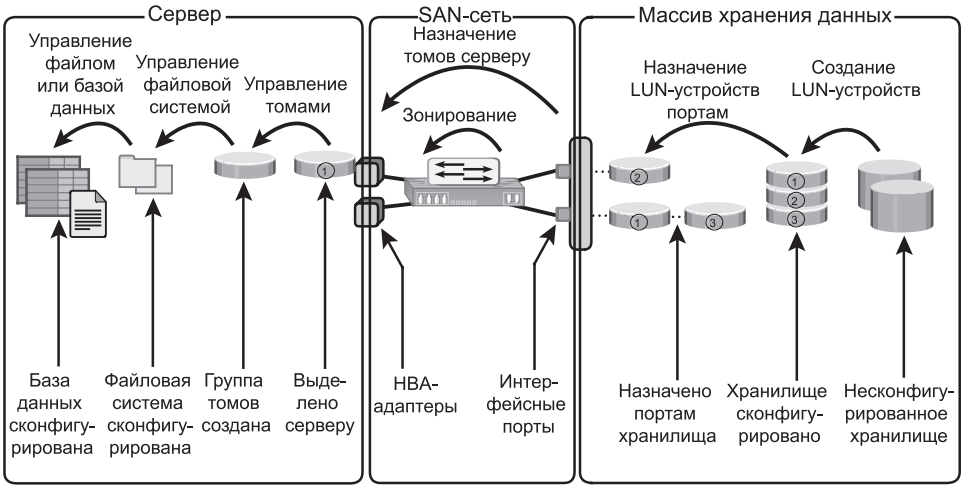


Рис. 15.7. Задачи, связанные с выделением хранилища

В виртуализированной среде предоставление хранилища виртуальной машине, на которой работает RDBMS-система, требует решения иных административных задач.

Так же, как и в невиртуализированной среде, необходимо по SAN-сети установить соединение между физическим сервером, на базе которого создаются виртуальные машины, и массивом хранения данных. На уровне SAN-сети может быть настроена VSAN-сеть, предназначенная для передачи данных между физическим сервером и массивом хранения данных. Эта VSAN-сеть изолирует трафик хранилища от любого другого трафика, имеющегося в SAN-сети. Кроме того, администратор может настроить внутри VSAN-сети зонирование.

На стороне хранилища администраторам нужно создать из общего пула хранения данных тонкие LUN-устройства и назначить их интерфейсным портам массива хранения данных. Точно так же, как и в физической среде, в массиве хранения данных нужно провести маскирование LUN-устройств.

На стороне физического сервера гипервизор обнаруживает назначенные LUN-устройства. Затем гипервизор создает логический том и файловую систему для хранения файлов виртуальных машин и управления ими. После этого администратор создает виртуальную машину и устанавливает на нее операционную систему и систему управления реляционными базами данных — RDBMS. При создании виртуальной машины гипервизор создает в своей собственной файловой системе файл виртуального диска и другие файлы виртуальной машины. Файлы виртуального диска появляются в виртуальной машине в виде SCSI-диска и используются для хранения данных RDBMS. Как вариант, гипервизор дает возможность виртуально создать тонкий виртуальный диск и назначить его виртуальной машине. У гипервизоров, как правило, имеются собственные возможности по использованию нескольких путей. Дополнительно на гипервизор может быть установлено программное средство, позволяющее направлять трафик по нескольким путям, поставленное сторонним производителем.

Пример 2. Управление пространством файловой системы

Во избежание дефицита пространства файловой системы администраторам нужно выполнить ряд задач для выгрузки данных из существующей файловой системы. Это действие включает также удаление нежелательных файлов или архивирование данных, к которым довольно давно никто не обращался.

Как вариант, администратор может расширить файловую систему, чтобы увеличить ее размер и избежать тем самым простоя приложения. Возможность динамического расширения файловой системы или логического тома зависит от используемой операционной системы или логического диспетчера томов (LVM). Этапы и факторы, учитываемые при расширении файловых систем, показаны в виде блок-схемы на рис. 15.8.

Пример 3. Отчет о стоимости ресурсов для начисления платежей

В данном примере исследуются задачи управления инфраструктурой хранения данных, которые необходимо выполнить для создания отчета о стоимости ресурсов для начисления платежей.

На рис. 15.9 показана конфигурация, развернутая в инфраструктуре хранения данных. Имеется три сервера, у каждого из которых по два HBA-адаптера и каждый из которых подключен к массиву хранения данных через два коммутатора, *SW1* и *SW2*. На каждом сервере запущены приложения отдельных подразделений. Для создания локальных и удаленных реплик применяется технология на основе использования массива хранения данных. Производственные устройства помечены буквой *A*, устройство локальной реплики — буквой *B*, устройство удаленной реплики — буквой *C*.

В отчете при анализе стоимости ресурсов для каждого подразделения документируется точный объем ресурсов хранилища, используемый каждым приложением. Если расчетная единица биллинга основана на полном объеме хранилища (используемый объем плюс предоставленная защита данных),

сконфигурированного для приложения, используемого подразделением, то для каждого приложения должен быть указан точный объем общего сконфигурированного пространства. Пример отчета показан на рис. 15.9. В нем имеются сведения для двух подразделений, Payroll_1 и Engineering_1.

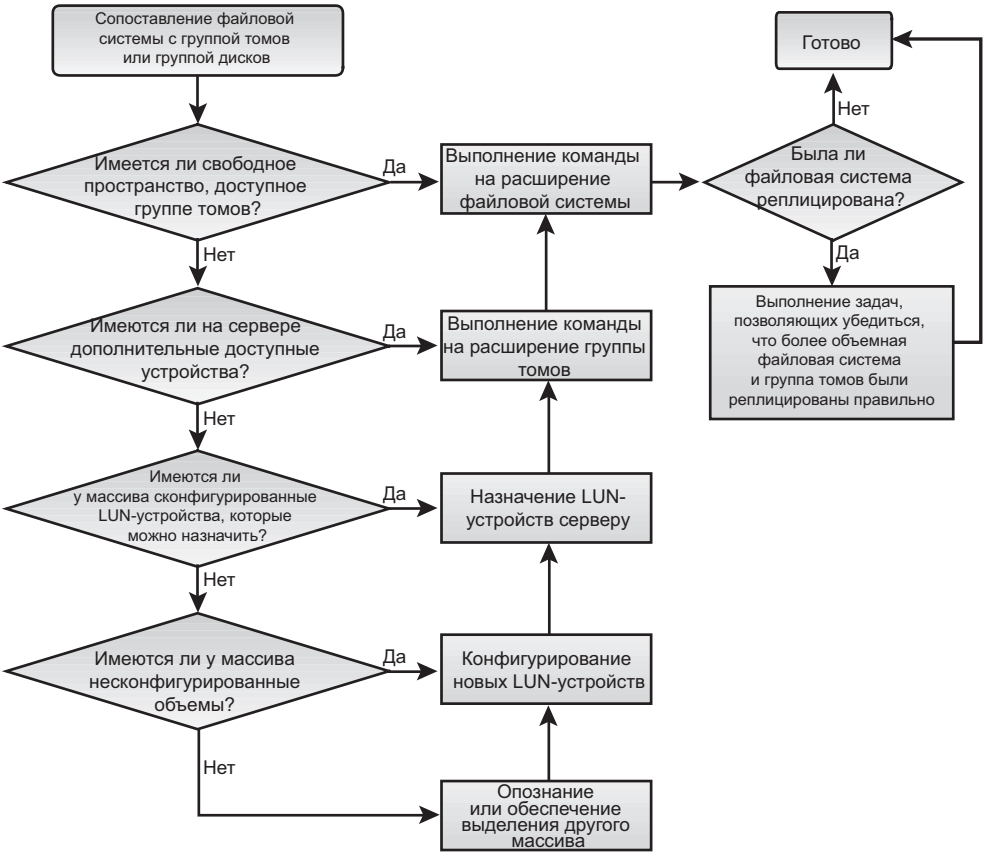
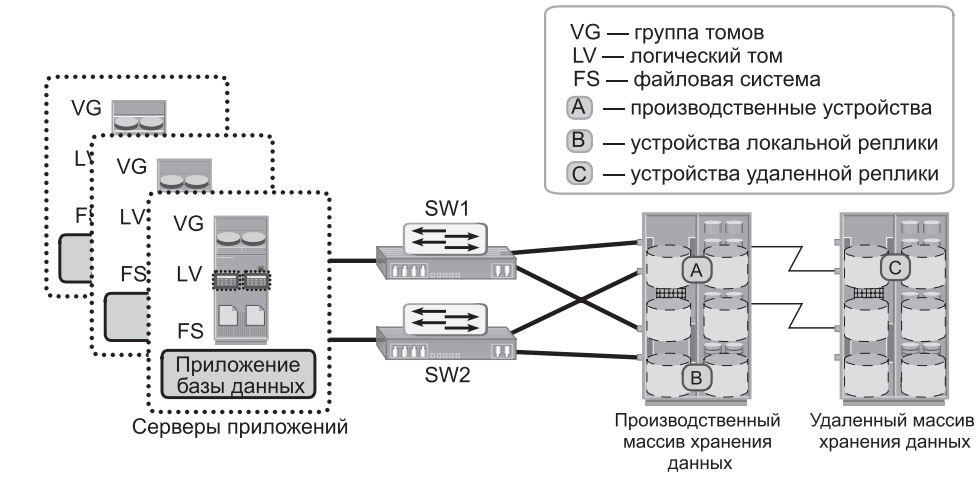


Рис. 15.8. Расширение файловой системы

На первом этапе определения стоимостей ресурсов, с приложением сопоставляется точный показатель полного объема сконфигурированного для него хранилища.

Как показано на рис. 15.10, пространство хранения данных приложения Payroll_1 прослеживается от файловой системы до логических томов, затем до групп томов и LUN-устройств массива. Когда приложения реплицируются, определяется также пространство хранилища, используемое локальной и удаленной репликами. В показанном примере приложение использует тома источника *Vol 1* и *Vol 2* (в производственном массиве хранения данных). Томами репликации являются тома локальной реплики *Vol 1* и *Vol 2*

(в производственном массиве хранения данных) и тома удаленной реплики *Vol 1* и *Vol 2* (в удаленном массиве хранения данных).



Приложение	Хранилище, Гбайт	Полный объем производственного хранилища, Гбайт	Полный объем хранилища локальной реплики, Гбайт	Полный объем хранилища удаленной реплики, Гбайт	Общий полный объем хранилища, Гбайт	Стоимость ресурсов из расчета по $5 за полный гигабайт
Payroll_1	100	200	100	125	425	$ 2125
Engineering_1	200	250	200	250	700	$ 3500

Рис. 15.9. Отчет о стоимости ресурсов

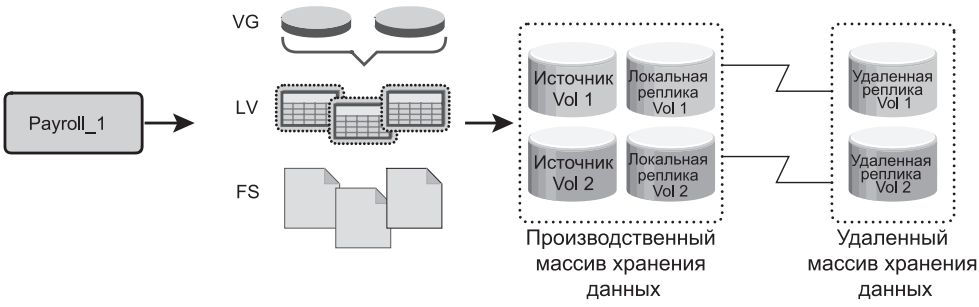


Рис. 15.10. Сопоставление объемов, сконфигурированных для приложения

После определения устройств массива хранения данных объем хранилища, выделенного приложению, может быть вычислен довольно легко. Предположим, что в данном примере тома источника *Vol 1* и *Vol 2* имеют объем по 50 Гбайт каждый, тогда хранилище, выделенное приложению, имеет объем 100 Гбайт (50 + 50). Хранилище, выделенное под локальную реплику,

имеет объем 100 Гбайт, и точно такой же объем, 100 Гбайт, отведен под удаленную реплику. Исходя из выделенного объема хранилища и на основе данных о RAID-защите, используемой для различных устройств массива хранения данных, определяется полный объем хранилища, сконфигурированного для данного приложения.

Производственные тома приложения Payroll_1 защищены с помощью применения технологии RAID 1, полное пространство, используемое производственными томами, составляет 200 Гбайт. Предположим, что локальные реплики хранятся на незащищенных томах, а удаленные реплики защищены с применением конфигурации RAID 5, тогда 100 Гбайт полного пространства используется локальной репликой и 125 Гбайт — удаленной репликой. Таким образом, общий полный объем, используемый приложением Payroll_1, составляет 425 Гбайт. Тогда общая стоимость хранилища, предоставленного приложению Payroll_1, будет равна $2125 (при условии, что стоимость 1 Гбайт хранилища составляет $5). Для создания отчета этот расчет должен повториться для каждого приложения, используемого на предприятии.

Отчеты о стоимости ресурсов могут быть расширены для включения заранее установленной стоимости других ресурсов с указанием, к примеру, имеющегося в конфигурации количества портов коммутатора, HBA-адаптеров и портов массива хранения данных. Отчеты о стоимости ресурсов для начисления платежей используют администраторы дата-центров, чтобы обеспечить потребителей ресурсов хранилища сведениями о стоимости запрашиваемых ими услуг.

15.3. Проблемы управления инфраструктурой хранилища данных

Мониторинг современной весьма сложной инфраструктуры хранения данных и управление ею — задачи непростые. Причина кроется в разнородности имеющихся в среде массивов хранения данных, сетей, серверов, баз данных и приложений. Например, разнородность массивов хранения данных может проявляться в их объеме, производительности, степени защиты и архитектурных построениях.

Каждый компонент дата-центра поставляется с предоставляемыми изготовителем средствами управления. Среда, в которой имеется множество инструментальных средств, затрудняет определение общего состояния среды, поскольку средства могут быть несовместимы друг с другом. В идеале средства управления должны сопоставлять информацию от всех компонентов, находящихся на одном производственном месте. Подобные средства обеспечивают сквозное представление среды и ускоряют анализ истинных причин возникновения той или иной ситуации для более быстрого принятия решения о выдаче предупреждений.

15.4. Выработка идеального решения

Идеальное решение должно предложить содержательное проникновение в суть состояния всей инфраструктуры и обеспечить проведение анализа истинных причин каждого сбоя. Это решение должно также предоставить возможность проведения централизованного мониторинга и управления в среде хранилища данных, использующей оборудование от нескольких поставщиков, и создать сквозное представление об инфраструктуре хранения данных.

Преимущество сквозного мониторинга заключается в возможности сопоставлять поведение одного компонента с поведением других компонентов. Зачастую отдельного взгляда на каждый компонент может быть недостаточно для выявления истинной причины возникновения той или иной проблемы. Система централизованного мониторинга и управления должна осуществлять сбор информации от всех компонентов, и управлять ими через однопользовательский интерфейс. Кроме того, она должна предоставлять механизм уведомления администраторов о различных событиях с использованием таких средств, как электронная почта и простой сетевой протокол управления — Simple Network Management Protocol (SNMP). У нее также должна быть возможность создания отчетов о мониторинге и запуска автоматических сценариев для автоматизации задач.

В основу идеального решения должны быть положены промышленные стандарты с использованием общепринятых API-интерфейсов, терминологии модели данных и систематики. Это позволит создать в среде разнородных устройств, служб, приложений и развернутых топологий систему управления, основанную на применении тех или иных политик.

Обычно для управления SAN-средой, сформированной из оборудования нескольких производителей, в качестве стандарта использовался протокол SNMP. Но одного протокола SNMP для предоставления подробной информации, требующейся при управлении SAN-средой, было недостаточно. К недостаткам протокола SNMP в SAN-среде можно отнести отсутствие функций автоматического обнаружения и слабые конструкции моделирования. Но даже при таких ограничениях SNMP по-прежнему удерживает доминирующую роль при управлении SAN-средой, хотя последние появившиеся открытые стандарты управления SAN-средой хранилища требуют повышения эффективности мониторинга и управления средами хранения данных.

15.4.1. Инициативные разработки в вопросах управления хранилищами данных

Для разработки универсального интерфейса управления была задействована ассоциация Storage Networking Industry Association (SNIA). SNIA разработала спецификацию под названием Storage Management Initiative-Specification (SMI-S). В основу спецификации положены система управления предприятием

с применением веб-технологий — Web-Based Enterprise Management (WBEM) и общая информационная модель — Common Information Model (CIM), разработанная рабочей группой по разработке, поддержке и продвижению стандартов, связанных с управлением в IT-средах, — Distributed Management Task Force (DMTF). Формально эта инициативная разработка была направлена на обеспечение возможности широкого взаимодействия внутри разнородных хранилищ данных и компонентов SAN-сети и управления ими. Дополнительную информацию по этому вопросу можно найти на сайте www.snia.org.

SMI-S предлагает пользователям и поставщикам ряд существенных преимуществ. Эта спецификация призвана сформировать стандартную абстрактную модель, на которую можно было бы отобразить физические и логические компоненты инфраструктуры хранилища данных. Эта модель используется прикладными программами управления, например программами управления ресурсами хранилища, управления устройствами и управления данными, обеспечивая тем самым сквозной контроль над ресурсами хранилища данных.

При использовании SMI-S разработчики программного обеспечения различных устройств имеют унифицированную модель объектов с подробностями, касающимися управления широким кругом компонентов хранилищ и SAN-сетей. Совместимые с SMI-S изделия существенно упрощают и ускоряют разработку и внедрение структур управления хранилищами на основе принятых на предприятии политик. Более того, спецификация SMI-S устраняет потребности в развертывании собственных интерфейсов управления, предоставленных поставщиком оборудования, и позволяет поставщикам сконцентрировать свои усилия на тех свойствах изделия, которые придают ему особую ценность.

15.4.2. Платформа управления в масштабах предприятия

Платформа управления в масштабах предприятия — enterprise management platform (EMP) представляет собой набор приложений, предоставляющий интегрированное решение для ведения мониторинга инфраструктуры хранилища данных предприятия и управления ею. Эти приложения обладают эффективными, гибкими и унифицированными структурами, обеспечивающими сквозное управление как физическими, так и виртуальными ресурсами. EMP предоставляет централизованную единую точку контроля ресурсов в среде хранения данных.

Эти приложения способны активно отслеживать состояние компонентов инфраструктуры хранилища и предупреждать пользователей о происходящих событиях. Предупреждения либо показываются на консоли, изображая сбойный компонент другим цветом, либо могут быть настроены на отправку сообщения по электронной почте. Кроме мониторинга EMP предоставляет необходимые функции управления, которые могут как реализовываться внутри самой платформы EMP, так и запускать утилиту управления, предоставленную изготовителем компонента.

Платформы EMP позволяют также без особого труда осуществлять планирование операций, которые должны выполняться на регулярной основе, например таких как предоставление ресурсов, управление конфигурацией и выяснение причин сбоев. Для облегчения задач управления инфраструктурой хранения данных эти платформы также предоставляют всесторонние возможности проведения анализа, корректировки и составления отчетов. Примерами EMP-платформ являются EMC ControlCenter и EMC Prosphere, описание которых дается в разделе 15.7 «Практическая реализация концепций: средства управления инфраструктурой от компании EMC».

15.5. Управление жизненным циклом информации

Как в традиционных дата-центрах, так и в виртуализированных средах управление информацией при его неправильной организации может обходиться весьма недешево. Кроме средств управления стратегия эффективного менеджмента требует также эффективности управления информацией. Эта стратегия направлена на решение следующих основных проблем современных дата-центров.

- **Взрыв цифровой вселенной.** Объемы информации растут в геометрической прогрессии. Многократному ускорению роста объемов информации способствует создание копий данных с целью обеспечения их высокой доступности и возможности повторного использования.
- **Усиление зависимости от информации.** Стратегическое использование информации играет весьма важную роль в определении успешности ведения бизнеса и предоставлении конкурентных преимуществ на рынке.
- **Изменение ценности информации.** Информация, обладающая ценностью сегодня, может быть менее важна уже завтра. Ценность информации со временем часто изменяется.

Формирование стратегии решения данных проблем предполагает понимание степени ценности информации на всем протяжении ее жизненного цикла. Зачастую информация имеет наивысшую ценность и частоту обращения к ней сразу после ее создания. С течением времени обращения к информации становятся все реже, и она становится все менее важной для организации. Понимание степени ценности информации помогает в развертывании инфраструктуры, соответствующей изменению этой ценности.

Например, в приложении заказов на покупки ценность информации (данных о клиенте) изменяется от времени размещения заказа и до времени истечения гарантийного срока (рис. 15.11). Ценность информации является наивысшей при получении компанией нового заказа на покупку и его

обработке с целью доставки товара. После выполнения заказа оперативный доступ к данным клиента уже не нужен. Компания может переместить эти данные в менее дорогостоящее второстепенное хранилище с меньшей производительностью на время, пока не истечет срок гарантии или пока надобность в данных не будет утрачена по иной причине. После истечения гарантийного срока на товар компания может удалить информацию.

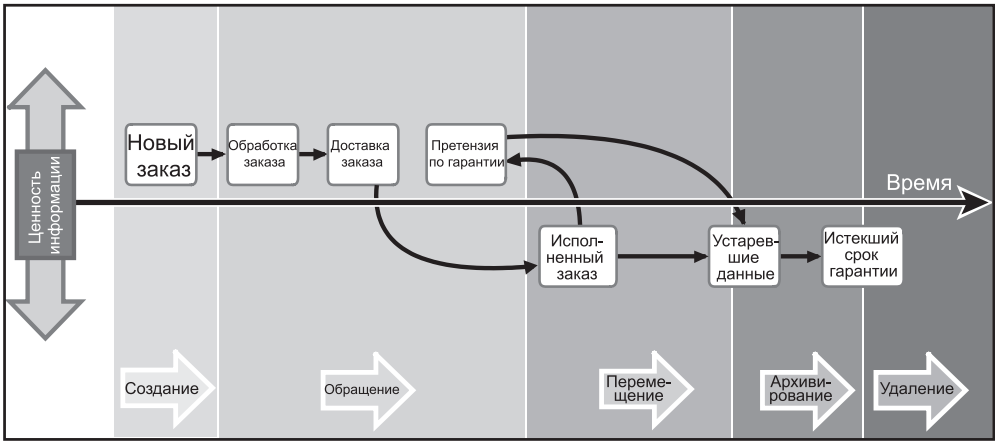


Рис. 15.11. Изменение ценности информации о заказе на покупку

Управление жизненным циклом информации — Information Lifecycle Management (ILM) является упреждающей стратегией, позволяющей IT-организациям эффективно управлять информацией на всем протяжении ее жизненного цикла на основе предопределенных бизнес-политик. От создания и до удаления данных ILM автоматически приводит бизнес-требования и процессы в соответствие с уровнями обслуживания. Это позволяет организациям оптимизировать инфраструктуру хранения данных для получения максимальной отдачи от капиталовложений. Реализация ILM-стратегии позволяет получить следующие основные преимущества, напрямую решающие проблемы управления информацией.

- **Низкая совокупная стоимость владения — Total Cost of Ownership (TCO).** Достигается благодаря приведению в соответствие с ценностью информации затрат на инфраструктуру и управление ее хранением. В результате ресурсы не расходуются впустую и сложность управления малоценными данными не выводится на уровень затрат на управление особо ценными данными.
- **Упрощенное управление.** Достигается благодаря интеграции этапов процесса и интерфейсов с отдельными средствами, а также за счет увеличения уровня автоматизации.

- **Обеспечение соответствия требованиям регуляторов.** Достигается осведомленностью о том, какие данные и на какой срок должны быть защищены.
- **Оптимизированное использование.** Достигается за счет развертывания многоуровневого хранения данных.

15.6. Многоуровневое хранение данных

Многоуровневое хранение данных является технологией, учреждающей иерархию различных типов хранилищ (уровней). Это позволяет хранить нужные данные на нужных уровнях, основываясь на требованиях уровня обслуживания при минимальных затратах. У каждого уровня имеются различные уровни защиты, производительности и стоимости. Например, хранилище первого уровня для хранения часто запрашиваемых данных может содержать высокопроизводительные твердотельные накопители (SSD) или FC-накопители, а в хранилище второго уровня, предназначенном для хранения реже запрашиваемых данных, могут использоваться менее дорогие SATA-накопители. Хранение часто запрашиваемых данных в SSD- или FC-накопителях повышает производительность приложения. Перемещение менее востребованных данных в SATA-накопители может высвободить объемы высокопроизводительных накопителей и сократить стоимость хранилища. Перемещение этих данных происходит на основе определенной политики разбиения на уровни. Эта политика может базироваться на таких параметрах, как тип файла, его размер, частота обращений к нему и т. д. Например, если в политике зафиксировано, что на более низкий уровень нужно перемещать файлы, к которым не было обращений в последние 30 дней, то все файлы, соответствующие этому условию, перемещаются на более низкий уровень.

Многоуровневое хранение данных может быть реализовано как с помощью ручных, так и с помощью автоматических процессов. *Осуществление многоуровневого хранения вручную* является традиционным методом, при котором администратор хранилища периодически отслеживает нагрузку на хранилище и перемещает данные между уровнями. Осуществление многоуровневого хранения вручную является довольно сложным процессом и занимает много времени. *Автоматизированное многоуровневое хранение данных* представляет собой процесс, при котором данные перемещаются между уровнями, не нарушая при этом режима работы хранилища. При автоматизированном многоуровневом хранении осуществляется активный мониторинг рабочей нагрузки приложения, часто востребуемые данные автоматически перемещаются на более производительный уровень, а редко востребуемые данные — на более объемный, но менее производительный уровень. Перемещение данных между различными уровнями может происходить как внутри массива хранения данных, так и между массивами.

15.6.1. Многоуровневое хранение в массиве хранения данных

Процесс многоуровневого хранения данных в пределах одного массива называется *внутренним многоуровневым хранением*. Он позволяет более эффективно использовать SSD-, FC- и SATA-накопители массива и дает возможность оптимизировать производительность и стоимость оборудования. Цель состоит в том, чтобы SSD-накопители были полностью задействованы для хранения наиболее востребованных данных с перемещением менее востребованных данных на SATA-накопители. Перемещение данных, осуществ-

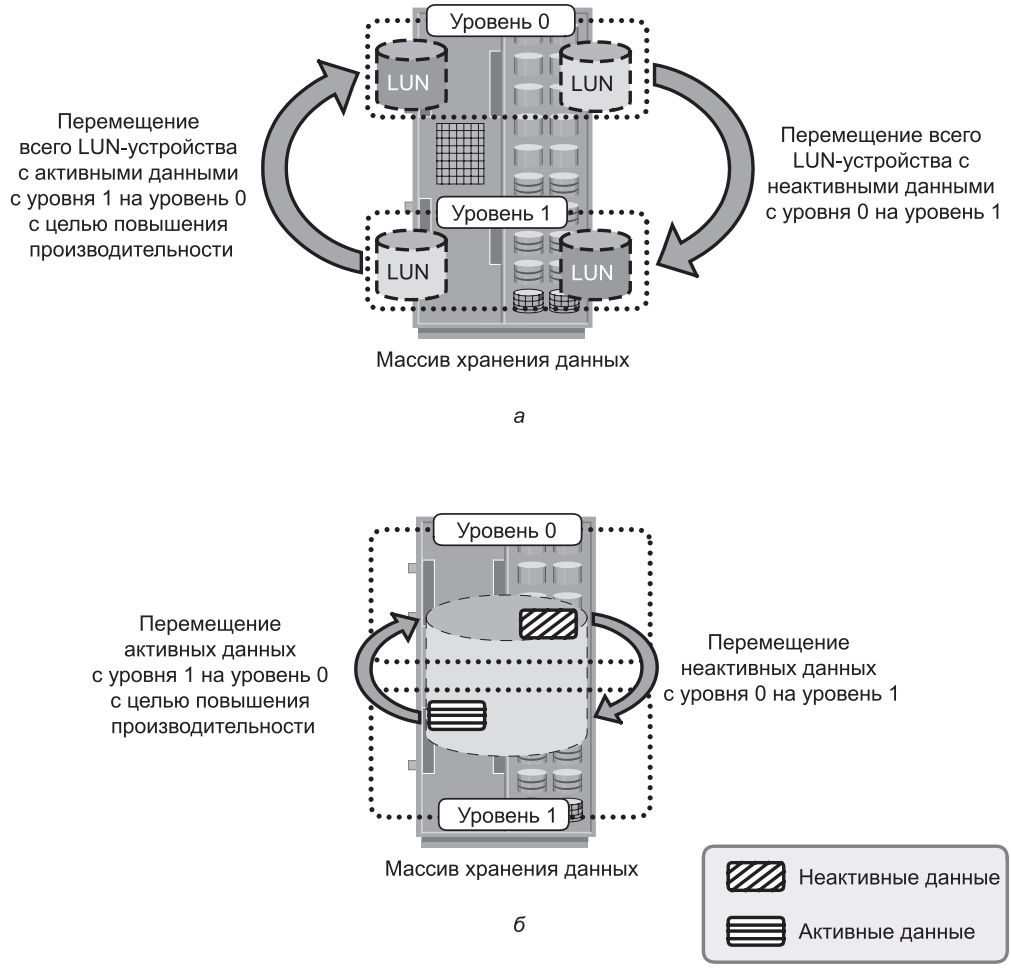


Рис. 15.12. Реализация многоуровневого хранения внутри массива хранения данных: *а* — многоуровневое хранение на уровне LUN-устройств; *б* — многоуровневое хранение на подуровне LUN-устройств

ляемое между уровнями, может проводиться на уровне LUN-устройств или на их подуровне. Производительность может быть дополнительно повышена за счет реализации многоуровневого кэша. Далее многоуровневое хранение на уровне LUN-устройств, на подуровне этих устройств и в кэше будет рассмотрено более подробно.

Обычно многоуровневое хранение осуществляется на уровне LUN-устройств, при этом с одного уровня хранилища на другой перемещается все LUN-устройство (рис. 15.12, *а*). В это перемещение включаются как активные, так и неактивные данные, находящиеся в этом LUN-устройстве. Этот метод не дает особых преимуществ как в экономии средств, так и в производительности. В настоящее время многоуровневое хранение данных может быть реализовано на подуровне LUN-устройств (рис. 15.12, *б*). При такой реализации LUN-устройство разбивается на более мелкие сегменты, на уровне которых и осуществляется многоуровневое хранение. Перемещение данных намного более мелкими порциями, например по 8 Мбайт, существенно повышает ценность перехода на автоматизированное многоуровневое хранение данных. Многоуровневое хранение данных на основе использования подуровней LUN-устройств позволяет эффективно перемещать активные данные на более быстрые накопители, а менее активные данные отправлять на менее быстрые накопители.

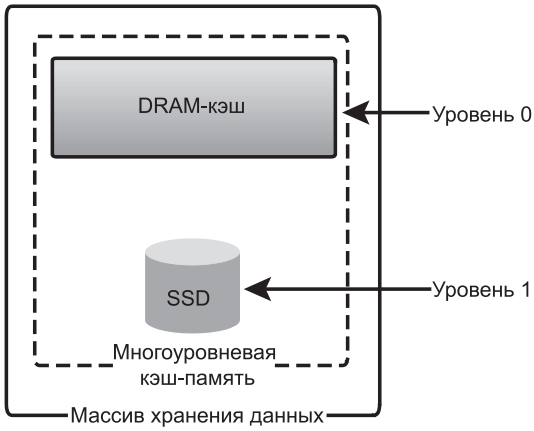


Рис. 15.13. Многоуровневое хранение данных в кэш-памяти

Многоуровневое хранение может быть реализовано также на уровне кэша (рис. 15.13). Большая кэш-память в массиве хранения данных повышает производительность, сохраняя в себе большое количество часто востребуемых данных, поэтому большинство операций чтения обслуживается непосредственно из кэш-памяти. Но наличие большой кэш-памяти в массиве хранения данных требует больших затрат на оборудование. Альтернативный

вариант повышения объема кэш-памяти предусматривает использование в массиве хранения данных твердотельных накопителей. При многоуровневом хранении данных в кэш-памяти твердотельные накопители используются как вторичная кэш-память большого объема, позволяющая проводить многоуровневое хранение данных между динамической памятью (DRAM), используемой в качестве первичного кэша, и памятью на твердотельных накопителях (SSD), используемой в качестве вторичного кэша. Серверное флеш-кэширование является еще одним уровнем кэша, в котором карта флеш-кэша установлена в сервер для дополнительного повышения производительности приложения.

15.6.2. Многоуровневое хранение между массивами хранения данных

Процесс многоуровневого хранения данных между массивами хранения называется *многоуровневым хранением данных между массивами*. Этот способ позволяет автоматизировать обнаружение активных или неактивных данных с целью их перемещения между массивами на уровни с другой производительностью или емкостью. На рис. 15.14 показан пример двухуровневой среды хранения данных. В этой среде первичное хранилище оптимизировано под достижение высокой производительности, а вторичное хранилище

Рис. 15.14. Реализация многоуровневого хранения данных между массивами хранения

оптимизировано под объем и стоимость. Механизм определения политик, который может быть как в программном, так и в аппаратном исполнении и в который закладываются применяемые политики, способствует перемещению неактивных или редко востребуемых данных из первичного хранилища во вторичное. Чаще всего многоуровневое хранение данных между массивами обусловливается требованиями, связанными с архивацией данных или достижением нужной совместимости оборудования. В качестве примера, механизм определения политик может быть настроен на перемещение всех файлов, находящихся в первичном хранилище, к которым не было обращений в течение месяца, в архив, находящийся во вторичном хранилище. Для каждого заархивированного файла механизм определения политик создает в первичном хранилище небольшой, не занимающий много места файл-заглушку, указывающий на данные, хранящиеся во вторичном хранилище. При попытке пользователя получить доступ к файлу по его прежнему месту в первичном хранилище ему совершенно незаметно предоставляется настоящий файл из вторичного хранилища.

15.7. Практическая реализация концепций: средства управления инфраструктурой от компании EMC

Сегодня бизнес сталкивается с трудностями управления собственной IT-инфраструктурой из-за наличия в своей среде большого количества разнородных ресурсов. Эти ресурсы могут быть физическими, виртуализированными или облачными. Чтобы удовлетворить различные потребности бизнеса, компания EMC предлагает различные средства. Продукты EMC ControlCenter и ProSphere являются наборами программ, способными осуществлять сквозное управление инфраструктурой хранения данных, а продукт EMC Unisphere является программой, управляющей массивами хранения данных производства компании EMC, например массивами VNX и VNXe. Продукт компании EMC под названием Unified Infrastructure Manager (UIM) является программой, управляющей инфраструктурой Vblock (облачными ресурсами). Дополнительную информацию по этим продуктам можно найти на сайте www.emc.com.

15.7.1. EMC ControlCenter и ProSphere

EMC ControlCenter представляет собой семейство приложений, предназначенных для управления ресурсами хранилищ — storage resource management (SRM). Это семейство предоставляет унифицированное решение, касающееся управления инфраструктурой хранения данных, составленной из

оборудования разных производителей. Оно помогает решить проблемы управления большой, сложной средой хранения данных, включающей хосты, сети хранения данных, хранилища и виртуализацию на всех уровнях. ControlCenter предоставляет такие возможности, как планирование использования хранилищ, предоставление объемов, ведение мониторинга и составление отчетов. Это средство позволяет реализовать стратегию управления жизненным циклом информации (ILM), предоставляя для этого всесторонний механизм управления многоуровневой инфраструктурой хранения данных. Оно также предоставляет сквозное представление всей сетевой инфраструктуры хранения данных, включающей ресурсы хранения данных SAN-сетей, NAS-устройств и хостов не только в обычной, но и в виртуализированной среде. Это средство предоставляет центральную консоль управления, способно обнаруживать новые компоненты, управлять квотами и событиями, проводить анализ истинных причин сбоев и составлять отчеты о стоимости ресурсов с целью начисления платежей. ControlCenter поставляется со встроенными функциями обеспечения безопасности, которые осуществляют контроль доступа, гарантируют конфиденциальность и целостность данных, ведут журнал событий и проводят аудит. В средстве используется интуитивно понятный и простой интерфейс, позволяющий проследить сложные взаимоотношения компонентов среды. Для обнаружения компонентов среды в ControlCenter используется программа-агент, работающая в фоновом режиме.

Продукт EMC ProSphere также является программой управления ресурсами хранилища, созданной для удовлетворения потребностей новой эры облачных вычислений. EMC ProSphere повышает продуктивность и уровни обслуживания в виртуализированной и облачной среде. ProSphere предоставляет следующие основные возможности.

- **Сквозное представление.** Программа предлагает интуитивно понятный и легкий в использовании интерфейс, позволяющий проследить изнутри сложные взаимодействия между объектами в больших виртуализированных средах.
- **Управление несколькими площадками.** С помощью единой консоли объединенная архитектура ProSphere собирает информацию со всех площадок и упрощает управление этой информацией между дата-центрами. Управление в ProSphere ведется из веб-браузера, что позволяет получать легкий доступ к удаленному управлению через Интернет.
- **Повышение продуктивности в расширяющихся виртуализированных средах.** ProSphere представляет инновационную технологию под названием Smart Groups, позволяющую для выполнения задач управления объединять объекты со сходными характеристиками в определяемые пользователем группы. Это дает возможность IT-специалистам применять подходы, основанные на заранее определенных политиках

управления для управления объектами или установки политик определения коллекций данных.

- **Ускорение, упрощение и повышение эффективности развертывания.** Обнаружение без использования программ-агентов устраняет трудоемкое развертывание и управление такими программами на хосте. Продукт ProSphere представлен в виде виртуальной оснастки, которая может быть установлена в короткие сроки.
- **Предоставление информационных технологий в виде услуг.** Теперь при использовании ProSphere уровни обслуживания могут отслеживаться от хоста до уровней хранения данных. Это позволяет организациям поддерживать постоянные уровни обслуживания в оптимальном соотношении цены/производительности, соответствующей целям ведения бизнеса по предоставлению информационных технологий в виде служб.

15.7.2. EMC Unisphere

EMC Unisphere представляет собой унифицированную платформу управления хранилищем данных, предоставляющую интуитивно понятный пользовательский интерфейс для управления массивами хранения данных типа EMC VNX и EMC VNXe. Unisphere допускает применение веб-технологий и поддерживает удаленное управление массивами хранения данных. Платформа Unisphere предоставляет следующие основные возможности:

- унифицированное управление файлового, блочного и объектного хранилища данных;
- единое предъявление пароля для всех устройств в домене управления;
- поддержка автоматизированного многоуровневого хранения данных и обеспечение их хранения на нужном уровне из соображений приемлемого соотношения цены/производительности;
- предоставление возможности управления как физическими, так и виртуальными компонентами.

15.7.3. EMC Unified Infrastructure Manager

EMC Unified Infrastructure Manager (UIM) представляет собой решение по управлению системами Vblock, которые рассматриваются в главе 13. Это средство позволяет настраивать инфраструктуру Vblock-ресурсов и активировать облачные службы. Оно предоставляет единый пользовательский интерфейс для управления несколькими Vblock-системами, исключающий необходимость отдельных настроек вычислительного устройства, сети и хранилища с использованием различных инструментов управления виртуальной инфраструктурой.

UIM предоставляет панель, на которой отображаются конфигурация инфраструктуры Vblock и используемые ресурсы. Это позволяет

администратору отслеживать конфигурацию и коэффициент использования ресурсов инфраструктуры Vblock и составлять планы обеспечения требуемых объемов. UIM также предоставляет топологию или схему Vblock-инфраструктуры, позволяющую администратору быстро найти и понять взаимосвязь компонентов и служб Vblock-инфраструктуры. Оно включает консоль предупреждений, которая позволяет администратору просматривать предупреждения, касающиеся ресурсов Vblock-инфраструктуры и связанных с ними услуг, затрагиваемых возникшими проблемами. При построении конфигурации ресурсов UIM выполняет проверку на соответствие самому передовому опыту создания подобных конфигураций ресурсов. Это средство также предотвращает назначение конфликтующих идентификаторов ресурсов, например случайное назначение одного и того же MAC-адреса более чем одному виртуальному сетевому адаптеру.

Резюме

Взрывной рост объема данных и их важности и рост зависимости бизнеса от цифровой информации привели к появлению более крупных и сложных инфраструктур хранения данных. Сложность управления этими инфраструктурами постоянно возрастает. Недостаточное управление инфраструктурами хранения данных в случае катастрофических сбоев может подвергнуть риску весь бизнес-процесс.

В данной главе была подробно рассмотрена деятельность по осуществлению мониторинга инфраструктуры хранения данных и управления ею. Кроме того, были исследованы особенности управления жизненным циклом информации, обеспечиваемые им преимущества, а также рассмотрено многоуровневое хранение данных.

Для получения дополнительной информации и материалов для изучения вопросов, касающихся хранения информации и управления хранилищами данных, а также вопросов виртуализации и облачных вычислений, можно обратиться к информационному ресурсу по адресу: http://education.emc.com/ismbook.

УПРАЖНЕНИЯ

1. Исследуйте спецификацию SMI-S и подготовьте презентацию на эту тему.
2. Исследуйте вопросы управления инфраструктурой облачных вычислений и услуг.
3. Исследуйте мультитенантность хранилищ и ее преимущества и недостатки.
4. В конструкторском бюро крупной компании ведется работа с более чем 600 000 чертежей, к которым конструкторы обращаются, повторно их используют и вносят в них изменения или обновления по мере надобности. Конструкторскому коллективу хочется получать незамедлительный доступ к чертежам своих текущих проектов, но на

данный момент он сдерживается существующей инфраструктурой, которая не в состоянии расширяться, чтобы отвечать требованиям по времени отклика. В этом коллективе чертежи классифицируются как наиболее часто востребуемые, часто востребуемые и востребуемые от случая к случаю.

- Предложите стратегию для конструкторского бюро, которая смогла бы помочь оптимизировать инфраструктуру хранения данных с использованием ILM.
- Объясните, как можно было бы использовать многоуровневое хранение на основе частоты обращения к чертежам.
- Дайте подробное описание аппаратных и программных компонентов, которые понадобятся для реализации вашей стратегии.
- Исследуйте продукты и решения, доступные в настоящее время для достижения поставленной цели.

5. Исследуйте управление хранилищем на основе использования объектов и расширяемых NAS-устройств.

Приложение А

Характеристики операций ввода-вывода, проводимых приложениями

Характеристики проводимых приложениями операций ввода-вывода оказывают влияние на общую производительность систем хранения данных и на проектирование решений по хранению данных. В этом приложении дается описание основных характеристик операций ввода-вывода.

Произвольный и последовательный

Ввод-вывод данных может быть либо произвольным, либо последовательным. К произвольному вводу выводу относятся следующие друг за другом операции чтения-записи по несмежным адресам, то есть обращение может распространяться на весь адресуемый объем LUN-устройства. Примерами приложений, создающих большей частью запросы на произвольный ввод-вывод, могут послужить приложения, осуществляющие обмен сообщениями и проводящие оперативную обработку транзакций — OLTP (online transaction processing).

К последовательному вводу-выводу относятся следующие друг за другом операции чтения-записи, осуществляемые по последовательно идущим адресам, то есть при таких операциях адреса логических блоков следуют по порядку. При проведении операций ввода-вывода с последовательным доступом время поиска на диске сокращается, поскольку для доступа к следующему блоку головка чтения-записи перемещается на незначительное расстояние. К примерам последовательного ввода-вывода можно отнести резервное копирование данных.

Операции чтения и записи

Еще одним аспектом рабочей нагрузки при осуществлении ввода-вывода служит соотношение количества запрашиваемых приложением операций чтения к количеству операций записи. Из суммы скорости чтения и скорости записи складывается скорость проведения операций ввода-вывода (количество операций ввода-вывода в секунду). Скорость проведения приложением операций ввода-вывода является одним из важнейших факторов, на основе которого определяется минимальное количество дисков, требующееся для приложения. Важную роль в повышении производительности системы хранения данных играет кэш данных. Показатели взаимодействия с кэшем данных операций ввода-вывода по чтению и записи сведены в табл. А.1.

Таблица А.1. Показатели взаимодействия операций ввода-вывода по чтению и записи с кэшем данных

ТИП ВВОДА-ВЫВОДА	ЧТЕНИЕ	ЗАПИСЬ
Произвольный	Эффективное использование кэша данных затруднено (сложно предсказать потребности в предвыборке); для достижения высокой производительности требуются несколько быстродействующих дисков	Кэширование дает положительный эффект, выражающийся в меньшем, чем у диска, времени отклика
Последовательный	Кэширование дает весьма существенный эффект (благодаря легкости предсказания предвыборки); чтение происходит со скоростью работы кэша данных	Кэширование дает положительный эффект; сброс данных из кэша на диск происходит довольно быстро, поскольку имеется возможность записи сразу всей дорожки диска

Обычно отношение количества операций чтения к количеству операций записи для наиболее распространенных бизнес-приложений бывает следующим.

- **Оперативная обработка транзакций (OLTP)**. 67 % занимают операции чтения и 33 % — записи.
- **Система поддержки принятия решений — decision support system (DSS)**. Иногда еще ее называют хранилищем полезных данных или системой бизнес-аналитики. В общей нагрузке ввода-вывода 80–90 % занимает чтение таблиц данных, включая частое сканирование таблиц (последовательное чтение).
- **Резервное копирование данных**. При условии, что файловая система не фрагментирована, резервное копирование на уровне файлов происходит в последовательном режиме.

Размер запроса на ввод-вывод

Размер выдаваемого приложением запроса на ввод-вывод может варьироваться в зависимости от типа приложения. Часть дополнительных данных, требующихся для осуществления ввода-вывода, имеет фиксированный размер. Если данные содержатся большими порциями, то эффективнее будет передавать более крупные блоки, поскольку при более объемной операции ввода-вывода хост сможет перемещать данные быстрее, чем при менее объемной операции. Время отклика при каждой большой транзакции больше времени отклика при отдельной небольшой транзакции, но общее время обслуживания множества мелких транзакций получается больше времени обслуживания одной транзакции, содержащей такой же объем данных. Наиболее распространенные приложения и их характеристики показаны в табл. А.2.

Таблица А.2. Характеристики приложений

ПРИЛОЖЕНИЕ	ТИП ПОИСКА	РАЗМЕР ЗАПРОСА НА ВВОД-ВЫВОД	ДОЛЯ ЗАПИСЕЙ В ОПЕРАЦИЯХ ВВОДА-ВЫВОДА
Microsoft Exchange	Произвольный	32 Кбайт	От средней до высокой
Приложения типа SAP и Oracle	Произвольный	~8 Кбайт	Зависит от конкретного приложения
Реляционные СУБД: ввод данных и OLTP-обработка	Произвольный	По размеру страницы базы данных или файловой системы	От средней до высокой
Реляционные СУБД: оперативное ведение журналов (транзакций)	Последовательный	512 байт+	Высокая, за исключением процессов архивирования
Реляционные СУБД: создание временного табличного пространства	Произвольный	По размеру страницы базы данных или файловой системы	Очень высокая

Приложение Б

Параллельный SCSI-интерфейс

В 1981 году компании Shugart Associates и NCR разработали системный интерфейс и назвали его Shugart Associates System Interface (SASI). SASI был разработан для создания собственного высокопроизводительного стандарта преимущественно для использования этими двумя компаниями. Но чтобы увеличить шансы SASI-интерфейса быть принятым в масштабах всей отрасли, стандарт был обновлен для получения более надежного интерфейса и переименован в SCSI. В 1986 году Американский национальный институт стандартов — American National Standards Institution (ANSI) признал новый SCSI промышленным стандартом.

Поскольку первоначально SCSI разрабатывался для жестких дисков, его часто сравнивают с интерфейсом IDE/ATA. SCSI обеспечивает повышенную производительность, масштабируемость и совместимость и поэтому хорошо подходит для компьютеров высокого класса. Но дороговизна не позволила ему набрать популярность у пользователей домашних или офисных настольных компьютеров.

До разработки SCSI интерфейсы, используемые для связи между устройствами, были разными для различных устройств. Например, интерфейс накопителя на жестком диске мог использоваться только с этим накопителем. Интерфейс SCSI был разработан, чтобы предоставить независимый от устройства механизм для подключения и получения доступа к базовым компьютерам. SCSI также предоставлял эффективную одноуровневую шину ввода-вывода, которая поддерживала сразу несколько устройств. В наши дни SCSI используется в основном как интерфейс жестких дисков. Тем не менее SCSI может использоваться для подключения к базовому компьютеру таких устройств, как накопители на магнитной ленте, принтеры и накопители на оптических носителях, не требуя при этом внесения каких-либо изменений в системное оборудование или программы. С годами SCSI-интерфейс претерпел радикальные изменения, превратившись в надежный промышленный стандарт.

Наряду с развитием SCSI-стандартов были подвергнуты ряду улучшений и SCSI-интерфейсы. Изначально SCSI-интерфейсом считался параллельный

SCSI — SCSI parallel interface (SPI). Сейчас конструктивно SCSI превращается в интерфейс с последовательным подключением — Serial Attached SCSI (SAS), который основан на конструкции двухточечного последовательного подключения с сохранением при этом всех остальных аспектов технологии SCSI.

Семейство стандартов SCSI

Стандартом SCSI определяются эталонная модель, устанавливающая общее поведение для SCSI-устройств, и абстрактная структура, типичная для всех реализаций систем ввода-вывода SCSI. В наборе SCSI-стандартов устанавливаются интерфейсы, функции и операции, необходимые для обеспечения совместимости соответствующих реализаций SCSI. Дополнительные сведения можно получить, изучив документ технического комитета T10 «SCSI Architecture Model-4 (SAM-4)» на сайте www.t10.org. Взаимоотношения между этим и другими стандартами, а также родственные по отношению друг к другу проекты в семействе стандартов SCSI показаны на рис. Б.1.

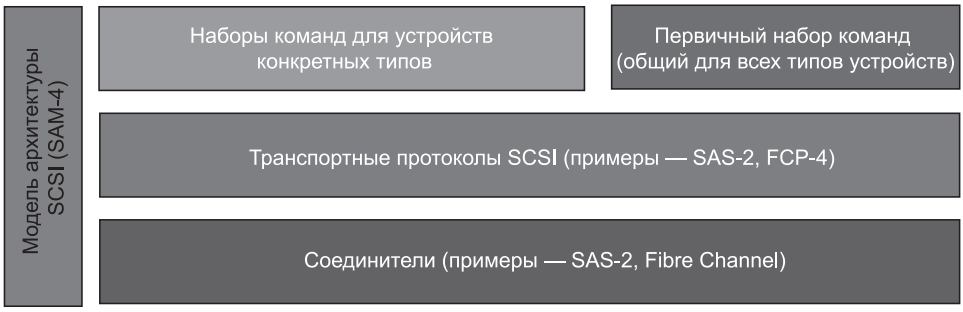


Рис. Б.1. Семейство стандартов SCSI

Описание компонентов семейства стандартов SCSI можно свести в следующий перечень.

- **Модель архитектуры SCSI**. Определяет модель SCSI-систем, функциональное разделение набора SCSI-стандартов и требования, предъявляемые ко всем SCSI-реализациям и стандартам реализаций.
- **Наборы команд для устройств конкретных типов**. Стандарты реализаций, определяющие конкретные типы устройств, включая модель устройства для каждого типа устройств. Эти стандарты определяют нужные команды и действия, характерные для заданного типа устройства, и предписывают требования, которых нужно придерживаться SCSI-устройству-инициатору при отправке команд SCSI-устройству-адресату определенного типа. Команды и действия для конкретного

типа устройств могут включать ссылки на команды и действия, общие для всех SCSI-устройств.

- **Общий набор команд**. Стандарт реализаций, определяющий модель для всех типов SCSI-устройств. Этот стандарт определяет нужные команды и действия, являющиеся общими для всех SCSI-устройств независимо от типа, и предписывает требования, которых нужно придерживаться SCSI-устройству-инициатору при отправке команд любому SCSI-устройству-адресату.
- **Транспортные протоколы SCSI**. Стандарты реализаций, определяющие требования по обмену информацией, позволяющие осуществлять этот обмен между различными SCSI-устройствами.
- **Соединители**. Стандарты реализаций, определяющие механизм связи, используемый транспортными протоколами SCSI. Эти стандарты могут описывать электрические и сигнальные требования, необходимые для взаимодействия SCSI-устройств по заданному соединению. Стандарты соединителей могут допускать соединение устройств, не относящихся к SCSI-устройствам, с применением способов, выходящих за пределы стандарта.

Клиент-серверная модель SCSI

В SCSI-среде понятие «инициатор-адресат» представляет собой не что иное, как клиент-серверную модель. В клиент-серверной модели SCSI конкретное SCSI-устройство работает как SCSI-устройство-адресат, SCSI-устройство-инициатор или SCSI-устройство, являющееся одновременно и инициатором, и адресатом. Устройства выполняют следующие функции.

- **SCSI-устройство-инициатор** — выдает команду SCSI-устройству-адресату на выполнение задачи. Примером инициатора может послужить SCSI-адаптер хоста.
- **SCSI-устройство-адресат** — выполняет команды в рамках задачи, полученной от SCSI-инициатора. Обычно в качестве адресата выступает периферийное SCSI-устройство, хотя в определенных реализациях устройством-адресатом может быть и адаптер хоста.

На рис. Б.2 показана клиент-серверная модель SCSI, в которой SCSI-инициатор, или клиент, отправляет запрос SCSI-адресату, или серверу. Адресат выполняет запрошенные задачи и отправляет выходные данные инициатору, используя интерфейс службы протокола.

В SCSI-устройстве-адресате имеется от одного до нескольких логических блоков. Под логическим блоком понимается объект, реализующий одну из функциональных моделей устройства, соответствующую описаниям стандартов SCSI-команд. Логический блок обрабатывает команду, отправленную SCSI-инициатором. У логического блока имеются два компонента: сервер

устройства и диспетчер задач. Сервер устройства реагирует на клиентские запросы, а диспетчер задач выполняет управленческие функции.

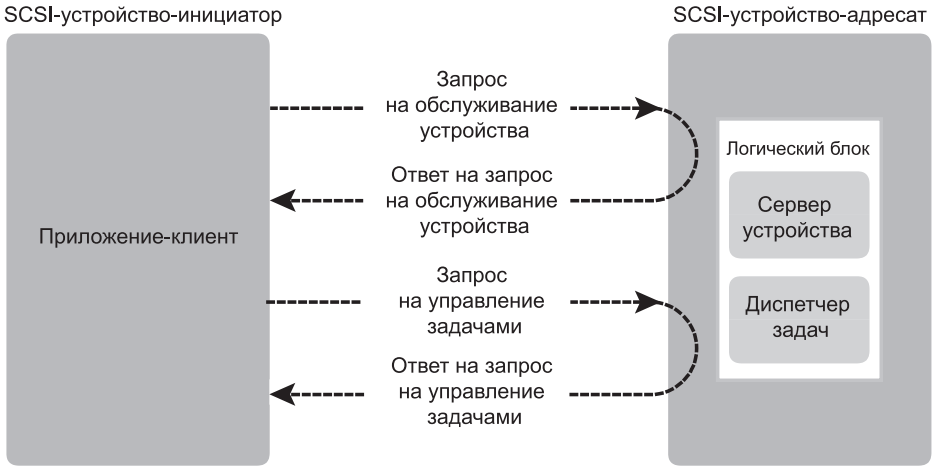


Рис. Б.2. Клиент-серверная модель SCSI

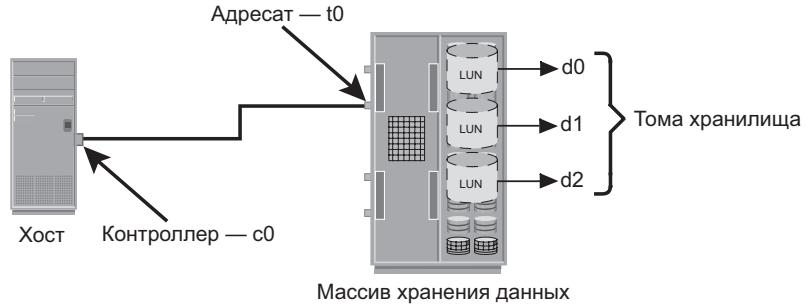
SCSI-устройство-инициатор состоит из приложения-клиента и функции управления задачами, инициирующей запросы на обслуживание устройства и на управление задачами. Каждый запрос на обслуживание устройства содержит блок описания команд — Command Descriptor Block (CDB), определяющий команды, которые нужно выполнить, и дается список вводимых данных, специфичных для конкретной команды, и другие параметры, определяющие порядок обработки команды. Приложение-клиент также создает задачи, объекты в логическом блоке, дающие представление о работе, связанной с командой или серией взаимосвязанных команд. Задача продолжает существовать до тех пор, пока не будет отправлен ответ о ее завершении — Task Complete Response или пока она не будет завершена в результате работы диспетчера задач или в результате возникновения условий выдачи исключения.

SCSI-устройства идентифицируются по определенным номерам, которые называются идентификаторами SCSI — SCSI ID. В узком SCSI (ширина шины = 8) устройства нумеруются от 0 до 7; в широком SCSI (ширина шины = 16) устройства нумеруются от 0 до 15. Эти ID-номера устанавливают приоритеты устройств на SCSI-шине. В узком SCSI устройство с номером 7 имеет наивысший приоритет, а устройство с номером 0 — самый низкий. В широком SCSI приоритет по нарастающей устанавливается для устройств с идентификаторами от 8 до 15, но всей последовательности идентификаторов широкого SCSI присваивается более низкий приоритет, чем идентификаторам узкого SCSI. Поэтому общая последовательность приоритетов для широкого SCSI имеет следующий вид: 7, 6, 5, 4, 3, 2, 1, 0, 15, 14, 13, 12, 11, 10, 9 и 8.

При инициализации устройства SCSI позволяет автоматически присваивать идентификатор на шине, что не дает двум и большему числу устройств использовать один и тот же SCSI ID.

Адресация в параллельном SCSI

При установлении связи в параллельном SCSI между инициатором и адресатом (рис. Б.3) идентификатор инициатора обеспечивает его уникальную идентификацию и используется в качестве начального элемента адреса. Этот идентификатор может находиться в диапазоне от 0 до 15, но чаще всего встречается диапазон от 0 до 7. Идентификатор адресата также обеспечивает его уникальную идентификацию и используется в качестве адреса для обмена командами и информацией о состоянии с инициаторами. Идентификатор адресата может находиться в диапазоне от 0 до 15.



Адресация со стороны хоста:
Том хранилища 1 — c0 t0 d0
Том хранилища 2 — c0 t0 d1
Том хранилища 3 — c0 t0 d2

Рис. Б.3. Связь между SCSI-инициатором и SCSI-адресатом (целевым устройством)

Для идентификации диска в SCSI-адресации используется соглашение об именах, действующее в операционной системе UNIX. В этой идентификации применяются три идентификатора: ID инициатора, ID адресата и номер LUN-устройства, при этом получается формат `cn|tn|dn`, который называют также ctd-адресацией. В нем `cn` означает идентификатор инициатора, который часто называют идентификатором контроллера, `tn` — идентификатор адресата (целевого устройства), например, `t0`, `t1`, `t2` и т. д., а `dn` — номер устройства, который отображает текущий адрес на блок устройства, например, `d0`, `d1` и `d2`. Номер устройства идентифицирует конкретную логическую единицу адресата (целевого устройства). В зависимости от производителя конкретного изделия реализация SCSI-адресации может иметь некоторые различия.

Приложение B

Упражнения по разработке SAN-сетей

Упражнение 1

Организация захотела создать FC-SAN-сеть с топологией полной решетки. Спецификация задействованных в этой конструкции серверов, систем хранения данных и коммутаторов выглядит следующим образом.

- Количество хостов = 30. У каждого хоста имеется два HBA-адаптера, у каждого из которых один порт.
- Количество массивов хранения данных = 4. У каждого массива имеется восемь интерфейсных портов.
- Доступны следующие элементы коммутации:
  - модульные FC-коммутаторы, у каждого из которых минимум 16 портов. За счет установки дополнительных адаптеров портов количество портов в коммутаторе может быть увеличено до 32. У каждого адаптера имеется 8 портов;
  - для обеспечения высокого уровня доступности между любыми двумя коммутаторами должны быть проложены минимум две линии межкоммутаторной связи (ISL-линии).

Какое минимальное количество коммутаторов понадобится для выполнения поставленных условий? Выберите для каждого FC-коммутатора такое количество портов, которое позволило бы оптимизировать затраты.

Решение

Общее количество портов хоста = 30 хостов · 2 порта = 60 портов.

Общее количество портов массивов хранения данных = 4 массива · · 8 портов = 32 порта.

Общее количество портов узлов = 60 + 32 = 92 порта.

Каждый FC-коммутатор может предоставить максимум 32 порта. При условии наличия 32 портов на каждом коммутаторе, и использовании четырех коммутаторов получится всего 128 портов. В четырех 32-портовых коммутаторах в топологии полной решетки 24 порта используются для ISL-линий, а остальные 104 порта могут использоваться для связи между узлами. Но системе коммутации для этой цели требуются 92 порта. Поэтому в целях оптимизации затрат организации следует развернуть три 32-портовых коммутатора и один коммутатор, имеющий 24 порта. В данной реализации для связи между узлами будут доступны 96 портов, из них 92 порта могут использоваться по прямому назначению, а 4 порта будут доступны для дальнейшего наращивания сети.

Упражнение 2

В IT-инфраструктуре организации имеются три массива хранения данных, подключенных напрямую к группе из 45 разнородных серверов. Чтобы обеспечить высокую доступность данных, у всех серверов имеется двойное подключение к массивам. Поскольку у каждого массива хранения данных имеется 32 интерфейсных порта, каждый из них может поддерживать максимум 16 серверов. Но у каждого имеющегося массива хранения данных дисковой емкости хватает на поддержку максимум 32 серверов. Организация в рамках расширения планирует приобрести еще 45 серверов.

Если организация будет и дальше использовать хранилище с непосредственным подключением, ей придется для подключения к этим новым серверам приобрести дополнительные массивы хранения данных. В организации считают, что имеющиеся у нее массивы хранения данных загружены не полностью, поэтому имеются планы реализовать FC-SAN-сеть, чтобы решить все проблемы расширения и загруженности оборудования. В организации используются высокопроизводительные приложения, поэтому необходимо свести к минимуму количество транзитных участков на пути доступа серверов к хранилищу.

Предложите топологию системы коммутации, отвечающую запросам и помогающую решить проблемы организации. Обоснуйте свой выбор топологии системы коммутации. Определите минимальное количество требующихся в системе коммутаторов при условии, что для реализации FC-SAN-сети доступны коммутаторы, у каждого из которых имеется 72 порта.

Решение

Для среды, требующей больших возможностей для расширения, топология полной решетки не подойдет. А неполная решетка предоставит больше возможностей для расширения, чем полная, но для того чтобы сетевой трафик дошел по назначению, может понадобиться несколько транзитных участков или ISL-линий. Поэтому можно порекомендовать использование топологии

«центр — периферия». Эта топология дает больше возможностей для расширения, чем топология полной решетки, и обеспечивает всем серверам в своей среде доступ к хранилищу с использованием всего лишь одного транзитного участка. Поскольку FC-трафик идет по обусловленной схеме (от периферии к центру), будет несложно подсчитать распределение нагрузки на ISL-линии:

> Общее количество портов серверов = 90 серверов · 2 порта = 180 портов;
>
> Общее количество портов массивов = 3 массива · 32 порта = 96 портов массивов;
>
> Количество коммутаторов центра = 96 портов массивов/72 порта на каждом коммутаторе = 2 коммутатора.

Коммутаторы центра предоставляют 144 порта, из которых 96 портов будут использоваться для подключения к массивам хранения данных. Остальные 48 портов могут использоваться для ISL-линий и дальнейшего расширения системы.

> Количество коммутаторов на периферии = 180 портов серверов/72 порта на каждом коммутаторе = 3 коммутатора.

Периферийные коммутаторы предоставляют 216 портов, из которых 180 портов будут использоваться для подключения к серверам. Остальные 36 портов могут использоваться для ISL-линий и будущего расширения системы.

> Количество портов периферийных коммутаторов, используемых для подключения к центральным коммутаторам = 6.

Это количество меньше количества оставшихся портов периферийных коммутаторов.

> Количество портов центральных коммутаторов, используемых для подключения к периферийным коммутаторам = 6.

Это количество меньше количества оставшихся портов центральных коммутаторов.

Получается, что для реализации системы коммутации по топологии «центр — периферия» понадобится как минимум два центральных и три периферийных коммутатора.

Приложение Г

Упражнения по доступности информации

Упражнение 1

У системы имеется три компонента, и нужно, чтобы все они работали круглосуточно с понедельника по пятницу. Сбой первого компонента случался в следующей хронологии:

- понедельник — без сбоев;
- вторник — с 5:00 до 7:00;
- среда — без сбоев;
- четверг — с 16:00 до 20:00;
- пятница — с 8:00 до 11:00.

Вычислите среднее время безотказной работы MTBF и среднее время восстановления MTTR компонента 1.

Решение

Формула для вычисления MTBF:

MTBF = Полное рабочее время/Количество сбоев.

Следовательно,

MTBF = (24 часа · 5 дней)/3 = 120 часов/3 = 40 часов.

Формула для вычисления MTTR:

MTTR = Общее время простоя/Количество сбоев.

Следовательно,

Общее время простоя = 2 часа во вторник + 4 часа в четверг + 3 часа в пятницу.

А в итоге:

MTTR = (9 часов/3) = 3 часа.

Упражнение 2

В системе имеется три компонента, и требуется, чтобы они функционировали в рабочее время с 8:00 до 17:00 с понедельника по пятницу. Сбой второго компонента случался в следующей хронологии:

- понедельник — с 8:00 до 11:00;
- вторник — без сбоев;
- среда — с 16:00 до 19:00;
- четверг — с 17:00 до 20:00;
- пятница — 13:00 до 14:00.

Вычислите показатели доступности компонента 2.

Решение

Доступность [%] = Период безотказной работы/(Период безотказной работы + Период простоя).

Общий период простоя системы = 3 часа в понедельник + 1 час в среду + 1 час в пятницу = 5 часов.

Общий период безотказной работы системы = Полное рабочее время – – Общий период простоя системы = 45 часов – 5 часов = 40 часов.

Доступность [%] = 40/45 = 88,9 %.

Рабочее время с 8:00 до 17:00, поэтому сбой компонента в другое время не будет расцениваться как простой оборудования.

Приложение Д

Сетевые технологии для удаленной репликации

Для удаленной репликации на большие расстояния развертываются различные сети на основе оптоволоконных технологий со спектральным уплотнением высокой плотности — dense wavelength division multiplexing (DWDM), с разреженным спектральным уплотнением — coarse wavelength division multiplexing (CWDM) и с применением синхронной оптической сети — synchronous optical network (SONET).

DWDM

Спектральное уплотнение высокой плотности — dense wavelength division multiplexing (DWDM) — это оптоволоконная технология, благодаря применению которой различные данные с различных каналов переносятся потоками на разных длинах волн по оптоволоконному кабелю одновременно. Это значительно отличается от обычной оптоволоконной системы, в которой один канал переносится потоком на одной длине волны по одному волокну. DWDM является оптоволоконной технологией передачи данных, в которой несколько отдельных потоков данных (или каналов) с разными длинами волн могут быть уплотнены в один многоцветный световой поток, передаваемый по одному оптическому волокну.

С помощью DWDM различные форматы данных с различными скоростями передачи могут передаваться вместе. То есть IP-, ESCON-, FC-, SONET- и ATM-данные могут проходить по оптоволокну одновременно (рис. Д.1).

Технология DWDM позволяет уплотнять и разуплотнять большое количество каналов. Каждому каналу выделяется собственный конкретный диапазон волн (λ). Все диапазоны волн, как правило, разносятся на 10 нм друг от друга. По мере совершенствования оптических технологий разнесение

может становиться все меньше, позволяя упаковывать (уплотнять) в одном оптоволокне еще больше каналов.

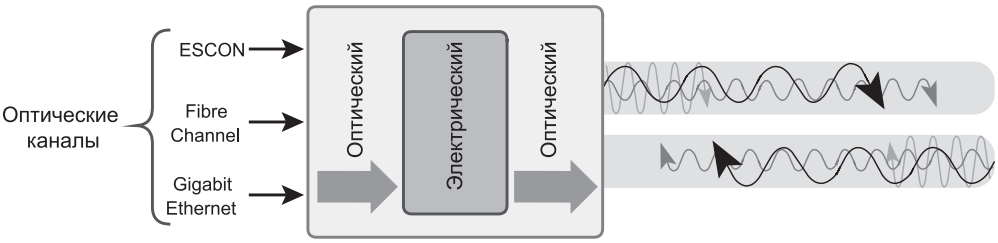


Рис. Д.1. Спектральное уплотнение высокой плотности (DWDM)

CWDM

Оптоволоконная технология с разреженным спектральным уплотнением — coarse wavelength division multiplexing (CWDM), как и технология DWDM, позволяет переносить различные данные из различных каналов потоками на разных длинах волн одновременно по одному и тому же оптоволокну. По сравнению с DWDM технология CWDM объединяет среды, содержащие меньшее количество каналов, при этом на ее реализацию тратится меньше средств. В CWDM используется разнесение между каналами 20 нм. При использовании технологии CWDM количество длин волн, выделяемых каналам для упаковки в одно оптоволокно, существенно уменьшается. CWDM-система поддерживает 16 каналов и менее, в то время как количество каналов, поддерживаемых DWDM-системой, — 16 и более.

SONET

Синхронная оптическая сеть — synchronous optical network (SONET) — представляет собой сетевую технологию, предназначенную для передачи большого объема данных по оптоволоконному кабелю на большие расстояния и работающую на физическом уровне. В SONET потоки данных, имеющие разные скорости. уплотняются в кадр и отправляются по сети. Европейским вариантом SONET является стандарт синхронной цифровой иерархии — synchronous digital hierarchy (SDH).

SONET/SDH использует универсальную процедуру кадрирования — generic framing procedure (GFP) и поддерживает передачу как пакетно-ориентированных (Ethernet, IP), так и посимвольных (FC) данных. Для передачи данных по оптоволоконному кабелю в SONET определяются оптическая линия

связи — optical carrier (OC) и электрически эквивалентный синхронно-транспортный сигнал — electrically equivalent synchronous transport signal (STS).

SONET позволяет передавать данные с большой скоростью. (Например, OC-768 обеспечивает скорость на линии вплоть до 40 Гбит/с.) Основной сигнал SONET/SDH передается со скоростью 51,84 Мбит/с и обозначается синхронно-транспортным сигналом первого уровня (STS-1) или OC-1. Кадр STS-1 является в SONET/SDH основной единицей передачи. Например, для получения высокоскоростных каналов можно объединить несколько каналов STS-1. Эквивалентом SONET уровня OC-3 и SDH уровня STM-1 (Synchronous Transport Module) является STS-3 (155,52 Мбит/с).

Сокращения и аббревиатуры

ACA	Auto Contingent Allegiance	Автоконтингентная зависимость
ACL	Access Control List	Список управления доступом
AD	Active Directory	Продукт компании Microsoft для обеспечения централизованной аутентификации и авторизации
AES	Advanced Encryption Standard	Улучшенный стандарт шифрования
AL-PA	Arbitrated Loop Physical Address	Физический адрес управляемой петли
ALU	Arithmetic Logic Unit	Арифметическое логическое устройство
ANSI	American National Standards Institute	Американский государственный институт стандартов
API	Application Programming Interface	Интерфейс прикладного программирования (набор функциональных обращений, устанавливающих связь между приложениями или между приложением и операционной системой)
AR	Automated Replication	Автоматическая репликация
ARB	Arbitration Frame	Арбитражный кадр
AS	Authentication Service	Служба аутентификации
ASCII	American Standard Code for Information Interchange	Американский стандарт по обмену информацией
ASIC	Application-Specific Integrated Circuit	Проблемно-ориентированная (специализированная) интегральная микросхема
ATAPI	Advanced Technology Attachment Packet Interface	Пакетный интерфейс периферийных устройств для AT-совместимых компьютеров, интерфейс ATAPI стандарт, позволяющий EIDE-контроллерам поддерживать накопители CD-ROM и НМЛ. Другое его название — ATA-4
ATM	Asynchronous Transfer Mode	Асинхронный способ передачи
AUI	Application User Interface	Прикладной пользовательский интерфейс

AVM	Automatic Volume Management	Автоматическое управление тома
BB_Credit	Buffer to Buffer Credit	Разрешение передачи от буфера к буферу
BBU	Battery Backup Unit	Батарейный резервный источник питания при сбое электросети
BC	Business Continuity	Непрерывность бизнеса (готовность к восстановлению данных и реагирование на аварийные ситуации, которые могут навредить бизнесу)
BCP	Business Continuity Planning	Планирование непрерывности ведения бизнеса
BCV	Business Continuance Volume	Тома для поддержания непрерывности ведения бизнеса (копия рабочих томов, как правило, используемая для восстановления данных. Одновременно архив может использоваться в других коммерческих целях, не нарушающих работы производственных систем делопроизводства)
BIA	Business Impact Analysis	Анализ влияния на бизнес
BIOS	Basic Input/Output System	Базовая система ввода/вывода
BLOB	Binary Large Object	Большой двоичный объект
BMR	Bare Metal Recovery	Восстановление системы с нуля
BURA	Backup, Recovery, and Archive	Резервирование, восстановление и архивирование
CA	Content Address	Контентный адрес (идентификатор, указывающий на содержимое файла, а не на его физическое расположение)
CAS	Content-Addressed Storage	Объектно-ориентированная система хранения данных с фиксированным контентом. Рентабельный вариант сетевого хранения данных
CCS	Common Command Set	Общий набор команд
CD	Compact Disk	Компакт-диск
CDB	Command Descriptor Block	Блок дескриптора команд
CDF	Content Descriptor File	Файл дескриптора контента (файл в формате XML, содержащий контентный адрес и метаданные объекта в системе хранения данных с контентной адресацией)
CDP	Continuous Data Protection	Постоянная защита данных
CD-R	Compact Disk-Recordable	Компакт-диск для записи
CD-ROM	Compact Disk Read-Only Memory	Постоянное запоминающее устройство на компакт-диске
CD-RW	Compact Disc Rewritable	Многоразовый компакт-диск (компакт-диск с возможностью перезаписи)

CE+	Compliance Edition Plus	Расширенный вариант соответствия
CG	Consistency Group	Согласованная группа
CHAP	Challenge-Handshake Authentication Protocol	Протокол, используемый инициатором и целевым устройством для взаимной аутентификации при помощи обмена секретным кодом или паролем
CHS	Cylinder, Head, and Sector	Цилиндр, головка и сектор
CIFS	Common Internet File System	Общая межсетевая файловая система
CIM	Common Information Model	Общая информационная модель
CKD	Count Key Data	Основные счетные данные
CLI	Command-Line Interface	Интерфейс командной строки
CmdSN	Command Sequence Number	Номер последовательности команды
CMI	CLARiiON Messaging Interface	Коммуникационный интерфейс CLARiiON
CMIP	Common Management Information Protocol	Протокол передачи общей управляющей информации
CMIS	Common Management Information Service	Общий информационный сервис управления
COFA	Copy on First Access	Копирование при первом доступе
COFW	Copy on First Write	Копирование при первой записи
CPC	Control Path Cluster	Специальное устройство хранения данных с установленным приложением Invista; устройство хранит параметры конфигурации приложения Invista
CPM	Content Protection Mirrored	Зеркальное копирование объекта данных для защиты данных при отказе оборудования
CPP	Content Protection Parity	Преобразование данных в сегменты с дополнительным сегментом контроля четности для защиты данных при отказе оборудования
CPU	Central Processing Unit	Центральный процессор
CRC	Cyclic Redundancy Check	Метод обнаружения ошибок в цифровых данных и проверки целостности данных. При этом методе определенное количество контрольных разрядов, часто называемых контрольной суммой, добавляют к передаваемому сообщению
CS_CTL	Class-Specific Control	Контроль по классам
CSMA/CD	Carrier Sense Multiple Access/ Collision Detection	Набор правил, определяющих реакцию сетевых устройств при попытке одновременного использования канала данных двумя устройствами (при конфликте устройств)

CWDM	Coarse Wave Division Multiplexing	Грубое волновое мультиплексирование
DAC	Discretionary Access Control	Произвольный контроль доступа
DACL	Discretionary Access Control List	Список произвольного контроля доступа
DAE	Disk Array Enclosure	Корпус дискового массива
DART	Data Access in Real Time	Доступ к данным в реальном времени
DAS	Direct-Attached Storage	Хранилище с прямым подключением
DBA	Database Administrator	Администратор базы данных
DBMS	Database Management System	Система управления баз данных
DCP	Data Collection Policy	Правила сбора данных
DDR SDRAM	Double Data Rate Synchronous Dynamic Random Access Memory	Динамическая синхронная оперативная память удвоенной скорости
DF-CTL	Data Field Control	Контроль области данных
DFS	Distributed File System	Распределенная файловая система
DHCP	Dynamic Host Configuration Protocol	Протокол динамического конфигурирования хоста
D_ID	Destination ID	ID назначения
DMTF	Distributed Management Task Force	Распределенное управление задачами
DMX	Direct Matrix	Прямая матрица
DMZ	Demilitarized Zone	Демилитаризованная зона
DNS	Domain Name System	Служба доменных имен
DoS	Denial of Service	Отказ в услуге
DPC	Data Path Controller	Контроллер пути к данным
DPE	Disk Processor Enclosure	Устройство управления диском
DRM	Digital Rights Management	Управление цифровыми правами
DSA	Directory System Agent	Агент системы каталогов
DVD	Digital Versatile Disk or Digital Video Disc	Цифровой многоцелевой диск или цифровой видеодиск
DVD-ROM	Digital Versatile Disk Read-Only Memory	Только читаемый DVD
DWDM	Dense Wave Division Multiplexing	Мультиплексирование по длине волны. Технология одновременной передачи данных из разных источников по одному оптоволоконному кабелю
ECA	Enginuity Consistency Assist	Мастер поддержки соответствия
ECC	EMC Control Center	Центр управления EMC
E_D_TOV	Error Detect Time Out Value	Период обнаружения ошибок
EE_Credit	End-to-End Credit	Разрешение на сквозную передачу пакетов данных

EIDE	Enhanced Integrated Drive Electronics	Улучшенная интегрированная электроника жестких дисков
EMP	Enterprise Management Platform	Платформа управления предприятием
EOF	End of Frame	Конец кадра
E_Port	Expansion Port	Порт расширения
ESCON	Enterprise Systems Connection	Подключение систем предприятия
ETL	Extract, Transform, and Load	Извлечение, трансформация и загрузка
EUI	Extended Unique Identifier	Расширенный уникальный идентификатор
EXT 2/3	Extended File System	Расширенная файловая система
FAT	File Allocation Table	Таблица размещения файлов
FBA	Fixed-Block Architecture	Архитектура фиксированных блоков
FC	Fibre Channel	Оптоволоконный канал
FC-AL	Fibre Channel Arbitrated Loop	Высокоскоростная последовательная шина, оптоволоконный интерфейс
F_CTL	Frame Control	Контроль кадра
FCIP	Fibre Channel over IP Protocol	Технология оптоволоконной передачи данных в IP-сетях
FCP	Fibre Channel Protocol	Протокол оптоволоконного канала
FC-PH	Fibre Channel Physical and Signaling Interface	Физический интерфейс оптоволоконного канала
FC-PI	Fibre Channel Physical Interface	Физический интерфейс оптоволоконного канала
FC-SP	Fibre Channel Security Protocol	Протокол безопасности оптоволоконного канала
FC-SW	Fibre Channel Switched Fabric	Многовходовая система коммутации оптоволоконного канала
FCWG	Fibre Channel Working Group	Рабочая группа оптоволоконного канала
FDDI	Fibre Distributed Data Interface	Интерфейс для передачи распределенных данных по оптоволоконным каналам
FICON	Fibre Connectivity	Связность оптоволоконных каналов
FIFO	First In First Out	Алгоритм «первым пришел — первым обслужен»
FLOGI	Fabric Login	Регистрация между оптоволоконными N_Port and F_port
FL_Port	Fabric Loop Port	Порт-шлюз (порт коммутатора, подключаемый к оптоволоконной управляемой петле)
F_Port	Fabric Port	Порт коммутатора, подключаемый к N-порту
FRU	Field Replaceable Unit	Сменный блок устройства
FSPF	Fabric Shortest Path First	Используемый в оптоволоконных сетях протокол маршрутизации, вычисляющий кратчайший путь между узлами
FTP	File Transfer Protocol	Протокол передачи файла

GB	Gigabyte	Гигабайт
GigE	Gigabit Ethernet	Гигабитный Ethernet
GBIC	Gigabit Interface Converter	Преобразователь гигабитного интерфейса
GB/s	Gigabyte per Second	Гигабайт в секунду
Gb/s	Gigabit per Second	Гигабит в секунду
GFP	Generic Framing Procedure	Способ мультиплексирования, позволяющий кодировать данные переменной длины для передачи кадрами постоянной длины
GFV	Global File Virtualization	Глобальная файловая виртуализация
GHz	Gigahertz	Гигагерц
GMD	Global Memory Director	Блок управления глобальной памяти
GUI	Graphical User Interface	Графический интерфейс пользователя
HBA	Host Bus Adapter	Адаптер главной шины
HDA	Head Disk Assembly	Блок дисков с головками
HDD	Hard Disk Drive	Жесткий диск, накопитель на жестком диске
HIPAA	Health Insurance Portability and Accountability Act	Закон об отчетности и безопасности медицинского страхования
HIPPI	High Performance Parallel Interface	Параллельный интерфейс высокой производительности
HSM	Hierarchical Storage Management	Управление иерархическим хранилищем
HTTP	Hypertext Transfer Protocol	Гипертекстовый протокол передачи
HWM	High Watermark	Высокий уровень заполнения
IA	Information Availability	Доступность информации
IDE/ATA	Integrated Device Electronics/ Advanced Technology Attachment	Встроенный интерфейс накопителей. Стандартный интерфейс для подключения запоминающих устройств к персональным компьютерам
IDS/IPS	Intrusion Detection/Intrusion Prevention System	Система обнаружения сетевых атак/система предотвращения сетевых атак
IEEE	Institute of Electrical and Electronics Engineers	Институт инженеров по электротехнике и радиоэлектронике
IETF	Internet Engineering Task Force	Комитет по инженерным вопросам (проблемам) Internet
iFCP	Internet Fibre Channel Protocol	Протокол оптоволоконного интернета
ILM	Information Lifecycle Management	Управление жизненным циклом информации
I/O	Input/Output	Ввод/вывод
IOPS	Input Output Per Second	Операций ввода-вывода в секунду
IP	Internet Protocol	Интернет-протокол
IP-SAN	Internet Protocol Storage Area Network	Сеть хранения и передачи данных по IP-протоколу

IPSec	Internet Protocol Security	Защита интернет-протокола (комплекс алгоритмов, протоколов и процедур, используемых для защиты передаваемых данных посредством аутентификации и/или кодировки каждого пакета в потоке данных)
iQN	iSCSI Qualified Name	Квалификационное имя iSCSI
IRM	Information Rights Management (Information Resource Manager)	Права управления информацией администратор [диспетчер] информационных ресурсов
iSCSI	Internet Small Computer Systems Interface Protocol	Протокол SCSI по IP-сети
iSCSI PDU	ISCSI Protocol Data Unit	Блок данных протокола ISCSI
ISL	Inter-Switch Link	Межкоммутаторный линк
iSNS	Internet Storage Name Server	Интернет-протокол службы имен хранилищ
ISO	International Organization for Standardization	Международная организация стандартизации
ITU	International Telecommunication Union	Международный союз телекоммуникаций
JBOD	Just a Bunch of Disks	Простая, без избыточности, группа жестких дисков в компьютере, которые не представляют собой ни одну из RAID-конфигураций
KDC	Key Distribution Center	Центр распределения ключей
KVM	Keyboard, Video, and Mouse	Клавиатура, видео и мышь
LACP	Link Aggregation Control Protocol	Контрольный протокол объединения каналов. Стандарт IEEE, объединяющий два или более физических канала передачи данных в один логический канал для увеличения доступности
LAN	Local Area Network	Локальная сеть
Lb	Least Blocks	Младшие блоки
LBA	Logical Block Addressing	Логическая адресация блоков
LC	Lucent Connector	Коннектор типа Lucent
LCA	Link Capacity Adjustment	Регулирование пропускной способности линии
LCAS	Link Capacity Adjustment Scheme	Схема регулирования пропускной способности линии
LCC	Link Control Card	Карта управления связью
LDAP	Lightweight Directory Access Protocol	Упрощенный протокол доступа к каталогам
Lo	Least I/Os	Минимальный I/Os
LR	Link Reset	Перезагрузка связи
LRR	Link Reset Response	Ответ на перезагрузку связи
LRU	Least Recently Used	Замещение давно не использовавшихся адресов

LUN	Logical Unit Number	Адрес дискового устройства в сетях хранения
LV	Logical Volume	Логический том
LVDS	Low-Voltage Differential Signaling	Дифференциальные сигналы низкого напряжения
LVM	Logical Volume Management	Управление логическими томами
LWM	Low Watermark	Низкий уровень заполнения
MAC	Media Access Control	Контроль доступа к носителю
MAID	Massive Array of Idle Disks	Массив простаивающих дисков
MAN	Metropolitan Area Network	Городская вычислительная сеть, мегаполисная цифровая (компьютерная) сеть, охватывающая регион диаметром до 50 км
MD5	Message-Digest Algorithm	Алгоритм получения свертки (профиля) сообщения — сжатой текстовой строки, полученной из текста сообщения применением односторонней хэш-функции. Используется для создания цифровой подписи
MIB	Management Information Base	База управляющей информации
MirrorView/A	MirrorView/Asynchronous	Зеркальный вид/Асинхронный
MirrorView/S	MirrorView/Synchronous	Зеркальный вид/Синхронный
MLC	Multi-Level Cell	Многоуровневая ячейка
MMF	Multimode Fiber	Многомодовое оптоволокно
MRU	Most Recently Used	Недавно использованные
MSS	Maximum Segment Size	Максимальный размер сегмента
MTBF	Mean Time Between Failure	Средняя наработка на отказ, среднее время безотказной работы
MTTR	Mean Time to Repair	Среднее время восстановления работоспособности
MTU	Maximum Transfer Unit	Максимальная единица передачи
NACA	Normal Auto Contingent Allegiance	Обычное автоматическое выполнение команд
NAS	Network-Attached Storage	Хранилище, привязанное к сети
NDMP	Network Data Management Protocol	Протокол управления сетевыми данными
NFS	Network File System	Сетевая файловая система
NIC	Network Interface Card	Сетевая интерфейсная плата, сетевой адаптер
NIS	Network Information Services	Информационная служба сети
NMC	NetWorker Management Console	Консоль управления NetWorker
NL_Port	Node Loop Port	Узловой порт с поддержкой петли
N_Port	Node Port	Порт узла

NTFS	New Technology File System	Файловая система новой технологии
NTP	Network Time Protocol	Протокол сетевого времени, протокол синхронизации времени по сетям с коммутацией пакетов и нестабильной сетевой задержкой
OLTP	Online Transaction Processing	Обработка транзакции онлайн
OS	Operating System	Операционная система
OSI	Open System Interconnection	Взаимодействие открытых систем
OTF	Open Tape Format	Открытый формат записи
OXID	Originator Exchange Identifier	Идентификатор запуска обмена
PAgP	Port Aggregation Protocol	Протокол объединения портов
PCI	Peripheral Component Interconnect	Взаимодействие периферийных компонентов
PDU	Protocol Data Unit	Единица обмена протокола
PIT	Point in Time	Копия, содержащая все данные в таком виде, в каком они были в указанный момент времени
PKI	Public Key Infrastructure	Инфраструктура публичного ключа
PRLI	Process Login	Процесс подключения
PV	Physical Volume	Физический том
PVID	Physical Volume Identifier	Идентификатор физического тома
QoS	Quality of Service	Качество обслуживания
RADIUS	Remote Authentication Dial-In User Service	Протокол аутентификации, авторизации и учета для контроля доступа к сетевым ресурсам по коммутируемым линиям
RAID	Redundant Array of Independent Disks	Избыточный дисковый массив
RAIN	Redundant Array of Inexpensive Nodes	Избыточный массив недорогих узлов
RAM	Random Access Memory	Оперативная память
R_A_TOV	Resource Allocation Time-Out Value	Максимальная задержка распределения ресурсов
RBAC	Role-Based Access Control	Ролевой контроль доступа
R_CTL	Routing Control	Контроль маршрутизации
RDBMS	Relational Database Management System	Система управления реляционными базами данных
RFC	Requests for Comments	Запрос комментариев
RLP	Reserved LUN Pool	Зарезервированный пул LUN
ROI	Return on Investment/Information	Возврат инвестиций/информации
ROM	Read-Only Memory	Постоянное запоминающее устройство
RPC	Remote Procedure Call	Вызов удаленных процедур
RPO	Recovery Point Objective	Директивный срок восстановления
RR	Round-Robin	Алгоритм кругового обслуживания

R_RDY	Receiver Ready	Готовность приемника
RSCN	Registered State Change Notification	Уведомление о зарегистрированном изменении состояния
RTD	Round-Trip Delay	Задержка подтверждения приема
RTO	Recovery Time Objective	Директивное время восстановления
R2T	Request to Transfer	Запрос на передачу
R/W	Read/Write	Считывание/запись
RX	Receiver	Приемник
SACK	Selective Acknowledge	Избирательное подтверждение
SACL	System Access Control List	Контрольный список доступа к системе
SAL	SCSI Application Layer	Уровень приложения SCSI
SAN	Storage Area Network	Сеть хранения данных
SAS	Serial Attached SCSI	SCSI с последовательным интерфейсом
SASI	Shugart Associate System Interface	Интерфейс ассоциированной системы Шугарт
SATA	Serial Advanced Technology Attachment	Последовательный интерфейс обмена данными с накопителями информации
SC	Standard Connector	Стандартный коннектор
SCN	State Change Notification	Уведомление об изменении состояния
SCSI	Small Computer System Interface	Интерфейс малой компьютерной системы
SDH	Synchronous Digital Hierarchy	Синхронная цифровая иерархия
SEC	Securities and Exchange Commission	Комиссия по ценным бумагам
SHA	Secure Hash Algorithm	Алгоритм безопасного хэширования
SiS	Single-Instance Storage	Система единичного хранения
SLA	Service Level Agreement	Договор об уровне услуг
SLC	Single-Level Cell	Одноуровневая ячейка
SLED	Single Large Expensive Drive	Одиночный большой и дорогой диск
SLP	Service Location Protocol	Протокол семейства TCP/IP, позволяющий осуществлять автоматическую настройку различных ресурсов
SLV	Symmetrix Logical Volume	Логический том Symmetrix
SMB	Server Message Block	Блок сообщений сервера
SMF	Single-Mode Fiber	Одномодовое оптоволокно
SMI	Storage Management Initiative	Инициатива управления системами хранения данных
SMTP	Simple Mail Transfer Protocol	Простой почтовый протокол передачи
SNIA	Storage Networking Industry Association	Ассоциация отрасли сетевого хранения
SNMP	Simple Network Management Protocol	Простой протокол управления сетью
SNS	Simple Name Server	Простой сервер имен
SOF	Start of Frame	Начало кадра

SONET	Synchronous Optical Networking	Синхронная оптическая сеть
SP	Storage Processor	Процессор системы хранения данных
SPE	Storage Processor Enclosure	Блок процессора системы хранения данных
SPI	SCSI Parallel Interface	Параллельный интерфейс SCSI
SPOF	Single Point of Failure	Отдельная точка сбоя
SPS	Standby Power Supply	Резервный источник электропитания
SRDF	Symmetrix Remote Data Facility	Программное обеспечение для удаленной репликации между дисковыми массивами EMC Symmetrix
SRDF/A	SRDF/Asynchronous	SRDF/Асинхронная
SRDF/	SRDF/Automated Replication	SRDF/Автоматическая репликация
SRDF/CE	SRDF/Cluster Enabler	SRDF/Помощник кластера
SRDF/CG	SRDF/Consistency Groups	SRDF/Согласованная группа
SRDF/DM	SRDF/Data Mobility	SRDF/Мобильность данных
SRDF/S	SRDF/Synchronous	SRDF/Синхронная
SRM	Storage Resource Management	Управление ресурсами хранилища
SSD	Solid-State Drive	Твердотельный накопитель
SSH	Secure Shell	SSH-протокол
SSID	Service Set Identifier	Идентификатор набора служб
SSL	Secure Sockets Layer	Уровень безопасности сокетов
ST	Straight Tip	Сведения из достоверных источников
StatSN	Status Sequence Number	Номер последовательности статуса
STPL	SCSI Transport Protocol Layer	Уровень транспортного протокола SCSI
STS	Synchronous Transport Signal	Синхронный транспортный сигнал
TB	Terabyte	Терабайт
TCO	Total Cost of Ownership	Общая цена владения
TCP	Transmission Control Protocol	Протокол контроля передачи
TGS	Ticket Granting Service	Сервис выдачи билетов
TLU	Tape Library Unit	Запоминающее устройство системы резервного копирования на магнитной ленте
TOE	TCP/IP Offload Engine	Карта обработки TCP/IP
TPI	Tracks per Inch	Треки на дюйм
TX	Transmitter	Передатчик
UDP	User Datagram Protocol	Пользовательский протокол данных
UFS	UNIX File System	Файловая система UNIX
UID	User identifier	Идентификатор пользователя
ULP	Upper-Layer Protocol	Протокол верхнего уровня
URL	Uniform Resource Locator	Унифицированный указатель информационного ресурса
USB	Universal Serial Bus	Универсальная последовательная шина

VC	Virtual Circuit	Виртуальный контур
VCAT	Virtual Concatenation	Виртуальная конкатенация
VG	Volume Group	Группа томов
VLAN	Virtual LAN	Виртуальная LAN
VPN	Virtual Private Network	Виртуальная частная сеть
VSAN	Virtual Storage Area Network	Виртуальная сеть хранения данных
VTL	Virtual Tape Library	Виртуальная ленточная библиотека
WAN	Wide Area Network	Глобальная вычислительная сеть
WBEM	Web-Based Enterprise Management	Управление предприятием на основе веб
WDM	Wavelength-Division Multiplexing	Мультиплексирование по длине волны
WORM	Write Once, Read Many	Однократная запись, многократное считывание
WORO	Write Once, Read Occasionally	Однократная запись, случайное считывание
WWN	World Wide Name	Имя в глобальной сети
WWNN	World Wide Node Name	Имя узла в глобальной сети
WWPN	World Wide Port Name	Имя порта в глобальной сети
XCM Environmental	Environmental Control Module	Контрольный модуль окружения
XML	Extensible Markup Language	Расширяемый язык разметки

Глоссарий

8b/10b encoding	Кодировка 8b/10b	Алгоритм, преобразующий 8-битные данные в 10-битный код без длинных последовательностей нулей или единиц
64b/66b encoding	Кодировка 64b/66b	Алгоритм, преобразующий 64-битные данные в 66-битный код без длинных последовательностей нулей или единиц
Access control	Контроль доступа	Службы управления доступом пользователя к ресурсам
Access Control List (ACL)	Список контроля доступа	Список разрешений, определяющий лиц, которым доступен ресурс, и их привилегии
Accessibility	Доступность	Возможность доступа авторизованным пользователям к запрашиваемой в нужном месте информации
Accountability services	Службы ведения учета	Служба, позволяющая администраторам отслеживать системную активность и выявлять осуществляющих эту активность лиц таким образом, что отрицание ответственности за совершенные действия становится практически невозможным
Active archive	Активный архив	Категория данных, почти или полностью не подлежащая изменениям, часто называемая фиксированным контентом
Active attack	Активная атака	Несанкционированное изменение информации, которое может угрожать целостности и доступности данных
Active changeable	Активные переменные данные	Категория часто изменяемых данных. Данные этой категории называют переменными данными

Active Directory (AD)	Active Directory	Продукт компании Microsoft для обеспечения централизованной аутентификации и авторизации
Active path	Активный путь	Путь, доступный в настоящее время и активно использующийся для операций ввода-вывода
Active/active	Активный — активный	Архитектура, предназначенная для обеспечения высокой доступности, в которой все компоненты активны и доступны для выполнения задачи в случае сбоя другого компонента
Active/passive	Активный — пассивный	Архитектура, предназначенная для обеспечения высокой доступности, в которой резервные компоненты находятся в состоянии ожидания выполнения задачи в случае отказа активного компонента
Actuator arm assembly	Актуатор	Устройство в накопителе для перемещения головок чтения-записи над поверхностью диска
Advanced Encryption Standard (AES)	Улучшенный стандарт шифрования	Алгоритм блочного шифрования, утвержденный Национальным институтом стандартов и технологий США (NIST)
Alert	Предупреждение	Уведомление о событии, требующем или не требующем внимания/вмешательства в зависимости от типа предупреждения
Application	Приложение	Компьютерная программа, которая обеспечивает логику выполнения вычислительных операций
Application Programming Interface (API)	Интерфейс прикладного программирования	Набор функциональных обращений, устанавливающих связь между приложениями или между приложением и операционной системой
Application virtualization	Виртуализация приложений	Способ упаковки приложений в контейнер для обеспечения изолированности и переносимости
Arbitrated Loop (AL)	Управляемая петля	Общая петля Fibre Channel, в рамках которой устройства конкурируют друг с другом за выполнение операций ввода-вывода; архитектура аналогична компьютерной сети Token Ring
Arbitration	Арбитраж	Способ определения того, какой узел получит контроль в системе управляемой петли (FC-AL) в случае попытки осуществления передачи данных несколькими узлами одновременно
Archive	Архив	Хранилище, в котором размещается фиксированный контент для длительного хранения

Array/disk array/ storage array	Массив/дисковый массив/ массив хранения данных	Группа накопителей на жестких дисках, работающих как единое целое
Asynchronous replication	Асинхронная репликация	Режим репликации, при котором введенная информация тут же посылается узлу-источнику. Эта информация последовательно записывается, передается и обновляется для узла-источника. В рамках SRDF копирование называется асинхронным режимом
Attack surface	Поверхность атаки	Общее количество возможных уязвимых мест
Attack vector	Вектор атаки	Серия шагов, необходимых для завершения атаки
Authentication	Аутентификация	Процесс установления подлинности лица, запрашивающего доступ
Authorization	Авторизация	Процесс определения прав доступа к ресурсам инициатора запроса
Automatic path failover	Автоматическое переключение путей	Автоматическое бесперебойное переключение пути передачи данных на альтернативный при его повреждении без прерывания операций уровня приложения
Availability	Доступность	Степень доступности компонента и возможности его функционирования согласно ожиданиям
Availability services	Службы доступности	Службы, гарантирующие авторизованным пользователям надлежащий и своевременный доступ к данным
Average queue size	Средняя длина очереди	Среднее число запросов в очереди
Average rotational latency	Средняя задержка при вращении	Половина времени, уходящая на полную прокрутку диска накопителя
Backup	Резервная копия	Копия данных
Backup catalog	Каталог резервного копирования	База данных, содержащая информацию о процессах резервного копирования и метаданные
Backup client	Программа резервного копирования	Программа для извлечения данных из рабочего узла и их отправки на узел хранения для сохранения резервной копии

Backup server	Сервер резервного копирования	Сервер, осуществляющий резервное копирование и хранящий каталог резервирования
Backup to disk	Резервное копирование на диск	Использование дисков для хранения резервных данных
Backup window	Окно резервного копирования	Интервал времени, в который источник доступен для проведения резервного копирования
Bandwidth (network)	Ширина полосы пропускания	Характеристика канала передачи данных: максимальное количество данных, передаваемых за единицу времени; измеряется в мегабитах в секунду (Мбит/с)
Bare-metal recovery (BMR)	Восстановление системы с нуля	Полное резервное копирование всех метаданных, системной информации и конфигурации приложений для полного восстановления системы
Battery Backup Unit (BBU)	Устройство аварийного батарейного питания	Батарейный резервный источник питания, используемый при сбое электросети
BB_Credit	BB_Credit	Определяет максимальное количество пакетов данных, передаваемых по каналу связи за единицу времени
BC Planning (BCP)	Планирование непрерывности бизнеса	Систематизированный подход, обеспечивающий функционирование компании во время и после аварии
Big data	Большие данные	Настолько большие наборы данных, что это делает неудобным манипулирование ими с помощью традиционных инструментов
Binary Large Object (BLOB)	Большой двоичный объект	Двоичная последовательность данных пользователя, представляющая фактическое содержание файла. Не зависит от имени и физического месторасположения файла
Bit	Бит	Базовая единица информации, может существовать в одном из двух возможных состояний.
Block	Блок	Единица выделения пространства на дисковом носителе данных. Имеет фиксированный объем
Block-level virtualization	Виртуализация блочного уровня	Уровень абстракции в сети хранения данных (SAN) между узлами и массивами хранения данных

Block size	Размер блока	Базовая единица для хранения и извлечения данных
Bridged topology	Схема мостовых соединений	Схема, объединяющая оптоволоконную сеть с IP-сетью
Broad network access	Широкополосный сетевой доступ	Ресурсы сети, доступные с использованием стандартных механизмов, которые позволяют применять гетерогенные тонкие или толстые клиентские платформы
Broadcast	Широковещательная передача	Одновременная передача сообщения всем приемникам
Buffer	Буфер	Область кратковременного хранения, как правило, в ОЗУ
Bunker site	Бункерная площадка	Промежуточная площадка между основной (рабочей) и удаленной. Используется при каскадной/мультитранзитной трехузловой репликации для уменьшения рисков, возникающих при двухузловой репликации
Bus	Шина	Набор каналов передачи данных, по которым осуществляется обмен информацией между аппаратными устройствами внутри компьютера
Business continuity (BC)	Непрерывность работы бизнеса	Готовность к восстановлению данных и реагированию на аварийные ситуации, которые могут навредить бизнесу
Business Impact Analysis (BIA)	Анализ воздействия на бизнес	Оценка результатов временной приостановки бизнес-процессов
Byte	Байт	Единица информации — 8 двоичных цифр
Cache	Кэш-память	Полупроводниковая память для временного размещения данных с целью минимизации времени обработки запросов на ввод-вывод данных с основного узла
Cache coherency	Синхронизация кэш-памяти	Две копии данных на разных адресах в кэш-памяти, идентичность которых постоянно поддерживается
Cache mirroring	Зеркальное кэширование	Каждая запись в кэш-память осуществляется на два разных адреса, расположенные на двух независимых платах памяти
Cache vaulting	Сохранение кэшированной информации	Процесс сброса содержимого кэш-памяти на специально выделенные физические диски при сбое электропитания

Call home	«Звонок домой»	Автоматическая отправка сообщения в центр поддержки поставщика при сбоях оборудования или процесса
Capacity management	Управление дисковым пространством	Обеспечение адекватного выделения ресурсов всем приложениям в соответствии с требуемым уровнем обслуживания
Capital Expenditure (CAPEX)	Капитальные затраты	Средства, израсходованные на физические ресурсы
Carrier Sense Multiple Access/ Collision Detection (CSMA/CD)	Множественный доступ с контролем несущей/ обнаружение конфликтов	Набор правил, определяющих реакцию сетевых устройств при попытке одновременного использования канала данных двумя устройствами (при конфликте устройств)
Cascade/Multihop Replication	Каскадная/ мультитранзитная репликация	Копирование данных с источника на промежуточный массив хранения данных, или бункер, при первом транзите, а затем на массив хранения данных удаленного узла при втором транзите
Challenge-Handshake Authentication Protocol (CHAP)	Протокол аутентификации по методу «вызов-приветствие»	Протокол, используемый инициатором и целевым устройством для взаимной аутентификации при помощи обмена секретным кодом или паролем
Channel	Канал	Высокоскоростная линия обмена данными между процессором и другими процессорами или устройствами
Chargeback report	Отчет о пользовании ресурсами	Отчет, позволяющий администраторам системы хранения данных определить количество ресурсов, потребленных каждым приложением/подразделением, для того чтобы распределить расходы на поддержание системы хранения данных между приложениями/подразделениями
Checksum	Контрольная сумма	Контроль избыточным кодом для проверки целостности данных путем выявления ошибок при передаче данных
C-H-S addressing	Адресация цилиндров, головок и секторов	Использование физических адресов, состоящих из номеров цилиндра, головки и сектора, для размещения информации по определенным ячейкам диска
Cipher	Шифр	Метод, при котором произвольные символы представляют блоки обычного текста

Class of Service (CoS)	Класс предоставления сервиса	FC-стандарты оценивают качество предоставляемых сетевых услуг (QoS), определяя каждый класс с помощью сервисного приоритета
Client-initiated backup	Резервное копирование по инициативе клиента	Ручной/автоматический процесс резервного копирования по инициативе клиента
Client/server model	Модель «клиент — сервер»	Модель запроса и использования клиентом услуг, предоставляемых сервером. Сервер способен обслуживать многих клиентов одновременно
Cloud	Облако	Модель, которая обеспечивает удобный сетевой доступ по требованию к общему пулу настраиваемых вычислительных ресурсов (например, сетей, серверов, систем хранения, приложений и сервисов), которые можно быстро выделять и высвобождать с минимальными издержками на управление или обслуживание с минимальным участием поставщика
Cloud scale	Масштабирование облака	Облако может обеспечить неограниченную масштабируемость для нужд конечных пользователей
Cloud service provider	Поставщик облачных услуг	Лицо, организация или объект, отвечающий за предоставление услуги потребителям облачных услуг
Cold backup	Холодное резервное копирование	Резервное копирование, требующее прекращения работы с данными на время копирования
Cold site	Площадка холодного резерва	Помещение, в которое в случае аварийной ситуации может быть перенесен рабочий процесс. Обеспечено минимальной IT- и прочей инфраструктурой. В нормальных условиях не используется
Command Line Interface (CLI)	Интерфейс командной строки	Пользовательский интерфейс приложения, принимающий набранные команды построчно в окне ввода команд
Command queuing	Очередь команд	Алгоритм, оптимизирующий порядок выполнения принятых команд
Common Information Model (CIM)	Common Information Model	Объектно-ориентированное описание объектов и взаимосвязей в среде управления бизнеса, поддерживаемое группой Distributed Management Task Force

Common Internet File System (CIFS)	Общая межсетевая файловая система	Клиент-серверный протокол компании Microsoft, позволяющий программам-клиентам запрашивать файлы и услуги с удаленных компьютеров по протоколу TCP/IP
Common Management Information Protocol (CMIP)	Протокол общей управляющей информации	Сетевой протокол управления, построенный на основе модели взаимодействия открытых систем (OSI)
Common Management Information Service (CMIS)	Общие службы передачи управляющей информации	Служба, используемая сетевыми элементами для управления сетью
Community cloud	Общественное облако	Облачная инфраструктура, совместно используемая несколькими организациями и поддерживающая определенное общество, со схожими задачами (например, выполняемая задача, требования к безопасности, политики, совместимости). Она может обслуживать организации или стороннего производителя и существовать на локальной или удаленной площадке
Compliance	Соответствие	Соблюдение требований государственных и отраслевых регуляторов
Compute (Host or Server)	Вычислитель (хост или сервер)	Вычислительная платформа, обеспечивающая работу приложений и баз данных
Concatenation	Конкатенация	Процесс логического соединения адресных пространств дисков и представления их единым адресным пространством большего объема
Confidentiality	Конфиденциальность	Обеспечение необходимой секретности информации
Configuration Management Database (CMDB)	База данных управления конфигурациями	База данных, содержащая информацию о компонентах информационной системы
Congestion Notification (CN)	Уведомление о перегрузке	Механизм обнаружения перегрузки и уведомления источника о необходимости перенести поток трафика с перегруженных каналов
Consistency group	Согласованная группа	Группа логических устройств, расположенная на одном или нескольких массивах хранения данных и управляемая как единое целое

Console	Терминал (консоль)	Основной интерфейс для просмотра, конфигурирования и обработки отчетов различных компонентов (управляемых объектов), а также управления ими
Content Address (CA)	Контентный адрес	Идентификатор, указывающий на содержимое файла, а не на его физическое расположение
Content Addressed Storage (CAS)	Система хранения данных с контентной адресацией	Объектно-ориентированная система хранения данных с фиксированным контентом. Рентабельный вариант сетевого хранения данных
Content authenticity	Подлинность контента	Достигается в два этапа: генерированием уникального контентного адреса и автоматизацией процесса его постоянной проверки и пересчета
Content Protection Mirrored (CPM)	Защита контента копированием	Зеркальное копирование объекта данных для защиты данных при отказе оборудования
Content Protection Parity (CPP)	Защита контента контролем четности	Преобразование данных в сегменты с дополнительным сегментом контроля четности для защиты данных при отказе оборудования
Continuous Data Protection (CDP)	Непрерывная защита данных	Технология, где точки восстановления или контрольные точки расположены достаточно часто для того, чтобы данные могли быть восстановлены без существенных потерь
Control station	Управляющая станция	Вычеслительное устройство осуществляет специальные операции по контролю, управлению и конфигурации системы NAS
Converged Enhanced Ethernet (CEE)	Конвергентный усовершенствованный стандарт Ethernet	Спецификация для существующих стандартов Ethernet, которая исключает наличие потерь, свойственных Ethernet
Converged network adapter (CNA)	Конвергентный сетевой адаптер	Технология, поддерживающая передачу данных сетей (TCP/IP) и сетей хранения данных (Fibre Channel) на одном адаптере ввода-вывода
Copy on First Access (CoFA)	Копирование при первом доступе	Метод полной репликации, копирующий данные с основного узла на целевой узел при запуске процесса записи на основном узле или при первичном выполнении операции чтения-записи на целевом узле. Копия сразу же становится доступной при запуске сессии

Copy on First Write (CoFW)	Копирование при первой записи	Метод виртуальной репликации, при котором копирование данных в предопределенную область массива производится в случае первичной записи на основной или целевой узел
Cryptography	Криптография	Техника шифрования информации в целях безопасности
Cumulative backup (differential backup)	Кумулятивное (дифференциальное) резервное копирование	Копирование данных, измененных с момента последнего полного резервного копирования
Cyclic redundancy check (CRC)	Контроль циклическим избыточным кодом	Метод обнаружения ошибок в цифровых данных и проверки целостности данных. При этом методе определенное количество контрольных разрядов, часто называемых контрольной суммой, добавляют к передаваемому сообщению
Cylinder	Цилиндр	Совокупность дорожек, равноотстоящих от центра, на всех рабочих поверхностях пластин жесткого диска
Data	Данные	Записанная информация
Data Access in Real Time (DART)	Доступ к данным в реальном времени	Специальная операционная система для систем Celerra, запускаемая на серверах Data Mover
Data center	Центр обработки данных	Предприятие или подразделение, которое предоставляет услуги централизованной обработки данных. Основные составляющие информационного центра — приложения, базы данных, операционные системы, сети и устройства хранения
Data Center Bridging (DCB)	Data Center Bridging	Пакет расширения протокола Ethernet, определяющий надежность транспорта в сетях хранении данных
Data Center Bridging exchange Protocol	Протокол Data Center Bridging exchange	Протокол обмена, обнаружения и настройки функциональности, позволяющий устройствам Converged Enhanced Ethernet (CEE) передавать и настраивать свои функции вместе с другими устройствами CEE в сети
Data compression	Сжатие данных	Процесс кодировки информации, сокращающий ее объем
Data consistency	Согласованность данных	Удобство применения, достоверность и целостность связанных компонентов данных

Data Encryption Standard	Стандарт шифрования данных	Криптографический алгоритм, опубликованный Национальным институтом стандартов и технологий США (NIST)
Data integrity	Целостность данных	Гарантия того, что данные не подверглись случайным изменениям
Data security	Защита данных	Способ обеспечения безопасности данных как от повреждения, так и от несанкционированного доступа
Data shedding	Отбрасывание разрядов данных	Процесс удаления данных, что делает их невосстанавливаемыми
Data store	Хранилище данных	Раздел кэш-памяти, содержащий данные
Data tampering	Подделка данных	Преднамеренное изменение данных
Data transfer rate	Скорость передачи данных	Объем данных в секунду, который диск может передать на контроллер
Database Management System (DBMS)	Система управления базами данных	Программа, обеспечивающая структурированный способ хранения данных в логически организованых и взаимосвязанных таблицах
Defense in depth	Глубокая защита	Осуществление контроля безопасности в каждой точке доступа всех путей доступа
Delta set	Разностный набор	При выполнении асинхронной репликации используются большие объемы кэш-памяти при хранении данных для временной буферизации незавершенных записей для целевого узла. Буферизованные данные отображают разницу, или разностный набор, между записями главного и целевого узлов
Demilitarized zone (DMZ)	Демилитаризованная зона	Хост или сеть, используемые в качестве буфера между частной сетью организации и внешней общедоступной сетью
Denial-of-Service (DoS) attack	Атака типа «отказ в обслуживании»	Атака, лишающая авторизованных пользователей возможности пользоваться системой и ее ресурсами
Dense Wave Division Multiplexing (DWDM)	Мультиплексирование (уплотнение) по длине волны высокой плотности	Технология одновременной передачи данных из разных источников по одному оптоволоконному кабелю. Для разных потоков данных используются волны разной длины

Desktop-as-a-Service (DaaS)	Рабочее место как сервис	Аутсорсинг виртуальных рабочих мест (VDI) для поставщика услуг других производителей. Как правило, услуга DaaS имеет многопользовательскую архитектуру и приобретается по подписке. В этой модели предоставления услуг поставщик услуг управляет серверной частью хранения данных, резервным копированием, безопасностью и модернизацией. Персональные данные заказчика копируются в виртуальное рабочее место и из виртуального рабочего места во время входа и выхода из системы и доступа к рабочему месту — независимо от устройства, местоположения и сети
Desktop virtualization	Виртуализация рабочих мест	Удаленный экран, хостинг или операции с графическим адаптером (рабочее место)
Device driver	Драйвер устройства	Специальная программа, обеспечивающая взаимодействие операционной системы с аппаратным обеспечением компьютера
Diffie-Hellman Challenge Handshake Authentication Protocol (DH-CHAP)	Протокол аутентификации по методу «вызов-приветствие»	Протокол безопасного обмена ключами, который обеспечивает аутентификацию между инициатором Fibre Channel и ответчиком
Direct-attached backup	Резервное копирование при прямом подключении	Устройство резервного копирования, подключаемое непосредственно к клиенту резервного копирования
Direct-attached storage (DAS)	Система хранения прямого подключения	Система хранения данных, напрямую подключаемая к серверу или рабочей станции
Director (Switch)	Коммутатор уровня «центр»	Класс соединительных устройств с большим количеством портов и резервными компонентами для построения сетей масштаба предприятия
Directory	Директория	Контейнер файловой системы, содержащий ссылки на разные файлы
Directory service (DS)	Служба каталогов	Приложение или набор приложений, хранящий и систематизирующий информацию о сетевых пользователях и ресурсах. Эта служба позволяет системным администраторам управлять доступом пользователей к ресурсам
Directory System Agent (DSA)	Агент системы каталогов	Каталог LDAP может быть распространен по многим серверам. У каждого агента системы каталогов имеется периодически синхронизируемая копия полного каталога

Disaster recovery	Восстановление после аварии	Процесс, методики и процедуры восстановления необходимых для возобновления бизнеса операций, включая восстановление доступа к данным
Disaster recovery plan (DRP)	План восстановления после аварии	План копирования данных при неожиданной или внезапной потере доступа к данным с упором на защиту данных. Входит в планирование непрерывного ведения бизнеса
Disaster restart	Аварийный перезапуск	Процесс перезапуска бизнес-операций с использованием сохранной копии данных
Discovery domain	Домен обнаружения	Объединяет устройства IP-сети хранения данных (IP SAN) в функциональную группу. Для обеспечения взаимодействия устройств друг с другом нужно конфигурировать их в рамках одного домена обнаружения
Discretionary Access Control (DAC)	Произвольный контроль доступа	Правила доступа, устанавливаемые владельцем объекта
Disk-buffered replication	Репликация с дисковой буферизацией	Сочетание технологий локальной и удаленной репликации; сначала создает локальную PIT-копию, а затем удаленную копию локальной PIT-копии
Disk drive (HDD)	Дисковый накопитель	Периферийное устройство для постоянного хранения данных
Disk image backup	Резервное копирование образов дисков	Резервное копирование, состоящее из копий каждого блока, содержащихся в области полезной емкости диска
Disk partitioning	Сегментирование диска	Создание логических разделов на жестком диске
Distributed computing	Распределенные вычисления	Любые вычисления, которые производятся на нескольких компьютерах, удаленных друг от друга, при этом каждый имеет роль в вычислительной задаче или обработке информации
Distributed file system (DFS)	Распределенная файловая система	Файловая система, распределяемая по нескольким узлам сети
Distributed Management Task Force (DMTF)	Подразделение проблем управления настольными системами	Организация, разрабатывающая стандарты управления для компьютерных систем и корпоративных сред

Domain ID	Идентификатор домена	Уникальный идентификатор, назначенный для каждого коммутатора (domain) в системе коомутации
Domain Name System (DNS)	Система доменных имен	Служба, которая устанавливает соответствие между легко читаемыми доменными именами и IP-адресами
Downtime	Время простоя	Период времени, в течение которого система недоступна
Dual-role	Узел с двойной функцией	Узел, выполняющий функции узла хранения и узла доступа
Dynamic Host Configuration Protocol (DHCP)	Протокол динамического конфигурирования узла	Система автоматического присвоения IP-адреса узлу
Elasticity	Гибкость	Быстрая и безопасная реакция на изменяющиеся требования к ресурсам
Encryption	Кодирование	Процесс преобразования информации посредством алгоритма (называемого шифром), делающий информацию недоступной для прочтения неавторизованными пользователями
End-to-End Credit (EE-Credit)	Разрешение на сквозную передачу пакетов данных	Механизм, контролирующий поток данных для использующих трафик буферов классов 1 и 2
Enterprise management platform (EMP)	Платформа управления уровня предприятия	Интегрированные приложения или набор приложений, контролирующих и управляющих средой информационного центра
Enterprise Resource Management (ERM)	Управление ресурсами предприятия	Программное обеспечение, управляющее всеми аспектами ресурсов, сервисов и функций организации
Enterprise Systems Connection (ESCON)	Средства связи систем предприятия	Оптический последовательный интерфейс между центральными компьютерами IBM и периферийными устройствами
Error-correction coding (ECC)	Кодирование с исправлением ошибок	Метод кодирования, обнаруживающий и исправляющий ошибки при передаче данных
Expansion port (E_Port)	Порт расширения	Порт для подключения двух FC-коммутаторов через протокол межкоммутационного канала (ISL)

Export	Экспорт	Предоставление файловой системы для клиентов с операционной системой UNIX, которые могут установить или получить доступ к удаленной файловой системе
eXtensible Markup Language (XML)	Расширяемый язык разметки	Универсальный формат для структурированных документов и данных в World Wide Web
Extent	Экстент	Набор последовательно адресуемых дисковых блоков, который является частью единичного виртуального диска
External transfer rate	Внешняя скорость передачи	Скорость передачи данных по интерфейсу к HBA
Fabric	Система коммутации	Fibre Channel-топология с одним или несколькими коммутаторами
Fabric Login (FLOGI)	Регистрация в системе коммутации	Регистрация между портами Fibre Channel, N_Port и F_port
Fabric Loop port (FL_Port)	Порт-шлюз	Порт коммутатора, подключаемый к управляемой петле Fibre Channel
Fabric port (F_Port)	Порт системы коммутации	Порт коммутатора, подключаемый к N-порту
Fabric Shortest Path First (FSPF)	Протокол кратчайшего пути (FSPF-протокол)	Используемый в Fibre Channel-сетях протокол маршрутизации, вычисляющий кратчайший путь между узлами
Failback	Восстановление после отказа	Операция, возобновляющая стандартные процессы. Восстановление после отказа запускается после начала обработки отказа
Failover	Аварийное переключение	Функция автоматического переключения на резервный компонент после сбоя активного компонента
Fan-in	Нагрузочная способность по входу	Количество портов устройств хранения, к которым можно получить доступ посредством одного инициатора в сети хранения данных
Fan-out	Нагрузочная способность по выходу	Количество инициаторов, которые могут получить доступ к одному порту устройства хранения в сети хранения данных
Fatal alert	Предупреждение о критической ситуации	Предупреждение с требованием немедленного реагирования о состоянии, которое может нарушить общую производительность или доступ к системе

Fault tolerance	Отказоустойчивость	Свойство системы или компонента, разработанных для бесперебойной работы методом включения резервного компонента или процедуры в случае отказа основных элементов
FCoE Forwarder (FCF)	Ретранслятор FCoE	Коммутирующий элемент Fibre Channel, который инкапсулирует пакеты Fibre Channel, полученные от порта Fibre Channel, в пакеты FCoE, а также декапсулирует пакеты FCoE, полученные от моста Ethernet, в пакеты Fibre Channel
Federated database	Интегрированная база данных	Набор баз данных, рассматриваемых как единая база, представленная в единичном пользовательском интерфейсе
Federation	Объединение	Различные хранилища данных в других местах, которые позволяют организациям беспрепятственно перемещать рабочие нагрузки
Fibre Channel (FC)	Оптоволоконный канал	Внутреннее соединение с поддержкой многочисленных протоколов и топологий. Происходит последовательная высокоскоростная передача данных по многочисленным как медным, так и оптическим линиям связи
Fibre Channel Industry Association (FCIA)	Отраслевая ассоциация Fibre Channel	Международное взаимовыгодное некоммерческое объединение производителей, системных интеграторов, разработчиков, поставщиков, отраслевых специалистов и конечных пользователей. Оно обеспечивает обширной основой для технологий инфраструктур Fibre Channel для поддержки широкого спектра приложений хранения данных и решений на базе IT
Fibre Channel over Ethernet (FCoE)	Fibre Channel поверх Ethernet	Стандарт использования протокола Fibre Channel по сетям Ethernet
Fibre Channel over IP protocol (FCIP)	Технология передачи данных Fibre Channel по IP-сетям	Туннельный протокол на базе протокола TCP/IP для подключения к оптоволоконной сети хранения данных через IP-сеть
Fibre Channel Protocol (FCP)	Протокол Fibre Channel	Транспортный протокол передачи SCSI-команд по сети Fibre Channel
Fibre Channel Security Protocol (FCSP)	Протокол безопасности в сетях Fibre Channel	Стандарт ANSI, описывающий применение функций обеспечения безопасности протокола в системах коммутации Fibre Channel
Fibre Connect (FICON)	Последовательный канал передачи данных	Высокоскоростной интерфейс ввода-вывода для соединения мэйнфреймов и устройств хранения

Fiber Distributed Data Interface (FDDI)	Волоконно-оптический интерфейс передачи данных	Стандарт ANSI для сетей Token Ring, основанный на использовании оптоволоконных кабелей для передачи данных со скоростью 100 Мбит/с
Field-Replaceable Unit (FRU)	Сменный узел	Системный компонент, который может быть заменен специалистом компании-поставщика
File-level access	Файловый доступ	Уровень абстракции при доступе блочного уровня, скрывающий от приложения детали адресации блоков
File-level virtualization	Файловая виртуализация	Механизм, обеспечивающий независимость данных, доступных на файловом уровне, от места, где физически располагаются файлы
File Transfer Protocol (FTP)	Протокол передачи файлов (FTP-протокол)	Сетевой протокол семейства TCP/IP, предназначенный для передачи файлов между компьютерами в Интернете
File server	Файловый сервер	Сервер, используемый для обмена файлами
File system	Файловая система	Способ организации и хранения данных в виде файлов
Firewall	Межсетевой экран	Специальное устройство или программа, проверяющие проходящий через него сетевой трафик и разрешающие или запрещающие прохождение этого трафика по определенным правилам
Firmware	Встроенное ПО	Программное обеспечение, загруженное производителем или встроенное в устройство
Fixed content	Фиксированный контент	Данные, которые не изменяются на протяжении своего жизненного цикла
Flash drives	Флеш-накопители	Устройства хранения данных, в которых используется память на основе полупроводников для хранения данных
Flow control	Управление потоком	Позволяет активировать управление трафиком сети для согласования пропускной способности устройства для отправки и приема данных
Flushing	Сброс кэш-памяти	Процесс записи данных из кэш-памяти на диск
Formatting	Форматирование	Процесс подготовки жесткого диска для хранения данных путем записи необходимой информации на диск

Force flushing	Принудительный сброс кэш-памяти	В случае ввода-вывода большого объема данных происходит принудительный сброс на диск всех модифицированных страниц кэш-памяти
Frame	Кадр	Поток данных, кодируемый канальным уровнем передачи данных для цифровой передачи по межузловой линии связи
Front-end controller	Входной контроллер	Принимает и обрабатывает команды ввода-вывода информации от главного узла и обеспечивает связь с кэш-памятью или системой хранения данных
Front-end port	Внешний порт	Обеспечивает взаимосвязь между системой запоминающих устройств и главным узлом или соединительными устройствами (коммутатором или управляющим устройством)
Full backup	Полное резервное копирование	Копирование всех данных с источника на устройство резервного копирования
Full duplex	Полный дуплекс	Одновременные передача и прием данных в одном канале
Full restore	Полное восстановление	Копирование всех данных с целевого узла на источник. Все данные целевого узла записываются поверх данных источника
Full stroke	Полный цикл	Время передвижения головки чтения-записи по всей ширине диска, от внутренней дорожки до внешней
Full virtualization	Полная виртуализация	Полная имитация аппаратного обеспечения, позволяющая программному обеспечению, как правило, гостевой операционной системе, запускаться в неизмененном виде. Гипервизор выступает посредником между ОС хоста и гостевой ОС
Full-volume mirroring	Зеркальное копирование целого тома	Целевой узел подсоединяется к источнику и устанавливается как зеркало источника. Зеркалирование производится при помощи копирования всех данных и синхронного обновления целевого узла при каждой записи на источник
Gateway NAS	Шлюз сетевой системы хранения	Устройство, состоящее из автономной головной части NAS и одного или нескольких массивов хранения данных
Generic Framing Procedure (GFP)	Процедура стандартного кадрирования	Способ мультиплексирования, позволяющий кодировать данные переменной длины для передачи кадрами постоянной длины

Gigabit Ethernet (GbE)	Gigabit Ethernet (GbE)	Группа стандартов Ethernet для передачи данных со скоростью 1 Гбит/с
Gigabit Interface Converter (GBIC)	Преобразователь гигабитного интерфейса	Трансивер, преобразующий электросигналы в оптические сигналы и наоборот
Global namespace	Глобальное пространство имен	Устанавливает соответствие имен логических путей и их физического месторасположения
Gold copy	Золотая копия	Копия скопированного устройства, сделанная до перезапуска приложений с применением скопированного устройства
Governance	Управление	Правила, процессы или законы функционирования и регулирования деятельности
Governance, Risk, and Compliance (GRC)	Управление, риски и соответствие требованиям регуляторов	Правила и нормы для соответствия требованиям государственных регуляторов и бизнеса и связанные с этим оценки рисков
Graphical User Interface (GUI)	Графический интерфейс пользователя	Интерфейс ввода команд в компьютер с помощью указательного устройства, например мыши, управляющей и активирующей графические изображения на мониторе
Grid computing	Распределенные вычисления	Одновременное использование ресурсов множества компьютеров в сети для обработки одной задачи
Guest operating system	Гостевая операционная система	Операционная система, которая установлена на виртуальной машине
Hard disk drive (HDD)	Жесткий диск	Устройство длительного хранения, содержащее данные в цифровом формате и использующее быстро вращающиеся диски с магнитными поверхностями
Hardware assist virtualization	Аппаратная поддержка виртуализации	Технология виртуализации, которая позволяет процессору компьютера виртуализировать инструкции для снижения нагрузки на оборудование системы
Heartbeat	Тактовый импульс	Механизм передачи сообщений, используемый приложением MirrorView для определения восстановления работоспособности вторичного устройства после того, как установлена его недоступность

Heterogeneous	Гетерогенный	Сборка и согласование различных аппаратных и программных систем для единого представления
Hierarchical Storage Management (HSM)	Управление иерархической системой хранения данных	Механизм управления, позволяющий перемещать данные с дорогих устройств хранения на дешевые
High availability	Высокая готовность	Гарантия сохранения всех данных в случае аварии источника
High Performance Computing (HPC)	Высокопроизводительные вычисления	Использование параллельной обработки для выполнения прогрессивных прикладных программ эффективно, надежно и быстро
High Performance Parallel Interface (HIPPI)	Высокоскоростной параллельный интерфейс	Высокоскоростная компьютерная шина для подключения устройства хранения данных
High watermark	Высокий уровень заполнения	Уровень заполнения кэш-памяти, при котором система начинает быстрый сброс кэш-памяти на диск
Host	Главный узел, хост	Компьютер — сервер или рабочая станция, на котором запускаются приложения
Host bus adapter (HBA)	Адаптер шины на узле	Аппаратное устройство, используемое для подключения компьютера к сети хранения данных или непосредственно к устройству хранения данных
Hot backup	Оперативное резервное копирование	Резервное копирование данных при запущенном и активном приложении, с которым работают пользователи
Hot site	Горячее резервное помещение	Компьютерное помещение с необходимыми аппаратными устройствами, операционной системой, приложением, сетевой поддержкой для выполнения операций по делопроизводству в случае аварии или недоступности приложения
Hot spare	Горячий резервный диск	Незадействованный накопитель, замещающий неисправный накопитель в работающем защищенном RAID-массиве
Hot swap	Горячая замена	Замена компонента аппаратных устройств компьютера аналогичным компонентом при включенном компьютере
Hub	Хаб, концентратор	Соединительное устройство, подключающее узлы к логическому циклу с общей полосой пропускания для всех узлов

Hybrid cloud	Гибридное облако	Облачная инфраструктура — комбинация двух или более облаков (частного, сообщества или публичного), которые остаются уникальными объектами, но связаны между собой стандартной или закрытой технологией, которая обеспечивает возможность переноса данных и приложений (например, резкое увеличение размера облака для балансировки нагрузки между облаками)
HyperText Markup Language (HTML)	Язык гипертекстовой разметки	Компьютерный язык, состоящий из набора тегов, который описывает, как документ отображается в веб-браузере
HyperText Transfer Protocol (HTTP)	Протокол передачи гипертекста	Протокол уровня приложений, работающий обычно поверх TCP/IP, который обеспечивает обмен файлами посредством Интернета
Hypervisor	Гипервизор	Платформа виртуализации, которая позволяет операционным системам работать параллельно на физическом хост-компьютере. Гипервизор несет ответственность за взаимодействие непосредственно с физическими ресурсами компьютера
Idle flushing	Сброс в фоновом режиме	Неторопливая запись данных из кэш-памяти на диск при среднем уровне заполнения кэш-памяти
In-band	Внутриполосная технология	Реализация, при которой конфигурации виртуализованной среды находятся в самом информационном канале
In-sync	Внутренняя синхронизация	Означает, что основное и второстепенное логические устройства содержат идентичные данные
Incremental backup	Инкрементное резервное копирование	Копирование данных, изменившихся с момента последнего полного или инкрементного резервного копирования
Information	Информация	Извлеченные из данных знания
Information Lifecycle Management (ILM)	Управление жизненным циклом информации	Проактивная и динамичная стратегия, помогающая бизнесу управлять растущими объемами информации на основе их коммерческой значимости
Information Rights Management (IRM)	Управление правами доступа к информации	Технология защиты конфиденциальной информации от несанкционированного доступа; иногда эту технологию называют управлением цифровыми правами уровня предприятия

Information Technology Infrastructure Library (ITIL)	Библиотека инфраструктур информационных технологий	Совокупность передовых практик для управления IT-услугами
Infrastructure-as-a-Service	Инфраструктура как сервис	Ресурсы, предоставляемые потребителю для обеспечения обработки, хранения, сетей и других фундаментальных вычислительных ресурсов, где потребитель может развертывать и выполнять произвольное программное обеспечение, включая операционные системы и приложения. Потребитель не может изменять облачную инфраструктуру или управлять ею, но имеет контроль над операционными системами, хранилищем, развернутыми приложениями. Также возможен ограниченный контроль отдельных сетевых компонентов (например, сетевого экрана хоста)
Initiator	Инициатор	Устройство, запускающее операцию запроса данных
Inode	Инод	Структура данных в файловой системе, содержащая информацию о файле или директории
I/O burst	Взрыв операций ввода-вывода	Большое количество записей, производимых за очень короткий период
Input/Output channel (I/O channel)	Канал ввода-вывода	Обеспечивает связь между шиной ввода-вывода данных и центральным процессором
I/O controller	Контроллер ввода-вывода	Компонент, обрабатывающий по одному запросу ввода-вывода за раз
Input Output per Second (IOPS)	Количество вводов-выводов в секунду	Количество операций чтения-записи в секунду
Integrated Device Electronics/Advanced Technology Attachment (IDE/ATA)	Интерфейс IDE/ATA	Стандартный интерфейс для подключения запоминающих устройств к персональным компьютерам
Integrity checking	Проверка целостности данных	Проверка соответствия контента файла цифровой подписи (хэшированный вывод данных, или КА)
Interface	Интерфейс	Устройство коммуникации двух элементов, например программного обеспечения с аппаратным устройством или пользователем

Internal transfer rate	Внутренняя скорость передачи	Скорость передачи данных с поверхности диска на читающие/пишущие головки
International Committee for Information Technology Standards (INCITS)	Международный комитет разработки стандартов информационных технологий	Форум разработчиков, производителей и пользователей информационных технологий, создающих и применяющих формализованные IT-стандарты. INCITS уполномочен и работает по правилам, утвержденным Американским национальным институтом стандартов (ANSI)
Internet Engineering Task Force (IETF)	Рабочая группа инженеров Интернет	Рабочая группа, устанавливающая стандарты для интернет-протоколов, например TCP/IP
Internet Protocol (IP)	Интернет-протокол	Сетевой протокол с коммутацией пакетов
Internet Protocol Security (IPSec)	Защита интернет-протокола	Комплекс алгоритмов, протоколов и процедур, используемых для защиты передаваемых данных посредством аутентификации и/или кодировки каждого пакета в потоке данных
Internet Protocol Storage Area Network (IP SAN)	Сеть хранения и передачи данных по IP-протоколу (IP SAN)	Гибридные сетевые решения для хранения данных, использующие сети IP
Internet Small Computer System Interface protocol (iSCSI)	Протокол iSCSI	Протокол на базе IP для интерфейса малых компьютерных систем. Передает блоки данных по стандартным IP-сетям
Internet Storage Name Service (iSNS)	Протокол службы имен хранилищ	Протокол автоматического обнаружения устройств хранения данных по сети IP
Inter-Switch Link (ISL)	Межкоммутаторная линия связи	Линия связи, объединяющая два маршрутизатора/оптоволоконных устройства с помощью E-портов
Intrusion Detection	Система обнаружения атак	Контроль посредством обнаружения проникновения в системы IT и попытки предотвращения атак путем прерывания соединения или активации блокирующей трафик функции межсетевого экрана
IP Storage	IP-хранилище	Хранилище, подключенное по сети TCP/IP
IT-as-a-Service	IT как услуга	Полная сквозная услуга по представлению и предоставлению инфраструктуры информационных технологий по требованию и как масштабируемый сервис

Jitter	Дрожание	Нежелательное колебание характеристик сигнала
Journal file system	Журналируемая файловая система	Файловая система, использующая отдельную область, называемую журналом, для отслеживания всех изменений файловой системы. С ее помощью можно легко восстановить данные при критических сбоях файловой системы
Jukebox	Дисковый автомат	Набор оптических дисков, объединенных в массив, используемый для хранения фиксированного контента
Jumbo frames	Jumbo-кадры	Большие кадры ethernet, используются в высокопроизводительных сетях для увеличения производительности на дальних расстояниях
Just a bunch of disks (JBOD)	Набор дисков	Набор дисков без скоординированного контроля управляющего программного обеспечения
k28.5	k28.5	Специальный десятибитный символ, используемый для обозначения начала команды управления в FC
Kerberos	Kerberos	Сетевой протокол аутентификации, позволяющий безопасно передавать данные через незащищенные сети и для безопасной идентификации
Key Distribution Center (KDC)	Центр распределения ключей	Сервер Kerberos, выполняющий задачи аутентификации и выдачи билетов доступа
LAN-based backup	Резервное копирование по локальной сети	Метод резервного копирования, при котором данные для резервного копирования передаются с сервера приложений на узел хранения через локальную сеть
Landing zone	Область парковки	Область жесткого диска рядом со шпинделем, где отдыхают головки чтения-записи
Latency	Задержка	Время, проходящее с момента запроса операции ввода-вывода до завершения запрошенной операции
Least Recently Used (LRU)	Замещение давно не использовавшихся адресов	Алгоритм использования кэш-памяти, при котором для освобождения или повторного использования выбираются адреса, к которым наиболее долго не было обращений
Level 1 (L1) cache	Кэш первого уровня	Дополнительная кэш-память центрального процессора. Содержит данные и программные инструкции, которые, скорее всего, потребуются процессору в ближайшее время

Lightweight Directory Access Protocol (LDAP)	Упрощенный протокол доступа к каталогам (LDAP)	Протокол доступа к каталогу через TCP/IP
Link aggregation	Объединение каналов	Это методика конфигурации сети для обеспечения высокой доступности. Она позволяет объединить несколько активных соединений Ethernet к одному коммутатору в один канал связи
Link Aggregation Control Protocol (LACP)	Контрольный протокол объединения каналов	Стандарт IEEE, объединяющий два или более физических канала передачи данных в один логический канал для увеличения доступности
Load balancing	Балансировка нагрузки	Метод равномерного распределения рабочей нагрузки по многочисленным компьютерным системам, сетевым каналам связи, процессорам, жестким дискам или другим ресурсам для обеспечения оптимального использования ресурсов
Local Area Network (LAN)	Локальная сеть	Инфраструктура связи на основе IP-протокола для подключения большого количества узлов к общему каналу в пределах небольшой области (как правило, в здании или на предприятии)
Local bus or I/O bus	Локальная шина или шина ввода-вывода	Высокоскоростная магистраль передачи данных, соединяющая процессор с периферийными устройствами
Local replication	Локальная репликация	Процесс создания копии рабочего тома на одном массиве хранения либо в пределах одного информационного центра
Log shipping	Доставка протокола записей	Метод репликации, при котором все события на ресурсе фиксируются файлом протокола и периодически передаются на удаленный узел, где воспроизводятся
Logical array	Логический массив	Подмножество дисков массива, которые могут быть объединены в логические группы, например в массив RAID
Logical Block Addressing (LBA)	Логическая (линейная) адресация блоков	Метод адресации блока на диске, использующий в качестве идентификатора подядковый номер блока (например, от 1 до 65 536) вместо номеров цилиндра, головки и сектора (C-H-S)
Logical Unit Number (LUN)	Номер логического устройства (логическое устройство)	Идентификатор логического запоминающего устройства, презентуемый хосту и используемый для доступа к данным на этом устройстве

Logical volume	Логический том	Виртуальный дисковый раздел, созданный в группе томов
Logical volume manager (LVM)	Диспетчер логических томов	Приложение, создающее логические тома и управляющее ими
Low watermark	Низкий уровень заполнения	Уровень заполнения кэш-памяти, при котором система хранения данных прекращает быстрый сброс данных на диск и переходит в фоновый режим сброса
LUN binding	Привязка логических устройств	Процесс создания логических устройств в RAID-массиве
LUN masking	Маскировка логических устройств	Процесс контроля доступа к данным, при котором узел видит только те логические устройства, к которым он имеет право доступа
Magnetic tape	Магнитная лента	Устройство хранения данных с последовательным доступом, используемое для хранения данных, резервного копирования и архивирования
Mail or import/export slot	Почтовая ячейка или ячейка импорта/экспорта	Ячейка для добавления магнитных лент в архив и их удаления из него без открытия главных дверей
Malware	Вредоносное программное обеспечение	Вредоносное программное обеспечение, разработано с целью снижения уровня конфиденциальности, целостности и доступности
Management Information Base (MIB)	База управляющей информации	Собрание объектов в (виртуальной) базе данных, используемой для управления сетевыми устройствами (например, роутерами и маршрутизаторами)
Maximum Transmission Unit (MTU)	Максимальный размер передаваемого блока данных	Параметр, определяющий максимальный размер блока данных, который может быть передан без фрагментации данных
MD5	MD5	128-битный алгоритм хеширования
Mean Time Between Failure (MTBF)	Среднее время между отказами	Расчет (в часах) среднего ожидаемого срока службы отдельного компонента
Mean Time To Repair (MTTR)	Среднее время восстановления	Необходимое в среднем время на восстановление неисправного компонента

Measured service	Сервис измерения показателей	Облачные системы автоматически контролируют и оптимизируют использование ресурсов за счет измерения показателей на определенном уровне абстрагирования в зависимости от типа сервиса (например, системы хранения данных, обработки, полосы пропускания и активных учетных записей пользователей). Использование ресурсов можно отслеживать, управлять ими и готовить отчеты. Это обеспечивает прозрачность для поставщика и потребителей используемых услуг
Media Access Control (MAC)	Контроль доступа к среде	Контролирующий физическую среду механизм в сети с разделяемой пропускной способностью
Memory virtualization	Виртуализация памяти	Технология выделения программе фиктивной непрерывной логической памяти, независимой от доступной физической памяти
Meta data	Метаданные	Информация о данных, описывающая такие характеристики данных, как содержание, качество и состояние
MetaLUN	Металогическое устройство	Расширенное логическое устройство, объединившее в себе многочисленные логические устройства
Metering	Сбор статистики	Мониторинг использования облачных ресурсов для предоставления и оценки стоимости
Metropolitan Area Network (MAN)	Городская сеть	Большая компьютерная сеть, как правило, в пределах города
Mirroring	Зеркалирование	Метод избыточности данных, когда все данные записываются одновременно на два дисковых устройства для защиты информации при аварии одного из дисков
Mixed topology	Топология смешанного типа	Топология системы резервирования, использующая топологии и локальных сетей, и сетей хранения данных
Mixed zoning	Комбинированное зонирование	Комбинация технологии Всемирной паутины и зонирования портов
Modification attack	Модификационная атака	Несанкционированная попытка злоумышленного изменения информации
Monitoring	Мониторинг	Процесс непрерывного сбора информации и анализ всей инфраструктуры хранения данных

Most Recently Used (MRU)	Замещение последних использовавшихся адресов	Алгоритм использования кэш-памяти, при котором для освобождения или повторного использования выбираются адреса, которые использовались в последнее время
Mounting	Монтирование	Процесс подключения файловой системы путем создания точки подключения. После монтирования файловая система может использоваться приложениями. Процесс установки картриджа с магнитной лентой в накопитель тоже называется монтированием
Multicast	Групповая передача	Одновременная передача пакетов данных на многочисленные адресные порты
Multi-Level Cell (MLC)	Многоуровневые ячейки	Элементарная ячейка во флеш-памяти, способная хранить несколько битов данных
Multimode Fiber (MMF)	Многомодовое оптоволокно	Оптоволоконный кабель, передающий многокомпонентные данные в виде световых лучей
Multipath I/O (MPIO)	Многопутевой ввод-вывод	Отказоустойчивый механизм хоста для прямых запросов ввода-вывода к устройству хранения данных с использованием более одного пути доступа
Multipathing	Передача по нескольким путям	Позволяет задействовать одновременно два или более информационных каналов для операций чтения-записи
Multiplexing	Мультиплексирование	Передача нескольких сигналов по одной линии или каналу связи
Multitenancy	Многопользовательский режим	Множество приложений, сосуществующих на одной платформе
Name server	Сервер имен	Узел, имплементирующий протокол преобразования логических имен
Namespace	Пространство имен	Виртуальный контейнер, наделяющий контекстом содержащиеся там данные (например, имена, технические термины, слова)
National Institute of Standards and Technology (NIST)	Национальный институт стандартов и технологий	Не подчиняющийся федеральным агентствам, в том числе администрации технологий Министерства торговли США, NIST стремится разрабатывать и продвигать измерения, стандарты и технологии для повышения производительности, облегчения торговли и улучшения качества жизни
Network	Сеть	Комплекс взаимосвязанных устройств для обмена ресурсами

Network-attached storage (NAS)	Сетевая система хранения данных	Специальное устройство для хранения файлов (с интегрированным или внешним запоминающим устройством), подключаемое к локальной сети
Network Data Management Protocol (NDMP)	Сетевой протокол управления данными	Открытый протокол, используемый для управления резервным копированием данных и связи между основным и вторичным устройствами хранения данных при восстановительных операциях в гетерогенном сетевом окружении
Network File System (NFS)	Сетевая файловая система	Стандартный метод общего доступа к файлам по сети в среде UNIX
Network Information System (NIS)	Сетевая информационная система	Система, позволяющая идентифицировать пользователя и предоставить доступ к ресурсу по сети
Network Interface Card (NIC)	Сетевая плата (карта)	Устройство для подключения компьютера к сети
Network latency	Сетевая задержка	Время, которое тратится на прохождение пакета через сеть от отправителя к получателю
Network layer firewalls	Межсетевой экран сетевого уровня	Межсетевой экран, реализованный на сетевом уровне модели OSI для проверки пакетов данных и выявления их соответствия настройкам правил безопасности
Network portal	Сетевой портал	Порт доступа устройства к любому узлу с технологией iSCSI
Network Time Protocol (NTP)	Протокол сетевого времени	Протокол синхронизации времени по сетям с коммутацией пакетов и нестабильной сетевой задержкой
Network topology	Сетевая топология	Схематическое описание устройства сети, включающее сетевые узлы и линии связи
Network virtualization	Сетевая виртуализация	Технология создания виртуальных сетей, не зависящих от физических сетей
Node	Узел	Подключаемое к сети устройство или элемент, например хост-узел или устройство хранения данных
Node loop port (NL-Port)	Узловой порт с поддержкой петли	Узловой порт с поддержкой топологии управляемой петли
Node port (N-port)	Узловой порт	Конечная точка в системе коммутации FC, как правило, порт адаптера главной шины или порт массива хранения данных, подсоединяемый к коммутатору

Non-protected restore	Незащищенное восстановление	Процесс восстановления, при котором целевой узел остается подключенным к источнику после завершения восстановительной операции и все операции записи на источник зеркально воспроизводятся на целевом узле
Nonrepudiation	Невозможность отказа от авторства	Гарантия невозможности последующего отрицания выполненного субъектом действия
Non-Volatile Random Access Memory (NVRAM)	Энергонезависимая память произвольного доступа	Память с произвольным доступом, не зависящая от потери данных из-за сбоя питания с использованием батареи или флеш-памяти
N_Port_ID virtualization (NPIV)	Виртуализация узловых портов	Конфигурация Fibre Channel, обеспечивающая несколько идентификаторов N_port, предоставленных одному физическому узловому порту (N_port)
Object-based storage device (OSD)	Объектное хранилище	Дисковое хранилище, хранит данные в контейнере, называемом объектом
Offline backup	Офлайн-режим (репликация базы данных)	Способ репликации, при котором база данных недоступна для выполнения операций ввода-вывода до ее окончания
On-demand self-service	Самообслуживание по требованию	Пользователь может в одностороннем порядке выделять вычислительные ресурсы, например серверное время и сетевое хранилище данных, при необходимости автоматически, без участия поставщика услуг
Online backup	Резервное копирование в онлайновом режиме	Вид резервного копирования, при котором у приложений есть доступ к копируемым данным
Online Transaction Processing (OLTP)	Оперативная обработка транзакций	Система, обрабатывающая транзакции в момент получения их компьютером и незамедлительно обновляющая данные
Open file agents	Агенты открытых файлов	Программы, взаимодействующие непосредственно с операционной системой и позволяющие осуществлять постоянное резервное копирование открытых файлов
Operating environment	Операционная среда	Термин, который используется для обозначения операционной системы массива хранения данных
Operational backup	Операционное резервное копирование	Сбор данных с возможностью восстановления когда-либо в будущем утерянных или поврежденных данных

Operational expenditure (OPEX)	Операционные затраты	Расходы, связанные с проведением обычных бизнес-операций
Optical Disc Drive (ODD)	Накопитель на оптических дисках	Накопитель, использующий лазерное излучение или электромагнитные волны светового спектра в процессе чтения и записи данных. Это компьютерное периферийное устройство, хранящее данные на оптических дисках
Orchestration	Оркестрация	Согласование отдельных ресурсов в системе для выполнения некоторого действия
Ordered set	Ordered set	Управляющие данные низкого уровня оптоволоконного канала (уровень FC-1), такие как разграничитель блоков данных или сигнализация, необходимая для передачи данных
Out-of-band	Внеполосная технология	Реализация, при которой конфигурации виртуализированной среды расположены вне информационного канала
Out-of-sync	Потеря синхронизации	Несогласованное состояние конечных данных, требующих полной синхронизации
Over commitment	Чрезмерное выделение	Выделение ресурсов (таких как память и ЦП) больше, чем физически доступно
P2V (physical to virtual)	От физических к виртуальным	Виртуализация физических серверов приложений на витуальные машины
Packet loss	Потеря пакетов	Ситуация, когда один или несколько пакетов данных, передаваемых по компьютерной сети, не доходят до получателя
Page	Страница	Единица выделения кэш-памяти
Para-virtualization	Паравиртуализация	Виртуализация сред, требующая изменения гостевых операционных систем в обмен на более высокую эффективность. Гостевая ОС адаптируется для запуска на гипервизоре
Parity	Контроль четности	Математический алгоритм, позволяющий обнаружить повреждение данных или восстановить их
Parity bit	Бит четности	Дополнительный бит, используемый для выявления ошибок при передаче данных. В современных средствах связи его используют для проверки точности всех передаваемых символов
Partition	Раздел	Часть физического или логического диска
Partitioning	Разбиение на разделы	Разделение дисков большой емкости на виртуальные тома меньшей емкости

Passive attack	Пассивная атака	Попытка несанкционированного доступа к информации без намерения ее изменения. Пассивные атаки могут угрожать конфиденциальности информации
Passive path	Пассивный путь	Путь, настроенный и готовый, но не используемый в данный момент. Обычно становится доступен при возникновении сбоя
Password	Пароль	Секретные данные, предоставляемые пользователем во время аутентификации для доступа к ресурсу
Payload	Полезная нагрузка	Часть потока данных, представляющая пользовательскую информацию и возможные служебные данные
Peripheral Component Interconnect (PCI)	Шина PCI	Стандартная шина для подключения устройств ввода-вывода к персональному компьютеру
Personally Identifiable Information (PII)	Личная информация	Любые данные о лице, которые могут определять данное лицо
Platform-as-a-Service	Платформа как сервис	Пользователь может использовать приложения поставщика, которые работают в облачной инфраструктуре. Приложения доступны с различных клиентских устройств через интерфейс тонкого клиента, например веб-браузера (допустим, электронная почта по веб-интерфейсу). Потребитель не может изменять или управлять базовой облачной инфраструктурой, включая сети, серверы, операционные системы, системы хранения данных или даже отдельные функции приложений, возможно, за исключением ограниченных пользовательских настроек приложения
Platter	Пластина	Круглая твердая пластина, покрытая магнитным материалом с обеих сторон. Используется в дисковом накопителе. Как правило, в одном накопителе несколько пластин
PLOGI (port login)	Регистрация портов	Выполняется между одним N-портом (инициатором) и другим N-портом (конечным портом устройства хранения данных) для запуска рабочего сеанса
Point-in-time (PIT) copy	Копия с указанием времени	Копия, содержащая все данные в таком виде, в каком они были в указанный момент времени
Port	Порт	Физическая точка подключения, к которой подсоединяется устройство

Port zoning	Зонирование портов	Доступ к данным определяется физическим портом, к которому подключен узел
Portal group	Группа порталов	Группа сетевых порталов, которые вместе могут поддержать сессию многоканального подключения
Prefetch (read ahead)	Предварительная выборка (предварительное считывание)	При последовательном чтении: считывание с диска блоков, следующих за уже запрошенными, и помещение их в кэш-память на всякий случай
Primitive sequence	Последовательность примитивов	Упорядоченный набор, передаваемый непрерывно до получения определенного ответа. Используется для инициализации управляемой петли
Private cloud	Частное облако	Виртуализированные ресурсы, доступные как услуга в рамках одной организации. Тем не менее они могут быть под управлением третьей стороны
Private key	Личный ключ	Криптографический ключ в асимметричной системе шифрования, не являющийся общедоступным
Process login (PRLI)	Процесс подключения	Подключение между портами N, используется для обмена служебными параметрами. Процесс проверки подключения зависит от протокола верхнего уровня
Production data	Производственные данные	Данные, генерируемые приложением, запущенным на сервере
Propagation	Распространение	Передача (распространение) сигналов через какую-либо среду с одного места на другое
Propagation delay	Задержка распространения	Время, необходимое для перемещения пакета данных от отправителя к получателю
Protocol	Протокол	Набор правил или стандартов, обеспечивающих коммуникацию систем или устройств
Protocol data unit (PDU)	Единица обмена протокола	Сообщение, передаваемое по сети между двумя узлами с целью коммуникации
Public cloud	Публичное облако	Услуги провайдера, предоставляемые общественности на основе договорных соглашений
Public key	Открытый ключ	Криптографический ключ, переданный в целях использования в асимметричном шифровании объекту, который имеет закрытый ключ

Public Key Infrastructure (PKI)	Инфраструктура открытых ключей	Программное обеспечение, аппаратные устройства, люди и процедуры, облегчающие безопасное создание цифровых сертификатов и управление ими
Quality of Service (QoS)	Качество услуг	Установленный уровень производительности в системе передачи данных
Queue	Очередь	Область ожидания запроса на ввод-вывод перед его обработкой контроллером ввода-вывода
Quiescent state	Состояние покоя	Состояние приложения или устройства, при котором данные не меняются. Обработка приостановлена, задачи либо завершены, либо не запущены
Quota	Квота	Ограничение объема ресурсов, доступного пользователю (например, квота на почтовые ящики, файловые системы)
RAID controller	Контроллер RAID	Специальное аппаратное устройство, выполняющее все вычисления, связанные с обслуживанием RAID-массива, и презентующее хосту дисковые тома
Random access memory (RAM)	Оперативная память	Энергозависимая память, разрешающая прямой доступ в любую ячейку памяти
Random I/O	Случайный ввод-вывод	Последовательные запросы на ввод-вывод, не получающие доступ к смежным областям в системе хранения данных
Rapid elasticity	Оперативность и гибкость	Возможность быстро и гибко выделять ресурсы, в некоторых случаях автоматически, быстрого горизонтального масштабирования и быстрого возвращения. Пользователю ресурсы, доступные для выделения, часто кажутся неограниченными с возможностью приобрести любое количество и в любое время
Raw capacity	Неформатированная емкость	Общий объем адресуемой емкости устройства хранения данных в системе хранения данных
Raw partition	Необработанный раздел	Раздел диска, не управляемый системой управления томами
Read-only memory (ROM)	Постоянное запоминающее устройство (ПЗУ)	Энергонезависимый вид памяти, доступный только для чтения

Read/write heads	Головки чтения-записи	Компоненты жесткого диска, читающие и записывающие данные на жесткий диск. У большинства дисков по две пишущие/читающие головки на каждой пластине, по одной для каждой из сторон
Recoverability	Восстанавливаемость	Возможность восстановления с данной конкретной копии для продолжения производственного процесса к директивному сроку восстановления
Recovery Point Objective (RPO)	Директивная точка восстановления	Момент времени, к которому система должна быть восстановлена после аварии. Определяет объем утраченных данных, который не будет критичным для бизнеса
Recovery Time Objective (RTO)	Директивное время восстановления	Время необходимого восстановления систем, приложений и функций после аварии. Определяет срок простоя, который не будет критичным для бизнеса
Redundancy	Избыточность	Включение в систему дополнительных компонентов (например, накопителя, адаптера главной шины, канала или данных), обеспечивающих непрерывность операции в случае выхода из строя одного из рабочих компонентов
Redundant Array of Independent Disks (RAID)	Избыточный дисковый массив, RAID-массив	Массив из нескольких дисков, управляемых контроллером, взаимосвязанных скоростными каналами и воспринимаемых внешней системой как единое целое. Служит для повышения надежности хранения данных и/или повышения скорости чтения-записи информации
Redundant Array of Inexpensive Nodes (RAIN)	Избыточный массив недорогих узлов	Реплицирование данных на несколько независимых узлов для обеспечения резервирования в CAS
Registered State Change Notification (RSCN)	Уведомление о зарегистрированном изменении состояния	Сообщение об изменениях состояния одного узла для других узлов в системе коммутации Fibre Channel
Reliability	Надежность	Гарантия работы системы в обычном режиме делопроизводства на определенный срок в данных условиях
Remote Authentication Dial-in User Service (RADIUS)	RADIUS	Протокол аутентификации, авторизации и учета для контроля доступа к сетевым ресурсам

Remote backup	Удаленное резервное копирование	Метод резервного копирования, при котором копия с основного устройства хранения данных выполняется на устройства резервного копирования, расположенные в другом месте
Remote Procedure Call (RPC)	Вызов удаленных процедур	Технология, позволяющая компьютерной программе выполнять подпрограмму или процедуру на другом компьютере
Remote replication	Удаленная репликация	Процесс копирования данных локального массива хранения данных на массив, расположенный в другом месте
Replica	Реплика, копия	Образ/копия данных, пригодная для использования другим приложением
Representational State Transfer	Передача репрезентативного состояния	Подход для получения контента с веб-сайта путем чтения назначенной веб-страницы, которая содержит файл XML, описывающий и включающий в себя необходимый контент
Repudiation attack	Атака для сокрытия авторства	Атака, делающая невозможным или затрудняющая определение авторства чего-либо
Resource pooling	Объединение ресурсов в пулы	Вычислительные ресурсы поставщика объединяются в пулы для обслуживания нескольких пользователей на основе многопользовательской модели с различными физическими и виртуальными ресурсами, динамически назначаемыми и переназначаемыми согласно запросам пользователей. Обеспечивается определенная независимость от местонахождения, так как заказчик обычно не может контролировать или знать точное местоположение предоставленных ресурсов, но может указать местоположение на более высоком уровне абстрагирования (например, страна, область или центр обработки данных). Примеры ресурсов: хранилище данных, вычислительные, память, полоса пропускания сети и виртуальные машины
Response time	Время ответа	Период времени, который необходим системе или устройству, чтобы среагировать на ввод
Restartability	Готовность к перезапуску	Определяет достоверность и готовность к использованию копии данных для перезапуска операций бизнес-процессов в аварийной ситуации
Restore	Восстановление	Возвращение данных в их изначальное, готовое к использованию состояние

Resynchronization	Ресинхронизация	Процесс восстановления только тех блоков данных, которые обновились после создания PIT-копии (копии с указанием времени)
Retention period	Период сохранности	Необходимый срок хранения резервных копий данных
Return on Investment (ROI)	Возврат инвестиций	Расчет финансовой выгоды от инвестирования средства в разработку/внесение изменений в систему
Rewind time	Время перемотки	Время, необходимое накопителю на магнитной ленте для перемотки к началу
Risk analysis	Анализ рисков	Анализ, выполняемый как часть BC-процессов и проверяющий интенсивность отказов компонентов и среднее время восстановления, исчисляемое в единицах среднего времени восстановления и среднего времени безотказной работы
Robotic arm	Манипулятор	Компонент библиотеки магнитных лент, используемый для перемещения лент из слотов к устройству чтения-записи и обратно
Role-based access control (RBAC)	Функциональный (ролевой) контроль доступа	Метод ограничительного доступа к системе для авторизованных пользователей на основе выполняемых ими функций
Roll back	Откат	Откат вторичной реплики к состоянию предыдущей PIT-копии (копии с указанием времени)
Rolling disaster	Повторяющиеся аварии	Аварии, начинающиеся и заканчивающиеся в разное время, между которыми может быть несколько миллисекунд или минут
Rotation speed	Скорость вращения	Скорость вращения магнитных пластин дисковода
Rotational latency	Задержка из-за вращения	Время, необходимое пластине жесткого диска для поворота и размещения головки чтения-записи в искомый сектор данных
Round-robin	Алгоритм кругового обслуживания	Алгоритм, при котором запросы направляются циклически по очереди по всем доступным путям
Round-trip delay (RTD)	Задержка подтверждения приема	Задержка между отправкой данных и получением подтверждения приема с удаленного узла
Router	Маршрутизатор, роутер	Межсетевое устройство, позволяющее проводить маршрутизацию информации в различных сетях

SAN-based backup	Сетевое резервное копирование	Метод резервного копирования данных по сети хранения данных
Save location	Место сохранения	Набор частных LUN, сохраняющих данные на момент времени перед их обновлением на источнике или целевом узле
Scale out	Горизонтальное масштабирование	Масштабирование или добавление ресурсов горизонтальным образом в соответствии с широким или лавинным спросом. В то же время вертикальное масштабирование — это увеличение производительности для соответствия требованиям
SCSI Application Layer (SAL)	Уровень приложения SCSI	Самый верхний уровень коммуникационной модели SCSI, содержащий клиентские и серверные приложения, запускающие и обрабатывающие операции ввода-вывода с помощью протокола приложения SCSI
SCSI Transport Protocol Layer (STPL)	Уровень транспортного протокола SCSI	Содержит службы и протоколы, обеспечивающие коммуникацию между инициатором и целевыми узлами
Sector	Сектор	Минимальная адресуемая логическая единица в дисковом накопителе, где физически хранятся данные
Secure Shell (SSH)	SSH-протокол	Сетевой протокол для обмена данными по защищенному каналу между двумя компьютерами
Secure Sockets Layer (SSL)	SSL-протокол	Протокол шифрования, обеспечивающий безопасное соединение через Интернет между клиентом и сервером с помощью криптографии с открытым ключом
Securities and Exchange Commission (SEC)	Комиссия по ценным бумагам	Правительственное учреждение США, отвечающее за выполнение законов по государственным ценным бумагам и регулирующее индустрию ценных бумаг и рынок акций
Security information management	Управление информационной безопасностью	Наборы данных, таких как журнал событий в центральном репозитории, используется для эффективного анализа
Seek time	Время поиска	Время, необходимое для перемещения головок чтения-записи между треками в дисковом накопителе
Seek time optimization	Оптимизация времени поиска	Алгоритм оптимизации движений головок чтения-записи, в результате применения которого время отклика может быть уменьшено

Selective Acknowledge (SACK)	Избирательное подтверждение	При избирательном подтверждении получатель данных может проинформировать отправителя обо всех успешно доставленных сегментах, давая ему возможность отправить повторно только утерянные данные
SendTargetDiscovery	Обнаружение целевых узлов при запросе	Команда, запускаемая инициатором для начала процесса обнаружения. Целевой узел выдает имена и адреса доступных хосту целевых узлов
SEQ_ID	Последовательный идентификатор	Идентификатор кадра как компонента последовательности
Sequence	Последовательность	Непрерывный набор кадров, пересылаемых от одного порта к другому
Serial Advanced Technology Attachment (SATA)	Последовательный интерфейс обмена данными с накопителями информации (SATA-интерфейс)	Версия интерфейса IDE/ATA, разработанного для последовательной передачи данных
Serial attached SCSI (SAS)	Последовательный SCSI	Последовательный протокол для подключений типа «точка — точка», который представляет альтернативу параллельному протоколу SCSI
Server-based virtualization	Виртуализация сервера	Это методика маскирования или абстрагирования физического аппаратного обеспечения от операционной системы. Она дает возможность параллельно работать на отдельных или кластеризованных физических машинах нескольким операционным системам
Server/host/compute virtualization	Виртуализация сервера/хоста/вычислений	Механизм, который позволяет различным операционным системам и приложениям работать одновременно на разных виртуальных машинах, созданных на одном физическом сервере или группе серверов
Serverless backup	Резервное копирование без использования сервера	Методология резервного копирования, которое использует устройство, отличное от сервера, для копирования данных в устройство резервного копирования
Server Message Block (SMB)	Блок серверных сообщений	Это протокол доступа сетевой файловой системы, предназначенный для доступа Windows-клиентов к файлам и другим ресурсам удаленного сервера Windows

Service catalog	Каталог услуг	Каталог со списком услуг, компонентов, атрибутов услуг и соответствующих цен
Service-Oriented Architecture (SOA)	Сервисно-ориентированная архитектура	Архитектура, специально созданная для поддержки конкретных услуг с ожидаемыми результатами
Service-level agreement (SLA)	Соглашение об уровне услуг	Соглашение между поставщиком и потребителем услуг
Service Location Protocol (SLP or srvloc)	Протокол обнаружения сервисов	Протокол, позволяющий компьютерам и другим устройствам обнаруживать сервисы, доступные в локальной сети без предварительной конфигурации
Service Set Identifier (SSID)	Идентификатор набора служб	Уникальный идентификатор из 32 символов, присваиваемый заголовку пакетов данных, передаваемых по беспроводной локальной сети. Этот идентификатор действует как пароль при попытке подключения мобильного устройства к беспроводной сети
Shared secret	Общий ключ	Предварительный ключ, известный только сторонам, участвующим в безопасном обмене данными
Simple Mail Transfer Protocol (SMTP)	Простой протокол передачи почты, SMTP-протокол	Стандартный интернет-протокол для пересылки электронной почты
Simple Network Management Protocol (SNMP)	Простой протокол управления сетью, SNMP-протокол	Протокол управления сетью, используемый для мониторинга состояния и производительности сетевых устройств
Simple Object Access Protocol (SOAP)	Простой протокол доступа к объектам	Протокол формирования пакета, который упаковывает сообщения XML для обмена данными между клиентом и веб-службами
Single-instance Storage (SiS)	Единичное хранение	Механизм, который позволяет системе избегать хранения многочисленных копий пользовательских данных при помощи идентификации каждого объекта посредством уникального идентификатора объекта
Single Large Expensive Drive (SLED)	Одиночный большой и дорогой дисковый накопитель	Высокопроизводительный и, как правило, дорогой подключаемый к компьютеру дисковый накопитель. Противопоставляется RAID-массиву

Single-Level Cell (SLC)	Одноуровневая ячейка	Используемая в твердотельных накопителях технология памяти, хранящая по одному биту в каждой ячейке памяти, в результате чего происходит более быстрая передача данных, снижается потребление энергии и увеличивается срок службы ячейки
Single-mode fiber (SMF)	Одномодовое оптоволокно	Тип оптоволокна, передающий данные в форме единичного светового луча, идущего по центру волокна
Single point of failure	Единая точка отказа	Сбой компонента, который может стать причиной недоступности всей системы или IT-услуг
Small Computer System Interface (SCSI)	Интерфейс малых компьютерных систем, SCSI-интерфейс	Популярный интерфейс, используемый для подключения периферийного устройства к компьютеру и передачи данных между ними
Snapshot	Мгновенная копия	PIT-копия (копия с указанием времени)
Sniffer	Перехватчик	Программный инструмент для идентификации пакетов сетевого трафика
Snooping	Перехват	Несанкционированный доступ к данным другого пользователя или организации
Software-as-a-Service	Программное обеспечение как услуга	Предоставляемая потребителю возможность использовать приложения провайдера, выполняемые на облачной инфраструктуре. Приложения доступны с различных клиентских устройств через интерфейс тонкого клиента, например веб-браузер (например, электронная почта через веб-интерфейс). Потребитель не может изменять базовую облачную инфраструктуру, включая сети, серверы, операционные системы, системы хранения данных или даже отдельные функции приложений, возможно, за исключением ограниченных пользовательских настроек приложения, или управлять ими
Solid-state drive (SSD)	Твердотельный накопитель	Энергонезависимое перезаписываемое компьютерное запоминающее устройство без движущихся механических частей
Source ID (S_ID)	Идентификатор источника	Стандартный Fibre Channel-адрес для порта источника
Spindle	Шпиндель	Связанная с двигателем деталь дискового накопителя, на которой крепятся пластины

Spoofing	Имитация соединения	Процесс успешной маскировки пользователем или программой под другого пользователя или программу путем фальсификации данных
Standby Power Supply (SPS)	Резервный источник электропитания	Источник питания, поддерживающий электропитание кэш-памяти в течение периода, достаточного для сброса кэш-памяти на диск
State Change Notification (SCN)	Уведомление об изменении состояния	Уведомление, направляемое на iSNS-сервер при подключении устройств к домену обнаружения или их отключении
Storage area network (SAN)	Сеть хранения данных	Высокоскоростная специальная сеть для устройств и серверов хранения данных
Storage array–based remote replication	Удаленная репликация массивов хранения данных	Репликация, запускаемая и завершаемая в массиве хранения данных
Storage controller	Контроллер хранилища	Устройство, которое обрабатывает запросы к хранилищу и направляет их на устройства хранения данных
Storage Management Initiative (SMI)	Инициатива управления системами хранения данных	Стандарт хранения, используемый для осуществления взаимодействия между системами хранения данных разных производителей
Storage network	Сеть хранения	Сети, основное назначение которых — передача данных между вычислительными системами и хранилищами, а также между хранилищами
Storage Networking Industry Association (SNIA)	Ассоциация SNIA	Ведущая некоммерческая организация, занимающаяся разработкой и продвижением стандартов, технологий и образовательных услуг в сфере управления информацией и хранения данных
Storage Node (Backup/Recovery)	Узел хранения (резервного копирования/восстановления)	Часть системы резервного копирования, контролирующая одно или несколько устройств резервного копирования (накопитель на магнитной ленте, библиотеку лент или дисковое устройство) и получающая резервные данные от клиентов резервного копирования
Storage Resource Management (SRM)	Управление ресурсами хранения	Управление ресурсами хранения (физическими и логическими), которые включают элементы системы хранения, устройства хранения данных, виртуальные устройства, дисковые тома и файловые ресурсы

Storage virtualization	Виртуализация хранилища	Абстрагирование внутренней функции системы хранения данных от приложений, вычислительных серверов или основной сети в целях предоставления возможности независимого сетевого управления хранилищем или данными на уровне приложений
Store	Запоминающее устройство	Получает данные от агентов, обрабатывает данные и обновляет репозиторий
Strip	Стрип	Группа последовательно адресованных блоков на каждом диске в RAID-массиве
Stripe	Страйп	Группа связных стрипов, охватывающая все диски дискового массива RAID
Stripe width	Ширина страйпа	Равняется количеству жестких дисков в RAID-массиве
Striping	Распределение	Разбивка и распределение данных по многочисленным жестким дискам
Structured data	Структурированные данные	Данные, которые могут быть организованы в ряды и колонки и обычно хранятся в базе данных или электронной таблице
Stub file	Файл-заглушка	Небольшой файл обычно объемом 8 Кбайт, содержащий метаданные оригинального файла
Superblock	Суперблок	Элемент файловой системы, содержит важную информацию о файловой системе: тип системы, даты создания и модификации, размер и размещение файловой системы, количество доступных ресурсов и флажок, указывающий на статус монтирования файловой системы
Swap file	Файл подкачки, страничный файл, своп	Это часть физического диска, замаскированная под физическую память операционной системы
Switched fabric	Коммутирующая матрица; коммутируемая связная архитектура; система коммутации	Топология Fibre Channel, где каждое устройство обладает уникальным выделенным путем ввода-вывода к устройству, с которым оно коммуницирует
Switch	Коммутатор	Более интеллектуальное устройство по сравнению с концентратором (хабом). Коммутаторы обеспечивают прямую трассировку данных от одного физического порта к другому

Switching	Коммутирование	Процесс соединения сегментов сети посредством аппаратного устройства, называемого коммутатором
Symmetrix Enginuity	Symmetrix Enginuity	Операционная среда для систем EMC Symmetrix
Symmetrix Remote Data Facility (SRDF)	SRDF	Программное обеспечение для удаленной репликации между дисковыми массивами EMC Symmetrix
Synchronous Digital Hierarchy (SDH)	Синхронная цифровая иерархия	Технология транспортных телекоммуникационных сетей. Описана в стандартах ITU G.707 и G.708
Synchronous Optical Networking (SONET)	Сеть синхронной оптической связи	Стандарт оптической телекоммуникационной передачи, при котором трафик от многочисленных подписчиков мультиплексируется и отправляется по кольцевой сети в виде оптического сигнала
System bus	Системная шина	Шина, осуществляющая передачу данных между процессором и памятью
Tag RAM	Теговая оперативная память	Интегрированная часть кэш-памяти, отслеживающая расположение данных в хранилище данных: место данных в памяти и место расположения данных на диске
Tampering	Фальсификация	Несанкционированная модификация, которая изменяет нормальное функционирование устройства, системы или пути обмена данными таким способом, который существенно снижает безопасность или функциональность
Tape cartridges	Ленточный картридж	Устройство, содержащее магнитные ленты для хранения данных
Tape drive	Накопитель на магнитной ленте	Устройство хранения данных, читающее и пишущее данные, хранящиеся на магнитной ленте
Target	Целевое устройство (в SCSI)	SCSI-устройство, выполняющее команды SCSI-инициатора
Target ID	Идентификатор целевого устройства	Уникальным образом идентифицирует целевое устройство и используется как адрес для обмена командами и статусной информацией с инициаторами
TCP Offload Engine (TOE)	Обработка блоков TCP/IP вне центрального процессора	Технологии для повышения производительности TCP/IP посредством передачи обработки TCP/IP сетевой карте

TCP/IP Offload Engine (TOE) card	Карта TOE	Снимает с хоста функции управления TCP-протоколом
Thin provisioning	Тонкое выделение ресурсов	Предоставление логического устройства нужной емкости и маскирование полной емкости
Threats	Угрозы	Атаки, которые могут быть выполнены в инфраструктуре информационной сети
Throughput	Пропускная способность	Количество данных, которые могут быть успешно переданы за единицу времени
Tiered storage	Многоуровневое хранение данных	Окружение, распределяющее системы хранения данных на два или более уровня, исходя из разницы в цене, производительности, объеме и функциональности
Total Cost of Ownership (TCO)	Общая стоимость владения	Смета по прямым и косвенным издержкам на закупку и эксплуатацию программного обеспечения и аппаратных устройств
Tracks	Треки	Логические концентрические кольца на магнитных пластинах дискового накопителя
Transmission code	Код передачи	Кодирование, используемое в FC-канале для оптимизации скорости передачи информации
Transmission Control Protocol (TCP)	Протокол управления передачей	Протокол установления соединения, организующий виртуальную связь перед отправкой информации с источника на конечный узел
Transmission word	Слово передачи	Единица передачи данных по оптоволокну, содержащая строку из четырех последовательных символов передачи, или байтов
Triangle/Multitarget	Треугольная/многоцелевая репликация	Трехузловой процесс удаленной репликации, когда данные вначале копируются с источника на промежуточный массив хранения данных (бункер), а потом на удаленный массив хранения данных
Trusted Computing Base (TCB)	Доверенная вычислительная база	Набор всех компонентов в вычислительной среде, которая обеспечивает безопасную среду
Tunneling protocol	Туннельный протокол	Протокол, содержащий полезную информацию, скрытый от протокола передачи данных с целью защиты данных
Universal Serial Bus (USB)	Универсальная последовательная шина	Широко используемый интерфейс последовательной шины для подключения периферийных устройств

Unstructured data	Неструктурированные данные	Данные, не обладающие четкой структурой, обычно хранящиеся в разных типах файлов
Upper-layer protocol (ULP)	Протокол верхнего уровня	Термин, обозначающий более абстрактный протокол при выполнении инкапсуляции
User Datagram Protocol (UDP)	Протокол пользовательских датаграмм	Протокол транспортного уровня без установления соединения, используемый в IP-сетях
User identifier (UID)	Идентификатор пользователя	Уникальный номер, идентифицирующий пользователя в UNIX-системе
Virtual Concatenation (VCAT)	Виртуальная конкатенация	Технология инверсного мультиплексирования, основанная на разделении одного высокоскоростного потока данных на несколько низкоскоростных с целью последующей независимой передачи по нескольким узкополосным линиям связи
Virtual Data Center (VDC)	Виртуализированный центр обработки данных	Виртуализированное представление физической инфраструктуры и услуг, которые она может предоставить
Virtual Desktop Infrastructure (VDI)	Инфраструктура виртуальных рабочих мест	Технология виртуализации рабочих мест, которая позволяет операционным системам рабочих мест выполняться на виртуальной машине (виртуальное рабочее место), находящейся на сервере в центре обработки данных. Пользователи могут удаленно получить доступ к этим рабочим местам с различных клиентских устройств, таких как ноутбуки, настольные компьютеры и мобильные устройства
Virtual E_Port (VE_Port)	Виртуальный E_Port	Виртуальный порт расширения в коммутаторе FCoE для межкоммутаторных линий
Virtual F_Port (VF_Port)	Виртуальный F_Port	Виртуальный порт коммутатора FCoE в системе коммутации
Virtual Fabric (VF)	Виртуальная система коммутации	Система коммутации, определяемая VF_ID, состоящая из разделов коммутаторов и портов n_port, имеющая единую систему управления и независимое адресное пространство
Virtual LAN (VLAN)	Виртуальная локальная сеть	Коммутируемая сеть, логически сегментированная по функциям, проектным группам или приложениям независимо от физического расположения сетевых пользователей

Virtual machine (VM)	Виртуальная машина (ВМ)	Программный образа компьютера, который функционирует как физическая машина. Он также отображается в сети, как и отдельная физическая машина. Несколько виртуальных машин могут работать на одной физической машине
Virtual pools	Виртуальный пул	Логическая группа или кластер ресурсов
Virtual private network (VPN)	Виртуальная частная сеть	Защищенная сеть связи, функционирующая по принципу туннелирования внутри другой сети
Virtual storage area network (VSAN)	Виртуальная сеть хранения данных	Набор портов коммутаторов в сети Fibre Channel, образующих виртуальную систему коммутации
Virtual tape library (VTL)	Виртуальная библиотека лент	Дисковый накопитель, логически представленный в приложении посредством программы-эмулятора как набор накопителей на магнитной ленте
Virtualization	Виртуализация	Технология маскировки физических ресурсов путем их логического представления
Virus	Вирус	Вредоносная компьютерная программа, которая может инфицировать компьютер без разрешения или ведома пользователя
VLAN tagging	Маркирование VLAN	Процесс, который вставляет маркер (тег) в пакет Ethernet. Тег содержит идентификатор виртуальной локальной сети
Volume group (VG)	Группа томов	Группа физических томов (диска), на которой можно создать логический том (раздел)
Vulnerability	Уязвимости	Ошибки механизма защиты данных, которые можно использовать как угрозы
Warning alert	Предупреждение	Состояние, требующее административного вмешательства для предотвращения нарушения доступа
Wavelength-Division Multiplexing (WDM)	Мультиплексирование по длине волны	Технология уплотнения нескольких оптических сигналов передачи по одному оптоволокну путем использования лазеров с различной длиной волн для передачи различных сигналов
Web-Based Enterprise Management (WBEM)	Управление предприятием на основе веб-технологий	Набор конфигураций управления и интернет-стандартов, разработанных ассоциацией DMTF и способствующих развитию веб-технологий
Web console	Веб-консоль	Веб-интерфейс, позволяющий проводить мониторинг сети хранения данных как по локальной сети, так и по сети Интернет

Wide area network (WAN)	Глобальная сеть	Объединенная сеть компьютеров по всему миру (выходящая за пределы городов и даже государств). Также используется для объединения многочисленных локальных сетей
World Wide Name (WWN)	Глобальное имя	Выдаваемый поставщиком 64-битный уникальный идентификационный номер, назначаемый узлам и портам оптоволоконной сети
World Wide Node Name (WWNN)	Глобальное имя узла	64-битный идентификатор узла, используемый при подключении к системе коммутации Fibre Channel
World Wide Port Name (WWPN)	Глобальное имя порта	64-битный идентификатор порта, используемый при подключении к системе коммутации Fibre Channel
Write aside size	Запись на резервный источник при больших размерах	Если запрос ввода-вывода превышает определенный размер, запись производится непосредственно на диск вместо записи в кэш-память. Это помогает избежать заполнения больших объемов кэш-памяти при крупных записях
Write-back cache	Кэш с отложенной записью	Данные размещаются в кэш, и сразу же на хост-узел отправляется подтверждение. Потом данные из кэш-памяти передаются (переносятся) на диск
Write cache	Кэш под запись	Часть кэш-памяти, выделяемая для временного хранения данных перед записью их на диск для постоянного хранения
Write Once Read Many (WORM)	Однократная запись — многократное считывание	Класс устройств хранения данных (например, оптических дисков), запись на которые возможна только один раз, а считывание — многократно
Write penalty	Издержки записи	Увеличение числа операций ввода-вывода в RAID-массивах, где каждая логическая операция записи реализуется через несколько физических операций записи на диски
Write splitting	Разветвление операций записи	Процесс перехвата операций записи и перенаправление их для одновременного обновления как источника, так и журнала
Write-through cache	Кэш со сквозной записью	Данные размещаются в кэш, оттуда сбрасываются на диск, а затем подтверждение о записи направляется хосту
ZIP	ZIP	Популярный вид сжатия данных и формат архивов; разработан Филом Кацем

Zone bit recording (ZBR)	Технология зональной записи	Метод записи данных, использующий преимущество структуры диска с размещением на внешних треках большего количества секторов, чем на внутренних
Zone set	Группа зон	Совокупность зон системы коммутации Fibre Channel, которую можно активировать и деактивировать как единое целое. Группу зон называют также конфигурацией зон
Zoning	Зонирование	Технология, позволяющая объединять узлы внутри системы коммутации Fibre Channel в группы. Узлы зоны могут взаимодействовать только друг с другом

От хранения данных к управлению информацией

2-е издание

Перевел с английского Н. Вильчинский

Координатор проекта	*Г. Смородин*
Заведующая редакцией	*Ю. Сергиенко*
Ведущий редактор	*Н. Римицан*
Научный редактор	*В. Дашонок*
Литературный редактор	*Н. Рощина*
Художник	*В. Шимкевич*
Корректоры	*М. Молчанова, В. Сайко*
Верстка	*Л. Соловьева*

ООО «Питер Пресс», 192102, Санкт-Петербург, ул. Андреевская (д. Волкова), 3, литер А, пом. 7Н.

Налоговая льгота — общероссийский классификатор продукции ОК 034-2014, 58.11.12.000 — Книги печатные профессиональные, технические и научные.

Подписано в печать 03.03.16. Формат 70x100/16. Бумага офсетная. Усл. п. л. 43,860. Тираж 1500. Заказ 0000.

Отпечатано в ОАО «Первая Образцовая типография». Филиал «Чеховский Печатный Двор».
142300, Московская область, г. Чехов, ул. Полиграфистов, 1.
Сайт: www.chpk.ru. E-mail: marketing@chpk.ru
Факс: 8(496) 726-54-10, телефон: (495) 988-63-87