

УЧЕБНИК ДЛЯ ВУЗОВ

В.Г. Олифер, Н.А. Олифер

**КОМПЬЮТЕРНЫЕ СЕТИ,
принципы, технологии, протоколы**

2-Е ИЗДАНИЕ

Библиотека БИУНР

Краткое содержание

<u>Часть I. Основы сетей передачи данных</u>	
Глава 1. Роль компьютерных сетей в мире телекоммуникаций.....	28
Глава 2. Общие принципы построения сетей.....	49
Глава 3. Открытые системы и модель OSI.....	132
Вопросы и упражнения к части I.....	161
<u>Часть II. Технологии физического уровня</u>	
Глава 4. Линии связи.....	168
Глава 5. Передача данных на физическом и канальном уровнях	193
Глава 6. Первичные сети.....	228
Вопросы и упражнения к части II.....	277
<u>Часть III. Локальные сети</u>	
Глава 7. Базовые технологии локальных сетей.....	280
Глава 8. Развитие технологии Ethernet.....	342
Глава 9. Структуризация LAN на физическом и канальном уровнях.....	366
Глава 10. Дополнительные функции мостов и коммутаторов.....	440
Вопросы и упражнения к части III.....	470
<u>Часть IV. Составные сети</u>	
Глава 11. Объединение сетей на основе сетевого уровня.....	476
Глава 12. Адресация в IP-сетях.....	495
Глава 13. Протоколы межсетевого и транспортного уровней TCP/IP.....	518
Глава 14. Протоколы маршрутизации и маршрутизаторы.....	561
Вопросы и упражнения к части IV	615
<u>Часть V. Глобальные сети</u>	
Глава 15. Качество обслуживания в пакетных сетях.....	620
Глава 16. Глобальные сети с коммутацией каналов.....	656
Глава 17. Глобальные сети с коммутацией пакетов.....	676
Глава 18. Организация доступа.....	758
Вопросы и упражнения к части V	772
<u>Часть VI. Средства анализа и управления сетями</u>	
Глава 19. Функции и архитектура систем управления сетями.....	776
Глава 20. Стандарты систем управления.....	791
Глава 21. Мониторинг и анализ локальных сетей.....	816
Вопросы и упражнения к части VI	829
Заключение. Заглядывая в будущее.....	830
Приложение. Ответы на вопросы и упражнения.....	832
Рекомендуемая литература.....	840
Алфавитный указатель.....	842

Содержание

Благодарности.....	20
От авторов.....	21
Для кого эта книга.....	21
Изменения во втором издании.....	22
Структура книги.....	23
От издательства.....	26
Часть I. Основы сетей передачи данных	
Глава 1. Роль компьютерных сетей в мире телекоммуникаций.....	28
Эволюция вычислительных систем.....	28
Системы пакетной обработки.....	29
Многотерминальные системы — прообраз сети.....	29
Глобальные сети.....	31
Локальные сети.....	32
Выводы.....	35
Конвергенция сетей.....	36
Сближение локальных и глобальных сетей.....	36
Конвергенция компьютерных и телекоммуникационных сетей.....	38
Выводы.....	41
Компьютерные сети — частный случай распределенных вычислительных систем.....	42
Мультипроцессорные компьютеры.....	42
Кластеры.....	43
Вычислительные сети.....	45
Распределенные программы.....	46
Выводы.....	47
Глава 2. Общие принципы построения сетей.....	49
Связь «точка-точка»	49
Связь компьютера с периферийными устройствами.....	49
Простейший случай взаимодействия двух компьютеров.....	52
Задача физической передачи данных по линиям связи.....	55
Выводы.....	57
Проблемы связи нескольких компьютеров.....	57
Топология физических связей.....	57
Адресация узлов сети.....	61
Коммутация и мультиплексирование.....	65
Коммутация каналов и пакетов.....	74
Дейтаграммная передача и виртуальные каналы в сетях с коммутацией пакетов.....	86
Выводы.....	90
Структуризация как средство построения больших сетей.....	92
Причины структуризации локальных и глобальных сетей.....	92
Физическая и логическая структуризация.....	92

Выводы.....	99
Основные программные и аппаратные компоненты сети.....	100
Многослойная модель сети.....	100
Сетевые службы и операционная система.....	101
Общая структура телекоммуникационной сети.....	102
Выводы.....	105
Сети операторов связи.....	105
Операторы связи и клиенты.....	106
Услуги, поставщики услуг и сетевая инфраструктура.....	107
Клиенты.....	108
Инфраструктура.....	109
Территория покрытия.....	110
Выводы.....	111
Корпоративные сети.....	112
Что дает предприятию использование сетей?	112
Сети отделов.....	115
Сети кампусов.....	116
Сети масштаба предприятия.....	117
Выводы.....	119
Требования к компьютерным сетям.....	120
Производительность.....	120
Надежность и безопасность.....	123
Расширяемость и масштабируемость	124
Прозрачность.....	125
Поддержка разных видов трафика.....	126
Управляемость	127
Совместимость	128
Качество обслуживания.....	128
Выводы.....	130
Глава 3. Открытые системы и модель OSI.....	132
Многоуровневый подход.....	132
Декомпозиция задачи сетевого взаимодействия	132
Протокол, интерфейс, стек протоколов.....	134
Выводы.....	136
Модель OS!	137
Общая характеристика модели OSI.....	137
Физический уровень	140
Канальный уровень.....	140
Сетевой уровень	142
Транспортный уровень.....	144
Сеансовый уровень.....	145
Представительный уровень.....	145
Прикладной уровень	146
Сетезависимые и сетенезависимые уровни	146
Выводы	148

Стандартизация сетей.....	148
Понятие «открытая система»	148
Модульность и стандартизация.....	150
Источники стандартов	151
Стандартные стеки коммуникационных протоколов	155
Выводы	160
Вопросы и упражнения к части I	161
Часть II. Технологии физического уровня	
Глава 4. Линии связи.....	168
Типы линий связи	168
Среда передачи информации.....	168
Аппаратура линий связи	173
Выводы	176
Характеристики линий связи.....	176
Группы характеристик линий связи.....	176
Спектральный анализ сигналов на линиях связи.....	177
Затухание и волновое сопротивление.....	180
Помехоустойчивость и достоверность	183
Полоса пропускания	185
Выводы.....	187
Пропускная способность.....	187
Определение пропускной способности	187
Связь между пропускной способностью и полосой пропускания линии.....	190
Выводы.....	192
Глава 5. Передача данных на физическом и канальном уровнях	193
Аналоговая модуляция.....	193
Методы аналоговой модуляции.....	194
Спектр модулированного сигнала.....	195
Дискретная модуляция аналоговых сигналов	197
Выводы.....	200
Цифровое кодирование.....	200
Требования к методам цифрового кодирования.....	201
Потенциальный код без возвращения к нулю.....	202
Метод биполярного кодирования с альтернативной инверсией.....	203
Потенциальный код с инверсией при единице.....	204
Биполярный импульсный код	205
Манчестерский код	205
Потенциальный код 2B1Q.....	206
Логическое кодирование.....	206
Выводы.....	210
Особенности протоколов канального уровня.....	211
Асинхронные протоколы	212
Синхронные символьно-ориентированные и бит-ориентированные протоколы.....	214

Передача с установлением соединения и без установления соединения.....	218
Обнаружение и коррекция ошибок	220
Компрессия данных.....	225
Выводы	226
Глава 6. Первичные сети.....	228
Коммутация каналов — основной принцип первичных сетей	228
Коммутация каналов на основе частотного и волнового мультиплексирования.....	228
Коммутация каналов на основе разделения времени	231
Общие свойства сетей с коммутацией каналов.....	234
Обеспечение дуплексного режима работы на основе технологий FDM, ТОМ и WDM.....	235
Выводы	236
Сети PDH.....	236
Иерархия скоростей	236
Методы мультиплексирования	238
Ограничения технологии PDH	239
Выводы.....	240
Сети SDH.....	241
История возникновения.....	243
Иерархия скоростей и методы мультиплексирования	244
Типы оборудования.....	246
Кадры STM-N.....	249
Типовые топологии.....	251
Методы обеспечения живучести сети	252
Синхронизация.....	259
Выводы.....	263
Сети DWDM.....	263
Принципы работы.....	264
Типовые топологии.....	66
Источники излучения.....	270
Оптические усилители	270
Оптическое волокно.....	271
Мультиплексоры ввода-вывода.....	272
Регенераторы	273
Оптические транспондеры.....	274
Оптические кросс-коннекторы.....	274
Выводы.....	276
Вопросы и упражнения к части II.....	277
Часть 111. Локальные сети	
Глава 7. Базовые технологии локальных сетей	280
Общая характеристика протоколов локальных сетей	280
Стандартная топология и разделяемая среда	280
Структура стандартов IEEE 802.x	282

Выводы	286
Протокол LLC	286
Три типа процедур уровня LLC	287
Структура кадров LLC и процедура LLC2.....	288
Выводы	292
Технология Ethernet	292
Метод доступа CSMA/CD	293
Максимальная производительность сети Ethernet	300
Форматы кадров технологии Ethernet	303
Спецификации физической среды Ethernet.....	307
Методика расчета конфигурации сети Ethernet.....	318
Выводы	322
Технология Token Ring	324
Маркерный метод доступа к разделяемой среде.....	324
Форматы кадров Token Ring	327
Физический уровень технологии Token Ring.....	330
Выводы	332
Технология FDDI	333
Основные характеристики технологии	333
Особенности метода доступа FDDI.....	336
Отказоустойчивость технологии FDDI.....	336
Физический уровень технологии FDDI	339
Сравнение FDDI с Ethernet и Token Ring	340
Выводы.....	341
Глава 8. Развитие технологии Ethernet.....	342
Fast Ethernet.....	342
Физический уровень технологии Fast Ethernet	343
Правила построения сегментов Fast Ethernet при наличии повторите- лей.....	350
Особенности технологии 100VG-AnyLAN.....	353
Выводы.....	355
Gigabit Ethernet.....	356
Средства обеспечения диаметра сети в 200 м на разделяемой среде.....	360
Спецификации физической среды стандарта 802.3z.....	361
Gigabit Ethernet на витой паре категории 5	362
Выводы.....	364
Глава 9. Структуризация LAN на физическом и канальном уровнях.....	366
Структурированная кабельная система.....	366
Иерархия в кабельной системе.....	367
Стандарты кабелей для зданий и кампусов	369
Выбор типа кабеля для горизонтальных подсистем	374
Выбор типа кабеля для вертикальных подсистем	377
Выбор типа кабеля для подсистемы кампюса.....	378
Выводы.....	379
Сетевые адAPTERы и концентРАТОРЫ.....	379

Функции и характеристики сетевых адаптеров.....	379
Классификация сетевых адаптеров.....	381
Основные и дополнительные функции концентраторов	382
Отключение портов.....	385
Поддержка резервных связей.....	386
Защита от несанкционированного доступа	387
Многосегментные концентраторы	388
Управление концентратором по протоколу SNMP.....	390
Конструктивное исполнение концентраторов.....	391
Выводы	393
Логическая структуризация сети с помощью мостов и коммутаторов....	394
Недостатки сети на одной разделяемой среде.....	394
Преимущества логической структуризации сети.....	397
Структуризация с помощью мостов и коммутаторов.....	400
Алгоритм работы прозрачного моста	402
Мосты с маршрутизацией от источника	406
Недостатки топологии сети на мостах.....	409
Выводы	411
Коммутаторы и полнодуплексные протоколы локальных сетей	412
Изменения в работе МАС-уровня при полнодуплексной работе.....	417
Проблема управления потоком данных при полнодуплексной работе...	419
Управление потоком кадров при полудуплексной работе.....	422
Выводы.....	423
Особенности реализации коммутаторов.....	424
Архитектура и конструктивное исполнение коммутаторов	425
Характеристики производительности коммутаторов	430
Трансляция протоколов канального уровня.....	435
Возможности коммутаторов по фильтрации трафика.....	436
Приоритетная обработка кадров.....	437
Выводы	438
Глава 10. Дополнительные функции мостов и коммутаторов.....	440
Алгоритм Spanning Tree.....	441
Альтернативные связи	441
Сеть и ее граф	442
Построение дерева в три приема.....	445
Пакеты BPDU и выбор корневого коммутатора	446
Выбор корневых и назначенных портов.....	447
Требуемое время	448
Недостатки и достоинства STA.....	450
Выводы	450
Агрегирование каналов в локальных сетях.....	451
Транки и логические каналы	451
Проблемы транковых соединений	453
Борьба с «размножением» пакетов.....	454
Выбор порта	455

Восстановление при отказе.....	457
Выводы	458
Виртуальные локальные сети.....	458
Назначение VLAN.....	459
Создание VLAN на основе одного коммутатора.....	461
Создание VLAN на основе нескольких коммутаторов.....	462
Выводы.....	464
Типовые схемы применения коммутаторов в локальных сетях.....	465
Сочетание коммутаторов и концентраторов	465
Стянутая в точку магистраль на коммутаторе.....	467
Распределенная магистраль на коммутаторах	468
Выводы.....	469
Вопросы и упражнения к части III.....	470
Часть IV. Составные сети	
Глава 11. Объединение сетей на основе сетевого уровня	476
Понятие «internetworking»	476
Ограничения мостов и коммутаторов.....	476
Архитектура составной сети	477
Принципы маршрутизации	480
Протоколы маршрутизации.....	485
Выводы	486
Реализация межсетевого взаимодействия средствами TCP/IP	487
Многоуровневая структура стека TCP/IP	487
Прикладной уровень	488
Транспортный уровень	488
Уровень межсетевого взаимодействия.....	489
Уровень сетевых интерфейсов.....	490
Сетезависимые и сетенезависимые уровни стека TCP/IP	491
Выводы.....	493
Глава 12. Адресация в IP-сетях.....	495
Типы адресов стека TCP/IP	495
Формы записи IP-адреса	497
Классы IP-адресов	498
Особые IP-адреса	499
Использование масок при IP-адресации.....	501
Выводы.....	502
Порядок назначения IP-адресов.....	503
Централизованное распределение адресов	503
Автоматизация процесса назначения IP-адресов.....	505
Выводы	506
Протоколы разрешения адресов.....	507
Отображение IP-адресов на локальные адреса.....	507
Организация доменов и доменных имен.....	511
Система доменных имен DNS.....	514
Выводы	516

Глава 13. Протоколы межсетевого и транспортного уровней TCP/IP	518
Протокол IP.....	518
Основные функции протокола IP.....	518
Структура IP-пакета.....	519
Выводы.....	522
Таблицы маршрутизации в IP-сетях	522
Примеры таблиц для маршрутизаторов различных типов	523
Назначение полей таблицы маршрутизации	525
Источники и типы записей в таблице маршрутизации	527
Маршрутизация без использования масок.....	529
Выводы.....	532
Маршрутизация с использованием масок.....	533
Использование масок для структуризации сети	533
Использование масок переменной длины	538
Технология бесклассовой междоменной маршрутизации	542
Выводы.....	545
Фрагментация IP-пакетов	546
MTU — параметр технологии.....	546
Параметры фрагментации	547
Порядок сборки фрагментов.....	547
Пример фрагментации	548
Выводы.....	550
Протоколы транспортного уровня TCP и UDP	551
Порты, мультиплексирование и демультиплексирование.....	551
Надежный протокол доставки сообщений TCP.....	552
Соединения.....	554
Последовательный номер и номер подтверждения.....	555
Окно приема	556
Накопительный принцип подтверждения.....	557
Время ожидания квитанции.....	558
Управление окном приема	558
Выводы	559
Глава 14. Протоколы маршрутизации и маршрутизаторы	561
Классификация протоколов маршрутизации.....	561
Назначение протоколов маршрутизации.....	561
Маршрутизация без таблиц.....	562
Адаптивная маршрутизация.....	563
Дистанционно-векторные алгоритмы.....	565
Алгоритмы состояния связей.....	565
Применение нескольких протоколов маршрутизации.....	566
Внешние и внутренние протоколы Интернета.....	567
Выводы.....	569
Дистанционно-векторный протокол RIP	570
Построение таблицы маршрутизации.....	570

Адаптация RIP-маршрутизаторов к изменениям состояния сети.....	575
Методы борьбы с ложными маршрутами в протоколе RIP	578
Выводы.....	580
Протокол состояния связей OSPF	580
Два этапа построения таблицы маршрутизации	580
Маршрутные объявления HELLO	581
Метрики и объявления.....	581
Стабильность OSPF.....	583
Выводы.....	584
Маршрутизаторы.....	585
Основные функции маршрутизатора	585
Дополнительные функциональные возможности маршрутизаторов.....	588
Классификация маршрутизаторов по областям применения.....	590
Основные технические характеристики маршрутизаторов.....	592
Корпоративные модульные концентраторы.....	593
Стирание граней между коммутаторами и маршрутизаторами.....	595
Выводы.....	600
Трансляция сетевых адресов.....	601
Причины подмены адресов.....	602
Традиционная технология NAT	602
Двойная трансляция	606
Выводы.....	607
Средства построения составных сетей стека Novell.....	608
Общая характеристика протокола IPX	608
Формат пакета протокола IPX.....	610
Маршрутизация протокола IPX.....	612
Выводы.....	614
Вопросы и упражнения к части IV	615
Часть V. Глобальные сети	
Глава 15. Качество обслуживания в пакетных сетях.....	620
Основные понятия качества обслуживания.....	620
Типы QoS.....	621
Соглашение об уровне обслуживания	623
Требования разных типов приложений.....	624
Параметры качества обслуживания.....	628
Выводы	629
Служба QoS	630
Модель службы QoS	630
Средства QoS узла.....	631
Протоколы сигнализации QoS	632
Выводы	634
Алгоритмы управления очередями.....	634
Традиционный алгоритм FIFO.....	636
Приоритетное обслуживание.....	636
Взвешенные настраиваемые очереди	639

Взвешенное справедливое обслуживание	641
Механизмы профилирования и формирования трафика.....	642
Планирование качества обслуживания.....	644
Выводы	655
Глава 16. Глобальные сети с коммутацией каналов	656
Аналоговые телефонные сети	657
Устройство	657
Передача данных.....	660
Выводы.....	662
Цифровые сети с интегральными услугами	662
Цели и история создания технологии ISDN.....	662
Пользовательские интерфейсы ISDN.....	665
Подключение пользовательского оборудования к сети ISDN.....	667
Адресация в сетях ISDN.....	669
Стек протоколов и структура сети ISDN.....	671
Использование служб ISDN для передачи данных	673
Выводы.....	675
Глава 17. Глобальные сети с коммутацией пакетов	676
Архитектура и терминология.....	676
Выводы.....	681
Арендуемые элементы глобальной сети.....	682
Выводы.....	682
Подключение к постоянным и коммутируемым каналам.....	683
Выделенные аналоговые каналы.....	683
Модемы для выделенных каналов	684
Модемы для коммутируемых аналоговых линий	687
Устройства DSU/CSU для подключения к выделенному каналу.....	691
Устройства для подключения к коммутируемому каналу ISDN	691
Интерфейсы DTE-DCE	691
Выводы.....	694
Техника виртуальных каналов и дейтаграммные сети	695
Принцип работы.....	695
От общего к частному.....	698
Сравнение	699
Выводы.....	701
Сети X.25.....	701
Назначение и структура сетей X.25.....	701
Адресация в сетях X.25.....	704
Стек протоколов сети X.25	704
Выводы.....	709
Сети frame relay	709
Назначение и общая характеристика.....	709
Стек протоколов frame relay	710
Поддержка качества обслуживания.....	713
Использование сетей frame relay.....	718

Выводы.....	719
Технология ATM	720
Основные принципы технологии ATM.....	722
Стек протоколов ATM.....	728
Уровень адаптации AAL.....	729
Протокол ATM.....	732
Категории услуг протокола ATM и управление трафиком	735
Выводы.....	741
«Чистые» сети IP	741
Общая структура глобальной сети IP.....	741
Протоколы канального уровня для выделенных линий.....	744
Использование выделенных линий для построения пакетной сети.....	749
Выводы.....	753
Сети IP поверх сетей ATM/FR	754
Выводы.....	757
Глава 18. Организация доступа.....	758
Схемы доступа.....	758
Типы взаимодействующих систем	758
Типы поддерживаемых служб.....	759
Типы используемых глобальных служб.....	760
Выводы	761
Доступ компьютер-сеть	762
Удаленный узел.....	763
Удаленное управление и терминальный доступ.....	765
Почта.....	766
Выводы	766
Удаленный доступ через промежуточную сеть.....	767
Общая схема двухступенчатого доступа	767
Технологии xDSL	769
Выводы	771
Вопросы и упражнения к части V	772
Часть VI. Средства анализа и управления сетями	
Глава 19. Функции и архитектура систем управления сетями.....	776
Функциональные группы задач управления	777
Управление конфигурацией сети и именованием.....	777
Обработка ошибок	778
Анализ производительности и надежности.....	778
Управление безопасностью.....	779
Учет работы сети.....	779
Системы управления системами.....	779
Выводы	781
Многоуровневое представление задач управления	781
Элементы сети	782
Управление элементами сети.....	783
Управление сетью.....	783

Управление услугами.....	783
Бизнес-управление.....	784
Выводы	784
Архитектуры систем управления сетями.....	784
Схема менеджер-агент	785
Структуры распределенных систем управления	787
Платформенный подход	789
Выводы.....	790
Глава 20. Стандарты систем управления.....	791
Стандарты систем управления на основе протокола SNMP	792
Концепции SNMP-управления.....	792
Примитивы протокола SNMP.....	793
Структура SNMP MIB.....	794
Форматы и имена объектов SNMP MIB.....	796
Формат сообщений SNMP	798
Спецификация RMON MIB.....	801
Недостатки протокола SNMP.....	803
Выводы.....	804
Стандарты управления OSI	805
Агенты и менеджеры	806
Управление системами, управление уровнем и операции уровня	807
Информационная модель управления.....	808
Управляющие знания и деревья знаний	809
Использование древовидных баз данных для хранения управляющих знаний.....	809
Правила определения управляемых объектов	811
Протокол CMIP и услуги CMIS.....	812
Сравнение протоколов SNMP и CMIP.....	814
Выводы.....	815
Глава 21. Мониторинг и анализ локальных сетей.....	816
Средства мониторинга и анализа	816
Классификация.....	816
Анализаторы протоколов.....	818
Сетевые анализаторы.....	819
Кабельные сканеры и тестеры	820
Многофункциональные портативные приборы мониторинга.....	820
Выводы.....	825
Мониторинг локальных сетей на основе коммутаторов	826
Наблюдение за трафиком.....	826
Управление виртуальными сетями	827
Вопросы и упражнения к части VI	829
Заключение. Заглядывая в будущее.....	830
Приложение. Ответы на вопросы и упражнения	832
Часть I	832

Часть II.....	833
Часть III.....	834
Часть IV.....	836
Часть V.....	837
Часть VI.....	839
Рекомендуемая литература.....	840
Алфавитный указатель	842

Библиотека БГУИР

Благодарности

Мы благодарны рецензентам книги: сотрудникам кафедры «Вычислительная техника» МИРЭА, а также д. т. н., профессору МГТУ им. Н. Э. Баумана Григорьеву Ю. А. и к. т. н., заместителю начальника ИВЦ ОАО Московский ММТ Прижукову Б. Ф. за ценные советы и замечания, позволившие улучшить качество этой книги.

Мы признательны сотрудникам издательства «Питер», особенно главному редактору издательства «Питер» и одновременно самому главному редактору нашей книги Екатерине Строгановой, ее заместителю Евгении Журавлевой, заведующему редакцией компьютерной литературы Илье Корнееву и литературному редактору Алексею Жданову, которые, осуществляя неусыпный надзор над нашим совместным литературным детищем и постоянно напоминая о долгे писателя перед читателем, в конце концов помогли превратить бесплотные файлы из нашего серого ноутбука в красивую книжку с глянцевой обложкой!

Виктор и Наталья Олифер

От авторов

Эта книга является результатом многолетнего опыта преподавания авторами курсов сетевой тематики в аудиториях государственных вузов, коммерческих учебных центров, а также учебных центров крупных предприятий и корпораций.

Основу книги составили материалы курсов «Проблемы построения корпоративных сетей», «Основы сетевых технологий», «Организация удаленного доступа», «Сети TCP/IP», «Стратегическое планирование сетей масштаба предприятия» и ряда других. Эти материалы прошли успешную проверку в бескомпромиссной и сложной аудитории, состоящей из слушателей с существенно разным уровнем подготовки и кругом профессиональных интересов. Среди них были студенты и аспиранты вузов, сетевые администраторы и интеграторы, начальники отделов автоматизации и преподаватели. Учитывая специфику аудитории, курсы лекций строились так, чтобы начинающий получил основу для дальнейшего изучения, а специалист систематизировал и актуализировал свои знания. В соответствии с такими же принципами написана и эта книга — она является фундаментальным курсом по компьютерным сетям, который сочетает широту охватывающей основных областей, проблем и технологий этой быстро развивающейся области знаний с основательным рассмотрением деталей каждой технологии и особенностей оборудования, в котором эта технология реализуется.

За время, прошедшее между первым и вторым изданиями, авторы опубликовали еще две книги, одну из которых — «Сетевые операционные системы» — можно считать прямым продолжением и дополнением книги «Компьютерные сети».

Для кого эта книга

Книга предназначена для студентов, аспирантов и технических специалистов, которые хотят получить базовые знания о принципах построения компьютерных сетей, понять особенности традиционных и перспективных технологий локальных и глобальных сетей, изучить способы создания крупных составных сетей и управления такими сетями.

Книга будет полезна начинающим специалистам в области сетевых технологий, имеющим только общие представления о работе сетей из опыта общения с персональными компьютерами и Интернетом, но хотели бы получить базовые знания, позволяющие продолжить изучение сетей самостоятельно.

Сложившимся сетевым специалистам книга может помочь в знакомстве с теми технологиями, с которыми им не приходилось сталкиваться в практической работе, а также стать справочным средством, в котором можно найти описание конкретного протокола, формата кадра и т. п.

Студенты высших учебных заведений, обучающиеся по направлению «220000. Информатика и вычислительная техника» и по специальностям «Вычислитель-

ные машины, комплексы, системы и сети», «Автоматизированные машины, комплексы, системы и сети», «Программное обеспечение вычислительной техники и автоматизированных систем» могут использовать книгу в качестве рекомендованного Министерством образования Российской Федерации учебного пособия.

Изменения во втором издании

После выхода в свет в 1999 года первого издания книги наша лекционная работа продолжалась, при этом значительная доля заказов на специализированные курсы пришлась на телекоммуникационные компании. Те изменения, которые заказчики просили внести в стандартные программы курсов, подсказали нам основные направления корректировки содержания второго издания книги. Мы благодарны нашим слушателям из компаний «Совинтел», Lucent Technologies, Lattelekom, Центробанк и многих других за плодотворные дискуссии, в результате которых появились новые разделы книги и были изменены (мы надеемся — в лучшую сторону) ряд формулировок и пояснений.

Конечно, за три года, прошедшие после выхода первого издания, сетевой мир не стоял на месте, появлялись новые и совершенствовались существующие протоколы, разрабатывались более современные типы оборудования. Однако все яти изменения не потребовали радикального пересмотра книги, поскольку большая ее часть посвящена традиционным принципам организации сетей, фундаментальным понятиям и базовым, устоявшимся сетевым технологиям. Тем не менее, как уже было сказано, некоторые дополнения были внесены, и ниже перечислены основные из них.

□ Начнем с «идеологического» изменения. Известно, что в эволюции компьютерных сетей основную роль сыграли две мощные ветви технической цивилизации — вычислительная техника и телекоммуникационные сети. Однако в первом издании основное внимание уделялось влиянию только одной из них — вычислительной технике. Во втором издании авторы постарались отдать должное телекоммуникационным технологиям: с этой целью в книгу были включены дополнительные разделы, в частности: «Конвергенция компьютерных и телекоммуникационных сетей», «Общая структура телекоммуникационной сети», «Сети операторов связи», «Услуги, поставщики услуг и сетевая инфраструктура», кроме того, по-новому были расставлены акценты и оценки во многих других разделах книги.

□ С более общих позиций рассматриваются во втором издании вопросы коммутации, при этом охватываются все виды современных телекоммуникационных сетей: первичные, телефонные, сети с коммутацией пакетов дейтаграммного типа и с виртуальными каналами.

□ Существенно расширено описание технологий SDK и DWDM, на основе которых строятся современные первичные сети, представляющие свои высокоскоростные цифровые каналы для вторичных сетей — компьютерных и теле-

фонных.

- Важным дополнением второго издания является глава «Качество обслуживания в пакетных сетях». За те несколько лет, которые прошли с момента публикации первого издания, проблема качества обслуживания (QoS) перестала быть темой, интересной только узким специалистам, занимающимся теоретическими проблемами построения сетей. Сейчас методы обеспечения QoS представляют собой базис современных сетей передачи данных, а значит, без этой темы ни один учебник по сетевым технологиям не может считаться сколько-нибудь полным.
- Во второе издание включено также описание таких повсеместно используемых механизмов, как агрегирование связей в локальных сетях и трансляция сетевых адресов (NAT),
- И, наконец, было исправлено много мелких ошибок и опечаток в тексте и рисунках, замеченных читателями и самими авторами.

Структура книги

Структура книги во втором издании претерпела изменения. Книга состоит из 21 главы, организованных в 6 частей.

- Первая часть, «Основы сетей передачи данных», является вводной. В ней рассматривается роль компьютерных сетей в мире телекоммуникаций, описываются эволюция и основные современные тенденции развития сетевых технологий, в частности конвергенция компьютерных и телекоммуникационных сетей. Далее авторы вводят читателя в курс всех основных проблем, с которыми он встретится в этой книге, причем проблемы эти появляются в естественной последовательности. Как связать компьютер с принтером? Какие задачи надо решить, чтобы обеспечить их электрическую связь между собой? А насколько усложнится задача, если компьютер требуется подсоединить не к периферийному устройству, а к другому компьютеру? И наконец, какие новые задачи появляются, если необходимо связать между собой группу компьютеров? Так читатель подводится к проблемам адресации, топологии, коммутации и мультиплексирования. В этой части впервые вводятся определения основных видов коммуникационных устройств: повторителя, концентратора, моста, коммутатора, маршрутизатора, мультиплексора. Заканчивается первая часть обсуждением вопросов стандартизации вообще и модели ISO/OSI в частности.
- Вторая часть, «Технологии физического уровня», состоит из трех глав, из которых первые две носят вспомогательный характер. В них описываются различные типы линий связи, детально излагаются современные методы передачи дискретной информации в сетях. Наличие этого материала в учебнике дает возможность читателю, не тратя время на просмотр большого количества литературы, получить необходимый минимум знаний в таких областях, как теория информации, спектральный анализ, физическое и логическое кодирование данных, обнаружение и коррекция ошибок. В завершающей главе, «Первичные се-

ти», изучается принцип коммутации каналов и технологии PDH, SDH/SONET, DWDM, создающие канальную инфраструктуру для глобальных телеммуникационных сетей.

□ В третьей части, «Локальные сети», даны подробные описания практически всех основных технологий локальных сетей, как традиционных, таких как Ethernet, Token Ring и FDDI, так и новых высокоскоростных, таких как Fast Ethernet, Gigabit Ethernet. Кроме того, рассматриваются некоторые факты истории становления технологий, их стандартизации и развития, помогающие понять место каждой технологии в ряду ей подобных, а также ее перспективы. Для технологий семейства Ethernet приводятся рекомендованные IEEE методики расчета корректности конфигурации сети. Значительное место уделяется оборудованию локальных сетей, работающему на физическом и канальном уровнях, — структурированным кабельным системам, сетевым адаптерам, повторителям и концентраторам различных технологий, а также мостам и наиболее динамично развивающемуся классу устройств локальных сетей — коммутаторам. Наряду с потребительскими характеристиками коммутаторов приведены типовые схемы их внутренней организации, рассмотрена поддержка ими резервных связей на основе протокола Spanning Tree, а также одна из наиболее перспективных технологий, реализованная в коммутаторах, — технология виртуальных сетей VLAN. Большое внимание в главе уделяется методам логической структуризации сетей, агрегированию физических каналов связи, рассматриваются причины повышения производительности, надежности и управляемости сетей, разделенных на логические сегменты.

□ В четвертой части, «Составные сети», рассмотрены принципы и механизмы объединения сетей на основе протоколов сетевого уровня. Поскольку в настоящее время стек TCP/IP является самым популярным средством организации межсетевого взаимодействия, то изложение материала в данной части проводится на примере IP-сетей. Описываются различные аспекты адресации: способы отображения локальных, сетевых и символьных адресов, использование масок и современных методов агрегирования IP-адресов в Интернете, а также способы автоматического конфигурирования узлов. Детально рассмотрена работа протокола IP по продвижению и фрагментации пакетов, изучается общий формат таблицы маршрутизации и примеры ее частных реализаций в программных и аппаратных маршрутизаторах различных типов. Приводится подробное описание протоколов маршрутизации RIP и OSPF, анализируются области применимости этих протоколов и возможности их комбинирования. Всестороннее рассмотрение в части IV протоколов стека TCP/IP придает ей самостоятельное значение введения в сети IP. Завершает данную часть описание типов и основных характеристик маршрутизаторов, вариантов их внутренней организации, а также методов комбинирования функций коммутации и маршрутизации в одном устройстве — коммутаторе 3-го уровня.

□ В пятой части, «Глобальные сети», подробно изучается основа любой территориальной сети — выделенные аналоговые и цифровые каналы, соответствующее оборудование доступа — модемы различных типов и цифровые устройства DSU/CSU, а также протоколы канального уровня HDLC, PPP и SLIP.

Важнейшей темой и этой части книги является качество обслуживания (QoS), именно на основе этого материала проводится описание широкого спектра технологий глобальных сетей — как классических X.25, так и более современных frame relay и ATM. Наряду с детальным рассмотрением архитектуры, протоколов и услуг сетей каждого типа изучается техника передачи данных на основе виртуальных каналов, отличающая эти технологии от технологий локальных сетей. Изучаются особенности построения глобальных сетей IP, как «чистых» сетей IP, так и наложенных, то есть работающих поверх сетей frame relay и ATM. В завершение рассматриваются схемы и средства удаленного доступа на основе аналоговых телефонных сетей и ISDN, позволяющие домашним и мобильным пользователям разделять ресурсы корпоративной сети.

□ Шестая часть, «Средства анализа и управления сетями», завершающая книгу, посвящена системам управления сетями. Эти сложные, громоздкие и дорогостоящие системы всегда вызывают много вопросов: В каких случаях их применение оправданно? Что делает платформа управления? Возможно ли создание единой системы управления для разнородной сети, построенной на оборудовании разных производителей? Приведенный материал знакомит читателя с набором функций, который должен поддерживаться системой управления в соответствии с международными стандартами, архитектурой систем управления крупными разнородными сетями, назначением платформ управления. Детально описываются протоколы SNMP, СМIP и структура стандартных управляющих баз MIB-I, MIB-II и RMON MIB, встраиваемых в оборудование. Часть завершается рассмотрением портативных приборов, выполняющих функции мониторинга сети.

Авторы стремились сделать работу читателя с книгой максимально эффективной. Подробный алфавитный указатель позволяет быстро найти интересующий материал по одному из многочисленных терминов, используемых в сетевой индустрии. Каждый раздел завершается выводами, которые призваны сконцентрировать внимание читателя на главных идеях, темах и результатах раздела, помогая ему не упустить из виду главное за обилием хотя и полезных, но частных фактов и деталей. В конце каждой части помещены вопросы и упражнения для проверки степени усвоения основных концепций, а в отдельных случаях и для углубления понимания некоторых идей. Ответы на некоторые из этих вопросов вы найдете в приложении к книге.

Дополнительную информацию читатели могут найти на сайте авторов www.olifer.ru, где помещены статьи на специальные темы, приводятся обзоры новых технологий, отрывки из книг, рецензии и рейтинги книг сетевой тематики. Также на сайте можно познакомиться с программами читаемых авторами курсов. С благодарностью примем ваши отзывы, замечания и вопросы по адресу author@olifer.ru.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу электронной почты comp@piter.com (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

На web-сайте издательства <http://www.piter.com> вы найдете подробную информацию о наших книгах.

Библиотека БГУИР

Часть I

Основы сетей передачи данных

ГЛАВА 1 Роль компьютерных сетей в мире телекоммуникаций

Компьютерные сети, называемые также вычислительными сетями, или сетями передачи данных, являются логическим результатом эволюции двух важнейших научно-технических отраслей современной цивилизации — компьютерных и телекоммуникационных технологий. С одной стороны, сети представляют собой частный случай распределенных вычислительных систем, в которых группа компьютеров согласованно выполняет набор взаимосвязанных задач, обмениваясь данными в автоматическом режиме. С другой стороны, компьютерные сети могут рассматриваться как средство передачи информации на большие расстояния, для чего в них применяются методы кодирования и мультиплексирования данных, получившие развитие в различных телекоммуникационных системах.

Эволюция вычислительных систем

Компьютерные сети, которым посвящена данная книга, отнюдь не являются единственным видом сетей, созданным человеческой цивилизацией. Даже водопроводы Древнего Рима можно рассматривать как один из наиболее древних примеров сетей, покрывающих большие территории и обслуживающих многочисленных клиентов. Другой, менее экзотический пример — электрические сети. В них легко можно найти все компоненты любой территориальной сети: источники ресурсов — электростанции, магистрали — высоковольтные линии электропередач, сеть доступа — трансформаторные подстанции, клиентское оборудование — осветительные и бытовые электроприборы.

Авторы отдают себе отчет в том, что многие понятия, концепции, методы и технологии, упоминаемые в данном разделе, возможно, еще не знакомы читателям, только что приступившим к чтению книги. Это касается, например, принципов коммутации каналов и коммутации пакетов, мультиплексирования на основе разделения времени и др. Мы предполагаем, однако, что для понимания эволюции компьютерных сетей достаточно представлять эти концепции в

самых общих чертах, на уровне идей, которые коротко поясняются в этой части книги. А некоторый дискомфорт, который может появиться у читателя из-за обилия новых понятий, компенсируется появлением у него дополнительных стимулов к изучению всего остального материала книги.

Системы пакетной обработки

Обратимся сначала к компьютерному корню вычислительных сетей. Первые компьютеры 50-х годов — большие, громоздкие и дорогие — предназначались для очень небольшого числа избранных пользователей. Часто эти монстры занимали целые здания. Такие компьютеры не были предназначены для интерактивной работы пользователя, а использовались в режиме пакетной обработки. Системы пакетной обработки, как правило, строились на базе мэйнфрейма — мощного и надежного компьютера универсального назначения. Пользователи подготавливали перфокарты, содержащие данные и команды программ, и передавали их в вычислительный центр. Операторы вводили эти карты в компьютер, а распечатанные результаты пользователи получали обычно только на следующий день (рис. 1.1). Таким образом, одна неверно набитая карта означала как минимум суточную задержку.

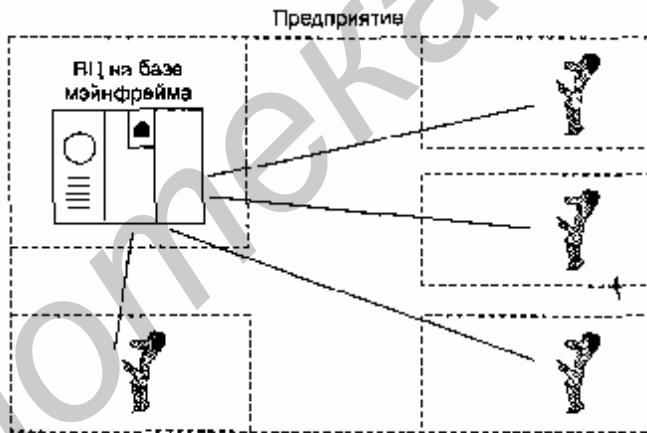


Рис. 1.1. Централизованная система на базе мэйнфрейма.

Конечно, для пользователей интерактивный режим работы, при котором можно с терминала оперативно руководить процессом обработки своих данных, был бы гораздо удобней. Но интересами пользователей на первых этапах развития вычислительных систем в значительной степени пренебрегали, поскольку пакетный режим — это самый эффективный режим использования вычислительной мощности, так как он позволяет выполнить в единицу времени больше пользовательских задач, чем любые другие режимы. Во главу угла ставилась эффективность работы самого дорогого устройства вычислительной машины — процессора, в ущерб эффективности работы использующих его специалистов.

Многотерминальные системы — прообраз сети

По мере удешевления процессоров в начале 60-х годов появились новые спосо-

бы организации вычислительного процесса, которые позволили учесть интересы пользователей. Начали развиваться интерактивные многотерминальные системы разделения времени (рис. 1.2). В таких системах компьютер отдавался в распоряжение сразу нескольким пользователям. Каждый пользователь получал собственный терминал, с помощью которого он мог вести диалог с компьютером. Причем время реакции вычислительной системы было достаточно мало для того, чтобы пользователю была не слишком заметна параллельная работа с компьютером и других пользователей. Разделяя таким образом компьютер, пользователи получили возможность за сравнительно небольшую плату пользоваться преимуществами компьютеризации.

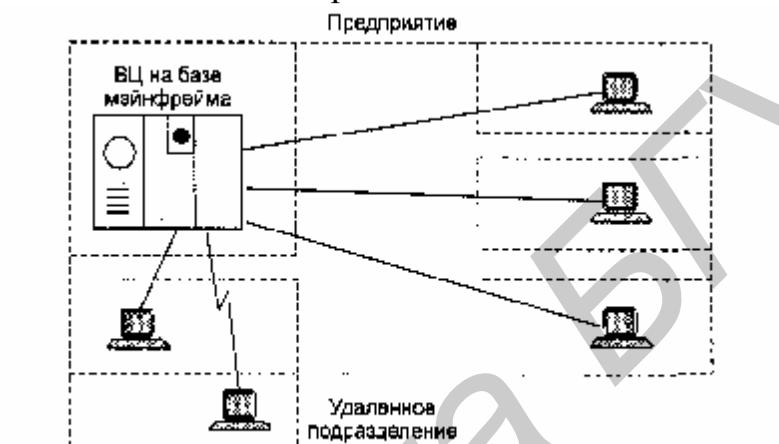


Рис. 1.2. Многотерминальная система — прообраз вычислительной сети

Терминалы, выйдя за пределы вычислительного центра, рассредоточились по всему предприятию. И хотя вычислительная мощность оставалась полностью централизованной, некоторые функции — такие как ввод и вывод данных — стали распределенными. Подобные многотерминальные централизованные системы внешне уже были очень похожи на локальные вычислительные сети. Действительно, рядовой пользователь работу за терминалом майнфрейма воспринимал примерно так же, как сейчас он воспринимает работу за подключенным к сети персональным компьютером. Пользователь мог получить доступ к общим файлам и периферийным устройствам, при этом у него поддерживалась полная иллюзия единоличного владения компьютером, так как он мог запустить нужную ему программу в любой момент и почти сразу же получить результат. (Некоторые, далекие от вычислительной техники пользователи даже были уверены, что все вычисления выполняются внутри их дисплея.)

Таким образом, многотерминальные системы, работающие в режиме разделения времени, стали первым шагом на пути создания локальных вычислительных сетей. Но до появления локальных сетей нужно было пройти еще большой путь, так как многотерминальные системы, хотя и имели внешние черты распределенных систем, все еще сохраняли сущность централизованной обработки данных.

С другой стороны, и потребность предприятий в создании локальных сетей в это время еще не созрела — в одном здании просто нечего было объединять в

сеть, так как из-за высокой стоимости вычислительной техники предприятия не могли себе позволить роскошь приобретения нескольких компьютеров. В этот период был справедлив так называемый «закон Гроша», который эмпирически отражал уровень технологии того времени. В соответствии с этим законом производительность компьютера была пропорциональна квадрату его стоимости, отсюда следовало, что за одну и ту же сумму было выгоднее купить одну мощную машину, чем две менее мощных — их суммарная мощность оказывалась намного ниже мощности дорогой машины.

Глобальные сети

А вот потребность в соединении компьютеров, находящихся на большом расстоянии друг от друга, к этому времени вполне назрела. Началось все с решения более простой задачи — доступа к компьютеру с терминалов, удаленных от него на многие сотни, а то и тысячи километров. Терминалы соединялись с компьютерами через телефонные сети с помощью модемов. Такие сети позволяли многочисленным пользователям получать удаленный доступ к разделяемым ресурсам нескольких мощных компьютеров класса суперЭВМ. Затем появились системы, в которых наряду с удаленными соединениями типа терминал-компьютер были реализованы и удаленные связи типа компьютер-компьютер. Компьютеры получили возможность обмениваться данными в автоматическом режиме, что, собственно, и является базовым механизмом любой вычислительной сети. На основе этого механизма в первых сетях были реализованы службы обмена файлами, синхронизации баз данных, электронной почты и другие, ставшие теперь традиционными сетевые службы.

Таким образом, хронологически первыми появились *глобальные сети (Wide Area Networks, WAN)*, то есть сети, объединяющие территориально рассредоточенные компьютеры, возможно находящиеся в различных городах и странах. Именно при построении глобальных сетей были впервые предложены и отработаны многие основные идеи и концепции современных вычислительных сетей. Такие, например, как многоуровневое построение коммуникационных протоколов, технология коммутации пакетов, маршрутизация пакетов в составных сетях.

Глобальные компьютерные сети очень многое унаследовали от других, гораздо более старых и распространенных глобальных сетей — телефонных. Главным результатом создания первых глобальных компьютерных сетей был отказ от принципа коммутации каналов, на протяжении многих десятков лет успешно использовавшегося в телефонных сетях.

Выделяемый на все время сеанса связи составной канал с постоянной скоростью не мог эффективно использоваться пульсирующим трафиком компьютерных данных, у которого периоды интенсивного обмена чередуются с продолжительными паузами. Натурные эксперименты и математическое моделирование показали, что пульсирующий и в значительной степени не чувствительный к задержкам компьютерный трафик гораздо эффективней передается сетями, использующими принцип коммутации пакетов, когда данные разделяются на

небольшие порции — пакеты, — которые самостоятельно перемещаются по сети за счет встраивания адреса конечного узла в заголовок пакета.

Так как прокладка высококачественных линий связи на большие расстояния обходится очень дорого, то в первых глобальных сетях часто использовались уже существующие каналы связи, изначально предназначенные совсем для других целей. Например, в течение многих лет глобальные сети строились на основе телефонных каналов тональной частоты, способных в каждый момент времени вести передачу только одного разговора в аналоговой форме. Поскольку скорость передачи дискретных компьютерных данных по таким каналам была очень низкой (десятки килобит в секунду), набор предоставляемых услуг в глобальных сетях такого типа обычно ограничивался передачей файлов, преимущественно в фоновом режиме, и электронной почтой. Помимо низкой скорости такие каналы имеют и другой недостаток — они вносят значительные искажения в передаваемые сигналы. Поэтому протоколы глобальных сетей, построенных с использованием каналов связи низкого качества, отличаются сложными процедурами контроля и восстановления данных. Типичным примером таких сетей являются сети X.25, разработанные еще в начале 70-х, когда низкоскоростные аналоговые каналы, арендуемые у телефонных компаний, были преобладающим типом каналов, соединяющих компьютеры и коммутаторы глобальной вычислительной сети.

Прогресс глобальных компьютерных сетей во многом определялся прогрессом телефонных сетей. С конца 60-х годов в телефонных сетях все чаще стала применяться передача голоса в цифровой форме, что привело к появлению высокоскоростных цифровых каналов, соединяющих АТС и позволяющих одновременно передавать десятки и сотни разговоров. Была разработана специальная технология плезиохронной цифровой иерархии (Plesiochronous Digital Hierarchy, PDH), предназначенная для создания так называемых первичных, или опорных, сетей. Такие сети не предоставляют услуг конечным пользователям, они являются фундаментом, на котором строятся скоростные цифровые каналы «точка-точка», соединяющие оборудование другой (так называемой наложенной) сети, которая уже работает на конечного пользователя.

Первоначально технология PDH, поддерживающая скорости до 140 Мбит/с, была внутренней технологией телефонных компаний. Однако со временем эти компании стали сдавать часть своих каналов PDH в аренду предприятиям, которые использовали их для создания собственных телефонных и глобальных компьютерных сетей.

Появившаяся в конце 80-х годов технология синхронной цифровой иерархии (Synchronous Digital Hierarchy, SDH) расширила диапазон скоростей цифровых каналов до 10 Гбит/с, а технология спектрального мультиплексирования (Dense Wave Division Multiplexing, DWDM) — до сотен гигабит и даже нескольких терабит в секунду.

К настоящему времени глобальные сети по разнообразию и качеству сервисов догнали локальные сети, которые долгое время были лидерами в этом отношении, хотя и появились на свет значительно позже.

Локальные сети

Важное событие, повлиявшее на эволюцию компьютерных сетей, произошло в начале 70-х годов. В результате технологического прорыва в области производства компьютерных компонентов появились большие интегральные схемы (БИС). Их сравнительно невысокая стоимость и хорошие функциональные возможности привели к созданию мини-компьютеров, которые стали реальными конкурентами мэйнфреймов. Эмпирический закон Гроша перестал соответствовать действительности, так как десяток мини-компьютеров, имея ту же стоимость, что и мэйнфрейм, выполнял некоторые задачи (как правило, хорошо распараллеливаемые) быстрее.

Даже небольшие подразделения предприятий получили возможность иметь собственные компьютеры. Мини-компьютеры выполняли задачи управления технологическим оборудованием, складом и другие задачи уровня отдела предприятия. Таким образом, появилась концепция распределения компьютерных ресурсов по всему предприятию. Однако при этом все компьютеры одной организации по-прежнему продолжали работать автономно (рис. 1.3).

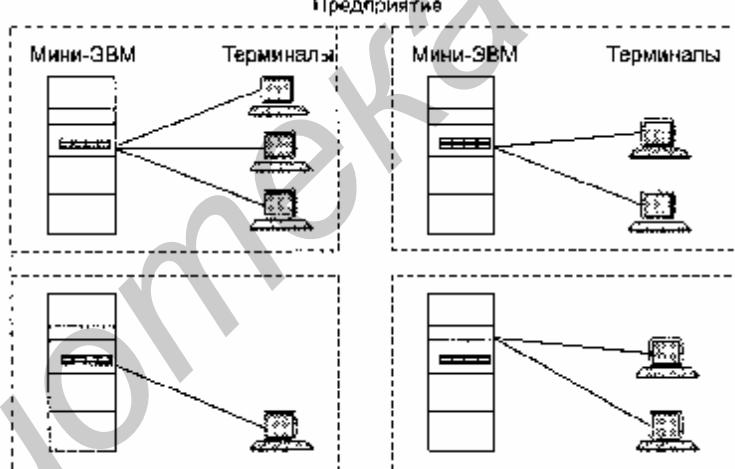


Рис. 1.3. Автономное использование нескольких мини-компьютеров на одном предприятии

Но шло время, потребности пользователей вычислительной техники росли. Их уже не удовлетворяла изолированная работа на собственном компьютере, им хотелось в автоматическом режиме обмениваться компьютерными данными с пользователями других подразделений. Ответом на эту потребность стало появление первых локальных вычислительных сетей (рис. 1.4).

Локальные сети (Local Area Networks, LAN) — это объединение компьютеров, сосредоточенных на небольшой территории, обычно в радиусе не более 1-2 км, хотя в отдельных случаях локальная сеть может иметь и более протяженные размеры, например в несколько десятков километров. В общем случае локальная сеть представляет собой коммуникационную систему, принадлежащую одной организации.

На первых порах для соединения компьютеров друг с другом использовались нестандартные программно-аппаратные средства. Разнообразные устройства сопряжения, использующие свой собственный способ представления данных на линиях связи, свои типы кабелей и т. п., могли соединять только те конкретные модели компьютеров, для которых были разработаны, например, мини-компьютеры PDP-11 с майнфреймом IBM 360 или компьютеры «Наири» с компьютерами «Днепр». Такая ситуация создала большой простор для творчества студентов — названия многих курсовых и дипломных проектов начинались тогда со слов «Устройство сопряжения...».

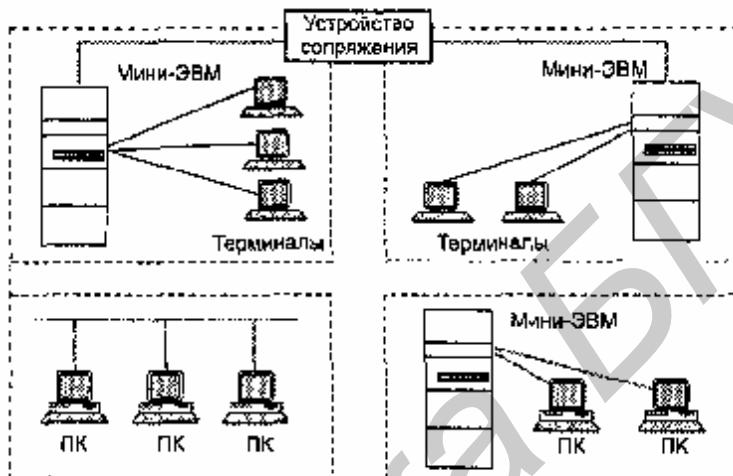


Рис. 1.4. Развличные типы связей в первых локальных сетях

В середине 80-х годов положение дел в локальных сетях стало кардинально меняться. Утвердились стандартные технологии объединения компьютеров в сеть — Ethernet, Arenet, Token Ring, Token Bus, несколько позже — FDDI. Мощным стимулом для их появления послужили персональные компьютеры. Эти массовые продукты явились идеальными элементами для построения сетей — с одной стороны, они были достаточно мощными для работы сетевого программного обеспечения, а с другой — явно нуждались в объединении своей вычислительной мощности для решения сложных задач, а также разделения дорогих периферийных устройств и дисковых массивов. Поэтому персональные компьютеры стали преобладать в локальных сетях, причем не только в качестве клиентских компьютеров, но и в качестве центров хранения и обработки данных, то есть сетевых серверов, потеснив с этих привычных ролей мини-компьютеры и майнфреймы.

Все стандартные технологии локальных сетей опирались на тот же принцип коммутации, который был с успехом опробован и доказал свои преимущества при передаче трафика данных в глобальных компьютерных сетях — принцип коммутации пакетов.

Стандартные сетевые технологии превратили процесс построения локальной сети из искусства в рутинную работу. Для создания сети достаточно было приобрести сетевые адAPTERы соответствующего стандарта, например Ethernet, стандартный кабель, присоединить адAPTERы к кабелю стандартными разъемами

ми и установить на компьютер одну из популярных сетевых операционных систем, например Novell NetWare. После этого сеть начинала работать и последующее присоединение каждого нового компьютера не вызывало никаких проблем — естественно, если на нем был установлен сетевой адаптер той же технологии.

Разработчики локальных сетей привнесли много нового в организацию работы пользователей. Так, намного проще и удобнее стало получать доступ к совместно используемым сетевым ресурсам — в отличие от глобальной в локальной сети пользователь освобождается от запоминания сложных идентификаторов разделяемых ресурсов. Для этих целей система предоставляет ему список ресурсов в удобной для восприятия форме, например в виде древовидной графической структуры («дерева» ресурсов). Еще один прием, рационализирующий работу пользователя в локальной сети, состоит в том, что после соединения с удаленным ресурсом пользователь получает возможность обращаться к нему с помощью тех же команд, которые он использовал при работе с локальными ресурсами. Последствием и одновременно движущей силой такого прогресса стало появление огромного числа непрофессиональных пользователей, освобожденных от необходимости изучать специальные (и достаточно сложные) команды для сетевой работы.

Может возникнуть вопрос — почему все эти удобства пользователи получили только с приходом локальных сетей? Главным образом, это связано с использованием в локальных сетях качественных кабельных линий связи, на которых даже сетевые адAPTERы первого поколения обеспечивали скорость передачи данных до 10 Мбит/с. При небольшой протяженности, свойственной локальным сетям, стоимость таких линий связи была вполне приемлемой. Поэтому экономное расходование пропускной способности каналов, которое было одной из главных целей технологий ранних глобальных сетей, никогда не выходило на первый план при разработке протоколов локальных сетей. В таких условиях основным механизмом прозрачного доступа к сетевым ресурсам локальных сетей стали периодические широковещательные объявления серверов о своих ресурсах и услугах. На основании таких объявлений клиентские компьютеры составляли списки имеющихся в сети ресурсов и предоставляли их пользователю. Конец 90-х выявил явного лидера среди технологий локальных сетей — семейство Ethernet, в которое вошли классическая технология Ethernet 10 Мбит/с, а также Fast Ethernet 100 Мбит/с и Gigabit Ethernet 1000 Мбит/с. Простые алгоритмы работы предопределили низкую стоимость оборудования Ethernet. Широкий диапазон иерархии скоростей позволяет рационально строить локальную сеть, применяя ту технологию семейства, которая в наибольшей степени отвечает задачам предприятия и потребностям пользователей. Важно также, что все технологии Ethernet очень близки друг к другу по принципам работы, что упрощает обслуживание и интеграцию этих сетей.

Выводы

- Вычислительная сеть — это совокупность компьютеров, соединенных ли-

ниями связи. Линии связи образованы кабелями, сетевыми адаптерами и другими коммуникационными устройствами. Все сетевое оборудование работает под управлением системного и прикладного программного обеспечения.

□ Вычислительные сети стали логическим результатом эволюции компьютерных и телекоммуникационных технологий. С одной стороны, они являются частным случаем распределенных вычислительных систем, а с другой стороны, могут рассматриваться как средство передачи информации на большие расстояния, для чего в них применяются методы кодирования и мультиплексирования данных, получившие развитие в различных телекоммуникационных системах.

□ Классифицируя сети по территориальному признаку, различают глобальные (WAN), локальные (LAN) и городские (MAN) сети.

□ Хронологически первыми появились глобальные сети. Они объединяют компьютеры, рассредоточенные на расстоянии сотен и тысяч километров. Традиционные глобальные компьютерные сети очень многое унаследовали от телефонных сетей. В основном они предназначены для передачи данных. В них часто используются уже существующие не очень качественные линии связи, что приводит к более низким, чем в локальных сетях, скоростям передачи данных и ограничивает набор предоставляемых услуг передачей файлов, преимущественно не в оперативном, а в фоновом режиме, с использованием электронной почты.

□ Локальные сети сосредоточены на территории не более 1-2 км; построены с использованием дорогих высококачественных линий связи, которые позволяют, применяя более простые методы передачи данных, чем в глобальных сетях, достигать высоких скоростей обмена данными порядка 100 Мбит/с. Предоставляемые услуги отличаются широким разнообразием и обычно предусматривают реализацию в режиме подключения (on-line).

□ Важнейший этап в развитии сетей — появление стандартных сетевых технологий: Ethernet, FDDI, Token Ring, позволяющих быстро и эффективно объединять компьютеры различных типов.

Конвергенция сетей

Сближение локальных и глобальных сетей

В конце 80-х годов отличия между локальными и глобальными сетями проявлялись весьма отчетливо.

□ Протяженность и качество линий связи. Локальные компьютерные сети по определению отличаются от глобальных сетей небольшими расстояниями между узлами сети. Это в принципе делает возможным использование в локальных сетях более качественных линий связи.

□ Сложность методов передачи данных. В условиях низкой надежности физических каналов в глобальных сетях требуются более сложные, чем в локальных

сетях, методы передачи данных и соответствующее оборудование.

□ Скорость обмена данными в локальных сетях (10, 16 и 100 Мбит/с) в то время была существенно выше, чем в глобальных (от 2,4 кбит/с до 2 Мбит/с).

□ Разнообразие услуг. Высокие скорости обмена данными породили в локальных сетях широкий набор услуг — это различные виды услуг файловой службы, услуги печати, услуги баз данных, электронная почта и др., в то время как глобальные сети в основном предоставляли почтовые услуги и иногда файловые услуги с ограниченными возможностями.

□ Масштабируемость. «Классические» локальные сети обладают плохой масштабируемостью из-за жесткости базовых топологий, определяющих способ подключения станций и длину линии. При этом характеристики сети резко ухудшаются при достижении определенного предела по количеству узлов или протяженности линий связи. Глобальным сетям присуща хорошая масштабируемость, так как они изначально разрабатывались в расчете на работу с произвольными топологиями и сколь угодно большим количеством абонентов.

Постепенно различия между локальными и глобальными типами сетевых технологий стали сглаживаться. Изолированные ранее локальные сети начали объединять друг с другом, при этом в качестве связующей среды использовались глобальные сети. Тесная интеграция локальных и глобальных сетей привела к значительному взаимопроникновению соответствующих технологий.

Сближение в методах передачи данных происходит на платформе цифровой (немодулированной) передачи данных по волоконно-оптическим линиям связи. Этую среду передачи данных используют практически все технологии локальных сетей для скоростного обмена информацией на расстояниях выше 100 м, на ней же построены современные магистрали первичных сетей SDH и DWDM, предоставляющие свои цифровые каналы для объединения оборудования глобальных компьютерных сетей.

Высокое качество цифровых каналов изменило требования к протоколам глобальных компьютерных сетей. На первый план вместо процедур обеспечения надежности вышли процедуры обеспечения гарантированной средней скорости доставки информации пользователем, а также механизмы приоритетной обработки пакетов особенно чувствительного к задержкам трафика, например голосового. Эти изменения нашли отражение в новых технологиях глобальных сетей, таких как frame relay и ATM. В этих сетях предполагается, что искажение битов происходит настолько редко, что ошибочный пакет выгоднее просто уничтожить, а все проблемы, связанные с его потерей, перепоручить программному обеспечению более высокого уровня, которое непосредственно не входит в состав сетей frame relay и ATM.

Большой вклад в сближение локальных и глобальных сетей внесло доминирование протокола IP. Этот протокол сегодня используется поверх любых технологий локальных и глобальных сетей — Ethernet, Token Ring, ATM, frame relay — для создания из различных подсетей единой составной сети.

Компьютерные глобальные сети 90-х, работающие на основе скоростных цифровых каналов, существенно расширили набор своих услуг и догнали в этом отношении локальные сети. Стало возможным создание служб, работа которых

связана с доставкой пользователю больших объемов информации в реальном времени — изображений, видеофильмов, голоса, в общем, всего того, что получило название мультимедийной информации. Наиболее яркий пример — гипертекстовая информационная служба World Wide Web, ставшая основным поставщиком информации в Интернете. Ее интерактивные возможности превзошли возможности многих аналогичных служб локальных сетей, так что разработчикам локальных сетей пришлось просто позаимствовать эту службу у глобальных сетей. Процесс переноса служб и технологий из глобальной сети Интернет в локальные приобрел такой массовый характер, что появился даже специальный термин — *intranet*-технологии (*intra* — внутренний).

В локальных сетях в последнее время уделяется такое же большое внимание методам обеспечения защиты информации от несанкционированного доступа, как и в глобальных сетях. Это обусловлено тем, что локальные сети перестали быть изолированными, чаще всего они имеют выход в «большой мир» через глобальные связи. Защита локальных сетей часто строится на тех же методах — шифрование данных, аутентификация и авторизация пользователей.

И, наконец, появляются новые технологии, изначально предназначенные для обоих видов сетей. Ярким представителем нового поколения технологий является технология ATM, которая может служить основой как глобальных, так и локальных сетей, эффективно объединяя все существующие типы трафика в одной транспортной сети. Другим примером может служить семейство технологий Ethernet, имеющее явные «локальные» корни. Новый стандарт Ethernet 10G, позволяющий передавать данные со скоростью 10 Гбит/с (в момент написания книги его разработка еще продолжалась), предназначен для магистралей как глобальных, так и крупных локальных сетей.

Одним из проявлений сближения локальных и глобальных сетей является появление сетей масштаба большого города, занимающих промежуточное положение между локальными и глобальными сетями. Городские сети, или сети мегаполисов (Metropolitan Area Networks, MAN), предназначены для обслуживания территории крупного города. Эти сети используют цифровые линии связи, часто оптоволоконные, со скоростями на магистрали от 155 Мбит/с и выше. Они обеспечивают экономичное соединение локальных сетей между собой, а также выход в глобальные сети. Эти сети первоначально были разработаны для передачи данных, но сейчас они поддерживают и такие услуги, как видеоконференции и интегральную передачу голоса и текста. Для сетей мегаполисов даже был разработан специальный протокол — SMDS (Switched Multimegabit Data Services), но позднее он был вытеснен более мощной технологией ATM. Современные сети типа MAN отличаются разнообразием предоставляемых услуг, позволяя своим клиентам объединять коммуникационное оборудование различного типа, в том числе и офисные АТС.

Конвергенция компьютерных и телекоммуникационных сетей

Ярко выраженная в последнее время тенденция сближения различных типов сетей характерна не только для локальных и глобальных компьютерных сетей, но

и для телекоммуникационных сетей других типов. К телекоммуникационным сетям, кроме компьютерных, относятся, телефонные сети, радиосети и телевизионные сети. Во всех них в качестве ресурса, предоставляемого клиентам, выступает информация.

Телефонные сети оказывают интерактивные услуги (interactive services), так как два абонента, участвующие в разговоре (или несколько абонентов, если это конференция), попеременно проявляют активность.

Радиосети и телевизионные сети оказывают широковещательные услуги (broadcast services), при этом информация распространяется только в одну сторону — из сети к абонентам, по схеме «один-ко-многим» (point-to-multipoint).

Конвергенция телекоммуникационных сетей идет по многим направлениям.

Прежде всего, наблюдается сближение видов услуг, предоставляемых клиентам. Компьютерные сети изначально разрабатывались для передачи алфавитно-цифровой информации, которую часто называют просто данными (data), в результате у компьютерных сетей имеется и другое название — сети передачи данных (data networks). Телефонные сети и радиосети созданы для передачи только голосовой информации, а телевизионные сети передают голос и изображение.

Первая попытка создания универсальной, так называемой мультисервисной сети, способной оказывать различные услуги, в том числе услуги телефонии и передачи данных, привела к появлению технологии цифровых сетей с интегральными услугами — ISDN. Однако разработчики этой технологии ориентировались на уровень требований по передаче данных, свойственный глобальным компьютерным сетям в 80-х годов. Сегодня же скорость в 2 Мбит/с, на которой происходит подключение к сети крупных абонентов в сети ISDN, уже не может считаться удовлетворительной. Широкомасштабное построение широкополосной (то есть высокоскоростной) сети с интегральными услугами B-ISDN (Broadband ISDN) планировалось осуществить в 90-е годы на базе технологии ATM, обладающей широким диапазоном скоростей, однако по ряду причин сети ATM стали качественной основой только сетей передачи данных, а телефонные услуги и широковещательные услуги радио и телевидения не получили в них большого распространения.

Сегодня на роль глобальной мультисервисной сети нового поколения, часто называемой в англоязычной литературе Next Generation Network (NGN), или New Public Network (NPN), претендует Интернет, но, естественно, не в его нынешнем виде. IP-технологиям, составляющим базис Интернета, еще предстоит пройти значительный путь, чтобы с одинаковым успехом поддерживать услуги WWW и телефонии, архивов данных и видео по требованию, аудио- и видеоновостей, мультимедийной почты. Особую привлекательность представляют собой новые виды комбинированных услуг, в которых сочетаются несколько традиционных услуг, например, услуга универсальной службы сообщений (Unified Messaging), объединяющей электронную почту, телефонию, факсимильную службу и пейджинговую связь. Наибольших успехов на практическом поприще достигла IP-телефония, услугами которой прямо или косвенно сегодня пользуются миллионы людей.

Технологическое сближение сетей происходит сегодня на основе цифровой передачи информации различного типа, метода коммутации пакетов и программирования услуг. Телефония уже давно сделала ряд шагов навстречу компьютерным сетям. Прежде всего, за счет представления голоса в цифровой форме, что делает принципиально возможным передачу телефонного и компьютерного трафика по одним и тем же цифровым каналам (телевидение также может сегодня передавать изображение в цифровой форме). Телефонные сети широко используют комбинацию методов коммутации каналов и пакетов. Так, для передачи служебных сообщений (называемых сообщениями сигнализации) применяются протоколы коммутации пакетов, аналогичные протоколам компьютерных сетей, а для передачи собственно голоса между абонентами коммутируется традиционный составной канал,

Дополнительные услуги телефонных сетей, такие как переадресация вызова, конференц-связь, телеголосование и др., могут создаваться с помощью так называемой интеллектуальной сети (Intelligent Network, IN), по своей сути являющейся компьютерной сетью с серверами, на которых программируется логика услуг.

Сегодня пакетные методы коммутации постепенно теснят традиционные для телефонных сетей методы коммутации каналов даже при передаче голоса. У этой тенденции есть достаточно очевидная причина — на основе метода коммутации пакетов можно более эффективно использовать пропускную способность каналов связи и коммутационного оборудования. Например, паузы в телефонном разговоре могут составлять до 40 % общего времени соединения, однако только пакетная коммутация позволяет «вырезать» паузы и использовать высвободившуюся пропускную способность канала для передачи трафика других абонентов. Другой веской причиной перехода к коммутации пакетов является популярность сети Интернет, построенной на основе данной технологии.

Использование коммутации пакетов для одновременной передачи через пакетные сети разнородного трафика — голоса, видео и текста — сделало актуальным разработку новых методов обеспечения требуемого качества обслуживания (Quality of Service, QoS). Методы QoS призваны минимизировать уровень задержек для чувствительного к ним трафика, например, голосового, и одновременно гарантировать среднюю скорость и динамичную передачу пульсаций для трафика данных. Поставленная задача усложняется тем, что изначально метод коммутации пакетов был рассчитан на слабо чувствительный к задержкам трафик, которому не очень мешали случайные задержки, возникающие при временном размещении пакетов в буфере промежуточного устройства сети. Тем не менее методы поддержки QoS в пакетных сетях вообще и сетях IP в частности уже существуют, причем они сохраняют сравнительно невысокую стоимость инфраструктуры сетевых магистралей. Это очень важно, так как по данному показателю компьютерные сети всегда были предпочтительней телефонных, и сохранение этого соотношения является еще одной причиной построения мультисервисной сети нового поколения на основе пакетных сетей.

Однако неверно было бы говорить, что методы коммутации каналов морально устарели и у них нет будущего. На новом витке развития они находят

свое применение, но только в других формах. Так, их используют сверхскоростные магистрали DWDM, где коммутация происходит на уровне спектральных каналов. В сетях IP сегодня начали активно применять технологию виртуальных каналов (в форме многопротокольной коммутации меток MPLS) — технологию, которая сочетает пакетную коммутацию с устойчивостью путей следования трафика сетей с коммутацией каналов.

Компьютерные сети также многое позаимствовали у телефонных и телевизионных сетей. Глобальные компьютерные сети строятся по такому же иерархическому принципу, что и телефонные, в соответствии с которым сети городов и районов объединяются в региональные сети, а те, в свою очередь, — в национальные и международные сети. Компьютерные сети берут на вооружение методы обеспечения отказоустойчивости телефонных сетей, за счет которых последние демонстрируют высокую степень надежности, так недостающую порой Интернету и корпоративным сетям.

Компьютерные сети успешно используют транспортную инфраструктуру, созданную в рамках тех или иных телекоммуникационных сетей: распределительные сети кабельного телевидения (с помощью кабельных модемов), телефонные абонентские окончания (с помощью оборудования xDSL), телевизионные сети абонентского доступа MMDS и LMDS, а также опорные сети SDH и DWDM. Методы широковещания на основе групповой адресации, эмулирующие широковещание телевизионных и радиосетей, медленно, но верно приживаются в Интернете и других сетях передачи данных.

Сегодня становится все более очевидным, что мультисервисная сеть нового поколения не может быть создана в результате «победы» какой-нибудь одной технологии или подхода. Ее может породить только процесс конвергенции, когда от каждой технологии будет взято все самое лучшее и соединено в некоторый новый сплав, который и даст требуемое качество для поддержки существующих и создания новых услуг. Появился новый термин — инфокоммуникационная сеть, который прямо говорит о двух составляющих современной сети - информационной (компьютерной) и телекоммуникационной.

Выводы

- В конце 80-х годов локальные и глобальные сети имели существенные отличия по протяженности и качеству линий связи, сложности методов передачи данных, скорости обмена данными, разнообразию услуг и масштабируемости.
- В дальнейшем в результате тесной интеграции локальных и глобальных сетей произошло взаимопроникновение соответствующих технологий.
- Одним из проявлений сближения локальных и глобальных сетей является появление сетей масштаба большого города, занимающих промежуточное положение между локальными и глобальными сетями. Городские сети (MAN) предназначены для обслуживания территории крупного города. При достаточно больших расстояниях между узлами (десятки километров) они обладают качественными линиями связи и высокими скоростями обмена, иногда даже более высокими, чем в традиционных локальных сетях. MAN обеспечивают эконо-

мичное соединение локальных сетей между собой, а также выход в глобальные сети.

- Тенденция сближения различных типов сетей характерна не только для локальных и глобальных компьютерных сетей, но и для телекоммуникационных сетей других типов. К телекоммуникационным сетям, кроме компьютерных, относятся телефонные сети, радиосети и телевизионные сети.
- Телефонные сети оказывают интерактивные услуги (interactive services), так как два абонента, участвующие в разговоре (или несколько абонентов, если это конференция), попеременно проявляют активность.
- Радиосети и телевизионные сети оказывают широковещательные услуги (broadcast services), при этом информация распространяется только в одну сторону — из сети к абонентам, по схеме «один ко многим» (point-to-multipoint).
- В настоящее время ведутся активные работы по созданию универсальных мультисервисных сетей, способных совмещать передачу голоса и данных.

Компьютерные сети — частный случай распределенных вычислительных систем

Компьютерные сети, как уже было отмечено, являются продуктом эволюции телекоммуникационных и вычислительных систем. Компьютерные сети относятся к распределенным (или децентрализованным) вычислительным системам. Поскольку основным признаком распределенной вычислительной системы является наличие нескольких центров обработки данных, то наряду с компьютерными сетями к распределенным системам относят также мультипроцессорные компьютеры и многомашинные вычислительные комплексы.

Мультипроцессорные компьютеры

В мультипроцессорных компьютерах имеется несколько процессоров, каждый из которых может относительно независимо от остальных выполнять свою программу. В мультипроцессоре существует общая для всех процессоров операционная система, которая оперативно распределяет вычислительную нагрузку между процессорами. Взаимодействие между отдельными процессорами организуется наиболее простым способом — через общую оперативную память (рис. 1.5).

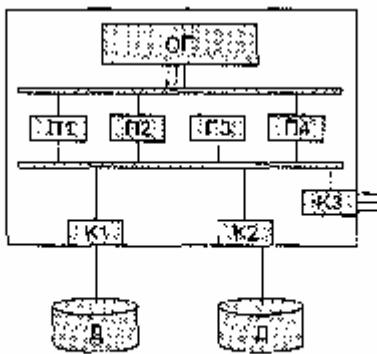


Рис. 1.5. Схема мультипроцессорной системы: ОП — оперативная память, П1-П4 — процессоры, К1-К3 — каналы ввода-вывода, Д — дисковые запоминающие устройства

Сам по себе процессорный блок не является законченным компьютером и поэтому не может выполнять программы без остальных блоков мультипроцессорного компьютера — памяти и периферийных устройств. Все периферийные устройства являются для всех процессоров мультипроцессорной системы общими. Мультипроцессору не свойственна территориальная распределенность — все его блоки располагаются в одном или нескольких близко расположенных конструктивах, как и у обычного компьютера.

Основное достоинство мультипроцессора — его высокая производительность, которая достигается за счет параллельной работы нескольких процессоров. Так как при наличии общей памяти взаимодействие процессоров происходит очень быстро, мультипроцессоры могут эффективно выполнять даже приложения с высокой степенью связи по данным.

Еще одним важным свойством мультипроцессорных систем является отказоустойчивость, то есть способность к продолжению работы при отказах некоторых элементов, например процессоров или блоков памяти. При этом производительность, естественно, снижается, но не до нуля, как в обычных системах, в которых отсутствует избыточность. Однако для того, чтобы мультипроцессор мог продолжать работу после отказа одного из процессоров, необходимо специальное программное обеспечение поддержки отказоустойчивости, которое может быть частью операционной системы или же представлять собой дополнительные служебные программы.

Кластеры

Кластер (многомашинная система) — это вычислительный комплекс, состоящий из нескольких компьютеров (каждый из которых работает под управлением собственной операционной системы), а также программные и аппаратные средства связи компьютеров, которые обеспечивают работу всех компьютеров комплекса как единого целого.

В отличие от мультипроцессора, в котором избыточность реализована на уровне процессорных блоков, кластер состоит из нескольких законченных, способных работать автономно, как правило, стандартных компьютеров, каждый из которых имеет обычную структуру, включающую один или несколько процес-

сорных блоков, оперативную память и периферийные устройства (рис. 1.6). Однако благодаря специальному программному и аппаратному обеспечению межкомпьютерных связей для пользователя кластер выглядит как единый компьютер. При этом каждый компьютер (называемый также узлом кластера) может быть как однопроцессорным, так и мультипроцессорным — на организацию кластера это влияния не оказывает.

Кластеры применяют для повышения надежности и производительности вычислительной системы. Надежность повышается за счет того, что при отказе одного из узлов кластера вычислительная нагрузка (или часть ее) переносится на другой узел. Для выполнения этой операции в кластере используется два типа связей между узлами: межпроцессорные связи и связи за счет разделяемых дисков.

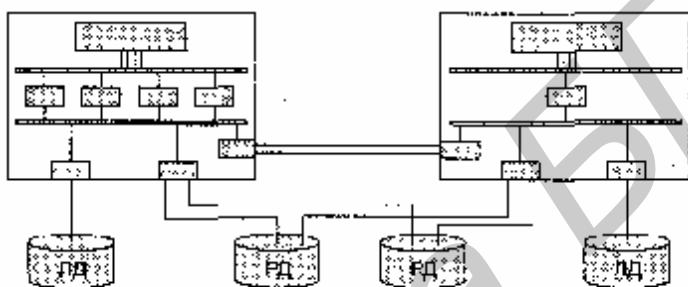


Рис. 1.6. Схема кластера

Межпроцессорные связи используются узлами для обмена служебной информацией. В частности, с помощью этих связей каждый узел кластера периодически проверяет состояние других узлов и выполняемых ими вычислительных задач. Если какой-либо узел или одна из его задач (входящая в набор защищаемых от отказов задач) изменили свое состояние на неработоспособное, то начинается процедура перемещения (реконфигурации) нагрузки на один из работоспособных узлов. В этой процедуре важную роль играют разделяемые диски. Защищаемая задача должна хранить свои данные на одном из таких дисков, чтобы новый узел смог продолжать их использовать после отказа основного. Так как надежность дисковых накопителей достаточно высока (ее можно повысить за счет дополнительных мер, например зеркалирования разделяемого диска), то существенно повышается и надежность кластера по сравнению с отдельным компьютером. Время перевода нагрузки на другой узел кластера при отказе значительно больше, чем время перехода на другой процессор в мультипроцессоре, так как оно связано с активизацией новой копии программного процесса на другом узле. При этом также возможна потеря части данных, находившихся в оперативной памяти отказавшего узла, но для определенных типов вычислительной нагрузки, например систем управления базами данных или web-серверов, эти потери не сказываются на возможности продолжать вычисления.

Если кластер применяется для повышения производительности, то каждая задача, распараллеливается на несколько ветвей, которые выполняются одновременно на нескольких узлах кластера.

Синхронизация работы нескольких копий задачи или их ветвей, а также син-

хронизация вырабатываемых ими данных осуществляются как за счет межпроцессорных связей, так и за счет разделяемой дисковой памяти. Менее тесные и менее скоростные связи между узлами кластера по сравнению со связями процессоров в мультипроцессоре диктуют область применения кластеров — это задачи, достаточно независимые по данным.

Для организации межпроцессорных связей в кластерах часто используются специализированные технологии, приспособленные к решению специфических задач взаимодействия компьютеров в кластере. Однако в последнее время все чаще для этой цели применяются стандартные технологии локальных сетей, например Fast Ethernet и Gigabit Ethernet.

Совместный доступ к дискам также может осуществляться различными способами. Наиболее популярными вариантами является применение интерфейса SCSI и технологии Fibre Channel, тенденция перехода на стандартные технологии локальных сетей в этой области пока выражена не так отчетливо, как в области межпроцессорных связей, но она тоже имеется.

Вычислительные сети

В вычислительных сетях программные и аппаратные связи являются еще более слабыми, а автономность обрабатывающих блоков проявляется в наибольшей степени — основными элементами сети являются стандартные компьютеры, не имеющие ни общих блоков памяти, ни совместно управляемых дисковых накопителей. Связь между компьютерами осуществляется с помощью специальных периферийных устройств — сетевых адаптеров, соединенных относительно протяженными каналами связи. Каждый компьютер работает под управлением собственной операционной системы, а какая-либо «общая» операционная система, распределяющая работу между компьютерами сети, отсутствует. Взаимодействие между компьютерами сети происходит путем передачи сообщений через сетевые адAPTERы и каналы связи. С помощью этих сообщений один компьютер обычно запрашивает доступ к локальным ресурсам другого компьютера. Таким ресурсом может являться, например, вычислительная мощность компьютера в целом, именно этот ресурс использует удаленный пользователь, когда запускает свою программу на «чужом» компьютере. В качестве совместно используемых ресурсов часто выступают данные, хранящиеся на диске, а также разнообразные периферийные устройства — принтеры, модемы, факс-аппараты и т. д.

Разделение локальных ресурсов каждого компьютера между всеми пользователями сети — основная цель создания вычислительной сети. Правда, для этого недостаточно снабдить компьютеры сетевыми адаптерами и соединить их кабельной системой, необходимы еще некоторые добавления к их операционным системам. На тех компьютерах, ресурсы которых должны быть доступны всем пользователям сети, необходимо добавить модули, которые постоянно будут находиться в режиме ожидания запросов, поступающих по сети от других компьютеров. Обычно такие модули называются программными серверами (server), так как их главная задача — обслуживать (serve) запросы на доступ к ресурсам

своего компьютера. На компьютерах, пользователи которых хотят получать доступ к ресурсам других компьютеров, также нужно добавить к операционной системе некоторые специальные программные модули, которые должны вырабатывать запросы на доступ к удаленным ресурсам и передавать их по сети на нужный компьютер. Такие модули обычно называют программными клиентами (client). Собственно сетевые адAPTERЫ и каналы связи решают в сети достаточно простую задачу — они передают сообщения с запросами и ответами от одного компьютера к другому, а основную работу по организации совместного использования ресурсов выполняют клиентские и серверные части операционных систем.

Пара модулей «клиент — сервер» обеспечивает совместный доступ пользователей к определенному типу ресурсов, например к файлам. В этом случае говорят, что пользователь имеет дело с файловой службой (service). Обычно сетевая операционная система поддерживает несколько видов сетевых служб для своих пользователей — файловую службу, службу печати, службу электронной почты, службу удаленного доступа и т. п.

ПРИМЕЧАНИЕ

В технической литературе англоязычный термин «service» обычно переводится как «служба», «сервис» или «услуга». Часто эти термины используются как синонимы. В то же время некоторые специалисты различают термин «служба», с одной стороны, и термины «сервис» и «услуга», с другой. Под «службой» понимается сетевой компонент, который реализует некоторый набор услуг, а «сервисом» называют описание того набора услуг, который предоставляется данной службой. Таким образом, сервис — это интерфейс между потребителем услуг и поставщиком услуг (службой). Далее будет использоваться термин «служба» во всех случаях, когда различие в значении этих терминов не носит принципиального характера.

Термины «клиент» и «сервер» используются не только для обозначения программных модулей, но и компьютеров, подключенных к сети. Если компьютер предоставляет свои ресурсы другим компьютерам сети, то он называется сервером, а если он их потребляет — клиентом. Иногда один и тот же компьютер может одновременно играть роли и сервера, и клиента.

Распределенные программы

Сетевые службы всегда представляют собой распределенные программы. Распределенная программа — это программа, которая состоит из нескольких взаимодействующих частей (в приведенном на рис. 1.7 примере — из двух), причем каждая часть, как правило, выполняется на отдельном компьютере сети.

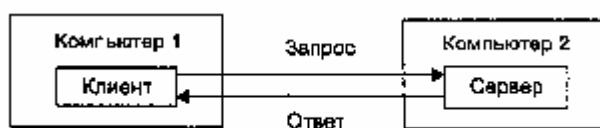


Рис. 1.7. Взаимодействие частей распределённого приложения

Сетевые службы относятся к системным распределенным программам. Однако в сети могут выполняться и распределенные пользовательские программы — приложения. Распределенное приложение также состоит из нескольких частей, каждая из которых выполняет какую-то определенную законченную работу по решению прикладной задачи. Например, одна часть приложения, выполняющаяся на компьютере пользователя, может поддерживать специализированный графический интерфейс, вторая — работать на мощном выделенном компьютере и заниматься статистической обработкой введенных пользователем данных, третья — заносить полученные результаты в базу данных на компьютере с установленной стандартной СУБД. Распределенные приложения в полной мере используют потенциальные возможности распределенной обработки, предоставляемые вычислительной сетью, и поэтому часто называются сетевыми приложениями.

Следует подчеркнуть, что не всякое приложение, выполняемое в сети, является сетевым. Существует большое количество популярных приложений, которые не являются распределенными и целиком выполняются на одном компьютере сети. Тем не менее и такие приложения могут использовать преимущества сети за счет встроенных в операционную систему сетевых служб. Значительная часть истории локальных сетей связана как раз с использованием таких нераспределенных приложений. Рассмотрим, например, как происходила работа пользователя с известной в свое время СУБД dBase. Обычно файлы базы данных, с которыми работали все пользователи сети, располагались на файловом сервере. Сама же СУБД хранилась на каждом клиентском компьютере в виде единого программного модуля. Программа dBase была рассчитана на обработку только локальных данных, то есть данных, расположенных на том же компьютере, что и сама программа. Пользователь запускал dBase на своем компьютере, и программа искала данные на локальном диске, совершенно не принимая во внимание существование сети. Чтобы обрабатывать с помощью dBase данные, расположенные на удаленном компьютере, пользователь обращался к услугам файловой службы, которая доставляла данные с сервера на клиентский компьютер и создавала для СУБД эффект их локального хранения.

Большинство приложений, используемых в локальных сетях в середине 80-х годов, являлись обычными нераспределенными приложениями. И это понятно — они были написаны для автономных компьютеров, а потом просто были перенесены в сетевую среду. Создание же распределенных приложений, хотя и сулило много преимуществ (снижение сетевого трафика, специализация компьютеров), оказалось делом совсем не простым. Нужно было решать множество дополнительных проблем — на сколько частей разбить приложение, какие функции возложить на каждую часть, как организовать взаимодействие этих частей, чтобы в случае сбоев и отказов оставшиеся части корректно завершали работу и т. д., и т. п. Поэтому до сих пор только небольшая часть приложений является распределенными, хотя очевидно, что именно за этим классом приложений будущее, так как они в полной мере могут использовать потенциальные возможности сетей по распараллеливанию вычислений.

Выводы

- Компьютерные сети представляют собой частный случай распределенных вычислительных систем, в которых группа компьютеров согласованно выполняет набор взаимосвязанных задач, обмениваясь данными в автоматическом режиме. Наряду с компьютерными сетями к распределенным системам относят также мультипроцессорные компьютеры и многомашинные вычислительные комплексы.
- В мультипроцессорных компьютерах имеется несколько процессоров, каждый из которых может независимо от остальных обращаться к общей памяти и выполнять собственную программу. Все периферийные устройства являются для всех процессоров мультипроцессорной системы общими. В мультипроцессоре существует общая для всех процессоров операционная система, которая распределяет вычислительную нагрузку между процессорами. Мультипроцессору не свойственна территориальная распределенность — все его блоки располагаются в одном или нескольких близко расположенных конструкивах, как и у обычного компьютера.
- Многомашинный комплекс (клuster) — это вычислительная система, состоящая из нескольких компьютеров (каждый из которых работает под управлением собственной операционной системы), а также программные и аппаратные средства связи компьютеров.
- Разделение локальных ресурсов каждого компьютера между всеми пользователями сети достигается с помощью программных модулей двух типов: клиентов (client), которые формируют запросы на доступ к удаленным компьютерам, и серверов (server), принимающих эти запросы из сети и предоставляющих запрашиваемые ресурсы. Несколько клиентов могут обращаться к одному серверу. Набор модулей «клиент — сервер» представляет собой распределенную программу, реализующую сетевую службу (service).
- Термины «клиент» и «сервер» используются не только для обозначения программных модулей, но и компьютеров, подключенных к сети. Если компьютер предоставляет свои ресурсы другим компьютерам сети, то он называется сервером, а если он их потребляет — клиентом. Иногда один и тот же компьютер может одновременно играть роли и сервера, и клиента.

ГЛАВА 2 Общие принципы построения сетей

При создании вычислительных сетей их разработчикам пришлось решить много проблем. В этом разделе мы рассмотрим только наиболее важные из них, причем в той последовательности, в которой они естественно возникали в процессе развития и совершенствования сетевых технологий.

Связь «точка-точка»

Наиболее простым случаем связи двух устройств является их непосредственное соединение физическим каналом, такое соединение называется связью «точка-точка» (point-to-point).

Связь компьютера с периферийными устройствами

Механизмы взаимодействия компьютеров в сети многое позаимствовали у схемы взаимодействия компьютера с периферийными устройствами, поэтому начнем рассмотрение принципов работы сети с этого «досетевого» случая. Соединение компьютера с периферийным устройством чаще всего представляет собой связь «точка-точка».

Для обмена данными между компьютером и периферийным устройством (ПУ) в компьютере предусмотрен внешний интерфейс, или порт (рис. 2.1), то есть набор проводов, соединяющих компьютер и периферийное устройство, а также набор правил обмена информацией по этим проводам.

Существуют как весьма специализированные интерфейсы, пригодные для подключения узкого класса устройств (например, графических мониторов высокого разрешения фирмы Vista), так и интерфейсы общего назначения, являющиеся стандартными и позволяющие подключать различные периферийные устройства. Примерами стандартных интерфейсов, используемых в компьютерах, являются параллельный интерфейс Centronics, предназначенный, как правило, для подключения принтеров, и последовательный интерфейс RS-232C, который поддерживается многими терминалами, принтерами, графопостроителями, манипуляторами типа «мышь» и многими другими устройствами.

Интерфейс реализуется со стороны компьютера совокупностью аппаратных и программных средств: контроллером ПУ и специальной программой, управляющей этим контроллером, которую часто называют драйвером соответствующего периферийного устройства.

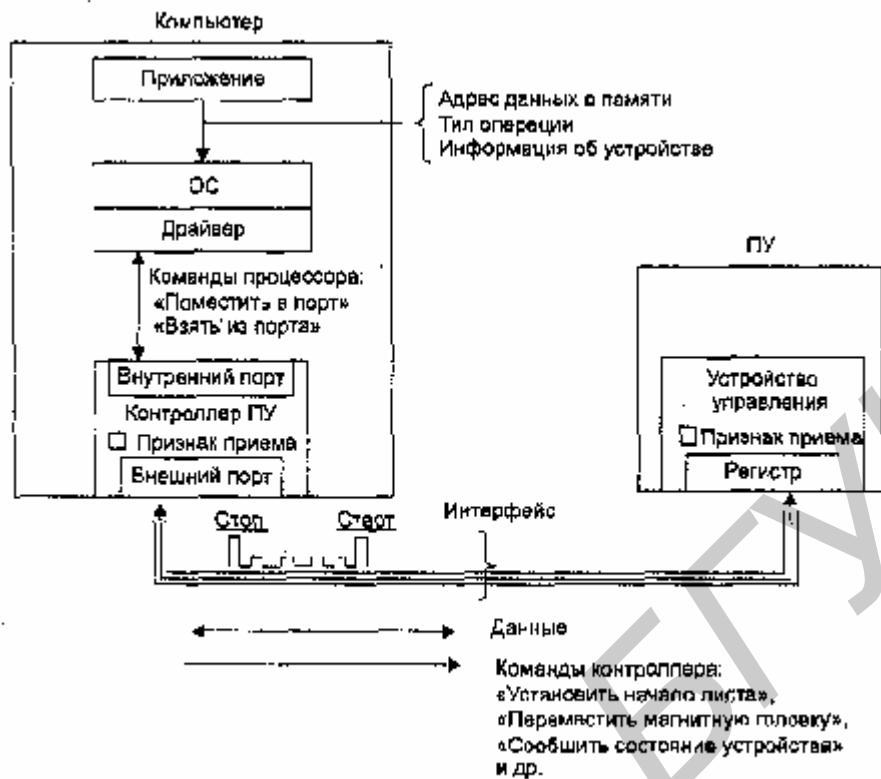


Рис. 2.1. Связь компьютера с периферийным устройством

Со стороны ПУ интерфейс чаще всего реализуется аппаратным устройством управления ПУ, хотя встречаются и программно-управляемые периферийные устройства.

Программа, выполняемая процессором, может обмениваться данными с помощью команд ввода-вывода с любыми модулями, подключенными к внутреннейшине компьютера, в том числе и с контроллерами ПУ.

Периферийные устройства могут принимать от компьютера как данные, например байты информации, которую нужно распечатать на бумаге, так и команды управления, в ответ на которые устройство управления ПУ может выполнить специальные действия, например, перевести головку диска на требуемую дорожку или же вытолкнуть лист бумаги из принтера. Периферийное устройство использует внешний интерфейс компьютера не только для приема информации, но и для передачи информации в компьютер, то есть обмен данными по внешнему интерфейсу, как правило, является двунаправленным. Так, например, даже принтер, который по своей природе является устройством вывода информации, возвращает в компьютер данные о своем состоянии.

Итак, приложение, которому требуется передать некоторые данные на периферийное устройство, обращается с запросом на выполнение операции ввода-вывода к операционной системе. В запросе указываются: адрес данных в оперативной памяти, идентифицирующая информация о периферийном устройстве и операция, которую надо выполнить. Получив запрос, операционная система запускает соответствующий драйвер, передавая ему в качестве параметра адрес выводимых данных. Дальнейшие действия по выполнению операции ввода-

вывода со стороны компьютера реализуются совместно драйвером и контроллером ПУ. Контроллер работает под управлением драйвера. Контроллеры ПУ принимают команды и данные от драйвера в свой внутренний буфер, который часто называется регистром, или портом, а затем производят необходимые преобразования данных и команд, полученных от драйвера, в соответствии с форматами, понятными устройству управления ПУ, и выдают их на внешний интерфейс.

Распределение обязанностей между драйвером и контроллером может быть разным, но чаще всего контроллер поддерживает набор простых команд по управлению периферийным устройством, а драйвер определяет последовательность их выполнения, заставляя периферийное устройство совершать более сложные действия по некоторому алгоритму. Например, контроллер принтера может поддерживать такие элементарные команды, как «Печать символа», «Перевод строки», «Возврат каретки» и т. п. Драйвер же принтера с помощью этих команд организует печать строк символов, разделение документа на страницы и другие более высокоуровневые операции. Для одного и того же контроллера можно разработать различные драйверы, которые с помощью одного и того же набора доступных команд будут реализовывать разные алгоритмы управления ПУ.

Рассмотрим схему передачи одного байта информации от прикладной программы на периферийное устройство. Программа, которой потребовалось выполнить обмен данными с ПУ, обращается к драйверу этого устройства, сообщая ему в качестве параметра адрес байта памяти, который нужно передать. Драйвер загружает значение этого байта в буфер контроллера ПУ, который начинает последовательно передавать биты в линию связи, представляя каждый бит соответствующим электрическим сигналом. Чтобы устройству управления ПУ стало понятно, что начинается передача байта, перед передачей первого бита информации контроллер ПУ формирует стартовый сигнал специфической формы, а после передачи последнего информационного бита — стоповый сигнал. Эти сигналы синхронизируют передачу байта.

Кроме информационных бит, контроллер может передавать бит контроля четности для повышения достоверности обмена. Устройство управления, обнаружив на соответствующей линии стартовый бит, выполняет подготовительные действия и начинает принимать информационные биты, формируя из них байт в своем приемном буфере. Если передача сопровождается битом четности, то выполняется проверка правильности передачи: при правильно выполненной передаче в соответствующем регистре устройства управления устанавливается признак завершения приема информации.

На драйвер обычно возлагаются наиболее сложные функции протокола (например, подсчет контрольной суммы последовательности передаваемых байтов, анализ состояния периферийного устройства, проверка правильности выполнения команды). Но даже самый примитивный драйвер контроллера должен поддерживать как минимум две операции: «Взять данные из контроллера в оперативную память» и «Передать данные из оперативной памяти в контроллер».

Простейший случай взаимодействия двух компьютеров

В самом простом случае связь компьютеров может быть реализована с помощью тех же самых средств, которые используются для связи компьютера с периферией, например, через последовательный интерфейс RS-232C. При этом, в отличие от процедуры обмена данными компьютера с периферийным устройством, когда программа работает, как правило, только с одной стороны (со стороны компьютера), здесь происходит взаимодействие двух программ, выполняемых на каждом из компьютеров.

Программа, работающая на одном компьютере, не может получить непосредственный доступ к ресурсам другого компьютера — его дискам, файлам, принтеру. Она может только «попросить» об этом другую программу, выполняемую на том компьютере, которому принадлежат эти ресурсы. Эти «просьбы» выражаются в виде сообщений, передаваемых по каналам связи между компьютерами. Сообщения могут содержать не только команды на выполнение некоторых действий, но и собственно информационные данные (например, содержимое некоторого файла).

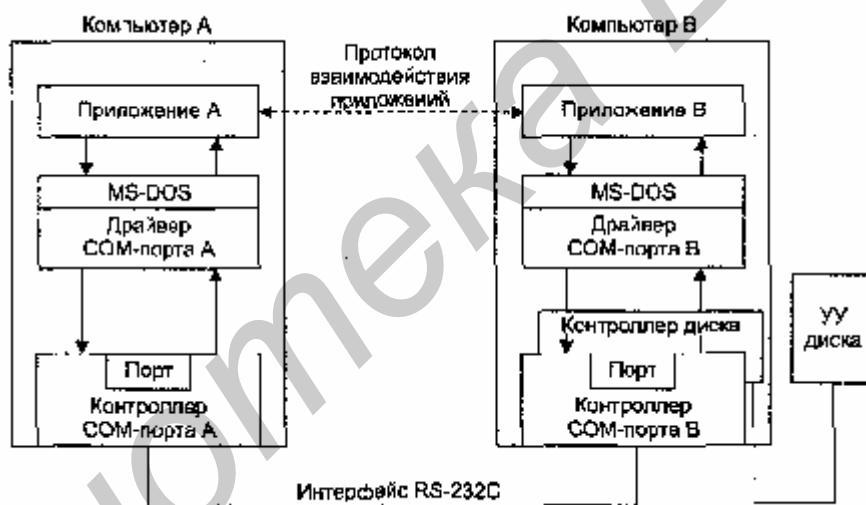


Рис. 2.2. Взаимодействие двух компьютеров

Рассмотрим случай, когда пользователю, работающему с текстовым редактором на персональном компьютере А, нужно прочитать часть некоторого файла, расположенного на диске персонального компьютера В (рис. 2.2). Предположим, что мы связали эти компьютеры по кабелю через СОМ-порты, которые, как известно, реализуют интерфейс RS-232C (такое соединение часто называют нуль-модемным). Пусть для определенности компьютеры работают под управлением MS-DOS, хотя принципиального значения в данном случае это не имеет.

Драйвер СОМ-порта вместе с контроллером СОМ-порта работают примерно так же, как и в описанном выше случае взаимодействия ПУ с компьютером. Однако при этом роль устройства управления ПУ выполняют контроллер и драйвер СОМ-порта другого компьютера. Вместе они обеспечивают передачу по кабелю между компьютерами одного байта информации. (В «настоящих»

локальных сетях подобные функции передачи данных в линию связи выполняются сетевыми адаптерами и их драйверами.)

Драйвер компьютера В периодически опрашивает признак завершения приема, устанавливаемый контроллером при правильно выполненной передаче данных, и при его появлении считывает принятый байт из буфера контроллера в оперативную память, делая его тем самым доступным для программ компьютера В. В некоторых случаях драйвер вызывается асинхронно, по прерываниям от контроллера. Аналогично реализуется и передача байта в другую сторону — от компьютера В к компьютеру А.

Таким образом, в распоряжении программ компьютеров А и В имеется средство для побайтового обмена данными. Но рассматриваемая в нашем примере задача значительно сложнее, так как, во-первых, нужно получить из удаленного компьютера не отдельный байт, а определенную часть заданного файла, во-вторых, эти данные находятся не в оперативной памяти этого компьютера, а на его периферийном устройстве. Все связанные с этим дополнительные проблемы должны решить программы более высокого, чем драйверы СОМ-портов, уровня. Для определенности назовем такие программы компьютеров А и В приложением А и приложением В соответственно.

Итак, приложение А должно сформировать сообщение-запрос для приложения В. В запросе необходимо указать имя файла, тип операции (в данном случае — чтение), смещение и размер области файла, содержащей нужные данные. Это сообщение помещается в буфер в оперативной памяти.

Чтобы передать данный запрос удаленному компьютеру В, приложение А обращается к драйверу СОМ-порта собственного компьютера и сообщает ему адрес буфера, в котором находится сообщение. Затем по только что описанной схеме драйвер и контроллер СОМ-порта компьютера А, взаимодействуя с драйвером и контроллером СОМ-порта компьютера В, передают сообщение байт за байтом приложению В.

Приложение В, получив сообщение, обращается к периферийному устройству, в данном случае диску, в соответствие с рассмотренной ранее схемой «локальная ОС — драйвер ПУ — контроллер ПУ — устройство управления ПУ». Считанные с диска данные приложение В помещает в буферную область оперативной памяти, а далее с помощью драйвера СОМ-порта передает их по каналу связи в компьютер А, где они и попадают к приложению А.

Описанные функции приложения А могла бы выполнить сама программа текстового редактора. Но включать эти функции в состав каждого приложения — текстовых редакторов, графических редакторов, систем управления базами данных и других приложений, у пользователей которых может возникать потребность в доступе к удаленным файлам, — не очень рационально (хотя существует большое количество программ, которые действительно самостоятельно решают все задачи по межмашинному обмену данными, например Kermit — программа обмена файлами через СОМ-порты, реализованная для различных ОС, Norton Commander 3.0 с функцией Link). Гораздо выгоднее создать специальный программный модуль, который будет выполнять функции формирования сообщений-запросов к удаленной машине и приема результатов для всех

приложений. Как уже было ранее сказано, такой служебный модуль называется клиентом.

На стороне же компьютера В должна работать другая специализированная программа — сервер, постоянно ожидающая прихода запросов на удаленный доступ к файлам, расположенным на диске этого компьютера. Сервер, приняв запрос из сети, обращается к локальному файлу, возможно, с участием локальной ОС.

Очень удобной и полезной функцией клиентской программы является способность отличить запрос к удаленному файлу от запроса к локальному файлу. Если клиентская программа умеет это делать, то приложения не должны заботиться о том, с каким файлом они работают (локальным или удаленным), клиентская программа сама распознает и перенаправляет (redirect) запрос к удаленной машине. Отсюда и название, часто используемое для клиентской части, — редиректор. Иногда функции распознавания выделяются в отдельный программный модуль, в этом случае редиректором называют не всю клиентскую часть, а только этот модуль.

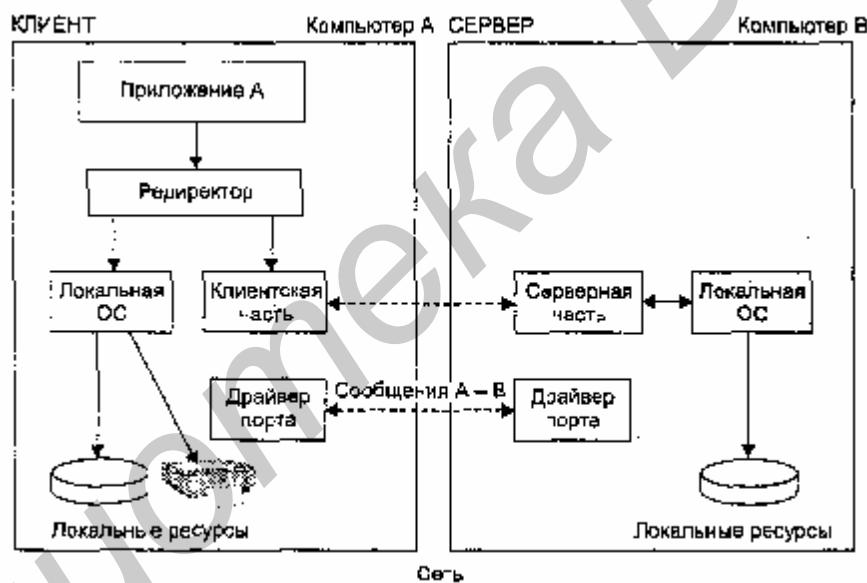


Рис. 2.3. Взаимодействие программных компонентов при связи двух компьютеров

Программные клиент и сервер выполняют системные функции по обслуживанию запросов всех приложений компьютера А на удалённый доступ к файлам компьютера В. Чтобы приложения компьютера В могли пользоваться файлами компьютера А, описанную схему нужно симметрично дополнить клиентом для компьютера В и сервером для компьютера А.

Схема взаимодействия клиента и сервера с приложениями и локальной операционной системой приведена на рис. 2.3. Несмотря на то, что мы рассмотрели очень простую схему связи только двух компьютеров, функции программ, обеспечивающих доступ к удаленным файлам, во многом совпадают с функциями модулей сетевой операционной системы, работающей в сети с более сложными аппаратными связями компьютеров.

Задача физической передачи данных по линиям связи

Даже при рассмотрении простейшей сети, состоящей всего из двух машин, можно увидеть многие проблемы, присущие любой вычислительной сети, в том числе проблемы, связанные с физической передачей сигналов по линиям связи. В вычислительной технике для представления данных используется двоичный код. Внутри компьютера единицам и нулям данных соответствуют дискретные электрические сигналы. Представление данных в виде электрических или оптических сигналов называется кодированием. Существуют различные способы кодирования двоичных цифр 1 и 0, например потенциальный способ, при котором единице соответствует один уровень напряжения, а нулю — другой, или импульсный способ, когда для представления цифр используются импульсы различной или одной полярности.

Аналогичные подходы могут быть использованы для кодирования данных и при передаче их между двумя компьютерами по линиям связи. Однако эти линии связи отличаются по своим электрическим характеристикам от тех, которые существуют внутри компьютера. Главное отличие внешних линий связи от внутренних состоит в их гораздо большей протяженности, а также в том, что они проходят вне экранированного корпуса по пространствам, зачастую подверженным воздействию сильных электромагнитных помех. Все это приводит к существенно большим искажениям прямоугольных импульсов (например, «заваливанию» фронтов), чем внутри компьютера. Поэтому для надежного распознавания импульсов на приемном конце линии связи при передаче данных внутри и вне компьютера не всегда можно использовать одни и те же скорости и способы кодирования. Например, медленное нарастание фронта импульса из-за высокой емкостной нагрузки линии требует передачи импульсов с меньшей скоростью (чтобы передний и задний фронты соседних импульсов не перекрывались и импульс успел дорастить до требуемого уровня).

В вычислительных сетях применяют как потенциальное, так и импульсное кодирование дискретных данных, а также специфический способ представления данных, который никогда не используется внутри компьютера, — модуляцию (рис. 2.4). При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо передает имеющаяся линия связи.

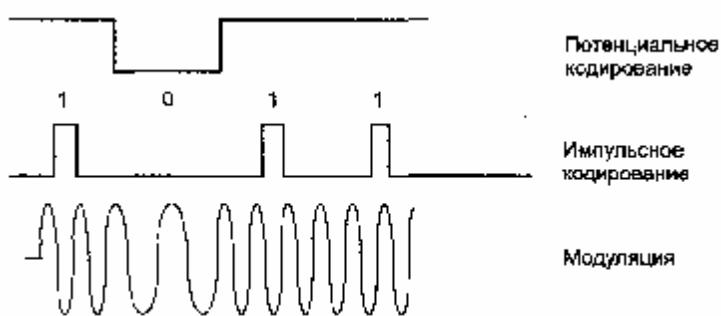


Рис. 2.4. Примеры представления дискретной информации

Потенциальное или импульсное кодирование применяется на каналах высокого качества, а модуляция на основе синусоидальных сигналов предпочтительнее в том случае, когда канал вносит сильные искажения в передаваемые сигналы. Обычно модуляция используется в глобальных сетях при передаче данных через аналоговые телефонные каналы связи, которые были разработаны для передачи голоса в аналоговой форме и поэтому плохо подходят для непосредственной передачи импульсов.

На способ передачи сигналов влияет и количество проводов в линиях связи между компьютерами. Для сокращения стоимости линий связи в сетях обычно стремятся к сокращению количества проводов и из-за этого используют не параллельную передачу всех битов одного байта или даже нескольких байтов, как это делается внутри компьютера, а последовательную, побитную передачу, требующую всего одной пары проводов.

Еще одной проблемой, которую нужно решать при передаче сигналов, является проблема взаимной синхронизации передатчика одного компьютера с приемником другого. При организации взаимодействия модулей внутри компьютера эта проблема решается очень просто, так как в этом случае все модули синхронизируются от общего тактового генератора. Проблема синхронизации при связи компьютеров может решаться разными способами, как путем обмена специальными тактовыми синхроимпульсами по отдельной линии, так и путем периодической синхронизации заранее обусловленными кодами или импульсами характерной формы, отличающейся от формы импульсов данных.

Несмотря на предпринимаемые меры — выбор соответствующей скорости обмена данными, линий связи с определенными характеристиками, способа синхронизации приемника и передатчика, — существует вероятность искажения некоторых битов передаваемых данных. Для повышения надежности передачи данных между компьютерами часто используется стандартный прием — подсчет контрольной суммы и передача ее по линиям связи после каждого байта или после некоторого блока байтов. Часто в протокол обмена данными включается как обязательный элемент сигнал-квитанция, который подтверждает правильность приема данных и посыпается от получателя отправителю.

В каждый сетевой интерфейс, будь то порт маршрутизатора, концентратора или коммутатора, встроены средства в той или иной мере решающие задачу надежного обмена двоичными сигналами, представленными соответствующими электромагнитными сигналами. Некоторые из сетевых устройств, такие как модемы и сетевые адAPTERы, специализируются на проблемах физической передачи данных. Модемы выполняют в глобальных сетях модуляцию и демодуляцию дискретных сигналов, синхронизируют передачу электромагнитных сигналов по линиям связи, проверяют правильность передачи по контрольной сумме и могут выполнять некоторые другие операции. Сетевые адAPTERы рассчитаны, как правило, на работу с определенной передающей средой — коаксиальным кабелем, витой парой, оптоволокном и т. п. Каждый тип передающей среды обладает определенными электрическими характеристиками, влияющими на способ использования данной среды, и определяет скорость передачи сигналов,

способ их кодирования и некоторые другие параметры.

До сих пор мы рассматривали вырожденную сеть, состоящую всего из двух машин. При объединении в сеть большего числа компьютеров возникает целый комплекс новых проблем.

Выводы

- Наиболее простым случаем связи двух устройств является их непосредственное соединение физическим каналом, такое соединение называется связью «точка-точка» (point-to-point).
- Для обмена данными с внешними устройствами (как с собственной периферией, так и с другими компьютерами) в компьютере предусмотрены интерфейсы, или порты, то есть наборы проводов, соединяющих компьютер с устройствами, а также наборы правил обмена информацией по этим проводам.
- Логикой передачи сигналов на внешний интерфейс управляют аппаратное устройство компьютера — контроллер и программный модуль — драйвер.
- Для того чтобы компьютер мог работать в сети, его операционная система должна быть дополнена клиентским и/или серверным модулем, а также средствами передачи данных между компьютерами. В результате такого добавления операционная система компьютера становится сетевой.
- При соединении «точка-точка» на первый план выходит задача физической передачи данных по линиям связи. Эта задача среди прочего включает кодирование и модуляцию данных, взаимную synchronization передатчика одного компьютера с приемником другого, а также подсчет контрольной суммы и передачу ее по линиям связи после каждого байта или после некоторого блока байтов.

Проблемы связи нескольких компьютеров

Топология физических связей

Как только компьютеров становится больше двух, появляется проблема выбора конфигурации физических связей, или топологии. Под топологией сети понимается конфигурация графа, вершинам которого соответствуют конечные узлы сети (например, компьютеры) и коммуникационное оборудование (например, маршрутизаторы), а ребрам — электрические и информационные связи между ними.

Число возможных вариантов конфигураций резко возрастает при увеличении числа связываемых устройств. Так если три компьютера мы можем связать двумя способами (рис. 2.5, а), то для четырех компьютеров можно предложить уже шесть топологически разных конфигураций (при условии неразличимости компьютеров), что и иллюстрирует рис. 2.5, б.

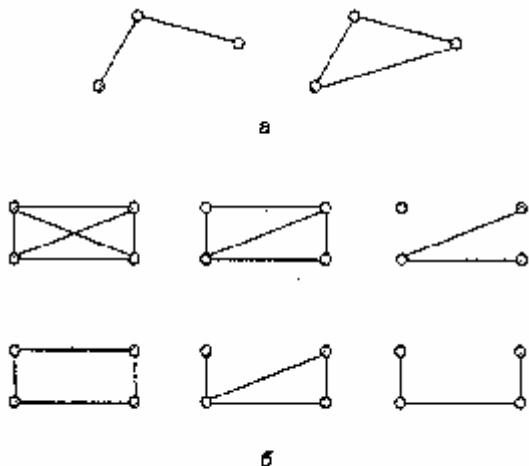


Рис. 2.5. Варианты связей компьютеров

Мы можем соединять каждый компьютер с каждым или же связывать их последовательно, предполагая, что они будут общаться, передавая сообщения друг другу «транзитом». При этом транзитные узлы должны быть оснащены специальными средствами, позволяющими им выполнять эту специфическую посредническую операцию. В качестве транзитного узла может выступать как универсальный компьютер, так и специализированное устройство.

От выбора топологии связей существенно зависят многие характеристики сети. Например, наличие между узлами нескольких путей повышает надежность сети и делает возможным балансировку загрузки отдельных каналов. Простота присоединения новых узлов, свойственная некоторым топологиям, делает сеть легко расширяемой. Экономические соображения часто приводят к выбору топологий, для которых характерна минимальная суммарная длина линий связи.

Среди множества возможных конфигураций различают полносвязные и неполносвязные.

Полносвязная топология (рис. 2.6, а) соответствует сети, в которой каждый компьютер непосредственно связан со всеми остальными. Несмотря на логическую простоту, этот вариант оказывается громоздким и неэффективным. Действительно, в таком случае каждый компьютер в сети должен иметь большое количество коммуникационных портов, достаточное для связи с каждым из остальных компьютеров сети. Для каждой пары компьютеров должна быть выделена отдельная физическая линия связи. (В некоторых случаях даже две, если невозможно использование этой линии для двусторонней передачи.) Полносвязные топологии в крупных сетях применяются редко, так как для связи N узлов требуется $N(N-1)/2$ физических дуплексных линий связей, то есть имеет место квадратичная зависимость. Чаще этот вид топологии используется в многомашинных комплексах или в сетях, объединяющих небольшое количество компьютеров.

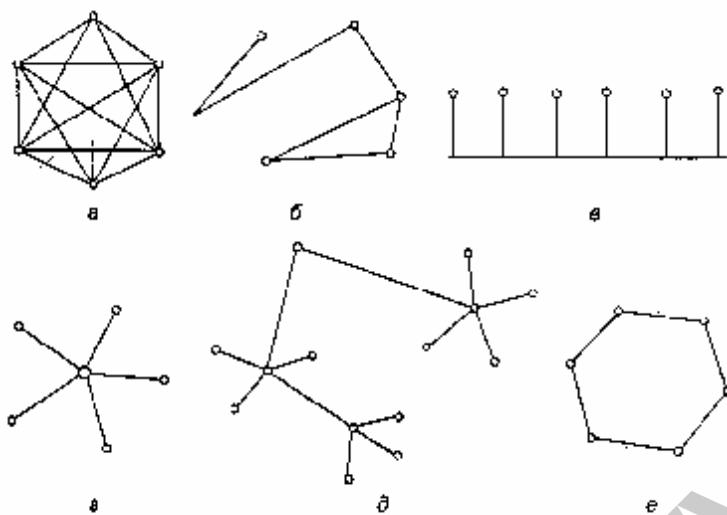


Рис. 2.6. Типовые топологии сетей

Все другие варианты основаны на неполносвязных топологиях, когда для обмена данными между двумя компьютерами может потребоваться промежуточная передача данных через другие узлы сети.

Ячеистая топология (mesh) получается из полносвязной путем удаления некоторых возможных связей (рис. 2.6, б). Ячеистая топология допускает соединение большого количества компьютеров и характерна, как правило, для крупных сетей.

Иногда термин «mesh» используют и для обозначения полносвязной или близкой к полносвязной топологии.

В сетях с кольцевой конфигурацией (рис. 2.6, е) данные передаются по кольцу от одного компьютера к другому. Главным достоинством кольца является то, что оно по своей природе обладает свойством резервирования связей. Действительно, любая пара узлов соединена здесь двумя путями — по часовой стрелке и против нее. Кольцо представляет собой очень удобную конфигурацию и для организации обратной связи — данные, сделав полный оборот, возвращаются к узлу-источнику. Поэтому отправитель может контролировать процесс доставки данных адресату. Часто это свойство кольца используется для тестирования связности сети и поиска узла, работающего некорректно. В то же время в сетях с кольцевой топологией необходимо принимать специальные меры, чтобы в случае выхода из строя или отключения какой-либо станции не прерывался канал связи между остальными станциями кольца.

Топология звезды (рис. 2.6.г) образуется в случае, когда каждый компьютер подключается отдельным кабелем к общему центральному устройству, называемому концентратором. В функции концентратора входит направление передаваемой компьютером информации одному или всем остальным компьютерам сети. В качестве концентратора может выступать как компьютер, так и специализированное устройство, такое как многовходовый повторитель, коммутатор или маршрутизатор. К недостаткам топологии типа звезды относится более высокая стоимость сетевого оборудования из-за необходимости приобретения специализированного центрального устройства. Кроме того, возможности по наращиванию количества узлов в сети ограничиваются количеством портов

концентратора. Иногда имеет смысл строить сеть с использованием нескольких концентраторов, иерархически соединенных между собой связями типа звезды (рис. 2.6, д). Получаемую в результате структуру называют также деревом. В настоящее время дерево является самым распространенным типом топологии связей, как в локальных, так и глобальных сетях.

Особым частным случаем звезды является конфигурация общая шина (рис. 2.6, в). Здесь в качестве центрального элемента выступает пассивный кабель, к которому по схеме «монтажного ИЛИ» подключается несколько компьютеров (такую же топологию имеют многие сети, использующие беспроводную связь — роль общей шины здесь играет общая радиосреда). Передаваемая информация распространяется по кабелю и доступна одновременно всем компьютерам, присоединенным к этому кабелю. Основными преимуществами такой схемы являются ее дешевизна и простота наращивания — то есть присоединения новых узлов к сети. Самый серьезный недостаток общей шины заключается в ее низкой надежности: любой дефект кабеля или какого-нибудь из многочисленных разъемов полностью парализует всю сеть. Другим недостатком общей шины является ее невысокая производительность, так как при таком способе подключения в каждый момент времени только один компьютер может передавать данные по сети, поэтому пропускная способность канала связи всегда делится здесь между всеми узлами сети. До недавнего времени общая шина являлась одной из самых популярных топологий для локальных сетей.

В то время как небольшие сети, как правило, имеют типовую топологию — звезда, кольцо или общая шина, для крупных сетей характерно наличие произвольных связей между компьютерами. В таких сетях можно выделить отдельные, произвольно связанные фрагменты (подсети), имеющие типовую топологию, поэтому их называют сетями со смешанной топологией (рис. 2.7).

В данном случае термин «концентратор» используется в широком смысле, обозначая любое многовходовое устройство, способное служить центральным элементом, например коммутатор или маршрутизатор.

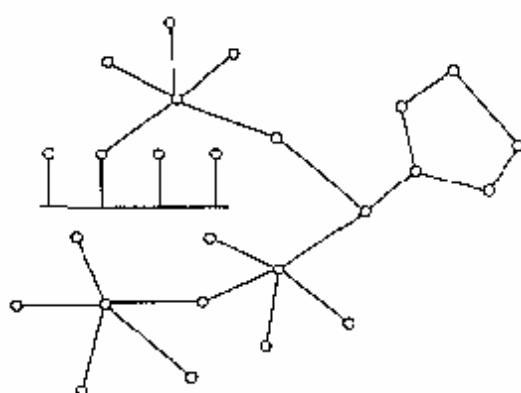


Рис. 2.7. Смешанная топология

Адресация узлов сети

Еще одной новой проблемой, которую нужно учитывать при объединении трех

и более компьютеров, является проблема их адресации, точнее адресации их сетевых интерфейсов. Один компьютер может иметь несколько сетевых интерфейсов. Например, для создания полносвязной структуры из N компьютеров необходимо, чтобы у каждого из них имелся $N - 1$ интерфейс.

Иногда вместо точного выражения «адрес сетевого интерфейса» мы будем использовать упрощенное — «адрес узла сети».

Адреса могут быть числовыми (например, 129.26.255.255) и символьными (site.domen.ги). Один и тот же адрес может быть записан в разных форматах, скажем, числовой адрес в предыдущем примере 125.26.35.18 может быть записан и шестнадцатеричными цифрами — 81.1a.ff.ff.

Адреса могут использоваться для идентификации не только отдельных интерфейсов, но и их групп (групповые адреса). С помощью групповых адресов данные могут направляться одновременно сразу нескольким узлам. Во многих технологиях компьютерных сетей поддерживаются так называемые широковещательные адреса. Данные, направленные по такому адресу, должны быть доставлены всем узлам сети.

Множество всех адресов, которые являются допустимыми в рамках некоторой схемы адресации, называется адресным пространством. Адресное пространство может иметь плоскую (линейную) организацию (рис. 2.8, а) или иерархическую организацию (рис. 2.8, б). В первом случае множество адресов никак не структурировано. При иерархической схеме адресации оно организовано в виде вложенных друг в друга подгрупп, которые, последовательно сужая адресуемую область, в конце концов, определяют отдельный сетевой интерфейс. В показанной на рисунке трехуровневой структуре адресного пространства адрес конечного узла задается тремя составляющими: идентификатором группы {К}, в которую входит данный узел, идентификатором подгруппы {L} и, наконец, идентификатором узла (n), однозначно определяющим его в подгруппе. Иерархическая адресация во многих случаях оказывается более рациональной, чем плоская. В больших сетях, состоящих из многих тысяч узлов, использование плоских адресов может привести к большим издержкам — конечным узлам и коммуникационному оборудованию придется оперировать с таблицами адресов, состоящими из тысяч записей. В противоположность этому, иерархическая система адресации позволяет при перемещении данных до определенного момента пользоваться только старшей составляющей адреса, затем для дальнейшей локализации адресата воспользоваться следующей по старшинству частью и в конечном счете — младшей частью. Примером иерархически организованных адресов являются обычные почтовые адреса, в которых последовательно уточняется место нахождения адресата: страна, город, улица, дом, квартира.

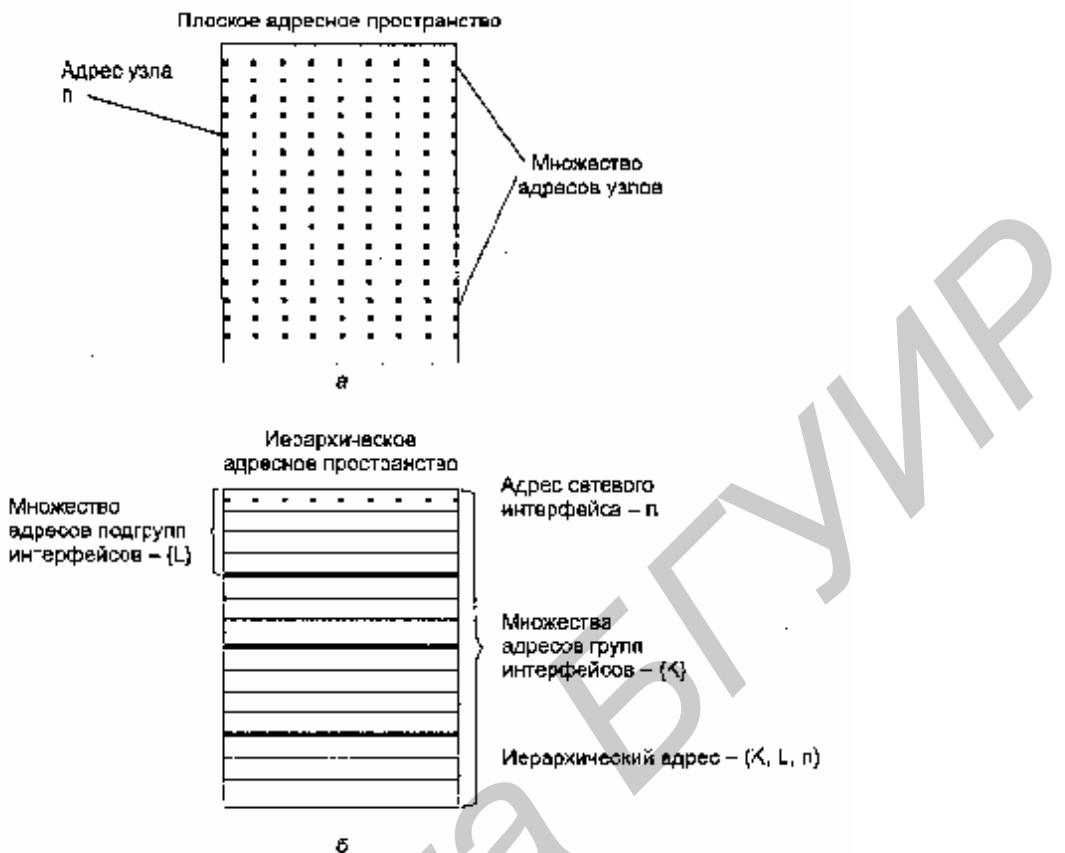


Рис. 2.8. Иерархическая структура адресного пространства

К адресу сетевого интерфейса и схеме его назначения можно предъявить несколько требований.

- Адрес должен уникально идентифицировать сетевой интерфейс в сети любого масштаба.
- Схема назначения адресов должна сводить к минимуму ручной труд администратора и вероятность дублирования адресов.
- Желательно, чтобы адрес имел иерархическую структуру, удобную для построения больших сетей.
- Адрес должен быть удобен для пользователей сети, а это значит, что он должен допускать символьное представление, например Server3 или www.Cisco.com.
- Адрес должен быть по возможности компактным, чтобы не перегружать память коммуникационной аппаратуры — сетевых адаптеров, маршрутизаторов и т. п.

Нетрудно заметить, что эти требования противоречивы — например, адрес, имеющий иерархическую структуру, скорее всего будет менее компактным, чем плоский. Символьные имена удобны для людей, но из-за переменного формата и потенциально большой длины их передача по сети не очень экономична. Так как все перечисленные требования трудно совместить в рамках какой-либо одной схемы адресации, на практике обычно используется сразу несколько схем, так что сетевой интерфейс компьютера может одновременно иметь несколько адресов-имен. Каждый адрес задействуется в той ситуации,

когда соответствующий вид адресации наиболее удобен. А для преобразования адресов из одного вида в другой используются специальные вспомогательные протоколы, которые называют иногда протоколами разрешения адресов (address resolution protocols).

Примером плоского числового адреса является МАС-адрес, предназначенный для однозначной идентификации сетевых интерфейсов в локальных сетях. Такой адрес обычно используется только аппаратурой, поэтому его стараются сделать по возможности компактным и записывают в виде двоичного или шестнадцатеричного значения, например 0081005e24a8. При задании МАС-адресов не требуется выполнение ручной работы, так как они обычно встраиваются в аппаратуру компанией-изготовителем, поэтому их называют также аппаратными (hardware) адресами. Использование плоских адресов является жестким решением — при замене аппаратуры, например сетевого адаптера, изменяется и адрес сетевого интерфейса компьютера.

Типичными представителями иерархических числовых адресов являются сетьевые IP- и IPX-адреса. В них поддерживается двухуровневая иерархия, адрес делится на старшую часть — номер сети и младшую — номер узла. Такое деление позволяет передавать сообщения между сетями только на основании номера сети, а номер узла используется после доставки сообщения в нужную сеть; точно так же, как название улицы используется почтальоном только после того, как письмо доставлено в нужный город. В последнее время, чтобы сделать маршрутизацию в крупных сетях более эффективной, предлагаются более сложные варианты числовой адресации, в соответствии с которыми адрес имеет три и более составляющих. Такой подход, в частности, реализован в новой версии протокола IPv6, предназначенного для работы в сети Интернет,

Символьные адреса или имена предназначены для запоминания людьми и поэтому обычно несут смысловую нагрузку. Символьные адреса легко использовать как в небольших, так и крупных сетях. Для работы в больших сетях символное имя может иметь иерархическую структуру, например ftp-arch1.ucl.ac.uk. Этот адрес говорит о том, что данный компьютер поддерживает ftp-архив в сети одного из колледжей Лондонского университета (University College London — ucl) и эта сеть относится к академической ветви (ac) Интернета Великобритании (United Kingdom - uk). При работе в пределах сети Лондонского университета такое длинное символное имя явно избыточно и вместо него удобно пользоваться кратким символьным именем, на роль которого хорошо подходит самая младшая составляющая полного имени, то есть имя ftp-arch1.

В современных сетях для адресации узлов применяются, как правило, одновременно все три приведенные выше схемы. Пользователи адресуют компьютеры символьными именами, которые автоматически заменяются в сообщениях, передаваемых по сети, числовыми номерами. С помощью этих числовых номеров сообщения передаются из одной сети в другую, а после доставки сообщения в сеть назначения вместо числового номера используется аппаратный адрес компьютера. Сегодня такая схема характерна даже для небольших автономных сетей, где, казалось бы, она явно избыточна — это делается для того, чтобы при

включении этой сети в большую сеть не нужно было менять состав операционной системы.

Проблема установления соответствия между адресами различных типов, которой занимаются протоколы разрешения адресов, может решаться как полностью централизованными, так и распределенными средствами. В случае централизованного подхода в сети выделяется один или несколько компьютеров (серверов имен), в которых хранится таблица соответствия друг другу имен различных типов, например символьных имен и числовых номеров. Все остальные компьютеры обращаются к серверу имен, чтобы по символьному имени найти числовой номер компьютера, с которым необходимо обменяться данными.

При другом, распределенном, подходе каждый компьютер сам решает задачу установления соответствия между адресами. Например, если пользователь указал для узла назначения числовой номер, то перед началом передачи данных компьютер-отправитель посылает всем компьютерам сети широковещательное сообщение с просьбой опознать это числовое имя. Все компьютеры, получив это сообщение, сравнивают заданный номер со своим собственным. Тот компьютер, у которого обнаружилось совпадение, посылает ответ, содержащий его аппаратный адрес, после чего становится возможным отправка сообщений по локальной сети.

Распределенный подход хорош тем, что не предполагает выделения специального компьютера, который к тому же часто требует ручного задания таблицы соответствия адресов. Недостатком распределенного подхода является необходимость широковещательных сообщений — такие сообщения перегружают сеть, так как они требуют обязательной обработки всеми узлами, а не только узлом назначения. Поэтому распределенный подход используется только в небольших локальных сетях. В крупных сетях распространение широковещательных сообщений по всем ее сегментам становится практически нереальным, поэтому для них характерен централизованный подход. Наиболее известной службой централизованного разрешения адресов является система доменных имен (Domain Name System, DNS) сети Интернет.

До сих пор мы говорили об адресах сетевых интерфейсов, которые указывают на порты узлов сети (компьютеров и коммуникационных устройств), однако конечной целью данных, пересылаемых по сети, являются не компьютеры или маршрутизаторы, а выполняемые на этих устройствах программы — процессы. Поэтому в адресе назначения наряду с информацией, идентифицирующей порт устройства, должен указываться адрес процесса, которому предназначены посылаемые по сети данные. После того как эти данные достигнут указанного в адресе назначения сетевого интерфейса, программное обеспечение компьютера должно их направить соответствующему процессу. Понятно, что адрес процесса не обязательно должен задавать его однозначно в пределах всей сети, достаточно обеспечить его уникальность в пределах компьютера. Примером адресов процессов являются номера портов TCP и UDP, используемые в стеке TCP/IP.

Коммутация и мультиплексирование

Обобщенная задача коммутации

Если топология сети не полносвязная, то обмен данными между произвольной парой конечных узлов (абонентов) должен идти в общем случае через транзитные узлы. Например, в сети, показанной на рис. 2.9, узлы 2 и 4, не связанные непосредственно, вынуждены передавать данные через транзитные узлы, в качестве которых могут выступить, например, узлы 1 и 5. Узел 1 должен выполнить передачу данных с интерфейса A на интерфейс B, а узел 5 — с интерфейса F на B. Последовательность транзитных узлов (сетевых интерфейсов) на пути от отправителя к получателю называется маршрутом.

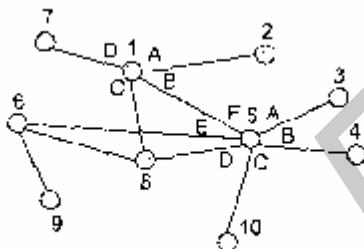


Рис. 2.9. Коммутация абонентов через сеть транзитных узлов

В самом общем виде задача соединения конечных узлов через сеть транзитных узлов называется задачей коммутации. Она может быть представлена в виде нескольких взаимосвязанных частных задач.

1. Определение информационных потоков, для которых требуется прокладывать пути.
2. Определение маршрутов для потоков.
3. Сообщение о найденных маршрутах узлам сети.
4. Продвижение потоков, то есть распознавание потоков и их локальная коммутация на каждом транзитном узле.
5. Мультиплексирование и демультиплексирование потоков.

Определение информационных потоков

Понятно, что через один транзитный узел может проходить несколько маршрутов, например, через узел 5 (см. рис. 2.9) проходят данные, направляемые узлом 4 каждому из остальных узлов, а также все данные, поступающие в узлы 3, 7 и 10. Транзитный узел должен уметь распознавать потоки данных, которые на него поступают, для того чтобы отрабатывать их передачу именно на тот свой интерфейс, который ведет к нужному узлу. Информационным потокам, или потоком данных (data flow, data stream), называют непрерывную последовательность байтов (которые могут быть агрегированы в более крупные единицы данных — пакеты, кадры, ячейки), объединенных набором общих признаков, выделяющих его из общего сетевого трафика. Например, все данные, поступающие от одного компьютера, можно определить как единый поток, а можно представить их как

совокупность нескольких подпотоков, каждый из которых в качестве дополнительного признака имеет адрес назначения. Каждый же из этих подпотоков, в свою очередь, можно разделить на подпотоки данных, относящихся к разным сетевым приложениям — электронной почте, копированию файлов, обращению к web-серверу.

В качестве обязательного признака при коммутации выступает адрес назначения данных, поэтому весь поток входящих в транзитный узел данных должен разделяться как минимум на подпотоки, имеющие различные адреса назначения. Тогда каждой паре конечных узлов будет соответствовать один поток и один маршрут. Однако, как уже было сказано, поток данных между двумя конечными узлами в общем случае может быть представлен несколькими разными потоками, причем для каждого из них может быть проложен свой особый маршрут. Действительно, на одной и той же паре конечных узлов может выполняться несколько взаимодействующих по сети приложений, которые предъявляют к ней свои особые требования. В таком случае выбор пути должен осуществляться с учетом характера передаваемых данных, например, для файлового сервера важно, чтобы передаваемые им большие объемы данных направлялись по каналам, обладающим высокой пропускной способностью, а для программной системы управления, которая посыпает в сеть короткие сообщения, требующие обязательной и немедленной отработки, при выборе маршрута более важна надежность линии связи и минимальный уровень задержек на маршруте. Кроме того, даже для данных, предъявляющих к сети одинаковые требования, может прокладываться несколько маршрутов, чтобы за счет распараллеливания добиться одновременного использования различных каналов и тем самым ускорить передачу данных.

Признаки потока могут иметь глобальное или локальное значение — в первом случае они однозначно определяют поток в пределах всей сети, а во втором — в пределах одного транзитного узла. Пара уникальных адресов конечных узлов для идентификации потока — это пример глобального признака. Примером признака, локально определяющего поток в пределах устройства, может служить номер (идентификатор) интерфейса устройства, с которого поступили данные. Например, узел 1 (см. рис. 2.9) может быть настроен передавать все данные, поступившие с интерфейса А, на интерфейс В, а данные, поступившие с интерфейса D, на интерфейс С. Такое правило позволяет отделить поток данных, поступающий из узла 2, от потока данных из узла 7 и направлять их для транзитной передачи через разные узлы сети, в данном случае данные из узла 2 через узел 5, а данные из узла 7 — через узел 8.

Существует особый тип признака — метка потока. Метка может иметь глобальное значение, уникально определяющее поток в пределах сети. В таком случае она назначается данным потоком на всем протяжении его пути следования от узла источника до узла назначения. В некоторых технологиях используются локальные метки потока, динамически меняющие свое значение при передаче данных от одного узла к другому.

Таким образом, при распознавании потоков во время коммутации в общем случае должны учитываться не только адреса назначения данных, но и другие их

признаки, которые влияют на маршрут перемещения данных по сети.

Определение маршрутов

Выбрать путь, или маршрут, передачи данных — значит определить последовательность транзитных узлов и их интерфейсов, через которые надо передавать данные, чтобы доставить их адресату. Определение пути — сложная задача, особенно когда конфигурация сети такова, что между парой взаимодействующих сетевых интерфейсов существует множество путей. Следует заметить, что множество альтернативных путей между двумя конечными узлами — это лишь множество потенциальных возможностей. Задача определения маршрутов состоит в выборе из всего этого множества одного или нескольких путей. И хотя в частном случае множества имеющихся и выбранных путей могут совпадать, чаще всего выбор останавливают на одном оптимальном¹ по некоторому критерию маршруте.

¹На практике для уменьшения вычислительной работы ограничиваются поиском не оптимального в математическом смысле, а рационального, то есть близкого к оптимальному, маршрута. Это касается, прежде всего, случая, когда задача выбора маршрута формулируется как задача многокритериальной оптимизации.

В качестве критериев оптимальности могут выступать, например, номинальная пропускная способность; загруженность каналов связи; задержки, вносимые каналами; количество промежуточных транзитных узлов; надежность каналов и транзитных узлов. Но даже в том случае, когда между конечными узлами существует только один путь, его нахождение может представлять собой при сложной топологии сети нетривиальную задачу.

Маршрут может определяться эмпирически («вручную») администратором сети, который, используя различные, часто не формализуемые соображения, анализирует топологию сети и определяет последовательность интерфейсов, которую должны пройти данные, чтобы достичь получателя. Среди побудительных мотивов выбора того или иного пути могут быть: особые требования к сети со стороны различных типов приложений, решение передавать трафик через сеть определенного поставщика услуг, предположения о пиковых нагрузках на некоторые каналы сети, соображения безопасности.

Однако эвристический подход к определению маршрутов мало пригоден для большой сети со сложной топологией. В этом случае такая задача решается чаще всего автоматически. Для этого конечные узлы и другие устройства сети оснащаются специальными программными средствами, которые организуют взаимный обмен служебными сообщениями, позволяющий каждому узлу составить свое представление о топологии сети. Затем на основе этого исследования и математических алгоритмов определяются рациональные маршруты.

Оповещение сети о выбранном маршруте

После того как маршрут определен (вручную или автоматически), надо «сообщить» о нем всем устройствам сети. Сообщение о маршруте должно нести каждому транзитному устройству примерно такую информацию: «если придут данные, относящиеся к потоку p , то нужно передать их на интерфейс F ». Сообщение о маршруте обрабатывается устройством, в результате создается новая запись в таблице коммутации, в которой локальному или глобальному признаку (признакам) потока (например, метке, номеру входного интерфейса или адресу назначения) ставится в соответствие номер интерфейса, на который устройство должно передавать данные, относящиеся к этому потоку. Конечно, детальное описание структуры сообщения о маршруте и содержимого таблицы коммутации зависит от конкретной технологии, однако эти особенности не сказываются на существе рассматриваемых процессов.

Передача информации о выбранных маршрутах так же, как и определение маршрута, может осуществляться и вручную, и автоматически. Администратор сети может зафиксировать маршрут, выполнив в ручном режиме конфигурирование устройства, например, жестко скоммутировав на длительное время определенные пары входных и выходных интерфейсов (так, например, работали «телефонные барышни» на первых коммутаторах). Он может также по собственной инициативе внести запись о маршруте в таблицу коммутации. Однако поскольку топология сети и информационных потоков может меняться (отказ или появление новых промежуточных узлов, изменение адресов или определение новых потоков), то гибкое решение задач определения и задания маршрутов предполагает постоянный анализ состояния сети и обновление маршрутов и таблиц коммутации, что очень сложно реализовать вручную.

Продвижение потоков

Далее, когда задачи определения и задания маршрута решены, должно произойти соединение, или коммутация, абонентов. Для каждой пары абонентов эта операция может быть представлена совокупностью нескольких (по числу транзитных узлов) локальных операций коммутации. Отправитель должен выставить данные на тот свой порт, из которого выходит найденный маршрут, а все транзитные узлы должны соответствующим образом выполнить «переброску» данных с одного своего порта на другой, другими словами, выполнить коммутацию. Устройство, функциональным назначением которого является выполнение коммутации, называется коммутатором (switch). Коммутатор производит коммутацию входящих в его порты информационных потоков, направляя их в соответствующие выходные порты (рис. 2.10).

Однако прежде чем выполнить коммутацию, коммутатор должен опознать поток. Для этого поступившие данные анализируются на предмет наличия в них признаков какого-либо из потоков, заданных в таблице коммутации. Если произошло совпадение, то эти данные направляются на интерфейс, который был определен для них в маршруте.

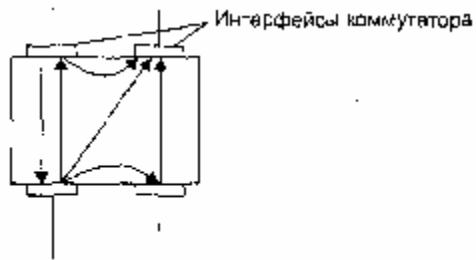


Рис. 2.10. Коммутатор

ПРИМЕЧАНИЕ

Термины коммутация, таблица коммутации и коммутатор в телекоммуникационных сетях могут трактоваться неоднозначно. Мы уже определили термин коммутация как процесс соединения абонентов сети через транзитные узлы. Этим же термином мы обозначаем и соединение интерфейсов в пределах отдельного транзитного узла. Коммутатором в широком смысле называется устройство любого типа, способное выполнять операции переключения потока данных с одного интерфейса на другой. Операция коммутации может быть выполнена и соответствии с различными правилами и алгоритмами. Некоторые способы коммутации и соответствующие им таблицы и устройства получили специальные названия (например, маршрутизация, таблица маршрутизации, маршрутизатор). В то же время за другими специальными типами коммутации и соответствующими устройствами закрепились те же самые названия: коммутация, таблица коммутации и коммутатор, которые здесь используются в узком смысле, например коммутация и коммутатор локальной сети. Для телефонных сетей, которые появились намного раньше компьютерных, также характерна аналогичная терминология — коммутатор является здесь синонимом телефонной станции. Из-за солидного возраста и гораздо большей (пока) распространенности телефонных сетей чаще всего в телекоммуникациях под термином «коммутатор» понимают именно телефонный коммутатор.

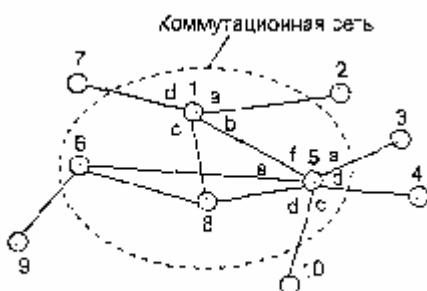


Рис. 2.11. Коммутационная сеть

Коммутатором может быть как специализированное устройство, так и универсальный компьютер со встроенным программным механизмом коммутации, в этом случае коммутатор называется программным. Компьютер может совмещать функции по коммутации данных, направляемых на другие узлы, с выполнением своих обычных функций как конечного узла. Однако во многих случаях более рациональным является решение, в соответствии с которым некоторые

узлы и сети выделяются специально для выполнения коммутации. Эти узлы образуют коммутационную сеть, к которой подключаются все остальные. На рис. 2.11 показана коммутационная сеть, образованная из узлов 1, 5, 6 и 8, к которой подключаются конечные узлы 2, 3, 4, 7, 9 и 10.

Мультиплексирование и демультиплексирование

Как было уже сказано, прежде чем выполнить переброску данных на определенные для них интерфейсы, коммутатор должен понять, к какому потоку они относятся. Эта задача должна решаться независимо от того, поступает ли на вход коммутатора только один поток в «чистом» виде, или «смешанный» поток, являющийся результатом агрегирования нескольких потоков. В последнем случае к задаче распознавания добавляется задача демультиплексирования — разделение суммарного агрегированного потока на несколько составляющих потоков. Как правило, операцию коммутации сопровождает также обратная операция — мультиплексирование (multiplexing), при которой из нескольких отдельных потоков образуется общий агрегированный поток, который можно передавать по одному физическому каналу связи. Операции мультиплексирования/демультиплексирования имеют такое же важное значение в любой сети, как и операции коммутации, потому что без них пришлось бы все коммутаторы связывать большим количеством параллельных каналов, что свело бы на нет все преимущества неполносвязной сети.

На рис. 2.12 показан фрагмент сети, состоящий из трех коммутаторов. Коммутатор 1 имеет пять сетевых интерфейсов. Рассмотрим, что происходит па интерфейсе Инт. 1. Сюда поступают данные с трех интерфейсов — Инт. 3, Инт. 4 и Инт. 5. Все их надо передать в общий физический канал, то есть выполнить операцию мультиплексирования. Мультиплексирование является способом обеспечения доступности имеющихся физических каналов одновременно для нескольких сеансов связи между абонентами сети.

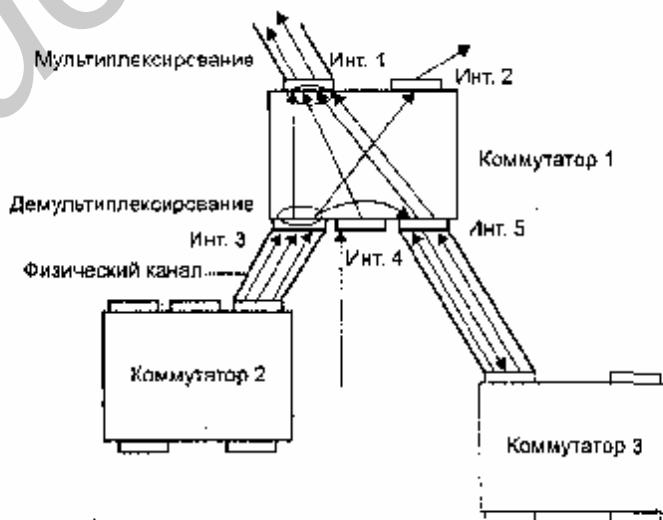


Рис. 2.12. Операции мультиплексирования и демультиплексирования потоков при коммутации

Существует множество способов мультиплексирования потоков в одном физическом канале, важнейшим из них является разделение времени. При этом способе каждый поток время от времени (с фиксированным или случайным периодом) получает физический канал в свое распоряжение и передает в это время по нему свои данные. Очень распространено также частотное разделение канала, когда каждый поток передаст данные в выделенном, ему частотном диапазоне. Технология мультиплексирования должна позволять получателю такого суммарного потока выполнять обратную операцию — разделение (демультиплексирование) данных на слагаемые потоки. На интерфейсе Инт. 3 коммутатор выполняет демультиплексирование потока на три составляющих подпотока. Один из них он передает на интерфейс Инт. 1, другой — на Инт. 2, а третий — на Инт. 5. А вот на интерфейсе Инт. 2 нет необходимости выполнять мультиплексирование или демультиплексирование — этот интерфейс выделен одному потоку в монопольное использование. Вообще на каждом интерфейсе могут одновременно выполняться обе задачи — мультиплексирования и демультиплексирования.

Частный случай коммутатора, у которого все входящие информационные потоки коммутируются на один выходной интерфейс, где мультиплексируются в один агрегированный поток и направляются в один физический канал, называется мультиплексором (multiplexer, mux) — такой коммутатор показан на рис. 2.13, а. Коммутатор, который имеет один входной интерфейс и несколько выходных, называется демультиплексором (рис. 2.13, б).

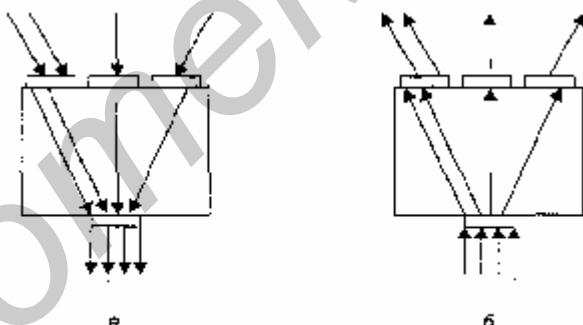


Рис. 2.13. Мультиплексор и демультиплексор

Разделяемая среда передачи данных

Еще одним параметром разделяемого канала связи является количество узлов, подключенных к такому каналу. В приведенных выше примерах к каналу связи подключались только два взаимодействующих узла, точнее - два интерфейса (рис. 2.14, а и б). В телекоммуникационных сетях используется и другой вид подключения, когда к одному каналу подключается несколько интерфейсов (рис. 2.14, б). Такое множественное подключение интерфейсов порождает уже рассматривавшуюся выше топологию «общая шина», иногда называемую также шлейфовым подключением. Во всех этих случаях возникает проблема согласованного использования канала.

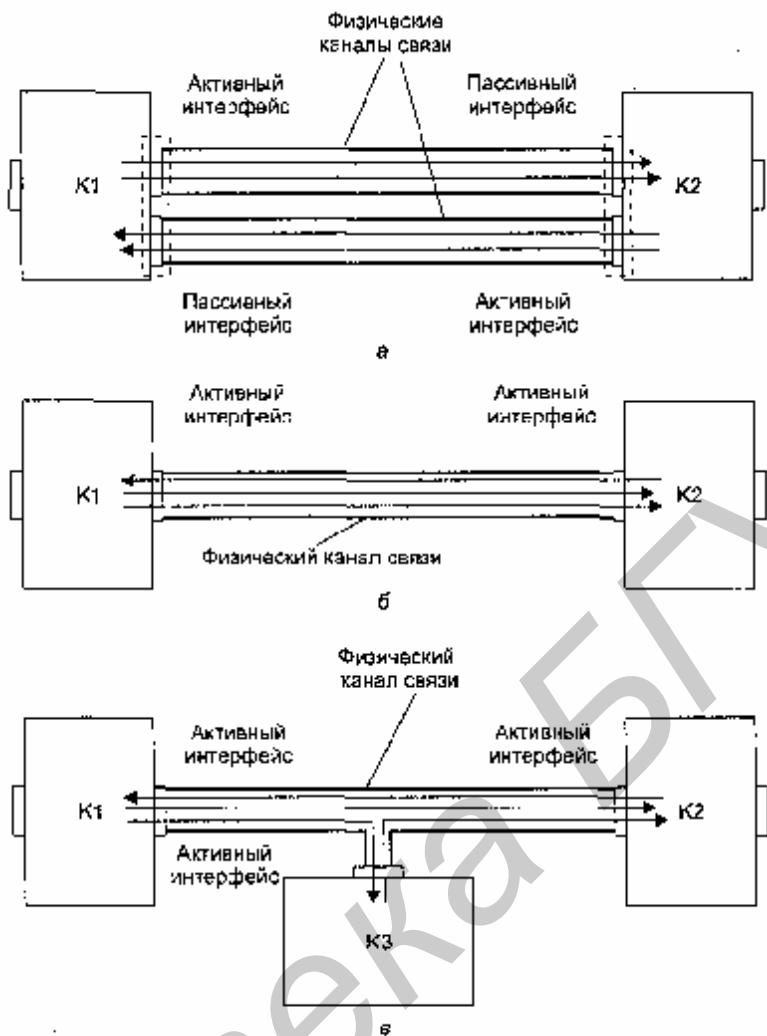


Рис. 2.14. Совместное использование канала связи

На рисунке показаны различные варианты разделения каналов связи между интерфейсами. На рис. 2.14, а коммутаторы K1 и K2 связаны двумя односторонними физическими каналами, то есть такими, по которым информация может передаваться только в одном направлении. В этом случае передающий интерфейс является активным и физическая среда передачи находится целиком и полностью под его управлением. Пассивный интерфейс только принимает данные. Проблема разделения канала между интерфейсами здесь отсутствует. Заметим, однако, что задача мультиплексирования потоков данных в канале при этом сохраняется. На практике два односторонних канала, реализующие в целом дуплексную связь между двумя устройствами, обычно считаются одним дуплексным каналом, а пара интерфейсов одного устройства — как передающая и принимающая части одного и того же интерфейса.

На рис. 2.14, в коммутаторы K1 и K2 связаны каналом, который может передавать данные в обе стороны, но только попарно. При этом возникает необходимость в механизме согласования доступа интерфейсов K1 и K2 к такому каналу. Обобщением этого варианта является случай, показанный на рис. 2.14, в, когда к каналу связи подключается несколько (больше двух) интерфейсов, образуя общую шину.

Совместно используемый несколькими интерфейсами физический канал называют разделяемым (shared)¹. Часто используется также термин «разделяемая среда передачи данных» (shared media). Разделяемые каналы связи используются не только для связей типа коммутатор-коммутатор, но и для связей компьютер-коммутатор и компьютер-компьютер.

¹ Нужно подчеркнуть, что термин «разделяемая среда» (shared media) традиционно относят именно к случаю разделения канала между интерфейсами и практически никогда — к случаю разделения канала между потоками.

Существуют различные способы решения задачи организации совместного доступа к разделяемым линиям связи. Одни из них используют централизованный подход, когда доступом управляет специальное устройство — арбитр, другие — децентрализованный. Внутри компьютера проблемы разделения линий связи между различными модулями также существуют — примером является доступ к системной шине, которым управляет либо процессор, либо специальный арбитр шины. В сетях организация совместного доступа к линиям связи имеет свою специфику из-за существенно большего времени распространения сигналов по линиям связи, поэтому процедуры согласования доступа к линии связи могут занимать слишком большой промежуток времени и приводить к значительным потерям производительности сети.

Несмотря на все эти сложности, в локальных сетях разделяемые среды используются очень часто. Этот подход, в частности, реализован в широко распространенных классических технологиях Ethernet, Token Ring, FDDI. В глобальных сетях разделяемые между интерфейсами среды практически не используются. Это объясняется тем, что большие временные задержки распространения сигналов вдоль протяженных каналов связи приводят к слишком длительным переговорным процедурам доступа к разделяемой среде, сокращая до неприемлемого уровня долю полезного использования канала связи на передачу данных абонентов.

Однако в последние годы наметилась тенденция отказа от разделяемых сред передачи данных и в локальных сетях. Это связано с тем, что за достигаемое таким образом удешевление сети приходится расплачиваться производительностью. Сеть с разделяемой средой при большом количестве узлов будет работать всегда медленнее, чем аналогичная сеть с индивидуальными линиями связи, так как пропускная способность индивидуальной линии связи достается одному компьютеру, а при ее совместном использовании — делится между всеми компьютерами сети. Часто с такой потерей производительности мирятся ради увеличения экономической эффективности сети. Не только в классических, но и в совсем новых технологиях, разработанных для локальных сетей, сохраняется режим разделяемых линий связи. Например, разработчики технологии Gigabit Ethernet, принятой в 1998 году в качестве нового стандарта, включили режим разделения передающей среды в свои спецификации наряду с режимом работы по индивидуальным линиям связи.

Коммутация каналов и пакетов

В общем случае решение каждой из частных задач коммутации — определение потоков и соответствующих маршрутов, фиксация маршрутов в конфигурационных параметрах и таблицах сетевых устройств, распознавание потоков и передача данных между интерфейсами одного устройства, мультиплексирование/демультиплексирование потоков и разделение среды передачи — зависит от решения остальных. Комплекс технических решений обобщенной задачи коммутации в своей совокупности составляет базис любой сетевой технологии. От того, какой механизм прокладки маршрутов, продвижения данных и совместного использования каналов связи заложен в тон или иной сетевой технологии, зависят ее фундаментальные свойства.

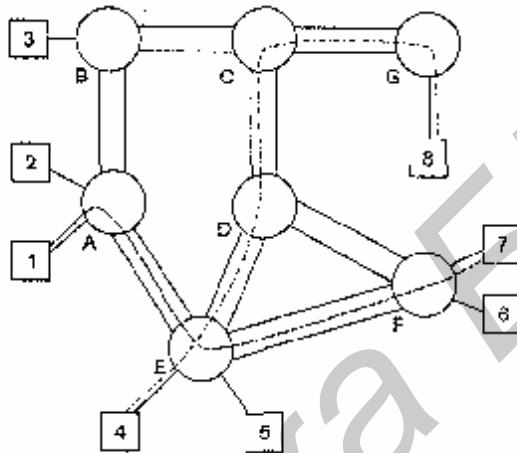


Рис. 2.15. Общая структура сети с коммутацией абонентов

Среди множества возможных подходов к решению задачи коммутации абонентов в сетях выделяют два основополагающих:

- коммутация каналов (circuit switching),
- коммутация пакетов (packet switching).

Внешне обе эти схемы соответствуют приведенной на рис. 2.15 структуре сети, однако возможности и свойства их различны.

Сети с коммутацией каналов имеют более богатую историю, они ведут свое происхождение от первых телефонных сетей. Сети с коммутацией пакетов сравнительно молоды, они появились в конце 60-х годов как результат экспериментов с первыми глобальными компьютерными сетями. Каждая из этих схем имеет свои преимущества и недостатки, но по долгосрочным прогнозам многих специалистов будущее принадлежит технологии коммутации пакетов, как более гибкой и универсальной.

Коммутация каналов

Коммутационная сеть в случае коммутации каналов образует между конечными узлами непрерывный составной физический канал из последовательно соединенных коммутаторами промежуточных канальных участков. Условием того, что несколько физических каналов при последовательном соединении образуют единый физический канал, является равенство скоростей передачи данных в

каждом из составляющих физических каналов. Равенство скоростей означает, что коммутаторы такой сети не должны буферизовать передаваемые данные.

В сети с коммутацией каналов перед передачей данных всегда необходимо выполнить процедуру установления соединения, в процессе которой и создается составной канал. И только после этого можно начинать передавать данные.

Например, если сеть, изображенная на рис. 2.15, работает по технологии коммутации каналов, то узел 1, чтобы передать данные узлу 7, прежде всего должен передать специальный запрос на установление соединения коммутатору А, указав адрес назначения 7. Коммутатор А должен выбрать маршрут образования составного канала, а затем передать запрос следующему коммутатору, в данном случае Е. Далее коммутатор Е передает запрос коммутатору F, а тот, в свою очередь, передает запрос узлу 7. Если узел 7 принимает запрос на установление соединения, он направляет по уже установленному каналу ответ исходному узлу, после чего составной канал считается скоммутированным и узлы 1 и 7 могут обмениваться по нему данными, например, вести телефонный разговор.

Техника коммутации каналов имеет свои достоинства и недостатки. Основные достоинства перечислены ниже.

- Постоянная и известная скорость передачи данных по установленному между конечными узлами каналу. Это позволяет пользователю сети на основе заранее произведенной оценки требуемой для качественной передачи данных пропускной способности установить в сети канал нужной скорости.
- Низкий и постоянный уровень задержки передачи данных через сеть. Это позволяет качественно передавать данные, чувствительные к задержкам (называемые также трафиком реального времени) — голос, видео, различную технологическую информацию.

К недостаткам сетей с коммутацией каналов относятся следующие.

- Отказ сети в обслуживании запроса на установление соединения. Такая ситуация может сложиться из-за того, что на некотором участке сети соединение нужно установить вдоль некоторого физического канала, через который уже проходит максимальное для данной техники мультиплексирования и для данного канала количество информационных потоков. Например, если между двумя телефонными станциями имеется физический канал, который допускает прохождение через него не более 30 соединений абонентов, и нужно установить через этот канал еще одно соединение, то произойдет отказ в обслуживании. Отказ может случиться и на конечном участке составного канала, например, если абонент может поддерживать только одно соединение, что характерно для многих телефонных сетей. При поступлении второго вызова к уже разговаривающему абоненту сеть в этом случае передаетзывающему абоненту короткие гудки — сигнал «занято».

- Нерациональное использование пропускной способности физических каналов. После установления соединения часть пропускной способности отводится составному каналу на все время соединения, то есть до тех пор, когда соединение не будет разорвано по инициативе абонентов или самой сети. В то же время во многих случаях абонентам не нужна пропускная способность канала на все

время соединения, например, в телефонном разговоре встречаются паузы, еще более неравномерным во времени является взаимодействие компьютеров. Невозможность динамического перераспределения пропускной способности физического канала является принципиальным ограничением сети с коммутацией пакетов, так как единицей коммутации здесь является информационный поток в целом. Модификация методом коммутации для снятия этого ограничения приводит к технике коммутации пакетов.

□ Обязательная задержка перед передачей данных из-за фазы установления соединения.

Достоинства и недостатки любой сетевой технологии — относительны. В определенных ситуациях на первый план выходят достоинства, а недостатки становятся несущественными. Так, техника коммутации каналом хорошо работает в тех случаях, когда нужно передавать только трафик телефонных разговоров, а с невозможностью «вырезать» паузы из разговора и более рационально использовать магистральные физические каналы между коммутаторами можно мириться. Однако при передаче очень неравномерного компьютерного трафика эта нерациональность уже выходит на первый план.

Коммутация пакетов

Техника коммутации пакетов была специально разработана для эффективной передачи компьютерного трафика. Эксперименты по созданию первых компьютерных сетей на основе техники коммутации каналов показали, что этот вид коммутации не позволяет достичь высокой общей пропускной способности сети. Типичные сетевые приложения генерируют трафик очень неравномерно, с высоким уровнем пульсации скорости передачи данных. Например, при обращении к удаленному файловому серверу пользователь сначала просматривает содержимое каталога этого сервера, что порождает передачу небольшого объема данных. Затем пользователь открывает требуемый файл в текстовом редакторе, и эта операция может создать достаточно интенсивный обмен данными, особенно если файл содержит объемные графические включения. После отображения нескольких страниц файла пользователь некоторое время работает с ними локально, что вообще не требует передачи данных по сети. Затем пользователь возвращает модифицированные копии страниц на сервер — и это снова порождает интенсивную передачу данных по сети.

Коэффициент пульсации трафика отдельного пользователя сети, равный отношению средней интенсивности обмена данными к максимально возможной, может достигать 1:50 или даже 1:100. Если для описанного сеанса организовать коммутацию канала между компьютером пользователя и сервером, то большую часть времени канал будет простаивать. В то же время коммутационные возможности, сети будут закреплены за данной парой абонентов и останутся недоступными другим пользователям сети.

При коммутации пакетов все передаваемые пользователем сети сообщения разбиваются в исходном узле на сравнительно небольшие части, называемые пакетами. Напомним, что сообщением называется логически завершенная порция

данных — запрос на передачу файла, ответ на этот запрос, содержащий весь файл и т. п. Сообщения могут иметь произвольную длину, от нескольких байт до многих мегабайт. Пакеты обычно тоже могут иметь переменную длину, но в узких пределах, например от 46 до 1500 байт. Каждый пакет снабжается заголовком, в котором указывается адресная информация, необходимая для доставки пакета узлу назначения, а также номер пакета, который будет использоваться узлом назначения для сборки сообщения (рис. 2.16). Пакеты транспортируются в сети как независимые информационные блоки. Коммутаторы сети принимают пакеты от конечных узлов и на основании адресной информации передают их друг другу, а в конечном итоге — узлу назначения.



Рис. 2.16. Разбиение сообщения на пакеты

Коммутаторы пакетной сети отличаются от коммутаторов каналов тем, что имеют внутреннюю буферную память для временного хранения пакетов, когда выходной порт коммутатора в момент принятия пакета занят передачей другого пакета (рис. 2.17). В этом случае пакет находится некоторое время в очереди пакетов в буферной памяти выходного порта, а когда до него дойдет очередь, то он передается следующему коммутатору. Такая схема передачи данных позволяет сглаживать пульсации трафика на магистральных связях между коммутаторами и тем самым использовать их наиболее эффективным образом для повышения пропускной способности сети в целом.

Конечно, для пары абонентов наиболее эффективным было бы предоставление им в единоличное пользование скоммутированного канала связи, как это делается в сетях с коммутацией каналов. При этом способе время взаимодействия этой пары абонентов было бы минимальным, так как данные без задержек передавались бы от одного абонента другому. Простота канала во время пауз передачи абонентов не интересует, для них важно быстрее решить свою собственную задачу. Сеть с коммутацией пакетов замедляет процесс взаимодействия конкретной пары абонентов, так как их пакеты могут ожидать в коммутаторах, пока по магистральным связям передаются другие пакеты, пришедшие в коммутатор ранее.

Тем не менее, общий объем передаваемых сетью компьютерных данных в единицу времени при коммутации пакетов выше, чем при коммутации каналов. Это происходит потому, что пульсации отдельных абонентов в соответствии с законом больших чисел распределяются во времени так, что их пики не совпадают. Поэтому коммутаторы постоянно и достаточно равномерно загружены работой, если число обслуживаемых ими абонентов действительно велико. На

рис. 2.17 показано, что трафик, поступающий от конечных узлов на коммутаторы, очень неравномерно распределен во времени. Однако коммутаторы более высокого уровня иерархии, которые обслуживают соединения между коммутаторами нижнего уровня, загружены более равномерно, и поток пакетов в магистральных каналах, соединяющих коммутаторы верхнего уровня, имеет почти максимальный коэффициент использования. Буферизация сглаживает пульсации, поэтому коэффициент пульсации на магистральных каналах гораздо ниже, чем на каналах абонентского доступа — он может быть равным 1:10 или далее 1:2.



Рис. 2.17. Сглаживание пульсаций трафика в сети с коммутацией пакетов

Более высокая эффективность сетей с коммутацией пакетов по сравнению с сетями с коммутацией каналов (при равной пропускной способности каналов связи) была доказана в 60-е годы как экспериментально, так и путем имитационного моделирования. Здесь уместна аналогия с мультипрограммными операционными системами. Каждая отдельная программа в такой системе выполняется дольше, чем в однопрограммной системе, когда программе выделяется все процессорное время, пока она не завершит свое выполнение. Однако общее число программ, выполняемых за единицу времени, в мультипрограммной системе больше, чем в однопрограммной.

Ниже перечислены достоинства сетей с коммутацией пакетов.

- Высокая общая пропускная способность сети при передаче пульсирующего трафика.
- Возможность динамически перераспределять пропускную способность физических каналов связи между абонентами в соответствии с реальными потребностями их трафика.

К недостаткам сетей с коммутацией пакетов относятся следующие:

- Неопределенность скорости передачи данных между абонентами сети, обусловленная зависимостью задержек в очередях буферов коммутаторов сети от общей загрузки сети.
- Переменная величина задержки пакетов данных, которые могут достигать значительных величин в моменты мгновенных перегрузок сети.
- Возможные потери данных из-за переполнения буферов.

В настоящее время активно разрабатываются и внедряются методы, позволяю-

щие преодолеть указанные недостатки, которые особенно остро проявляются для чувствительного к задержкам трафика, требующего при этом постоянной скорости передачи. Эти методы, называемые методами обеспечения качества обслуживания (Quality of Service, QoS), рассматриваются в главе 15.

Сети с коммутацией пакетов, в которых реализованы методы обеспечения качества обслуживания, позволяют одновременно передавать различные виды трафика, в том числе такие важные, как телефонный и компьютерный. Поэтому методы коммутации пакетов сегодня считаются наиболее перспективными для построения конвергентной сети, которая обеспечит комплексные качественные услуги для абонентов любого типа. Тем не менее, и методы коммутации каналов нельзя списывать со счетов. Сегодня они с успехом работают не только в традиционных телефонных сетях, но и, широко применяются для образования высокоскоростных постоянных соединений в так называемых первичных (опорных) сетях технологий SDH и DWDM, которые активно используются для создания магистральных физических каналов между коммутаторами телефонных или компьютерных сетей. В будущем вполне возможно появление новых технологий коммутации, в том или ином виде комбинирующих принципы коммутации пакетов и каналов.

Коммутация сообщений

Коммутация сообщений по своим принципам близка к коммутации пакетов. Под коммутацией сообщений понимается передача единого блока данных между транзитными компьютерами сети с временной буферизацией этого блока на диске каждого компьютера (рис. 2.18). Сообщение в отличие от пакета имеет произвольную длину, которая определяется не технологическими соображениями, а содержанием информации, составляющей сообщение. Например, сообщением может быть текстовый документ, файл с кодом программы, электронное письмо.

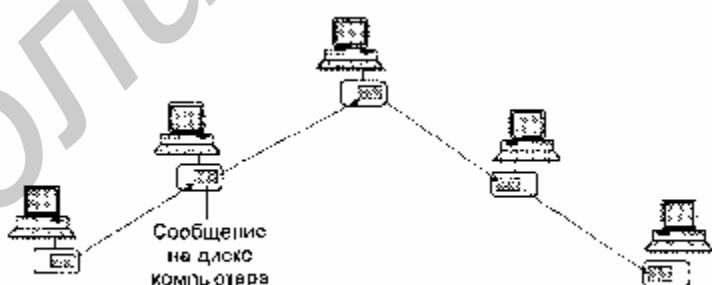


Рис. 2.18. Коммутация сообщений

Транзитные компьютеры могут соединяться между собой как сетью с коммутацией пакетов, так и сетью с коммутацией каналов. Сообщение хранится в транзитном компьютере на диске, причем время хранения может быть достаточно большим, если компьютер загружен другими работами или сеть временно перегружена.

По такой схеме обычно передаются сообщения, не требующие немедленного ответа, чаще всего сообщения электронной почты. Режим передачи с промежуточным хранением на диске называется режимом хранения и передачи (store-and-forward).

Режим коммутации сообщений разгружает сеть для передачи трафика, требующего быстрого ответа, например трафика службы WWW или файловой службы.

Количество транзитных компьютеров стараются по возможности уменьшить. Если компьютеры подключены к сети с коммутацией пакетов, то число промежуточных компьютеров обычно уменьшается до двух. Например, когда пользователь передает почтовое сообщение своему серверу исходящей почты, тот сразу старается передать сообщение серверу входящей почты адресата. Но если компьютеры связаны между собой телефонной сетью, то часто используется несколько промежуточных серверов, так как прямой доступ к конечному серверу может быть невозможен в данный момент из-за перегрузки телефонной сети (абонент занят) или экономически невыгоден из-за высоких тарифов на дальнюю телефонную связь.

Техника коммутации сообщений появилась в компьютерных сетях раньше техники коммутации пакетов, потоком была вытеснена последней, как более эффективной по критерию пропускной способности сети. Запись сообщения на диск занимает достаточно много времени, кроме того, наличие дисков предполагает использование в качестве коммутаторов специализированных компьютеров, что удорожает сеть.

Сегодня коммутация сообщений работает только для некоторых не оперативных служб, причем чаще всего поверх сети с коммутацией пакетов, как служба прикладного уровня.

Постоянная и динамическая коммутация

Как сети с коммутацией пакетов, так и сети с коммутацией каналов можно разделить на два класса по другому признаку — на сети с динамической коммутацией и сети с постоянной коммутацией,

В первом случае сеть разрешает устанавливать соединение по инициативе пользователя сети. Коммутация выполняется на время сеанса связи, а затем (опять же по инициативе одного из взаимодействующих пользователей) связь разрывается. В общем случае любой пользователь сети может соединиться с любым другим пользователем сети. Обычно период соединения между парой пользователей при динамической коммутации составляет от нескольких секунд до нескольких часов и завершается при выполнении определенной работы — передачи файла, просмотра страницы текста или изображения и т. п.

Во втором случае сеть не предоставляет пользователю возможность выполнить динамическую коммутацию с другим произвольным пользователем сети. Вместо этого сеть разрешает паре пользователей заказать соединение на длительный период времени. Соединение устанавливается не пользователями, а персоналом, обслуживающим сеть. Время, на которое устанавливается постоянная

коммутация, измеряется обычно несколькими месяцами. Режим постоянной (permanent) коммутации в сетях с коммутацией каналов часто называется сервисом выделенных (dedicated), или арендуемых (leased), каналов. В том случае, когда постоянное соединение через сеть коммутаторов устанавливается с помощью автоматических процедур, инициированных обслуживающим персоналом, его часто называют полупостоянным (semi-permanent) соединением, в отличие от режима ручного конфигурирования каждого коммутатора.

Примерами сетей, поддерживающих режим динамической коммутации, являются телефонные сети общего пользования, локальные сети, сети TCP/IP.

Наиболее популярными сетями, работающими в режиме постоянной коммутации, сегодня являются сети технологии SDH, на основе которых строятся выделенные каналы связи с пропускной способностью в несколько гигабит в секунду.

Некоторые типы сетей поддерживают оба режима работы. Например, сети X.25 и ATM могут предоставлять пользователю возможность динамически связаться с любым другим пользователем сети и в то же время отправлять данные по постоянному соединению одному вполне определенному абоненту.

Пропускная способность сетей с коммутацией пакетов

Одним из отличий метода коммутации пакетов от метода коммутации каналов является неопределенность пропускной способности соединения между двумя абонентами. В методе коммутации каналов после образования составного канала пропускная способность сети при передаче данных между конечными узлами известна — это пропускная способность канала. Данные после задержки, связанной с установлением канала, начинают передаваться на максимальной для канала скорости (рис. 2.19, а). Время передачи сообщения в сети с коммутацией каналов $T_{к.к}$ равно сумме задержки распространения сигнала по линии связи $t_{з.р.}$ и задержки передачи сообщения $t_{з.п.}$. Задержка распространения сигнала зависит от скорости распространения электромагнитных волн в конкретной физической среде, которая колеблется от 0,6 до 0,9 скорости света в вакуме. Время передачи сообщения равно V/C , где V — объем сообщения в битах, а C — пропускная способность канала в битах в секунду.

В сети с коммутацией пакетов наблюдается принципиально другая картина.

Процедура установления соединения в этих сетях, если она используется, занимает примерно такое же время, как и в сетях с коммутацией каналов, поэтому будем сравнивать только время передачи данных.

На рис. 2.19, б показан пример передачи в сети с коммутацией пакетов. Предполагается, что в сеть передается сообщение того же объема, что и сообщение, иллюстрируемое рис. 2.19, а, однако оно разделено на пакеты, каждый из которых снабжен заголовком. Время передачи сообщения в сети с коммутацией пакетов обозначено на рисунке $T_{к.н.}$. При передаче этого сообщения, разбитого на пакеты, по сети с коммутацией пакетов возникают дополнительные временные задержки.

Во-первых, это задержки в источнике передачи, который, помимо передачи

собственно сообщения, тратит дополнительное время на передачу заголовков $t_{\text{п.з.}}$, плюс к этому добавляются задержки $t_{\text{инт.}}$, вызванные интервалами между передачей каждого следующего пакета (это время уходит на формирование очередного пакета стеком протоколов).

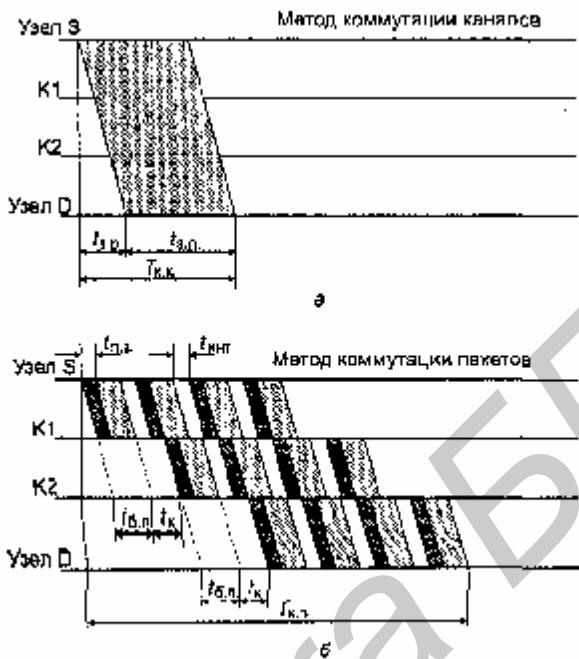


Рис. 2.19. Задержки передачи данных в сетях с коммутацией каналов и пакетов

Во-вторых, дополнительное время тратится в каждом коммутаторе. Здесь задержки складываются из времени буферизации пакета $t_{\text{б.п.}}$ (коммутатор не может начать передачу пакета, не приняв его полностью и свой буфер) и времени коммутации $t_{\text{к.}}$ Бремя буферизации равно времени приема пакета с битовой скоростью протокола. Время коммутации складывается из времени ожидания пакета в очереди и времени перемещения пакета в выходной порт. Если время перемещения пакета фиксировано и обычно невелико (от нескольких микросекунд до нескольких десятков микросекунд), то время ожидания пакета в очереди колеблется в очень широких пределах и заранее неизвестно, так как зависит от текущей загрузки сети пакетами.

Проведем грубую оценку задержки в передаче данных в сетях с коммутацией пакетов по сравнению с сетями с коммутацией каналов на простейшем примере. Пусть тестовое сообщение, которое нужно передать в обоих видах сетей, составляет 200 Кбайт. Отправитель находится от получателя на расстоянии 5000 км. Пропускная способность линий связи составляет 2 Мбит/с. Время передачи данных по сети с коммутацией каналов складывается из времени распространения сигнала, которое для расстояния 5000 км можно оценить примерно в 25 мс, и времени передачи сообщения, которое при пропускной способности 2 Мбит/с и размере сообщения 200 Кбайт равно примерно 800 мс, то есть всего передача данных занимает 825 мс.

Оценим дополнительное время, которое потребуется для передачи этого сообщения по сети с коммутацией пакетов. Будем считать, что путь от отправителя

до получателя пролегает через 10 коммутаторов. Исходное сообщение разбивается на пакеты в 1 Кбайт, всего 200 пакетов. Вначале оценим задержку, которая возникает в исходном узле. Предположим, что доля служебной информации, размещенной в заголовках пакетов, по отношению к общему объему сообщения составляет 10 %. Следовательно, дополнительная задержка, связанная с передачей заголовков пакетов, составляет 10 % от времени передачи целого сообщения, то есть 80 мс. Если принять интервал между отправкой пакетов равным 1 мс, тогда дополнительные потери за счет интервалов составят 200 мс. Итого, в исходном узле из-за пакетирования сообщения при передаче возникает дополнительная задержка в 280 мс.

Каждый из 10 коммутаторов вносит задержку коммутации, которая может иметь большой разброс, от долей до тысяч миллисекунд. В данном примере примем, что на коммутацию в среднем тратится 20 мс. Кроме того, при прохождении сообщений через коммутатор возникает задержка буферизации пакета. Эта задержка при величине пакета 1 Кбайт и пропускной способности линии 2 Мбит/с равна 4 мс. Общая задержка, вносимая 10 коммутаторами, составит примерно 240 мс. В результате дополнительная задержка, созданная сетью с коммутацией пакетов, составляет 520 мс. Учитывая, что вся передача данных в сети с коммутацией каналов занимает 825 мс, эту дополнительную задержку можно считать существенной.

Хотя приведенный расчет носит очень приблизительный характер, но он делает более понятными те причины, которые приводят к тому, что процесс передачи для определенной пары абонентов в сети с коммутацией пакетов является более медленным, чем в сети с коммутацией каналов.

Неопределенная пропускная способность сети с коммутацией пакетов — это плата за ее общую эффективность при некотором ущемлении интересов отдельных абонентов. Аналогично, в мультипрограммной операционной системе время выполнения приложения предсказать заранее невозможно, так как оно зависит от количества других приложений, с которыми делит процессор данное приложение.

На эффективность работы сети существенно влияют размеры пакетов, которые передает сеть. Слишком большие пакеты приближают сеть с коммутацией пакетов к сети с коммутацией каналов, поэтому эффективность сети при этом падает. Слишком маленькие пакеты заметно увеличивают долю служебной информации, так как каждый пакет несет с собой заголовок фиксированной длины, а количество пакетов, на которые разбиваются сообщения, при уменьшении размера пакета резко возрастает. Существует некоторая золотая середина, которая обеспечивает максимальную эффективность работы сети, однако ее трудно определить точно, так как она зависит от многих факторов, некоторые из них к тому же постоянно меняются в процессе работы сети. Поэтому разработчики протоколов для сетей с коммутацией пакетов выбирают пределы, в которых может находиться длина пакета, а точнее его поле данных, так как заголовок, как правило, имеет фиксированную длину. Обычно нижний предел поля данных выбирается равным нулю, что разрешает передавать служебные пакеты без пользовательских данных, а верхний предел не превышает значения 4 Кбайт.

Приложения при передаче данных пытаются занять максимальный размер поля данных, чтобы быстрее выполнить обмен данными, а небольшие пакеты обычно используются для квитанций о доставке пакета.

При выборе размера пакета необходимо учитывать также и интенсивность битовых ошибок канала. На ненадежных каналах необходимо уменьшать размеры пакетов, так как это уменьшает объем повторно передаваемых данных при искаражениях пакетов.

Ethernet — пример стандартной технологии коммутации пакетов

Рассмотрим, каким образом описанные выше общие подходы к решению наиболее важных проблем построения сетей воплощены в наиболее популярной сетевой технологии — Ethernet.

Сетевая технология — это согласованный набор стандартных протоколов и реализующих их программно-аппаратных средств (например, сетевых адаптеров, драйверов, кабелей и разъемов), достаточный для построения вычислительной сети. Эпитет «достаточный» подчеркивает то обстоятельство, что этот набор представляет собой минимальный набор средств, с помощью которых можно построить работоспособную сеть. Возможно, эту сеть можно улучшить, например, за счет выделения в ней подсетей, что сразу потребует кроме протоколов стандарта Ethernet применения протокола IP, а также специальных коммуникационных устройств — маршрутизаторов. Улучшенная сеть будет, скорее всего, более надежной и быстродействующей, но за счет надстроек над средствами технологии Ethernet, которая составляет базис сети.

Термин «сетевая технология» чаще всего используется в описанном выше узком смысле, но иногда применяется и его расширенное толкование как любого набора средств и правил для построения сети, например «технология сквозной маршрутизации», «технология создания защищенного канала», «технология IP-сетей».

Протоколы, на основе которых строится сеть определенной технологии (в узком смысле), специально разрабатывались для совместной работы, поэтому от разработчика сети не требуется дополнительных усилий по организации их взаимодействия. Иногда сетевые технологии называют базовыми технологиями, имея в виду то, что на их основе строится базис любой сети. Примерами базовых сетевых технологий могут служить наряду с Ethernet такие известные технологии локальных сетей, как Token Ring и FDDI, или же технологии территориальных сетей X.25 и frame relay. Для получения работоспособной сети в этом случае достаточно приобрести программные и аппаратные средства, относящиеся к одной базовой технологии — сетевые адAPTERы с драйверами, концентраторы, коммутаторы, кабельную систему и т. п., — и соединить их в соответствии с требованиями стандарта на данную технологию.

Основной принцип, положенный в основу Ethernet, — случайный метод доступа к разделяемой среде передачи данных. В качестве такой среды может использоваться толстый или тонкий коаксиальный кабель, витая пара, оптоволокно или радиоволны (кстати, первой сетью, построенной на принципе случайно-

го доступа к разделяемой среде, была радиосеть Aloha Гавайского университета).

В стандарте Ethernet строго зафиксирована топология электрических связей. Компьютеры подключаются к разделяемой среде в соответствии с типовой структурой «общая шина» (рис. 2.20). С помощью разделяемой во времени шины любые два компьютера могут обмениваться данными. Управление доступом к линии связи осуществляется специальными контроллерами — сетевыми адаптерами Ethernet. Каждый компьютер, а более точно, каждый сетевой адаптер, имеет уникальный адрес. Передача данных происходит со скоростью 10 Мбит/с. Эта величина является пропускной способностью сети Ethernet.

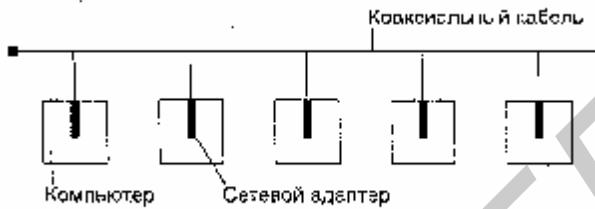


Рис. 2.20. Сеть Ethernet

Суть случайного метода доступа состоит в следующем. Компьютер в сети Ethernet может передавать данные по сети, только если сеть свободна, то есть если никакой другой компьютер в данный момент не занимается обменом. Поэтому важной частью технологии Ethernet является процедура определения доступности среды.

После того как компьютер убеждается, что сеть свободна, он начинает передачу, при этом «захватывает» среду. Время монопольного использования разделяемой среды одним узлом ограничивается временем передачи одного кадра. Кадр — это единица данных, которыми обмениваются компьютеры в сети Ethernet. Кадр имеет фиксированный формат и наряду с полем данных содержит различную служебную информацию, например адрес получателя и адрес отправителя.

Сеть Ethernet устроена так, что при попадании кадра в разделяемую среду передачи данных все сетевые адAPTERы одновременно начинают принимать этот кадр. Все они анализируют адрес назначения, располагающийся в одном из начальных полей кадра, и, если этот адрес совпадает с их собственным адресом, кадр помещается во внутренний буфер сетевого адаптера. Таким образом компьютер-адресат получает предназначенные ему данные.

Иногда может возникать ситуация, когда одновременно два или более компьютера решают, что сеть свободна, и начинают передавать информацию. Такая ситуация, называемая коллизией, препятствует правильной передаче данных по сети. В стандарте Ethernet предусмотрен алгоритм обнаружения и корректной обработки коллизий. Вероятность возникновения коллизии зависит от интенсивности сетевого трафика.

После обнаружения коллизии сетевые адAPTERы, которые пытались передать свои кадры, прекращают передачу и после паузы случайной длительности пы-

таются снова получить доступ к среде и передать тот кадр, который вызвал коллизию.

Главным достоинством сетей Ethernet, благодаря которому они стали такими популярными, является их экономичность. Для построения сети достаточно иметь по одному сетевому адаптеру для каждого компьютера плюс один физический сегмент коаксиального кабеля нужной длины. Другие базовые технологии, например Token Ring, для создания даже небольшой сети требуют наличия дополнительного устройства — концентратора.

Кроме того, в сетях Ethernet реализованы достаточно простые алгоритмы доступа к среде, адресации и передачи данных. Простая логика работы сети ведет к упрощению и, соответственно, удешевлению сетевых адаптеров и их драйверов. По той же причине адаптеры сети Ethernet обладают высокой надежностью.

И, наконец, еще одним замечательным свойством сетей Ethernet является их хорошая расширяемость, то есть легкость подключения новых узлов.

Другие базовые сетевые технологии — Token Ring, FDDI, — хотя и обладают многими индивидуальными чертами, в то же время имеют много общих свойств с Ethernet. В первую очередь — это применение регулярных фиксированных топологий (иерархическая звезда и кольцо), а также разделяемых сред передачи данных. Существенные отличия одной технологии от другой связаны с особенностями используемого метода доступа к разделяемой среде. Так, отличия технологии Ethernet от технологии Token Ring во многом определяются спецификой заложенных в них методов разделения среды — случайного алгоритма доступа в Ethernet и метода доступа путем передачи маркера в Token Ring.

Дейтаграммная передача и виртуальные каналы в сетях с коммутацией пакетов

В сетях с коммутацией пакетов сегодня применяется два класса механизмов продвижения пакетов — дейтаграммная передача и виртуальные каналы. Примерами сетей, реализующих дейтаграммный механизм продвижения, являются сети Ethernet, IP и IPX. С помощью виртуальных каналов передают данные сети X.25, frame relay и ATM. В данном разделе мы рассмотрим базовые принципы каждого из подходов.

Дейтаграммный способ передачи данных основан на том, что все передаваемые пакеты обрабатываются независимо друг от друга, пакет за пакетом. Выбор следующего узла — например, коммутатора Ethernet или маршрутизатора IP/IPX — происходит только на основании адреса узла назначения, содержащегося в заголовке пакета. Принадлежность пакета к определенному потоку между двумя конечными узлами и двумя приложениями, работающими на этих узлах, никак не учитывается.

Решение о том, какому узлу передать пришедший пакет, принимается на основе таблицы, содержащей набор адресов назначения и адресную информацию, од-

назначено определяющую следующий (транзитный или конечный) узел. Такие таблицы имеют разные названия — например, для сетей Ethernet они обычно называются таблицей продвижения (forwarding table), а для сетевых протоколов, таких как IP и IPX, — таблицей маршрутизации (routing table). Далее для простоты будем пользоваться термином «таблица маршрутизации» в качестве обобщенного названия таблиц такого рода, используемых для дейтаграммной передачи на основании только адреса назначения конечного узла.

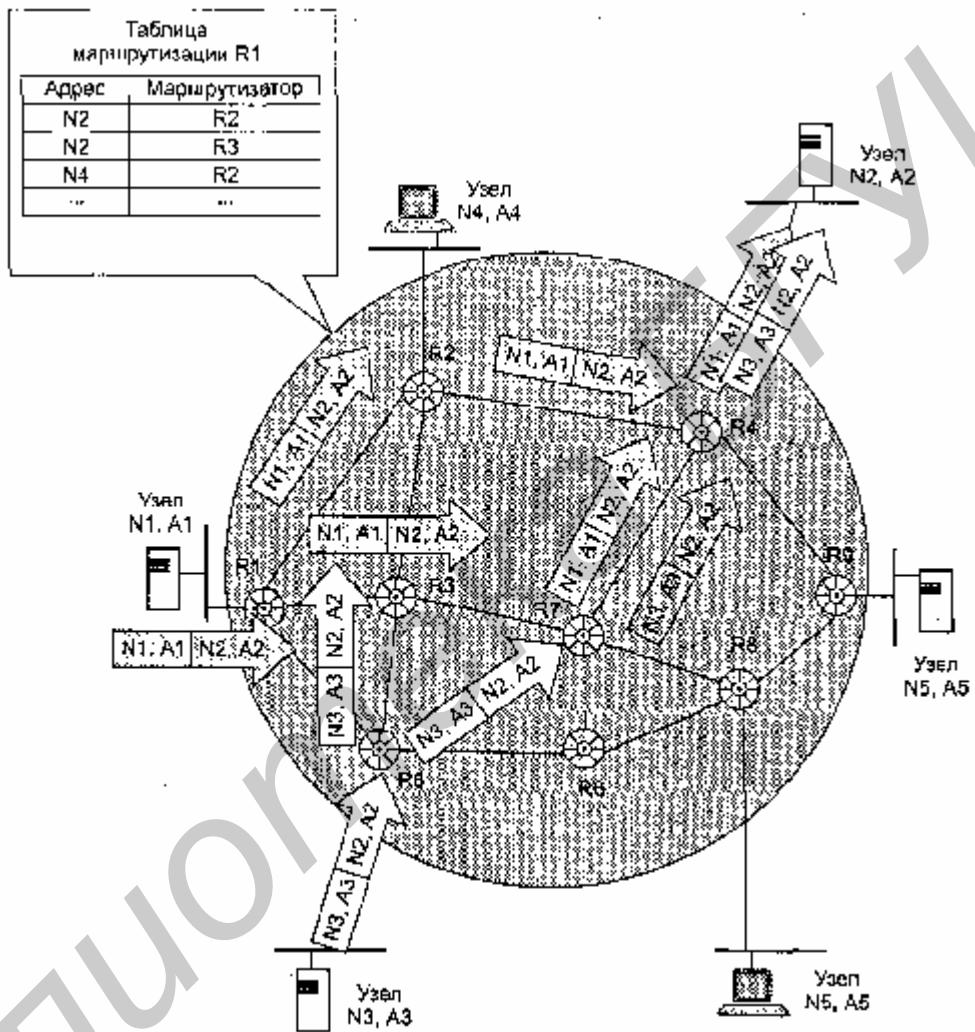


Рис. 2.21. Дейтаграммный принцип передачи пакетов

В таблице маршрутизации для одного и того же адреса назначения может содержаться несколько записей, указывающих соответственно на различные адреса следующего маршрутизатора. Такой подход используется для повышения производительности и надежности сети. В примере, показанном на рис. 2.21, пакеты, поступающие в маршрутизатор R1 для узла назначения с адресом N2, A2 в целях баланса нагрузки, распределяются между двумя следующими маршрутизаторами — R2 и R3, что снижает нагрузку на каждый из них, а значит, уменьшает очереди и ускоряет доставку. Некоторая «размытость» путей следования пакетов с одним и тем же адресом назначения через сеть является прямым следствием принципа независимой обработки каждого пакета, присущего

дейтаграммным протоколам. Пакеты, следующие по одному и тому же адресу назначения, могут добираться до него разными путями и вследствие изменения состояния сети, например отказа промежуточных маршрутизаторов.

Такая особенность дейтаграммного механизма, как размытость путей следования трафика через сеть, также в некоторых случаях является недостатком, например, если пакетам определенного сеанса между двумя конечными узлами сети необходимо обеспечить заданное качество обслуживания. Современные методы поддержки QoS работают эффективней, когда трафик, которому нужно обеспечить гарантии обслуживания, всегда проходит через одни и те же промежуточные узлы.

Механизм виртуальных каналов (virtual circuit, или virtual channel) создает в сети устойчивые пути следования трафика через сеть с коммутацией пакетов. Этот механизм учитывает существование в сети потоков данных.

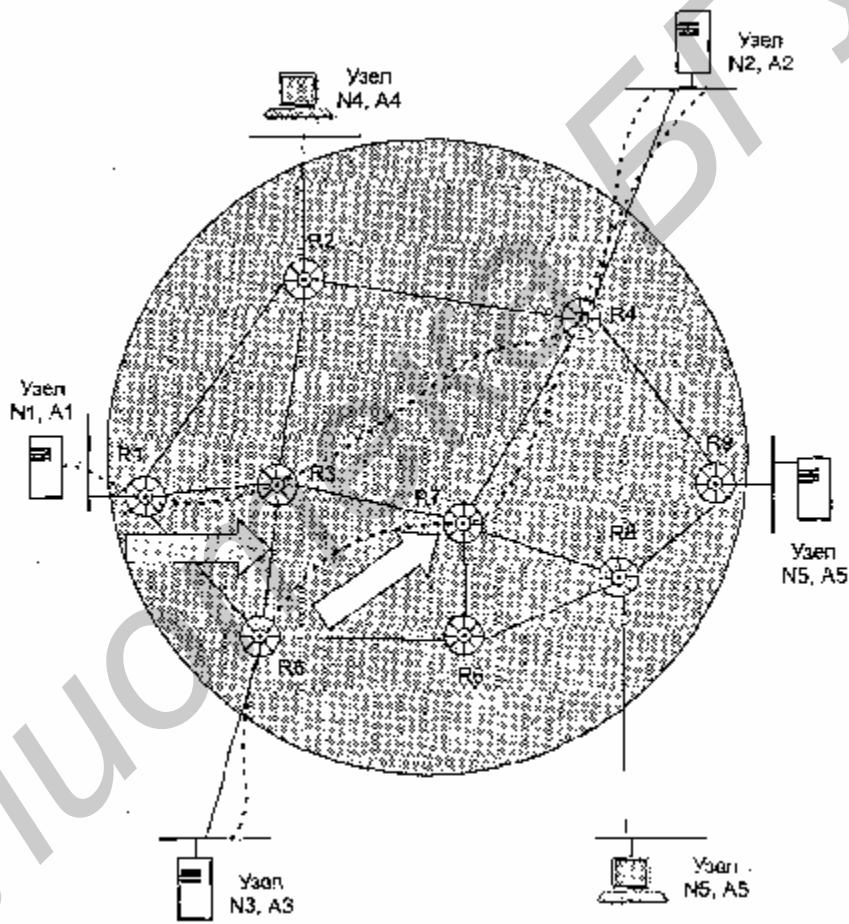


Рис. 2.22. Принцип работы виртуального канала

Если целью является прокладка для всех пакетов потока единого пути через сеть, то необходимым (но не всегда единственным) признаком такого потока должно быть наличие для всех его пакетов общих точек входа в сеть и выхода из сети. Именно для передачи таких потоков в сети создаются виртуальные каналы. На рис. 2.22 показан фрагмент сети, в которой проложены два виртуальных каналов. Первый проходит от конечного узла с адресом N1, A1 до конечного узла с адресом N2, A2 через промежуточные коммутаторы сети R1, R3 и R4.

Второй обеспечивает продвижение данных по пути N3, A3-R5-R7-R4-N2, A2. Между двумя конечными узлами может быть проложено несколько виртуальных каналов, как полностью совпадающих в отношении пути следования через транзитные узлы, так и отличающихся.

Сеть только обеспечивает возможность передачи трафика вдоль виртуального канала, решение же о том, какие именно потоки будут передаваться по этим каналам, принимают сами конечные узлы. Узел может использовать один и тот же виртуальный канал для передачи всех потоков, которые имеют общие с данным виртуальным каналом конечные точки, или же только части из них. Например, для потока реального времени можно использовать один виртуальный канал, а для трафика электронной почты - другой. В последнем случае разные виртуальные каналы будут предъявлять разные требования к качеству обслуживания, и удовлетворить их потенциально будет проще, чем в том случае, когда по одному виртуальному каналу передается трафик с разными требованиями к параметрам QoS.

Важной особенностью сетей с виртуальными каналами является использование локальных адресов пакетов при принятии решения о продвижении. Вместо достаточно длинного адреса узла назначения (его длина должна позволять уникально идентифицировать все узлы и подсети в сети, так как, технология ATM, например, оперирует с адресами длиной в 20 байт) применяется локальная, то есть меняющаяся от узла к узлу метка, которой помечаются вес пакеты, перемещаемые по определенному виртуальному каналу. Эта метка в различных технологиях называется по-разному: номер логического канала (Logical Channel number, LCN) в технологии X.25, идентификатор соединения уровня канала данных (Data Link Connection Identifier, DLCI) в технологии frame relay, идентификатор виртуального канала (Virtual Channel Identifier, VCI) в технологии ATM. Однако назначение ее везде одинаково — промежуточный узел, называемый в этих технологиях коммутатором, читает значение метки из заголовка пришедшего пакета и просматривает свою таблицу коммутации, в которой указывается, на какой выходной порт нужно передать пришедший пакет. Таблица коммутации содержит записи только о проходящих через данный коммутатор виртуальных каналах, а не обо всех имеющихся в сети узлах (или подсетях, если применяется иерархический способ адресации). Обычно в крупной сети количество проложенных через узел виртуальных каналов существенно меньше количества узлов и подсетей, поэтому размер таблицы коммутации намного меньше таблицы маршрутизации, а, следовательно, просмотр ее занимает гораздо меньше времени и не требует от коммутатора большой вычислительной мощности.

Идентификатор виртуального канала (именно такое название метки будет использоваться далее) также намного короче адреса конечного узла — по той же причине, - поэтому и избыточность заголовка пакета, который теперь не содержит длинного адреса, а переносит по сети только идентификатор, существенно меньше.

Выводы

- При связывании более двух компьютеров появляются новые проблемы; выбор топологии и схемы адресации, организация совместного использования связей и построение механизма коммутации.
- Выбор топологии определяет многие характеристики сети. Наиболее распространены следующие типы конфигураций; полносвязная, ячеистая, кольцевая, звезда и иерархическая звезда (дерево). Конфигурация общая шина является частным случаем конфигурации звезда. Здесь в качестве центрального элемента выступает пассивный кабель, к которому по схеме «монтажного ИЛИ» подключается несколько компьютеров.
- Адреса могут использоваться для идентификации отдельных интерфейсов, их групп (групповые адреса), а также сразу всех сетевых интерфейсов сети (широковещательные адреса).
- Адреса могут быть числовыми и символьными, аппаратными и сетевыми, плоскими и иерархическими. Для преобразования адресов из одного вида в другой используются специальные вспомогательные протоколы, которые называют иногда протоколами разрешения адресов (address resolution).
- В неполносвязных сетях соединение абонентов осуществляется путем коммутации, то есть связывании через сеть транзитных узлов. При этом должны быть решены следующие задачи: определение информационных потоков и маршрутов для них, сообщение о найденных маршрутах узлам сети, продвижение данных в каждом транзитном узле — распознавание потоков на входе и переключение их на соответствующий выход, а также мультиплексирование и демультиплексирование потоков.
- Устройство, функциональным назначением которого является выполнение коммутации, называется коммутатором (switch). Коммутатором может быть как специализированное устройство, так и универсальный компьютер со встроенным программным механизмом коммутации.
- Информационным потоком, или потоком данных (data flow, data stream), называют непрерывную последовательность байтов (или более крупных единиц данных — пакетов, кадров, ячеек), объединенных набором общих признаков, который выделяет его из общего сетевого трафика. При коммутации в качестве признаков могут выступать адреса получателя и/или отправителя, идентификатор приложения, метка потока и другие параметры данных, влияющие на выбор маршрута.
- При прокладывании маршрута могут учитываться различные факторы, отражающие состояние и характеристики сети: номинальная пропускная способность; загруженность каналов связи; задержки, вносимые каналами; количество промежуточных транзитных узлов; надежность каналов и транзитных узлов и др.
- Маршруты могут определяться «вручную» администратором сети, однако в больших сетях они чаще всего определяются автоматически. Для этого конечные узлы и другие устройства сети оснащаются специальными программными средствами, которые организуют взаимный обмен служебными сообщениями,

позволяющий каждому узлу составить свое представление о топологии сети.

□ Записи о маршрутах заносятся в таблицы коммутации, устройств сети. В каждой записи признаку (признакам) потока (например, метке, номеру входного интерфейса или адресу назначения) ставится в соответствие номер интерфейса, на который устройство должно передавать данные, относящиеся к этому потоку.

□ Демультиплексирование — это разделение суммарного агрегированного потока на несколько составляющих потоков, мультиплексирование (multiplexing) — объединение нескольких отдельных потоков в один общий агрегированный поток, который можно передавать по одному физическому каналу связи. Важнейшими способами мультиплексирования являются частотное и временное разделение канала. Совместно используемый несколькими интерфейсами физический канал называют разделяемым, (shared).

□ В сетях для соединения абонентов используются три метода коммутации: коммутация каналов, коммутация пакетов и коммутация сообщений.

□ Как коммутация каналов, так и коммутация пакетов может быть либо динамической, либо постоянной.

□ В сетях с коммутацией каналов абонентов соединяет составной канал, образуемый коммутаторами сети по запросу одного из абонентов.

□ Сети с коммутацией каналов хорошо коммутируют потоки данных постоянной интенсивности, например потоки данных, создаваемые разговаривающими по телефону собеседниками, но не могут перераспределять пропускную способность магистральных каналов между потоками абонентских каналов динамически.

□ Коммутация сообщений предназначена для организации взаимодействия пользователей в автономном (off-line) режиме, когда не ожидается немедленной реакции на сообщение. При этом методе коммутации сообщение передается через несколько транзитных компьютеров, где оно целиком буферизуется на диске.

□ Сети с коммутацией пакетов были специально разработаны для эффективной передачи пульсирующего компьютерного трафика. Буферизация пакетов разных абонентов в коммутаторах позволяет сгладить неравномерности интенсивности трафика каждого абонента и равномерно загрузить каналы связи между коммутаторами.

□ В сетях с коммутацией пакетов объем передаваемых данных от всех абонентов сети в единицу времени больше, чем при использовании сети с коммутацией каналов. Однако для каждой пары абонентов пропускная способность сети может оказаться ниже, чем у сети с коммутацией каналов, за счет очередей пакетов в коммутаторах.

□ Сети с коммутацией пакетов могут работать в одном из двух режимов: дейтаграммном режиме или режиме виртуальных каналов. Дейтаграммный способ передачи данных основан на том, что все передаваемые пакеты обрабатываются независимо друг от друга, пакет за пакетом. Механизм виртуальных каналов учитывает существование в сети потоков данных и прокладывает для всех пакетов потока единый маршрут.

Структуризация как средство построения больших сетей

Причины структуризации локальных и глобальных сетей

В сетях с небольшим (10-30) количеством компьютеров чаще всего используется одна из типовых топологий — общая шина, кольцо, звезда или полносвязная сеть. Все перечисленные топологии обладают свойством однородности, то есть все компьютеры в такой сети имеют одинаковые права в отношении доступа к другим компьютерам (за исключением центрального компьютера при соединении звезды). Такая однородность структуры делает простой процедуру наращивания числа компьютеров, облегчает обслуживание и эксплуатацию сети. Однако при построении больших сетей однородная структура связей превращается из преимущества в недостаток. В таких сетях использование типовых структур порождает различные ограничения, важнейшими из которых являются:

- ограничения на длину связи между узлами;
- ограничения на количество узлов в сети;
- ограничения на интенсивность трафика, порождаемого узлами сети.

Например, технология Ethernet на тонком коаксиальном кабеле позволяет использовать кабель длиной не более 185 м, к которому можно подключить не более 30 компьютеров. Однако, если компьютеры интенсивно обмениваются информацией между собой, иногда приходится снижать число подключенных к кабелю компьютеров до 20, а то и до 10, чтобы каждому компьютеру доставалась приемлемая доля общей пропускной способности сети.

Для снятия этих ограничений используются специальные методы структуризации сети и специальное структурообразующее оборудование — повторители, концентраторы, мосты, коммутаторы, маршрутизаторы. Оборудование такого рода также называют коммуникационным, имея в виду, что с его помощью отдельные сегменты сети взаимодействуют между собой.

Физическая и логическая структуризация

Различают топологию физических связей (физическую структуру сети) и топологию логических связей (логическую структуру сети). Конфигурация физических связей определяется электрическими соединениями компьютеров, здесь ребра графа соответствуют отрезкам кабеля, связывающим пары узлов. Логические связи представляют собой маршруты передачи данных между узлами сети и образуются путем соответствующей настройки коммуникационного оборудования.

Физическая структуризация сети

Простейшее из коммуникационных устройств — повторитель (repeater) — используется для физического соединения различных сегментов кабеля локальной

сети с целью увеличения общей длины сети. Повторитель передает сигналы, приходящие из одного сегмента сети, в другие ее сегменты (рис. 2.23). Повторитель позволяет преодолеть ограничения на длину линий связи за счет улучшения качества передаваемого сигнала — восстановления его мощности и амплитуды, улучшения фронтов и т. п.

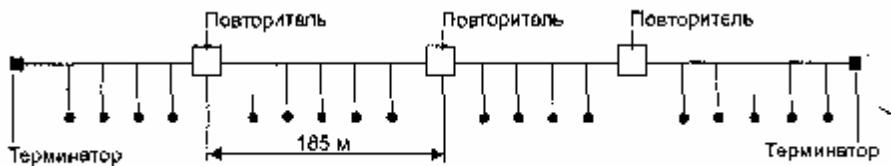


Рис. 2.23. Повторители позволяют увеличить длину сети Ethernet

Повторитель, который имеет несколько портов и соединяет несколько физических сегментов, часто называют концентратором (concentrator), или хабом (hub). Эти названия (hub — основа, центр деятельности) отражают тот факт, что в данном устройстве сосредоточиваются все связи между сегментами сети.

Концентраторы характерны практически для всех базовых технологий локальных сетей — Ethernet, ArcNct, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, IOOVG-AnyLAN.

Нужно подчеркнуть, что в работе концентраторов любых технологий много общего — они повторяют сигналы, пришедшие с одного из своих портов, на других своих портах. Разница состоит в том, на каких именно портах повторяются входные сигналы. Так, концентратор Ethernet повторяет входные сигналы на всех своих портах, кроме того, с которого сигналы поступают (рис. 2.24, а). А концентратор Token Ring (рис. 2.24, б) повторяет входные сигналы, поступающие с некоторого порта, только на одном порту — на том, к которому подключен следующий в кольце компьютер.

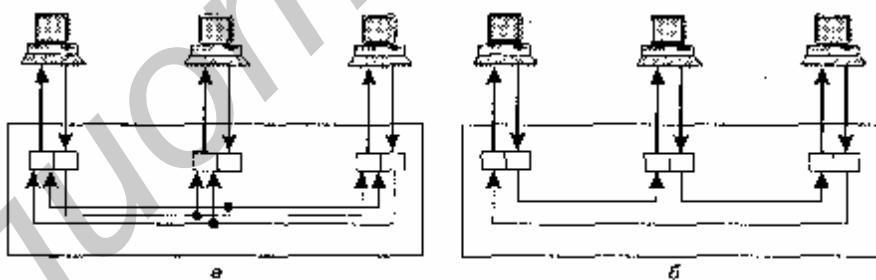


Рис. 2.24. Концентраторы различных технологий

ВНИМАНИЕ

Добавление в сеть концентратора всегда изменяет ее физическую топологию, но при этом оставляет без изменения логическую топологию.

(Не все из перечисленных технологий актуальны, ArcNct и IOOVG-AnyLAN могут рассматриваться сегодня только в качестве примеров оригинальных технических решений.)

Как уже было сказано, под физической топологией понимается конфигурация связей, образованных отдельными частями кабеля, а под логической — конфигурация информационных потоков между компьютерами сети. Во многих случаях физическая и логическая топологии сети совпадают. Например, сеть,

представленная на рис. 2.25, а, имеет физическую топологию кольцо. Компьютеры этой сети получают доступ к кабелям кольца за счет передачи друг другу специального кадра — маркера, причем этот маркер также передается последовательно от компьютера к компьютеру в том же порядке, в котором компьютеры образуют физическое кольцо, то есть компьютер А передает маркер компьютеру В, компьютер В — компьютеру С и т. д.

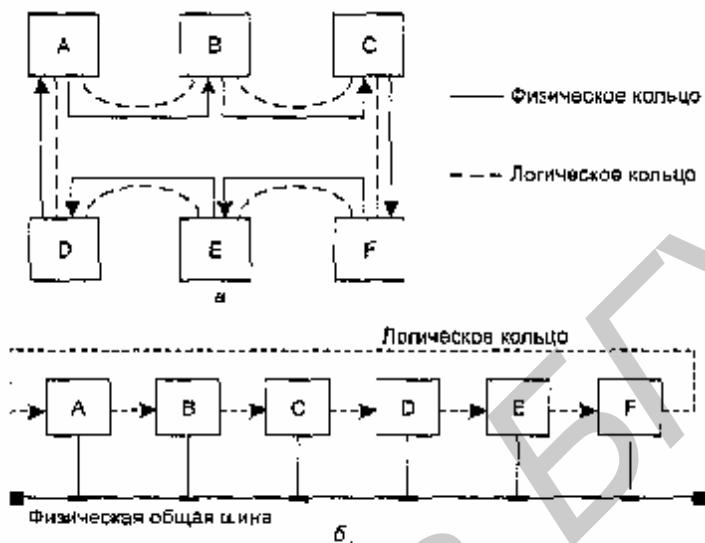


Рис. 2.25. Логическая и физическая топологии сети

Сеть, показанная на рис. 2.25, б, является примером несовпадения физической и логической топологий. Физически компьютеры соединены по топологии общая шина. Доступ же к шине происходит не по алгоритму случайного доступа, применяемому в технологии Ethernet, а путем передачи маркера и кольцевом порядке: от компьютера А — компьютеру В, от компьютера В — компьютеру С и т. д. Здесь порядок передачи маркера уже не повторяет физические связи, а определяется логическим конфигурированием драйверов сетевых адаптеров. Ничто не мешает настроить сетевые адаптеры и их драйверы так, чтобы компьютеры образовали кольцо в другом порядке, например: В, А, С... При этом физическая структура сети никак не изменяется.

Другим примером несовпадения физической и логической топологий сети является уже рассмотренная на рис. 2.24, а сеть. Концентратор Ethernet поддерживает в сети физическую топологию звезды. Однако логическая топология сети осталась без изменений — это общая шина. Так как концентратор повторяет данные, пришедшие с любого порта, на всех остальных портах, то они появляются одновременно на всех физических сегментах сети, как и в сети с физической общей шиной. Логика доступа к сети совершенно не меняется: все компоненты алгоритма случайного доступа — определение незанятости среды, захват среды, распознавание и отработка коллизий — остаются в силе.

Физическая структуризация сети с помощью концентраторов полезна не только для увеличения расстояния между узлами сети, но и для повышения ее надежности. Например, если какой-либо компьютер сети Ethernet с физической общей шиной из-за сбоя начинает непрерывно передавать данные по общему кабелю, то вся сеть выходит из строя и для решения проблемы остается только

один выход — вручную отсоединить сетевой адаптер этого компьютера от кабеля. В сети Ethernet, построенной с использованием концентратора, эта проблема может быть решена автоматически — концентратор отключает свой порт, если обнаруживает, что присоединенный к нему узел слишком долго монопольно занимает сеть. Концентратор может блокировать некорректно работающий узел и в других случаях, выполняя роль некоторого управляющего узла.

Логическая структуризация сети

Физическая структуризация сети полезна во многих отношениях, однако в ряде случаев, обычно относящихся к сетям большого и среднего размера, невозможно обойтись без логической структуризации сети. Наиболее важной проблемой, не решаемой путем физической структуризации, остается проблема перераспределения передаваемого трафика между различными физическими сегментами сети.

В большой сети естественным образом возникает неоднородность информационных потоков: сеть состоит из множества подсетей рабочих групп, отделов, филиалов предприятия и других административных образований. В одних случаях наиболее интенсивный обмен данными наблюдается между компьютерами, принадлежащими к одной подсети, и только небольшая часть обращений происходит к ресурсам компьютеров, находящихся вне локальных рабочих групп. На других предприятиях, особенно там, где имеются централизованные хранилища корпоративных данных, активно используемые всеми сотрудниками предприятия, наблюдается обратная ситуация: интенсивность внешних обращений выше интенсивности обмена между «соседними» машинами. Но независимо от того, в какой пропорции распределяются внешний и внутренний трафики, для повышения эффективности работы сети неоднородность информационных потоков необходимо учитывать.

Сеть с типовой топологией (шина, кольцо, звезда), в которой все физические сегменты рассматриваются в качестве одной разделяемой среды, оказывается неадекватной структуре информационных потоков в большой сети. Например, в сети с общей шиной взаимодействие любой пары компьютеров занимает ее на все время обмена, поэтому при увеличении числа компьютеров в сети шина становится узким местом. Компьютеры одного отдела вынуждены ждать, когда окончит обмен пара компьютеров другого отдела, и это при том, что необходимость в связи между компьютерами двух разных отделов возникает гораздо реже и требует совсем небольшой пропускной способности.

Этот случай иллюстрирует рис. 2.26, а. Здесь показана сеть, построенная с использованием концентраторов. Пусть компьютер А, находящийся в одной подсети с компьютером В, посылает ему данные. Несмотря на разветвленную физическую структуру сети, концентраторы распространяют любой кадр по всем ее сегментам. Поэтому кадр, посланный компьютером А компьютеру В, хотя и не нужен компьютерам отделов 2 и 3, в соответствии с логикой работы концентраторов поступает на эти сегменты тоже. И до тех пор, пока компьютер В

не получит адресованный ему кадр, ни один из компьютеров этой сети не сможет передавать данные.

Такая ситуация возникает из-за того, что логическая структура данной сети осталась однородной — она никак не учитывает увеличение интенсивности трафика внутри отдела и предоставляет всем парам компьютеров равные возможности по обмену информацией (рис. 2.20, б).

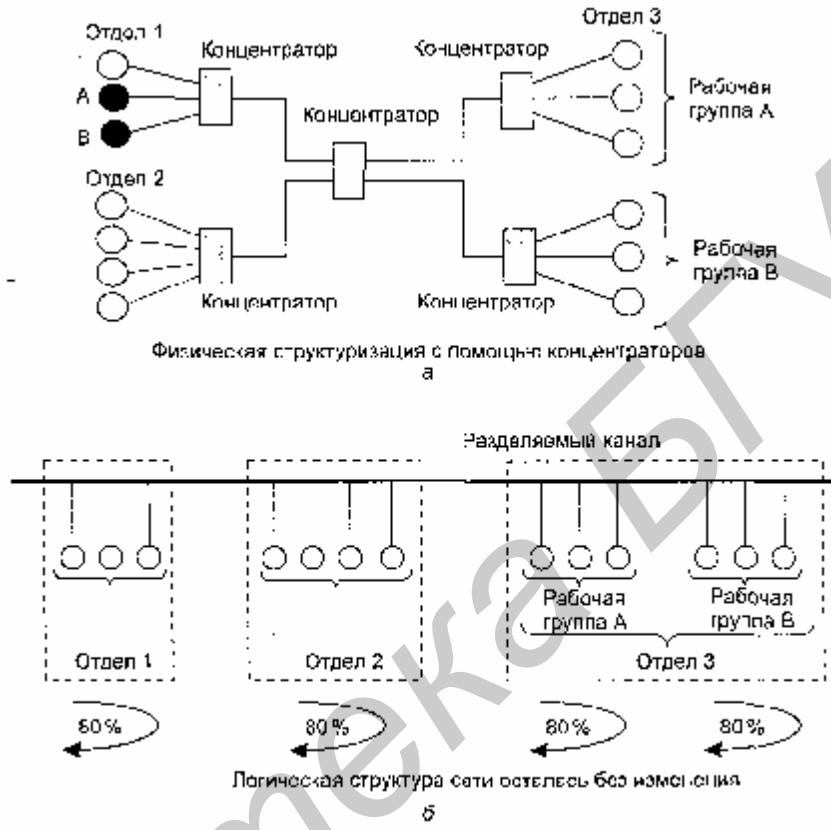


Рис. 2.26. Противоречие между логической структурой сети и структурой информационных потоков

Решение проблемы состоит в отказе от идеи общей однородной разделяемой среды. Например, в рассмотренном выше примере желательно было бы сделать так, чтобы кадры, которые передают компьютеры отдела 1, выходили бы за пределы этой части сети в том и только в том случае, если они направлены какому-либо компьютеру из других отделов. С другой стороны, в сеть каждого из отделов должны попадать те и только те кадры, которые адресованы узлам этой сети.

При такой организации работы сети ее производительность существенно повысится, так как компьютеры одного отдела не будут простоять в то время, когда обмениваются данными компьютеры других отделов.

Нетрудно заметить, что в предложенном решении мы отказались от идеи общей разделяемой среды в пределах всей сети, хотя и оставили ее в пределах каждого отдела. Пропускная способность линий связи между отделами не должна совпадать с пропускной способностью среды внутри отделов. Если трафик между отделами составляет только 20 % от трафика внутри отдела (как уже отмечалось, эта величина может быть другой), то и пропускная способность линий связи и коммуникационного оборудования, соединяющего отделы, может быть

значительно ниже внутреннего трафика сети отдела.

ВНИМАНИЕ

Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента называется локализацией трафика. Логическая структуризация сети — это процесс разбиения сети на сегменты с локализованным трафиком.

Для логической структуризации сети используются такие коммуникационные устройства, как мосты, коммутаторы, маршрутизаторы и шлюзы.

Мост (bridge) делит разделяемую среду передачи сети на части (часто называемые логическими сегментами), передавая информацию из одного сегмента в другой только в том случае, если такая передача действительно необходима, то есть если адрес компьютера назначения принадлежит другой подсети. Тем самым мост изолирует трафик одной подсети от трафика другой, повышая общую производительность передачи данных в сети. Локализация трафика не только экономит пропускную способность, но и уменьшает возможность несанкционированного доступа к данным, так как кадры не выходят за пределы своего сегмента и их сложнее перехватить злоумышленнику.

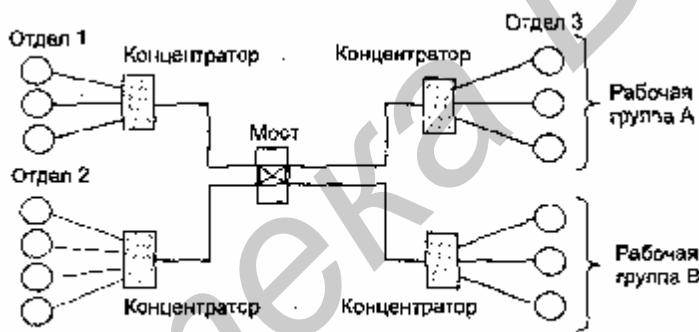


Рис. 2.27. Логическая структуризация сети с помощью моста

На рис. 2.27 показана сеть, которая была получена из сети с центральным концентратором (см. рис. 2.26) путем его замены мостом. Сети 1-го и 2-го отделов состоят из отдельных логических сегментов, а сеть отдела 3 — из двух логических сегментов. Каждый логический сегмент построен на базе концентратора и имеет простейшую физическую структуру, образованную отрезками кабеля, связывающими компьютеры с портами концентратора.

Мосты используют для локализации трафика аппаратные адреса компьютеров. Это затрудняет распознавание принадлежности того или иного компьютера к определенному логическому сегменту — сам адрес не содержит никакой информации по этому поводу. Поэтому мост достаточно упрощенно представляет деление сети на сегменты — он запоминает, через какой порт на него поступил кадр данных от каждого компьютера сети, и в дальнейшем передает кадры, предназначенные для этого компьютера, на этот порт. Точной топологии связей между логическими сегментами мост не знает. Из-за этого применение мостов приводит к значительным ограничениям на конфигурацию связей сети — сегменты должны быть соединены таким образом, чтобы в сети не образовывались

замкнутые контуры.

Коммутатор (switch) по принципу обработки кадров ничем не отличается от моста. Основное его отличие от моста состоит в том, что он является своего рода коммуникационным мультипроцессором, так как каждый его порт оснащен специализированным процессором, который обрабатывает кадры по алгоритму моста независимо от процессоров других портов. За счет этого общая производительность коммутатора обычно намного выше производительности традиционного моста, имеющего один процессорный блок. Можно сказать, что коммутаторы — это мосты нового поколения, которые обрабатывают кадры в параллельном режиме.

Ограничения, связанные с применением мостов и коммутаторов — по топологии связей, а также ряд других, — привели к тому, что в ряду коммуникационных устройств появился еще один тип оборудования — маршрутизатор (router). Маршрутизаторы более надежно и более эффективно, чем мосты, изолируют трафик отдельных частей сети друг от друга. Маршрутизаторы образуют логические сегменты посредством явной адресации, поскольку используют не пло- ские аппаратные, а составные числовые адреса. В этих адресах имеется поле номера сети, так что все компьютеры, у которых значение этого поля одинаково, принадлежат к одному сегменту, называемому в данном случае подсетью (subnet).

(В терминах Интернета маршрутизатор часто называют шлюзом (gateway))

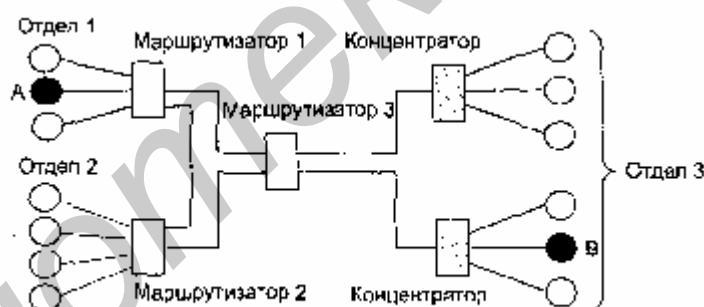


Рис. 2.28. Логическая структуризация сети с помощью маршрутизаторов

Кроме локализации трафика маршрутизаторы выполняют еще много других полезных функций. Так, маршрутизаторы могут работать в сети с замкнутыми контурами, при этом они осуществляют выбор наиболее рационального маршрута из нескольких возможных. Сеть, представленная на рис. 2.28, отличается от своей предшественницы (см. рис. 2.27) тем, что между подсетями отделов 1 и 2 проложена дополнительная связь, которая может использоваться как для повышения производительности сети, так и для повышения ее надежности.

Другой очень важной функцией маршрутизаторов является их способность связывать в единую сеть подсети, построенные с использованием разных сетевых технологий, например Ethernet и X.25.

Кроме перечисленных устройств отдельные части сети может соединять шлюз (gateway). Обычно основной причиной, по которой и сети используют шлюз,

является необходимость объединить сети с разными типами системного и прикладного программного обеспечения, а не желание локализовать трафик. Тем не менее шлюз обеспечивает и локализацию трафика в качестве некоторого побочного эффекта.

Крупные сети практически никогда не строятся без логической структуризации. Для отдельных сегментов и подсетей характерны типовые однородные топологии базовых технологий, и для их объединения всегда используется оборудование, обеспечивающее локализацию трафика, — мосты, коммутаторы, маршрутизаторы и шлюзы.

Выводы

- Различают топологию физических связей (физическую структуру сети), определяемую электрическими соединениями компьютеров, и топологию логических связей (логическую структуру сети), определяемую маршрутами передачи данных между узлами сети.
- Логическая и физическая структуры сети в общем случае не совпадают.
- Повторитель (repeater) — является средством физической структуризации сети, он используется для физического соединения двух или более сегментов
- Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента, называется локализацией трафика.
- Логическая структуризация сети — это процесс разбиения сети на сегменты с локализованным трафиком. Логическая структуризация сети направлена на повышение производительности и безопасности сети.
- Средствами логической структуризации служат такие коммуникационные устройства, как мосты (bridge), коммутаторы (switch), маршрутизаторы (router) и шлюзы (gateway).

Основные программные и аппаратные компоненты сети

Многослойная модель сети

Даже в результате достаточно поверхностного рассмотрения работы в сети становится ясно, что вычислительная сеть — это сложный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов. Изучение сети в целом предполагает знание принципов работы ее отдельных элементов:

- компьютеров;
- коммуникационного оборудования;
- операционных систем;
- сетевых приложений.

Весь комплекс программно-аппаратных средств сети может быть описан многослойной моделью. В основе любой сети лежит аппаратный слой стандартизованных компьютерных платформ. В настоящее время в сетях широко и успешно применяются компьютеры различных классов — от персональных компьютеров до майнфреймов и суперЭВМ. Набор компьютеров в сети должен соответствовать набору разнообразных задач, решаемых сетью.

Второй слой — это коммуникационное оборудование. Хотя компьютеры и являются центральными элементами обработки данных в сетях, в последнее время не менее важную роль стали играть коммуникационные устройства. Кабельные системы, повторители, мосты, коммутаторы, маршрутизаторы и модульные концентраторы из вспомогательных компонентов сети превратились в основные наряду с компьютерами и системным программным обеспечением как по влиянию на характеристики сети, так и по стоимости. Сегодня коммуникационное устройство может представлять собой сложный специализированный мультипроцессор, который нужно конфигурировать, оптимизировать и администрировать. Изучение принципов работы коммуникационного оборудования требует знакомства с большим количеством протоколов, используемых как в локальных, так и глобальных сетях.

Третим слоем, образующим программную платформу сети, являются операционные системы (ОС). От того, какие концепции управления локальными и распределенными ресурсами положены в основу сетевой ОС, зависит эффективность работы всей сети. При проектировании сети важно учитывать, насколько просто данная операционная система может взаимодействовать с другими ОС сети, насколько она обеспечивает безопасность и защищенность данных, до какой степени она позволяет наращивать число пользователей, можно ли перенести ее на компьютер другого типа и многие другие соображения.

Самым верхним слоем сетевых средств являются различные сетевые приложения, такие как сетевые базы данных, почтовые системы, средства архивирования данных, системы автоматизации коллективной работы и др. Очень важно представлять диапазон возможностей, предоставляемых приложениями для различных областей применения, а также знать, насколько они совместимы с

другими сетевыми приложениями и операционными системами.

Сетевые службы и операционная система

Для конечного пользователя сеть — это не компьютеры, кабели и концентраторы и даже не информационные потоки, для него сеть — это, прежде всего, тот набор сетевых служб, с помощью которых он получает возможность просмотреть список имеющихся в сети компьютеров, прочитать удаленный файл, распечатать документ на «чужом» принтере или послать почтовое сообщение. Именно совокупность предоставляемых возможностей — насколько широк их выбор, насколько они удобны, надежны и безопасны — определяет для пользователя облик той или иной сети.

Кроме собственно обмена данными, сетевые службы должны решать и другие, более специфические задачи, например задачи, порождаемые распределенной обработкой данных. К таким задачам относится обеспечение непротиворечивости нескольких копий данных, размещенных на разных машинах (служба репликации), или организация выполнения одной задачи параллельно на нескольких машинах сети (служба вызова удаленных процедур). Среди сетевых служб можно выделить административные, то есть такие, которые в основном ориентированы не на простого пользователя, а на администратора и служат для организации правильной работы сети в целом. Служба администрирования учетных записей о пользователях сети, которая позволяет администратору вести общую базу данных о пользователях сети, система мониторинга сети, позволяющая захватывать и анализировать сетевой трафик, служба безопасности, в функции которой может входить среди прочего выполнение процедуры логического входа с последующей проверкой пароля, — все это примеры административных служб. Реализация сетевых служб осуществляется программными средствами. Основные службы — файловая служба и служба печати — обычно предоставляются сетевой операционной системой, а вспомогательные, например служба баз данных, факса или передачи голоса, — системными сетевыми приложениями, или утилитами, работающими в тесном контакте с сетевой ОС. Вообще говоря, распределение служб между ОС и утилитами достаточно условно и меняется в конкретных реализациях ОС.

При разработке сетевых служб приходится решать проблемы, которые свойственны любым распределенным приложениям: определение протокола взаимодействия между клиентской и серверной частями, распределение функций между ними, выбор схемы адресации приложений и др.

Одним из главных показателей качества сетевой службы является ее удобство. Для одного и того же ресурса может быть разработано несколько служб, по-разному решающих, в общем-то одну и ту же задачу. Отличия могут заключаться в производительности или уровне удобства предоставляемых услуг. Например, файловая служба может быть основана на использовании команды передачи файла из одного компьютера в другой по имени файла, а это требует от пользователя знания этого имени. Та же файловая служба может быть реализована и так, что пользователь монтирует удаленную файловую систему к ло-

кальному каталогу, а далее обращается к удаленным файлам как к своим собственным, что гораздо удобнее. Качество сетевой службы зависит и от качества пользовательского интерфейса — интуитивной понятности, наглядности, рациональности.

При определении степени удобства разделяемого ресурса часто употребляют термин «прозрачность». Прозрачный доступ — это такой доступ, при котором пользователь не замечает, где расположен нужный ему ресурс — на собственном или удаленном компьютере. После того как он смонтировал удаленную файловую систему в свое дерево каталогов, доступ к удаленным файлам становится для него совершенно прозрачным. Сама операция монтирования также может иметь разную степень прозрачности — в сетях с меньшей прозрачностью пользователь должен знать и задавать в команде имя компьютера, на котором хранится удаленная файловая система, в сетях с большей степенью прозрачности соответствующий программный компонент сети производит поиск разделяемых томов файлов безотносительно мест их хранения, а затем предоставляет их пользователю в удобном для него виде, например в виде списка или набора значков.

Для обеспечения прозрачности важен способ адресации (именования) разделяемых сетевых ресурсов. Имена разделяемых сетевых ресурсов не должны зависеть от их физического расположения на том или ином компьютере. В идеале пользователь не должен ничего менять в своей работе, если администратор сети переместил том или каталог с одного компьютера на другой. Сам администратор и сетевая операционная система имеют информацию о расположении файловых систем, но от пользователя она скрыта. Такая степень прозрачности пока редко встречается в сетях — обычно для получения доступа к ресурсам определенного компьютера сначала приходится устанавливать с ним логическое соединение. Такой подход применяется, например, в сетях Windows NT.

Общая структура телекоммуникационной сети

Несмотря на то что различия между компьютерными, телефонными, телевизионными и первичными сетями, безусловно, существенны, все эти сети на достаточно высоком уровне абстракции имеют подобные структуры. Телекоммуникационная сеть в общем случае состоит из следующих компонентов (рис. 2.29):

- сети доступа (access network);
- магистральной сети, или магистрали (core network, или backbone);
- информационных центров, или центров управления сервисами (data centers, или services control point).

Как сеть доступа, так и магистральная сеть строятся на основе коммутаторов. Каждый коммутатор оснащен некоторым количеством портов, которые соединяются с портами других коммутаторов каналами связи. Сеть доступа составляет нижний уровень иерархии телекоммуникационной сети. К этой сети подключаются конечные (терминалные) узлы — оборудование, установленное у пользователей (абонентов, клиентов) сети. В случае компьютерной сети конечными узлами являются компьютеры, телефонной — телефонные аппараты, а

телевизионной или радиосети — соответствующие телевизионные и радиоприемники. Основное назначение сети доступа — концентрация информационных потоков, поступающих по многочисленным каналам связи от оборудования пользователей, в сравнительно небольшом количестве узлов магистральной сети. Сеть доступа, как и телекоммуникационная сеть в целом, может состоять из нескольких уровней (на рисунке их показано два). Коммутаторы, установленные в узлах нижнего уровня, мультиплексируют информацию, поступающую по многочисленным абонентским каналам, называемыми часто абонентскими окончаниями (local loop), и передают ее коммутаторам верхнего уровня, чтобы те, в свою очередь, передали ее коммутаторам магистрали.

Количество уровней сети доступа зависит от ее размера, небольшая сеть доступа может состоять из одного уровня, а крупная — из двух-трех. Следующие уровни осуществляют дальнейшую концентрацию трафика, собирая его и мультиплексируя в более скоростные каналы.

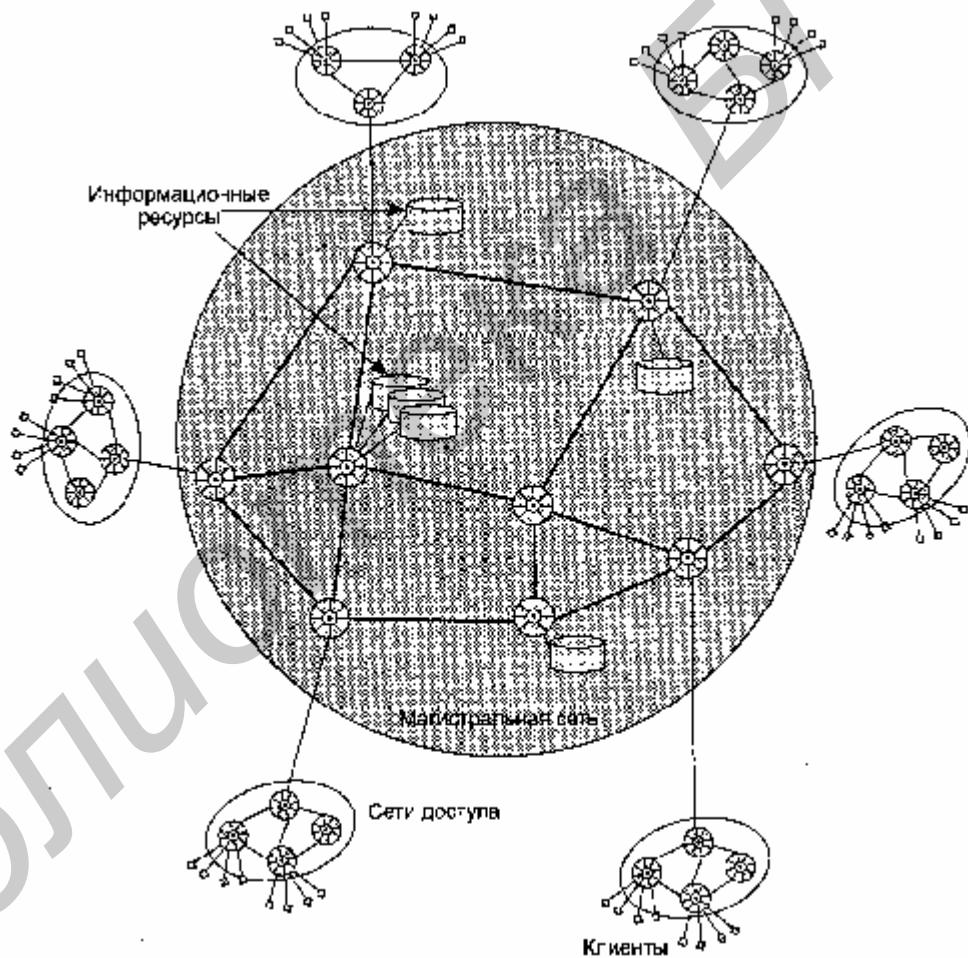


Рис. 2.29. Структура телекоммуникационной сети

Магистральная сеть объединяет отдельные сети доступа, выполняя функции транзита трафика между ними по высокоскоростным каналам. Коммутаторы магистрали могут оперировать не только с информационными соединениями между отдельными пользователями, но и с агрегированными информационными потоками, переносящими данные большого количества пользовательских соединений. В результате информация с помощью магистрали попадает в сеть доступа получателей, демультиплексируется там и коммутируется таким образом, что на входной порт оборудования пользователя поступает только та информация, которая ему адресована.

В том случае, когда абонент-получатель подключен к тому же коммутатору доступа, что и абонент- отправитель (непосредственно или через подчиненные по иерархии связей коммутаторы), последний самостоятельно выполняет необходимую операцию коммутации.

Информационные центры, или центры управления сервисами, — это собственные информационные ресурсы сети, на основе которых осуществляется обслуживание пользователей. В таких центрах может храниться информация двух типов:

- пользовательская информация, то есть та, которая непосредственно интересует конечных пользователей сети;
- вспомогательная служебная информация, помогающая предоставлять некоторые услуги пользователям.

Примером информационных ресурсов первого типа могут служить web-порталы, на которых расположена разнообразная справочная и новостная информация, информация электронных магазинов и т. п. В телефонных сетях такими центрами являются службы экстренного вызова (например, милиции, скорой помощи) и справочные службы различных организаций и предприятий — вокзалов, аэропортов, магазинов и т. п. В телевизионных сетях такими центрами являются телестудии, поставляющие «живую» картинку или же воспроизводящие ранее записанные сюжеты или фильмы.

Ресурсами второго типа являются, например, различные системы аутентификации и авторизации пользователей, с помощью которых организация, владеющая сетью, проверяет права пользователей на получение тех или иных услуг; системы биллинга, которые в коммерческих сетях подсчитывают плату за полученные услуги; базы данных учетной информации пользователей, хранящие имена и пароли, а также перечни услуг, на которые подписан каждый пользователь. В телефонных сетях существуют централизованные центры управления сервисами (Services Control Point, SCP), в которых установлены компьютеры, хранящие программы нестандартной обработки телефонных вызовов пользователей, например вызовов к бесплатным справочным службам коммерческих предприятий (так называемые службы 800) или вызовов при проведении телеголосования. Еще одним из распространенных видов вспомогательного информационного центра является централизованная система управления сетью, которая представляет собой программное обеспечение, работающее на одном или не-

скольких компьютерах.

Естественно, у сетей каждого конкретного типа имеется много особенностей, тем не менее их структура в целом соответствует описанной выше. В то же время, в зависимости от назначения и размера сети, в ней могут отсутствовать или же иметь несущественное значение некоторые составляющие обобщенной структуры. Например, в небольшой локальной компьютерной сети нет ярко выраженных сетей доступа и магистрали — они сливаются в общую и достаточно простую структуру. В корпоративной сети, как правило, отсутствует система биллинга, так как услуги сотрудником предприятия оказываются не на коммерческой основе. В некоторых телефонных сетях могут отсутствовать информационные центры, а в телевизионных — сеть доступа приобретает вид распределительной сети, так как информация в ней распространяется только в одном направлении — из сети к абонентам.

Выводы

- Вычислительная сеть — это многослойный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов: компьютеров, коммуникационного оборудования, операционных систем, сетевых приложений.
- Для конечного пользователя сеть представляет собой, прежде всего, набор сетевых служб, обеспечивающих ему совместное с другими пользователями сети использование сетевых ресурсов. Основные службы — файловая служба и служба печати — обычно предоставляются сетевой ОС, а вспомогательные, например служба баз данных, факса или передачи голоса, — системными сетевыми приложениями.
- Важнейшим свойством сетевой службы является ее прозрачность.
- Телекоммуникационная сеть в общем случае состоит из следующих компонентов: сети доступа (access network), магистрали (backbone), информационных центров (data centers).
- Сеть доступа предназначена для концентрации информационных потоков, поступающих по многочисленным каналам связи от оборудования пользователей, в сравнительно небольшом количестве узлов магистральной сети.
- Магистральная сеть объединяет отдельные сети доступа, выполняя функции транзита трафика между ними по высокоскоростным каналам.
- Информационные центры — это собственные информационные ресурсы сети, на основе которых осуществляется обслуживание пользователей.

Сети операторов связи

Компьютерные сети можно классифицировать по различным критериям. Деление на локальные и глобальные сети происходит по территориальному признаку, то есть по величине территории, которую покрывает сеть. Другим важным признаком классификации сетей является назначение предоставляемых услуг

— сети операторов связи (сети поставщиков услуг) оказывают общедоступные услуги, а корпоративные сети — услуги сотрудникам только того предприятия, которое владеет сетью.

Операторы связи и клиенты

Существуют сети, которые создаются специально для оказания общедоступных, или публичных (public), телекоммуникационных услуг. Примерами таких сетей являются городские, региональные, национальные и международные телефонные сети. Их услугами пользуются многочисленные клиенты — владельцы домашних и мобильных телефонов, а также предприятия (корпоративные пользователи). Еще одной традиционной телекоммуникационной услугой является предоставление в аренду каналов связи. У первичных сетей PDH/SDH, создаваемых телекоммуникационным предприятием для объединения своих АТС, обычно остается неиспользованная для внутренних нужд канальная емкость, которую логично сдавать в аренду. Типичными потребителями этой услуги являются крупные предприятия, которые создают с помощью арендованных каналов собственные сети — телефонные или компьютерные.

По мере роста популярности компьютерной обработки данных к набору телекоммуникационных услуг добавляется услуга объединения локальных сетей предприятий с помощью общедоступной территориальной сети передачи данных, например, сети технологии X.25, frame relay, ATM или IP. Интернет-революция 90-х годов породила такую распространенную общедоступную услугу, как доступ к Интернету для обмена сообщениями электронной почты и использования ресурсов многочисленных web-сайтов. Вскоре Интернет стал использоваться предприятиями не только для доступа к «чужим» информационным ресурсам, но и для объединения своих собственных, то есть как типичная сеть передачи данных, оказывающая транспортные услуги. На стыке телефонных и компьютерных сетей начали появляться новые типы общедоступных услуг, опирающиеся на возможности комплексного использования различных технологий.

Специализированное предприятие, которое создает телекоммуникационную сеть для оказания общедоступных услуг, владеет этой сетью и поддерживает ее работу, традиционно называется оператором связи (telecommunication earlier). Операторы связи осуществляют свою деятельность на коммерческой основе, заключая договоры с потребителями услуг.

Операторы связи отличаются друг от друга:

- набором предоставляемых услуг;
- территорией, в пределах которой предоставляются услуги;
- типом клиентов, на которых ориентируются их услуги;
- имеющейся во владении оператора инфраструктурой — линиями связи, коммутационным оборудованием, информационными серверами и т. п.

Особенностью современных операторов связи является то, что они, как правило, оказывают услуги нескольких типов, например услуги телефонии и доступа в Интернет. Поэтому при классификации полезно рассматривать всех оператор-

ров связи, а не только тех, которые оказывают услуги компьютерных сетей. (Существуют и более формальные определения этого термина, например, в «Законе о связи» Российской Федерации, но они кроме организационно-технических затрагивают также и юридические аспекты, что выходит за рамки данной книги.).

Услуги, поставщики услуг и сетевая инфраструктура

Услуги можно разделить на несколько уровней и групп. На рис. 2.30 показаны только некоторые основные уровни и группы, но и эта неполная картина хорошо иллюстрирует широту спектра современных телекоммуникационных услуг и сложность их взаимосвязей. Услуги более высокого уровня опираются на услуги нижележащих уровней. Группы услуг выделены по типу сетей, которые их оказывают, — телефонные или компьютерные (для полноты картины нужно было бы дополнить рисунок услугами телевизионных и радиосетей).

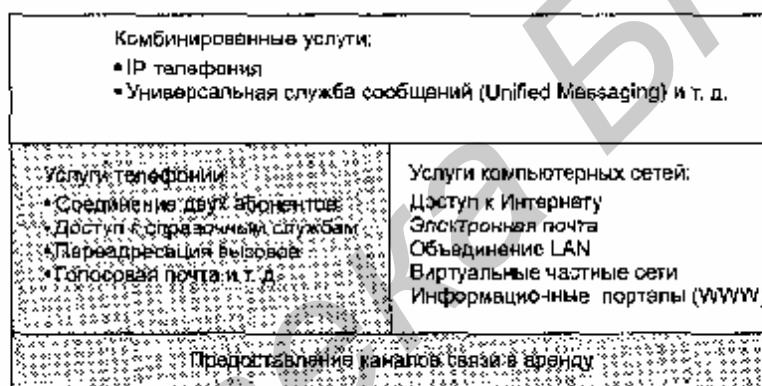


Рис. 2.30. Классификация услуг телекоммуникационной сети (закрашенные области соответствуют традиционным услугам операторов связи)

Услуги предоставления каналов связи в аренду являются услугами самого нижнего уровня, так как пользователь при этом должен еще построить с помощью этих каналов собственную сетевую инфраструктуру (установить телефонные коммутаторы или коммутаторы пакетных сетей), прежде чем начать извлекать из них какую-либо пользу.

Следующий уровень составляют две большие группы услуг: телефонные услуги и услуги компьютерных сетей. Этот уровень, в свою очередь, можно делить на подуровни, так как из более простых услуг строятся более сложные. Например, на основе услуги доступа в Интернет, которая заключается в простом транспортном подключении компьютера или локальной сети ко всемирной общедоступной сети, можно оказывать услуги электронной почты. Или же на основе базовой телефонной услуги соединения абонентов можно создать услугу голосовой почты.

Верхний уровень сегодня занимают комбинированные услуги, реализация которых требует совместного оперативного взаимодействия компьютерных и телефонных сетей.

Услуги можно разделить и по другому принципу — услуги, состоящие в передаче трафика в неизменном виде между абонентами сети, и услуги, состоящие в

предоставлении пользователю информации, созданной оператором или операторами связи. Телефонный разговор — это пример услуги первого типа, так как информация создается абонентами сети, а оператор только доставляет ее от одного абонента к другому. К этому же типу относится услуга соединения двух локальных сетей клиента с помощью сети передачи данных оператора. Примерами услуг второго типа являются услуги справочных служб телефонной сети или услуги какого-либо web-сайта. Услуги второго типа обычно называют информационными.

Еще один общеупотребительный признак классификации услуг — наличие или отсутствие интерактивности. Если при оказании услуги попеременно и в реальном масштабе времени активны два (или более) абонента, то это интерактивная услуга. Телефонный разговор — классический пример интерактивной услуги. Если же абонент только получает информацию, передаваемую по сети, то это не интерактивная услуга. Например, таковы услуги радио и телевидения, аналогичные услуги предоставляют web-сайты, однако здесь грань между интерактивностью и неинтерактивностью провести зачастую сложно, так как пользователь может активно участвовать в просмотре содержания сайта, отвечая на вопросы анкеты или играя в игры.

Каждый оператор связи оказывает те услуги из общего набора, которые ему представляются наиболее соответствующими его профилю и которые могут принести ему экономическую выгоду. В большинстве стран мира (и в России тоже) операторы связи должны получать лицензии от государственных органов на оказание тех или иных услуг связи. Такое положение существовало не всегда — практически во всех странах были операторы, которые являлись фактическими монополистами на рынке телекоммуникационных услуг в масштабах страны. В современном конкурентном мире происходит процесс демонополизации этой области деятельности, бывшие монополисты продолжают работать, но им приходится бороться за клиентов с новыми операторами, которых часто называют альтернативными, так как они действительно являются для пользователей альтернативой традиционному оператору, давая им свободу выбора.

Клиенты

Все множество клиентов — потребителей инфотелекоммуникационных услуг — можно разделить на два больших лагеря: массовые индивидуальные клиенты и корпоративные клиенты.

В первом случае местом потребления услуг выступает квартира или частный дом, а клиентами — жильцы этой квартиры, которым нужны, прежде всего, базовые услуги — телефонная связь, телевидение, радио и (пока еще не всем) выход в Интернет. Для массовых клиентов очень важна экономичность услуги — низкая месячная оплата, возможность использования стандартных терминалльных устройств, таких как телефонные аппараты, телевизионные приемники, персональные компьютеры, а также возможность использования существующей в квартире проводки в виде телефонной пары и телевизионного коаксиального кабеля. Сложные в обращении и дорогие терминалльные устройства, такие

как, например, компьютеризированные телевизоры или IP-телефоны, вряд ли станут массовыми до тех пор, пока не приблизятся по стоимости к обычным телевизорам или телефонам и не будут поддерживать простой интерфейс с пользователем, не требующий для его освоения прослушивания специальных курсов. Существующая в наших домах проводка — это серьезное ограничение для предоставления услуг доступа в Интернет и новых услуг компьютерных сетей, так как она не была рассчитана на передачу данных, а подведение к каждому дому нового качественного кабеля, например волоконно-оптического, — дело дорогое. Поэтому доступ в Интернет чаще всего осуществляется с помощью низкоскоростного модемного соединения по телефонной сети, но постепенно приобретают популярность новые технологии — так называемые технологии цифровых абонентских линий, позволяющие передавать по существующей телефонной проводке данные с гораздо более высокими скоростями, чем обычные модемы.

Корпоративные клиенты — это предприятия и организации различного профиля. Мелкие предприятия по набору предпочтительных услуг не слишком отличаются от массовых клиентов — это та же базовая телефония и телевидение, только телефонных номеров такому предприятию может потребоваться не один, а два-три, да и потребности в передаче данных сводятся к стандартному модемному доступу к информационным ресурсам Интернета. Крупные же предприятия, состоящие из нескольких территориально рассредоточенных отделений и филиалов, а также имеющие сотрудников, часто работающих дома, нуждаются в расширенном наборе услуг. Прежде всего, такой услугой является виртуальная частная сеть (Virtual Private Network, VPN), когда оператор связи создает для предприятия иллюзию того, что все его отделения и филиалы соединены частной сетью, то есть сетью, полностью принадлежащей предприятию-клиенту и полностью управляемой предприятием-клиентом, в то время как на самом деле при этом используется сеть оператора, то есть общедоступная сеть, которая одновременно передает данные многих клиентов. Услуги VPN могут предоставляться как для телефонии (например, корпоративные пользователи звонят по сокращенным внутренним номерам), так и для сетей передачи данных. В последнее время корпоративные пользователи все чаще пользуются не только коммуникационными, но и информационными услугами операторов — например, переносят свои собственные web-сайты и базы данных на территорию оператора, поручая последнему поддерживать их работу и обеспечивать быстрый доступ к ним для сотрудников предприятия и, возможно, других пользователей сети оператора.

Крупные корпоративные клиенты требуют расширенного набора услуг и согласны платить за него больше, чем за стандартный набор, если услуги предоставляются с высоким качеством. Поэтому оператор может пойти на прокладку новой физической линии связи до помещения такого клиента и на установку сложных коммуникационных устройств.

Существуют операторы, которые оказывают услуги как массовым, так и корпоративным абонентам, в то же время существуют операторы, которые специализируются только на одной категории клиентов.

Инфраструктура

Кроме субъективных и правовых причин на формирование набора предлагаемых оператором услуг оказывает серьезное влияние материально-технический фактор, так как для оказания определенной услуги оператор должен владеть соответствующей аппаратно-программной инфраструктурой. Так, для оказания услуг по аренде каналов оператор должен иметь в своем распоряжении транспортную сеть — например, первичную сеть PDH/SDH или же сеть с коммутацией каналов, такую как ISDN. Для оказания информационных web-услуг — создать свой собственный сайт, который должен быть соединен с Интернетом, чтобы пользователи Интернета смогли получить к нему доступ.

В тех случаях, когда у оператора отсутствует вся необходимая инфраструктура для оказания некоторой услуги, он может воспользоваться услугами другого оператора, на основе которых, а также собственных элементов инфраструктуры требуемая услуга может быть сконструирована. Например, для создания общедоступного web-сайта электронной коммерции оператор связи может не иметь собственной IP-сети, соединенной с Интернетом. Для этого ему достаточно только создать информационное наполнение сайта и поместить его на компьютере другого оператора, сеть которого имеет подключение к Интернету. Аренда физических каналов связи для создания собственной телефонной или компьютерной сети является другим типичным примером предоставления услуг при отсутствии одного из элементов аппаратно-программной инфраструктуры. Оператора, который предоставляет услуги другим операторам связи, часто называют оператором операторов (*carrier of carriers*),

Физические каналы связи и другие элементы первичной транспортной инфраструктуры играют ключевую роль при оказании любых коммуникационных услуг, так без них передача информации на расстояние становится просто невозможной. Фактически от наличия или отсутствия собственной транспортной инфраструктуры зависит и название предприятия, оказывающего информационно-коммуникационные услуги — традиционное «оператор связи» или новое «поставщик (провайдер) услуг». Хотя каждый оператор связи, безусловно, оказывает услуги своим клиентам, то есть является поставщиком услуг, в этих терминах имеются различные оттенки. Говоря «оператор связи», обычно подчеркивают то обстоятельство, что компания владеет собственной транспортной инфраструктурой, поддерживает ее функционирование (оперирует ею) и на этой основе предоставляет услуги. Обычно традиционный оператор связи в первую очередь оказывает низкоуровневые транспортные услуги — простую передачу трафика (телефонного или данных) между географическими пунктами без его дополнительной обработки. Когда же говорят «поставщик услуг», то акцент делается на том, что предприятие оказывает новые высокоуровневые услуги, например доступ в Интернет, размещение в своей сети информационных ресурсов (web-сайтов, баз данных предприятий), и не обязательно владеет собственной развитой транспортной инфраструктурой, так как часто для их эффективной реализации достаточно арендованных сетевых ресурсов.

Территория покрытия

По степени покрытия территории, на которой услуги предоставляются, операторы делятся на локальных, региональных, национальных и транснациональных.

Локальный оператор работает на территории города или сельского района. Традиционный локальный оператор — это оператор городской телефонной сети, который владеет всей соответствующей транспортной инфраструктурой: физическими каналами между помещениями абонентов (квартирами, домами и офисами) и узлом связи, телефонными станциями (АТС) и каналами связи между телефонными станциями. Сегодня к традиционным локальным операторам добавились альтернативные, которые часто являются поставщиками услуг нового типа, прежде всего, связанных с Интернетом, но иногда конкурируют с традиционными операторами и в секторе телефонии. Несмотря на демонополизацию телекоммуникационной отрасли, физическими каналами доступа к абонентам по-прежнему владеют традиционные локальные операторы, то есть «городские телефонные сети» (ГТС), поэтому альтернативным операторам проще оказывать высокоуровневые услуги, не столь сильно зависящие от наличия прямой связи с абонентом, такие, например, как доступ в Интернет, размещение в своих узлах информационных ресурсов и т. п. А для организации доступа абонентов к этим ресурсам заключать договор с традиционным оператором, который направляет трафик непосредственно подключенных к нему абонентов в сеть альтернативного оператора, если абонент подписался на соответствующую услугу. Здесь мы видим естественную специализацию операторов — каждый занимается тем делом, для которого в большей степени подходит его инфраструктура, при этом сотрудничество приносит дополнительный эффект, порождая новые услуги.

Региональные и национальные операторы оказывают услуги на большой территории, также располагая соответствующей инфраструктурой. Традиционные операторы этого масштаба выполняют транзитную передачу телефонного трафика между телефонными станциями локальных операторов, имея в своем распоряжении крупные транзитные АТС, связанные высокоскоростными физическими каналами связи. Это операторы операторов, их клиентами являются, как правило, локальные операторы или крупные предприятия, имеющие отделения и филиалы в различных городах региона или страны. Располагая развитой транспортной инфраструктурой, такие операторы обычно оказывают услуги дальней связи, передавая транзитом большие объемы информации без их какой-либо обработки. Альтернативные операторы регионального и национального масштаба могут как иметь, так и не иметь собственную транспортную инфраструктуру. В первом случае они конкурируют с традиционными операторами на рынке услуг дальней связи, а во втором стараются оказывать дополнительные информационные услуги, заключая договоры с большим числом локальных операторов, а также с каким-либо из операторов, предоставляющих услуги дальней связи — для организации взаимодействия между своими ин-

формационными ресурсами.

Выводы

- Специализированное предприятие, которое создает телекоммуникационную сеть для оказания общедоступных услуг, владеет этой сетью и поддерживает ее работу, традиционно называется оператором связи (telecommunication carrier).
- Операторы связи отличаются друг от друга: набором предоставляемых услуг, территорией, в пределах которой предоставляются услуги, типом клиентов, на которых ориентируются их услуги, а также имеющейся во владении оператора инфраструктурой — линиями связи, коммутационным оборудованием, информационными серверами и т. п.
- Множество клиентов может быть разделено на массовых индивидуальных клиентов и корпоративных клиентов.
- Оператора, который предоставляет услуги другим операторам связи, часто называют оператором операторов (carrier of carriers).
- По степени покрытия территории, на которой услуги предоставляются, операторы делятся на локальных, региональных, национальных и транснациональных.

Корпоративные сети

Корпоративная сеть — это сеть, главным назначением которой является поддержание работы конкретного предприятия, владеющего данной сетью. Пользователями корпоративной сети являются только сотрудники данного предприятия. В отличие от сетей операторов связи, корпоративные сети, в общем случае, не оказывают услуг сторонним организациям или пользователям. В зависимости от масштаба предприятия, а также от сложности и многообразия решаемых задач различают сети отдела, сети кампуса и корпоративные сети (термин «корпоративные»- в данной классификации приобретает узкое значение — сеть большого предприятия). Прежде чем обсуждать характерные особенности каждого из перечисленных типов сетей, остановимся на побудительных мотивах, которые приводят предприятия к решению обзавестись собственной компьютерной сетью.

Что дает предприятию использование сетей?

Вопрос о том, что дает предприятию использование сетей, естественным образом порождает другие вопросы: в каких случаях развертывание на предприятии вычислительных сетей предпочтительнее использования автономных компьютеров или многомашинных систем? Какие новые возможности появляются на предприятии с появлением там вычислительной сети? И наконец, всегда ли

предприятию нужна сеть?

Если не вдаваться в частности, то конечной целью использования компьютерных сетей на предприятии является повышение эффективности его работы, которое может выражаться, например, в увеличении прибыли предприятия. Действительно, если благодаря компьютеризации снизились затраты на производство уже существующего продукта, сократились сроки разработки новой модели или ускорилось обслуживание заказов потребителей — это означает, что данному предприятию действительно нужна была сеть.

Концептуальным преимуществом сетей, которое вытекает из их принадлежности к распределенным системам, перед автономно работающими компьютерами является их способность выполнять параллельные вычисления. За счет этого в системе с несколькими обрабатывающими узлами в принципе может быть достигнута производительность, превышающая максимально возможную на данный момент производительность любого отдельного, сколь угодно мощного процессора. Распределенные системы потенциально имеют лучшее соотношение производительность-стоимость, чем централизованные системы.

Еще одно очевидное и важное достоинство распределенных систем — это их принципиально более высокая отказоустойчивость. Под отказоустойчивостью понимается способность системы выполнять свои функции (может быть, не в полном объеме) при отказах отдельных элементов аппаратуры и неполной доступности данных. Основой повышенной отказоустойчивости распределенных систем является избыточность. Избыточность обрабатывающих узлов (процессоров в многопроцессорных системах или компьютеров в сетях) позволяет при отказе одного узла переназначать приписанные ему задачи на другие узлы. С этой целью в распределенной системе могут быть предусмотрены процедуры динамической или статической реконфигурации. В вычислительных сетях некоторые наборы данных могут дублироваться на внешних запоминающих устройствах нескольких компьютеров сети, так что при отказе одного из них данные остаются доступными.

Использование территориально распределенных вычислительных систем больше соответствует распределенному характеру прикладных задач в некоторых предметных областях, таких как автоматизация технологических процессов, банковская деятельность и т. п. Во всех этих случаях имеются рассредоточенные по некоторой территории отдельные потребители информации — сотрудники, организации или технологические установки. Эти потребители автономно решают свои задачи, поэтому рациональнее предоставлять им собственные вычислительные средства, но в то же время, поскольку решаемые ими задачи логически тесно взаимосвязаны, их вычислительные средства должны быть объединены в единую систему. Адекватным решением в такой ситуации является использование вычислительной сети.

Для пользователя распределенные системы дают еще, кроме выше названных, и такие преимущества, как возможность совместного использования данных и устройств, а также возможность гибкого распределения работ по всей системе. Подобное разделение дорогостоящих периферийных устройств, таких как дисковые массивы большой емкости, цветные принтеры, графопостроители, моде-

мы, оптические диски, во многих случаях является основной причиной развертывания сети на предприятии. Пользователь современной вычислительной сети работает за своим компьютером, часто не отдавая себе отчета в том, что при этом он пользуется данными другого мощного компьютера, находящегося за сотни километров от него. Он отправляет электронную почту через модем, подключенный к коммуникационному серверу, общему для нескольких отделов его предприятия. У пользователя создается иллюзия, что эти ресурсы подключены непосредственно к его компьютеру или же «почти» подключены, так как для их использования нужны незначительные дополнительные действия по сравнению с использованием действительно собственных ресурсов.

В последнее время стал преобладать другой побудительный мотив развертывания сетей, гораздо более важный в современных условиях, чем экономия средств за счет разделения между сотрудниками корпорации дорогой аппаратуры или программ. Этим мотивом стало стремление обеспечить сотрудникам оперативный доступ к обширной корпоративной информации. В условиях жесткой конкурентной борьбы в любом секторе рынка выигрывает, в конечном счете, та компания, сотрудники которой могут быстро и правильно ответить на любой вопрос клиента — о возможностях их продукции, об условиях ее применения, о решении любых возможных проблем и т. п. На большом предприятии даже хороший менеджер вряд ли знает все характеристики каждого из выпускаемых продуктов, тем более что их номенклатура может обновляться каждый квартал, если не месяц. Поэтому очень важно, чтобы менеджер имел возможность со своего компьютера, подключенного к корпоративной сети, скажем в Магадане, передать вопрос клиента на сервер, расположенный в центральном отделении предприятия в Новосибирске, и оперативно получить качественный ответ, удовлетворяющий клиента. В таком случае клиент не обратится в другую компанию, а будет пользоваться услугами данного менеджера и впредь.

Наличие сети приводит к совершенствованию коммуникаций между сотрудниками предприятия, а также его клиентами и поставщиками. Сети снижают потребность предприятий в других формах передачи информации, таких как телефон или обычная почта. Зачастую именно возможность организации электронной почты является одной из причин развертывания на предприятии вычислительной сети. Все большее распространение получают новые технологии, позволяющие передавать по сетевым каналам связи не только компьютерные данные, но голосовую и видеинформацию. Корпоративная сеть, которая интегрирует данные и мультимедийную информацию, может использоваться для организации аудио- и видеоконференций, кроме того, на ее основе может быть создана собственная внутренняя телефонная сеть.

Конечно, вычислительным сетям присущи и свои проблемы. Эти проблемы в основном связаны с организацией эффективного взаимодействия отдельных частей распределенной системы.

Во-первых, это сложности, связанные с программным обеспечением — операционными системами и приложениями. Программирование для распределенных систем принципиально отличается от программирования для централизованных систем. Так, сетевая операционная система, выполняя в общем случае

все функции по управлению локальными ресурсами компьютера, сверх того решает многочисленные задачи по предоставлению сетевых сервисов. Разработка сетевых приложений осложняется из-за необходимости организовать совместную работу их частей, выполняющихся на разных машинах. Много забот доставляет обеспечение совместимости программного обеспечения, устанавливаемого в узлах сети.

Во-вторых, много проблем связано с транспортировкой сообщений по каналам связи между компьютерами. Основные задачи здесь — обеспечение надежности (чтобы передаваемые данные не терялись и не искажались) и производительности (чтобы обмен данными происходил с приемлемыми задержками). В структуре общих затрат на вычислительную сеть расходы на решение «транспортных вопросов» составляют существенную часть, в то время как в централизованных системах эти проблемы полностью отсутствуют.

В-третьих, это вопросы, связанные с обеспечением безопасности, которые гораздо сложнее решаются в вычислительной сети, чем в автономно работающем компьютере. В некоторых случаях, когда безопасность особенно важна, от использования сети лучше вообще отказаться.

Можно приводить еще много «за» и «против» использования сетей, но главным доказательством эффективности является бесспорный факт их повсеместного распространения. Трудно найти сколько-нибудь крупное предприятие, на котором не было хотя бы односегментной сети персональных компьютеров; все больше и больше появляется крупных сетей с сотнями рабочих станций и десятками серверов, некоторые большие организации и предприятия обзаводятся частными глобальными сетями, объединяющими их филиалы, удаленные на тысячи километров. В каждом конкретном случае для создания сети были свои резоны, но верно и общее утверждение: что-то в этих сетях все-таки есть.

Сети отделов

Сети отделов — это сети, которые используются сравнительно небольшой группой сотрудников, работающих в одном отделе предприятия. Эти сотрудники решают некоторые общие задачи, например ведут бухгалтерский учет или занимаются маркетингом. Считается, что отдел может насчитывать до 100-150 сотрудников.

Главной целью сети отдела является разделение локальных ресурсов, таких как приложения, данные, лазерные принтеры и модемы. Обычно сети отделов имеют один или два файловых сервера, не более тридцати пользователей (рис. 2.31) и не разделяются на подсети. В этих сетях локализуется большая часть трафика предприятия. Сети отделов обычно создаются на основе какой-либо одной сетевой технологии — Ethernet, Token Ring. Для такой сети характерен один или максимум два типа операционных систем. Небольшое количество пользователей делает возможным использование в сетях отделов одноранговых сетевых ОС, таких, например, как Windows 98.

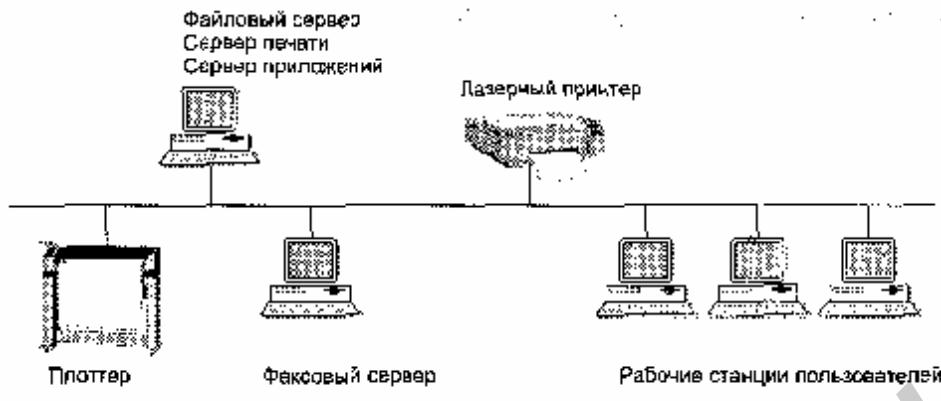


Рис. 2.31. Пример сети масштаба стада

Задачи управления сетью на уровне отдела относительно просты: добавление новых пользователей, устранение простых отказов, установка новых узлов и новых версий программного обеспечения. Такой сетью может управлять сотрудник, посвящающий обязанностям администратора только часть своего времени. Чаще всего администратор сети отдела не имеет специальной подготовки, но является тем человеком в отделе, который лучше всех разбирается в компьютерах, и само собой получается так, что он занимается администрированием сети.

Существует и другой тип сетей, близкий к сетям отделов, — сети рабочих групп. К таким сетям относят совсем небольшие сети, включающие до 10-20 компьютеров. Характеристики сетей рабочих групп практически не отличаются от описанных выше характеристик сетей отделов. Такие свойства, как простота сети и однородность, здесь проявляются в наибольшей степени, в то время как сети отделов могут приближаться в некоторых случаях к следующему по масштабу типу сетей — сетям кампусов.

Сети кампусов

Сети кампусов получили свое название от английского слова *campus* — студенческий городок. Именно на территории университетских городков часто возникала необходимость объединения нескольких мелких сетей в одну большую сеть. Сейчас это название не связывают со студенческими городками, а используют для обозначения сетей любых предприятий и организаций.

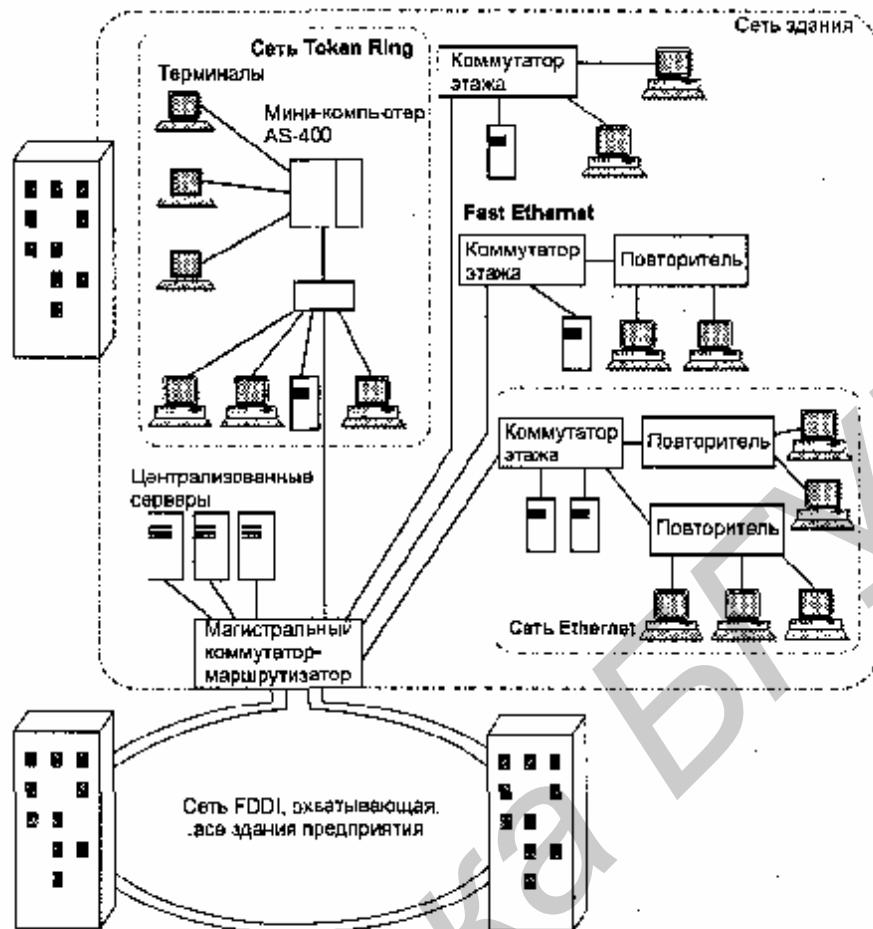


Рис. 2.32. Пример сети кампуса.

Главными особенностями сетей кампусов является то, что они объединяют множество сетей различных отделов одного предприятия в пределах отдельного здания или в пределах одной территории, покрывающей площадь в несколько квадратных километров (рис. 2.32). При этом глобальные соединения в сетях кампусов не используются. Службы такой сети включают взаимодействие между сетями отделов, доступ к общим базам данных предприятия, доступ к общим факс-серверам, высокоскоростным модемам и высокоскоростным принтерам. В результате сотрудники каждого отдела предприятия получают доступ к некоторым файлам и ресурсам сетей других отделов. Важной службой, предоставляемой сетями кампусов, стал доступ к корпоративным базам данных независимо от того, на каких типах компьютеров они располагаются.

Именно на уровне сети кампуса возникают проблемы интеграции неоднородного аппаратного и программного обеспечения. Типы компьютеров, сетевых операционных систем, сетевого аппаратного обеспечения могут отличаться в каждом отделе. Отсюда вытекают сложности управления сетями кампусов. Администраторы должны быть в этом случае более квалифицированными, а средства оперативного управления сетью — более совершенными.

Сети масштаба предприятия

Корпоративные сети называют также сетями масштаба предприятия, что соответствует дословному переводу термина «enterprise-wide networks», используемого в англоязычной литературе для обозначения этого типа сетей. Сети масштаба предприятия (корпоративные сети) объединяют большое количество компьютеров на всех территориях отдельного предприятия. Они могут быть сложно связаны и покрывать город, регион или даже континент. Число пользователей и компьютеров может измеряться тысячами, а число серверов — сотнями, расстояния между сетями отдельных территорий могут оказаться такими, что использование глобальных связей становится необходимым (рис. 2.33). Для соединения удаленных локальных сетей и отдельных компьютеров в корпоративной сети применяются разнообразные телекоммуникационные средства, в том числе телефонные каналы, радиоканалы, спутниковая связь. Корпоративную сеть можно представить в виде «островков» локальных сетей, «плавающих» в телекоммуникационной среде.

Непременным атрибутом такой сложной и крупномасштабной сети является высокая степень неоднородности (гетерогенности) — нельзя удовлетворить потребности тысяч пользователей с помощью однотипных программных и аппаратных средств. В корпоративной сети обязательно используются различные типы компьютеров — от майнфреймов до персоналок, несколько типов операционных систем и множество различных приложений. Неоднородные части корпоративной сети должны работать как единое целое, предоставляя пользователям по возможности удобный и простой доступ ко всем необходимым ресурсам.

Появление корпоративных сетей — это хорошая иллюстрация известного философского постулата о переходе количества в качество. При объединении отдельных сетей крупного предприятия, имеющего филиалы в разных городах и даже странах, в единую сеть многие количественные характеристики объединенной сети превосходят некоторый критический порог, за которым начинается новое качество. В этих условиях существующие методы и подходы к решению традиционных задач сетей меньших масштабов для корпоративных сетей оказались непригодными. На первый план вышли такие задачи и проблемы, которые в сетях рабочих групп, отделов и даже кампусов либо имели второстепенное значение, либо вообще не проявлялись. Примером может служить простейшая (для небольших сетей) задача — ведение учетных данных о пользователях сети.

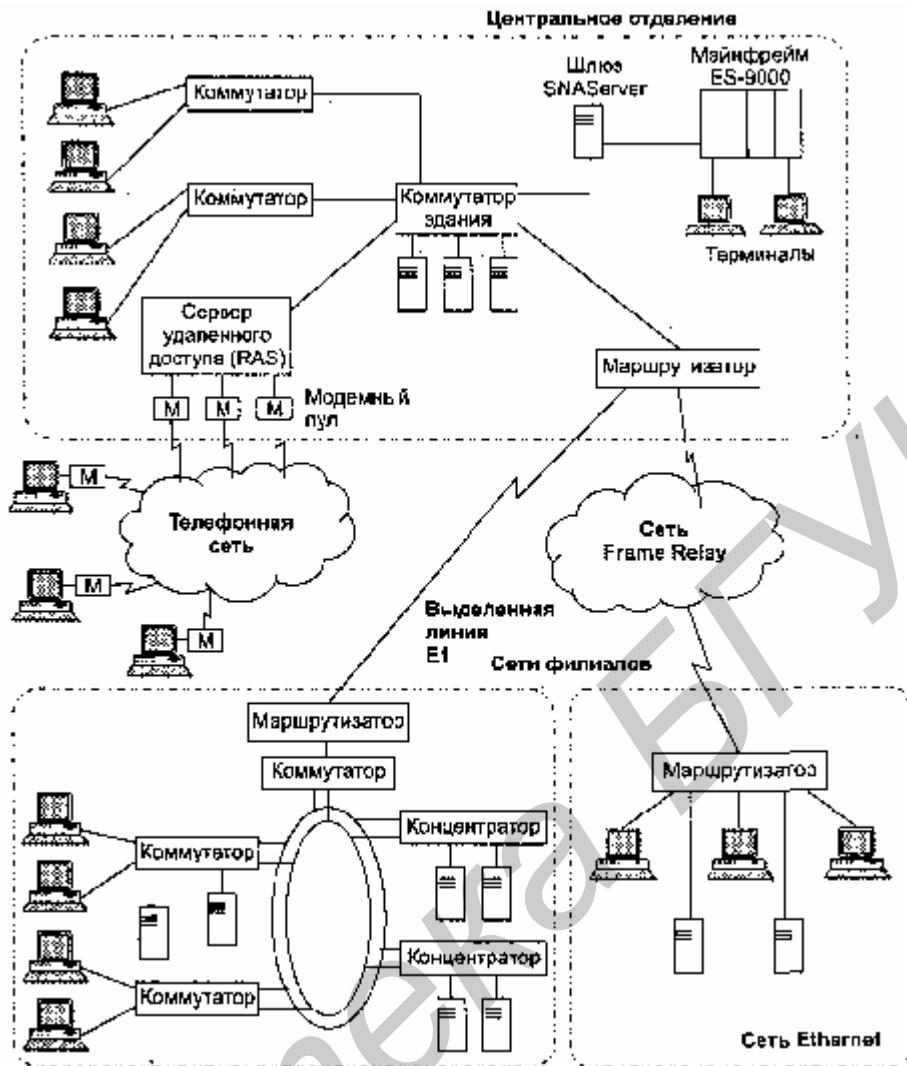


Рис. 2.33. Пример корпоративной сети

Наиболее простой способ ее решения — помещение учетных данных всех пользователей в локальную базу учетных данных каждого компьютера, к ресурсам которого эти пользователи должны иметь доступ. При попытке доступа данные извлекаются из локальной учетной базы и на их основе доступ предоставляется или не предоставляется. Для небольшой сети, состоящей из 5-10 компьютеров и примерно такого же количества пользователей, этот способ работает очень хорошо. Но если в сети насчитывается несколько тысяч пользователей, каждому из которых нужен доступ к нескольким десяткам серверов, то, очевидно, это решение становится крайне неэффективным. Администратор должен повторить несколько десятков раз (по числу серверов) операцию занесения учетных данных каждого пользователя. Сам пользователь также вынужден повторять процедуру логического входа каждый раз, когда ему нужен доступ к ресурсам нового сервера. Хорошее решение этой проблемы для крупной сети — использование централизованной справочной службы, в базе которой хранятся учетные записи всех пользователей сети. Администратор один раз выполняет операцию занесения данных пользователя в эту базу, а пользователь один раз выполняет процедуру логического входа, причем не в отдельный сервер, а в

сеть целиком.

При переходе от более простого типа сетей к более сложному — от сетей отдела к корпоративной сети — географические расстояния увеличиваются, поддержание связи компьютеров становится все более сложным и дорогостоящим. По мере увеличения масштабов сети повышаются требования к ее надежности, производительности и функциональным возможностям. По сети циркулируют все возрастающие объемы данных, и сеть должна обеспечивать их безопасность и защищенность наряду с доступностью. Все это приводит к тому, что корпоративные сети строятся на основе наиболее мощного и разнообразного оборудования и программного обеспечения.

Выводы

- Корпоративная сеть — это сеть, главным назначением которой является поддержание работы конкретного предприятия, владеющего данной сетью. Пользователями корпоративной сети являются только сотрудники данного предприятия.
- Использование вычислительных сетей дает предприятию следующие возможности:
 - разделение дорогостоящих ресурсов;
 - совершенствование коммуникаций;
 - улучшение доступа к информации;
 - быстрое и качественное принятие решений;
 - свобода в территориальном размещении компьютеров.
- В зависимости от масштаба производственного подразделения, в пределах которого действует сеть, различают сети отделов, сети кампусов и сети предприятий.
- Сети отделов используются небольшой группой сотрудников в основном с целью разделения дорогостоящих периферийных устройств, приложений и данных; имеют один-два файловых сервера и не более 30 пользователей; обычно не разделяются на подсети; создаются на основе какой-либо одной сетевой технологии; могут работать на базе одноранговых сетевых ОС.
- Сети кампусов объединяют сети отделов в пределах отдельного здания или одной территории площадью в несколько квадратных километров, при этом глобальные соединения не используются. На уровне сети кампуса возникают проблемы интеграции и управления неоднородным аппаратным и программным обеспечением.
- Сети предприятий (корпоративные сети) объединяют большое количество компьютеров на всех территориях отдельного предприятия. Для корпоративной сети характерны:
 - масштабность — тысячи пользовательских компьютеров, сотни серверов, огромные объемы хранимых и передаваемых по линиям связи данных, множество разнообразных приложений;
 - высокая степень гетерогенности — типы компьютеров, коммуникационного оборудования, операционных систем и приложений различны;

- использование глобальных связей — сети филиалов соединяются с помощью телекоммуникационных средств, в том числе телефонных каналов, радиоканалов, спутниковой связи.

Требования к компьютерным сетям

Главным требованием, предъявляемым к сетям, является выполнение сетью того набора услуг, для оказания которых она предназначена: например, предоставление доступа к файловым архивам или страницам публичных web-сайтов, обмен электронной почтой в пределах предприятия или в глобальных масштабах, интерактивный обмен речевыми сообщениями IP-телефонии и т. п. Все остальные требования — производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость — связаны с качеством выполнения основной задачи. И хотя все эти требования весьма важны, часто понятие «качество обслуживания» (Quality of Service, QoS) компьютерной сети трактуется более узко — в него включаются только две самые важные характеристики сети — производительность и надежность.

Производительность

Потенциально высокая производительность — это одно из основных преимуществ распределенных систем, к которым относятся компьютерные сети. Это свойство обеспечивается принципиальной, но к сожалению не всегда практически реализуемой возможностью распараллеливания работ между несколькими компьютерами сети.

Существует несколько основных характеристик производительности сети:

- время реакции;
- скорость передачи данных;
- задержка передачи и вариация задержки передачи.

Время реакции сети является интегральной характеристикой производительности сети с точки зрения пользователя. Именно эту характеристику имеет в виду пользователь, когда говорит: «Сегодня сеть работает медленно».

В общем случае время реакции определяется как интервал времени между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на этот запрос.

Очевидно, что значение этого показателя зависит от типа службы, к которой обращается пользователь, от того, какой пользователь и к какому серверу обращается, а также от текущего состояния элементов сети — загруженности сегментов, коммутаторов и маршрутизаторов, через которые проходит запрос, загруженности сервера и т. п.

Поэтому имеет смысл использовать также и средневзвешенную оценку времени реакции сети, усредняя этот показатель по пользователям, серверам и времени дня (от которого в значительной степени зависит загрузка сети).

Время реакции сети обычно складывается из нескольких составляющих. В общем случае в него входит время подготовки запросов на клиентском компьютере, время передачи запросов между клиентом и сервером через сегменты сети и промежуточное коммуникационное оборудование, время обработки запросов на сервере, время передачи ответов от сервера клиенту и время обработки получаемых от сервера ответов на клиентском компьютере.

Ясно, что разложение времени реакции на составляющие не интересует пользователя — ему важен конечный результат, однако для сетевого специалиста очень важно выделить из общего времени реакции составляющие, соответствующие этапам собственно сетевой обработки данных, — передачу данных от клиента к серверу через сегменты сети и коммуникационное оборудование.

Знание сетевых составляющих времени реакции дает возможность оценить производительность отдельных элементов сети, выявить узкие места и в случае необходимости выполнить модернизацию сети для повышения ее общей производительности.

Скорость передачи данных отражает объем данных, переданных сетью или ее частью в единицу времени. Пропускная способность уже не является пользовательской характеристикой, так как она говорит о скорости выполнения внутренних операций сети — передаче пакетов данных между узлами сети через различные коммуникационные устройства. Зато она непосредственно характеризует качество выполнения основной функции сети — транспортировки сообщений — и поэтому чаще используется при анализе производительности сети, чем время реакции.

Скорость передачи данных измеряется либо в битах в секунду, либо в пакетах в секунду. Пропускная способность может быть мгновенной, максимальной и средней.

Средняя скорость вычисляется путем деления общего объема переданных данных на время их передачи, причем выбирается достаточно длительный промежуток времени — час, день или неделя.

Мгновенная скорость отличается от средней тем, что для усреднения выбирается очень маленький промежуток времени — например, 10 мс или 1 с.

Максимальная скорость ~ это наибольшая мгновенная пропускная способность, зафиксированная в течение периода наблюдения. Максимально достижимая скорость передачи данных называется пропускной способностью элемента сети.

Чаще всего при проектировании, настройке и оптимизации сети используются такие показатели, как средняя и максимальная пропускные способности. Средняя пропускная способность отдельного элемента или всей сети позволяет оценить работу сети на большом промежутке времени, в течение которого в силу закона больших чисел пики и спады интенсивности трафика компенсируют друг друга. Пропускная способность позволяет оценить возможности сети справляться с пиковыми нагрузками, характерными для особых периодов работы сети, например утренних часов, когда сотрудники предприятия почти одновременно регистрируются в сети и обращаются к разделяемым файлам и базам данных.

Пропускную способность можно измерять между любыми двумя узлами или точками сети, например между клиентским компьютером и сервером, между входным и выходным портами маршрутизатора. Для анализа и настройки сети очень полезно знать данные о пропускной способности отдельных элементов сети.

Важно отметить, что из-за последовательного характера передачи данных различными элементами сети общая пропускная способность сети любого составного пути в сети будет равна минимальной из пропускных способностей составляющих элементов маршрута. Для повышения пропускной способности составного пути необходимо в первую очередь обратить внимание на самые медленные элементы. Иногда полезно оперировать общей пропускной способностью сети, которая определяется как максимальное количество информации, переданной между всеми узлами сети в единицу времени. Этот показатель характеризует качество сети в целом, не дифференцируя его по отдельным сегментам или устройствам.

Обычно при определении пропускной способности сегмента или устройства в передаваемых данных не выделяется трафик какого-то определенного пользователя, приложения или компьютера — подсчитывается общий объем передаваемой информации. Тем не менее для более точной оценки качества обслуживания такая детализации желательна, и в последнее время системы управления сетями все чаще позволяют ее выполнять.

Задержка передачи определяется как задержка между моментом поступления данных на вход какого-либо сетевого устройства или части сети и моментом появления их на выходе этого устройства. Этот параметр производительности по смыслу близок ко времени реакции сети, но отличается тем, что всегда характеризует только сетевые этапы обработки данных, без задержек обработки конечными узлами сети. Обычно качество сети характеризуют величинами максимальной задержки передачи и вариацией задержки. Не все типы трафика чувствительны к задержкам передачи, во всяком случае, к тем величинам задержек, которые характерны для компьютерных сетей, — обычно задержки не превышают сотен миллисекунд, реже — нескольких секунд. Такого порядка задержки пакетов, порождаемых файловой службой, службой электронной почты или службой печати, мало влияют на качество этих служб с точки зрения пользователя сети. С другой стороны, такие же задержки пакетов, переносящих голосовые данные или видеоизображение, могут приводить к значительному снижению качества предоставляемой пользователю информации — возникновению эффекта «эха», невозможности разобрать некоторые слова, дрожание изображения и т. п.

Пропускная способность и задержки передачи являются независимыми параметрами, так что сеть может обладать, например, высокой пропускной способностью, но вносить значительные задержки при передаче каждого пакета. Пример такой ситуации дает канал связи, образованный геостационарным спутником. Пропускная способность этого канала может быть весьма высокой, например 2 Мбит/с, в то время как задержка передачи всегда составляет не менее 0,24 с, что определяется скоростью распространения электрического сигнала (около

300 000 км/с) и длиной канала (72 000 км).

Надежность и безопасность

Одной из первоначальных целей создания распределенных систем, к которым относятся и вычислительные сети, являлось достижение большей надежности по сравнению с отдельными вычислительными машинами.

Важно различать несколько аспектов надежности. Для технических устройств используются такие показатели надежности, как среднее время наработки на отказ, вероятность отказа, интенсивность отказов. Однако эти показатели пригодны только для оценки надежности простых элементов и устройств, которые могут находиться только в двух состояниях — работоспособном или неработоспособном. Сложные системы, состоящие из многих элементов, кроме состояний работоспособности и неработоспособности, могут иметь и другие промежуточные состояния, которые эти характеристики не учитывают. В связи с этим для оценки надежности сложных систем применяется другой набор характеристик.

Готовность, или коэффициент готовности (availability), означает долю времени, в течение которого система может быть использована. Готовность может быть улучшена путем введения избыточности в структуру системы: ключевые элементы системы должны существовать в нескольких экземплярах, чтобы при отказе одного из них функционирование системы обеспечивали другие.

Чтобы компьютерную систему можно было отнести к высоконадежным, она должна как минимум обладать высокой готовностью, но этого недостаточно. Необходимо обеспечить сохранность данных и защиту их от искажений. Кроме этого, должна поддерживаться согласованность (непротиворечивость) данных, например, если для повышения надежности на нескольких файловых серверах хранится несколько копий данных, то нужно постоянно обеспечивать их идентичность.

Так как сеть работает на основе механизма передачи пакетов между конечными узлами, то одной из характерных характеристик надежности является вероятность доставки пакета узлу назначения без искажений. Наряду с этой характеристикой могут использоваться и другие показатели: вероятность потери пакета (по любой из причин — из-за переполнения буфера маршрутизатора, из-за несовпадения контрольной суммы, из-за отсутствия работоспособного пути к узлу назначения и т. д.), вероятность искажения отдельного бита передаваемых данных, отношение потерянных пакетов к доставленным.

Другим аспектом общей надежности является безопасность (security), то есть способность системы защитить данные от несанкционированного доступа. В распределенной системе это сделать гораздо сложнее, чем в централизованной. В сетях сообщения передаются по линиям связи, часто проходящим через общедоступные помещения, в которых могут быть установлены средства прослушивания линий. Другим уязвимым местом могут быть оставленные без присмотра персональные компьютеры. Кроме того, всегда имеется потенциальная угроза взлома защиты сети от неавторизованных пользователей, если сеть име-

ет выходы в глобальные сети общего пользования.

Еще одной характеристикой надежности является отказоустойчивость (fault tolerance). В сетях под отказоустойчивостью понимается способность системы скрывать от пользователя отказ отдельных ее элементов. Например, если копии таблицы базы данных хранятся одновременно на нескольких файловых серверах, то пользователи могут просто не заметить отказ одного из них. В отказоустойчивой системе отказ одного из ее элементов приводит к некоторому снижению качества ее работы (деградации), а не к полному останову. Так, при отказе одного из файловых серверов в предыдущем примере увеличивается только время доступа к базе данных из-за уменьшения степени распараллеливания запросов, но в целом система продолжает выполнять свои функции.

Расширяемость и масштабируемость

Термины расширяемость и масштабируемость иногда используют как синонимы, но это неверно — каждый из них имеет четко определенное самостоятельное значение.

Расширяемость (extensibility) означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, служб), наращивания длины сегментов сети и замены существующей аппаратуры более мощной. При этом принципиально важно, что легкость расширения системы иногда может обеспечиваться в некоторых весьма ограниченных пределах. Например, локальная сеть Ethernet, построенная на основе одного сегмента толстого коаксиального кабеля, обладает хорошей расширяемостью, в том смысле, что позволяет легко подключать новые станции. Однако такая сеть имеет ограничение на число станций — оно не должно превышать 30-40. Хотя сеть допускает физическое подключение к сегменту и большего числа станций (до 100), но при этом чаще всего резко снижается производительность сети. Наличие такого ограничения и является признаком плохой масштабируемости системы при хорошей расширяемости.

Масштабируемость (scalability) означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается. Для обеспечения масштабируемости сети приходится применять дополнительное коммуникационное оборудование и специальным образом структурировать сеть. Например, хорошей масштабируемостью обладает многосегментная сеть,строенная с использованием коммутаторов и маршрутизаторов и имеющая иерархическую структуру связей. Такая сеть может включать несколько тысяч компьютеров и при этом обеспечивать каждому пользователю сети нужное качество обслуживания.

Прозрачность

Прозрачность (transparency) сети достигается в том случае, когда сеть представляется пользователям не как множество отдельных компьютеров, связанных

между собой сложной системой кабелей, а как единая традиционная вычислительная машина с системой разделения времени. Известный лозунг компании Sun Microsystems: «Сеть — это компьютер» — говорит именно о такой прозрачной сети.

Прозрачность может быть достигнута на двух различных уровнях — на уровне пользователя и на уровне программиста. На уровне пользователя прозрачность означает, что для работы с удаленными ресурсами он использует те же команды и привычные ему процедуры, что и для работы с локальными ресурсами. На программном уровне прозрачность заключается в том, что приложению для доступа к удаленным ресурсам требуются те же вызовы, что и для доступа к локальным ресурсам. Прозрачность на уровне пользователя достигается проще, так как все особенности процедур, связанные с распределенным характером системы, маскируются от пользователя программистом, который создает приложение. Прозрачность на уровне приложения требует скрытия всех деталей распределенности средствами сетевой операционной системы.

Сеть должна скрывать все особенности операционных систем и различия в типах компьютеров. Пользователь компьютера Macintosh должен иметь возможность обращаться к ресурсам, поддерживаемым UNIX-системой, а пользователь UNIX должен иметь возможность разделять информацию с пользователями Windows 95. Подавляющее число пользователей ничего не хочет знать о внутренних форматах файлов или о синтаксисе команд UNIX. Пользователь терминала IBM 3270 должен иметь возможность обмениваться сообщениями с пользователями сети персональных компьютеров без необходимости вникать в секреты трудно запоминаемых адресов.

Концепция прозрачности может быть применена к различным аспектам сети. Например, прозрачность расположения означает, что от пользователя не требуется знаний о месте расположения программных и аппаратных ресурсов, таких как процессоры, принтеры, файлы и базы данных. Имя ресурса не должно включать информацию о месте его расположения, поэтому имена типа `mashinel: prog.c` или `\\\ftp_serv\pub` прозрачными не являются. Аналогично, прозрачность перемещения означает, что ресурсы должны свободно перемещаться из одного компьютера в другой без изменения своих имен. Еще одним из возможных аспектов прозрачности является прозрачность параллелизма, заключающаяся в том, что процесс распараллеливания вычислений происходит автоматически, без участия программиста, при этом система сама распределяет параллельные ветви приложения по процессорам и компьютерам сети. В настоящее время нельзя сказать, что свойство прозрачности в полной мере присуще многим вычислительным сетям, это скорее цель, к которой стремятся разработчики современных сетей.

Поддержка разных видов трафика

Компьютерные сети изначально предназначены для совместного доступа пользователя к ресурсам компьютеров: файлам, принтерам и т. п. Трафик, создаваемый этими традиционными службами компьютерных сетей, имеет свои осо-

бенности и существенно отличается от трафика сообщений в телефонных сетях или, например, в сетях кабельного телевидения. Однако 90-е годы стали годами проникновения в компьютерные сети трафика мультимедийных данных, представляющих в цифровой форме речь и видеоизображение. Компьютерные сети стали использоваться для организации видеоконференций, обучения и развлечения на основе видеофильмов и т. п. Естественно, что для динамической передачи мультимедийного трафика требуются иные алгоритмы и протоколы и, соответственно, другое оборудование. Хотя доля мультимедийного трафика пока невелика, он уже начал свое проникновение как в глобальные, так и локальные сети, и этот процесс, очевидно, будет продолжаться с возрастающей скоростью. Главной особенностью трафика, образующегося при динамической передаче голоса или изображения, является наличие жестких требований к синхронности передаваемых сообщений. Для качественного воспроизведения непрерывных процессов, которыми являются звуковые колебания или изменения интенсивности света в видеоизображении, необходимо получение измеренных и закодированных амплитуд сигналов с той же частотой, с которой они были измерены на передающей стороне. При запаздывании сообщений будут наблюдаться искажения.

В то же время трафик компьютерных данных характеризуется крайне неравномерной интенсивностью поступления сообщений в сеть при отсутствии жестких требований к синхронности доставки этих сообщений. Например, доступ пользователя, работающего с текстом на удаленном диске, порождает случайный поток сообщений между удаленным и локальным компьютерами, зависящий от действий пользователя по редактированию текста, причем задержки при доставке в определенных (и достаточно широких с компьютерной точки зрения) пределах мало влияют на качество обслуживания пользователя сети. Все алгоритмы компьютерной связи, соответствующие протоколы и коммуникационное оборудование были рассчитаны именно на такой «пульсирующий» характер трафика, поэтому необходимость передавать мультимедийный трафик требует внесения принципиальных изменений как в протоколы, так и в оборудование. Сегодня практически все новые протоколы в той или иной степени предоставляют поддержку мультимедийного трафика.

Особую сложность представляет совмещение в одной сети традиционного компьютерного и мультимедийного трафика. Передача исключительно мультимедийного трафика компьютерной сетью хотя и связана с определенными сложностями, но вызывает меньшие трудности. А вот случай сосуществования двух типов трафика с противоположными требованиями к качеству обслуживания является намного более сложной задачей. Обычно протоколы и оборудование компьютерных сетей относят мультимедийный трафик к факультативному, поэтому качество его обслуживания оставляет желать лучшего. Сегодня затрачиваются большие усилия по созданию сетей, не ущемляющих интересы каждого из типов трафика. Наиболее близки к этой цели сети на основе технологии ATM, разработчики которой изначально учитывали случай сосуществования разных типов трафика в одной сети.

Управляемость

Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети. В идеале средства управления сетями представляют собой систему, осуществляющую наблюдение, контроль и управление каждым элементом сети — от простейших до самых сложных устройств, при этом такая система рассматривает сеть как единое целое, а не как разрозненный набор отдельных устройств.

Хорошая система управления наблюдает за сетью и, обнаружив проблему, активизирует определенное действие, исправляет ситуацию и уведомляет администратора о том, что произошло и какие шаги предприняты. Одновременно с этим система управления должна накапливать данные, на основании которых можно планировать развитие сети. Наконец, система управления должна быть независима от производителя и обладать удобным интерфейсом, позволяющим выполнять все действия с одной консоли.

Решая тактические задачи, администраторы и технический персонал сталкиваются с ежедневными проблемами обеспечения работоспособности сети. Эти задачи требуют быстрого решения, обслуживающий сеть персонал должен оперативно реагировать на сообщения о неисправностях, поступающих от пользователей или автоматических средств управления сетью. Постепенно становятся заметны более общие проблемы производительности, конфигурирования сети, обработки сбоев и безопасности данных, требующие стратегического подхода, то есть планирования сети. Планирование, кроме этого, включает прогноз изменений требований пользователей к сети, вопросы применения новых приложений, новых сетевых технологий и т. п.

Полезность системы управления особенно ярко проявляется в больших сетях: корпоративных или публичных глобальных. Без системы управления в таких сетях нужно присутствие квалифицированных специалистов по эксплуатации в каждом здании каждого города, где установлено оборудование сети, что в итоге приводит к необходимости содержания огромного штата обслуживающего персонала.

В настоящее время в области систем управления сетями много нерешенных проблем. Явно недостаточно действительно удобных, компактных и многопротокольных средств управления сетью. Большинство существующих средств во все не управляют сетью, а всего лишь осуществляют наблюдение за ее работой. Они следят за сетью, но не выполняют активных действий, если с сетью что-то произошло или может произойти. Мало масштабируемых систем, способных обслуживать как сети масштаба отдела, так и сети масштаба предприятия, — очень многие системы управляют только отдельными элементами сети и не анализируют способность сети выполнять качественную передачу данных между конечными пользователями сети.

Совместимость

Совместимость, или интегрируемость, означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение, то есть в ней могут сосуществовать различные операционные системы, поддерживающие разные стеки коммуникационных протоколов, а также аппаратные средства и приложения от разных производителей. Сеть, состоящая из разнотипных элементов, называется неоднородной, или гетерогенной, а если гетерогенная сеть работает без проблем, то она является интегрированной. Основной путь построения интегрированных сетей — использование модулей, выполненных в соответствии с открытыми стандартами и спецификациями.

Качество обслуживания

В общем случае качество обслуживания (Quality of Service, QoS) определяет вероятностные оценки выполнения тех или иных требований, предъявляемых к сети приложениями или пользователями. Например, при передаче голосового трафика через сеть под качеством обслуживания чаще всего понимают гарантии того, что голосовые пакеты будут доставляться сетью с задержкой не более N мс, при этом вариация задержки не превысит M мс, и эти характеристики станут выдерживаться сетью с вероятностью 0,95 на определенном временном интервале. То есть приложению, которое передает голосовой трафик, важно, чтобы сеть гарантировала соблюдение именно этого, приведенного выше набора характеристик качества обслуживания. Файловому сервису нужны гарантии средней полосы пропускания и расширения ее на небольших интервалах времени до некоторого максимального уровня для быстрой передачи пульсаций. В идеале сеть должна гарантировать особые параметры качества обслуживания, сформулированные для каждого отдельного приложения. Однако по понятным причинам разрабатываемые и уже существующие механизмы QoS ограничиваются решением более простой задачи — гарантированием неких усредненных требований, заданных для основных типов приложений.

Чаще всего параметры, фигурирующие в разнообразных определениях качества обслуживания, регламентируют следующие показатели работы сети:

- скорость передачи данных;
- задержки передачи пакетов;
- уровень потерь и искажений пакетов.

Качество обслуживания гарантируется для некоторого потока данных. Напомним, что поток данных — это последовательность пакетов, имеющих некоторые общие признаки, например адрес узла-источника, информация, идентифицирующая тип приложения (номер порта TCP/UDP) и т. п. К потокам применимы такие понятия, как агрегирование и дифференцирование. Так, поток данных от одного компьютера может быть представлен как совокупность потоков от разных приложений, а потоки от компьютеров одного предприятия агрегированы в один поток данных абонента некоторого поставщика услуг.

Механизмы поддержки качества обслуживания сами по себе не создают пропускной способности. Сеть не может дать больше того, что имеет. Так что фактическая пропускная способность каналов связи и транзитного коммуникационного оборудования — это ресурсы сети, являющиеся отправной точкой для работы механизмов QoS. Механизмы QoS только управляют распределением имеющейся пропускной способности в соответствии с требованиями приложений и настройками сети. Самый очевидный способ перераспределения пропускной способности сети состоит в управлении очередями пакетов.

Поскольку данные, которыми обмениваются два конечных узла, проходят через некоторое количество промежуточных сетевых устройств, таких как концентраторы, коммутаторы и маршрутизаторы, то поддержка QoS требует взаимодействия всех сетевых элементов на пути трафика, то есть «из конца в конец» (*«end-to-end»*, e2e). Любые гарантии QoS настолько хороши, насколько их обеспечивает наиболее «слабый» элемент в цепочке между отправителем и получателем. Поэтому нужно хорошо понимать, что поддержка QoS только в одном сетевом устройстве, пусть даже и магистральном, может весьма незначительно улучшить качество обслуживания или же совсем не повлиять на параметры QoS.

Реализация в компьютерных сетях механизмов поддержки QoS является сравнительно новой тенденцией. Долгое время компьютерные сети существовали без таких механизмов, и это объясняется в основном двумя причинами. Во-первых, большинство приложений, выполняемых в сети, были нетребовательными. Для таких приложений задержки пакетов или отклонения средней пропускной способности в достаточно широком диапазоне не приводили к значительной потере функциональности. Примерами нетребовательных приложений являются наиболее распространенные в сетях 80-х годов приложения электронной почты или удаленного копирования файлов.

Во-вторых, сама пропускная способность 10-мегабитных сетей Ethernet во многих случаях не являлась дефицитом. Так, разделяемый сегмент Ethernet, к которому было подключено 10-20 компьютеров, изредка копирующих небольшие текстовые файлы, не превышающие несколько сотен килобайт, позволял трафику каждой пары взаимодействующих компьютеров пересекать сеть так быстро, как это требовалось породившим этот трафик приложениям.

В результате большинство сетей работало с тем качеством транспортного обслуживания, которое обеспечивало потребности приложений. Правда, никаких гарантий относительно нахождения задержек пакетов или пропускной способности, с которой пакеты передаются между узлами, в определенных пределах эти сети не давали. Более того, при временных перегрузках сети, когда значительная часть компьютеров одновременно начинала передавать данные с максимальной скоростью, задержки и пропускная способность становились такими, что работа приложений давала сбой — шла слишком медленно, с прерываниями сеансов и т. п.

Транспортный сервис, который предоставляли такие сети, получил название *best effort*, то есть сервис «с максимальными усилиями». Сеть старается обработать поступающий трафик как можно быстрее, но при этом никаких гарантий

относительно результата своих усилий не даст. Примерами являются большинство популярных технологий, разработанных в 80-е годы: Ethernet, Token Ring, IP, X.25. Сервис «с максимальными усилиями» основан на некотором справедливом алгоритме обработки очередей, возникающих при перегрузках сети, когда в течение некоторого времени скорость поступления пакетов в сеть превышает скорость продвижения этих пакетов. В простейшем случае алгоритм обработки очереди рассматривает пакеты всех потоков как равноправные и пропускает их в порядке поступления (First Input First Output, FIFO). В том случае, когда очередь становится слишком большой (не умещается в буфере), проблема решается простым отбрасыванием вновь поступающих пакетов.

Очевидно, что сервис «с максимальными усилиями» обеспечивает приемлемое качество обслуживания только в тех случаях, когда производительность сети намного превышает средние потребности, то есть является избыточной. В такой сети пропускная способность достаточна даже для поддержания трафика пиковых периодов нагрузки. Также очевидно, что такое решение не экономично — по крайней мере, по отношению к пропускным способностям сегодняшних технологий и инфраструктур, особенно для глобальных сетей. Так как пиковые нагрузки и области, где они возникают, трудно предсказать, то такой путь не дает долговременного решения.

Тем не менее построение сетей с избыточной пропускной способностью, будучи самым простым способом обеспечения нужного уровня качества обслуживания, иногда применяется на практике. Например, некоторые поставщики сетевых услуг TCP/IP предоставляют гарантию качественного обслуживания, постоянно поддерживая определенный уровень превышения пропускной способности своих магистралей над потребностями своих клиентов.

В условиях, когда многие механизмы поддержания качества обслуживания только разрабатываются, использование для этих целей избыточной пропускной способности часто оказывается единственным возможным, хотя и времененным решением.

Выводы

- Главным требованием, предъявляемым к сетям, является выполнение сетью того набора услуг, для оказания которых она предназначена. Все остальные требования — производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость — связаны с качеством выполнения этой основной задачи.
- Существует два основных подхода к обеспечению качества работы сети. Первый состоит в том, что сеть гарантирует пользователю соблюдение некоторой числовой величины показателя качества обслуживания. Например, сети frame relay и ATM могут гарантировать пользователю заданный уровень пропускной способности. При втором подходе — «с максимальными усилиями» (best effort) — сеть старается по возможности более качественно обслужить пользователя, но ничего при этом не гарантирует.
- К основным характеристикам производительности сети относятся: время ре-

акции, которое определяется как время между возникновением, запроса к какому-либо сетевому сервису и получением ответа на него; пропускная способность, которая отражает объем данных, переданных сетью в единицу времени, и задержка передачи, которая равна интервалу между моментом поступления пакета на вход какого-либо сетевого устройства и моментом его появления на выходе этого устройства.

- Для оценки надежности сетей используются различные характеристики, в том числе: коэффициент готовности, означающий долю времени, в течение которого система может быть использована; безопасность, то есть способность системы защитить данные от несанкционированного доступа; отказоустойчивость — способность системы работать в условиях отказа некоторых ее элементов.
- Расширяемость означает возможность сравнительно легкого добавления наращивания длины сегментов сети и замены существующей аппаратуры более мощной.
- Масштабируемость означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается.
- Прозрачность — свойство сети скрывать от пользователя детали своего внутреннего устройства, упрощая тем самым его работу в сети.
- Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети.
- Совместимость означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение.
- Качество обслуживания (Quality of Service, QoS) определяет количественные оценки вероятности того, что сеть будет передавать определенный поток данных между двумя определенными узлами в соответствии с потребностями приложения или пользователя.

ГЛАВА 3 Открытые системы и модель OSI

Универсальный тезис о пользе стандартизации, справедливый для всех отраслей, в компьютерных сетях приобретает особое значение. Суть сети — это соединение разного оборудования, а значит, проблема совместимости является одной из наиболее острых. Без принятия всеми производителями общепринятых правил построения оборудования прогресс в деле «строительства» сетей был бы невозможен. Поэтому все развитие компьютерной отрасли в конечном счете отражено в стандартах — любая новая технология только тогда приобретает «законный» статус, когда ее содержание закрепляется в соответствующем стандарте.

В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия. Именно на основе этого подхода была разработана стандартная семиуровневая модель взаимодействия открытых систем, ставшая своего рода универсальным языком сетевых специалистов.

Многоуровневый подход

Декомпозиция задачи сетевого взаимодействия

Организация взаимодействия между устройствами и сети является сложной задачей. Как известно, для решения сложных задач используется универсальный прием — декомпозиция, то есть разбиение одной сложной задачи на несколько более простых задач-модулей (рис. 3.1). Декомпозиция состоит в четком определении функций каждого модуля, а также порядка их взаимодействия (интерфейсов). В результате достигается логическое упрощение задачи, а кроме того, появляется возможность модификации отдельных модулей без изменения остальной части системы.

При декомпозиции часто используют многоуровневый подход. Он заключается в следующем. Все множество модулей, решающих частные задачи, разбивают на группы и упорядочивают по уровням, образующим иерархию. В соответствии с принципом иерархии для каждого промежуточного уровня можно указать непосредственно примыкающие к нему соседние вышележащий и нижележащий уровни (рис. 3.2). Группа модулей, составляющих каждый уровень, должна быть сформирована таким образом, чтобы все модули этой группы для выполнения своих задач обращались с запросами только к модулям соседнего нижележащего уровня. С другой стороны, результаты работы всех модулей, отнесенных к некоторому уровню, могут быть переданы только модулям соседнего вышележащего уровня. Такая иерархическая декомпозиция задачи предполагает четкое определение функции каждого уровня и интерфейсов между уровнями. Интерфейс определяет набор функций, которые нижележащий уровень предоставляет вышележащему. В результате иерархической декомпозиции дос-

тигается относительная независимость уровней, а значит, возможность их автономной разработки и модификации.

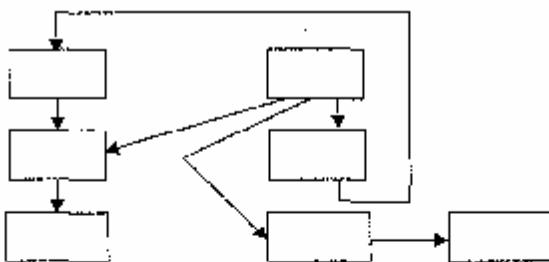


Рис. 3.1. Пример декомпозиции задачи

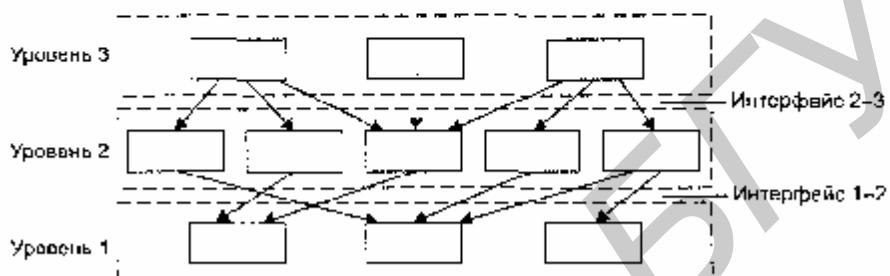


Рис. 3.2. Многоуровневый подход — создание иерархии задач

Средства решения задачи организации сетевого взаимодействия, конечно, тоже могут быть представлены в виде иерархически организованного множества модулей. Например, модулям нижнего уровня можно поручить вопросы, связанные с надежной передачей информации между двумя соседними узлами, а модулям следующего, более высокого уровня — транспортировку сообщений в пределах всей сети. Очевидно, что последняя задача — организация связи двух любых, не обязательно соседних узлов — является более общей и поэтому ее решение может быть получено путем многократных обращений к нижележащему уровню.

Так, связывание узлов А и В (рис. 3.3) может быть сведено к последовательному связыванию пар промежуточных смежных узлов. Таким образом, модули вышележащего уровня при решении своих задач рассматривают средства нижележащего уровня как инструмент.

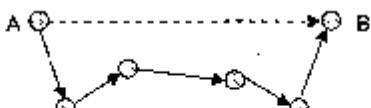


Рис. 3.3. Декомпозиция задачи связывания произвольной пары узлов на более частные задачи связывания пар соседних узлов

Протокол, интерфейс, стек протоколов

Многоуровневое представление средств сетевого взаимодействия имеет свою специфику, связанную с тем, что в процессе обмена сообщениями участвуют

две стороны, то есть в данном случае необходимо организовать согласованную работу двух «иерархий», работающих на разных компьютерах. Оба участника сетевого обмена должны принять множество соглашений. Например, они должны согласовать уровни и форму электрических сигналов, способ определения размера сообщений, договориться о методах контроля достоверности и т. п. Другими словами, соглашения должны быть приняты для всех уровней, начиная от самого низкого — уровня передачи битов — до самого высокого, реализующего сервис для пользователей сети.

На рис. 3.4 показана модель взаимодействия двух узлов. С каждой стороны средства взаимодействия представлены четырьмя уровнями. Процедура взаимодействия этих двух узлов может быть описана в виде набора правил взаимодействия каждой пары соответствующих уровней обеих участвующих сторон. Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются протоколом.



Рис. 3.4. Взаимодействие двух узлов

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с четко определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила принято называть интерфейсом. Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню. В сущности, протокол и интерфейс выражают одно и то же понятие, но традиционно в сетях за ними закрепили разные области действия: протоколы определяют правила взаимодействия модулей одного уровня в разных узлах, а интерфейсы — правила взаимодействия модулей соседних уровней в одном узле.

Средства каждого уровня должны отрабатывать, во-первых, свой собственный протокол, а во-вторых, интерфейсы с соседними уровнями.

Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется стеком коммуникационных протоколов.

Коммуникационные протоколы могут быть реализованы как программно, так и аппаратно. Протоколы нижних уровней часто реализуются комбинацией программных и аппаратных средств, а протоколы верхних уровней — как правило, чисто программными средствами.

Программный модуль, реализующий некоторый протокол, часто для краткости также называют «протоколом». При этом соотношение между протоколом — формально определенной процедурой и протоколом — программным модулем, реализующим эту процедуру, аналогично соотношению между алгоритмом решения некоторой задачи и программой, решающей эту задачу.

Понятно, что один и тот же алгоритм может быть запрограммирован с разной степенью эффективности. Точно так же и протокол может иметь несколько программных реализаций, Именно поэтому при сравнении протоколов следует учитывать не только логику их работы, но и качество программных решений. Более того, на эффективность взаимодействия устройств в сети влияет качество всей совокупности протоколов, составляющих стек, в частности, насколько рационально распределены функции между протоколами разных уровней и насколько хорошо определены интерфейсы между ними.

Протоколы реализуются не только компьютерами, но и другими сетевыми устройствами — концентраторами, мостами, коммутаторами, маршрутизаторами и т. д. Действительно, в общем случае связь компьютеров в сети осуществляется не напрямую, а через различные коммуникационные устройства. В зависимости от типа устройства в нем должны быть встроенные средства, реализующие тот или иной набор протоколов.

Чтобы еще раз пояснить понятия «протокол» и «интерфейс», рассмотрим пример, не имеющий отношения к вычислительным сетям, а именно обсудим взаимодействие двух предприятий А и В, связанных между собой деловым сотрудничеством. Между предприятиями существуют многочисленные договоренности и соглашения, такие, например, как регулярные поставки продукции одного предприятия другому. В соответствии с этой договоренностью начальник отдела продаж предприятия А регулярно в начале каждого месяца посылает официальное сообщение начальнику отдела закупок предприятия В о том, сколько и какого товара может быть поставлено в этом месяце, В ответ на это сообщение начальник отдела закупок предприятия В посылает в ответ заявку установленного образца на требуемое количество продукции. Возможно, процедура взаимодействия этих начальников включает дополнительные согласования, в любом случае существует установленный порядок взаимодействия, который можно считать «протоколом уровня начальников». Начальники посыпают свои сообщения и заявки через своих секретарей. Порядок взаимодействия начальника и секретаря соответствует понятию межуровневого интерфейса «начальник-секретарь». На предприятии А обмен документами между начальником и секретарем идет через специальную папку, а на предприятии В начальник общается с секретарем по факсу. Таким образом, интерфейсы «начальник-секретарь» на этих двух предприятиях отличаются.

После того как сообщения переданы секретарям, начальников не волнует, каким образом эти сообщения будут перемещаться дальше — обычной или электронной почтой, факсом или нарочным. Выбор способа передачи — это уровень компетенции секретарей, они могут решать данный вопрос, не уведомляя об этом своих начальников, так как их протокол взаимодействия связан только с передачей сообщений, поступающих сверху, и не касается содержания этих

сообщений. На рис. 3.5 показано, что в качестве протокола взаимодействия «секретарь-секретарь» используется обмен письмами. При решении других вопросов начальники могут взаимодействовать по другим правилам-протоколам, но это не влияет на работу секретарей, для которых не важно, какие сообщения отправлять, а важно, чтобы они дошли до адресата. Итак, в данном случае мы имеем дело с двумя уровнями — уровнем начальников и уровнем секретарей, и каждый из них имеет собственный протокол, который может быть изменен независимо от протокола другого уровня. Эта независимость протоколов друг от друга делает привлекательным многоуровневый подход.

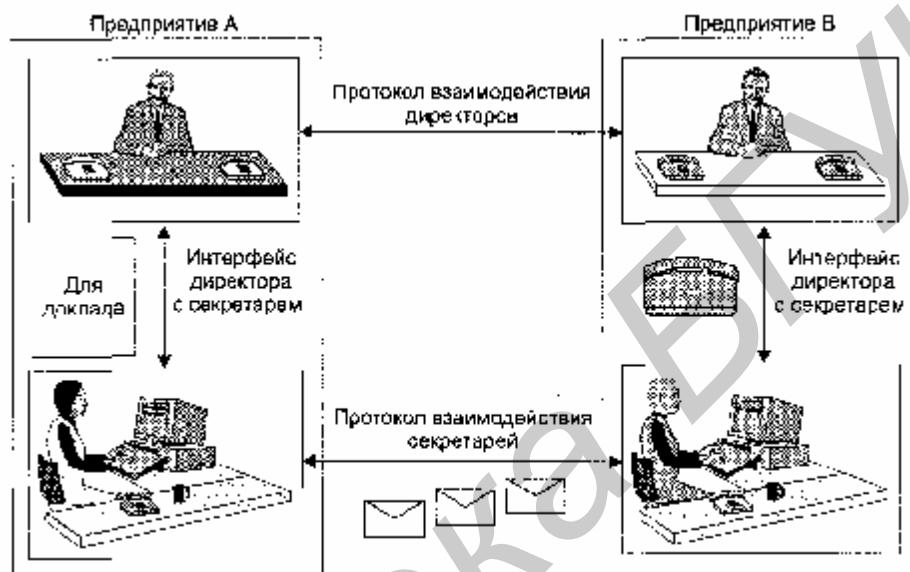


Рис. 3.5. Пример многоуровневого взаимодействия предприятий

Выводы

- В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия.
- Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются протоколом.
- Формализованные правила, определяющие взаимодействие сетевых компонентов соседних уровней одного узла, называются интерфейсом. Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню.
- Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется стеком коммуникационных протоколов.

Модель OSI

Общая характеристика модели OSI

Из того, что протокол является соглашением, принятым двумя взаимодействующими объектами, в данном случае двумя работающими в сети компьютерами, совсем не следует, что он обязательно является стандартным. Но на практике при реализации сетей стремятся использовать стандартные протоколы. Это могут быть фирменные, национальные или международные стандарты. В начале 80-х годов ряд международных организаций по стандартизации — ISO, ITU-T и некоторые другие — разработали модель, которая сыграла значительную роль в развитии сетей. Эта модель называется моделью взаимодействия открытых систем (Open System Interconnection, OSI) или моделью OSI. Модель OSI определяет различные уровни взаимодействия систем в сетях с коммутацией пакетов, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень. Модель OSI была разработана на основании большого опыта, полученного при создании компьютерных сетей, в основном глобальных, в 70-е годы. Полное описание этой модели занимает более 1000 страниц текста.

В модели OSI (рис. 3.6) средства взаимодействия делятся на семь уровней: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический. Каждый уровень имеет дело с одним определенным аспектом взаимодействия сетевых устройств.

Модель OSI описывает только системные средства взаимодействия, реализуемые операционной системой, системными утилитами, системными аппаратными средствами. Модель не включает средства взаимодействия приложений конечных пользователей. Свои собственные протоколы взаимодействия приложения реализуют, обращаясь к системным средствам. Поэтому необходимо различать уровень взаимодействия приложений и прикладной уровень.

Следует также иметь в виду, что приложение может взять на себя функции некоторых верхних уровней модели OSI. Например, некоторые СУБД имеют встроенные средства удаленного доступа к файлам. В этом случае приложение, выполняя доступ к удаленным ресурсам, не использует системную файловую службу; оно обходит верхние уровни модели OSI и обращается напрямую к системным средствам, ответственным за транспортировку сообщений по сети, которые располагаются на нижних уровнях модели OSI.

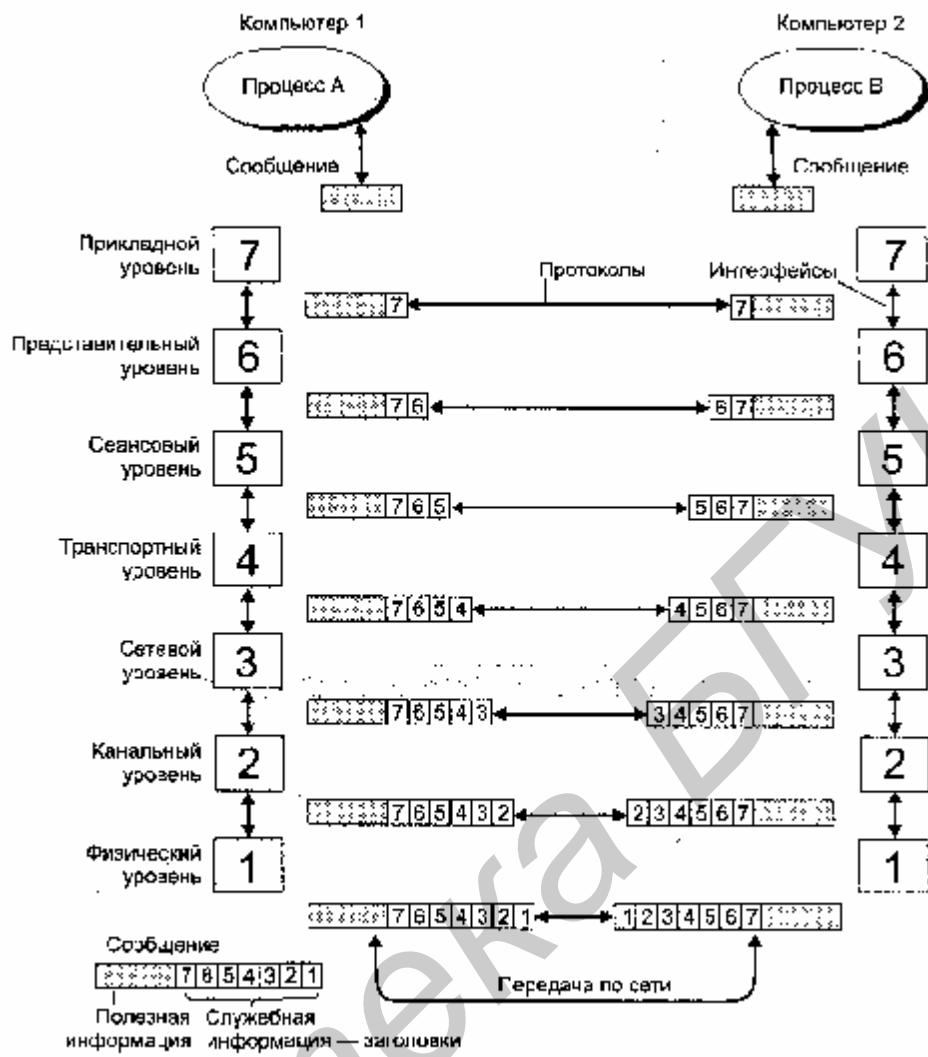


Рис. 3.6. Модель взаимодействия открытых систем ISO/OSI

Итак, пусть приложение обращается с запросом к прикладному уровню, например к файловой службе. На основании этого запроса программное обеспечение прикладного уровня формирует сообщение стандартного формата. Обычное сообщение состоит из заголовка и поля данных. Заголовок содержит служебную информацию, которую необходимо передать через сеть прикладному уровню машины-адресата, чтобы сообщить ему, какую работу надо выполнить. В нашем случае заголовок, очевидно, должен содержать информацию о месте нахождения файла и о типе операции, которую необходимо над ним выполнить. Поля данных сообщения может быть пустым или содержать какие-либо данные, например те, которые необходимо записать в удаленный файл. Но для того чтобы доставить эту информацию по назначению, предстоит решить еще много задач, ответственность за которые несут нижележащие уровни.

После формирования сообщения прикладной уровень направляет его вниз по стеку представительному уровню. Протокол представительского уровня на основании информации, полученной из заголовка, прикладного уровня, выполняет требуемые действия и добавляет к сообщению собственную служебную информацию — заголовок представительского уровня, в котором содержатся указания для протокола представительского уровня машины-адресата. Полученное

в результате сообщение передается вниз сеансовому уровню, который, в свою очередь, добавляет свой заголовок и т. д. (Некоторые реализации протоколов помещают служебную информацию не только в начале сообщения в виде заголовка, но и в конце, в виде так называемого «концевика».) Наконец, сообщение достигает нижнего, физического уровня, который собственно и передает его по линиям связи машине-адресату. К этому моменту сообщение «обрастает» заголовками всех уровней (рис. 3.7).

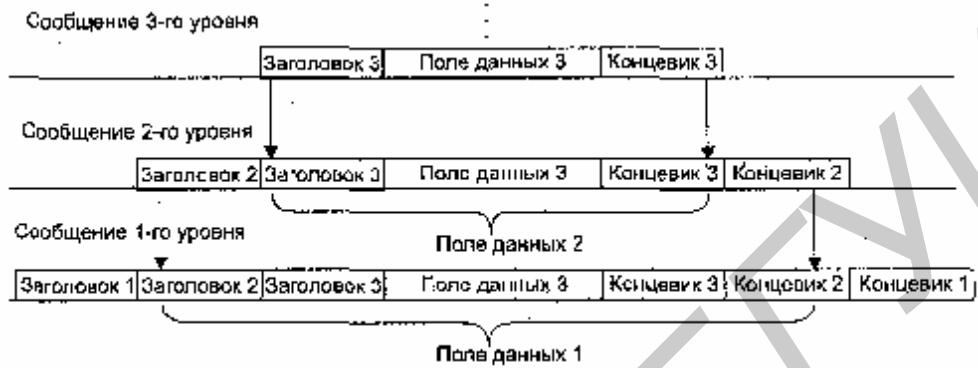


Рис. 3.7. Вложенность сообщений различных уровней

Когда сообщение по сети поступает на машину-адресат, оно принимается ее физическим уровнем и последовательно перемещается вверх с уровня на уровень. Каждый уровень анализирует и обрабатывает заголовок своего уровня, выполняя соответствующие данному уровню функции, а затем удаляет этот заголовок и передает сообщение вышестоящему уровню.

Наряду с термином сообщение (message) существуют и другие термины, применяемые сетевыми специалистами для обозначения единиц обмена данными в процедурах обмена. В стандартах ISO для обозначения единиц обмена данными, с которыми имеют дело протоколы разных уровней, используется общее название протокольный блок данных (Protocol Data Unit, PDU). Для обозначения блоков данных определенных уровней часто используются специальные названия: кадр (frame), пакет (packet), дейтаграмма (datagram), сегмент (segment).

Физический уровень

Физический уровень (Physical layer) имеет дело с передачей битов по физическим каналам связи, таким, например, как коаксиальный кабель, витая пара, оптоволоконный кабель или цифровой территориальный канал. К этому уровню имеют отношение характеристики физических сред передачи данных, такие как полоса пропускания, помехозащищенность, волновое сопротивление и др. На этом же уровне определяются характеристики электрических сигналов, передающих дискретную информацию, например крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов. Кроме этого, здесь стандартизуются типы разъемов и

назначение каждого контакта.

Функции физического уровня реализуются во всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом.

Примером протокола физического уровня может служить спецификация 10Base-T технологии Ethernet, которая определяет в качестве используемого кабеля неэкранированную витую пару категории 3 с волновым сопротивлением 100 Ом, разъем RJ-45, максимальную длину физического сегмента 100 м, манчестерский код для представления данных в кабеле, а также некоторые другие характеристики среды и электрических сигналов.

Канальный уровень

На физическом уровне просто пересылаются биты. При этом не учитывается, что в некоторых сетях, в которых линии связи используются (разделяются) по-переменно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня (Data Link Layer) является проверка доступности среды передачи. Другой задачей канального уровня является реализация механизмов обнаружения и коррекции ошибок. Для этого на канальном уровне биты группируются в наборы, называемые кадрами (frames). Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность битов в начало и конец каждого кадра для его выделения, а также вычисляет контрольную сумму, обрабатывая все байты кадра определенным способом и добавляя контрольную сумму к кадру. Когда кадр приходит по сети, получатель снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка. Канальный уровень может не только обнаруживать ошибки, но и исправлять их за счет повторной передачи поврежденных кадров. Необходимо отметить, что функция исправления ошибок не является обязательной для канального уровня, поэтому в некоторых протоколах этого уровня она отсутствует, например в Ethernet и frame relay.

В протоколах канального уровня, используемых в локальных сетях, заложена определенная структура связей между компьютерами и способы их адресации.

Хотя канальный уровень и обеспечивает доставку кадра между любыми двумя узлами локальной сети, он это делает только в сети с совершенно определенной топологией связей, именно той топологией, для которой он был разработан. К таким типовым топологиям, поддерживаемым протоколами канального уровня локальных сетей, относятся общая шина, кольцо и звезда, а также структуры, полученные из них с помощью мостов и коммутаторов. Примерами протоколов канального уровня являются протоколы Ethernet, Token Ring, FDDI.

В локальных сетях протоколы канального уровня используются компьютерами, мостами, коммутаторами и маршрутизаторами. В компьютерах функции канального уровня реализуются совместными усилиями сетевых адаптеров и их

драйверов.

В глобальных сетях, которые редко обладают регулярной топологией, канальный уровень часто обеспечивает обмен сообщениями только между двумя соседними компьютерами, соединенными индивидуальной линией связи. Примерами протоколов «точка-точка» (как часто называют такие протоколы) могут служить широко распространенные протоколы PPP и LAP-B. В таких случаях для доставки сообщений между конечными узлами через всю сеть используются средства сетевого уровня. Именно так организованы сети X.25. Иногда в глобальных сетях функции канального уровня в чистом виде выделить трудно, так как в одном и том же протоколе они объединяются с функциями сетевого уровня. Примерами такого подхода могут служить протоколы технологий ATM и frame relay.

В целом канальный уровень представляет собой весьма мощный и законченный набор функций по пересылке сообщений между узлами сети. В некоторых случаях протоколы канального уровня оказываются самодостаточными транспортными средствами и могут допускать работу поверх себя непосредственно протоколов прикладного уровня или приложений, без привлечения средств сетевого и транспортного уровней. Например, существует реализация протокола управления сетью SNMP непосредственно поверх Ethernet, хотя стандартно этот протокол работает поверх сетевого протокола IP и транспортного протокола UDP. Естественно, что применение такой реализации будет ограниченным — она не подходит для составных сетей разных технологий, например Ethernet и X.25, и даже для такой сети, в которой во всех сегментах применяется Ethernet, но между сегментами существуют петлевидные связи. А вот в двухсегментной сети Ethernet, объединенной мостом, реализация SNMP над канальным уровнем будет вполне работоспособной.

Тем не менее для обеспечения качественной транспортировки сообщений в сетях любых топологий и технологий функций канального уровня оказывается недостаточно, поэтому в модели OSI решение этой задачи возлагается на два следующих уровня — сетевой и транспортный.

Канальный уровень обеспечивает передачу пакетов данных, поступающих от протоколов верхних уровней, узлу назначения, адрес которого также указывает протокол верхнего уровня. Протоколы канального уровня оформляют переданные им пакеты в кадры собственного формата, помещая указанный адрес назначения в одно из полей такого кадра, а также сопровождая кадр контрольной суммой. Протокол канального уровня имеет локальный смысл, он предназначен для доставки кадров данных, как правило, в пределах сетей с простой топологией связей и однотипной или близкой технологией, например в односегментных сетях Ethernet или же в многосегментных сетях Ethernet и Token Ring иерархической топологии, разделенных только мостами и коммутаторами. Во всех этих конфигурациях адрес назначения имеет локальный смысл для данной сети и не изменяется при прохождении кадра от узла-источника к узлу назначения. Возможность передавать данные между локальными сетями разных технологий связана с тем, что в этих технологиях используются адреса одинакового формата, к тому же производители сетевых адаптеров обеспечивают уникаль-

ность адресов независимо от технологии.

Другой областью действия протоколов канального уровня являются связи типа «точка-точка» глобальных сетей, когда протокол канального уровня ответственен за доставку кадра непосредственному соседу. Адрес в этом случае не имеет принципиального значения, а на первый план выходит способность протокола восстанавливать искаженные и утерянные кадры, так как плохое качество территориальных каналов, особенно коммутируемых телефонных, часто требует выполнения подобных действий.

Если же перечисленные выше условия не соблюдаются, например связи между сегментами Ethernet имеют петлевидную структуру, либо объединяемые сети используют различные способы адресации, как это имеет место в сетях Ethernet и X.25, то протокол канального уровня не может в одиночку справиться с задачей передачи кадра между узлами и требует помощи протокола сетевого уровня.

Сетевой уровень

Сетевой уровень (Network layer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать совершенно разные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей. Функции сетевого уровня достаточно разнообразны. Начнем их рассмотрение на примере объединения локальных сетей.

Протоколы канального уровня локальных сетей обеспечивают доставку данных между любыми узлами только в сети с соответствующей типовой топологией, например топологией иерархической звезды. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Можно было бы усложнить протоколы канального уровня для поддержания петлевидных избыточных связей, но принцип разделения обязанностей между уровнями приводит к другому решению. Чтобы, с одной стороны, сохранить простоту процедур передачи данных для типовых топологий, а с другой — допустить использование произвольных топологий, вводится дополнительный сетевой уровень.

На сетевом уровне сам термин сеть наделяют специфическим значением. В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Внутри сети доставка данных обеспечивается соответствующим канальным уровнем, а вот доставкой данных между сетями занимается сетевой уровень, который и поддерживает возможность правильного выбора маршрута передачи сообщения даже в том случае, когда характер структуры связей между составляющими сетями отличается от принятого в протоколах канального уровня.

Сети соединяются между собой специальными устройствами, называемыми

маршрутизаторами. Маршрутизатор — это устройство, которое собирает информацию о топологии межсетевых соединений и на ее основании пересыпает пакеты сетевого уровня в сеть назначения. Чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество транзитных передач между сетями, или хопов (hop — прыжок), каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, через которые проходит пакет.

На рис. 3.8 показаны четыре сети, связанные тремя маршрутизаторами. Между узлами А и В данной сети пролегают два маршрута: первый через маршрутизаторы 1 и 3, а второй через маршрутизаторы 1, 2 и 3.

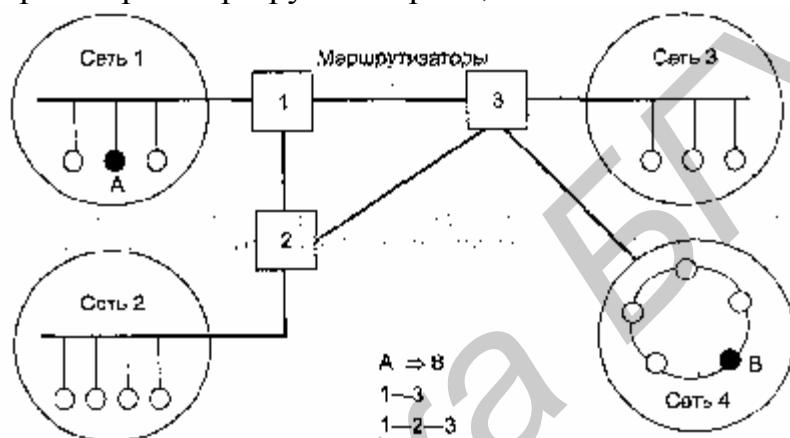


Рис. 3.8. Пример составной сети

Проблема выбора наилучшего пути называется маршрутизацией, и ее решение является одной из главных задач сетевого уровня. Эта проблема осложняется тем, что самый короткий путь не всегда самый лучший. Часто критерием при выборе маршрута является время передачи данных по этому маршруту; оно зависит от пропускной способности каналов связи и интенсивности трафика, которая может изменяться с течением времени. Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки, в то время как другие принимают решения на основе средних показателей за длительное время. Выбор маршрута может осуществляться и по другим критериям, например надежности передачи.

В общем случае функции сетевого уровня шире, чем функции передачи сообщений по связям с нестандартной структурой, которые мы сейчас рассмотрели на примере объединения нескольких локальных сетей. Сетевой уровень решает также задачи согласования разных технологий, упрощения адресации в крупных сетях и создания надежных и гибких барьеров на пути нежелательного трафика между сетями.

Сообщения сетевого уровня принято называть пакетами (packet). При организации доставки пакетов на сетевом уровне используется понятие «номер сети». В этом случае адрес получателя состоит из старшей части — номера сети и младшей — номера узла в этой сети. Все узлы одной сети должны иметь одну и

ту же старшую часть адреса, поэтому термину «сеть» на сетевом уровне можно дать и другое, более формальное определение: сеть — это совокупность узлов, сетевой адрес которых содержит один и тот же номер сети.

На сетевом уровне определяются два вида протоколов. Первый вид — сетевые протоколы (routed protocols) — реализуют продвижение пакетов через сеть. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. Однако часто к сетевому уровню относят и другой вид протоколов, называемых протоколами обмена маршрутной информацией или просто протоколами маршрутизации (routing protocols). С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений. Протоколы сетевого уровня реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

На сетевом уровне работают протоколы еще одного типа, которые отвечают за отображение адреса узла, используемого на сетевом уровне, в локальный адрес сети. Такие протоколы часто называют протоколами разрешения адресов (Address Resolution Protocol, ARP). Иногда их относят не к сетевому уровню, а к канальному, хотя тонкости классификации не изменяют их сути.

Примерами протоколов сетевого уровня являются протокол межсетевого взаимодействия IP стека TCP/IP и протокол межсетевого обмена пакетами IPX стека Novell.

Транспортный уровень

На пути от отправителя к получателю пакеты могут быть искажены или потеряны. Хотя некоторые приложения имеют собственные средства обработки ошибок, существуют и такие, которые предпочитают сразу иметь дело с надежным соединением. Транспортный уровень (Transport layer) обеспечивает приложениям или верхним уровням стека — прикладному и сеансовому — передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными прикладными протоколами через общий транспортный протокол, а главное — способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Выбор класса сервиса транспортного уровня определяется, с одной стороны, тем, в какой степени задача обеспечения надежности решается самими приложениями и протоколами более высоких, чем транспортный, уровней, а с другой стороны, этот выбор зависит от того, насколько надежной является система транспортировки данных в сети, обеспечиваемая уровнями, расположенными ниже транспортного — сетевым, канальным и физическим. Так, например, если качество каналов передачи связи очень высокое и вероятность возникновения ошибок, не обнаруженных протоколами более низких уровней, невелика, то ра-

зумно воспользоваться одним из облегченных сервисов транспортного уровня, не обремененных многочисленными проверками, квитированием и другими приемами повышения надежности. Если же транспортные средства нижних уровней изначально очень ненадежны, то целесообразно обратиться к наиболее развитому сервису транспортного уровня, который работает, используя максимум средств для обнаружения и устранения ошибок, включая предварительное установление логического соединения, контроль доставки сообщений по контрольным суммам и циклической нумерации пакетов, установление тайм-аутов доставки и т. п.

Как правило, все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети — компонентами их сетевых операционных систем. В качестве примера транспортных протоколов можно привести протоколы TCP и UDP стека TCP/IP и протокол SPX стека Novell.

Протоколы нижних четырех уровней обобщенно называют сетевым транспортом или транспортной подсистемой, так как они полностью решают задачу транспортировки сообщений с заданным уровнем качества в составных сетях с произвольной топологией и различными технологиями. Оставшиеся три верхних уровня решают задачи предоставления прикладных сервисов на основании имеющейся транспортной подсистемы.

Сеансовый уровень

Сеансовый уровень (Session layer) обеспечивает управление взаимодействием: фиксирует, какая из сторон является активной в настоящий момент, предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, а не начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется в виде отдельных протоколов, хотя функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

Представительный уровень

Представительный уровень (Presentation layer) имеет дело с формой представления передаваемой по сети информации, не меняя при этом ее содержания. За счет уровня представления информация, передаваемая прикладным уровнем одной системы, всегда понятна прикладному уровню другой системы. С помощью средств данного уровня протоколы прикладных уровней могут преодолеть синтаксические различия в представлении данных или же различия в кодах символов, например кодов ASCII и EBCDIC. На этом уровне может выполняться шифрование и дешифрование данных, благодаря которым секретность обмена данными обеспечивается сразу для всех прикладных служб. Примером такого протокола является протокол Secure Socket Layer (SSL), который обеспечи-

ваєт секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Прикладной уровень

Прикладной уровень (Application layer) — это в действительности просто набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или Гипертекстовые web-страницы, а также организуют свою совместную работу, например, по протоколу электронной почты. Единица данных, которой оперирует прикладной уровень, обычно называется сообщением (message).

Существует очень большое разнообразие служб прикладного уровня. Приведем в качестве примера хотя бы несколько наиболее распространенных реализаций файловых служб: NCP в операционной системе Novell Net Ware, SMB в Microsoft Windows NT, NFS, FTP и TFTP, входящие в стек TCP/IP.

Сетезависимые и сетенезависимые уровни

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня — физический, канальный и сетевой — являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети и используемым коммуникационным оборудованием. Например, переход на оборудование FDDI означает полную смену протоколов физического и канального уровней во всех узлах сети.

Три верхних уровня — прикладной, представительный и сеансовый — ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют какие бы то ни было изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию 100VG-AnyLAN не требует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает детали функционирования нижних уровней от верхних. Это позволяет разрабатывать приложения, не зависящие от технических средств непосредственной транспортировки сообщений.

На рис. 3.9 показаны уровни модели OSI, на которых работают различные элементы сети. Компьютер с установленной на нем сетевой ОС взаимодействует с другим компьютером по протоколам всех семи уровней. Это взаимодействие компьютеры осуществляют опосредовано через различные коммуникационные устройства: концентраторы, модемы, мосты, коммутаторы, маршрутизаторы, мультиплексоры. В зависимости от типа коммуникационное устройство может работать либо только на физическом уровне (повторитель), либо на физическом

и канальном (мост), либо на физическом, канальном и сетевом, иногда захватываю-
ая и транспортный уровень (маршрутизатор). На рис. 3.10 показано соответст-
вие функций различных коммуникационных устройств уровням модели OSI.

Модель OSI представляет хотя и очень важную, но только одну из многих мо-
делий коммуникаций. Эти модели и связанные с ними стеки протоколов могут
отличаться количеством уровней, их функциями, форматами сообщений, служ-
бами, поддерживаемыми на верхних уровнях, и прочими параметрами.

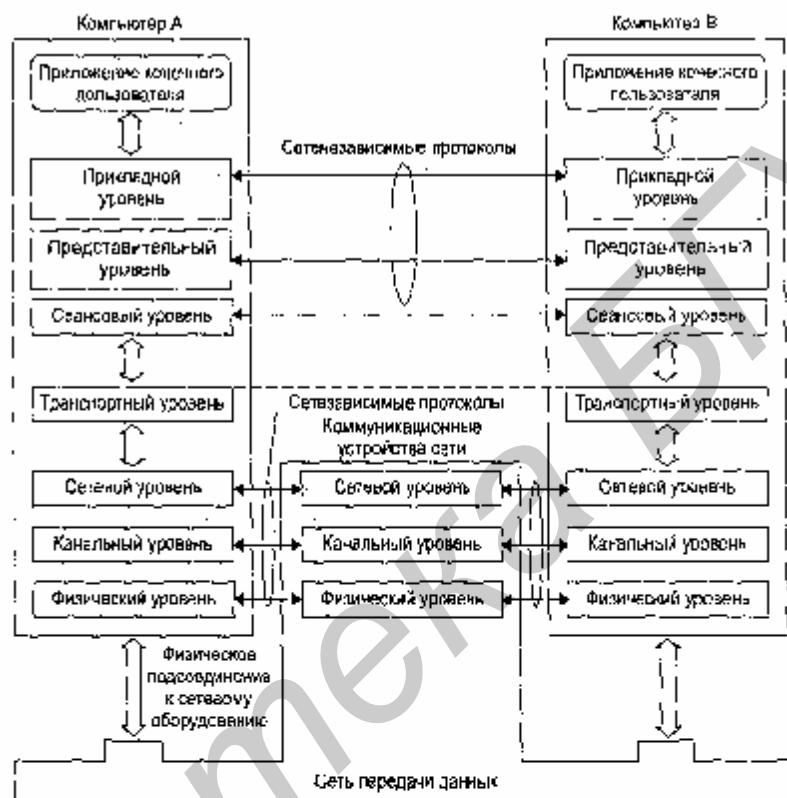


Рис. 3.9. Сетевозависимые и сетевозависимые уровни модели OSI

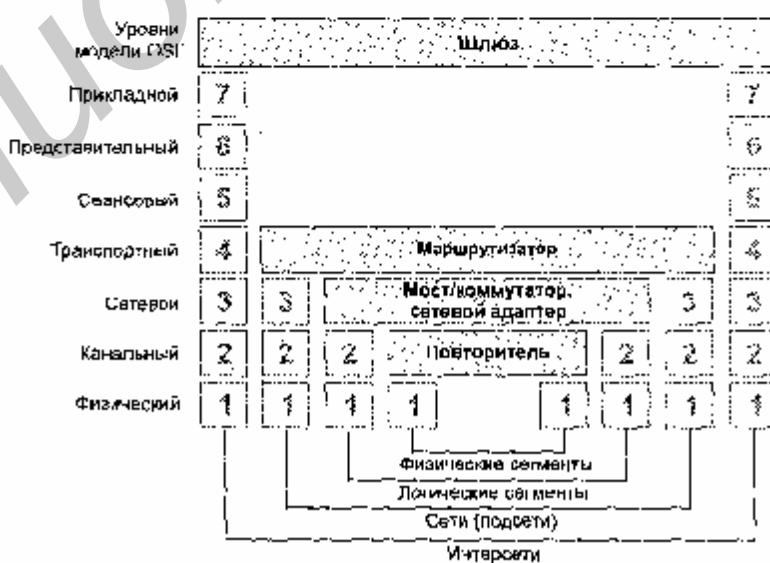


Рис. 3.10. Соответствие функций различных устройств сети уровням модели OSI

Выводы

- Эталонная модель OSI определяет семь уровней взаимодействия систем в сетях с коммутацией пакетов, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень.
- Физический уровень (Physical layer) является самым нижним уровнем. На этом уровне определяются характеристики электрических сигналов, передающих дискретную информацию, например крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала. Кроме того, здесь стандартизуются типы разъемов и назначение каждого контакта.
- Канальный уровень (Data-link layer) по-разному определяется в технологиях локальных и глобальных сетей. В сетях WAN, для которых характерна произвольная топология, на канальном уровне решаются проблемы надежной связи двух соседних узлов. В сетях LAN, имеющих типовую топологию — звезда, кольцо, общая шина, — на этом уровне решается задача обеспечения надежной связи любой пары узлов. Канальный уровень отвечает за формирование кадров (frame), физическую адресацию, разделение передающей среды, контроль ошибок.
- Сетевой уровень (Network layer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать в общем случае различные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей. Сетевой уровень манипулирует пакетами (packet).
- Транспортный уровень (Transport layer) обеспечивает приложениям или верхним уровням стека — прикладному и сеансовому — передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов сервиса, которые отличаются срочностью, возможностью восстановления прерванной связи, способностью к обнаружению и исправлению ошибок передачи.
- Сеансовый уровень (Session layer) обеспечивает управление взаимодействием: фиксирует, какая из сторон является активной в настоящий момент, предоставляет средства синхронизации.
- Представительный уровень (Presentation layer) имеет дело с формой представления передаваемой по сети информации, не меняя при этом ее содержания.
- Прикладной уровень (Application layer) — самый верхний уровень, являющийся, по сути, набором разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые web-страницы.
- Единица данных, которой оперируют верхние три уровня, обычно называется сообщением (message).

Стандартизация сетей

Понятие «открытая система»

Модель OSI, как это следует из ее названия (Open System Interconnection), описывает взаимосвязи открытых систем. Что же такое открытая система?

В широком смысле открытой системой может быть названа любая система (компьютер, вычислительная сеть, ОС, программный пакет, другие аппаратные и программные продукты), которая построена в соответствии с открытыми спецификациями.

Напомним, что под термином «спецификация» в вычислительной технике понимают формализованное описание аппаратных или программных компонентов, способов их функционирования, взаимодействия с другими компонентами, условий эксплуатации, ограничений и особых характеристик. Попятно, что не всякая спецификация является стандартом. В свою очередь, под открытыми спецификациями понимаются опубликованные, общедоступные спецификации, соответствующие стандартам и принятые в результате достижения согласия после всестороннего обсуждения всеми заинтересованными сторонами.

Использование при разработке систем открытых спецификаций позволяет третьим сторонам разрабатывать для этих систем различные аппаратные или программные средства расширения и модификации, а также создавать программно-аппаратные комплексы из продуктов разных производителей.

Для реальных систем полная открытость является недостижимым идеалом. Как правило, даже в системах, называемых открытыми, этому определению соответствуют лишь некоторые части, поддерживающие внешние интерфейсы. Например, открытость семейства операционных систем UNIX заключается, кроме всего прочего, в наличии стандартизованного программного интерфейса между ядром и приложениями, что позволяет легко переносить приложения из среды одной версии UNIX в среду другой версии. Еще одним примером частичной открытости является применение в достаточно закрытой операционной системе Novell NetWare открытого интерфейса Open Driver Interface (ODI) для включения в систему драйверов сетевых адаптеров независимых производителей. Чем больше открытых спецификаций использовано при разработке системы, тем более открытой она является.

Модель OSI касается только одного аспекта открытости, а именно открытости средств взаимодействия устройств, связанных в вычислительную сеть. Здесь под открытой системой понимается сетевое устройство, готовое взаимодействовать с другими сетевыми устройствами с использованием стандартных правил, определяющих формат, содержание и значение принимаемых и отправляемых сообщений.

Если две сети построены с соблюдением принципов открытости, то это дает следующие преимущества:

- возможность построения сети из аппаратных и программных средств различных производителей, придерживающихся одного и того же стандарта;
- возможность безболезненной замены отдельных компонентов сети другими, более совершенными, что позволяет сети развиваться с минимальными затратами;
- возможность легкого сопряжения одной сети с другой;

- простоту освоения и обслуживания сети.

Ярким примером открытой системы является международная сеть Интернет. Эта сеть развивалась в полном соответствии с требованиями, предъявляемыми к открытым системам. В разработке ее стандартов принимали участие тысячи специалистов-пользователей этой сети из различных университетов, научных организаций и фирм-производителей вычислительной аппаратуры и программного обеспечения, работающих в разных странах. Само название стандартов, определяющих работу сети Интернет — Request For Comments (RFC), что можно перевести как «запрос на комментарии», — показывает гласный и открытый характер принимаемых стандартов. В результате сеть Интернет сумела объединить в себе самое разнообразное оборудование и программное обеспечение огромного числа сетей, разбросанных по всему миру.

Модульность и стандартизация

Модульность — это одно из неотъемлемых и естественных свойств вычислительных сетей. Модульность проявляется не только в многоуровневом представлении коммуникационных протоколов в конечных узлах сети, хотя это, безусловно, важная и принципиальная особенность сетевой архитектуры. Сеть состоит из огромного числа различных модулей — компьютеров, сетевых адаптеров, мостов, маршрутизаторов, модемов, операционных систем и модулей приложений. Разнообразные требования, предъявляемые предприятиями к компьютерным сетям, привели к такому же разнообразию выпускаемых для построения сети устройств и программ. Эти продукты отличаются не только основными функциями (имеются в виду функции, выполняемые, например, повторителями, мостами или программными редиректорами), но и многочисленными вспомогательными функциями, предоставляющими пользователям или администраторам дополнительные удобства, такие как автоматизированное конфигурирование параметров устройства, автоматическое обнаружение и устранение некоторых неисправностей, возможность программного изменения связей в сети и т. п. Разнообразие увеличивается также потому, что многие устройства и программы отличаются сочетаниями тех или иных основных и дополнительных функций — существуют, например, устройства, сочетающие основные возможности коммутаторов и маршрутизаторов, к которым добавляется еще и набор некоторых дополнительных функций, характерный только для данного продукта.

В результате не существует компаний, которая смогла бы обеспечить производство полного набора всех типов и подтипов оборудования и программного обеспечения, требуемого для построения сети. Но так как все компоненты сети должны работать согласованно, совершенно необходимым оказалось принятие многочисленных стандартов, которые, если не во всех, то хотя бы в большинстве случаев, гарантировали бы совместимость оборудования и программ различных фирм-изготовителей. Таким образом, понятия модульности и стандартизации в сетях неразрывно связаны, и модульный подход только тогда дает преимущества, когда он сопровождается следованием стандартам.

В результате открытый характер стандартов и спецификаций важен не только для коммуникационных протоколов, но и для всех многочисленных функций разнообразных устройств и программ, выпускаемых для построения сети. Нужно отметить, что большинство стандартов, принимаемых сегодня, носят открытый характер. Время закрытых систем, точные спецификации на которые были известны только фирме-производителю, ушло. Все осознали, что возможность легкого взаимодействия с продуктами конкурентов не снижает, а, наоборот, повышает ценность изделия, так как его можно применить в большем количестве работающих сетей, построенных на продуктах разных производителей. Поэтому даже фирмы, ранее выпускавшие весьма закрытые системы — такие как IBM, Novell или Microsoft, — сегодня активно участвуют в разработке открытых стандартов и применяют их в своих продуктах.

Сегодня в секторе сетевого оборудования и программ с совместимостью продуктов разных производителей сложилась следующая ситуация. Практически все продукты, как программные, так и аппаратные, совместимы по функциям и свойствам, которые были внедрены в практику уже достаточно давно и стандарты на которые уже разработаны и приняты по крайней мере 3-4 года назад. В то же время очень часто принципиально новые устройства, протоколы и свойства оказываются несовместимыми даже у ведущих производителей. Такая ситуация характерна не только для тех устройств или функций, стандарты на которые еще не успели принять (это естественно), но и для устройств, стандарты на которые существуют уже несколько лет. Совместимость достигается только после того, как все производители реализуют принятый стандарт в своих изделиях, причем одинаковым образом.

Источники стандартов

Работы по стандартизации вычислительных сетей ведутся большим количеством организаций. В зависимости от статуса организаций различают следующие виды стандартов:

- стандарты отдельных фирм (например, стек протоколов DECnet фирмы Digital Equipment или графический интерфейс OPEN LOOK для UNIX-систем фирмы Sun);
- стандарты специальных комитетов и объединений, создаваемых несколькими фирмами, например, стандарты технологии ATM, разрабатываемые специально созданным объединением ATM Forum, насчитывающим около 100 коллективных участников, или стандарты союза Fast Ethernet Alliance по разработке стандартов 100 Мбит Ethernet;
- национальные стандарты, например, стандарт FDDI, представляющий один из многочисленных стандартов, разработанных Американским национальным институтом стандартов (ANSI), или стандарты безопасности для операционных систем, разработанные Национальным центром компьютерной безопасности (NCSC) Министерства обороны США;
- международные стандарты, например, модель и стек коммуникационных протоколов Международной организации по стандартизации (ISO), многочис-

ленные стандарты Международного союза электросвязи (ITU), в том числе стандарты на сети с коммутацией пакетов X.25, сети frame relay, ISDN, модемы и многие другие.

Некоторые стандарты, непрерывно развиваясь, могут переходить из одной категории в другую. В частности, фирменные стандарты на продукцию, получившую широкое распространение, обычно становятся международными стандартами «де-факто», так как вынуждают производителей из разных стран следовать фирменным стандартам, чтобы обеспечить совместимость своих изделий с этими популярными продуктами. Например, из-за феноменального успеха персонального компьютера компании IBM фирменный стандарт на архитектуру IBM PC стал международным стандартом де-факто.

Более того, ввиду широкого распространения некоторые фирменные стандарты становятся основой для национальных и международных стандартов «де-юре». Например, стандарт Ethernet, первоначально разработанный компаниями Digital Equipment, Intel и Xerox, через некоторое время и в несколько измененном виде был принят как национальный стандарт IEEE 802.3, а затем организация ISO утвердила его в качестве международного стандарта ISO 8802.3.

Далее приводятся краткие сведения об организациях, наиболее активно и успешно занимающихся разработкой стандартов в области вычислительных сетей.

□ Международная организация по стандартизации (International Organization for Standardization, ISO), часто называемая также International Standards Organization, представляет собой ассоциацию ведущих национальных организаций по стандартизации разных стран. Главным достижением ISO явилась модель взаимодействия открытых систем OSI, которая в настоящее время является концептуальной основой стандартизации в области вычислительных сетей. В соответствии с моделью OSI этой организацией был разработан стандартный стек коммуникационных протоколов OSI.

□ Международный союз электросвязи (International Telecommunications Union, ITU) — организация, являющаяся в настоящее время специализированным органом Организации Объединенных Наций. Наиболее значительную роль в стандартизации вычислительных сетей играет постоянно действующий в рамках этой организации Международный консультативный комитет по телефонии и телеграфии (МККТТ) (Consultative Committee on International Telegraphy and Telephony, CCITT). В результате проведенной в 1993 году реорганизации ITU CCITT несколько изменил направление своей деятельности и сменил название — теперь он называется сектором телекоммуникационной стандартизации ITU (ITU Telecommunication Standardization Sector, ITU-T). Основу деятельности ITU-T составляет разработка международных стандартов в области телефонии, телематических служб (электронной почты, факсимальной связи, телетекста, телекса и т. д.), передачи данных, аудио- и видео сигналов. За годы своей деятельности ITU-T выпустил огромное число рекомендаций-стандартов. Свою работу ITU-T строит на изучении опыта сторонних организаций, а также на результатах собственных исследований. Раз в четыре года издаются труды ITU-T в виде так называемой «Книги», которая на самом деле представляет собой це-

лый набор обычных книг, сгруппированных в выпуски, которые, в свою очередь, объединяются в тома. Каждый том и выпуск содержат логически взаимосвязанные рекомендации, Например, том III Синей книги содержит рекомендации для цифровых сетей с интеграцией услуг (ISDN), а весь том VIII (за исключением выпуска VIII.1. который содержит рекомендации серии V для передачи данных по телефонной сети) посвящен рекомендациям серии X: X.25 для сетей с коммутацией пакетов, X.400 для систем электронной почты, X.500 для глобальной справочной службы и многим другим.

- Институт инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronics Engineers, IEEE) — национальная организация США, определяющая сетевые стандарты. В 1981 году рабочая группа 802 этого института сформулировала основные требования, которым должны удовлетворять локальные вычислительные сети. Группа 802 определила множество стандартов, из них самыми известными являются стандарты 802.1, 802.2, 802.3 и 802.5, которые описывают общие понятия, используемые в области локальных сетей, а также стандарты на два нижних уровня сетей Ethernet и Token Ring.
- Европейская ассоциация производителей компьютеров (European Computer Manufacturers Association, ECMA) — некоммерческая организация, активно сотрудничающая с ITU-T и ISO, занимается разработкой стандартов и технических обзоров, относящихся к компьютерной и коммуникационной технологиям. Известна своим стандартом ECMA-101, используемым при передаче отформатированного текста и графических изображений с сохранением оригинального формата.
- Ассоциация производителей компьютеров и оргтехники (Computer and Business Equipment Manufacturers Association, СВЕМА) — организация американских фирм-производителей аппаратного обеспечения; аналогична европейской ассоциации ЕКМА; участвует в разработке стандартов на обработку информации и соответствующее оборудование.
- Ассоциация электронной промышленности (Electronic Industries Association, EIA) — промышленно-торговая группа производителей электронного и сетевого оборудования; является национальной, коммерческой ассоциацией США; проявляет значительную активность в разработке стандартов для проводов, коннекторов и других сетевых компонентов. Ее наиболее известный стандарт - RS-232C.
- Министерство обороны США (Department of Defense, DoD) имеет многочисленные подразделения, занимающиеся созданием стандартов для компьютерных систем. Одной из самых известных разработок DoD является стек транспортных протоколов TCP/IP.
- Американский национальный институт стандартов (American National Standards Institute, ANSI) — эта организация представляет США в Международной организации по стандартизации ISO. Комитеты ANSI ведут работу по разработке стандартов в различных областях вычислительной техники. Так, комитет ANSI X3T9.5 совместно с фирмой IBM занимается стандартизацией локальных сетей крупных ЭВМ (архитектура сетей SNA). Известный стандарт FDDI также является результатом деятельности этого комитета ANSI. В области микроком-

пьютеров ANSI разрабатывает стандарты на языки программирования, интерфейс SCSI. ANSI разработал рекомендации по переносимости для языков C, FORTRAN, COBOL.

Особую роль в выработке международных открытых стандартов играют стандарты Интернета. Ввиду большой и постоянно растущей популярности Интернета, эти стандарты становятся международными стандартами «де-факто», многие из которых затем приобретают статус официальных международных стандартов за счет их утверждения одной из вышеперечисленных организаций, в том числе ISO и ITU-T. Существует несколько организационных подразделений, отвечающих за развитие Интернета и, в частности, за стандартизацию средств Интернета.

Основным из них является Internet Society (ISOC) — профессиональное сообщество, которое занимается общими вопросами эволюции и роста Интернета как глобальной коммуникационной инфраструктуры. Под управлением ISOC работает Internet Architecture Board (IAB) — организация, в ведении которой находится технический контроль и координация работ для Интернета. IAB координирует направление исследований и новых разработок для стека TCP/IP и является конечной инстанцией при определении новых стандартов Интернета.

В IAB входят две основные группы: Internet Engineering Task Force (IETF) и Internet Research Task Force (IRTF). IETF — это инженерная группа, которая занимается решением ближайших технических проблем Интернета. Именно IETF определяет спецификации, которые затем становятся стандартами Интернета. В свою очередь, IRTF координирует долгосрочные исследовательские проекты по протоколам TCP/IP.

В любой организации, занимающейся стандартизацией, процесс выработки и принятия стандарта состоит из ряда обязательных этапов, которые, собственно, и составляют процедуру стандартизации. Рассмотрим эту процедуру на примере разработки стандартов Интернета.

1. Сначала в IETF представляется так называемый рабочий проект (draft) в виде, доступном для комментариев. Он публикуется в Интернете, после чего широкий круг заинтересованных лиц включается в обсуждение, этого документа, в него вносятся исправления, и наконец наступает момент, когда можно зафиксировать содержание документа. На этом этапе проекту присваивается номер RFC (возможен и другой вариант развития событий — после обсуждения рабочий проект отвергается и удаляется из Интернета).

2. После присвоения номера проект приобретает статус, предлагаемого стандарта. В течение 6 месяцев этот предлагаемый стандарт проходит проверку практикой, в результате в него вносятся изменения.

3. Если результаты практических исследований показывают эффективность предлагаемого стандарта, то ему со всеми внесенными изменениями присваивается статус проекта стандарта. Затем в течение не менее 4 месяцев проходят его дальнейшие испытания «на прочность», в число которых входит создание по крайней мере двух программных реализаций.

4. Если во время пребывания в ранге проекта стандарта в документ не было внесено никаких исправлений, то ему может быть присвоен статус официаль-

ного стандарта Интернета. Список утвержденных официальных стандартов Интернета публикуется в виде документа RFC и доступен в Интернете.

Следует заметить, что все стандарты Интернета носят название RFC с соответствующим порядковым номером, но далеко не все RFC являются стандартами Интернета — часто эти документы представляют собой комментарии к какому-либо стандарту или просто описания некоторой проблемы Интернета.

Стандартные стеки коммуникационных протоколов

Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. В настоящее время и сетях используется большое количество стеков коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI. Все эти стеки, кроме SNA на нижних уровнях (физическем и канальном), используют одни и те же хорошо стандартизованные протоколы Ethernet, Token Ring, FDDI и некоторые другие, которые позволяют использовать во всех сетях одну и ту же аппаратуру. Зато на верхних уровнях все стеки работают по своим собственным протоколам. Эти протоколы часто не соответствуют рекомендуемому моделью OSI разбиению на уровни. В частности, функции сеансового и представительного уровня, как правило, объединены с прикладным уровнем. Такое несоответствие связано с тем, что модель OSI появилась как результат обобщения уже существующих и реально используемых стеков, а не наоборот.

Стек OSI

Следует четко различать модель OSI и стек OSI. В то время как модель OSI является концептуальной схемой взаимодействия открытых систем, стек OSI представляет собой набор вполне конкретных спецификаций протоколов. В отличие от других стеков протоколов, стек OSI полностью соответствует модели OSI, он включает спецификации протоколов для всех семи уровней взаимодействия, определенных в этой модели. На нижних уровнях стек OSI поддерживает Ethernet, Token Ring, FDDI, протоколы глобальных сетей, X.25 и ISDN, — то есть использует разработанные вне стека протоколы нижних уровней, как и все другие стеки. Протоколы сетевого, транспортного и сеансового уровней стека OSI специфицированы и реализованы различными производителями, но распространены пока мало. Наиболее популярными протоколами стека OSI являются прикладные протоколы. К ним относятся: протокол передачи файлов FTAM, протокол эмуляции терминала VTP, протоколы справочной службы X.500, электронной почты X.400 и ряд других.

Протоколы стека OSI отличает большая сложность и неоднозначность спецификаций. Эти свойства явились результатом общей политики разработчиков стека, стремившихся учесть в своих протоколах все случаи жизни и все существующие и появляющиеся технологии. К этому нужно еще добавить и последст-

вия большого количества политических компромиссов, неизбежных при принятии международных стандартов по такому злободневному вопросу, как построение открытых вычислительных сетей.

Из-за своей сложности протоколы OSI требуют больших затрат вычислительной мощности центрального процессора, что делает их более подходящими для мощных машин, а не для сетей персональных компьютеров.

Стек OSI — международный, независимый от производителей стандарт. Его поддерживает правительство США в своей программе GOSIP, в соответствие с которой все компьютерные сети, устанавливаемые в правительственные учреждениях США после 1990 года, должны либо непосредственно поддерживать стек OSI, либо обеспечить переход на этот стек в будущем. Тем не менее стек OSI более популярен в Европе, чем в США, так как в Европе осталось меньше старых сетей, работающих по своим собственным протоколам. Большинство организаций пока только планируют переход к стеку OSI, и очень немногие приступили к созданию пилотных проектов. Из тех, кто работает в этом направлении, можно назвать Военно-морское ведомство США и сеть NFSNET. Одним из крупнейших производителей, поддерживающих OSI, является компания AT&T, ее сеть Star group полностью базируется на этом стеке.

Стек TCP/IP

Стек TCP/IP был разработан по инициативе Министерства обороны США более 20 лет назад для связи экспериментальной сети ARPAnet с другими сетями как набор общих протоколов для разнородной вычислительной среды. Большой вклад в развитие стека TCP/IP, который получил свое название по популярным протоколам IP и TCP, внес университет Беркли, реализовав протоколы стека в своей версии ОС UNIX. Популярность этой операционной системы привела к широкому распространению протоколов TCP, IP и других протоколов стека. Сегодня этот стек используется для связи компьютеров всемирной информационной сети Интернет, а также в огромном числе корпоративных сетей.

Стек TCP/IP на нижнем уровне поддерживает все популярные стандарты физического и канального уровней: для локальных сетей — это Ethernet, Token Ring, FDDI, для глобальных — протоколы работы на аналоговых коммутируемых и выделенных линиях SLIP, PPP, протоколы территориальных сетей X.25 и ISDN. Основными протоколами стека, давшими ему название, являются протоколы IP и TCP. Эти протоколы в терминологии модели OSI относятся к сетевому и транспортному уровням соответственно. IP обеспечивает продвижение пакета по составной сети, а TCP гарантирует надежность его доставки.

За долгие годы использования в сетях различных стран и организаций стек TCP/IP вобрал в себя большое количество протоколов прикладного уровня. К ним относятся такие популярные протоколы, как протокол пересылки файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP, используемый и электронной почте сети Интернет, гипертекстовые сервисы службы WWW и многие другие.

Сегодня стек TCP/IP представляет собой самый распространенный стек транс-

портных протоколов вычислительных сетей. Действительно, только в сети Интернет объединено более 10 миллионов компьютеров по всему миру, взаимодействующих друг с другом с помощью стека протоколов TCP/IP.

Стремительный рост популярности Интернета привел и к изменениям в расстановке сил в мире коммуникационных протоколов — протоколы TCP/IP, на которых построен Интернет, стали быстро теснить бесспорного лидера прошлых лет — стек IPX/SPX компании Novell. Сегодня в мире общее количество компьютеров, на которых установлен стек TCP/IP, намного превышает количество компьютеров, на которых работает стек IPX/SPX, и это говорит о резком переломе в отношении администраторов локальных сетей к протоколам, используемым на настольных компьютерах, так как именно они составляют подавляющее число мирового компьютерного парка и именно на них раньше почти везде работали протоколы компании Novell, необходимые для доступа к файловым серверам NetWare. Процесс становления стека TCP/IP в качестве стека номер один в любых типах сетей продолжается, и сейчас любая промышленная операционная система обязательно включает программную реализацию этого стека в своем комплекте поставки.

Хотя протоколы TCP/IP неразрывно связаны с Интернетом и каждый из много-миллионной армады компьютеров Интернета работает на основе этого стека, существует большое количество локальных, корпоративных и территориальных сетей, непосредственно не являющихся частями Интернета, в которых также используют протоколы TCP/IP. Чтобы отличать их от Интернета, эти сети называют сетями TCP/IP или просто IP-сетями.

Поскольку стек TCP/IP изначально создавался для глобальной сети Интернет, он имеет много особенностей, дающих ему преимущество перед другими протоколами, когда речь заходит о построении сетей, включающих глобальные связи. В частности, очень полезным свойством, делающим возможным применение этого протокола в больших сетях, является его способность фрагментировать пакеты. Действительно, большая составная сеть часто состоит из сетей, построенных на совершенно разных принципах. В каждой из этих сетей может быть установлена, собственная величина максимальной длины единицы передаваемых данных (кадра). В таком случае при переходе из одной сети, имеющей большую максимальную длину, в сеть с меньшей максимальной длиной может возникнуть необходимость деления передаваемого кадра на несколько частей. Протокол IP стека TCP/IP эффективно решает эту задачу.

Другой особенностью технологии TCP/IP является гибкая система адресации, позволяющая проще чем другие протоколы аналогичного назначения включать в интерсеть сети разных технологий. Это свойство также способствует применению стека TCP/IP для построения больших гетерогенных сетей.

В стеке TCP/IP очень экономно используются возможности широковещательных рассылок. Это свойство совершенно необходимо при работе на медленных каналах связи, характерных для территориальных сетей.

Однако, как и всегда, за получаемые преимущества надо платить, и платой здесь оказываются высокие требования к ресурсам и сложность администрирования IP-сетей. Мощные функциональные возможности протоколов стека

TCP/IP требуют для своей реализации больших вычислительных затрат. Гибкая система адресации и отказ от широковещательных рассылок приводят к наличию в IP-сети разнообразных централизованных служб типа DNS, DHCP и т. п. Каждая из этих служб направлена па облегчение администрирования сети, в том числе и на облегчение конфигурирования оборудования, по в то же время сама требует пристального внимания со стороны администраторов. Можно приводить и другие доводы за и против стека протоколов Интернета, однако факт остается фактом — сегодня это самый популярный стек протоколов, широко используемый как в глобальных, так и локальных сетях.

Стек IPX/SPX

Стек IPX/SPX является оригинальным стеком протоколов фирмы Novell, разработанным для сетевой операционной системы NetWare еще в начале 80-х годов. Протоколы сетевого и сеансового уровней Internet work Packet Exchange (IPX) и Sequenced Packet Exchange (SPX), которые дали название стеку, являются прямой «адаптацией» протоколов XNS фирмы Xerox, распространенных и гораздо меньшей степени, чем стек IPX/SPX. Популярность стека IPX/SPX непосредственно связана с операционной системой Novell NetWare, популярность которой сейчас значительно уступает операционным системам Microsoft.

Многие особенности стека IPX/SPX обусловлены ориентацией ранних версий ОС NetWare (до версии 4.0) на работу в локальных сетях небольших размеров, состоящих из персональных компьютеров со скромными ресурсами. Понятно, что для таких компьютеров компании Novell нужны были протоколы, на реализацию которых требовалось бы минимальное количество оперативной памяти (ограниченной в IBM-совместимых компьютерах под управлением MS-DOS объемом 640 Кбайт) и которые бы быстро работали на процессорах небольшой вычислительной мощности. В результате протоколы стека IPX/SPX до недавнего времени хорошо работали в локальных сетях и не очень — в больших корпоративных сетях, так как они слишком перегружали медленные глобальные связи широковещательными пакетами, интенсивно использующими несколькоими протоколами этого стека (например, для установления связи между клиентами и серверами). Это обстоятельство, а также тот факт, что стек IPX/SPX является собственностью фирмы Novell и на его реализацию нужно получать лицензию (то есть открытые спецификации не поддерживались), долгое время, ограничивали распространность его только сетями NetWare. Однако с момента выпуска версии NetWare 4.0 фирма Novell внесла и продолжает вносить в свои протоколы серьезные изменения, направленные на их адаптацию для работы в корпоративных сетях. Сейчас стек IPX/SPX реализован не только в NetWare, но и в нескольких других популярных сетевых ОС, например SCO UNIX, Sun Solaris, Microsoft Windows NT/2000.

Стек NetBIOS/SMB

Стек NetBIOS/SMB широко используется в продуктах компаний IBM и Microsoft. На физическом и канальном уровнях этого стека задействованы все наиболее распространенные протоколы Ethernet, Token Ring, FDDI и др. На верхних уровнях работают протоколы NetBEUI и SMB.

Протокол NetBIOS (Network Basic Input/Output System) появился в 1984 году как сетевое расширение стандартных функций базовой системы ввода-вывода (BIOS) IBM PC для сетевой программы PC Network фирмы IBM. В дальнейшем этот протокол был заменен так называемым протоколом расширенного пользовательского интерфейса NetBEUI — NetBIOS Extended User Interface. Для обеспечения совместимости приложений в качестве интерфейса к протоколу NetBEUI был сохранен интерфейс NetBIOS. Протокол NetBEUI разрабатывался как эффективный протокол, потребляющий немного ресурсов и предназначенный для сетей, насчитывающих не более 200 рабочих станций. Этот протокол содержит много полезных сетевых функций, которые можно отнести к сетевому, транспортному и сеансовому уровням модели OSI, однако с его помощью невозможна маршрутизация пакетов. Это ограничивает применение протокола NetBEUI локальными сетями, не разделенными на подсети, и делает невозможным его использование в составных сетях. Некоторые ограничения NetBEUI снимаются реализацией этого протокола NBF (NetBEUI Frame), которая включена в операционную систему Microsoft Windows NT.

Протокол SMB (Server Message Block) выполняет функции сеансового, представительного и прикладного уровней. На основе SMB реализуется файловая служба, а также службы печати и передачи сообщений между приложениями.

Стеки протоколов SNA фирмы IBM, DECnet корпорации Digital Equipment и AppleTalk/AFP фирмы Apple применяются в основном в операционных системах и сетевом оборудовании этих фирм.

Модель OSI	IBM/Microsoft TCP/IP	Novell	Стек OSI	
Прикладной	SMB	Telnet, FTP, SNM P, SMT P, WW W	NCP, SAP	X.400, X.500, FTAM
Представительный				Представительный и протокол OSI
Сеансовый	Net-BIOS	TCP		Сеансовый протокол OSI
Транспортный			SPX	Транспортный протокол OSI
Сетевой		IP, RIP,	IPX, RIP,	ES-ES, IS-IS

		OSPF	NLS P		
Канальный		802.3 (Ethernet), 802.5 (Token Ring), FDDI, Fast Ethernet, SLIP, 1QOVG-Anyl_AN, X.25, ATM, LAP-B, LAP-D, PPP			
Физический		Коаксиал, экранированная и неэкранированная витая пара, оптоволокно, радиоволны			

Рис. 3.11. Соответствие популярных стеков протоколов модели OSI

На рис. 3.11 показано соответствие некоторых, наиболее популярных протоколов уровням модели OSI. Часто это соответствие весьма условно, так как модель OSI — это только руководство к действию, причем достаточно общее, а конкретные протоколы разрабатывались для решения специфических задач, причем многие из них появились до разработки модели OSI. В большинстве случаев разработчики стеков отдавали предпочтение скорости работы сети в ущерб модульности — ни один стек, кроме стека OSI, не разбит на семь уровней. Чаще всего в стеке явно выделяются 3-4 уровня: уровень сетевых адаптеров, в котором реализуются протоколы физического и канального уровней, сетьевой уровень, транспортный уровень и уровень служб, включающий в себя функции сеансового, представительского и прикладного уровней.

Выводы

- Открытой системой может быть названа любая система (компьютер, вычислительная сеть, ОС, программный пакет, другие аппаратные и программные продукты), которая построена в соответствии с общедоступными спецификациями, соответствующими стандартам и принятыми в результате публичного обсуждения всеми заинтересованными сторонами.
- Модульность и стандартизация являются необходимыми условиями построения открытых систем.
- В зависимости от статуса организаций различают следующие виды стандартов: стандарты отдельных компаний, стандарты специальных комитетов и объединений, национальные стандарты, международные стандарты.
- Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI.

Вопросы и упражнения к части I

1. Чем можно объяснить тот факт, что глобальные сети появились раньше локальных?
2. Назовите основные тенденции сближения технологий локальных и глобальных сетей.
3. Может ли сеть оператора связи одновременно быть корпоративной сетью?
4. Поясните использование термина «сеть» в следующих предложениях:
 - сеть нашего предприятия включает сеть Ethernet и сеть Token Ring;
 - маршрутизатор — это устройство, которое соединяет сети;
 - чтобы получить выход в Интернет, необходимо получить у поставщика услуг Интернет номер сети;
 - в последнее время IP-сети становятся все более распространенными;
 - гетерогенность корпоративной сети приводит к тому, что на первый план часто выходит проблема согласования сетей.
5. Всякое ли приложение, выполняемое в сети, можно назвать сетевым?
6. Что общего и в чем отличие между взаимодействием компьютеров в сети и взаимодействием компьютера с периферийным устройством?
7. Как распределяются функции между сетевым адаптером и его драйвером?
8. Какая информация передается по каналу, связывающему внешние интерфейсы компьютера и ПУ:
 - данные, поступающие от контроллера на ПУ (A);
 - команды управления, которые контроллер передаст на устройство управления ПУ (B);
 - данные, возвращаемые устройством управления ПУ в компьютер (C); О команды, которые устройство управления ПУ передает в компьютер (D)?
9. Какие (какое) из перечисленных действий, как правило, выполняются драйвером периферийного устройства:
 - преобразование байта из внешнего регистра (порта) контроллера в последовательность битов (A);
 - передача каждого бита в линию связи (B);
 - загрузка данных из оперативной памяти во внутренний буфер контроллера (C);
 - обрамление байта стартовым и стоповым битами — синхронизация (D); О формирование бита четности (E)?
10. Что такое редиректор ОС (возможно несколько правильных ответов)? О Клиентский модуль (A). О Серверный модуль (Б).
 - Модуль, который распознает запросы к удаленным ресурсам (C). О Модуль, отвечающий за администрирование пользователей (D).
11. К какому компоненту сетевой ОС может быть отнесен драйвер? Какие из перечисленных компонентов непременно должны быть в составе ОС, чтобы компьютер мог выполнять запросы из сети:
 - серверный модуль;
 - коммуникационные средства;

клиентский модуль?

12. К какому типу топологии можно отнести структуру, образованную: тремя, связанными друг с другом в виде треугольника узлами (А); четырьмя связанными друг с другом в виде квадрата узлами (В); тремя последовательно (последний не связан с первым) соединенными друг с другом узлами (С):

полносвязная;

ячеистая;

кольцо;

звезда?

13. Какой тип конфигурации наиболее распространен сегодня в локальных сетях?

14. К какому типу адреса можно отнести следующие адреса:

www. Intu.ru;

20-34-a2-00-c2-27;

128.345.23.170?

15. Назовите главные недостатки полносвязной топологии, а также топологий типа общая шина, звезда, кольцо.

16. Какую топологию имеет односегментная сеть Ethernet, построенная на основе концентратора: общая шина или звезда?

17. Какие более частные задачи включает в себя обобщенная задача коммутации абонентов?

18. Что такое поток данных?

19. Какой способ коммутации более эффективен: коммутация каналов или коммутация пакетов?

20. Объясните разницу между тремя понятиями:

О логические соединения, на которых основаны некоторые протоколы; О виртуальные каналы в сетях с коммутацией пакетов; О составные каналы в сетях с коммутацией каналов.

21. Заполните следующую таблицу, отметив в каждой строке, для какого способа коммутации (каналов или пакетов) характерно указанное свойство.

Свойство	Способ коммутации
Гарантированная пропускная способность (полоса) для каждого использующих абонентов	
Сеть может отказать абоненту в установлении соединения	
Трафик не является прерывистым передается без задержек	
Ресурсы сети используются симметрично при передаче пульсирующего трафика	
Адрес не используется только на этапе установления соединения	
Пропускная способность сети для абонентов постоянна, задержки передачи имеют случайный характер	
Сеть всегда готова принять данные от абонента	
Адрес передается с каждым пакетом	
Мультиплексирование на основе синхронного разделения времени	

22. Определите, на сколько увеличится время передачи данных в сети с коммутацией пакетов по сравнению с сетью с коммутацией каналов, если известно:

- общий объем передаваемых данных -- 200 Кбайт;
- суммарная длина канала — 5000 км (скорость передачи сигнала примите равной 0,66 скорости света);
- пропускная способность канала — 2 Мбайт/с;
- размер пакета без учета заголовка — 4 Кбайт;
- заголовок — 40 байт;
- межпакетный интервал — 1 мс;
- количество промежуточных коммутаторов — 10;
- время коммутации — 20 мс.

23. Какие из перечисленных концепций характерны для сетевой технологии Ethernet:

- коммутация пакетов;
- коммутация каналов;
- произвольная топология;
- топология общая шина;
- иерархическая адресация;
- плоская адресация;
- числовая адресация;
- разделяемая передающая среда;
- типовая топология;
- временное мультиплексирование?

24. Какие из следующих утверждений верны:

- разделение линий связи приводит к повышению пропускной способности канала (A);
- конфигурация физических связей может совпадать с конфигурацией логических связей (B);
- главной задачей службы разрешения имен является проверка сетевых имен и адресов на допустимость (C);
- протоколы без установления соединений называются также дейтаграммными протоколами (D)?

25. Какое из этих устройств можно назвать коммутатором:

- маршрутизатор (A);
- мост (B);
- автоматическая телефонная станция (C);
- мультиплексор (D);
- электрический выключатель (E);
- ни одно из них (F)?

26. Какие из перечисленных свойств характерны для сетей с дейтаграммным продвижением данных, а какие для сетей с виртуальными каналами:

- устойчивые пути следования трафика (A);
- учет существования в сети потоков (B);
- независимая маршрутизация каждого пакета (C);
- выбор следующего узла осуществляется только на основании адреса назначения (D);
- коммутация пакетов (E); использование меток (F);

○ уменьшение накладных расходов на передачу в пакете служебной информации (G)?

27. Какой англоязычный термин соответствует определению «сеть, предназначенная для концентрации информационных потоков, поступающих по многочисленным каналам связи от оборудования пользователей»?

○ Backbone.

○ Access network.

○ Core network.

○ Operating network.

28. Каким образом альтернативный оператор может обеспечить массовым клиентам доступ к ресурсам своей сети?

○ Проложив собственные абонентские окончания до помещений клиентов (A).

○ Построив свою собственную магистраль (B).

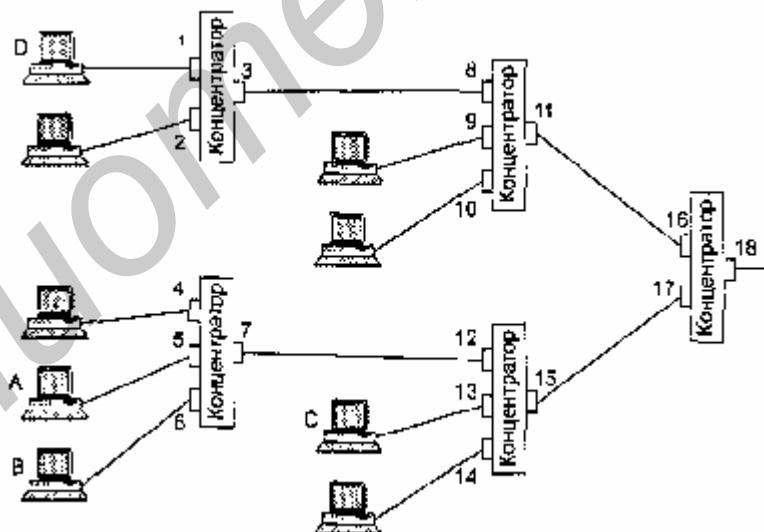
○ Взяв абонентские окончания в аренду у традиционного оператора связи (C).

○ Заключив договор с традиционным оператором связи о маршрутизации трафика клиентов в свою сеть (D).

29. В чем отличие логической структуризации сети от физической?

30. Определите функциональное назначение основных типов коммуникационного оборудования — повторителей, концентраторов, мостов, коммутаторов, маршрутизаторов.

31. Если все коммуникационные устройства в приведенном ниже фрагменте сети являются концентраторами, то на каких портах появится кадр, если его отправил компьютер А компьютеру В? Компьютеру С? Компьютеру D?



32. Если в предыдущем упражнении изменить условия и считать, что все коммуникационные устройства являются коммутаторами, то на каких портах появится кадр, посланный компьютером А компьютеру В? Компьютеру С? Компьютеру D?

33. Что такое «открытая система»? Приведите примеры закрытых систем.

34. Поясните разницу в употреблении терминов «протокол» и «интерфейс*» применительно к многоуровневой модели взаимодействия устройств в сети.

35. Что стандартизует модель OSI?

36. Что стандартизует стек OSI?

37. Почему в модели OSI семь уровней?
38. Дайте краткое описание функций каждого уровня и приведите примеры стандартных протоколов для каждого уровня модели OSI.
39. Являются ли термины «спецификация» и «стандарт» синонимами?
40. Какая организация разработала основные стандарты сетей Ethernet и Token Ring?
41. Из перечисленных названий стандартных стеков коммуникационных протоколов выделите названия, которые относятся к одному и тому же стеку: TCP/IP, Microsoft, IPX/SPX, Novell, Internet, DoD, NetBIOS/SMB, DECnet.
42. В чем состоит отличие локальных сетей от глобальных на уровне служб? А на уровне транспортной системы?
43. Назовите наиболее часто используемые характеристики производительности сети.
44. Что важнее для передачи мультимедийного трафика: надежность или синхронность?
45. Поясните значение указанных ниже англоязычных терминов. Что объединяет каждую группу?
- Availability; fault tolerance; security; extensibility.
 - Bridge, router, repeater, gateway, hub, concentrator, switch.
 - Packet, frame, stream, flow, datagram, message.
 - Telecommunication carrier, access network, backbone, data center.
 - Point-to-point, end-to-end, Quality of Service, best effort.
 - Circuit switching, packet switching, store-and-forward.

Часть II
Технологии
физического уровня

Библиотека БГУИР

ГЛАВА 4 Линии связи

При построении сетей применяются линии связи, использующие различную физическую среду: телефонные и телеграфные провода, подвешенные в воздухе, медные коаксиальные кабели, медные витые пары, волоконно-оптические кабели, радиоволны. При выборе того или иного типа линий связи разработчики, прежде всего, учитывают их технические характеристики, стоимость, а также простоту монтажа. Сегодня наиболее перспективными являются волоконно-оптические кабели. На них строятся как магистрали крупных территориальных и городских сетей, так и высокоскоростные линии связи локальных сетей. Популярной средой является также витая пара, которая характеризуется отличным отношением качество/стоимость.

Типы линий связи

Среда передачи информации

Линия связи (рис. 4.1) состоит в общем случае из физической среды, по которой передаются информационные сигналы, аппаратуры передачи данных и промежуточной аппаратуры. Синонимом термина линия связи (line) является термин канал связи (channel).



Рис. 4.1. Состав линии связи

Физическая среда передачи данных (medium) может представлять собой кабель, то есть набор проводов, изоляционных и защитных оболочек, соединительных разъемов, а также земную атмосферу или космическое пространство, через которые распространяются информационные сигналы. В современных телекоммуникационных системах информация передается с помощью электрического тока или напряжения, радиосигналов или световых сигналов — все эти физические процессы представляют собой колебания электромагнитного поля различной частоты и природы.

частоты и природы.

В зависимости от среды передачи данных линии связи разделяются на (рис. 4.2):

- проводные (воздушные);
- кабельные (médные и волоконно-оптические);
- радиоканалы наземной и спутниковой связи.

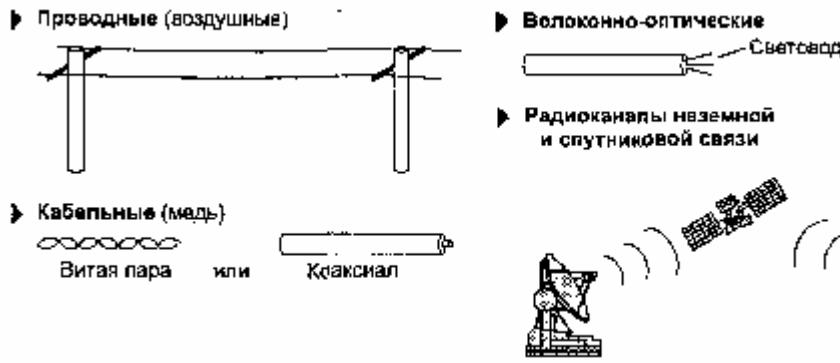


Рис. 4.2. Типы линий связи

Проводные (воздушные) линии связи представляют собой провода без каких-либо изолирующих или экранирующих оплеток, проложенные между столбами и висящие в воздухе. По таким линиям связи традиционно передаются телефонные или телеграфные сигналы, но при отсутствии других возможностей эти линии используются и для передачи компьютерных данных. Скоростные качества и помехозащищенность этих линий оставляют желать много лучшего. Сегодня проводные линии связи быстро вытесняются кабельными.

Кабельные линии имеют достаточно сложную конструкцию. Кабель состоит из проводников, заключенных в несколько слоев изоляции: электрической, электромагнитной, механической, а также, возможно, климатической. Кроме того, кабель может быть оснащен разъемами, позволяющими быстро выполнять присоединение к нему различного оборудования. В компьютерных (и телекоммуникационных) сетях применяются три основных типа кабеля: кабели на основе скрученных пар медных проводов, коаксиальные кабели с медной жилой, а также волоконно-оптические кабели (первые два типа кабелей называют также медными кабелями).

В зависимости от условий прокладки и эксплуатации кабели делятся на внутренние кабели (кабели зданий) и внешние кабели, которые, в свою очередь, подразделяются на подземные, подводные и кабели воздушной проводки.

Скрученная пара проводов называется витой парой (twisted pair). Скручивание проводов снижает влияние внешних и взаимных помех на полезные сигналы, передаваемые по кабелю. Для неответственных применений внутри здания иногда используются симметричные кабели из нескрученных пар — так называемая «лапша».

Основные особенности конструкции кабелей схематично показаны на рис. 4.3.

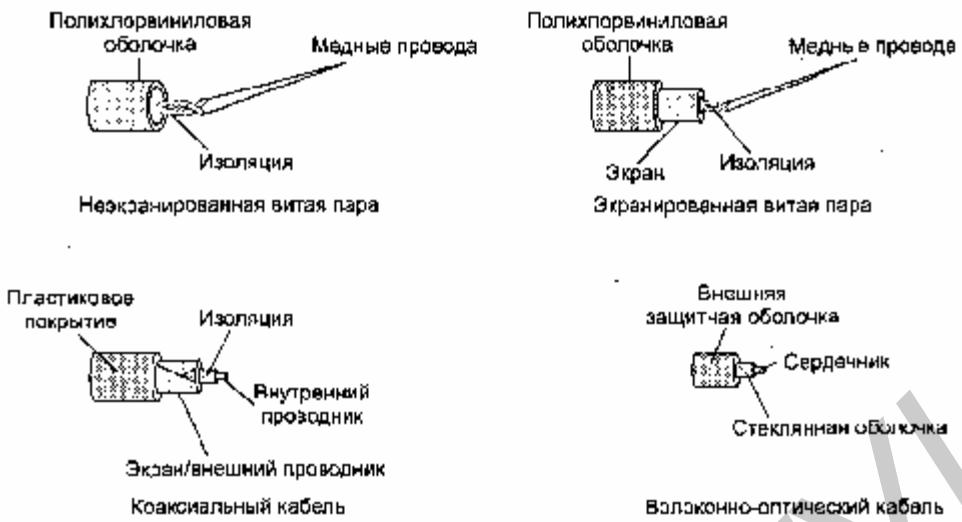


Рис. 4.3. Устройство кабелей

Кабели на основе витой пары называются симметричными кабелями из-за того, что они состоят из двух одинаковых в конструктивном отношении проводников. Симметричный кабель может быть как экранированным — на основе экранированной витой пары (Shielded Twisted Pair, STP), так и неэкранированным — на основе неэкранированной витой пары (Unshielded Twisted Pair, UTP).

Нужно отличать электрическую изоляцию проводящих жил, которая имеется в любом кабеле, от электромагнитной изоляции. Первая состоит из непроводящего диэлектрического слоя — бумаги или полимера, например поливинилхлорида или полистирола. Во втором случае кроме электрической изоляции проводящие жилы помещаются также внутрь электромагнитного экрана, в качестве которого чаще всего применяется проводящая медная оплетка. Симметричный кабель может состоять из нескольких витых пар. В настоящее время кабельные системы зданий чаще всего строятся на основе неэкранированной витой пары, при этом наиболее часто используется, витая пара так называемой категории 5 — в соответствии с классификацией американского национального стандарта для кабелей такого назначения.

Коаксиальный кабель (coaxial) состоит из несимметричных пар проводников. Каждая пара представляет собой внутреннюю медную жилу и соосную с ней внешнюю жилу, которая может быть полой медной трубой или оплеткой, отделенной от внутренней жилы диэлектрической изоляцией. Внешняя жила играет двоякую роль — по ней передаются информационные сигналы, также она является экраном, защищающим внутреннюю жилу от внешних электромагнитных полей. Существует несколько типов коаксиального кабеля, отличающихся характеристиками и областями применения — для локальных компьютерных сетей, для глобальных телекоммуникационных сетей, для кабельного телевидения и т. п.

Волоконно-оптический кабель (optical fiber) состоит из тонких (5-60 микрон) гибких стеклянных волокон (волоконных счетоводов), по которым распространяются световые сигналы. Это наиболее качественный тип кабеля — он обеспечивает передачу данных с очень высокой скоростью (до 10 Гбит/с и выше) и к тому же лучше других типов передающей среды обеспечивает защиту данных

от внешних помех (в силу особенностей распространения света такие сигналы легко экранировать).

Каждый световод состоит из центрального проводника света (сердцевины) — стеклянного волокна, и стеклянной оболочки, обладающей меньшим показателем преломления, чем сердцевина. Распространяясь по сердцевине, лучи света не выходят за ее пределы, отражаясь от покрывающего слоя оболочки. В зависимости от распределения показателя преломления и от величины диаметра сердечника различают:

- многомодовое волокно со ступенчатым изменением показателя преломления (рис. 4.4, а);
- многомодовое волокно с плавным изменением показателя преломления (рис. 4.4, б);
- одномодовое волокно (рис. 4.4, в).

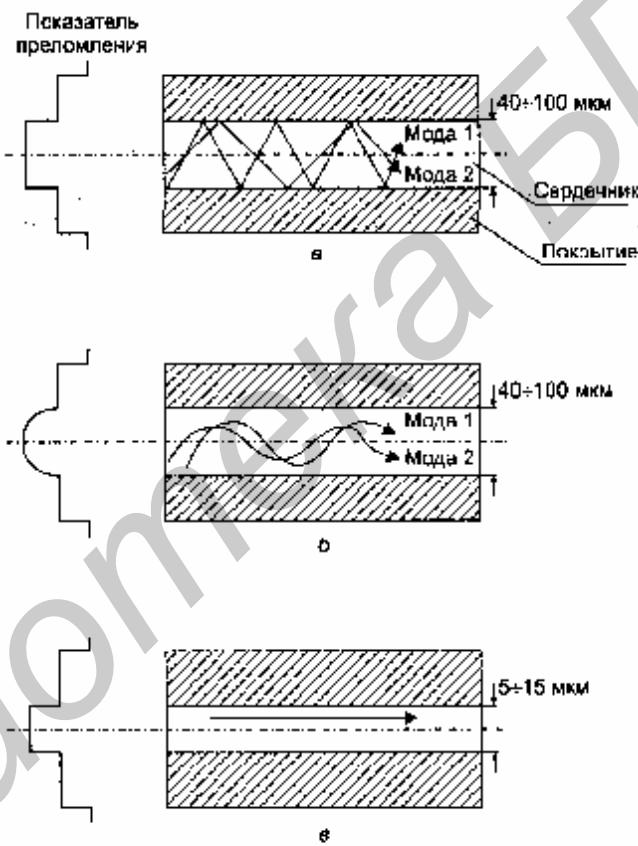


Рис. 4.4. Типы оптического кабеля

Понятие «мода» описывает режим распространения световых лучей во внутреннем сердечнике кабеля. В одномодовом кабеле (Single Mode Fiber, SMF) используется центральный проводник очень малого диаметра, соизмеримого с длиной волны света — от 5 до 10 мкм. При этом практически все лучи света распространяются вдоль оптической оси световода, не отражаясь от внешнего проводника. Изготовление сверхтонких качественных волокон для одномодового кабеля представляет сложный технологический процесс, что делает одномодовый кабель достаточно дорогим. Кроме того, в волокно такого маленького диаметра достаточно сложно направить пучок света, не потеряв при этом зна-

чительную часть его энергии.

В многомодовых кабелях (Multi Mode Fiber, MMF) используются более широкие внутренние сердечники, которые легче изготовить технологически. В стандартах определены два наиболее употребительных многомодовых кабеля: 62,5/125 мкм и 50/125 мкм, где 62,5 мкм или 50 мкм — диаметр центрального проводника, а 125 мкм — диаметр внешнего проводника.

В многомодовых кабелях во внутреннем проводнике одновременно существует несколько световых лучей, отражающихся от внешнего проводника под разными углами. Угол отражения луча называется модой луча. В многомодовых кабелях с плавным изменением коэффициента преломления режим распространения каждой моды имеет сложный характер. Интерференции лучей разных мод ухудшает качество передаваемого сигнала, что приводит к искажениям передаваемых импульсов в многомодовом оптическом волокне. Многомодовые кабели проще изготавливать, поэтому они дешевле одномодовых, но и их характеристики существенно хуже, чем одномодовых. В результате многомодовые кабели используются в основном для передачи данных на небольшие расстояния (до 300-2000 м) на скоростях не более 1 Гбит/с, а одномодовые — для передачи данных со сверхвысокими скоростями в несколько десятков гигабит в секунду (а при использовании технологии DWDM — до нескольких терабит в секунду), на расстояниях от нескольких километров (локальные и городские сети) до нескольких десятков и даже сотен километров (дальняя связь).

В качестве источников излучения света в волоконно-оптических кабелях применяются:

- светодиоды, или светоизлучающие диоды (Light Emitted Diode, LED);
- полупроводниковые лазеры, или лазерные диоды (Laser Diode).

Для одномодовых кабелей применяются только лазерные диоды, так как при таком малом диаметре оптического волокна световой поток, создаваемый светодиодом, невозможно без больших потерь направить в волокно — он имеет чересчур широкую диаграмму направленности излучения, в то время как лазерный диод — узкую. Поэтому более дешевые светодиодные излучатели используются только для многомодовых кабелей.

Волоконно-оптические кабели обладают отличными характеристиками всех типов: электромагнитными, механическими (хорошо гнутся, а в соответствующей изоляции обладают хорошей механической прочностью). Однако у них есть один серьезный недостаток — сложность соединения волокон с разъемами и между собой при необходимости наращивания (увеличения длины) кабеля. Сама стоимость волоконно-оптических кабелей ненамного превышает стоимость кабелей на витой паре, однако проведение монтажных работ с оптоволокном обходится намного дороже из-за трудоемкости операций и высокой стоимости применяемого монтажного оборудования. Так, присоединение оптического волокна к разъему требует проведения высокоточной обрезки волокна в плоскости строго перпендикулярной оси волокна, а также выполнения соединения путем сложной операции склеивания, а не обжатия, как это делается для витой пары. В случае же некачественных соединений резко сужается полоса пропускания волоконно-оптических кабелей и линий.

Радиоканалы наземной и спутниковой связи образуются с помощью передатчика и приемника радиоволн. Существует большое разнообразие типов радиоканалов, отличающихся как используемым частотным диапазоном, так и дальностью канала. Диапазоны коротких, средних и длинных волн (КВ, СВ и ДВ), называемые также диапазонами амплитудной модуляции (Amplitude Modulation, AM) по типу используемого в них метода модуляции сигнала, обеспечивают дальнюю связь, но при невысокой скорости передачи данных. Более скоростными являются каналы, работающие на диапазонах ультракоротких волн (УКВ), для которых характерна частотная модуляция (Frequency Modulation, FM), а также диапазонах сверхвысоких частот (СВЧ, или microwaves). В диапазоне СВЧ (свыше 4 ГГц) сигналы уже не отражаются ионосферой Земли, и для устойчивой связи требуется наличие прямой видимости между передатчиком и приемником. Поэтому такие частоты используют либо спутниковые каналы, либо радиорелейные каналы, где это условие выполняется.

В компьютерных сетях сегодня применяются практически все описанные типы физических сред передачи данных, но наиболее перспективными являются волоконно-оптические кабели. На них сегодня строятся как магистрали крупных территориальных и городских сетей, так и высокоскоростные линии связи локальных сетей. Популярной средой является также витая пара, которая характеризуется отличным отношением качества к стоимости, а также простотой монтажа. Спутниковые каналы и радиосвязь используются чаще всего в тех случаях, когда кабельные линии связи применить нельзя — например, при прохождении канала через малонаселенную местность или же для связи с мобильными пользователями сети. Пока наиболее популярными являются мобильные телефонные сети, а мобильные компьютерные сети представлены сетями радио-Ethernet, имеющими несравненно меньшее распространение. В мобильных сетях нового, так называемого третьего поколения (3d generation, 3G) предусматривается одновременная передача голоса и компьютерных данных, при этом каждый вид трафика считается одинаково важным.

Аппаратура линий связи

Аппаратура передачи данных, или АПД (Data Circuit Terminating Equipment, DCE) в компьютерных сетях непосредственно присоединяет компьютеры или локальные сети пользователя к линии связи и является, таким образом, пограничным оборудованием. Традиционно аппаратуру передачи данных включают в состав линии связи. Примерами DCE являются модемы, терминальные адAPTERы сетей ISDN, устройства подключения к цифровым каналам. Обычно DCE работает на физическом уровне, отвечая за передачу информации в физическую среду (в линию) и прием из нее сигналов нужной формы и мощности.

Аппаратура пользователя линии связи, вырабатывающая данные для передачи по линии связи и подключаемая непосредственно к аппаратуре передачи данных, носит обобщенное название оконечное оборудование данных, или ООД (Data Terminal Equipment, DTE). Примером DTE могут служить компьютеры, коммутаторы или маршрутизаторы. Эту аппаратуру не включают в состав ли-

ний связи.

Разделение оборудования на классы DCE и DTE в локальных сетях является достаточно условным. Например, адаптер локальной сети можно считать как принадлежностью компьютера, то есть DTE, так и составной частью канала связи, то есть DCE. В глобальной сети это разделение становится более четким, при этом устройство DCE обычно относят к глобальной сети, а устройство DCE — уже нет.

Промежуточная аппаратура обычно используется на линиях связи большой протяженности. Она решает две основные задачи:

- улучшение качества сигнала;
 - создание постоянного составного канала связи между двумя абонентами сети.
- В локальных сетях промежуточная аппаратура может совсем не использоваться, если протяженность физической среды — кабелей или радиоэфира — позволяет одному сетевому адаптеру принимать сигналы непосредственно от другого сетевого адаптера без промежуточного усиления. В противном случае применяются устройства типа повторителей и концентраторов.

В глобальных сетях необходимо обеспечить качественную передачу сигналов на расстояния в сотни и тысячи километров. Поэтому без усилителей (повышающих мощность сигналов) и регенераторов (наряду с повышением мощности восстанавливающих форму импульсных сигналов, искажившихся при передаче на большое расстояние), установленных через определенные расстояния, построить территориальную линию связи невозможно. В глобальной сети необходима также и промежуточная аппаратура другого рода — мультиплексоры, демультиплексоры и коммутаторы. Эта аппаратура решает вторую указанную задачу, то есть создает между двумя абонентами сети непрерывный составной канал из отрезков физической среды — кабелей с усилителями. Причем некоторые из этих отрезков, обладающие широкой полосой пропускания, например отрезки волоконно-оптического или коаксиального кабеля, одновременно участвуют в образовании сразу нескольких составных каналов. Такой высокоскоростной канал, по которому передаются одновременно данные от большого числа сравнительно низкоскоростных абонентских линий, обычно называют уплотненным каналом. Наличие промежуточной коммутационной аппаратуры избавляет создателей глобальной сети от необходимости прокладывать отдельную кабельную линию для каждой пары соединяемых узлов сети.

Важно отметить, что приведенные на рис. 4.1 мультиплексор, демультиплексор и коммутатор образуют составной канал на долговременной основе, например на месяц или год, причем абонент не может влиять на процесс коммутации этого канала — указанные устройства управляются по отдельным входам (на рисунке не показаны), абоненту недоступным. А когда нужно образовать новое постоянное соединение между какими-либо двумя конечными узлами сети, находящимися, например, в разных городах, то мультиплексоры, коммутаторы и демультиплексоры настраиваются оператором канала соответствующим образом.

Промежуточная аппаратура канала связи прозрачна для пользователя, он ее не замечает и не учитывает в своей работе. Для него важны только качество полу-

ченного канала в целом, влияющее на скорость и надежность передачи дискретных данных. В действительности же невидимая пользователями промежуточная аппаратура образует сложную сеть. Эту сеть называют первичной сетью, так как сама по себе она никаких высоковолневых служб (например, файловой или передачи голоса) не поддерживает, а только служит основой для построения компьютерных, телефонных или иных сетей, которые иногда называют наложенными, или вторичными, сетями.

В зависимости от типа промежуточной аппаратуры все линии связи делятся на аналоговые и цифровые. В аналоговых линиях промежуточная аппаратура предназначена для усиления аналоговых сигналов, то есть сигналов, которые имеют непрерывный диапазон значений. Такие линии связи традиционно применялись в телефонных сетях для связи АТС между собой. Для создания высокоскоростных каналов, которые мультиплексируют несколько низкоскоростных аналоговых абонентских каналов, при аналоговом подходе обычно используется техника частотного мультиплексирования (Frequency Division Multiplexing, FDM).

В цифровых линиях связи передаваемые сигналы имеют конечное число состояний. Как правило, элементарный сигнал, то есть сигнал, передаваемый за один такт работы передающей аппаратуры, имеет 2, 3 или 4 состояния, которые передаются в линиях связи импульсами или потенциалами прямоугольной формы. С помощью таких сигналов передаются как компьютерные данные, так и оцифрованные речь и изображение (именно из-за общего вида представления информации современными компьютерными, телефонными и телевизионными сетями стали возможны общие первичные сети). В цифровых каналах связи используется специальная промежуточная аппаратура — регенераторы, которые улучшают форму импульсов и обеспечивает их ресинхронизацию, то есть восстанавливают период их следования. Промежуточная аппаратура мультиплексирования и коммутации первичных сетей работает по принципу временного мультиплексирования каналов (Time Division Multiplexing, TDM), когда каждому низкоскоростному каналу выделяется определенная доля времени (тайм-слот, или квант) высокоскоростного канала.

В настоящее время аналоговые каналы стали применяться в первичных сетях нового типа, использующих метод мультиплексирования по длине волны (Wavelength Division Multiplexing, WDM). В первичных сетях WDM каждый канал передает свою информацию с помощью световой волны определенной длины (и, соответственно, частоты). Такой канал также называется спектральным каналом, так как ему выделяется определенная полоса спектра светового излучения. Аппаратура передачи дискретных компьютерных данных по аналоговым линиям связи существенно отличается от аппаратуры такого же назначения, предназначеннй для работы с цифровыми линиями. Аналоговая линия связи предназначена для передачи сигналов произвольной формы и не предъявляет никаких требований к способу представления единиц и нулей аппаратурой передачи данных (это справедливо для сетей FDM и WDM/DWDM), а в цифровой — все параметры передаваемых линией импульсов стандартизованы. Другими словами, на цифровых линиях связи протокол физического уровня опре-

делен, а на аналоговых линиях — нет (есть и исключения из этого правила, некоторые сети DWDM для передачи информации по спектральному каналу требуют цифрового кодирования определенного вида).

Выводы

- Линии связи могут использовать, кроме кабеля, промежуточную аппаратуру, прозрачную для пользователей. Промежуточная аппаратура выполняет две основные функции: усиливает сигналы и обеспечивает постоянную коммутацию между парой пользователей линии.
- В зависимости от типа промежуточной аппаратуры линии связи делятся на аналоговые и цифровые. В аналоговых линиях связи для уплотнения низкоскоростных каналов абонентов в общий высокоскоростной канал используется метод разделения частот (FDM), а в цифровых — метод разделения времени (TDM). В последнее время все большее распространение получает аппаратура мультиплексирования информации по длине волны — WDM и DWDM, позволяющая передавать данные со скоростью в несколько терабит в секунду.
- Кабели на основе витой пары делятся на неэкранированные (UTP) и экранированные (STP). Кабели UTP проще в изготовлении и монтаже, зато кабели STP обеспечивают более высокий уровень защищенности.
- Волоконно-оптические кабели обладают отличными электромагнитными и механическими характеристиками, недостаток их состоит в сложности и высокой стоимости монтажных работ.
- Волоконно-оптические кабели делятся на многомодовые и одномодовые. Многомодовые кабели применяются для передачи данных на небольшие расстояния со скоростью до 1 гигабит в секунду. Одномодовые кабели являются основой сверхскоростных дальних линий связи, обеспечивающих передачу информации со скоростями в сотни-тысячи гигабит в секунду на расстояние в несколько сот километров.

Характеристики линий связи

Группы характеристик линий связи

К основным характеристикам линий связи относятся параметры распространения и параметры влияния. Первые характеризуют процесс распространения полезного сигнала в зависимости от внутренних параметров линии, например погонной индуктивности медного кабеля. Вторые описывают степень влияния на полезный сигнал других сигналов — внешних помех, помех от других пар проводников в медном кабеле. Те и другие характеристики важны, так как сигнал на выходе линии связи всегда является результатом воздействия на исходный сигнал как внутренних, так и внешних факторов.

В каждой из этих групп можно выделить первичные и вторичные параметры.

Первичные параметры описывают физическую природу линии связи, например погонное активное сопротивление, погонную индуктивность, погонную емкость и погонную проводимость изоляции медного кабеля, или же зависимость коэффициента преломления оптического волокна от расстояния от оптической оси. Вторичные параметры выражают некоторый обобщенный результат процесса распространения сигнала по линии связи и не зависят от ее природы. Например, важным вторичным параметром распространения любой линии связи является степень ослабления мощности сигнала при прохождении им определенного расстояния вдоль линии связи — так называемое затухание сигнала. Для медных кабелей не менее важен и такой вторичный параметр влияния, как степень ослабления помехи от соседней витой пары, — он позволяет оценить, не будут ли вызывать передаваемые по одной паре сигналы ложное срабатывание приемника, подключенного к соседней паре на той же стороне кабеля, что и передатчик.

При описании вторичных параметров, подходя к линии связи как к кибернетическому «черному ящику», мы не строим внутреннюю модель этой физической системы, а подаем на нее некоторые эталонные воздействия и по отклику строим нужную вторичную характеристику.

Такой подход позволяет достаточно просто и однотипно определять характеристики линий связи любой природы, не прибегая к сложным теоретическим исследованиям и построением аналитических моделей. Чаще всего в качестве эталонных сигналов для исследования реакций линий связи используются синусоидальные сигналы различных частот. Это связано с тем, что сигналы подобного типа часто встречаются в технике и с их помощью можно представить любую функцию времени — как непрерывный процесс колебаний звука или изменения изображения, так и прямоугольные импульсы, соответствующие дискретной информации компьютера.

Спектральный анализ сигналов на линиях связи

Из теории гармонического анализа известно, что любой периодический процесс можно представить в виде суммы синусоидальных колебаний различных частот и различных амплитуд (рис. 4.5). Каждая составляющая синусоида называется также гармоникой, а набор всех гармоник называют спектральным разложением исходного сигнала. Непериодические сигналы можно представить в виде интеграла синусоидальных сигналов с непрерывным спектром частот. Например, спектральное разложение идеального импульса (единичной мощности и пулевой длительности) имеет составляющие всего спектра частот, от $-\infty$ до $+\infty$ (рис. 4.6).

Техника нахождения спектра любого исходного сигнала хорошо известна. Для некоторых сигналов, которые описываются аналитически (например, для последовательности прямоугольных импульсов одинаковой длительности и амплитуды), спектр легко вычисляется на основании формул Фурье.

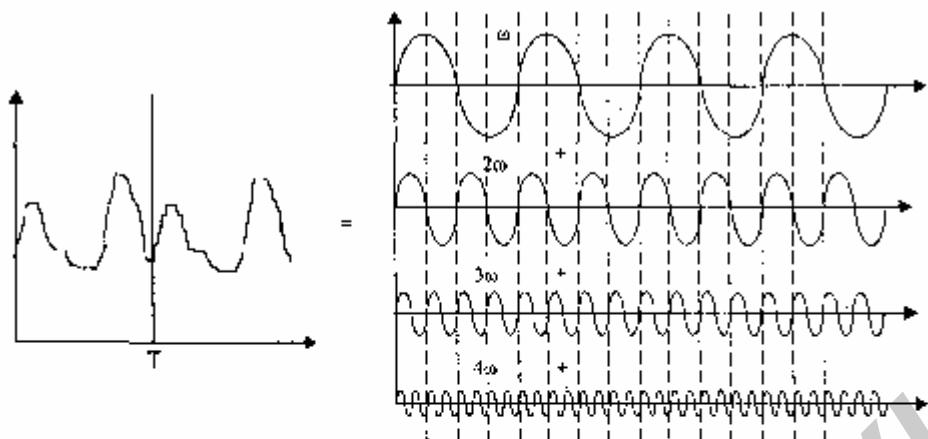


Рис. 4.5. Представления периодического сигнала суммой синусоид

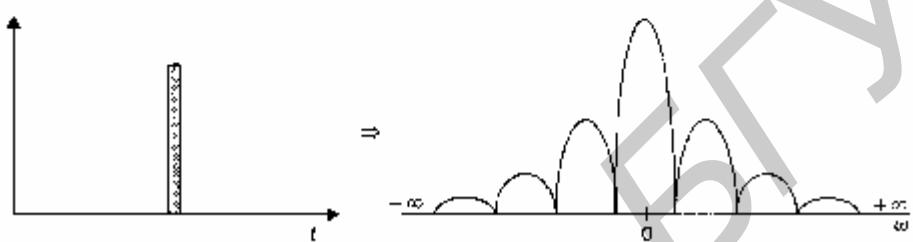


Рис. 4.6. Спектральное разложение идеального импульса

Для сигналов произвольной формы, встречающихся на практике, спектр можно найти с помощью специальных приборов — спектральных анализаторов, которые измеряют спектр реального сигнала и отображают амплитуды составляющих гармоник на экране, распечатывают их на принтере или передают для обработки и хранения в компьютер.

Искажение передающим каналом синусоиды какой-либо частоты приводит в конечном счете к искажению амплитуды и формы передаваемого сигнала любого вида. Искажения формы проявляются в том случае, когда синусоиды различных частот искажаются неодинаково. Если это аналоговый сигнал, передающий речь, то изменяется тембр голоса за счет искажения обертонов — боковых частот. При передаче импульсных сигналов, характерных для компьютерных сетей, искажаются низкочастотные и высокочастотные гармоники, в результате фронты импульсов теряют свою прямоугольную форму (рис. 4.7). Вследствие этого на приемном конце линии сигналы могут плохо распознаваться.

Линия связи искажает передаваемые сигналы из-за того, что ее физические параметры отличаются от идеальных. Так, например, медные провода всегда представляют собой некоторую распределенную по длине комбинацию активного сопротивления, емкостной и индуктивной нагрузок (рис. 4.8). В результате для синусоид различных частот линия будет обладать разным полным сопротивлением, а значит, и передаваться они будут по-разному. Волоконно-оптический кабель также имеет отклонения от идеальной среды передачи света — вакуума. Если линия связи включает промежуточную аппаратуру, то последняя также может вносить дополнительные искажения, так как невозможно создать устройства, которые бы одинаково хорошо передавали весь спектр си-

нусоид, от нуля до бесконечности.

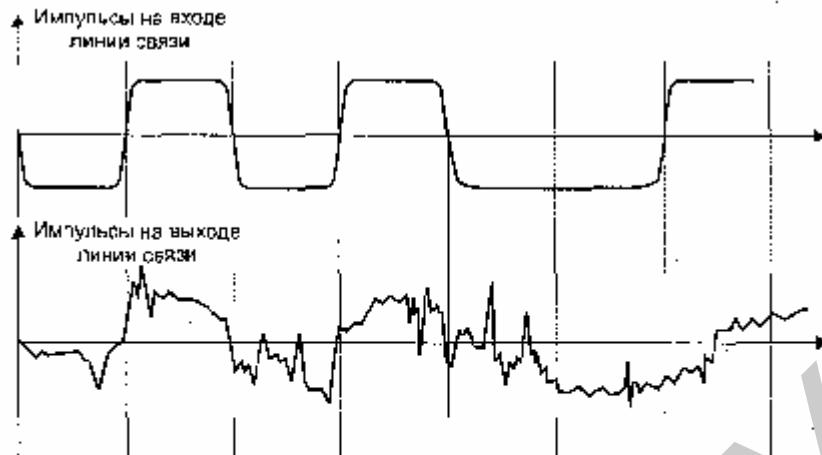


Рис. 4.7. Искажение импульсов в линии связи

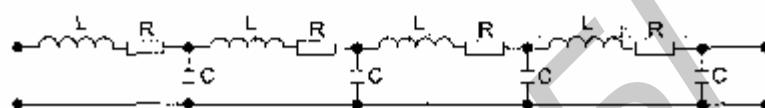


Рис. 4.8. Представление линии как распределенной индуктивно-емкостной нагрузки

Кроме искажений сигналов, вносимых внутренними физическими параметрами линии связи, существуют и внешние помехи, которые вносят свой вклад и искажение формы сигналов на выходе линии. Эти помехи создают различные электрические двигатели, электронные устройства, атмосферные явления и т. д. Несмотря на защитные меры, предпринимаемые разработчиками кабелей, и наличие усилительно-коммутирующей аппаратуры, полностью компенсировать влияние внешних помех не удается. Кроме внешних помех в кабеле существуют и внутренние помехи — так называемые наводки одной пары проводников на другую. В результате сигналы на выходе линии связи обычно имеют сложную форму (как это и показано на рис. 4.7), по которой иногда трудно понять, какая дискретная информация была подана на вход линии.

Качество исходных сигналов (крутизна фронтов, общая форма импульсов) зависит от качества передатчика, генерирующего сигналы в линию связи. Одной из важных характеристик передатчика является спектральная характеристика, то есть спектральное разложение генерируемых им сигналов. Для генерации качественных прямоугольных импульсов необходимо, чтобы спектральная характеристика передатчика представляла собой как можно более узкую полосу. Например, лазерные диоды имеют значительно меньшую ширину спектра излучения (1-2 нм) по сравнению со светодиодами (30-50 нм) при генерации импульсов, поэтому частота модуляции лазерных диодов может быть намного выше, чем светодиодов.

Затухание и волновое сопротивление

Степень искажения синусоидальных сигналов линиями связи оценивается по таким характеристикам, как затухание и полоса пропускания.

Затухание показывает, насколько уменьшается мощность эталонного синусоид-

дальнего сигнала на выходе линии связи по отношению к мощности сигнала на входе этой линии. Затухание A обычно измеряется в децибелах, дБ (decibel, dB) и вычисляется по следующей формуле:

$$A = 10 \log_{10} P_{\text{вых}} / P_{\text{вх}}.$$

Здесь $P_{\text{вых}}$ — мощность сигнала на выходе линии, $P_{\text{вх}}$ — мощность сигнала на входе линии.

Так как мощность выходного сигнала кабеля без промежуточных усилителей всегда меньше, чем мощность входного сигнала, затухание кабеля всегда является отрицательной величиной.

Степень затухания мощности синусоидального сигнала при прохождении им по линии связи обычно зависит от частоты синусоиды, поэтому полной характеристикой будет зависимость затухания от частоты во всем представляющем для практики диапазоне (рис. 4.9). Наряду с этой характеристикой можно также использовать такие характеристики линии связи, как амплитудно-частотная и фазо-частотная зависимости. Две последние характеристики дают более точное представление о характере передачи сигналов через линию связи, чем характеристика затухания, так как на их основе, зная форму исходного сигнала, всегда можно найти форму выходного сигнала.

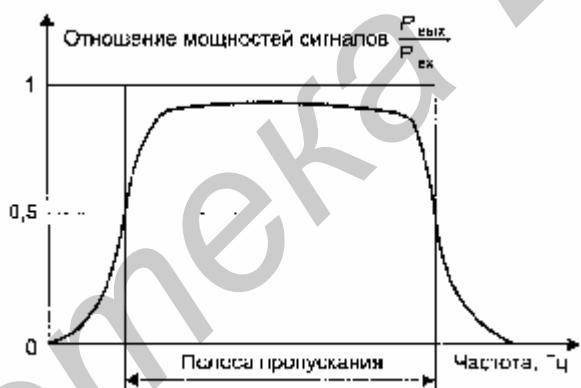


Рис. 4.9. Зависимость затухания от частоты

Для этого необходимо найти спектр входного сигнала, преобразовать амплитуду составляющих его гармоник в соответствии с амплитудно-частотной характеристикой, а фазу — в соответствии с фазо-частотной характеристикой, а затем найти форму выходного сигнала, сложив или проинтегрировав преобразованные гармоники.

Затухание является более обобщенной характеристикой линии связи, так как позволяет судить не о точной форме сигнала, а о его мощности (интегральной результирующей от формы сигнала). Но на практике затухание чаще используется в качестве характеристики линий связи, в частности, в стандартах на такую важную составляющую линии связи, как кабель, затухание является одной из основных характеристик.

Чаще всего при описании параметров линии связи приводятся значения затухания всего в нескольких точках общей зависимости, при этом каждая из этих точек соответствует определенной частоте, на которой измеряется затухание. Отдельное значение затухания называют коэффициентом затухания. Применение

всего нескольких значений вместо полной характеристики связано, с одной стороны, со стремлением упростить измерения при проверке качества линии, а с другой стороны, на практике часто заранее известна основная частота передаваемого сигнала, то есть та частота, гармоника которой имеет наибольшую амплитуду и мощность. Поэтому достаточно знать затухание на этой частоте, чтобы приблизительно оценить искажения передаваемых по линии сигналов. Более точные оценки возможны при знании затухания на нескольких частотах, соответствующих нескольким основным гармоникам передаваемого сигнала.

Чем меньше затухание, тем выше качество линии связи. Обычно затуханием характеризуют пассивные участки линии связи, состоящие из кабелей и кроссовых секций, без усилителей и регенераторов. Например, кабель для внутренней проводки в зданиях на витой паре категории 5, на которой работают практически все технологии локальных сетей, характеризуется затуханием не ниже -23,6 дБ для частоты 100 МГц при длине кабеля 100 м.

Частота 100 МГц выбрана потому, что кабель этой категории предназначен для высокоскоростной передачи данных, сигналы которых имеют значимые гармоники с частотой примерно 100 МГц. Более качественный кабель категории 6 должен уже иметь на частоте 100 МГц затухание не ниже -20,6 дБ, то есть и меньшей степени снижать мощность сигнала. Часто оперируют с абсолютными значениями затухания, опуская его знак, так как затухание всегда отрицательно для пассивного, не содержащего усилители и регенераторы, участка линии, например непрерывного кабеля.

На рис. 4.10 показаны типовые зависимости затухания от частоты для кабелей на неэкранированной витой паре категорий 5 и 6.

Оптический кабель имеет существенно более низкие (по абсолютной величине) величины затухания, обычно в диапазоне от 0,2 до 3 дБ при длине кабеля в 1000 м. Практически все оптические волокна имеют сложную зависимость затухания от длины волны, которая имеет три так называемых «окна прозрачности». На рис. 4.11 показана характерная зависимость затухания для оптического волокна. Из рисунка видно, что область эффективного использования современных волокон ограничена волнами длин 850 нм, 1300 нм и 1550 нм, при этом окно 1550 нм обеспечивает наименьшие потери, а значит, максимальную дальность при фиксированной мощности передатчика и фиксированной чувствительности приемника. Выпускаемый многомодовый кабель обладает двумя первыми окнами прозрачности, то есть 850 нм и 1300 нм, а одномодовый кабель — двумя окнами прозрачности в диапазонах 1310 нм и 1550 нм.

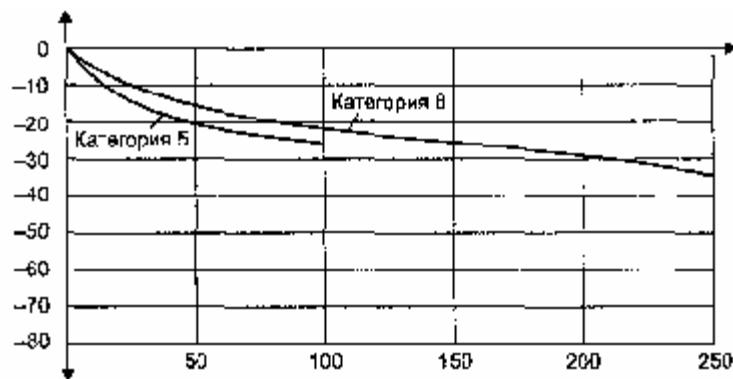


Рис. 4.10. Затухание незащищенного кабеля на витой паре

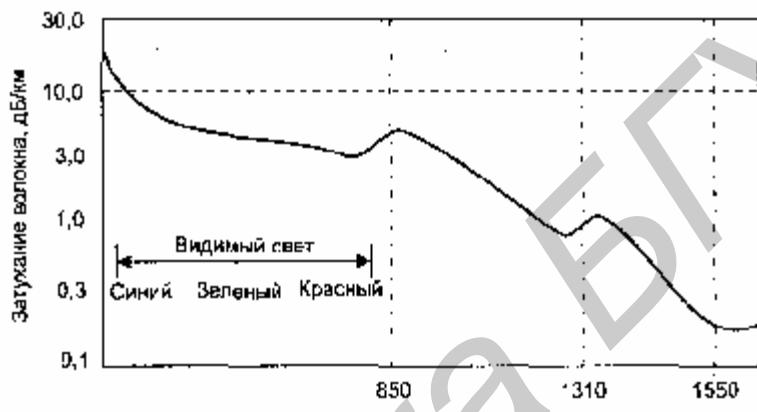


Рис. 4.11. Окна прозрачности оптического волокна

В качестве характеристики мощности передатчика часто используется абсолютный уровень мощности сигнала. Уровень мощности, как и затухание, измеряется в децибелах. При этом в качестве базового значения мощности сигнала, относительно которого измеряется текущая мощность, принимается значение в 1 мВт. Таким образом, уровень мощности p вычисляется по следующей формуле:

$$p = 10 \lg P/1 \text{ мВт} [\text{дБм}].$$

Здесь P — мощность сигнала в милливаттах, а дБм (dBm) — единица измерения уровня мощности (децибел на 1 мВт).

Важным вторичным параметром распространения медной линии связи является ее волновое сопротивление. Этот параметр представляет собой полное (комплексное) сопротивление, которое встречает электромагнитная волна определенной частоты при распространении вдоль однородной цепи. Волновое сопротивление измеряется в Омах и зависит от таких первичных параметров линии связи, как активное сопротивление, погонная индуктивность и погонная емкость, а также от частоты самого сигнала. Выходное сопротивление передатчика должно быть согласовано с волновым сопротивлением линии, иначе затухание сигнала будет чрезмерно большим.

Помехоустойчивость и достоверность

Помехоустойчивость линии определяет ее способность уменьшать уровень помех, создаваемых во внешней среде или на внутренних проводниках самого кабеля. Помехоустойчивость линии зависит от типа используемой физической среды, а также от экранирующих и подавляющих помехи средств самой линии. Наименее помехоустойчивыми являются радиолинии, хорошей устойчивостью обладают кабельные линии и отличной — волоконно-оптические линии, мало чувствительные к внешнему электромагнитному излучению. Обычно для уменьшения помех, появляющихся из-за внешних электромагнитных полей, проводники экранируют и/или скручивают. Параметры, характеризующие помехоустойчивость, относятся к параметрам влияния линии связи.

Первичными параметрами влияния медного кабеля являются электрическая и магнитная связь. Электрическая связь определяется отношением наведенного тока в цепи, подверженной влиянию, к напряжению, действующему во влияющей цепи. Магнитная связь — это отношение электродвижущей силы, наведенной в цепи, подверженной влиянию, к току во влияющей цепи. Результатом электрической и магнитной связи являются наведенные сигналы (наводки) в цепи, подверженной влиянию. Существует несколько различных параметров, характеризующих устойчивость кабеля к наводкам.

Перекрестные наводки на ближнем конце (Near End Cross Talk, NEXT) определяют устойчивость кабеля в том случае, когда наводка образуется в результате действия сигнала, генерируемого передатчиком, подключенным к одной из соседних пар на том же конце кабеля, на котором работает подключенный к подверженной влиянию паре приемник (рис. 4.12). Показатель NEXT, выраженный в децибелах, равен $10 \lg P_{\text{вых}} / P_{\text{нав}}$ — мощность выходного сигнала, $P_{\text{нав}}$ — мощность наведенного сигнала.

Чем меньше значение NEXT, тем лучше кабель. Так, для витой пары категории 5 показатель NEXT должен быть меньше -27 дБ на частоте 100 МГц.

Перекрестные наводки на дальнем конце (Far End Cross Talk, FEXT) позволяют оценить устойчивость кабеля к наводкам для случая, когда передатчик и приемник подключены к разным концам кабеля. Очевидно, что этот показатель должен быть лучше, чем NEXT, так как до дальнего конца кабеля сигнал приходит ослабленный затуханием каждой пары.

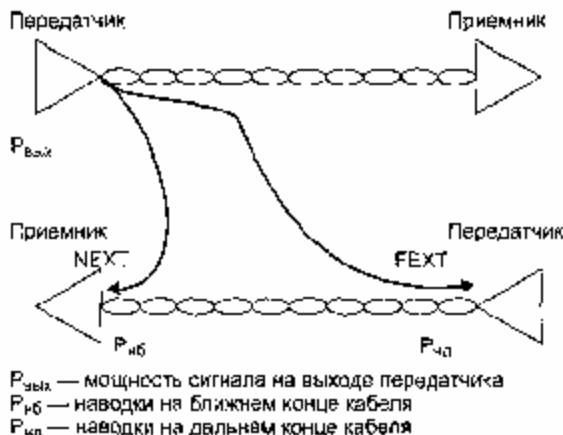


Рис. 4.12. Переходное затухание

Показатели NEXT и FEXT обычно используются применительно к кабелю, состоящему из нескольких витых пар, так как в этом случае взаимные наводки одной пары на другую могут достигать значительных величин. Для одинарного коаксиального кабеля (то есть состоящего из одной экранированной жилы) этот показатель не имеет смысла, а для двойного коаксиального кабеля он также не применяется вследствие высокой степени защищенности каждой жилы. Оптические волокна также не создают сколько-нибудь заметных помех друг для друга.

В связи с тем, что в некоторых новых технологиях используется передача данных одновременно по нескольким витым парам, в последнее время стали применяться также показатели перекрестных наводок с приставкой PS (PowerSUM), такие как PS NEXT и PS FEXT. Эти показатели отражают устойчивость кабеля к суммарной мощности перекрестных наводок на одну из пар кабеля от всех остальных передающих пар (рис. 4.13).

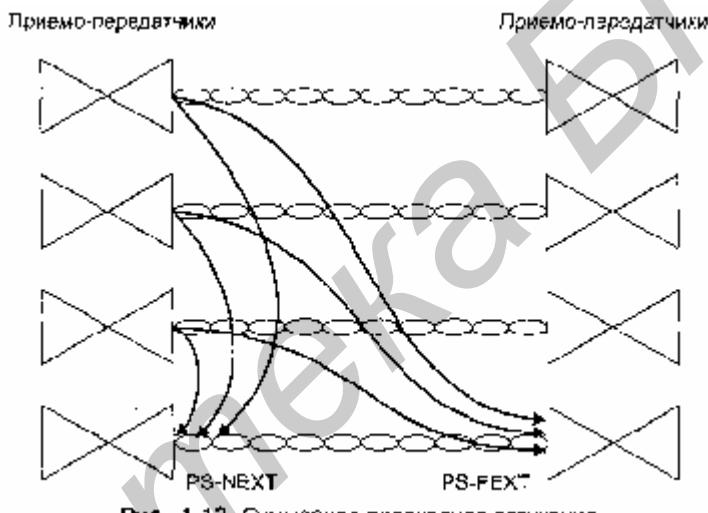


Рис. 4.13. Суммарное переходное затухание

Применяется также такой практически важный показатель, как защищенность кабеля (ACR). Защищенностью определяется как разность между уровнями полезного сигнала и помех. Чем больше значение защищенности кабеля, тем в соответствии с формулой Шеннона с потенциально более высокой скоростью можно передавать данные по этому кабелю. На рис. 4.14 показаны типичная характеристика зависимости защищенности кабеля на неэкранированной витой паре от частоты сигнала.

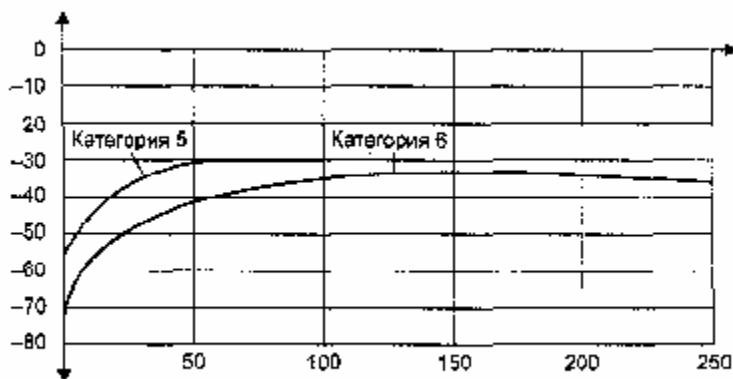


Рис. 4.14. Защищенность витой пары

Достоверность передачи данных характеризует вероятность искажения для каждого передаваемого бита данных. Иногда этот же показатель называют интенсивностью битовых ошибок (Bit Error Rate, BER). Величина BER для каналов связи без дополнительных средств защиты от ошибок (например, самокорректирующихся кодов или протоколов с повторной передачей искаженных кадров) составляет, как правило, 10^{-4} - 10^{-6} , в оптоволоконных линиях связи — 10^{-9} . Значение достоверности передачи данных, например, в 10^{-4} говорит о том, что в среднем из 10 000 бит искажается значение одного бита.

Искажения бит происходят как из-за наличия помех на линии, так и по причине искажений формы сигнала ограниченной полосой пропускания линии. Поэтому для повышения достоверности передаваемых данных нужно повышать степень помехозащищенности линии, снижать уровень перекрестных наводок в кабеле, а также использовать более широкополосные линии связи.

Полоса пропускания

Полоса пропускания — это еще одна вторичная характеристика, которая, с одной стороны, непосредственно зависит от затухания, а с другой стороны, прямо влияет на такой важнейший показатель линии связи, как максимально возможная скорость передачи информации.

Полоса пропускания (bandwidth) — это непрерывный диапазон частот, для которого затухание не превышает некоторый заранее заданный предел. То есть полоса пропускания определяет диапазон частот синусоидального сигнала, при которых этот сигнал передается по линии связи без значительных искажений (часто граничными частотами считаются частоты, на которых мощность выходного сигнала уменьшается в два раза по отношению к входному, что соответствует затуханию в -3 дБ). Как мы увидим ниже, ширина полосы пропускания в наибольшей степени влияет на максимально возможную скорость передачи информации по линии связи. Именно этот факт нашел отражение в английском эквиваленте рассматриваемого термина (width — ширина).

Таким образом, амплитудно-частотная характеристика, полоса пропускания и затухание являются универсальными характеристиками, и их знание позволяет сделать вывод о том, как через линию связи будут передаваться сигналы любой формы.

Полоса пропускания зависит от типа линии и ее протяженности. На рис. 4.15 показаны полосы пропускания линий связи различных типов, а также наиболее часто используемые в технике связи частотные диапазоны.

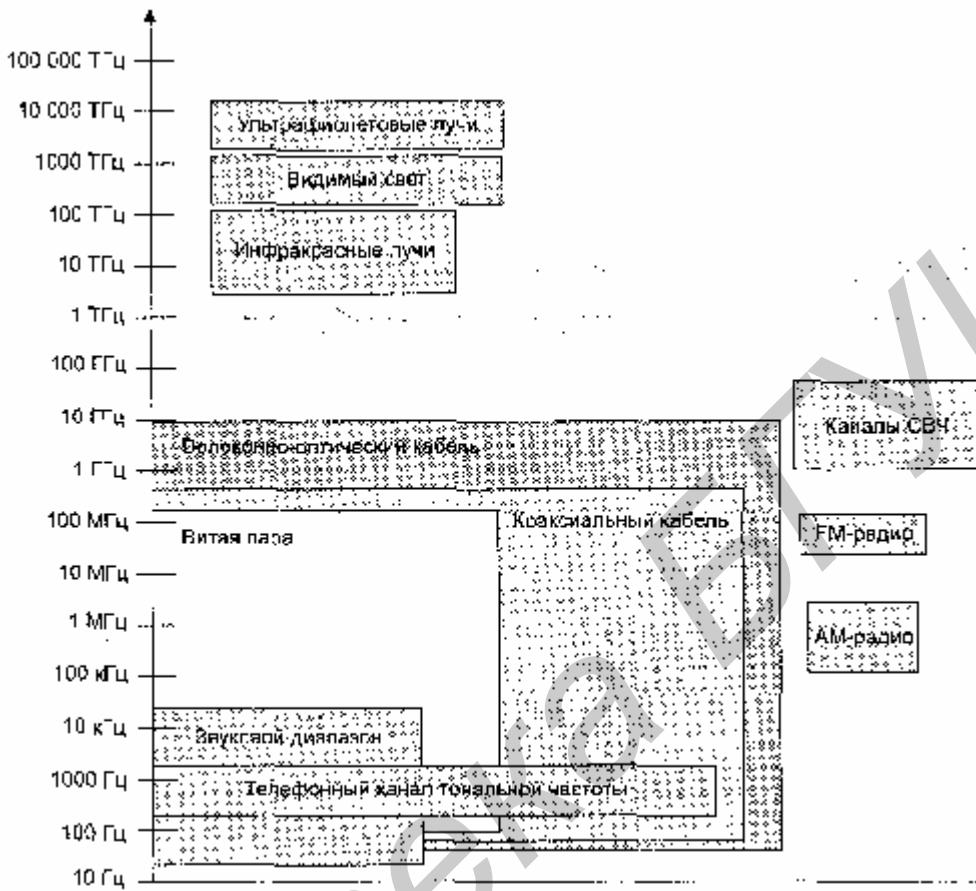


Рис. 4.15. Полосы пропускания линий связи и популярные частотные диапазоны

Выводы

- Параметры, характеризующие линию связи, делятся на параметры передачи и параметры влияния. Параметры передачи описывают процесс распространения информационного сигнала вдоль линии связи в зависимости от физических характеристик линии связи. Параметры влияния описывают степень влияния на процесс распространения информационного сигнала различного рода помех, возникающих как во внешней по отношению к кабелю среде, так и помех, создаваемых соседними передающими парами проводников данного кабеля.
- Как параметры передачи, так и параметры влияния можно разделить на первичные и вторичные. Первичные параметры представляют собой физические параметры, описывающие физическую природу линии связи. К первичным параметрам передачи медного кабеля относятся: активное погонное сопротивление, погонная индуктивность, погонная емкость и погонная проводимость изоляции медного кабеля, к вторичным — затухание и волновое сопротивление.
- Переходное затухание является одним из наиболее значимых вторичных параметров влияния медных линий связи.
- Полоса пропускания определяет диапазон частот, которые передаются лини-

ей связи с приемлемым затуханием.

Пропускная способность

Определение пропускной способности

Пропускная способность (количество бит информации, передаваемое в единицу времени) и достоверность передачи данных (вероятность доставки неискаженного бита или же вероятность искажения бита) интересуют разработчика компьютерной сети в первую очередь, поскольку эти характеристики прямо влияют на производительность и надежность создаваемой сети.

Пропускная способность и достоверность передачи данных зависят, с одной стороны, от характеристик физической среды, а с другой — определяются характеристиками способа передачи данных. Следовательно, нельзя говорить о пропускной способности линии связи, до того как для нее определен протокол физического уровня. Например, поскольку для цифровых линий всегда определен протокол физического уровня, задающий битовую скорость передачи данных, то для них всегда известна и пропускная способность — 64 кбит/с, 2 Мбит/с и т. п.

В тех же случаях, когда только предстоит определить, какой из множества существующих протоколов можно использовать на данной линии, очень важными являются остальные характеристики линии, такие как полоса пропускания, перекрестные наводки, помехоустойчивость и другие характеристики.

пропускная способность (*throughput*) линии характеризует максимально возможную скорость передачи данных по линии связи. Пропускная способность измеряется в битах в секунду (бит/с), а также в производных единицах, таких как килобит в секунду (кбит/с), мегабит в секунду (Мбит/с), гигабит в секунду (Гбит/с) и т. д.

ПРИМЕЧАНИЕ

Пропускная способность линий связи и коммуникационного сетевого оборудования традиционно измеряется в битах в секунду, а не в байтах в секунду. Это связано с тем, что данные в сетях передаются последовательно, то есть побитно, а не параллельно, байтами, как это происходит между устройствами внутри компьютера. Такие единицы измерения, как килобит, мегабит или гигабит, в сетевых технологиях строго соответствуют степеням 10 (то есть килобит — это 1000 бит, а мегабит — это 1000000 бит), как это принято во всех отраслях науки и техники, а не близким к этим числам степеням 2, как это принято в программировании, где приставка «кило»- равна $2^{10} = 1024$, а « mega » — $2^{20} = 1048576$.

Пропускная способность линии связи зависит не только от ее характеристик, таких как затухание и полоса пропускания, но и от спектра передаваемых сиг-

налов. Если значимые гармоники сигнала (то есть те гармоники, амплитуды которых вносят основной вклад в результирующий сигнал) попадают в полосу пропускания линии, то такой сигнал будет хорошо передаваться данной линией связи и приемник сможет правильно распознать информацию, отправленную по линии передатчиком (рис. 4.16, а). Если же значимые гармоники выходят за границы полосы пропускания линии связи, то сигнал будет значительно исказяться, приемник будет ошибаться при распознавании информации, а значит, информация не сможет передаваться с заданной пропускной способностью (рис. 4.16, б).

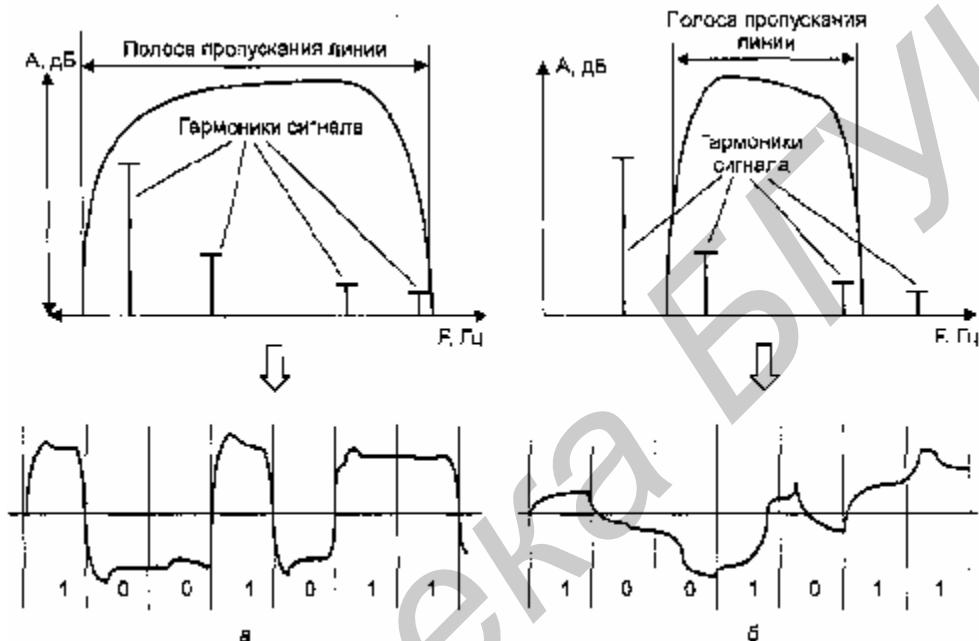


Рис. 4.16. Соответствие между полосой пропускания линии связи и спектром сигнала

Выбор способа представления дискретной информации в виде сигналов, подаваемых на линию связи, называется физическим, или линейным, кодированием. От выбранного способа кодирования зависит спектр сигналов и, соответственно, пропускная способность линии. Таким образом, для одного способа кодирования линия может обладать одной пропускной способностью, а для другого — другой. Например, витая пара категории 3 может передавать данные с пропускной способностью 10 Мбит/с при способе кодирования стандарта физического уровня 10Base-T и 33 Мбит/с при способе кодирования стандарта 100Base-T4. В примере, приведенном на рис. 4.16, принят следующий способ кодирования: логическая 1 представлена па линии положительным потенциалом, а логический 0 — отрицательным.

Теория информации говорит, что любое различимое и непредсказуемое изменение принимаемого сигнала несет в себе информацию. В соответствии с этим прием синусоиды, у которой амплитуда, фаза и частота остаются неизменными, информации не несет, так как изменение сигнала хотя и происходит, но является хорошо предсказуемым. Аналогично, не несут в себе информации импульсы на тактовойшине компьютера, так как их изменения также постоянны во времени. А вот импульсы на шине данных предсказать заранее нельзя, поэтому

они переносят информацию между отдельными блоками или устройствами компьютера.

Большинство способов кодирования используют изменение какого-либо параметра периодического сигнала — частоты, амплитуды и фазы синусоиды или же знак потенциала последовательности импульсов. Периодический сигнал, параметры которого изменяются, называют несущим сигналом или несущей частотой, если в качестве такого сигнала используется синусоида.

Если сигнал изменяется так, что можно различить только два его состояния, то любое его изменение будет соответствовать наименьшей единице информации — биту. Если же сигнал может иметь более двух различных состояний, то любое его изменение будет нести несколько битов информации.

Количество изменений информационного параметра несущего периодического сигнала в секунду измеряется в бодах (baud). Период времени между соседними изменениями информационного сигнала называется тактом работы передатчика.

Пропускная способность линии в битах в секунду в общем случае не совпадает с числом бод. Она может быть как выше, так и ниже числа бод, и это соотношение зависит от способа кодирования.

Если сигнал имеет более двух различных состояний, то пропускная способность в битах в секунду будет выше, чем число бод. Например, если информационными параметрами являются фаза и амплитуда синусоиды, причем различаются 4 состояния фазы в 0, 90, 180 и 270° и два значения амплитуды сигнала, то информационный сигнал может иметь 8 различных состояний. В этом случае modem, работающий со скоростью 2400 бод (с тактовой частотой 2400 Гц), передаст информацию со скоростью 7200 бит/с, так как при одном изменении сигнала передается три бита информации.

При использовании сигналов с двумя различими состояниями может наблюдаться обратная картина. Это часто происходит потому, что для надежного распознавания приемником пользовательской информации каждый бит в последовательности кодируется путем нескольких изменений информационного параметра несущего сигнала. Например, при кодировании единичного значения бита импульсом положительной полярности, а нулевого значения бита импульсом отрицательной полярности физический сигнал дважды изменяет свое состояние при передаче каждого бита. При таком кодировании пропускная способность линии в два раза ниже, чем число бод, передаваемое по линии.

На пропускную способность линии оказывает влияние не только физическое, но и логическое кодирование. Логическое кодирование выполняется до физического кодирования и подразумевает замену битов исходной информации новой последовательностью битов, несущей ту же информацию, но обладающей, кроме этого, дополнительными свойствами, например возможностью для приемной стороны обнаруживать ошибки в принятых данных. Сопровождение каждого байта исходной информации одним битом четности — это пример очень часто применяемого способа логического кодирования при передаче данных с помощью модемов. Другим примером логического кодирования может служить шифрование данных, обеспечивающее их конфиденциальность при передаче

через общественные каналы связи. При логическом кодировании чаще всего исходная последовательность бит заменяется более длинной последовательностью, поэтому пропускная способность канала по отношению к полезной информации при этом уменьшается.

Связь между пропускной способностью и полосой пропускания линии

Чем выше частота несущего периодического сигнала, тем больше информации в единицу времени передается по линии и тем выше пропускная способность линии при фиксированном способе физического кодирования. Однако, с другой стороны, с увеличением частоты периодического несущего сигнала увеличивается и ширина спектра этого сигнала, то есть разность между максимальной и минимальной частотами того набора синусоид, которые в сумме дадут выбранную для физического кодирования последовательность сигналов. Линия передает этот спектр синусоид с теми искажениями, которые определяются ее полосой пропускания. Чем больше несоответствие между полосой пропускания линии и шириной спектра передаваемых информационных сигналов, тем больше сигналы искажаются и тем вероятнее ошибки в распознавании информации принимающей стороной, а значит, скорость передачи информации на самом деле оказывается меньше, чем можно было предположить.

Связь между полосой пропускания линии и ее максимально возможной пропускной способностью, вне зависимости от принятого способа физического кодирования, установил Клод Шеннон:

$$C = F \log_2(1 + P_c/P_{\text{ш}}).$$

Здесь C — максимальная пропускная способность линии в битах в секунду, F — ширина полосы пропускания линии в герцах, P_c — мощность сигнала, $P_{\text{ш}}$ — мощность шума.

Из этого соотношения видно, что хотя теоретического предела пропускной способности линии с фиксированной полосой пропускания не существует, на практике такой предел имеется. Действительно, повысить пропускную способность линии можно за счет увеличения мощности передатчика или же уменьшения мощности шума (помех) на линии связи. Обе эти составляющие поддаются изменению с большим трудом. Повышение мощности передатчика ведет к значительному увеличению его габаритов и стоимости. Снижение уровня шума требует применения специальных кабелей с хорошими защитными экранами, что весьма дорого, а также снижения шума в передатчике и промежуточной аппаратуре, чего достичь весьма не просто. К тому же влияние мощностей полезного сигнала и шума на пропускную способность ограничено логарифмической зависимостью, которая растет далеко не так быстро, как прямопропорциональная. Так, при достаточно типичном исходном отношении мощности сигнала к мощности шума в 100 раз повышение мощности передатчика и два раза даст только 15 % увеличения пропускной способности линии.

Близким по сути к формуле Шеннона является другое соотношение, получен-

ное Найквистом, которое также определяет максимально возможную пропускную способность линии связи, но без учета шума на линии;

$$C=2F \log_2 M.$$

Здесь M — количество различимых состояний информационного параметра.

Если сигнал имеет два различимых состояния, то пропускная способность равна удвоенному значению ширине полосы пропускания линии связи (рис. 4.17, а). Если же передатчик использует более двух устойчивых состояний сигнала для кодирования данных, то пропускная способность линии повышается, так как за один такт работы передатчик передает несколько битов исходных данных, например два бита при наличии четырех различимых состояний сигнала (рис. 4.17, б).

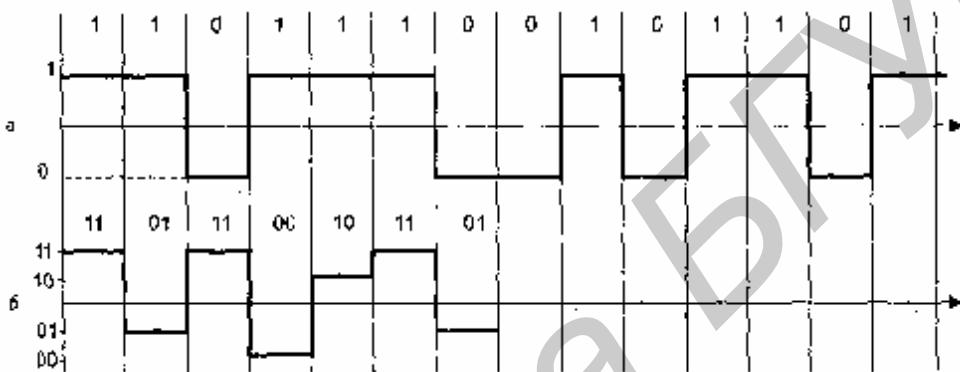


Рис. 4.17. Повышение скорости передачи за счет дополнительных состояний сигнала

Хотя формула Найквиста явно не учитывает наличие шума, косвенно его влияние отражается в выборе количества состояний информационного сигнала. Для повышения пропускной способности канала хотелось бы увеличить это количество до значительных величин, но на практике мы не можем этого сделать из-за шума на линии. Например, для примера, приведенного на рис. 4.17, можно увеличить пропускную способность линии еще в два раза, применив для кодирования данных не 4, а 16 уровней. Однако если амплитуда шума часто превышает разницу между соседними 16-ю уровнями, то приемник не сможет устойчиво распознавать передаваемые данные. Поэтому количество возможных состояний сигнала фактически ограничивается соотношением мощности сигнала и шума, а формула Найквиста определяет предельную скорость передачи данных в том случае, когда количество состояний уже выбрано с учетом возможностей устойчивого распознавания приемником.

Приведенные соотношения дают предельное значение пропускной способности линии, а степень приближения к этому пределу зависит от конкретных методов физического кодирования, рассматриваемых ниже, а также мощности шума, то есть различного рода помех.

Выводы

- Пропускная способность линии связи является одной из ее важнейших потребительских характеристик, так как она говорит о том, с какой максимальной скоростью пользователь может передавать по этой линии дискретные данные.
- Пропускная способность линии связи зависит от ее внутренних параметров, в

частности — полосы пропускания, внешних параметров — уровня помех и степени ослабления помех, а также принятого способа кодирования дискретных данных.

- Формула Шеннона определяет максимально возможную пропускную способность линии связи при фиксированных значениях полосы пропускания линии и отношении мощности сигнала к шуму.
- Формула Найквиста выражает максимально возможную пропускную способность линии связи через полосу пропускания и количество состояний информационного сигнала.

ГЛАВА 5 Передача данных на физическом и канальном уровнях

При передаче дискретных данных по каналам связи применяются два основных типа физического кодирования — на основе синусоидального несущего сигнала и на основе последовательности прямоугольных импульсов. Первый способ часто называется также модуляцией, или аналоговой модуляцией, подчеркивая тот факт, что кодирование осуществляется за счет изменения параметров аналогового сигнала. Второй способ обычно называют цифровым кодированием. Эти способы отличаются шириной спектра результирующего сигнала и сложностью аппаратуры, необходимой для их реализации.

При использовании прямоугольных импульсов спектр результирующего сигнала получается весьма широким. Это не удивительно, если вспомнить, что спектр идеального импульса имеет бесконечную ширину. Применение синусоиды приводит к спектру гораздо меньшей ширины при той же скорости передачи информации. Однако для синусоидальной модуляции требуется более сложная и дорогая аппаратура, чем для генерирования прямоугольных импульсов.

В настоящее время все чаще данные, изначально имеющие аналоговую форму — речь, телевизионное изображение, — передаются по каналам связи в дискретном виде, то есть в виде последовательности единиц и нулей. Процесс представления аналоговой информации в дискретной форме называется дискретной модуляцией. Термины «модуляция» и «кодирование» часто используют как синонимы.

Аналоговая модуляция

Аналоговая модуляция применяется для передачи дискретных данных по каналам с узкой полосой частот, типичным представителем которых является канал тональной частоты, предоставляемый в распоряжение пользователям общественных телефонных сетей. Типичная амплитудно-частотная характеристика канала тональной частоты представлена на рис. 5.1. Этот канал передает частоты в диапазоне от 300 до 3400 Гц, таким образом, его полоса пропускания равна 3100 Гц. Хотя человеческий голос имеет гораздо более широкий спектр — примерно от 100 Гц до 10 кГц, — для приемлемого качества передачи речи диапазон в 3100 Гц является хорошим решением. Строгое ограничение полосы пропускания тонального канала связано с использованием аппаратуры уплотнения и коммутации каналов в телефонных сетях.

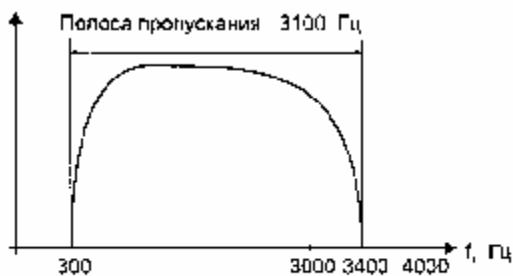


Рис. 5.1. Амплитудно-частотная характеристика канала тональной частоты.

Устройство, которое выполняет функции модуляции несущей синусоиды на передающей стороне и демодуляции на приемной стороне, носит название модем (модулятор- демодулятор).

Методы аналоговой модуляции

При физическом кодировании способом аналоговой модуляции информация кодируется изменением амплитуды, частоты или фазы синусоидального сигнала несущей частоты. Основные способы аналоговой модуляции показаны на рис. 5.2. На диаграмме (рис. 5.2, а) показана последовательность битов исходной информации, представленная потенциалами высокого уровня для логической единицы и потенциалом нулевого уровня для логического нуля. Такой способ кодирования называется потенциальным кодом и часто используется при передаче данных между блоками компьютера.

При амплитудной модуляции (рис. 5.2, б) для логической единицы выбирается один уровень амплитуды синусоиды несущей частоты, а для логического нуля — другой. Этот способ редко используется в чистом виде на практике из-за низкой помехоустойчивости, но часто применяется в сочетании с другим видом модуляции — фазовой модуляцией.

При частотной модуляции (рис. 5.2, б) значения 0 и 1 исходных данных передаются синусоидами с различной частотой — f_0 и f_1 . Этот способ модуляции не требует сложных схем в модемах и обычно применяется в низкоскоростных модемах, работающих на скоростях 300 или 1200 бит/с.

При фазовой модуляции (рис. 5.2, г) значениям данных 0 и 1 соответствуют сигналы одинаковой частоты, но различной фазы, например 0 и 180° или 0, 90, 180 и 270° .

В скоростных модемах часто используются комбинированные методы модуляции, как правило, амплитудная в сочетании с фазовой.

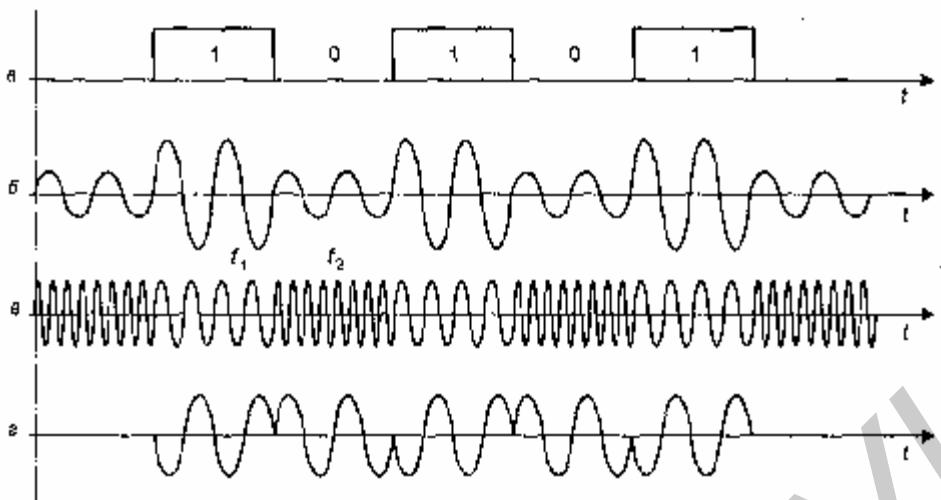


Рис. 5.2. Различные типы модуляции

Спектр модулированного сигнала

Спектр результирующего модулированного сигнала зависит от типа модуляции и скорости модуляции, то есть желаемой скорости передачи битов исходной информации.

Рассмотрим сначала спектр сигнала при потенциальном кодировании. Пусть логическая единица кодируется положительным потенциалом, а логический ноль — отрицательным потенциалом такой же величины. Для упрощения вычислений предположим, что передается информация, состоящая из бесконечной последовательности чередующихся единиц и нулей, как это и показано на рис. 5.2, а. Заметим, что в данном случае величины бод и битов в секунду совпадают.

Для потенциального кодирования спектр непосредственно получается из формул Фурье для периодической функции. Если дискретные данные передаются с битовой скоростью N бит/с, то спектр состоит из постоянной составляющей нулевой частоты и бесконечного ряда гармоник с частотами $f_0, 3f_0, 5f_0, 7f_0, \dots$, где $f_0 = N/2$. Амплитуды этих гармоник убывают достаточно медленно — с коэффициентами $1/3, 1/5, 1/7, \dots$ от амплитуды гармоники f_0 (рис. 5.3, а). В результате спектр потенциального кода требует для качественной передачи широкую полосу пропускания. Кроме того, нужно учесть, что реально спектр сигнала постоянно меняется в зависимости от того, какие данные передаются по линии связи. Например, передача длинной последовательности пуль или единиц сдвигает спектр в сторону низких частот, а в крайнем случае, когда передаваемые данные состоят только из единиц (или только из нулей), спектр состоит из гармоники нулевой частоты. При передаче чередующихся единиц и нулей постоянная составляющая отсутствует. Поэтому спектр результирующего сигнала потенциального кода при передаче произвольных данных занимает полосу от некоторой величины, близкой к 0 Гц, до примерно $7f_0$ (гармониками с частотами выше $7f_0$ можно пренебречь из-за их малого вклада в результирующий сигнал). Для канала тональной частоты верхняя граница при потенциальном коди-

ровании достигается для скорости передачи данных в 971 бит/с, а нижняя неприемлема для любых скоростей, так как полоса пропускания канала начинается с 300 Гц. В результате потенциальные коды на каналах тональной частоты никогда не используются.

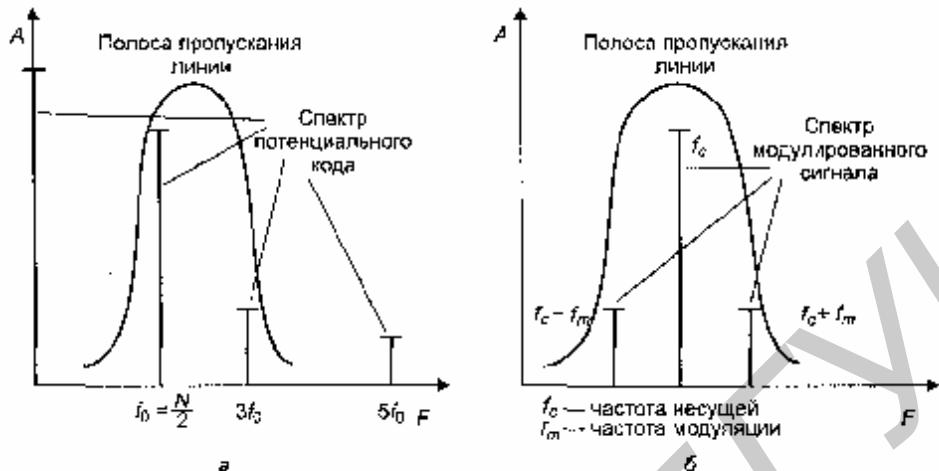


Рис. 5.3. Спектры сигналов при потенциальном кодировании и амплитудной модуляции

При амплитудной модуляции спектр состоит из синусоиды несущей частоты f_c и двух боковых гармоник: $(f_c + f_m)$ и $(f_c - f_m)$, где f_m — частота изменения информационного параметра синусоиды, которая совпадает со скоростью передачи данных при использовании двух уровней амплитуды (рис. 5.3, б). Частота f_m определяет пропускную способность линии при данном способе кодирования. При небольшой частоте модуляции ширина спектра сигнала также оказывается небольшой (равной $2f_m$), поэтому сигналы не будутискажаться линией, если ее полоса пропускания будет больше или равна $2f_m$.

Для канала тональной частоты такой способ модуляции приемлем при скорости передачи данных не больше $3100/2=1550$ бит/с. Если же для представления данных используются 4 уровня амплитуды, то пропускная способность канала повышается до 3100 бит/с.

При фазовой и частотной модуляциях спектр сигнала получается более сложным, чем при амплитудной, так как боковых гармоник здесь образуется более двух, но они тоже симметрично расположены относительно основной несущей частоты, а их амплитуды быстро убывают. Поэтому эти виды модуляции также хорошо подходят для передачи данных по каналу тональной частоты.

Для повышения скорости передачи данных используют комбинированные методы модуляции. Наиболее распространеными являются методы квадратурной амплитудной модуляции (Quadrature Amplitude Modulation, QAM). Эти методы основаны на сочетании фазовой модуляции с 8 значениями величин сдвига фазы и амплитудной модуляции с 4 уровнями амплитуды. Однако из возможных 32 комбинаций сигнала используются далеко не все. Например, в так называемых решетчатых кодах (trellis code) допустимы всего 6, 7 или 8 комбинаций для представления исходных данных, а остальные комбинации являются запрещенными. Такая избыточность кодирования требуется для распознавания модемом ошибочных сигналов, являющихся следствием искажений из-за помех, которые

на телефонных каналах, особенно коммутируемых, весьма значительны по амплитуде и продолжительны по времени.

Дискретная модуляция аналоговых сигналов

Одной из основных тенденций развития сетевых технологий является передача в одной сети как дискретных, так и аналоговых по своей природе данных. Источниками дискретных данных являются компьютеры и другие вычислительные устройства, а источниками аналоговых данных являются такие устройства, как телефоны, видеокамеры, аудио- и видеовоспроизводящая аппаратура. На ранних этапах решения этой проблемы в территориальных сетях все типы данных передавались в аналоговой форме, при этом дискретные по своему характеру компьютерные данные преобразовывались в аналоговую форму с помощью модемов.

Однако по мере развития техники съема и передачи аналоговых данных выяснилось, что передача их в аналоговой форме не позволяет улучшить качество принятых на другом конце линии данных, если они существенно исказились при передаче. Сам аналоговый сигнал не дает никаких указаний ни о том, что произошло искажение, ни о том, как его исправить, поскольку форма сигнала может быть любой, в том числе и такой, которую зафиксировал приемник. Улучшение же качества линий, особенно территориальных, требует огромных усилий и капиталовложений. Поэтому на смену аналоговой технике записи и передачи звука и изображения пришла цифровая техника. Эта техника использует так называемую дискретную модуляцию исходных непрерывных во времени аналоговых процессов.

Дискретные способы модуляции основаны на дискретизации непрерывных процессов, как по амплитуде, так и по времени (рис. 5.4). Рассмотрим принципы дискретной модуляции на примере импульсно-кодовой модуляции, ИКМ (Pulse Amplitude Modulation, PAM), которая широко применяется в цифровой телефонии.

Амплитуда исходной непрерывной функции измеряется с заданным периодом — за счет этого происходит дискретизация по времени. Затем каждый замер представляется в виде двоичного числа определенной разрядности, что означает дискретизацию по значениям функции — непрерывное множество возможных значений амплитуды заменяется дискретным множеством ее значений. Устройство, которое выполняет подобную функцию, называется аналого-цифровым преобразователем (АЦП). После этого замеры передаются по каналам связи в виде последовательности единиц и нулей. При этом применяются те же методы кодирования, что и в случае передачи изначально дискретной информации, то есть, например, методы, основанные на коде B8ZS или 2B1Q.

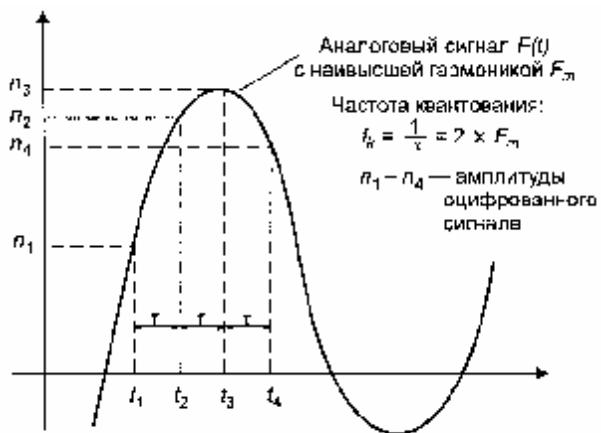


Рис. 5.4. Дискретная модуляция непрерывного процесса.

На приемной стороне линии коды преобразуются в исходную последовательность битов, а специальная аппаратура, называемая цифро-аналоговым преобразователем (ЦАП), производит демодуляцию оцифрованных амплитуд непрерывного сигнала, восстановливая исходную непрерывную функцию времени.

Дискретная модуляции основана на теории отображения Найквиста—Котельникова. В соответствии с этой теорией аналоговая непрерывная функция, переданная в виде последовательности ее дискретных по времени значений, может быть точно восстановлена, если частота дискретизации была в два или более раз выше, чем частота самой высокой гармоники спектра исходной функции.

Если это условие не соблюдается, то восстановленная функция будет существенно отличаться от исходной.

Преимуществом цифровых методов записи, воспроизведения и передачи аналоговой информации является возможность контроля достоверности считанных с носителя или полученных по линии связи данных. Для этого можно применять те же методы, которые применяются для компьютерных данных (и рассматриваются более подробно далее), — вычисление контрольной суммы, повторная передача искаженных кадров, применение самокорректирующихся кодов.

Для качественной передачи голоса в методе ИКМ используется частота квантования амплитуды звуковых колебаний в 8000 Гц. Это связано с тем, что в аналоговой телефонии для передачи голоса был выбран диапазон от 300 до 3400 Гц, который достаточно качественно передает все основные гармоники собеседников. В соответствии с теоремой Найквиста—Котельникова для качественной передачи голоса достаточно выбрать частоту дискретизации, в два раза превышающую самую высокую гармонику непрерывного сигнала, то есть $2 \times 3400 = 6800$ Гц. Выбранная в действительности частота дискретизации 8000 Гц обеспечивает некоторый запас качества. В методе ИКМ обычно используется 7 или 8 бит кода для представления амплитуды одного замера. Соответственно это дает 127 или 256 градаций звукового сигнала, что оказывается вполне достаточным для качественной передачи голоса. При использовании метода ИКМ для передачи одного голосового канала необходима пропускная способность 56 или 64 кбит/с в зависимости от того, каким количеством битов представляется каждый замер. Если для этих целей используется 7 бит, то при частоте передачи

замеров в 8000 Гц получаем:

$$8000 \times 7 = 56000 \text{ бит/с или } 56 \text{ кбит/с;}$$

а для случая 8 бит:

$$8000 \times 8 - 64000 \text{ бит/с или } 64 \text{ кбит/с.}$$

Стандартным является цифровой канал 64 кбит/с, который также, называется элементарным каналом цифровых телефонных сетей.

Передача непрерывного сигнала в дискретном виде требует от сетей жесткого соблюдения временного интервала в 125 мкс (соответствующего частоте дискретизации 8000 Гц) между соседними замерами, то есть требует синхронной передачи данных между узлами сети. При несоблюдении синхронности приывающих замеров исходный сигнал восстанавливается неверно, что приводит к искажению голоса, изображения или другой мультимедийной информации, передаваемой по цифровым сетям. Так, искажение синхронизации в 10 мс может привести к эффекту «эха», а сдвиги между замерами в 200 мс приводят к невозможности распознавания произносимых слов. В то же время потеря одного замера при соблюдении синхронности между остальными замерами практически не оказывается на воспроизведении звука. Это происходит за счет сглаживающих устройств в цифро-аналоговых преобразователях, которые основаны на свойстве инерционности любого физического сигнала — амплитуда звуковых колебаний не может мгновенно измениться на большую величину.

На качество сигнала после ЦАП влияет не только синхронность поступления на его вход замеров, но и погрешность дискретизации амплитуд этих замеров. В теореме Найквиста—Котельникова предполагается, что амплитуды функции измеряются точно, в то же время использование для их хранения двоичных чисел с ограниченной разрядностью несколько искажает эти амплитуды. Соответственно искажается восстановленный непрерывный сигнал, что называется шумом дискретизации (по амплитуде).

Существуют и другие методы дискретной модуляции, позволяющие представить замеры голоса в более компактной форме, например в виде последовательности 4-битных или 2-битных чисел. При этом один голосовой канал требует меньшей пропускной способности, например 32 кбит/с, 16 кбит/с или еще меньше. С 1985 года применяется стандарт CCITT кодирования голоса, называемый Adaptive Differential Pulse Code Modulation (ADPCM). Коды ADPCM основаны на нахождении разностей между последовательными замерами голоса, которые затем и передаются по сети. В коде ADPCM для хранения одной разности используется 4 бит, и голос передается со скоростью 32 кбит/с. Более современный метод, Linear Predictive Coding (LPC), делает замеры исходной функции реже, но использует прогнозирование направления, в котором изменяется амплитуда сигнала. При помощи этого метода можно понизить скорость передачи голоса до 9600 бит/с. Представленные в цифровой форме непрерывные данные легко можно передать через компьютерную сеть. Для этого достаточно поместить несколько замеров в кадр какой-нибудь стандартной сетевой технологии, снабдить кадр правильным адресом назначения и отправить адресату. Адресат должен извлечь из кадра замеры и подать их с частотой квантования (для голоса — с частотой 8000 Гц) на цифро-аналоговый преобразова-

тель. По мере поступления следующих кадров с замерами голоса операция должна повторяться. Если кадры будут прибывать достаточно синхронно, то качество голоса может быть достаточно высоким. Однако, как мы уже знаем, кадры в компьютерных сетях могут задерживаться как в конечных узлах (при ожидании доступа к разделяемой среде), так и в промежуточных коммуникационных устройствах — мостах, коммутаторах и маршрутизаторах. Поэтому качество голоса при передаче в цифровой форме через компьютерные сети обычно бывает невысоким. Для качественной передачи оцифрованных непрерывных сигналов — голоса, изображения — сегодня используют специальные цифровые сети, такие как ISDN, ATM, а также сети цифрового телевидения. Тем не менее для передачи внутрикорпоративных телефонных разговоров сегодня характерны сети frame relay, задержки передачи кадров которых укладываются в допустимые пределы.

Выводы

- При передаче дискретных данных по узкополосному каналу тональной частоты, используемому в телефонии, наиболее подходящими оказываются способы аналоговой модуляции, при которых несущая синусоида модулируется исходной последовательностью двоичных цифр. Эта операция осуществляется специальными устройствами — модемами.
- Для низкоскоростной передачи данных изменяют частоту несущей синусоиды. Более высокоскоростные модемы работают на комбинированных способах квадратурной амплитудной модуляции (QAM), для которой характерны 4 уровня амплитуды несущей синусоиды и 8 уровней фазы. Не все из возможных 32 сочетаний метода QAM используются для передачи данных, запрещенные сочетания позволяют распознавать искаженные данные на физическом уровне.
- Аналоговая информация также может передаваться по телекоммуникационным сетям в цифровой форме. Это повышает качество передачи, так как при этом могут применяться эффективные методы обнаружения и исправления ошибок, недоступные для систем аналоговой передачи.
- Для качественной передачи речи в цифровой форме используется частота оцифровывания в 8 кГц и 8 бит для представления амплитуды сигнала, что дает скорость элементарного голосового канала в 64 кбит/с.

Цифровое кодирование

При цифровом кодировании дискретной информации применяют потенциальные и импульсные коды. В потенциальных кодах для представления логических единиц и нулей используется только значение потенциала сигнала, а его перепады, формирующие законченные импульсы, во внимание не принимаются. Импульсные коды позволяют представить двоичные данные либо импульсами определенной полярности, либо частью импульса — перепадом потенциала определенного направления.

Требования к методам цифрового кодирования

При использовании прямоугольных импульсов для передачи дискретной информации необходимо выбрать такой способ кодирования, который одновременно достигал бы нескольких целей:

- имел при одной и той же битовой скорости наименьшую ширину спектра результирующего сигнала;
- обеспечивал синхронизацию между передатчиком и приемником;
- обладал способностью распознавать ошибки;
- обладал низкой стоимостью реализации.

Более узкий спектр сигналов позволяет на одной и той же линии (с одной и той же полосой пропускания) добиваться более высокой скорости передачи данных. Кроме того, часто к спектру сигнала предъявляется требование отсутствия постоянной составляющей, то есть отсутствия постоянного тока между передатчиком и приемником. В частности, применение различных трансформаторных схем гальванической развязки препятствует прохождению постоянного тока.

Синхронизация передатчика и приемника нужна для того, чтобы приемник точно знал, в какой момент времени необходимо считывать новую информацию с линии связи. Эта проблема в сетях решается сложнее, чем при обмене данными между близко расположенными устройствами, например между блоками внутри компьютера или же между компьютером и принтером. На небольших расстояниях хорошо работает схема, основанная на отдельной тактирующей линии связи (рис. 5.5), так что информация снимается с линии данных только в момент прихода тактового импульса. В сетях использование этой схемы вызывает трудности из-за неоднородности характеристик проводников в кабелях. На больших расстояниях неравномерность скорости распространения сигнала может привести к тому, что тактовый импульс придет несколько позже или раньше соответствующего сигнала данных, и бит данных будет пропущен или считан повторно. Другой причиной, по которой в сетях отказываются от использования тактирующих импульсов, является экономия проводников в дорогостоящих кабелях.



Рис. 5.5. Синхронизация приемника и передатчика на небольших расстояниях

Поэтому в сетях применяются так называемые самосинхронизирующиеся коды, сигналы которых несут для передатчика указания о том, в какой момент времени нужно осуществлять распознавание очередного бита (или нескольких битов, если код ориентирован более чем на два состояния сигнала). Любой резкий перепад сигнала — так называемый фронт — может служить хорошим указанием для синхронизации приемника с передатчиком.

При использовании синусоид в качестве несущего сигнала результирующий код обладает свойством самосинхронизации, так как изменение амплитуды несущей частоты дает возможность приемнику определить момент появления входного кода.

Распознавание и коррекцию искаженных данных сложно осуществить средствами физического уровня, поэтому чаще всего эту работу берут на себя протоколы, лежащие выше: канальный, сетевой, транспортный или прикладной. С другой стороны, распознавание ошибок на физическом уровне экономит время, так как приемник не ждет полного помещения кадра в буфер, а отбраковывает его сразу при распознавании ошибочных битов внутри кадра.

Требования, предъявляемые к методам кодирования, являются взаимно противоречивыми, поэтому каждый из рассматриваемых ниже популярных методов цифрового кодирования обладает своими преимуществами и своими недостатками по сравнению с другими.

Потенциальный код без возвращения к нулю

Рисунок 5.6, а иллюстрирует уже упомянутый ранее метод потенциального кодирования, называемый также кодированием без возвращения к нулю (Non Return to Zero, NRZ). Последнее название отражает то обстоятельство, что при передаче последовательности единиц сигнал не возвращается к нулю в течение такта (как мы увидим ниже, в других методах кодирования возврат к нулю в этом случае происходит). Метод NRZ прост в реализации, обладает хорошей распознаваемостью ошибок (из-за двух резко отличающихся потенциалов), но не обладает свойством самосинхронизации. При передаче длинной последовательности единиц или нулей сигнал на линии не изменяется, поэтому приемник лишен возможности определять по входному сигналу моменты времени, когда нужно в очередной раз считывать данные. Даже при наличии высокоточного тактового генератора приемник может ошибиться с моментом съема данных, так как частоты двух генераторов никогда не бывают полностью идентичными. Поэтому при высоких скоростях обмена данными и длинных последовательностях единиц или нулей небольшое рассогласование тактовых частот может привести к ошибке в целый такт и, соответственно, считыванию некорректного значения бита.

Другим серьезным недостатком метода NRZ является наличие низкочастотной составляющей, которая приближается к нулю при передаче длинных последовательностей единиц или нулей. Из-за этого многие каналы связи, не обеспечивающие прямого гальванического соединения между приемником и источником, этот вид кодирования не поддерживают. В результате в чистом виде код

NRZ в сетях не используется. Тем не менее используются его различные модификации, в которых устраняют как плохую самосинхронизацию кода NRZ, так и проблемы постоянной составляющей. Привлекательность кода NRZ, из-за которой имеет смысл заняться его улучшением, состоит в достаточно низкой частоте основной гармоники f_0 , которая равна $N/2$ Гц, как это было показано в предыдущем разделе. У других методов кодирования, например манчестерского, основная гармоника имеет более высокую частоту.

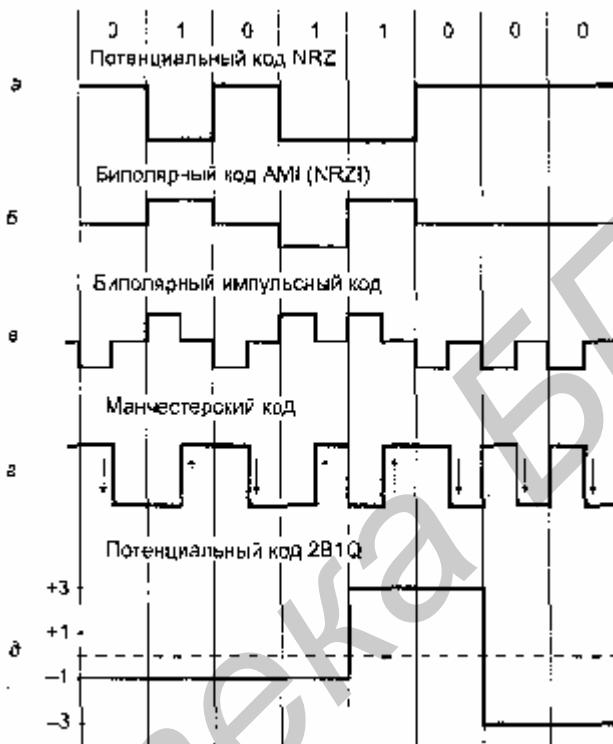


Рис. 5.6. Способы дискретного кодирования данных

Метод биполярного кодирования с альтернативной инверсией

Одной из модификаций метода NRZ является метод биполярного кодирования с альтернативной инверсией (Bipolar Alternate Mark Inversion, AMI). В этом методе (рис. 5.6, б) используются три уровня потенциала — отрицательный, нулевой и положительный. Для кодирования логического нуля используется нулевой потенциал, а логическая единица кодируется либо положительным потенциалом, либо отрицательным, при этом потенциал каждой новой единицы противоположен потенциалу предыдущей.

Код AMI частично ликвидирует проблемы постоянной составляющей и отсутствия самосинхронизации, присущие коду NRZ. Это происходит при передаче длинных последовательностей единиц. В этих случаях сигнал на линии представляет собой последовательность разнополярных импульсов с тем же спектром, что и у кода NRZ, передающего чередующиеся нули и единицы, то есть без постоянной составляющей и с основной гармоникой $N/2$ Гц (где N — битовая скорость передачи данных). Длинные же последовательности нулей также опасны для кода AMI, как и для кода NRZ — сигнал вырождается в постоянный потенциал нулевой амплитуды. Поэтому код AMI требует дальнейшего улуч-

шения, хотя задача упрощается — осталось справиться только с последовательностями нулей.

В целом, для различных комбинаций битов на линии использование кода AMI приводит к более узкому спектру сигнала, чем для кода NRZ, а значит, и к более высокой пропускной способности линии. Например, при передаче чередующихся единиц и нулей основная гармоника f_0 имеет частоту $N/4$ Гц. Код AMI предоставляет также некоторые возможности по распознаванию ошибочных сигналов. Так, нарушение строгого чередования полярности сигналов говорит о ложном импульсе или исчезновении с линии корректного импульса. Сигнал с некорректной полярностью называется запрещенным сигналом (signal violation).

В коде AMI используются не два, а три уровня сигнала на линии. Дополнительный уровень требует увеличение мощности передатчика примерно на 3 дБ для обеспечения той же достоверности приема битов на линии, что является общим недостатком кодов с несколькими состояниями сигнала по сравнению с кодами, которые различают только два состояния.

Потенциальный код с инверсией при единице

Существует код, похожий на AMI, но только с двумя уровнями сигнала. При передаче нуля он передает потенциал, который был установлен в предыдущем такте (то есть не меняет его), а при передаче единицы потенциал инвертируется на противоположный. Этот код называется потенциальным кодом с инверсией при единице (Non Return to Zero with ones Inverted, NRZI). Он удобен в тех случаях, когда наличие третьего уровня сигнала весьма нежелательно, например в оптических кабелях, где устойчиво распознаются два состояния сигнала — свет и темнота.

Для улучшения потенциальных кодов, подобных AMI и NRZI, используются два метода. Первый метод основан на добавлении и исходный код избыточных битов, содержащих логические единицы. Очевидно, что в этом случае длинные последовательности нулей прерываются и код становится самосинхронизирующимся для любых передаваемых данных. Исчезает также постоянная составляющая, а значит, еще более сужается спектр сигнала. Но этот метод снижает полезную пропускную способность линии, так как избыточные единицы пользовательской информации не несут. Другой метод основан на предварительном «перемешивании» исходной информации таким образом, чтобы вероятность появления единиц и нулей на линии становилась близкой. Устройства, или блоки, выполняющие такую операцию, называются скремблерами (scramble — свалка, беспорядочная куча). При скремблировании используется известный алгоритм, поэтому приемник, получив двоичные данные, передает их на дескремблер, который восстанавливает исходную последовательность битов. Избыточные биты при этом по линии не передаются. Оба метода относятся к логическому, а не физическому кодированию, так как форму сигналов на линии они не определяют. Более детально они изучаются в следующем разделе.

Биполярный импульсный код

Кроме потенциальных кодов в сетях используются и импульсные коды, в которых данные представлены полным импульсом или же его частью — фронтом. Наиболее простым случаем такого подхода является биполярный импульсный код, в котором единица представлена импульсом одной полярности, а ноль — другой (рис. 5.6, в). Каждый импульс длится половину такта. Такой код обладает отличными самосинхронизирующими свойствами, но постоянная составляющая может присутствовать, например, при передаче длинной последовательности единиц или нулей. Кроме того, спектр у него ниже, чем у потенциальных кодов. Так, при передаче всех нулей или единиц частота основной гармоники кода будет равна N Гц, что в два раза выше основной гармоники кода NRZ и в четыре раза выше основной гармоники кода AMI при передаче чередующихся единиц и нулей. Из-за слишком широкого спектра биполярный импульсный код используется редко.

Манчестерский код

В локальных сетях до недавнего времени самым распространенным методом кодирования был так называемый манчестерский код (рис. 5.6, в). Он применяется в технологиях Ethernet и Token Ring.

В манчестерском коде для кодирования единиц и нулей используется перепад потенциала, то есть фронт импульса. При манчестерском кодировании каждый такт делится на две части. Информация кодируется перепадами потенциала, происходящими в середине каждого такта. Единица кодируется перепадом от низкого уровня сигнала к высокому, а ноль — обратным перепадом. В начале каждого такта может происходить служебный перепад сигнала, если нужно представить несколько единиц или нулей подряд. Так как сигнал изменяется по крайней мере один раз за такт передачи одного бита данных, то манчестерский код обладает хорошими самосинхронизирующими свойствами. Полоса пропускания манчестерского кода уже, чем у биполярного импульсного. У него также нет постоянной составляющей, а основная гармоника в худшем случае (при передаче последовательности единиц или нулей) имеет частоту N Гц, а в лучшем (при передаче чередующихся единиц и нулей) она равна $N/2$ Гц, как и у кодов AMI или NRZ. В среднем ширина полосы манчестерского кода в полтора раза уже, чем у биполярного импульсного кода, а основная гармоника колеблется вблизи значения $3N/4$. Манчестерский код имеет еще одно преимущество перед биполярным импульсным кодом. В последнем для передачи данных используются три уровня сигнала, а в манчестерском — два.

Потенциальный код 2B1Q

На рис. 5.6, З показан потенциальный код с четырьмя уровнями сигнала для ко-

дирования данных. Это код 2B1Q, название которого отражает его суть — каждые два бита (2B) передаются за один такт сигналом, имеющим четыре состояния (1Q). Паре бит 00 соответствует потенциал -2,5 В; паре бит 01 — потенциал -0,833 В; паре 11 — потенциал +0,833 В; а паре 10 — потенциал +2,5 В. При этом способе кодирования требуются дополнительные меры по борьбе с длинными последовательностями одинаковых пар битов, так как при этом сигнал превращается в постоянную составляющую. При случайном чередовании битов спектр сигнала в два раза уже, чем у кода NRZ, так как при той же битовой скорости длительность такта увеличивается в два раза. Таким образом, с помощью кода 2B1Q можно по одной и той же линии передавать данные в два раза быстрее, чем с помощью кода AMI или NRZI. Однако для его реализации мощность передатчика должна быть выше, чтобы четыре уровня четко различались приемником на фоне помех.

Логическое кодирование

Логическое кодирование используется для улучшения потенциальных кодов типа AMI, NRZI или 2Q1B. Логическое кодирование должно заменять длинные последовательности битов, приводящие к постоянному потенциалу, вкраплениями единиц. Как уже отмечалось выше, для логического кодирования характерны два метода — избыточные коды и скремблирование.

Избыточные коды

Избыточные коды основаны на разбиении исходной последовательности битов на порции, которые часто называют символами. Затем каждый исходный символ заменяется новым, который имеет большее количество битов, чем исходный. Например, логический код 4B/5B, используемый в технологиях FDDI и Fast Ethernet, заменяет исходные символы длиной в 4 бит символами длиной в 5 бит. Так как результирующие символы содержат избыточные биты, то общее количество битовых комбинаций в них больше, чем в исходных. Так, в коде 4B/5B результирующие символы могут содержать 32 битовых комбинации, в то время как исходные символы — только 16 (табл. 5.1). Поэтому в результирующем коде можно отобрать 16 таких комбинаций, которые не содержат большого количества нулей, а остальные считать запрещенными кодами (*code violation*). Кроме устранения постоянной составляющей и придания коду свойства самосинхронизации, избыточные коды позволяют приемнику распознавать искаженные биты. Если приемник принимает запрещенный код, значит, на линии произошло искажение сигнала.

После разбиения получившийся код 4B/5B передается по линии путем физического кодирования по одному из методов потенциального кодирования, чувствительному только к длинным последовательностям нулей. Символы кода 4B/5B длиной 5 бит гарантируют, что при любом их сочетании на линии не может встретиться более трех нулей подряд.

Таблица 5.1. Соответствие исходных и результирующих кодов 4В/5В

Исходный код	Результирующий код	Исходный код	Результирующий код
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Буква В в названии кода означает, что элементарный сигнал имеет два состояния (от английского *binary* — двоичный). Имеются также коды и с тремя состояниями сигнала, например, в коде 8Б/6Т для кодирования 8 бит исходной информации используется код из 6 сигналов, каждый из которых имеет три состояния. Избыточность кода 8Б/6Т выше, чем кода 4В/5В, так как на 256 исходных кодов приходится $3^6 = 729$ результирующих символов, Использование таблицы перекодировки является очень простой операцией, поэтому этот подход не усложняет сетевые адаптеры и интерфейсные блоки коммутаторов и маршрутизаторов.

Для обеспечения заданной пропускной способности линии передатчик, использующий избыточный код, должен работать с повышенной тактовой частотой. Так, для передачи кодов 4В/5В со скоростью 100 Мбит/с передатчик должен работать с тактовой частотой 125 МГц. При этом спектр сигнала на линии расширяется по сравнению со случаем, когда по линии передается чистый, не избыточный код. Тем не менее спектр избыточного потенциального кода оказывается уже спектра манчестерского кода, что оправдывает дополнительный этап логического кодирования, а также работу приемника и передатчика на повышенной тактовой частоте.

Скрэмблирование

Скрэмблирование наряду с избыточным кодированием является способом логического кодирования информации. Методы скрэмблирования заключаются в побитном вычислении результирующего кода на основании битов исходного кода и полученных в предыдущих тактах битов результирующего кода. Например, скрэмблер может реализовывать следующее соотношение:

$$B_i = A_i B_{i-3} B_{i-5}$$

Здесь B_i — двоичная цифра результирующего кода, полученная на i -м такте работы скрэмблера, A_i — двоичная цифра исходного кода, поступающая на i -м такте на вход скрэмблера, B_{i-3} и B_{i-5} — двоичные цифры результирующего кода, полученные на предыдущих тактах работы скрэмблера, соответственно на 3 и на 5 тактов ранее текущего такта, и объединенные операцией исключающего ИЛИ (сложение по модулю 2).

Например, для исходной последовательности 110110000001 скрэмблер даст следующий результирующий код (первые три цифры результирующего кода будут совпадать с исходным кодом, так как еще нет нужных предыдущих цифр):

$$\begin{aligned}B_1 &= A_1 = 1 \\B_2 &= A_2 = 1 \\B_3 &= A_3 = 0 \\B_4 &= A_4 \quad B_1 = 1 \ 1 = 0 \\B_5 &= A_5 \quad B_2 = 1 \ 1 = 0 \\B_6 &= A_6 \quad B_3 \quad B_1 = 0 \ 0 \ 1 = 1 \\B_7 &= A_7 \quad B_4 \quad B_2 = 0 \ 0 \ 1 = 1 \\B_8 &= A_8 \quad B_5 \quad B_3 = 0 \ 0 \ 0 = 0 \\B_9 &= A_9 \quad B_6 \quad B_4 = 0 \ 1 \ 0 = 1 \\B_{10} &= A_{10} \quad B_7 \quad B_5 = 0 \ 1 \ 0 = 1 \\B_{11} &= A_{11} B_8 B_6 = 001 = 1 \\B_{12} &= A_{12} B_9 B_7 = 111 = 1\end{aligned}$$

Таким образом, на выходе скрэмблера появится последовательность 110001101111, в которой нет последовательности из шести нулей, присутствовавшей в исходном коде.

После получения результирующей последовательности приемник передает ее дескрэмблеру, который восстанавливает исходную последовательность на основании обратного соотношения:

$$C_i = B_{i-3} B_{i-5} = (A_i \ B_{i-3} \ B_{i-5}) \ B_{i-3} B_{i-5} = A_i.$$

Различные алгоритмы скрэмблирования отличаются количеством слагаемых, дающих цифру результирующего кода, и сдвигом между слагаемыми. Так, в сетях ISDN при передаче данных от сети к абоненту используется преобразование со сдвигами в 5 и 23 позиции, а при передаче данных от абонента в сеть — со сдвигами 18 и 23 позиции.

Существуют и более простые методы борьбы с последовательностями единиц, также относимые к классу скрэмблирования.

Для улучшения кода Bipolar AMI используются два метода, основанные на искусственном искажении последовательности нулей запрещенными символами.

Рисунок 5.7 иллюстрирует использование метода B8ZS (Bipolar with 8-Zeros Substitution) и метода HDB3 (High-Density Bipolar 3-Zcros) для корректировки кода АМИ. Исходный код состоит из двух длинных последовательностей нулей: в первом случае — из 8, а во втором — из 5.

Код B8ZS исправляет только последовательности, состоящие из 8 нулей. Для этого он после первых трех нулей вместо оставшихся пяти нулей вставляет пять цифр: V-1*-0-V-1*. Здесь V обозначает сигнал единицы, запрещенной для данного такта полярности, то есть сигнал, не изменяющий полярность предыдущей единицы, 1* — сигнал единицы корректной полярности (знак звездочки отмечает тот факт, что в исходном коде в этом такте была не единица, а ноль). В результате на 8 тактах приемник наблюдает два искажения — очень маловероятно, что это случилось из-за шума на линии или других сбоев передачи. Поэтому приемник считает такие нарушения кодировкой 8 последовательных нулей и

после приема заменяет их исходными 8 нулями. Код B8ZS построен так, что его постоянная составляющая равна нулю при любых последовательностях двоичных цифр.

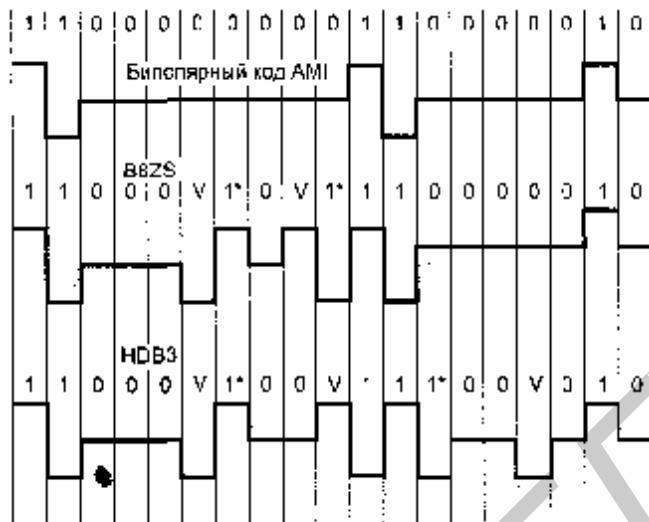


Рис. 5.7. Коды B8ZS и HDB3

Код HDB3 исправляет любые четыре подряд идущих нуля в исходной последовательности. Правила формирования кода HDB3 более сложные, чем кода B8ZS. Каждые четыре нуля заменяются четырьмя сигналами, в которых имеется один сигнал V. Для подавления постоянной составляющей полярность сигнала V чередуется при последовательных заменах. Кроме того, для замены используются два образца четырехтактовых кодов. Если перед заменой исходный код содержал нечетное число единиц, задействуется последовательность OOOV, а если число единиц было четным — последовательность 1*OOV.

Улучшенные потенциальные коды обладают достаточно узкой полосой пропускания для любых последовательностей единиц и нулей, которые встречаются в передаваемых данных. На рис. 5.8 приведены спектры сигналов разных кодов, полученные при передаче произвольных данных, в которых различные сочетания нулей и единиц в исходном коде равновероятны. При построении графиков спектр усреднялся по всем возможным наборам исходных последовательностей. Естественно, что результирующие коды могут иметь и другое распределение нулей и единиц. Из рисунка видно, что потенциальный код NRZ обладает хорошим спектром с одним недостатком — у него имеется постоянная составляющая. Коды, полученные из потенциального путем логического кодирования, обладают более узким спектром, чем манчестерский, даже при повышенной тактовой частоте (на рисунке спектр кода 4B/5B должен был бы примерно совпадать с кодом B8ZS, но он сдвинут в область более высоких частот, так как его тактовая частота повышена на 1/4 по сравнению с другими кодами). Этим объясняется применение потенциальных избыточных и скремблированных кодов в современных технологиях, подобных FDDI, Fast Ethernet, Gigabit Ethernet, ISDN и т. п. вместо манчестерского и биполярного импульсного кодирования.



Рис. 6.8. Спектры потенциальных и импульсных кодов

Выводы

- На широкополосных каналах связи применяются потенциальные и импульсные методы кодирования, в которых данные представлены различными уровнями постоянного потенциала сигнала либо полярностями импульса или его фронта.
- При использовании потенциальных кодов особое значение приобретает задача синхронизации приемника с передатчиком, так как при передаче длинных последовательностей нулей или единиц сигнал на входе приемника не изменяется и приемнику сложно определить момент съема очередного бита данных.
- Наиболее простым потенциальным кодом является код без возвращения к нулю (NRZ), однако он не является самосинхронизирующимся и создает постоянную составляющую.
- Наиболее популярным импульсным кодом является манчестерский код, в котором информацию несет направление перепада сигнала в середине каждого такта. Манчестерский код применяется в технологиях Ethernet и Token Ring.
- Для улучшения свойств потенциального кода NRZ используются методы логического кодирования, исключающие длинные последовательности нулей. Эти методы основаны:
 - на введении избыточных битов в исходные данные (коды типа 4B/5B);
 - скремблировании исходных данных (коды типа 2B1Q).
- Улучшенные потенциальные коды обладают более узким спектром, чем импульсные, поэтому они находят применение в высокоскоростных технологиях, таких как FDDI, Fast Ethernet, Gigabit Ethernet.

Особенности протоколов канального уровня

Канальный уровень обеспечивает передачу пакетов данных, поступающих от протоколов верхних уровней, узлу назначения, адрес которого также указывает протокол верхнего уровня. Протоколы канального уровня оформляют переданные им пакеты в кадры собственного формата, помещая указанный адрес назначения в одно из полей такого кадра, а также сопровождая кадр контрольной суммой. Протокол канального уровня обычно работает в пределах одной сети, которая, как правило, входит в более крупную составную сеть, объединяемую протоколами сетевого уровня. Адреса, с которыми работает протокол канального уровня, используются для доставки кадров только в пределах этой сети, а для перемещения пакетов из сети в сеть применяются уже адреса следующего,

сетевого уровня. Поэтому таблицы продвижения, на основе которых работают устройства канального уровня, содержат только адреса узлов своей сети, что существенно сокращает количество записей в них, а следовательно, повышает скорости их просмотра и продвижения кадра.

Типичными представителями протоколов канального уровня являются протоколы семейства Ethernet, которые применяются в основном в локальных сетях, а в последнее время все чаще стали использоваться в сетях масштаба города (мегаполиса). В глобальных сетях протоколы канального уровня применяются в пределах «вырожденной» сети — канала «точка-точка», а также в сетях с произвольной топологией, но уже после того, как в них проложен виртуальный путь. Примером протокола канального уровня первого типа является протокол PPP (Point-to-Point Protocol), а второго — протоколы канального уровня сетей frame relay и ATM. В коммутаторах глобальных сетей, поддерживающих эти протоколы, таблицы продвижения также содержат адреса, имеющие локальное (то есть только для данного коммутатора) значение, поэтому они, как и таблицы продвижения коммутаторов локальных сетей, имеют небольшие размеры и просматриваются относительно быстро — по сравнению с таблицами маршрутизации, включающими адреса всех сетей составной сети и иногда состоящими из десятков тысяч записей (как это и происходит с таблицами маршрутизации магистральных маршрутизаторов Интернета).

Далее будут рассмотрены общие особенности протоколов канального уровня, характерные для протоколов как локальных, так и глобальных сетей. Каждый протокол канального уровня можно охарактеризовать следующим набором свойств:

- асинхронный/синхронный режим передачи байтов;
- символьно-ориентированный/бит-ориентированный;
- с предварительным установлением соединения/дейтаграммный;
- с обнаружением искаженных данных/без обнаружения;
- с обнаружением потерянных данных/без обнаружения;
- с восстановлением искаженных и потерянных данных/без восстановления;
- с поддержкой динамической компрессии данных/без поддержки.

Некоторые из этих свойств присущи не только протоколам канального уровня, но и протоколам более высоких уровней.

Асинхронные протоколы

Асинхронные протоколы предоставляют наиболее старый способ связи. Эти протоколы оперируют не с кадрами, а с отдельными символами, которые представлены байтами со старт-стоповыми символами. Асинхронные протоколы ведут свое происхождение от тех времен, когда два человека связывались с помощью телетайпов по каналу «точка-точка». С развитием техники коммуникаций асинхронные протоколы стали применяться для связи телетайпов, разного рода клавиатур и дисплеев с вычислительными машинами. Единицей передаваемых данных в асинхронных протоколах является не кадр данных, а отдельный символ, который чаще всего представлен одним байтом. Некоторые сим-

волы имеют управляющий характер, например символ <CR> предписывает телетайпу или дисплею выполнить возврат каретки на начало строки. В этих протоколах существуют управляющие последовательности, обычно начинающиеся с символа <ESC>. Эти последовательности вызывают на управляемом устройстве достаточно сложные действия — например, загрузку нового шрифта на принтер.

В асинхронных протоколах применяются стандартные наборы символов, чаще всего ASCII или EBCDIC. Так как первые 32 или 27 кодов в этих наборах являются специальными кодами, которые не отображаются на дисплее или принтере, то они использовались асинхронными протоколами для управления режимом обмена данными. В самих пользовательских данных, которые представляли собой буквы, цифры, а также такие знаки, как @, %, \$ и т. п., специальные символы никогда не встречались, так что проблемы их отделения от пользовательских данных не существовало.

Описанный режим работы называется асинхронным, или старт-стопным. Дополнительной причиной его использования является наличие устройств, которые генерируют байты данных в случайные моменты времени. Так работает клавиатура дисплея или другого терминального устройства, с которого человек вводит данные для обработки их компьютером.

В асинхронном режиме каждый байт данных сопровождается специальными сигналами «старт» и «стоп» (рис. 5.9, а). Назначение этих сигналов состоит в том, чтобы, во-первых, известить приемник о приходе данных и, во-вторых, чтобы дать приемнику достаточно времени для выполнения некоторых функций, связанных с синхронизацией, до поступления следующего байта. Сигнал «старт» имеет продолжительность в один тактовый интервал, а сигнал «стоп» может длиться один, полтора или два такта, поэтому говорят, что используется один, полтора или два бита в качестве стопового сигнала, хотя пользовательские биты эти сигналы не представляют.

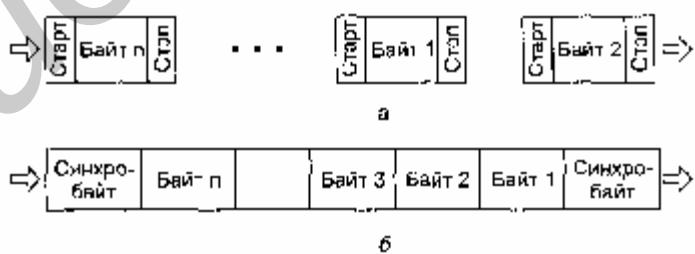


Рис. 5.9. Асинхронная (а) и синхронная (б) передачи из уравнение байтов

Асинхронным описанный режим называется потому, что каждый байт может быть несколько смещен во времени относительно побитовых тактов предыдущего байта. Такая асинхронность передачи байтов не влияет на корректность принимаемых данных, так как в начале каждого байта происходит дополнительная синхронизация приемника с источником за счет битов «старт». Более «свободные» временные допуски определяют низкую стоимость оборудования асинхронной системы.

Постепенно асинхронные протоколы усложнялись и стали наряду с отдельными символами использовать целые блоки данных, то есть кадры. Например, популярный протокол XMODEM передает файлы между двумя компьютерами по асинхронному модему. Начало приема очередного блока файла инициируется символьной командой — принимающая сторона постоянно передает символ ASCII NAK. Передающая сторона, приняв NAK, отправляет очередной блок файла, состоящий из 128 байт данных, заголовка и концевика. Заголовок состоит из специального символа SOH (Start Of Header) и номера блока. Концевик содержит контрольную сумму блока данных. Приемная сторона, получив новый блок, проверяет его номер и контрольную сумму. В случае совпадения этих параметров с ожидаемыми приемник отправляет символ ACK, а в противном случае — символ NAK, после чего передатчик должен повторить передачу данного блока. В конце передачи файла передается символ EOF.

Как видно из описания протокола XMODEM, часть управляющих операций выполнялась в асинхронных протоколах посылкой в асинхронном режиме отдельных символов, в то же время часть данных пересыпалась блоками, что более характерно для синхронных протоколов.

При синхронном режиме передачи старт-стопные биты между каждой парой байтов отсутствуют. Пользовательские данные собираются в кадр, который предваряется байтом синхронизации (рис. 5.9, б). Байт синхронизации (синхробайт) — это байт, содержащий заранее известный код, например 0111110, который оповещает приемник о приходе кадра данных. При его получении приемник должен войти в режим байтовой синхронизации с передатчиком, то есть правильно понимать начало очередного байта кадра. Иногда применяется несколько синхробайтов для обеспечения более надежной синхронизации приемника и передатчика. Так как при передаче длинного кадра у приемника могут появиться проблемы с синхронизацией битов, то в этом случае используются самосинхронизирующиеся коды.

Синхронные символьно-ориентированные и бит-ориентированные протоколы

В синхронных протоколах между пересылаемыми символами (байтами) нет стартовых и стоповых сигналов, поэтому отдельные символы в этих протоколах пересыпать нельзя. Все обмены данными осуществляются кадрами, которые имеют в общем случае заголовок, поле данных и концевик (рис. 5.10). Все биты кадра передаются непрерывным синхронным потоком, что значительно ускоряет передачу данных.

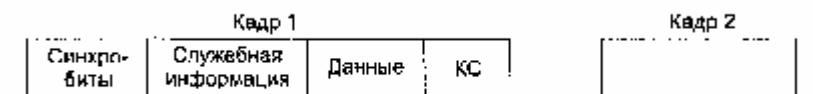


Рис. 5.10. Кадры синхронных протоколов

Так как байты в этих протоколах не отделяются друг от друга служебными сигналами, то одной из первых задач приемника является распознавание границы байтов. Затем приемник должен найти начало и конец кадра, а также определить

лить границы каждого поля кадра — адреса назначения, адреса источника, других служебных полей заголовка, поля данных и контрольной суммы, если она имеется.

Большинство протоколов допускает использование в кадре поля данных переменной длины. Иногда и заголовок может иметь переменную длину. Обычно протоколы определяют максимальное значение, которое может иметь длина поля данных. Эта величина называется максимальной единицей передачи данных (Maximum Transfer Unit, MTV). В некоторых протоколах задается также минимальное значение, которое может иметь длина поля данных. Например, протокол Ethernet требует, чтобы поле данных содержало по крайней мере 46 байт данных (если приложение хочет отправить меньшее количество байтов, то оно обязано дополнить их до 46 байт любыми значениями). Другие протоколы разрешают использовать поле данных нулевой длины, например PPP или FDDI. Существуют также протоколы с кадрами фиксированной длины, например, в протоколе ATM кадры имеют фиксированный размер 53 байт, включая служебную информацию. Для таких протоколов необходимо решить только первую часть задачи — распознать начало кадра (кадр протокола ATM имеет специальное название — ячейка, или cell).

Синхронные протоколы канального уровня бывают двух типов: символьно-ориентированные (байт-ориентированные) и бит-ориентированные. Для обоих характерны одни и те же методы синхронизации битов. Главное различие между ними заключается в методе синхронизации символов и кадров.

Символьно-ориентированные протоколы

Символьно-ориентированные протоколы используются в основном для передачи блоков отображаемых символов, например текстовых файлов. Так как при синхронной передаче нет стоповых и стартовых битов, для синхронизации символов необходим другой метод. Синхронизация достигается за счет того, что передатчик добавляет два или более управляющих символа, называемых символами SYN, перед каждым блоком символов. В коде ASCII символ SYN имеет двоичное значение 0010110, это несимметричное относительно начала символа значение позволяет легко разграничивать отдельные символы SYN при их последовательном приеме. Символы SYN выполняют две функции: во-первых, они обеспечивают приемнику битовую синхронизацию, во-вторых, как только битовая синхронизация достигается, они позволяют приемнику начать распознавание границ символов SYN. После того как приемник начал отделять один символ от другого, можно задавать границы начала кадра с помощью другого специального символа. Обычно в символьных протоколах для этих целей используется символ STX (Start of Text, ASCII 0000010). Другой символ отмечает окончание кадра — ETX (End of Text, ASCII 0000011).

Однако такой простой способ выделения начала и конца кадра хорошо работал только в том случае, если внутри кадра не было символов STX и ETX. При подключении к компьютеру алфавитно-цифровых терминалов такая задача действительно не возникала. Тем не менее синхронные символьно-ориентированные

протоколы позднее стали использоваться и для связи компьютера с компьютером, а в этом случае данные внутри кадра могут быть любые, если, например, между компьютерами передается программа. Наиболее популярным протоколом такого типа был протокол BSC компании IBM. Он работал в двух режимах — непрозрачном, в котором некоторые специальные символы внутри кадра запрещались, и прозрачном, в котором разрешались передачи внутри кадра любых символов, в том числе и ETX. Прозрачность достигалась за счет того, что перед управляющими символами STX и ETX всегда вставлялся символ DLE (Data Link Escape). Такая процедура называется стаффингом символов (от английского stuff — всякая всячина, заполнитель). А если в поле данных кадра встречалась последовательность DLE ETX, то передатчик удваивал символ DLE, то есть порождал последовательность DLE DLE ETX. Приемник, встретив подряд два символа DLE, всегда удалял первый, но оставшийся символ DLE уже не рассматривал как начало управляющей последовательности, то есть оставшиеся символы DLE ETX считал просто пользовательскими данными.

Бит-ориентированные протоколы

Потребность в паре символов в начале и конце каждого кадра вместе с дополнительными символами DLE означает, что символьно-ориентированная передача не эффективна для передачи двоичных данных, так как приходится в поле данных кадра добавлять достаточно много избыточных данных. Кроме того, формат управляющих символов для разных кодировок различен, например, в коде ASCII символ SYN равен 0010110, а в коде EBCDIC - 00110010. Так что этот метод допустим только с определенным типом кодировки, даже если кадр содержит чисто двоичные данные. Чтобы преодолеть эти проблемы, сегодня почти всегда используется более универсальный метод, называемый бит-ориентированной передачей. Этот метод сейчас применяется при передаче как двоичных, так и символьных данных. На рис. 5.11 показаны три различные схемы бит-ориентированной передачи. Они отличаются способом обозначения начала и конца каждого кадра.

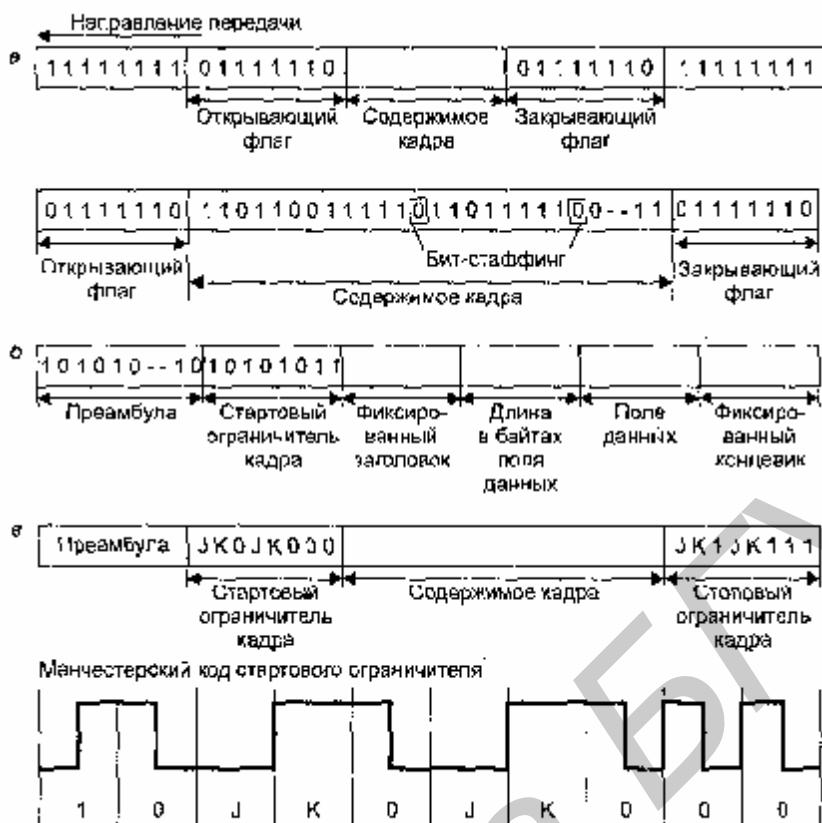


Рис. 5.11. Способы выделения начала и конца кадра при синхронной передаче

Первая схема, показанная на рис. 5.11, а, похожа на схему с символами STX и ETX в символьно-ориентированных протоколах. Начало и конец каждого кадра отмечается одной и той же 8-битовой последовательностью — 01111110, называемой флагом. Термин «бит-ориентированный» используется потому, что принимаемый поток битов сканируется приемником на побитовой основе для обнаружения стартового флага, а затем во время приема — для обнаружения стопового флага. Поэтому длина кадра в этом случае не обязательно должна быть кратна 8 бит.

Чтобы обеспечить синхронизацию приемника, передатчик посылает последовательность байтов простого (каждый состоит из 11111111), предшествующую стартовому флагу.

Для достижения прозрачности данных в этой схеме необходимо, чтобы флаг не присутствовал в поле данных кадра. Это достигается с помощью приема, известного как вставка 0 бита, — бит-стаффинг. Схема вставки бита работает только во время передачи поля данных кадра. Если эта схема обнаруживает, что подряд передано пять единиц, то она автоматически вставляет дополнительный ноль (даже если после этих пяти единиц и так идет ноль). Поэтому последовательность 01111110 никогда не появится в поле данных кадра. Аналогичная схема работает и приемнике и выполняет обратную функцию. Когда после пяти единиц обнаруживается ноль, он автоматически удаляется из поля данных кадра. Бит-стаффинг гораздо более экономичен, чем байт-стаффинг, так как вместо лишнего байта вставляется один бит, следовательно, скорость передачи пользовательских данных в этом случае замедляется в меньшей степени.

Во второй схеме (см. рис. 5.11, б) для обозначения начала кадра имеется только стартовый флаг, а для определения конца кадра используется поле длины кадра, которое при фиксированных размерах заголовка и концевика чаще всего имеет смысл длины поля данных кадра. Эта схема наиболее применима в локальных сетях. В этих сетях для обозначения факта незанятости среды в исходном состоянии вообще не передается никаких символов. Чтобы все остальные станции вошли в битовую синхронизацию, посылающая станция предваряет содержимое кадра последовательностью битов, известной как преамбула, которая состоит из чередования единиц и нулей 101010... Войдя в битовую синхронизацию, приемник исследует входной поток на побитовой основе, пока не обнаружит байт начала кадра 10101011, который выполняет роль символа STX. За этим байтом следует заголовок кадра, в котором в определенном месте находится поле длины поля данных. Таким образом, в этой схеме приемник просто отсчитывает заданное количество битов, чтобы определить окончание кадра.

Третья схема (см. рис. 5.11, в) использует для обозначения начала и конца кадра флаги, которые включают запрещенные для данного кода сигналы (code violations, V). Например, при манчестерском кодировании вместо обязательного изменения полярности сигнала в середине тактового интервала уровень сигнала остается неизменным и низким (запрещенный сигнал J) или неизменным и высоким (запрещенный сигнал K). Начало кадра отмечается последовательностью JK0JK000, а конец — последовательностью JK1JK100. Этот способ очень экономичен, так как не требует ни бит-стаффинга, ни поля длины, но его недостаток заключается в зависимости от принятого метода физического кодирования. При использовании избыточных кодов роль сигналов J и K играют запрещенные символы, например, в коде 4B/5B этими символами являются коды 11000 и 10001.

Каждая из трех схем имеет свои преимущества и недостатки. Флаги позволяют отказаться от специального дополнительного поля, но требуют специальных мер: либо по разрешению размещения флага в поле данных за счет бит-стаффинга, либо по использованию в качестве флага запрещенных сигналов, что делает эту схему зависимой от способа кодирования.

Протоколы с гибким форматом кадра

Для большей части протоколов канального уровня характерны кадры, состоящие из служебных полей фиксированной длины. Исключение делается только для поля данных с целью экономной пересылки как небольших квитанций, так и больших файлов. Способ определения окончания кадра путем задания длины поля данных, рассмотренный выше, как раз рассчитан на такие кадры с фиксированной структурой и фиксированными размерами служебных полей.

Однако существует ряд протоколов, в которых кадры имеют гибкую структуру. Например, к таким протоколам относится очень популярный протокол канального уровня PPP, используемый для соединений типа «точка-точка». Еще одним примером, но уже для другого уровня, является прикладной протокол управления сетями SNMP. Кадры таких протоколов состоят из неопределен-

го количества полей, каждое из которых может иметь переменную длину. Начало такого кадра отмечается некоторым стандартным образом, например флагом, а затем протокол последовательно просматривает поля кадра и определяет их количество и размеры. Каждое поле обычно описывается двумя дополнительными полями фиксированного размера. Например, если в кадре встречается поле, содержащее некоторую символьную строку, то в кадр вставляются три поля: «Тип», «Длина» и «Значение».

Дополнительные поля «Тип» и «Длина» имеют фиксированный размер в один байт, поэтому протокол легко находит границы поля «Значение». Так как количество таких полей также неизвестно, для определения общей длины кадра используется либо общее поле «Длина», которое помещается в начале кадра и относится ко всем полям данных, либо закрывающий флаг. Подобный способ представления данных переменной длины получил название TLV — аббревиатура от английских слов «Type, Length, Value».

Передача с установлением соединения и без установления соединения

При передаче кадров данных на канальном уровне используются как дейтаграммные процедуры, работающие без установления соединения (connectionless), так и процедуры с предварительным установлением логического соединения (connection-oriented).

При дейтаграммной передаче кадр посыпается сеть «без предупреждения», и никакой ответственности за его утерю протокол не несет (рис. 5.12, а). Предполагается, что сеть всегда готова принять кадр от конечного узла. Дейтаграммный метод работает быстро, так как никаких предварительных действий перед отправкой данных не требуется. Однако при таком методе трудно организовать в рамках протокола отслеживание факта доставки кадра узлу назначения. Этот метод не гарантирует доставку пакета.

Передача с установлением соединения более надежна, но требует больше времени для передачи данных и вычислительных затрат от конечных узлов.

В этом случае узлу-получателю отправляется служебный кадр специального формата с предложением установить соединение (рис. 5.12, б). Если узел-получатель согласен с этим, то он посыпает в ответ другой служебный кадр, подтверждающий установление соединения и предлагающий для данного логического соединения некоторые параметры, которые будут использоваться в рамках данного соединения. Это могут быть, например, идентификатор соединения, максимальное значение поля данных кадров, количество кадров, которые можно отправить без получения подтверждения и т. п. Узел-инициатор соединения может завершить процесс установления соединения отправкой третьего служебного кадра, в котором сообщит, что предложенные параметры ему подходят. На этом логическое соединение считается установленным, и в его рамках можно передавать информационные кадры с пользовательскими данными. После передачи некоторого законченного набора данных, например

определенного файла, узел инициирует разрыв данного логического соединения, посыпая соответствующий служебный кадр.

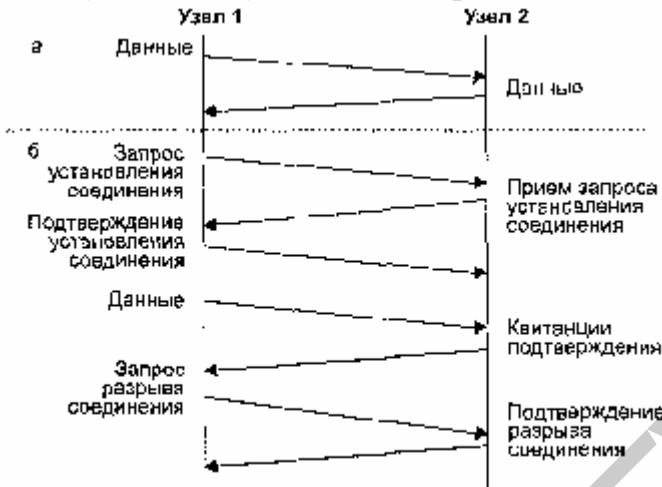


Рис. 5.12. Протоколы без установления соединения (а) и с установлением соединения (б)

Заметим, что, в отличие от протоколов дейтаграммного типа, которые поддерживают только один тип кадра — информационный, протоколы, работающие с установлением соединения, должны поддерживать как минимум два типа кадров — служебные, для установления (и разрыва) соединения, и информационные, переносящие собственно пользовательские данные.

Логическое соединение может быть рассчитано как на передачу данных в одном направлении — от инициатора соединения, так и в обоих направлениях.

Процедура установления соединения может использоваться для достижения различных целей.

- Для взаимной аутентификации либо пользователей, либо оборудования (маршрутизаторы тоже могут иметь имена и пароли, которые нужны для уверенности в том, что злоумышленник не подменил корпоративный маршрутизатор и не отвел поток данных в свою сеть для анализа).
- Для согласования изменяемых параметров протокола: MTU, различных таймаутов и т. п.
- Для обнаружения и коррекции ошибок. Установление логического соединения дает точку отсчета для задания начальных значений номеров кадров. При потере нумерованного кадра приемник, во-первых, получает возможность обнаружить этот факт, а во-вторых, он может сообщить передатчику, какой в частности кадр нужно передать повторно.
- В некоторых технологиях процедуру установления логического соединения используют при динамической настройке коммутаторов сети для маршрутизации всех последующих кадров, которые будут проходить через сеть в рамках данного логического соединения. Так работают сети технологий X.25, frame relay и ATM.

Как видно из приведенного списка, при установлении соединения могут преследоваться разные цели, в некоторых случаях — несколько одновременно. Здесь мы рассмотрим использование логического соединения для обнаружения и коррекции ошибок, а остальные случаи будут рассматриваться в последующих главах по мере необходимости.

Обнаружение и коррекция ошибок

Канальный уровень должен обнаруживать ошибки передачи данных, связанные с искажением битов в принятом кадре данных или с потерей кадра, и по возможности их корректировать.

Большая часть протоколов канального уровня выполняет только первую задачу — обнаружение ошибок, считая, что корректировать ошибки, то есть повторно передавать данные, содержащие искаженную информацию, должны протоколы верхних уровней. Так работают такие популярные протоколы локальных сетей, как Ethernet, Token Ring, FDDI, а также протоколы глобальных сетей frame relay и ATM. Однако существуют протоколы канального уровня, например LLC2 для локальных сетей или HDLC для глобальных, которые самостоятельно решают задачу восстановления искаженных или потерянных кадров.

Очевидно, что протоколы должны работать наиболее эффективно в типичных условиях сети. Поэтому для сетей, в которых искажения и потери кадров являются очень редкими событиями, разрабатываются протоколы, не предусматривающие процедур устранения ошибок. Действительно, наличие процедур восстановления данных потребовало бы от конечных узлов дополнительных вычислительных затрат, которые в условиях надежной работы сети являлись бы избыточными.

Напротив, если в сети искажения и потери случаются часто, то желательно уже на канальном уровне использовать протокол с коррекцией ошибок, а не оставлять эту работу протоколам верхних уровней. Протоколы верхних уровней, например транспортного или прикладного, работая с большими тайм-аутами, восстанавливают потерянные данные с большой задержкой. В глобальных сетях первых поколений, например сетях X.25, которые работали через ненадежные каналы связи, протоколы канального уровня всегда выполняли процедуры восстановления потерянных и искаженных кадров.

Поэтому нельзя считать, что один протокол лучше другого потому, что он восстанавливает ошибочные кадры, а другой протокол — нет. Каждый протокол должен работать в тех условиях, для которых он разработан.

Методы обнаружения ошибок

Все методы обнаружения ошибок основаны на передаче в составе кадра данных избыточной служебной информации, по которой можно судить с некоторой степенью вероятности о достоверности принятых данных. Эту служебную информацию принято называть контрольной суммой, или последовательностью контроля кадра (Frame Check Sequence, FCS). Контрольная сумма вычисляется как функция от основной информации, причем необязательно только путем суммирования. Принимающая сторона повторно вычисляет контрольную сумму кадра по известному алгоритму и в случае ее совпадения с контрольной суммой, вычисленной передающей стороной, делает вывод о том, что данные

были переданы через сеть корректно.

Существует несколько распространенных алгоритмов вычисления контрольной суммы, отличающихся вычислительной сложностью и способностью обнаруживать ошибки в данных.

Контроль по паритету представляет собой наиболее простой метод контроля данных. В то же время это наименее мощный алгоритм контроля, так как с его помощью можно обнаружить только одиночные ошибки в проверяемых данных. Метод заключается в суммировании по модулю 2 всех битов контролируемой информации. Например, для данных 100101011 результатом контрольного суммирования будет значение 1. Результат суммирования также представляет собой один бит данных, который пересыпается вместе с контролируемой информацией. При искажении в процессе пересылки любого одного бита исходных данных (или контрольного разряда) результат суммирования будет отличаться от принятого контрольного разряда, что говорит об ошибке. Однако двойная ошибка, например 110101010, будет неверно принята за корректные данные. Поэтому контроль по паритету применяется к небольшим порциям данных, как правило, к каждому байту, что дает коэффициент избыточности для этого метода 1/8. Метод редко применяется в вычислительных сетях из-за значительной избыточности и невысоких диагностических способностей.

Вертикальный и горизонтальный контроль по паритету представляет собой модификацию описанного выше метода. Его отличие состоит в том, что исходные данные рассматриваются в виде матрицы, строки которой составляют байты данных. Контрольный разряд подсчитывается отдельно для каждой строки и для каждого столбца матрицы. Этот метод обнаруживает большую часть двойных ошибок, однако обладает еще большей избыточностью. На практике сейчас также почти не применяется.

Циклический избыточный контроль (Cyclic Redundancy Check, CRC) является в настоящее время наиболее популярным методом контроля в вычислительных сетях (и не только в сетях, например, этот метод широко применяется при записи данных на гибкие и жесткие диски). Метод основан на рассмотрении исходных данных в виде одного многоразрядного двоичного числа. Например, кадр стандарта Ethernet, состоящий из 1024 байт, будет рассматриваться как одно число, состоящее из 8192 бит. В качестве контрольной информации рассматривается остаток от деления этого числа на известный делитель R. Обычно в качестве делителя выбирается семнадцати- или тридцатиразрядное число, чтобы остаток от деления имел длину 16 разрядов (2 байт) или 32 разряда (4 байт). При получении кадра данных снова вычисляется остаток от деления на тот же делитель R, но при этом к данным кадра добавляется и содержащаяся в нем контрольная сумма. Если остаток от деления на R равен нулю, то делается вывод об отсутствии ошибок в полученном кадре, в противном случае кадр считается искаженным.

Этот метод обладает более высокой вычислительной сложностью, но его диагностические возможности гораздо выше, чем у методов контроля по паритету. Метод CRC обнаруживает все одиночные ошибки, двойные ошибки и ошибки в нечетном числе битов. Метод обладает также невысокой степенью избыточно-

сти. Например, для кадра Ethernet размером в 1024 байт контрольная информация длиной в 4 байт составляет только 0,4 %.

Методы восстановления искаженных и потерянных кадров

Методы коррекции ошибок в вычислительных сетях основаны на повторной передаче кадра данных в том случае, если кадр теряется и не доходит до адресата или приемник обнаружил в нем искажение информации. Чтобы убедиться в необходимости повторной передачи данных, отправитель нумерует отправляемые кадры и для каждого кадра ожидает от приемника так называемой положительной квитанции — служебного кадра, извещающего о том, что исходный кадр был получен и данные в нем оказались корректными. Время этого ожидания ограничено — при отправке каждого кадра передатчик запускает таймер, и если по его истечении положительная квитанция не получена, кадр считается утерянным. Приемник в случае получения кадра с искаженными данными может отправить отрицательную квитанцию — явное указание на то, что данный кадр нужно передать повторно.

Существует два подхода к организации процесса обмена квитанциями: с простыми и с организацией «окна».

Метод с простыми (Idle Source) требует, чтобы источник, пославший кадр, ожидал получения квитанции (положительной или отрицательной) от приемника и только после этого посыпал следующий кадр (или повторял искаженный). Если же квитанция не приходит в течение тайм-аута, то кадр (или квитанция) считается утерянным и его передача повторяется. На рис. 5.13, а видно, что в этом случае производительность обмена данными существенно снижается, — хотя передатчик и мог бы послать следующий кадр сразу же после отправки предыдущего, он обязан ждать прихода квитанции. Снижение производительности этого метода коррекции особенно заметно на низкоскоростных каналах связи, то есть в территориальных сетях.

Второй метод называется методом «скользящего окна» (sliding window). В этом методе для повышения коэффициента использования линии источнику разрешается передать некоторое количество кадров в непрерывном режиме, то есть в максимально возможном для источника темпе, без получения на эти кадры положительных ответных квитанций. (Далее, где это не искажает существо рассматриваемого вопроса, положительные квитанции для краткости будут называться просто «квитациями».) Количество кадров, которые разрешается передавать таким образом, называется размером окна. Рисунок 5.13, б иллюстрирует данный метод для окна размером в W кадров.

В начальный момент, когда еще не послано ни одного кадра, окно определяет диапазон кадров с номерами от 1 до W включительно. Источник начинает передавать кадры и получать и ответ квитанции. Для простоты предположим, что квитанции поступают в той же последовательности, что и кадры, которым они соответствуют. В момент t при получении первой квитанции K , окно сдвигается на одну позицию, определяя новый диапазон от 2 до ($W+1$).

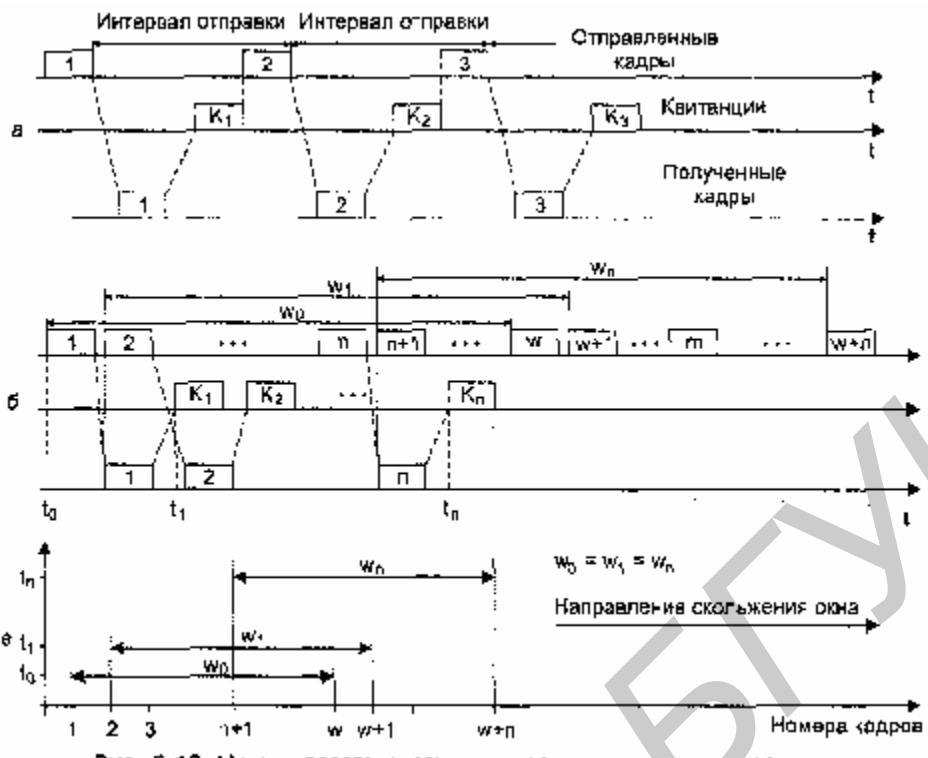


Рис. 5.13. Методы восстановления искаженных и потерянных кадров

Процессы отправки кадров и получения квитанций идут достаточно независимо друг от друга. Рассмотрим произвольный момент времени t_n , когда источник получил квитанцию на кадр с номером n . Окно сдвинулось вправо и определило диапазон разрешенных к передаче кадров от $(n + 1)$ до $(w + n)$. Все множество кадров, выходящих из источника, можно разделить на перечисленные ниже группы (см. рис. 5.13, б).

- Кадры с номерами от 1 до n уже были отправлены и квитанции на них получены, то есть они находятся за пределами окна слева.
- Кадры, начиная с номера $(n + 1)$ и кончая номером $(w + n)$, находятся в пределах окна и потому могут быть отправлены, не дожидаясь прихода какой-либо квитанции. Этот диапазон может быть разделен еще на два поддиапазона:
 - кадры с номерами от $(n + 1)$ до m , которые уже отправлены, но квитанции на них еще не получены;
 - кадры с номерами от m до $(w + n)$, которые пока не отправлены, хотя запрета на это нет.
- Все кадры с номерами, большими или равными $(w + n + 1)$, находятся за пределами окна справа и поэтому пока не могут быть отправлены.

Перемещение окна вдоль последовательности номеров кадров иллюстрирует рис. 5.13, в. Здесь t_u — исходный момент, t и t_n — моменты прихода квитанций на первый и n -й кадр соответственно. Каждый раз, когда приходит квитанция, окно сдвигается влево, но его размер при этом не меняется и остается равным w . Заметим, что хотя в данном примере размер окна в процессе передачи остается постоянным, в реальных протоколах (например, TCP) можно встретить варианты данного алгоритма с изменяющимся размером окна.

Итак, при отправке кадра с номером n источнику разрешается передать еще $w -$

1 кадров до получения квитанции на кадр n , так что в сеть последним уйдет кадр с номером $(w + n - 1)$. Если же за это время квитанция на кадр n так и не пришла, то процесс передачи приостанавливается, и по истечении некоторого тайм-аута кадр n (или квитанция на него) считается утерянным, и он передается снова.

Если же поток квитанций поступает более-менее регулярно, в пределах допуска в W кадров, то скорость обмена достигает максимально возможной величины для данного канала и принятого протокола.

Метод скользящего окна более сложен в реализации, чем метод с простоями, так как передатчик должен хранить в буфере все кадры, на которые пока не получены положительные квитанции. Кроме того, требуется отслеживать несколько параметров алгоритма: размер окна w , номер кадра, на который получена квитанция, номер кадра, который еще можно передать до получения новой квитанции.

В некоторых реализациях скользящего окна от приемника не требуется посыпать квитанции на каждый принятый корректный кадр. Если среди пришедших кадров нет «прогалин», то приемнику достаточно послать квитанцию только на последний принятый кадр, и эта квитанция будет свидетельствовать отправителю о том, что все предыдущие кадры также дошли благополучно.

Другие методы используют отрицательные квитанции. Отрицательные квитанции бывают двух типов — групповые и избирательные. Групповая квитанция содержит номер кадра, начиная с которого нужно повторить передачу всех кадров, отправленных передатчиком в сеть. Избирательная отрицательная квитанция требует повторной передачи только одного кадра.

Метод скользящего окна имеет два параметра, которые могут заметно влиять на эффективность передачи данных между передатчиком и приемником, — размер окна и величина тайм-аута ожидания квитанции.

В надежных сетях, когда кадры искажаются и теряются редко, для повышения скорости обмена данными размер окна нужно увеличивать, так как при этом передатчик будет посылать кадры с меньшими паузами. В ненадежных сетях размер окна следует уменьшать, так как при частых потерях и искажениях кадров резко возрастает объем вторично передаваемых через сеть кадров, а значит, пропускная способность сети во многом расходуется вхолостую и полезная пропускная способность сети падает.

Выбор тайм-аута зависит не от надежности сети, а от задержек передачи кадров сетью. Во многих реализациях метода скользящего окна величина окна и тайм-аут выбираются адаптивно, в зависимости от текущего состояния сети.

Метод скользящего окна реализован во многих протоколах: LLC2, LAP-B, X.25, TCP, Novell NCP Burst Mode.

Метод с простоями является частным случаем метода скользящего окна, когда размер окна равен единице.

Компрессия данных

Компрессия, или сжатие, данных применяется для сокращения времени их пе-

редачи. Так как на компрессию данных передающая сторона тратит дополнительное время, к которому нужно еще прибавить аналогичные затраты времени на декомпрессию этих данных принимающей стороной, то выгоды от сокращения времени на передачу сжатых данных обычно бывают заметны только для низко-скоростных каналов. Этот порог скорости для современной аппаратуры составляет около 64 кбит/с. Многие программные и аппаратные средства сети способны выполнять динамическую компрессию данных в отличие от статической, когда данные предварительно компрессируются (например, с помощью популярных архиваторов типа WinZip), а уже затем отсылаются в сеть.

На практике может использоваться ряд алгоритмов компрессии, каждый из которых применим к определенному типу данных. Некоторые модемы (называемые интеллектуальными) предлагают адаптивную компрессию, при которой в зависимости от передаваемых данных выбирается определенный алгоритм компрессии. Рассмотрим некоторые из общих алгоритмов компрессии данных. Десятичная упаковка. Когда данные состоят только из чисел, значительную экономию можно получить путем уменьшения количества используемых на цифру битов с 7 до 4, используя простое двоичное кодирование десятичных цифр вместо кода ASCII. Просмотр таблицы ASCII показывает, что старшие три бита всех кодов десятичных цифр содержат комбинацию 011. Если все данные в кадре информации состоят из десятичных цифр, то, поместив в заголовок кадра соответствующий управляющий символ, можно существенно сократить длину кадра.

Относительное кодирование. Альтернативой десятичной упаковке при передаче числовых данных с небольшими отклонениями между последовательными цифрами является передача только этих отклонений имеете с известным опорным значением. Такой метод используется, в частности, в рассмотренном выше методе цифрового кодирования голоса ADPCM, передающем в каждом такте только разницу между соседними замерами голоса.

Символьное подавление. Часто передаваемые данные содержат большое количество повторяющихся байтов. Например, при передаче черно-белого изображения черные поверхности будут порождать большое количество нулевых значений, а максимально освещенные участки изображения — большое количество байтов, состоящих из всех единиц. Передатчик сканирует последовательность передаваемых байтов и, если обнаруживает последовательность из трех или более одинаковых байтов, заменяет ее специальной трехбайтовой последовательностью, в которой указывает значение байта, количество его повторений, а также отмечает начало этой последовательности специальным управляющим символом.

Коды переменной длины. В этом методе кодирования используется тот факт, что не все символы в передаваемом кадре встречаются с одинаковой частотой. Поэтому во многих схемах кодирования коды часто невстречающихся символов заменяют кодами меньшей длины, а редко встречающихся — кодами большей длины.

Такое кодирование называется также статистическим кодированием. Из-за того, что символы имеют разную длину, для передачи кадра возможна только

бит-ориентированная передача.

При статистическом кодировании коды выбираются таким образом, чтобы при анализе последовательности битов можно было бы однозначно определить соответствие определенной порции битов тому или иному символу или же запрещенной комбинации битов. Если данная последовательность битов представляет собой запрещенную комбинацию, то необходимо к ней добавить еще один бит и повторить анализ. Например, если при неравномерном кодировании для наиболее часто встречающегося символа «Р» выбран код 1, состоящий из одного бита, то значение 0 однобитного кода будет запрещенным. Иначе мы сможем закодировать только два символа. Для другого часто встречающегося символа «О» можно использовать код 01, а код 00 оставить как запрещенный. Тогда для символа «А» можно выбрать код 001, для символа «П» — код 0001 и т. п.

Вообще, неравномерное кодирование наиболее эффективно, когда неравномерность распределения частот передаваемых символов велика, как при передаче длинных текстовых строк. Напротив, при передаче двоичных данных, например кодов программ, оно малоэффективно, так как 8-битные коды при этом распределены почти равномерно.

Одним из наиболее распространенных алгоритмов, на основе которых строятся неравномерные коды, является алгоритм Хафмана, позволяющий строить коды автоматически, на основании известных частот появления символов. Существуют адаптивные модификации метода Хафмана, которые позволяют строить дерево кодов «на ходу», по мере поступления данных от источника.

Многие модели коммуникационного оборудования, такие как модемы, мосты, коммутаторы и маршрутизаторы, поддерживают протоколы динамической компрессии, позволяющие сократить объем передаваемой информации в 4, а иногда и в 8 раз. В таких случаях говорят, что протокол обеспечивает коэффициент сжатия 1:4 или 1:8. Существуют стандартные протоколы компрессии, например V.42bis, а также большое количество нестандартных, фирменных протоколов. Реальный коэффициент компрессии зависит от типа передаваемых данных, так, графические и текстовые данные обычно сжимаются хорошо, а коды программ — хуже.

Выводы

- Основной задачей протоколов канального уровня является доставка кадра узлу назначения в сети определенной технологии и достаточно простой (регулярной) топологии.
- Асинхронные протоколы разрабатывались для обмена данными между низкоскоростными старт-стопными устройствами: телетайпами, алфавитно-цифровыми терминалами и т. п. В этих протоколах для управления обменом данными используются не кадры, а отдельные символы из нижней части кодовых таблиц ASCII или EBCDIC. Пользовательские данные могут оформляться в кадры, но байты в таких кадрах всегда отделяются друг от друга стартовыми и стоповыми сигналами.
- Синхронные протоколы посылают данные непрерывным битовым потоком,

без разделения его на байты.

- В зависимости от способа выделения начала и конца кадра синхронные протоколы делятся на символьно-ориентированные и бит-ориентированные. В первых для этой цели используются символы кодов ASCII или EBCDIC, а в последних — специальный набор битов, называемый флагом. Бит-ориентированные протоколы более рационально расходуют поле данных кадра, так как для исключения из него значения, совпадающего с флагом, добавляют к нему только один дополнительный бит, а символьно-ориентированные протоколы добавляют целый символ.
- В дейтаграммных протоколах отсутствует процедура предварительного установления соединения, за счет этого срочные данные отправляются в сеть без задержек.
- Протоколы с установлением соединения могут обладать многими дополнительными свойствами, отсутствующими у дейтаграммных протоколов. Наиболее часто в них реализуется такое свойство, как способность восстанавливать искаженные и потерянные кадры.
- Для обнаружения искажений наиболее популярны методы, основанные на циклических избыточных кодах (CRC), которые выявляют многократные ошибки.
- Для восстановления кадров используется метод повторной передачи на основе квитанций. Этот метод работает по алгоритму с простоями источника, а также по алгоритму скользящего окна.
- Для повышения полезной скорости передачи данных в сетях применяется динамическая компрессия данных на основе различных алгоритмов. Коэффициент сжатия зависит от типа данных и применяемого алгоритма и может колебаться в пределах от 1:2 до 1:8.

ГЛАВА 6 Первичные сети

Первичные, или опорные, сети предназначены для создания коммутируемой инфраструктуры, с помощью которой можно достаточно быстро и гибко организовать постоянный канал с топологией «точка-точка» между двумя пользовательскими устройствами, подключенными к такой сети. Так как канал является коммутируемым и по истечении какого-то, пусть и большого, периода времени коммутация разрывается, то такие каналы называют также полупостоянными (*semipermanent*). На основе полупостоянных каналов, образованных первичными сетями, работают вторичные сети — компьютерные или телефонные. Так как последние работают «поверх» инфраструктуры первичной сети, то их иногда называют также наложенными (*overlay*). Каналы, предоставляемые первичными сетями своим пользователям, отличаются высокой пропускной способностью — обычно от 2 Мбит/с до 10 Гбит/с.

Коммутация каналов — основной принцип первичных сетей

Первичные сети основаны на технике коммутации каналов. Для создания аб-

нентского канала коммутаторы первичных сетей должны поддерживать какую-либо технику мультиплексирования и коммутации.

В настоящее время для мультиплексирования абонентских каналов используются:

- техника частотного мультиплексирования (Frequency Division Multiplexing, FDM);
- техника мультиплексирования с разделением времени (Time Division Multiplexing, TDM);
- техника мультиплексирования по длине волны (Wave Division Multiplexing).

Коммутация каналов на основе частотного и волнового мультиплексирования

Техника частотного мультиплексирования каналов (FDM) была разработана для телефонных сетей, но применяется она и для других видов сетей, например сетей кабельного телевидения.

Рассмотрим особенности этого вида мультиплексирования на примере телефонной сети.

Речевые сигналы имеют спектр шириной примерно в 10 000 Гц, однако основные гармоники укладываются в диапазон от 300 до 3400 Гц. Поэтому для качественной передачи речи достаточно образовать между двумя собеседниками канал с полосой пропускания в 3100 Гц, который и используется в телефонных сетях для соединения двух абонентов. В то же время полоса пропускания кабельных систем с промежуточными усилителями, соединяющими телефонные коммутаторы между собой, обычно составляет сотни килогерц, а иногда и сотни мегагерц. Однако непосредственно передавать сигналы нескольких абонентских каналов по широкополосному каналу невозможно, так как все они работают в одном и том же диапазоне частот и сигналы разных абонентов смешиваются между собой так, что разделить их невозможно.

Для разделения абонентских каналов характерна техника модуляции высокочастотного несущего синусоидального сигнала низкочастотным речевым сигналом (рис. 6.1). Эта техника подобна технике аналоговой модуляции при передаче дискретных сигналов модемами, только вместо дискретного исходного сигнала используются непрерывные сигналы, порождаемые звуковыми колебаниями. В результате спектр модулированного сигнала переносится в другой диапазон, который симметрично располагается относительно несущей частоты и имеет ширину, приблизительно совпадающую с шириной модулирующего сигнала.

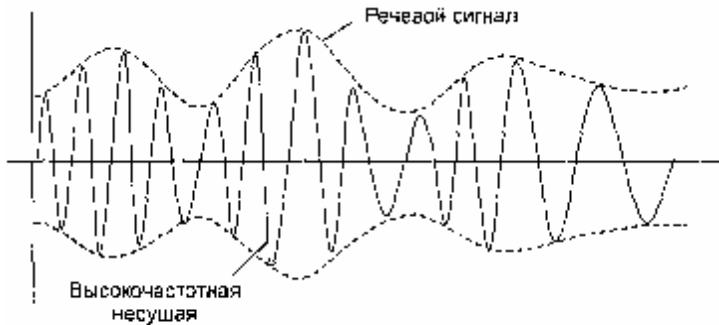


Рис. 5.1. Модуляция речевым сигналом

Если сигналы каждого абонентского канала перенести в собственный диапазон частот, то в одном широкополосном канале можно одновременно передавать сигналы нескольких абонентских каналов.

На входы FDM-коммутатора поступают исходные сигналы от абонентов телефонной сети. Коммутатор выполняет перенос частоты каждого канала в свой диапазон частот. Обычно высокочастотный диапазон делится на полосы, которые отводятся для передачи данных абонентских каналов (рис. 6.2). Чтобы низкочастотные составляющие сигналов разных каналов не смешивались между собой, полосы делают шириной в 4 кГц, а не в 3,1 кГц, оставляя между ними страховую промежуток в 900 Гц. В канале между двумя FDM-коммутаторами одновременно передаются сигналы всех абонентских каналов, но каждый из них занимает свою полосу частот. Такой канал называют уплотненным.

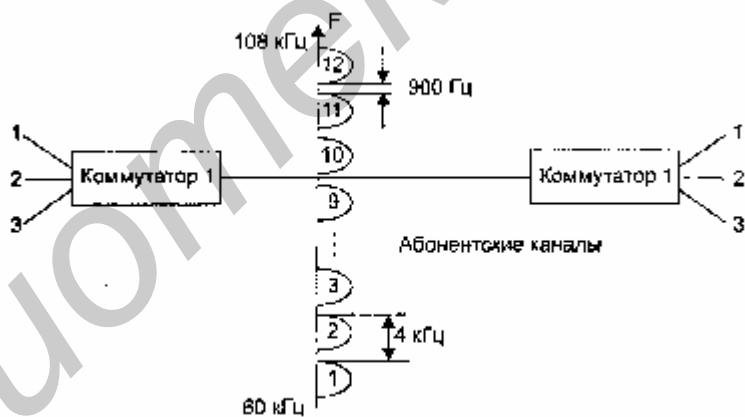


Рис. 6.2. Коммутация на основе частотного уплотнения

Выходной FDM-коммутатор выделяет модулированные сигналы каждой несущей частоты и передает их на соответствующий выходной канал, к которому непосредственно подключен абонентский телефон.

В сетях на основе FDM-коммутации принято несколько уровней иерархии уплотненных каналов. Первый уровень уплотнения образуют 12 абонентских каналов, которые составляют базовую группу каналов, занимающую полосу частот шириной в 48 кГц с границами от 60 до 108 кГц. Второй уровень уплотнения образуют 5 базовых групп, которые составляют супергруппу, с полосой частот шириной в 240 кГц и границами от 312 до 552 кГц. Супергруппа передает данные 60 абонентских каналов тональной частоты. Десять супергрупп обра-

зуют главную группу, которая используется для связи между коммутаторами на больших расстояниях. Главная группа передает данные 600 абонентов одновременно и требует от канала связи полосу пропускания шириной не менее 2520 кГц с границами от 564 до 3084 кГц.

FDM-коммутаторы могут выполнять как динамическую, так и постоянную коммутацию. При динамической коммутации один абонент инициирует соединение с другим абонентом, посыпая в сеть номер вызываемого абонента. Коммутатор динамически выделяет данному абоненту одну из свободных полос своего уплотненного канала. При постоянной коммутации за абонентом полоса в 4 кГц закрепляется на длительный срок путем настройки коммутатора по отдельному входу, недоступному пользователям.

Принцип коммутации на основе разделения частот остается неизменным и в сетях другого вида, меняются только границы полос, выделяемых отдельному абонентскому каналу, а также количество низкоскоростных каналов в уплотненном высокоскоростном.

Первичные сети с мультиплексированием по длине волны (WDM и DWDM) используют тот же принцип частотного разделения каналов, но только информационным сигналом в них является не электрический ток, а свет. Соответственно, изменяется и частотный диапазон, в котором образуются пользовательские полупостоянные каналы — это инфракрасный диапазон с длинами волн от 850 до 1565 им, что соответствует частотам от 196 до 350 ТГц (1 Т Гц равен 10^{12} Гц).

В магистральном канале обычно мультиплексируется несколько спектральных каналов — до 16, 32, 40, 80 или 160 (начиная с 16 каналов, такая техника мультиплексирования называется обычно плотной, то есть Dense WDM или DWDM). Внутри такого спектрального канала данные могут кодироваться как дискретным способом, так и аналоговым. По сути, WDM и DWDM — это реализации идеи частотного аналогового мультиплексирования, но в другой форме. Отличие сетей WDM/DWDM от сетей FDM в предельных скоростях, с которыми они могут передавать информацию. Если сети FDM обычно обеспечивают на магистральных каналах одновременную передачу до 600 разговоров, что соответствует суммарной скорости в 36 Мбит/с (для сравнения с цифровыми каналами скорость пересчитана из расчета 64 кбит/с на один разговор), то сети DWDM обеспечивают общую пропускную способность до сотен гигабит и даже нескольких терабит в секунду.

Более подробно технология DWDM рассматривается в разделе «Сети DWDM» данной главы.

Коммутация каналов на основе разделения времени

Коммутация на основе техники разделения частот разрабатывалась в расчете на передачу непрерывных сигналов, представляющих голос. При переходе к цифровой форме представления голоса была разработана новая техника мультиплексирования, ориентирующаяся на дискретный характер передаваемых данных.

Эта техника носит название мультиплексирования с разделением времени (Time Division Multiplexing, TDM)¹. Реже используется и другое ее название — техника синхронного режима передачи (Synchronous Transfer Mode, STM). Рисунок 6.3 поясняет принцип коммутации каналов на основе техники TDM.

Аппаратура TDM-сетей — мультиплексоры, коммутаторы, демультиплексоры — работает в режиме разделения времени, поочередно обслуживая в течение цикла своей работы все абонентские каналы. Цикл работы оборудования TDM равен 125 мкс, что соответствует периоду следования замеров голоса в цифровом абонентском канале. Это значит, что мультиплексор или коммутатор успевает вовремя обработать любой абонентский канал и передать его очередной замер далее по сети. Каждому соединению выделяется один квант времени цикла работы аппаратуры, называемый также тайм-слотом. Длительность тайм-слота зависит от числа абонентских каналов, обслуживаемых мультиплексором TDM или коммутатором.

(Формальную аббревиатуру TDM можно применять и к сетям с коммутацией пакетов, так как там также имеет место режим разделения времени, но только асинхронный. На практике же аббревиатура TDM используется без уточняющих режим работы терминов «синхронный» или «асинхронный» и относится всегда к синхронному варианту TDM.)

Мультиплексор принимает информацию по N входным каналам от конечных абонентов, каждый из которых передает данные по абонентскому каналу со скоростью 64 кбит/с — 1 байт каждые 125 мкс. В каждом цикле мультиплексор выполняет следующие действия:

- прием от каждого канала очередного байта данных;
- составление из принятых байтов уплотненного кадра, называемого также обоймой;
- передача уплотненного кадра на выходной канал с битовой скоростью, равной N x 64 кбит/с.

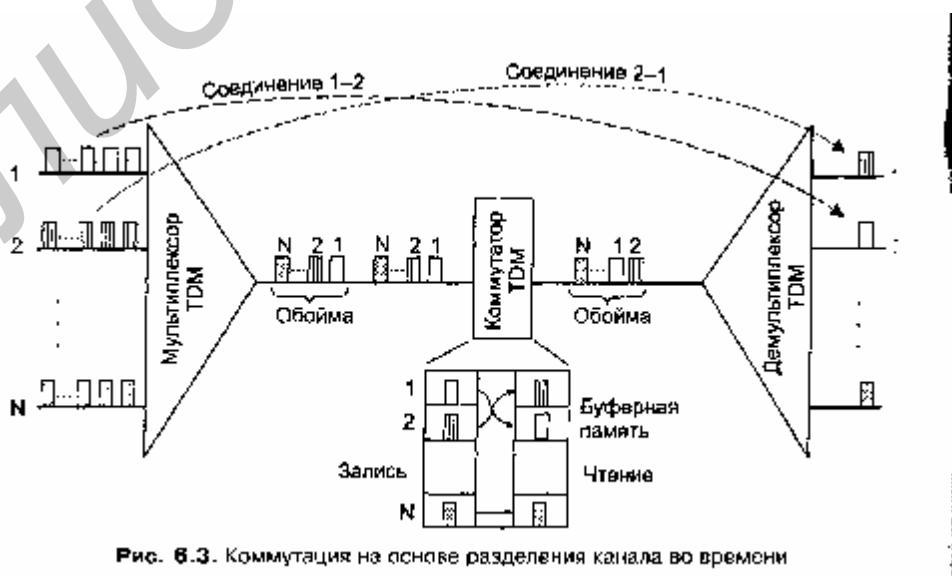


Рис. 6.3. Коммутация на основе разделения канала во времени

Порядок следования байта в обойме соответствует номеру сходного канала, от

которого этот байт получен. Количество обслуживаемых мультиплексором абонентских каналов зависит от его быстродействия. Например, мультиплексор T1, представляющий собой первый промышленный мультиплексор, работавший по технологии TDM, поддерживает 24 входных абонентских канала, создавая на выходе обоймы стандарта T1, передаваемые с битовой скоростью 1,544 Мбит/с.

Демультиплексор выполняет обратную задачу — он разбирает байты уплотненного кадра и распределяет их по своим нескольким выходным каналам, при этом он считает, что порядковый номер байта в обойме соответствует номеру выходного канала.

Коммутатор принимает уплотненный кадр по скоростному каналу от мультиплексора и записывает каждый байт из него в отдельную ячейку своей буферной памяти, причем в том порядке, и котором эти байты были упакованы в уплотненный кадр. Для выполнения операции коммутации байты извлекаются из буферной памяти не в порядке поступления, а в том порядке, который соответствует поддерживаемым в сети соединениям абонентов. Так, например, если первый абонент левой части сети рис. 6.3 должен соединиться со вторым абонентом в правой части сети, то байт, записанный в первую ячейку буферной памяти, будет извлекаться из нее вторым. «Перемешивая»- нужным образом байты в обойме, коммутатор обеспечивает соединение конечных абонентов в сети.

Однажды выделенный номер тайм-слота остается в распоряжении соединения «входной канал — выходной слот» в течение всего времени существования этого соединения, даже если передаваемый трафик является пульсирующим и не всегда требует захваченного количества тайм-слотов. Это означает что соединение в сети TDM всегда обладает известной и фиксированной пропускной способностью, кратной 64 кбит/с.

Работа оборудования TDM напоминает работу сетей с коммутацией пакетов, так как каждый байт данных можно считать некоторым элементарным пакетом. Однако в отличие от пакета компьютерной сети, «пакет» сети TDM не имеет индивидуального адреса. Его адресом является порядковый номер в обойме или номер выделенного тайм-слота в мультиплексоре или коммутаторе. Сети, использующие технику TDM, требуют синхронной работы всего оборудования, что и определило второе название этой техники — синхронный режим передач (STM). Нарушение синхронности разрушает требуемую коммутацию абонентов, так как при этом теряется адресная информация. Поэтому перераспределение тайм-слотов между различными каналами и оборудований TDM невозможно, даже если в каком-то цикле работы мультиплексора тайм-слот одного из каналов оказывается избыточным, так как на входе этого канала в этот момент нет данных для передачи (например, абонент телефонной сети молчит).

Существует модификация техники TDM, называемая статистическим разделением канала во времени (Statistical TDM, STDM). Эта техника разработана специально для того, чтобы с помощью временно свободных тайм-слотов одного канала можно было увеличить пропускную способность остальных. Для решения этой задачи каждый байт данных дополняется полем адреса небольшой длины, например в 4 или 5 бит, что позволяет мультиплексировать 16 или 32

канала. Фактически STDM представляет собой уже технику коммутации пакетов, но только с очень упрощенной адресацией и узкой областью применения. Техника STDM не нашла широкого применения и используется в основном в нестандартном оборудовании подключения терминалов к мэйнфреймам. Развитием идей статистического мультиплексирования стала технология асинхронного режима передачи — ATM, которая вобрала в себя лучшие черты техники коммутации каналов и пакетов.

Сети TDM могут поддерживать либо режим динамической коммутации, либо режим полупостоянной коммутации, а иногда и оба этих режима. Так, например, основным режимом цифровых телефонных сетей, работающих на основе технологии TDM, является динамическая коммутация, но они поддерживают также и постоянную коммутацию, предоставляя своим абонентам службу выделенных каналов. Первичные сети режим динамической коммутации не используют, хотя перспективы предоставлении клиентам первичной сети возможности коммутировать канал по своей инициативе в настоящее время обсуждаются.

Существует аппаратура, которая поддерживает только режим постоянной коммутации. К ней относится оборудование типа T1/E1, а также высокоскоростное оборудование SDH. Такое оборудование используется для построения первичных сетей, основной функцией которых является создание выделенных каналов между коммутаторами, поддерживающими динамическую коммутацию.

Сегодня практически все данные — голос, изображение, компьютерные данные — передаются в цифровой форме. Поэтому выделенные каналы TDM-технологии, которые обеспечивают нижний уровень для передачи цифровых данных, являются универсальными каналами для построения сетей любого типа: телефонных, телевизионных и компьютерных.

Общие свойства сетей с коммутацией каналов

Сети с коммутацией каналов обладают несколькими важными общими свойствами независимо от того, какой тип мультиплексирования в них используется. Сети с динамической коммутацией требуют предварительной процедуры установления соединения между абонентами. Для этого в сеть передается адрес вызываемого абонента, который проходит через коммутаторы и настраивает их на последующую передачу данных. Запрос на установление соединения маршрутизируется от одного коммутатора к другому и в конце концов, достигает вызываемого абонента. Сеть может отказать в установлении соединения, если емкость требуемого выходного канала уже исчерпана. Для FDM-коммутатора емкость выходного канала равна количеству частотных полос этого канала, а для TDM-коммутатора — количеству тайм-слотов, на которые делится цикл работы канала. Сеть отказывает в соединении также в том случае, если запрашиваемый абонент уже установил соединение с кем-нибудь другим. В первом случае говорят, что занят коммутатор, а во втором — абонент. Возможность отказа в соединении является недостатком метода коммутации каналов.

Если соединение может быть установлено, то ему выделяется фиксированная

полоса частот в FDM-сетях или же фиксированная пропускная способность в TDM-сетях. Эти величины остаются неизменными и течение всего периода соединения. Гарантированная пропускная способность сети после установления соединения является важным свойством, необходимым для таких приложений, как передача голоса, изображения или управления объектами в реальном масштабе времени. Однако динамически изменять пропускную способность канала по требованию абонента сети с коммутацией каналов не могут, что делает их неэффективными в условиях пульсирующего трафика.

Недостатком сетей с коммутацией каналов является невозможность применения пользовательской аппаратуры, работающей с разной скоростью. Отдельные части составного канала работают с одинаковой скоростью, так как сети с коммутацией каналов не буферизуют данные пользователей.

Сети с коммутацией каналов хорошо приспособлены для коммутации потоков данных постоянной скорости, когда единицей коммутации является не отдельный байт или пакет данных, а долговременный синхронный поток данных между двумя абонентами. Для таких потоков сети с коммутацией каналов добавляют минимум служебной информации для маршрутизации данных через сеть, используя временную позицию каждого бита потока в качестве его адреса назначения в коммутаторах сети.

Обеспечение дуплексного режима работы на основе технологий FDM, ТОМ и WDM

В зависимости от направления возможной передачи данных способы передачи данных по линии связи делятся на следующие типы:

- симплексный — передача осуществляется по линии связи только в одном направлении;
- полудуплексный — передача ведется и обоих направлениях, но попеременно во времени (примером такой передачи служит технология Ethernet);
- дуплексный — передача ведется одновременно в двух направлениях.

Дуплексный режим — наиболее универсальный и производительный способ работы канала. Самым простым вариантом организации дуплексного режима является использование двух независимых физических каналов (двух пар проводников или двух световодов) в кабеле, каждый из которых работает в симплексном режиме, то есть передает данные в одном направлении. Именно такая идея лежит в основе реализации дуплексного режима работы на многих сетевых технологиях, например Fast Ethernet или ATM.

Иногда такое простое решение оказывается недоступным или неэффективным. Чаще всего это происходит в тех случаях, когда для дуплексного обмена данными имеется всего один физический канал, а организация второго связана с большими затратами. Например, при обмене данными с помощью модемов через телефонную сеть у пользователя имеется только один физический канал связи с АТС — двухпроводная линия, и приобретать второй вряд ли целесообразно. В таких случаях дуплексный режим работы организуется на основе раз-

деления канала на два логических подканала с помощью техники FDM или TDM.

Модемы для организации дуплексного режима работы на двухпроводной линии применяют технику FDM. Модемы, использующие частотную модуляцию, работают на четырех частотах: две частоты — для кодирования единиц и нулей в одном направлении, а остальные две частоты — для передачи данных в обратном направлении.

При цифровом кодировании дуплексный режим на двухпроводной линии организуется с помощью техники TDM. Часть тайм-слотов используется для передачи данных в одном направлении, а часть — для передачи в другом направлении. Обычно тайм-слоты противоположных направлений чередуются, из-за чего такой способ иногда называют «пинг-понговой» передачей. TDM-разделение линии характерно, например, для цифровых сетей с интеграцией услуг (ISDN) на абонентских двухпроводных окончаниях.

В волоконно-оптических кабелях с одним оптическим волокном для организации дуплексного режима работы применяется передача данных в одном направлении с помощью светового пучка одной длины волны, а в обратном — другой длины волны. Собственно, решение частной задачи — создания двух независимых спектральных каналов в окне прозрачности оптического волокна — и привело к созданию новой технологии на основе мультиплексирования по длине волны. Сначала в каждом направлении использовалось по одному каналу, что обеспечивало только дуплексный режим работы, а потом стали применять по 2 или 4 канала, и такая технология получила название Wave Division Multiplexing (WDM). Дальнейшее повышение числа каналов до 16, 32 и 40 заставило несколько изменить название, технология получила приставку *dense* (плотное) — Dense WDM, или DWDM.

Выводы

- Первичные, или опорные, сети предназначены для создания коммутируемой инфраструктуры, с помощью которой можно достаточно быстро и гибко организовать постоянный канал с топологией «точка-точка» между двумя пользовательскими устройствами, подключенными к такой сети.
- Современные первичные сети используют технику коммутации каналов различного типа: с частотным мультиплексированием (FDM), с разделением времени (TDM), с мультиплексированием по длине волны (WDM/DWDM).
- В сетях FDM каждому абонентскому каналу выделяется полоса частот шириной ~4 кГц. Существует иерархия каналов FDM, при этом 12 абонентских каналов образуют группу каналов первого уровня иерархии (базовую группу) с полосой 48 кГц, 5 каналов первого уровня объединяются в канал второго уровня иерархии (супергруппу) с полосой 240 кГц, а 10 каналов второго уровня составляют канал третьего уровня иерархии (главную группу) с полосой в 2,4 мГц.

Сети PDH

Существует два поколения технологии цифровых первичных сетей — технология плезиохронной («плезио» означает «почти», то есть почти синхронной) цифровой иерархии (Plesiochronous Digital Hierarchy. PDH) и более поздняя технология — синхронная цифровая иерархия (Synchronous Digital Hierarchy, SDH). В Америке технологии SDH соответствует стандарт SONET.

Цифровая аппаратура мультиплексирования и коммутации была разработана в конце 60-х годов компанией AT&T для решения проблемы связи крупных коммутаторов телефонных сетей между собой. Каналы с частотным уплотнением, применяемые до этого на участках АТС—АТС, исчерпали свои возможности по организации высокоскоростной многоканальной связи по одному кабелю. В технологии FDM для одновременной передачи данных 12 или 60 абонентских каналов использовалась витая пара, а для повышения скорости связи приходилось прокладывать кабели с большим количеством пар проводов или более дорогие коаксиальные кабели. Кроме того, метод частотного уплотнения высокочувствителен к разного рода помехам, которые всегда присутствуют в территориальных кабелях, да и высокочастотная несущая речи сама создает помехи в приемной аппаратуре, будучи плохо отфильтрованной.

Иерархия скоростей

Для решения этих проблем была разработана аппаратура T1, которая позволяла в цифровом виде мультиплексировать, передавать и коммутировать (на постоянной основе) данные 24 абонента. Так как абоненты по-прежнему пользовались обычными телефонными аппаратами, то есть передача голоса шла в аналоговой форме, то мультиплексоры T1 сами осуществляли оцифровывание голоса с частотой 8000 Гц и кодировали голос путем импульсно-кодовой модуляции (Pulse Code Modulation, PCM). В результате каждый абонентский канал образовывал цифровой поток данных 64 кбит/с. Для соединения магистральных АТС каналы T1 представляли собой слишком слабые средства мультиплексирования, поэтому в технологии была реализована идея образования каналов с иерархией скоростей. Четыре канала типа T1 объединяются в канал следующего уровня цифровой иерархии — T2, передающий данные со скоростью 6,312 Мбит/с, а семь каналов T2 дают при объединении канал T3, передающий данные со скоростью 44,736 Мбит/с. Аппаратура T1, T2 и T3 может взаимодействовать между собой, образуя иерархическую сеть с магистральными и периферийными каналами трех уровней скоростей.

С середины 70-х годов выделенные каналы, построенные на аппаратуре T1, стали сдаваться телефонными компаниями в аренду на коммерческих условиях, перестав быть внутренней технологией этих компаний. Сети T1, а также более скоростные сети T2 и T3, позволяют передавать не только голос, но и любые данные, представленные в цифровой форме, — компьютерные данные, телевизионное изображение, факсы и т. п.

Технология цифровой иерархии была позже стандартизована CCITT. При этом в нее были внесены некоторые изменения, что привело к несовместимости американской и международной версий цифровых сетей. Американская версия распространена сегодня кроме США также в Канаде и Японии (с некоторыми различиями), а в Европе применяется международный стандарт. Аналогом каналов Т в международном стандарте являются каналы типа E1, E2 и E3 с другими скоростями — соответственно 2,048 Мбит/с, 8,488 Мбит/с и 34,368 Мбит/с. Американский вариант технологии также был стандартизован ANSI.

Таблица 6.1. Иерархия цифровых скоростей

Америка			CCITT (Европа)			
Обозначение скорости	Количество голосовых каналов	Количество каналов	Скорость, Мбит/с	Количество голосовых	Количество каналов	предыду- л « ё» 8 »
DS-0	1	1	64	1	1	64
DS-1	24	24	1,544	30	30 -	2,048
DS-2	96	4	6,312	120	4	8,488
DS-3	672	/	44,736	480	4	34,368
ms-4	4032	6	274,176	1920	4	139,264

Несмотря на различия американской и международных версий технологии цифровой иерархии, для обозначения иерархии скоростей принято использовать одни и те же обозначения — DS_n (Digital Signal n). В табл. 6.1 приводятся значения для всех введенных стандартами уровней скоростей обеих технологий.

На практике в основном используются каналы T1/E1 и T3/E3.

Методы мультиплексирования

Мультиплексор T1 обеспечивает передачу данных 24 абонентов со скоростью 1,544 Мбит/с в кадре, имеющем достаточно простой формат. В этом кадре последовательно передается по одному байту каждого абонента, а после 24 байт вставляется один бит синхронизации. Первоначально устройства T1 (которые дали имя также и всей технологии, работающей на скорости 1,544 Мбит/с) работали только на внутренних тактовых генераторах, и каждый кадр с помощью битов синхронизации мог передаваться асинхронно. Аппаратура T1, а также более скоростная аппаратура T2 и T3 за долгие годы существования претерпела значительные изменения. Сегодня мультиплексоры и коммутаторы первичной сети работают на централизованной тактовой частоте, распределаемой из одной точки всей сети. Однако принцип формирования кадра остался, поэтому биты синхронизации в кадре по-прежнему присутствуют. Суммарная скорость пользовательских каналов составляет $24 \times 64 = 1,536$ Мбит/с, а еще 8 кбит/с добавляют биты синхронизации.

В аппаратуре T1 назначение восьмого бита каждого байта в кадре разное и зависит от типа передаваемых данных и поколения аппаратуры. В частности, с помощью этого бита передается служебная информация, переносящая номер вызываемого абонента и другие сведения, необходимые для установления соединения между абонентами сети. Протокол, обеспечивающий такое соединение, называется в телефонии протоколом сигнализации (signaling protocol), этот термин иногда используется и в компьютерных сетях, работающих на основе техники логических соединений (например, frame relay или ATM).

При передаче голоса в сетях T1 все 24 канала являются абонентскими, поэтому управляющая и контрольная информация передается восьмым (наименее значащим) битом замеров голоса. В ранних версиях сетей T1 служебным был 8-й бит каждого байта кадра, поэтому реальная скорость передачи пользовательских данных составляла 56 кбит/с (обычно восьмой бит отводился под такие служебные данные, как номер вызываемого телефонного абонента, сигнал занятости линии, сигнал снятия трубки и т. п.). Затем технология была улучшена, и для служебных целей стали использовать только каждый шестой кадр. Таким образом, в пяти кадрах из шести пользовательские данные представлены всеми восемью битами, а в шестом — только семью.

При передаче компьютерных данных канал T1 предоставляет для пользовательских данных только 23 канала, а 24-й канал отводится для служебных целей, в основном — для восстановления искаженных кадров. Для одновременной передачи как голосовых, так и компьютерных данных используются все 24 канала.

ла, причем компьютерные данные передаются со скоростью 56 кбит/с. Техника использования восьмого бита для служебных целей получила название «кражи бита» (bit robbing).

При мультиплексировании 4 каналов T1 в один канал T2 между кадрами DS-1 по-прежнему используется один бит синхронизации, а кадры DS-2 (которые состоят из 4 последовательных кадров DS-1) разделяются 12 служебными битами, которые предназначены не только для разделения кадров, но и для их синхронизации. Соответственно, кадры DS-3 состоят из 7 кадров DS-2, разделенных служебными битами.

Международная версия этой технологии описана в стандартах G.700-G.706. Она более логична, так как не использует схему «кражи бита». Кроме того, она основана на постоянном коэффициенте кратности скорости 4 при переходе к следующему уровню иерархии. Вместо восьмого бита в канале E1 на служебные цели отводятся два байта из 32, а именно нулевой (для целей синхронизации приемника и передатчика) и шестнадцатый (в нем передается служебная информация сигнализации). Для голосовых каналов или каналов данных остается 30 каналов со скоростью передачи 64 кбит/с каждый.

Пользователь может арендовать несколько каналов 64 кбит/с (56 кбит/с) в канале T1/E1. Такой канал называется «дробным» (fractional) каналом T1/E1. В этом случае пользователю отводится несколько тайм-слотов работы мультиплексора.

Физический уровень технологии PDH поддерживает различные виды кабелей: витую пару, коаксиальный кабель и волоконно-оптический кабель. Основным вариантом абонентского доступа к каналам T1/E1 является кабель из двух витых пар с разъемами RJ-48. Две пары требуются для организации дуплексного режима передачи данных со скоростью 1,544/2,048 Мбит/с. Для представления сигналов используется: в каналах T1 — биполярный потенциальный код B8ZS, в каналах E1 — биполярный потенциальный код HDB3. Для усиления сигнала на линиях T1 через каждые 1800 м (одна миля) устанавливаются регенераторы и аппаратура контроля линии.

Коаксиальный кабель благодаря своей широкой полосе пропускания поддерживает канал T2/E2 или 4 канала T1/E1. Для работы каналов T3/E3 обычно используется либо коаксиальный кабель, либо волоконно-оптический кабель, либо каналы СВЧ.

Физический уровень международного варианта технологии определяется стандартом G.703, названием которого обозначается тип интерфейса маршрутизатора или моста, подключаемого к каналу E1. Американский вариант интерфейса носит название T1.

Ограничения технологии PDH

Как американский, так и международный варианты технологии PDH обладают некоторыми недостатками.

Одним из основных недостатков является сложность и неэффективность операций мультиплексирования и демультиплексирования пользовательских данных.

Сам термин «плезиохронный», используемый для этой технологии, говорит о причине такого явления — отсутствии полной синхронности потоков данных при объединении низкоскоростных каналов и более высокоскоростные. Изначально асинхронный подход к передаче кадров породил вставку опта или нескольких битов синхронизации между кадрами.

В результате для извлечения пользовательских данных из объединенного канала необходимо полностью демультиплексировать кадры этого объединенного канала. Например, чтобы получить данные одного абонентского канала 64 кбит/с. из кадров канала Т3, необходимо произвести демультиплексирование этих кадров до уровня кадров Т2, затем — до уровня кадров Т1. а затем демультиплексировать и сами кадры Т1. Для преодоления этого недостатка в сетях PDH реализуют некоторые дополнительные приемы, уменьшающие количество операции демультиплексирования при извлечении пользовательских данных из высокоскоростных каналов. Например, одним из таких приемом является «обратная доставка» (back hauling). Пусть коммутатор 1 канала Т3 принимает поток данных, состоящий из 672 пользовательских каналов. И пусть при этом он должен передать данные одного из этих каналов пользователю, подключенному к низкоскоростному выходу коммутатора. Весь же остальной поток данных он должен направить транзитом через другие коммутаторы в некоторый конечный демультиплексор 2, где поток Т3 будет полностью разобран на каналы 64 кбит/с.

Для экономии времени коммутатор 1 пропускает операцию демультиплексирования для своего пользователя, откладывая ее на тот момент, когда конечный демультиплексор выполнит операцию разбора кадров и вернет данные одного из каналов коммутатору 1. Естественно, такие сложные взаимоотношения коммутаторов усложняют работу сети, требуют ее тонкого конфигурирования, что ведет к большому объему ручной работы и ошибкам.

Другим существенным недостатком технологии PDH является отсутствие развитых встроенных процедур контроля и управления сетью. Служебные биты дают мало информации о состоянии канала, не позволяют его конфигурировать и т. п. Нет в технологии и процедур поддержки отказоустойчивости, очень полезных для первичных сетей, на основе которых строятся ответственные междугородные и международные сети. В системных сетях управлению уделяется большое внимание, причем считается, что управляющие процедуры желательно встраивать в основной протокол передачи данных сети.

Третий недостаток состоит в слишком низких по современным понятиям скоростях иерархии PDH. Волоконно-оптические кабели позволяют передавать данные со скоростями в несколько гигабит в секунду по одному волокну, что обеспечивает консолидацию в одном кабеле десяткой тысяч пользовательских каналов, но это свойство технология PDH не реализует ее иерархия скоростей заканчивается уровнем 139 Мбит/с.

Все эти недостатки устранены в новой технологии первичных цифровых сетей, получившей название синхронной цифровой иерархии (Synchronous Digital Hierarchy, SDH).

Выводы

- Цифровые первичные сети PDH позволяют образовывать каналы с пропускной способностью от 64 кбит/с до 140 Мбит/с, предоставляя своим абонентам скорости четырех уровней иерархии.
- Первоначально сети PDH (аппаратура канала Т1) предназначались для связи телефонных станций между собой. В американском варианте телефонная сигнализация передается в восьмом бите кадра PDH, международный вариант (E1) предусматривает для этих целей использование 16-го байта кадра.
- Недостатком сетей PDH является невозможность непосредственного выделения данных низкоскоростного канала из данных высокоскоростного канала, если каналы работают на несмежных уровнях иерархии скоростей. Например, нельзя выделить данные канала E1 из данных канала E3 — необходимо последовательно демультиплексировать канал E3 на каналы E2, а затем канал E2 на каналы E1.

Сети SDH

Технология синхронной цифровой иерархии (Synchronous Digital Hierarchy, SDH) разработана для создания надежных транспортных сетей, позволяющих гибко формировать цифровые каналы широкого диапазона скоростей — от единиц мегабит до десятков гигабит в секунду. Основная область применения технологии SDH — первичные сети операторов связи, но иногда такие сети строят и крупные предприятия и организации, имеющие разветвленную структуру подразделений и филиалов, покрывающих большую территорию, например, в сетях предприятий энергетического комплекса или железнодорожных компаний.

Каналы SDH относятся к классу полупостоянных (semipermanent) — формирование (provisioning) канала происходит по инициативе оператора, сети SDH, пользователи же лишены такой возможности, поэтому каналы SDH обычно применяются для передачи достаточно устойчивых во времени потоков. Из-за полупостоянного характера соединений в технологии SDH чаще используется термин кросс-коннект (cross-connect), а не коммутация.

Сети SDH относятся к классу сетей с коммутацией каналов, использующих синхронное мультиплексирование с разделением времени (Time Division Multiplexing, TDM), при котором информация от отдельных абонентов адресуется относительным временным положением внутри составного кадра, а не явным адресом, как это происходит в сетях с коммутацией пакетов.

Каналы SDH обычно применяют для объединения большого количества периферийных (и менее скоростных) каналов, работающих по технологии плезиохронной цифровой иерархии (Plesiochronous Digital Hierarchy, PDH). Пример использования каналов SDH для соединения между собой абонентского оборудования разного типа приведен на рис. 6.4.

Сети SDH обладают многими достоинствами, главные из которых перечислены

ниже.

- Гибкая иерархическая схема мультиплексирования цифровых потоков разных скоростей, позволяющая вводить (add, insert) в магистральный канал и выводить (drop) из него пользовательскую информацию любого поддерживаемого технологией уровня скорости, не мультиплексируя поток в целом — а это означает не только гибкость, но и экономию оборудования. Схема мультиплексирования стандартизована на международном уровне, что обеспечивает совместимость оборудования разных производителей.

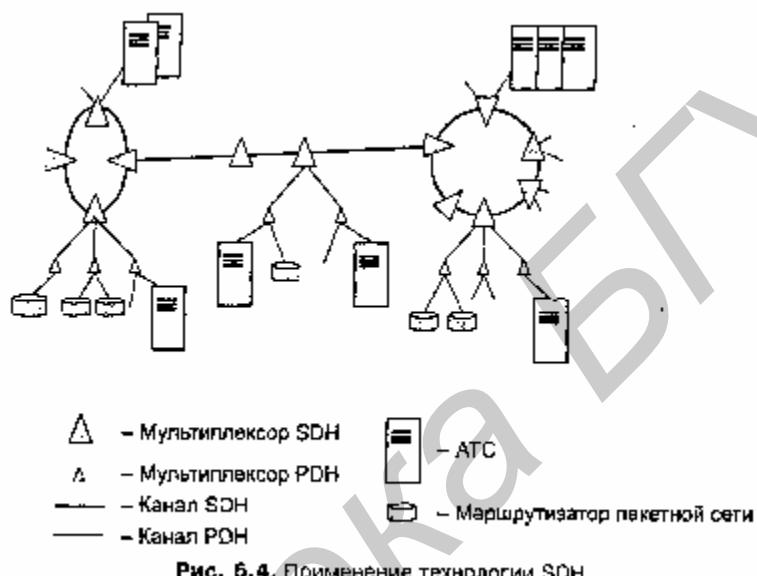


Рис. 6.4. Применение технологии SDH

- Отказоустойчивость сети. Сети SDH обладают высокой степенью «живучести» — технология предусматривает автоматическую реакцию оборудования на такие типичные отказы, как обрыв кабеля, отказ порта, выход из строя мультиплексора или отдельной его карты, направляя трафик по резервному пути или переходя на резервный модуль. Переход на резервный путь происходит очень быстро — обычно в течение 50 мс.
- Мониторинг и управление сетью на основе информации, встроенной в заголовки кадров. Это обеспечивает обязательный уровень управляемости сети, не зависящий от производителя оборудования, и создает основу для наращивания функций менеджмента в фирменных системах управления.
- Высокое качество транспортного обслуживания для трафика любого типа — голосового, видео и компьютерного. Техника мультиплексирования TDM, лежащая в основе SDH, обеспечивает трафику каждого абонента гарантированную пропускную способность, а также низкий и фиксированный уровень задержек.

Сети SDH добились прочного положения в телекоммуникационном мире — сегодня они составляют фундамент практически всех крупных сетей — региональных, национальных и международных. Укрепляет это положение и то, что технология SDH может легко интегрироваться с технологией DWDM, обеспечивающей передачу информации по оптическим магистралям с еще более вы-

сокими скоростями — сотни гигабит в секунду и выше — за счет мультиплексирования с разделением по длине волны. В магистральных сетях с ядром DWDM сети SDH будут играть роль сети доступа, то есть ту же роль, которую играют сети PDH по отношению к SDH.

У технологии SDH есть, естественно, и недостатки. Сегодня чаще всего говорят о ее неспособности динамически перераспределять пропускную способность между абонентами сети — свойстве, обеспечиваемом пакетными сетями. Значимость этого недостатка будет возрастать по мере увеличения доли и ценности трафика данных по отношению к стандартному голосовому трафику.

История возникновения

Технология синхронной цифровой иерархии первоначально была разработана компанией Bellcore под названием «Синхронные оптические сети» — Synchronous Optical NETs, SONET. Эта технология явилась развитием технологии PDH, которая появилась в 60-е годы для построения качественных и относительно недорогих цифровых каналов между телефонными станциями. PDH долгое время хорошоправлялась со своими обязанностями в качестве магистральной технологии, предоставляя пользователям каналы T1 (1,5 Мбит/с) и T3 (45 Мбит/с) в американском варианте или каналы E1 (2 Мбит/с), E3 (34 Мбит/с) и E4 (140 Мбит/с) в европейском и международном вариантах. Быстрое развитие телекоммуникационных технологий привело к необходимости расширения иерархии скоростей PDH и использовании всех возможностей, которые предоставляла новая среда — волоконно-оптические линии связи.

Одновременно с повышением верхней границы линейки скоростей нужно было освободиться от недостатков PDH, которые выявились за время эксплуатации этих сетей. Одним из основных недостатков PDH является, прежде всего, принципиальная невозможность выделения отдельного низкоскоростного потока из высокоскоростного без полного демультиплексирования последнего. Кроме того, в технологии PDH не были предусмотрены встроенные средства обеспечения отказоустойчивости и управления сетью.

Все эти недостатки были учтены и преодолены разработчиками технологии SONET. Первый вариант стандарта технологии SONET появился в 1984 году. Затем эта технология была стандартизована комитетом T1 ANSI. Международная стандартизация технологии проходила под эгидой Европейского института телекоммуникационных стандартов (ETSI) и CCITT, совместно с ANSI и ведущими телекоммуникационными компаниями Америки, Европы и Японии. Основной целью разработчиков международного стандарта было создание такой технологии, которая бы позволяла передавать трафик всех существующих цифровых каналов уровня PDH (как американских T1-T3, так и европейских E1-E4) в рамках высокоскоростной магистральной сети, использующей волоконно-оптические кабели, и обеспечила бы иерархию скоростей, продолжающую иерархию технологий PDH до скорости в несколько Гбит/с.

В результате длительной работы удалось разработать международный стандарт Synchronous Digital Hierarchy, SDH (спецификации ITU-T G.702, G.703, G.704,

G.707, G.708, G.709, G.773, G.774, G.782, G.783, G.784, G.957, G.958, Q.811, Q.812 и ETSI — ETS 300 147). Стандарты SONET также были доработаны так, что аппаратура и сети SDH и SONET являются совместимыми и могут мультиплексировать входные потоки практически любого стандарта PDH — и американского, и европейского.

Иерархия скоростей и методы мультиплексирования

Технология SONET/SDH поддерживает определенную иерархию скоростей, которую иллюстрирует табл. 6.2.

Таблица 6.2. Иерархия скоростей SONET/SDH

SOH	SONET	Скорость
	STS-1, OC-1	51,84 Мбит/с
STM-1	STS-3, OC-3	155,520 Мбит/с
STM-3	OC-9	466,560 Мбит/с
STM-4	OC-12	622,080 Мбит/с
STM-6	OC-18	933,120 Мбит/с
STM-8	OC-24	1,244 Гбит/с
STM-1 2	OC-3G	1,866 Гбит/с
STM-1 6	OC-48	2,488 Гбит/с
STM-G4	OC-192	9,953 Гбит/с
STM-256	OC-768	39,81 Гбит/с

В стандарте SDH все уровни скоростей (и, соответственно, форматы кадров для этих уровней) имеют общее название: STM-N (Synchronous Transport Module level N). В технологии SONET существует два обозначения для уровней скоростей: STS-N (Synchronous Transport Signal level N), употребляемое для случая передачи данных электрическим сигналом, и OC-N (Optical Carrier level N), употребляемое в случае передачи данных по волоконно-оптическому кабелю. Далее для упрощения изложения сосредоточимся на технологии SDH.

Кадры STM-N имеют достаточно сложную структуру, позволяющую агрегировать в общий магистральный поток потоки SDH и PDH различных скоростей, а также выполнять операции ввода-вывода без полного демультиплексирования магистрального потока.

Операции мультиплексирования и ввода-вывода выполняются с использованием виртуальных контейнеров (Virtual Container, VC), которые позволяют переносить через сеть SDH блоки данных PDH. Виртуальный контейнер содержит кроме блоков данных PDH некоторую служебную информацию, в частности заголовок пути контейнера (Path OverHead, POH), в котором переносится статистическая информация о процессе прохождении контейнера вдоль пути от его начальной до конечной точки (сообщения об ошибках), а также другие служебные данные, например индикатор установления соединения между конечными точками. В результате виртуальный контейнер имеет больший размер, чем соответствующий размер блока данных PDH, который он переносит. Например,

виртуальный контейнер VC-12, переносящий 32 байта данных потоки E1, состоит из 35 байт.

В технологии SDH (рис. 6.5) определено несколько типов виртуальных контейнеров, предназначенных для транспортировки основных типов блоков данных PDH: VC-11 (1,5 Мбит/с), VC-12 (2 Мбит/с), VC-2 (6 Мбит/с), VC-3 (34/45 Мбит/с) и VC-4(140 Мбит/с).

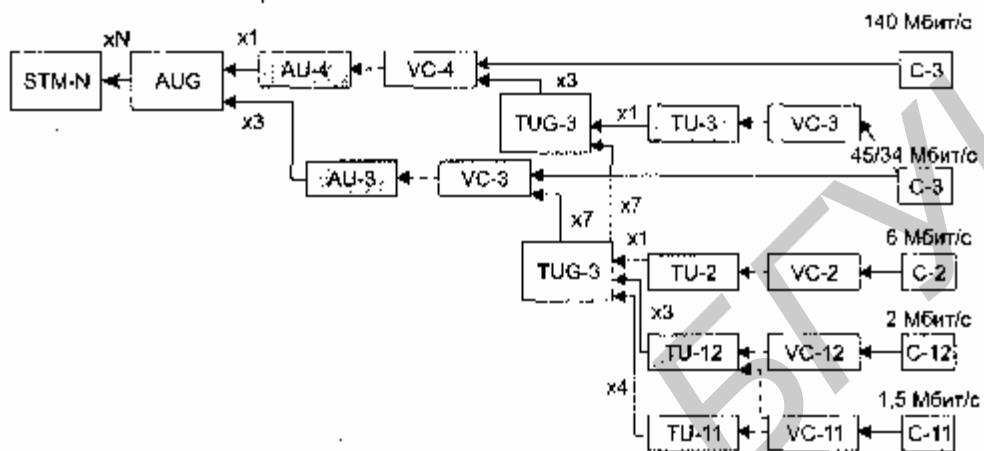


Рис. 6.5. Схема мультиплексирования данных в SDH

Виртуальные контейнеры являются единицей коммутации мультиплексоров SDH. В каждом мультиплексоре существует таблица соединений (называемая также таблицей кросс-соединений), в которой указано, например, что контейнер VC-12 порта P1 соединен с контейнером VC-12 порта P5, а контейнер VC-3 порта P8 соединен с контейнером VC-3 порта P9. Таблицу соединений формирует администратор сети с помощью системы управления или управляющего терминала в каждом мультиплексоре так, чтобы обеспечить сквозной путь для соединения конечных точек сети, к которым присоединено оборудование пользователей.

Для совмещения в рамках одной сети синхронной передачи кадров STM-N с асинхронным характером переносимых этими кадрами пользовательских данных PDH, в технологии SDH применяются указатели (pointers). Концепция указателей является ключевой в технологии SDH, она заменяет выравнивание скоростей асинхронных источников с помощью дополнительных битов, принятые в технологии PDH. Указатель определяет текущее положение виртуального контейнера в структуре более высокого уровня — трибутарном блоке (Tributary Unit, TU) или административном блоке (Administrative Unit, TU). Применение указателя позволяет виртуальному контейнеру «плавать» в определенных пределах внутри своего трибутарного или административного блока, который, в свою очередь, занимает уже фиксированное положение в кадре. Собственно, основное отличие этих блоков от виртуального контейнера — это добавленное поле указателя. Система указателей и позволяет мультиплексору находить положение пользовательских данных в синхронном потоке байтов кадров STM-N и «на лету» извлекать их оттуда, что не позволяет механизм мультиплексирова-

ния, принятый в PDH.

Трибутарные блоки объединяются в группы, которые, в свою очередь, сходят в административные блоки. Группа из N административных блоков (AUG) и образует полезную нагрузку кадра STM-N, помимо этого в кадр входит заголовок с общей для всех административных блоков служебной информацией. На каждом шаге преобразования к предыдущим данным добавляется несколько служебных байтов, помогающих распознать структуру группы блоков или блока и затем с помощью указателей определить начало пользовательских данных.

На рис. 6.5 структурные единицы кадра SDH, содержащие указатели, заштрихованы, а связь между контейнерами и блоками, допускающая сдвиг данных по фазе, показана пунктиром.

Схема мультиплексирования SDH предоставляет разнообразные возможности по объединению пользовательских потоков PDH. Например, для кадра STM-1 можно реализовать такие варианты:

- один поток E4;
- 63 потока E1;
- один поток E3 и 42 потока E1.

Другие варианты читатель может предложить сам.

Типы оборудования

Основным элементом сети SDH является мультиплексор (рис. 6.6), Мультиплексор обычно оснащен некоторым количеством портов PDH и SDH, например портами PDH 2 Мбит/с, 34 Мбит/с и портами SDH 155 Мбит/с STM-1 и 622 Мбит/с STM-4. Порты мультиплексора SDH делятся на агрегатные и трибутарные (трибы). Трибутарные порты часто называют также портами ввода-вывода, а агрегатные — линейными портами. Эта терминология отражает типовые топологии сетей SDH с ярко выраженной магистралью в виде цепи или кольца, через которую передаются потоки данных, поступающие от пользователей сети через порты ввода-вывода, то есть как бы «втекающие» в агрегированный поток (tributary — приток).

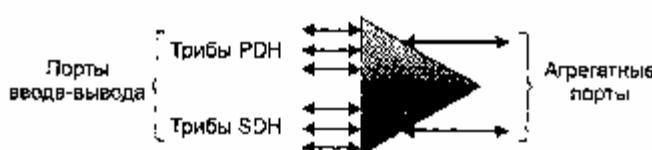


Рис. 6.6. Мультиплексор SDH

Мультиплексоры SDH обычно делят на терминальные мультиплексоры (Terminal Multiplexor, TM) и мультиплексоры ввода-вывода (Add-Drop Multiplexor, ADM). Разница между ними состоит не в составе портов, а в положении мультиплексора в сети SDH. Терминальный мультиплексор завершает агрегатные каналы, мультиплексируя в них большое количество каналов ввода-вывода (tributary). Мультиплексор ввода-вывода транзитом передает агрегатные

каналы, занимая промежуточное положение на магистрали (в кольце, цепи или смешанной топологии). При этом данные трибутарных каналов вводятся или выводятся в агрегатный канал. Агрегатные порты мультиплексора поддерживают максимальный для данной модели уровень скорости STM-N, который является характеристикой мультиплексора в целом, например, STM-4 или STM-64.

Иногда различают так называемые кросс-коннекторы (Digital Cross-Connect, DXC) — мультиплексоры, которые выполняют операции коммутации над произвольными виртуальными контейнерами, в отличие от мультиплексоров ввода-вывода, поддерживающих только специальные операции, сводящиеся к коммутации контейнера из агрегатного потока с соответствующим контейнером трибутарного потока. Чаще всего кросс-коннекторы реализуют соединения между трибутарными портами (точнее, виртуальными контейнерами, формируемыми из данных трибутарных портов), но могут применяться кросс-коннекторы и агрегатных портов, то есть контейнеров VC-1 и их групп. Последний вид мультиплексоров пока применяется реже, чем остальные, так как требует наличия большого количества агрегатных портов и ячеистой топологии сети, а это существенно удорожает как сам мультиплексор, так и сеть в целом. Большинство производителей выпускает универсальные мультиплексоры, которые могут использоваться как терминальные, ввода-вывода и кросс-коннекторы — в зависимости от набора установленных модулей с агрегатными и трибутарными портами. Ограничением может являться возможность установки нескольких агрегатных портов — производители часто выпускают модели мультиплексоров, позволяющих устанавливать только одну агрегатную карту с двумя портами — минимальная конфигурация, обеспечивающая работу в сети с топологией кольцо или цепь. Такое построение мультиплексора снижает его стоимость, но может усложнить проектирование сети, если требуется реализовать ячеистую топологию на максимальной для мультиплексора скорости.

Кроме мультиплексоров в состав сети SDH могут входить регенераторы, необходимые для преодоления ограничений по расстоянию между мультиплексорами, зависящих от мощности оптических передатчиков, чувствительности приемников и затухания волоконно-оптического кабеля. Регенератор преобразует оптический сигнал в электрический и обратно, при этом восстанавливается форма сигнала и его временные характеристики. В настоящее время регенераторы SDH применяются достаточно редко, так как стоимость их ненамного меньше стоимости мультиплексора, а функциональные возможности несопоставимы.

Стек протоколов SDH состоит из протоколов 4-х уровней.

- Физический уровень, названный в стандарте фотонным (photonic), имеет дело с кодированием битов информации путем модуляции света. Для кодирования оптического сигнала применяется потенциальный код NRZ.
- Уровень секции (section) поддерживает физическую целостность сети. Секцией в технологии SDH называется каждый непрерывный отрезок волоконно-оптического кабеля, который соединяет пару устройств SONET/SDH между собой, например мультиплексор и регенератор, регенератор и регенератор. Сек-

цию часто называют регенераторной секцией, имея в виду, что выполнение функций этого уровня не требует, чтобы окончные устройства секции были мультиплексорами. Протокол регенераторной секции имеет дело с определенной частью заголовка кадра, называемой заголовком регенераторной секции (RSOH), и на основе служебной информации может проводить тестирование секции и поддерживать операции административного контроля.

- Уровень линии (line) отвечает за передачу данных между двумя мультиплексорами сети. Протокол этого уровня работает с кадрами уровня STM-N для выполнения различных операций мультиплексирования и демультиплексирования, а также вставки и удаления пользовательских данных. Протокол линии также ответственен за проведения операций реконфигурирования линии в случае отказа какого-либо ее элемента — оптического волокна, порта или соседнего мультиплексора. Линию также часто называют мультиплексной секцией.
- Уровень тракта (path) отвечает за доставку данных между двумя конечными пользователями сети. Тракт (путь) — это составное виртуальное соединение между пользователями. Протокол тракта должен принять данные, поступающие в пользовательском формате, например формате T1, и преобразовать их в синхронные кадры STM-N.

На рис. 6.7 показано распределение протоколов SDH по типам оборудования SDH.

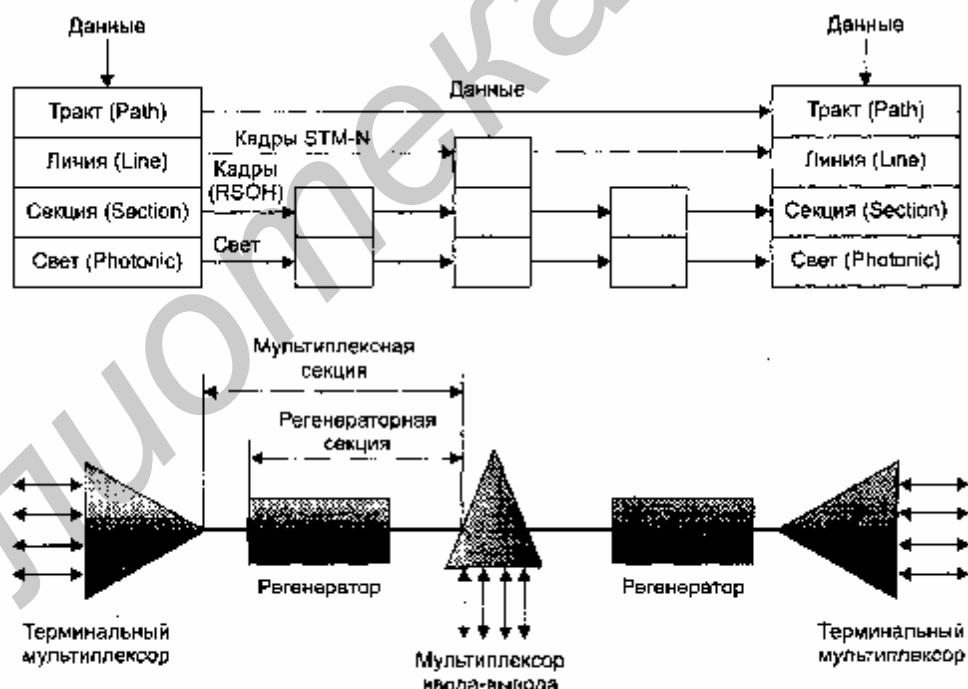


Рис. 6.7. Стек протоколов технологии SDH.

Кадры STM-N

На рис. 6.8 показаны основные элементы кадра STM-1. Кадр обычно представляют в виде матрицы, состоящей из 270 столбцов и 9 строк. Первые 9 байтов каждой строки отводятся под служебные данные заголовков, а из последующих

261 байта 260 отводятся под полезную нагрузку (данные таких структур, как AUG, AU, TUG, TU и VC), а один байт каждой строки отводится под заголовок тракта, что позволяет контролировать соединение «из конца в конец».

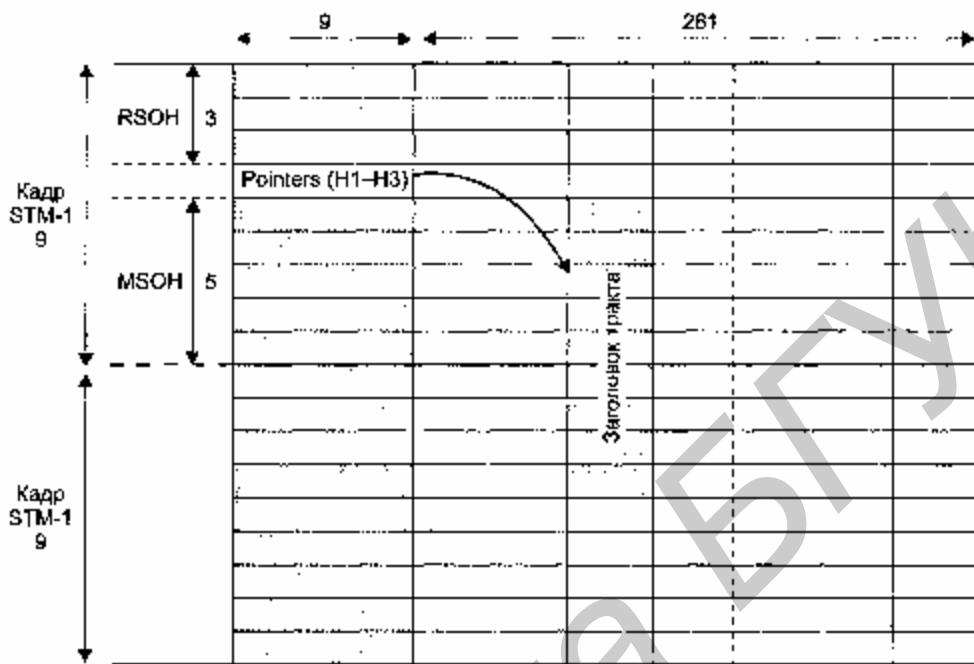


Рис. 6.8. Структура кадра STM-1

Заголовок регенераторной секции RSOH содержит:

- синхробайты;
- байты контроля ошибок для регенераторной секции;
- один байт служебного аудиоканала (64 кбит/с);
- три байта канала передачи данных (Data Communication Channel, DCC), работающего со скоростью 192 кбит/с;
- байты, зарезервированные для использования по усмотрению национальных операторов связи.

Указатели H1, H2, H3 задают положение начала виртуального контейнера VC-4 или трех виртуальных контейнеров VC-3 относительно поля указателей.

В заголовке протокола мультиплексной секции содержатся:

- байты контроля ошибок для мультиплексной секции;
- шесть байтов канала передачи данных (Data Communication Channel, DCC), работающего со скоростью 576 кбит/с;
- два банта протокола автоматической защиты трафика (байты K1 и K2), обеспечивающего живучесть сети;
- байт передачи сообщений статуса системы синхронизации.

Остальные байты заголовка MSOH либо зарезервированы для использования национальными операторами связи, либо не используются.

Рассмотрим механизм работы указатели H1, H2, H3 на примере кадра STM-1, переносящего контейнер VC-4. Указатель занимает 9 байтов четвертого ряда кадра, причем под каждое из полей H1, H2 и H3 в этом случае отводится по три

байта. Разрешенные значения указателя находятся в диапазоне 0-782, причем указатель отмечает начало контейнера VC-4 в трехбайтовых единицах. Например, если указатель имеет значение 27, то первый байт VC-4 находится на расстоянии $27 \times 3 = 81$ байт от последнего байта поля указателей, то есть является 90-м байтом (нумерация начинается с единицы) в 4-й строке кадра STM-1. Фиксированное значение указателя позволяет учесть фазовый сдвиг между источником данных, в качестве которого может выступать мультиплексор PDH, оборудование пользователя с интерфейсом PDH или другой мультиплексор SDH, и данным мультиплексором. В результате виртуальный контейнер передается в двух последовательных кадрах STM-1, как и показано на рис. 6.8.

Указатель может отрабатывать не только фиксированный фазовый сдвиг, но и рассогласование тактовой частоты мультиплексора с тактовой частотой устройства, от которого принимаются пользовательские данные. Для компенсации этого эффекта значение указателя периодически наращивается или уменьшается на единицу.

Если скорость поступления данных контейнера VC-4 меньше, чем скорость отправки STM-1, то у мультиплексора периодически (этот период зависит от величины рассогласования частоты синхронизации) возникает нехватка пользовательских данных для заполнения соответствующих полей виртуального контейнера. Поэтому мультиплексор вставляет три «холостых» (незначащих) байта в данные виртуального контейнера» после чего продолжает заполнение VC-4 «подоспевшими» за время паузы пользовательскими данными. Указатель наращивается на единицу, что отражает запаздывание начала очередного контейнера VC-4 на три байта. Эта операция над указателем называется положительным выравниванием. В итоге средняя скорость отправляемых пользовательских данных становится равной скорости их поступления, причем без вставки дополнительных битов в стиле PDH.

Если же скорость поступления данных VC-4 выше, чем скорость отправки кадра STM-1, то у мультиплексора периодически возникает потребность вставки в кадр «лишних», то есть преждевременно пришедших байтов, для которых в поле VC-4 нет места. Для их размещения используются три младших байта указателя, то есть поле H3 (само значение указателя умещается в байты полей H1 и H2). Указатель при этом уменьшается на единицу, поэтому такая операция носит название отрицательного выравнивания.

То, что выравнивание контейнера VC-4 происходит с дискретностью в три байта, объясняется достаточно просто. Дело в том, что в кадре STM-1 может переноситься либо один контейнер VC-4, либо три контейнера VC-3. Каждый из контейнеров VC-3 имеет в общем случае независимое значение фазы относительно начала кадра, а также собственную величину рассогласования частоты. Указатель VC-3 в отличие от указателя VC-1 состоит уже не из девяти, а из трех байтов: H1, H2, H3 (каждое из этих полей — однобайтовое). Эти три указателя помещаются в те же байты, что и указатель VC-4, но по схеме с чередованием байтов (byte interleaving), то есть в порядке H1-1, H1-2, H1-3, H2-1, H2-2, H2-3, H3-1, H3-2, H3-3 (второй индекс идентифицирует определенный контейнер VC-3). Значения указателей VC-3 интерпретируются в байтах, а не трех-

байтовых единицах. При отрицательном выравнивании контейнера VC-3 лишний байт помещается в соответствующий байт H3-1, H3-2 или H3-3 — в зависимости от того, над каким из контейнеров VC-3 проводится операция.

Таким образом, размер смещения для контейнеров VC-4 выбран в расчете на унификацию описанных операций над контейнерами любого типа, размещаемыми непосредственно в AUG кадра STM-1. Выравнивание контейнеров более низкого уровня всегда происходит с шагом в один байт.

При объединении блоков TU и AU в группы в соответствии с описанной схемой (см. рис. 6.8) выполняется их последовательное побайтное расслоение, так что период следования пользовательских данных в кадре STM-N совпадает с периодом их следования в трибутиарных портах. Это исключает необходимость в их временной буферизации — поэтому говорят, что мультиплексоры SDH передают данные в реальном масштабе времени.

Типовые топологии

В сетях SDH применяются различные топологии связей. Наиболее часто используются кольца и линейные цепи мультиплексоров, также находит все большее применение ячеистая топология, близкая к полносвязной.

Кольцо SDH строится из мультиплексоров ввода-вывода, имеющих, по крайней мере, по два агрегатных порта (рис. 6.9, а). Пользовательские потоки вводятся в кольцо и выводятся из кольца через трибутиарные порты, образуя соединения «точка-точка» (на рисунке показаны в качестве примера два таких соединения). Кольцо является классической регулярной топологией, обладающей потенциальной отказоустойчивостью — при однократном обрыве кабеля или выходе из строя мультиплексора соединение сохранится, если его направить по кольцу в противоположном направлении. Кольцо обычно строится на основе кабеля с двумя оптическими волокнами, но иногда для повышения надежности и пропускной способности применяют четыре волокна.

Цепь (рис. 6.9, б) — это линейная последовательность мультиплексоров, из которых два оконечных играют роль терминальных, а остальные — мультиплексоров ввода-вывода. Обычно сеть с топологией цепи применяется в тех случаях, когда узлы имеют соответствующее географическое расположение, например вдоль магистрали железной дороги или трубопровода. Правда, в таких случаях может применяться и плоское кольцо (рис. 6.9, в), обеспечивающее более высокий уровень отказоустойчивости за счет использования двух дополнительных волокон в магистральном кабеле и по одному дополнительному агрегатному порту у терминальных мультиплексоров.

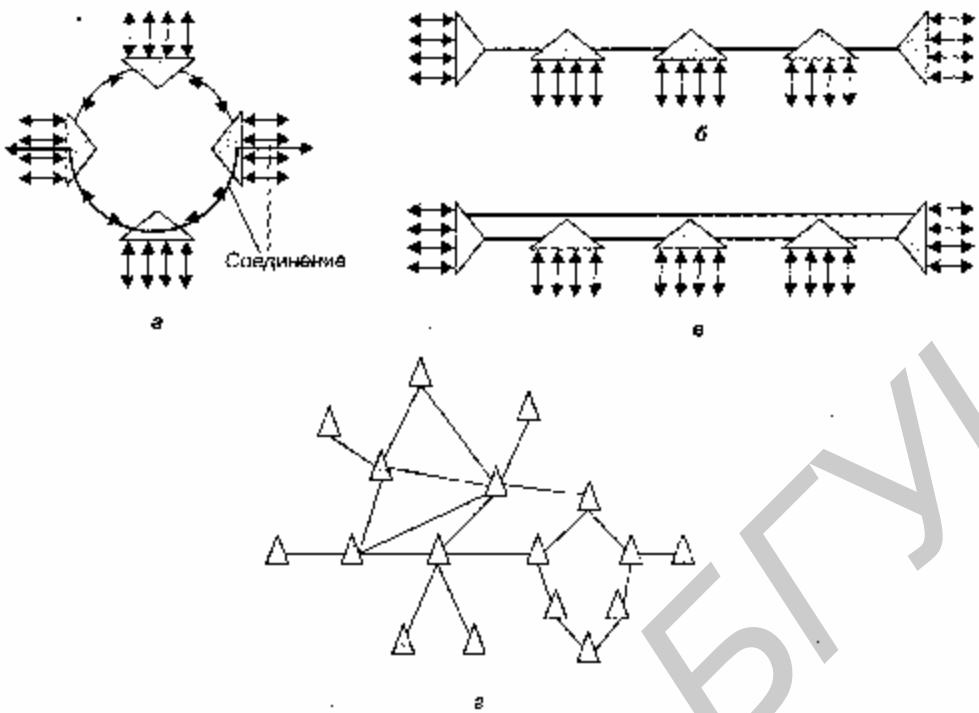


Рис. 6.9. Типовые топологии

Эти базовые топологии могут комбинироваться при построении сложной и разветвленной сети SDH, образуя участки с радиально-кольцевой топологией, соединениями «кольцо-кольцо» и т. п. Наиболее общим случаем является ячеистая топология сети (рис. 6.9, г), при которой мультиплексоры соединяются друг с другом большим количеством связей, за счет чего сеть может достичь очень высокой степени производительности и надежности.

Методы обеспечения живучести сети

Одной из сильных сторон первичных сетей SDH является разнообразный набор средств отказоустойчивости, который позволяет сети быстро (за десятки миллисекунд) восстанавливать работоспособность в случае отказа какого-либо элемента сети — канала связи, порта или карты мультиплексора, мультиплексора в целом.

В SDH в качестве общего названия механизмов отказоустойчивости используется термин автоматическое защитное переключение (Automatic Protection Switching, APS), отражающий факт перехода (переключения) на резервный путь или резервный элемент мультиплексора при отказе основного. Сети, поддерживающие такой механизм, в стандартах SDH названы «самовосстанавливающимися» (точнее, «самозалечивающимися» — selfhealing).

В оборудовании и сетях SDH применяются следующие основные виды автоматической защиты, отличающиеся типом защищаемого (путем резервирования) элемента сети:

- Equipment Protection Switching, EPS — защита блоков и элементов оборудо-

вания SDH;

- Card Protection, CP — защита агрегатных и трибутарных карт мультиплексора;
- Multiplex Section Protection, MSP — защита мультиплексной секции, то есть участка сети между двумя смежными мультиплексорами SDH;
- Sub-Network Connection Protection, SNC-P — защита пути (соединения) через сеть для определенного виртуального контейнера;
- Multiplex Section Shared Protection Ring, MS-SPRing — разделяемая между пользовательскими соединениями защита путей в кольцевой топологии.

В сетях SDH применяются схемы защиты 1 + 1, 1:1 и 1:N.

Защита 1 + 1 означает, что резервный элемент выполняет ту же работу, что и основной. Например, при защите трибутарной карты по схеме 1+1 трафик проходит как через рабочую карту (резервируемую), так и через защитную (резервную). Защита 1:1 подразумевает, что защитный элемент в нормальном режиме не выполняет функции защищаемого элемента, а переключается на них только в случае отказа. Защита 1:N предусматривает выделение одного защитного элемента на N защищаемых. При отказе одного из защищаемых элементов его функции начинает выполнять защитный, при этом остальные элементы остаются без защиты — до тех пор, пока отказавший элемент не будет заменен.

Защита блоков (EPS) применяется для таких жизненно важных элементов мультиплексора, как процессорный блок, блок коммутации (кросс-коннекта), блок питания, блок ввода сигналов синхронизации и т. п. EPS обычно работает по схеме 1 + 1 или 1:1.

Защита карт (CP) позволяет мультиплексору автоматически продолжать работу в случае отказа одной из агрегатных или трибутарных карт. Используется защита по схемам 1+1, 1:1 и 1:N. Защита 1 + 1 обеспечивает непрерывность транспортного сервиса, так как трафик пользовательских соединений не прерывается при отказе карты.

В приведенном на рис. 6.10 примере в мультиплексоре поддерживается защита трибутарных двухпортовых карт по схеме 1 + 1. Одна из трибутарных карт является основной (или рабочей — working), а другая — защитной (protection). Режим работы пары связанных таким образом карт задается командой конфигурирования мультиплексора. В режиме, когда обе трибутарные карты являются работоспособными, трафик обрабатывается параллельно каждой картой.

Для переключения трафика между трибутарными картами используется дополнительная карта-переключатель. Входящий трафик (add) каждого порта поступает на входной мост карты-переключателя, который разветвляет трафик и передает его на входы соответствующих портов трибутарных карт. Агрегатная карта получает оба сигнала STM-N от трибутарных карт и выбирает сигнал только от активной в данный момент карты. Выходящий трафик (drop) от агрегатной карты также обрабатывается обеими трибутарными картами, но карта-переключатель передаст на выход только трафик от активной карты.

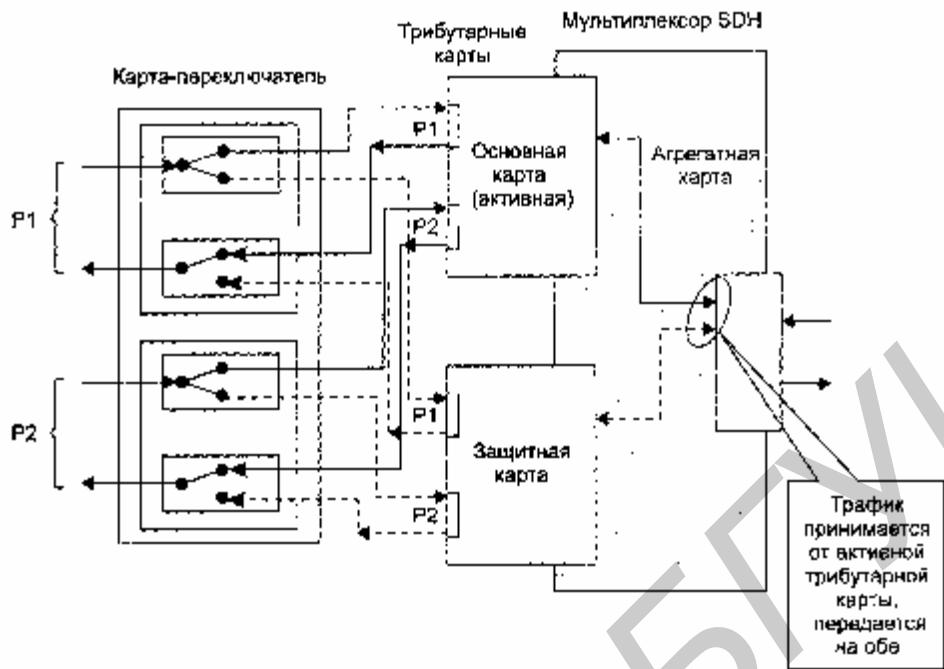


Рис. 6.10. Защита карт по схеме 1+1

При отказе основной карты (или другом событии, требующем перехода на защитную карту — деградации сигнала, ошибке сигнала, удалении карты) агрегатная карта по команде от блока управления мультиплексором переходит на прием сигнала от защитной трибултарной карты. Одновременно карта-переключатель также начинает передавать на выход сигналы выходящего трафика от защитной карты.

Данный способ обеспечивает автоматическую защиту всех соединений, проходящих через защищаемую карту. При установлении СР-защиты конфигурация соединений рабочей карты дублируется для защитной карты.

Защита мультиплексной секции (MSP) действует более избирательно по сравнению с защитой карт. Защищается секция между двумя мультиплексорами, включающая два порта и линию связи (возможно, в свою очередь, включающую регенераторы, но не мультиплексоры). Обычно применяется схема защиты 1 + 1. При этом для рабочего канала (верхняя пара соединенных кабелем портов на рис. 6.11, а) конфигурируется защитный канал (нижняя пара портов). При установлении защиты MSP в каждом мультиплексоре необходимо выполнить конфигурирование, указав связь между рабочим и защитным портами. В исходном состоянии весь трафик передается как по рабочему, так и по защитному каналам.

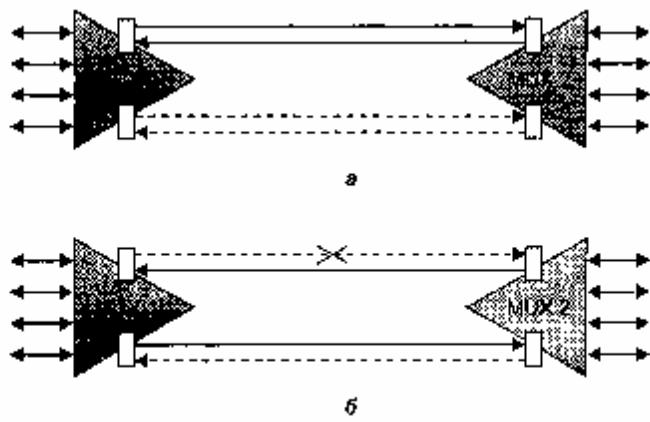


Рис. 6.11. Защита MSP

Существует однонаправленная и двунаправленная защита MSP. При однонаправленной защите (именно этот случай показан на рисунке) решение о переключении принимает только один из мультиплексоров — тот, который является приемным для отказавшего канала. Этот мультиплексор после обнаружения отказа (отказ порта, ошибка сигнала, деградация сигнала и т. п.) переходит на прием по защитному каналу. При этом передача и прием ведутся по разным портам (рис. 6.11,б).

При двунаправленной защите MSP при отказе рабочего канала в каком-либо направлении выполняется полное переключение на защитные порты мультиплексоров. Для уведомления передающего (по рабочему каналу) мультиплексора о необходимости переключения принимающий мультиплексор использует протокол, называемый протоколом «К-байт». Этот протокол указывает в двух байтах заголовка кадра STM-N статус рабочего и защитного каналов, а также детализирует информацию об отказе.

Механизм MSP обеспечивает защиту всех соединений, проходящих через защищаемую мультиплексную секцию.

Время переключения защиты MSP, согласно требованиям стандарта, не должно превышать 50 мс.

Защита соединения (SNC-P) обеспечивает переключение определенного пользовательского соединения на альтернативный путь при отказе основного пути. Объектом защиты SNC-P является трибутарный трафик, помещенный в виртуальный контейнер определенного типа (например, в VC12, VC-3 или VC-4). Используется схема защиты 1 + 1.

Защита SNC-P конфигурируется в двух мультиплексорах — входном, в котором трибутарный трафик, помещенный в виртуальный контейнер, разветвляется, а также выходном, в котором сходятся два альтернативных пути трафика. Пример защиты SNC-P показан на рис. 6.12. В мультиплексоре ADM1 для виртуального контейнера VC-4 трибутарного порта T2 заданы два соединения: с одним из четырех контейнеров VC-4 агрегатного порта A1 и с одним из четырех контейнеров VC-4 агрегатного порта A2. Одно из соединений конфигурируется как рабочее, а второе — как защитное, при этом трафик передается по обоим соединениям. Промежуточные (для данных соединений) мультиплексо-

ры конфигурируются обычным образом. В выходном мультиплексоре контейнер VC-4 трибутарного порта Т3 также соединяется с контейнерами - агрегатного порта А1 и агрегатного порта А2. Из двух поступающих на порт Т3 потоков выбирается тот, качество которого выше (при равном нормальном качестве выбирается сигнал из агрегатного порта, выбранном при конфигурировании в качестве рабочего).

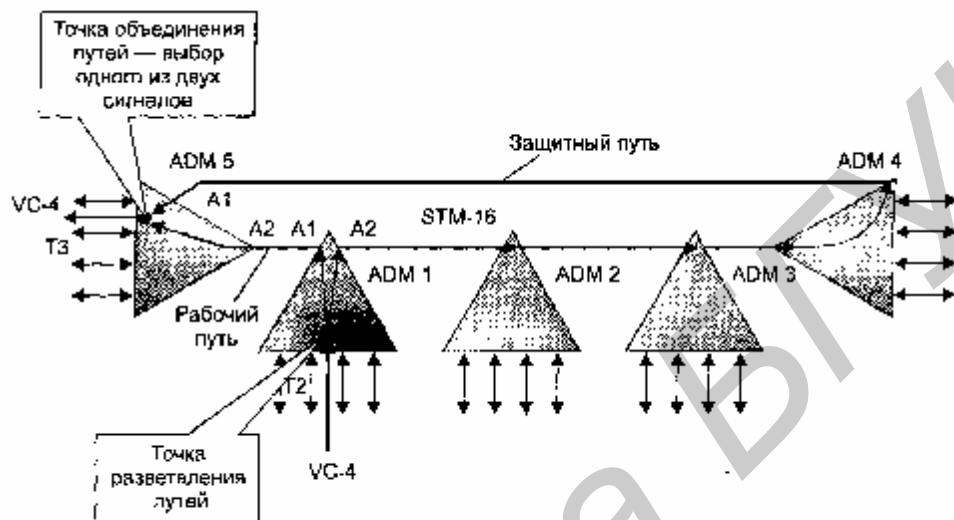


Рис. 6.12. Защита SNC-P

Защита SNC-P работает в любых топологиях сетей SDH, в которых имеются альтернативные пути следования трафика, то есть кольцевых и ячеистых.

Защита с разделением кольца (MS-SPRing) обеспечивает в некоторых случаях более экономичную защиту трафика в кольце. Хотя защита SNC-P вполне подходит для кольцевой топологии сети SDH, в некоторых случаях ее применение уменьшает полезную пропускную способность кольца, так как каждое соединение потребляет удвоенную полосу пропускания вдоль всего кольца. Так, в кольце STM-16 можно установить только 16 защищенных по SNC-P соединений VC-4 (рис. 6.13).

Защита MS-SPRing позволяет использовать пропускную способность кольца более эффективно, так как полоса пропускания не резервируется заранее для каждого соединения. Вместо этого резервируется половина пропускной способности кольца, но эта резервная полоса выделяется для соединений динамически, по мере необходимости, то есть после обнаружения факта отказа линии или мультиплексора. Степень экономии полосы при применении защиты MS-SPRing зависит от распределения трафика.

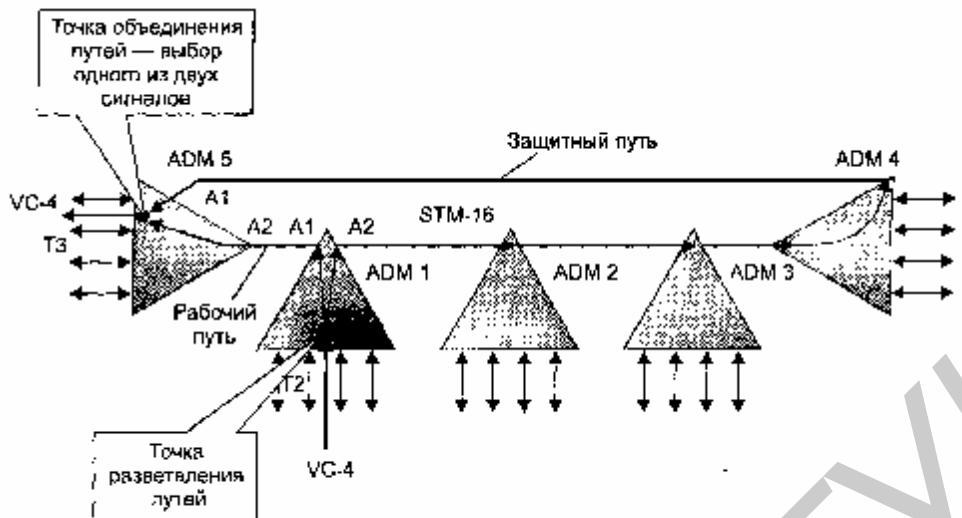


Рис. 6.12. Защита SNC-P

Если весь трафик сходится в один мультиплексор, то есть имеет распределение «звезды», защита MS-SPRing экономии по сравнению с SNC-P вообще не дает. Пример такой ситуации представлен на рис. 6.14, а, где центром тяготения трафика является мультиплексор А, а в кольце установлено те же 16 защищенных соединений, что и в примере защиты SNC-P на рис. 6.13. Для защиты соединений резервируется 8 из 16 виртуальных контейнеров агрегатного потока STM-16.

При возникновении неисправности, например обрыве линии, как это показано на рис. 6.14, б, трафик в мультиплексорах, между которыми нарушилась связь, «разворачивается» в обратном направлении. Для этого используются резервные виртуальные контейнеры агрегатных портов, с которыми соединяются виртуальные контейнеры пострадавших соединений. В то же время соединения, на которые отказ не повлиял, работают в прежнем режиме, не подключая резервные контейнеры. Для уведомления мультиплексоров о реконфигурации кольца используется протокол К-байт. Время переключения на защитные соединения MS-SPRing составляет около 50 мс. При смешанном распределении трафика экономия полосы в кольце MS-SPRing может быть еще более значительной.

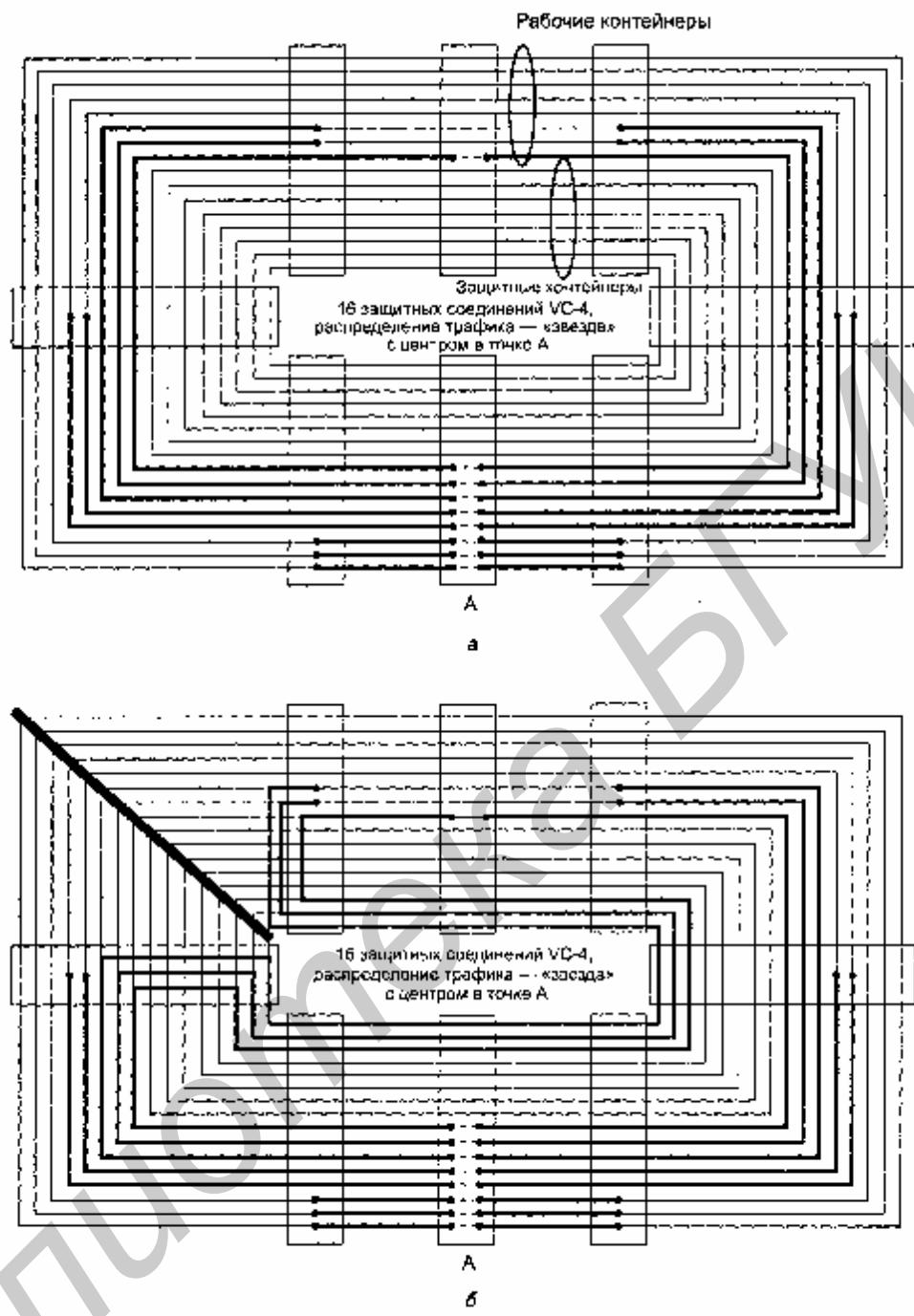


Рис. 6.14. MS-SPRing — защита с разделением кольца

Синхронизация

Устойчивая работа сети SDH во многом зависит от качества синхронизации между ее узлами. В сети SDH применяется иерархический метод принудительной синхронизации с парами таймеров «ведущий—ведомый». Этот метод применяется не только в сетях SDH, но и практически во всех крупных TDM-сетях: первичных на основе технологии PDH и телефонных.

Очевидно, что отказоустойчивость сетей SDH должна распространяться и на

отказы источника синхронизации. Для обеспечения этого свойства мультиплексор SDH может использовать несколько дублирующих источников синхронизации.

- Сигнал внешнего сетевого таймера с частотой 2048 кГц, называемого также первичным эталонным генератором, ПЭГ (Primary Reference Clock, PRC), в соответствии с рекомендациями G.811. Его точность должна быть не хуже 1×10^{-11} . Первичный эталонный таймер обычно представляет собой хронирующий атомный источник тактовых импульсов (цезиевые или рубидиевые часы). Его калибруют вручную или автоматически по сигналам мирового скоординированного времени UTC (Universal Time Coordinated).
- Сигнал со вторичного задающего генератора (БЗГ), который обычно является ведомым по отношению к ПЭГ, но может являться и самостоятельным источником.
- Сигнал со спутника глобальной системы позиционирования (Global Positioning System, GPS).
- Сигнал внутреннего таймера узла SDH. Точность сигналов внутреннего таймера обычно невелика, порядка $(1...5) \times 10^{-6}$.
- Сигнал 2048 кГц, выделяемый из линейного (или трибутарного) сигнала STM-N. Обычно точность такого источника синхронизации составляет 5×10^{-8} .
- Сигнал с пользовательского (трибутарного) интерфейса PDH.

Так как сигналы трибутарных потоков 2 Мбит/с «плавают» внутри виртуальных контейнеров VC-12, то их использование в качестве источников синхронизации в сетях SDH нецелесообразно. Низкая точность внутреннего таймера мультиплексора также не позволяет добиться хорошей синхронизации передающего и принимающего узлов SDH. Поэтому основными источниками надежной и точной синхронизации являются сигналы первичного эталонного генератора и сигналы, выделяемые из кадров STM-N.

В качестве внешних источников здесь выступают как внешние таймеры, подключаемые к специальным синхронизирующими входам мультиплексора, так и сигналы STM-N линейного входа (и трибутарных входов, если они поддерживают какой-либо уровень STM, а не PDH). Одна схема мультиплексора осуществляет выбор источника синхронизации для внутренних элементов мультиплексора, а другая — для внешних.

Для синхронных сетей общего пользования характерна иерархия задающих генераторов, в которой уровень сигнала каждого генератора синхронизируется по эталону сигнала более высокого уровня, имеющего более высокую точность.

Иерархия синхронизирующих источников — это сеть, состоящая из нескольких слоев генераторов, называемых также стратум-таймерами (от stratum — слой). Сеть синхронизации содержит один генератор слоя Stratum 1 и несколько генераторов более низких слоев, от Stratum 2 до Stratum 4 (рис. 6.15).

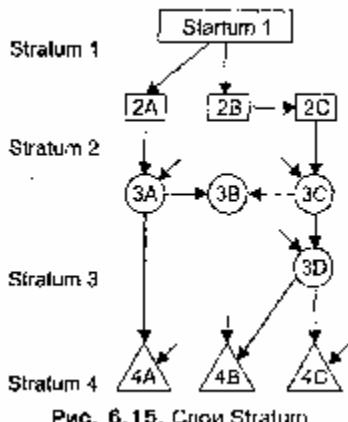


Рис. 6.15. Слои Stratum

Генератор Stratum 1 посылает эталонные сигналы тактовой частоты нескольким генераторам слоя Stratum 2. В свою очередь, эти генераторы посылают сигналы другим генераторам слоев Stratum 2 и Stratum 3. Аналогично, генераторы слоя Stratum 3 синхронизируют другие элементы слоев Stratum 3 и Stratum 4. Генератор, фаза которого подстраивается по входному сигналу, полученному от генератора более высокого или того же уровня качества, называется ведомым задающим генератором, или ВЗГ (Secondary Reference Clock, SRC). Ведомый задающий генератор высшего качества занимает вторую ступень в иерархии слоев синхронизации, то есть соответствует слою Stratum 2 и устанавливается обычно в транзитных узлах сети. Ведомые генераторы третьего уровня качества Stratum 3 устанавливаются обычно в локальных (терминальных) узлах сети. Генераторы каждого слоя должны удовлетворять стандартным требованиям к точности частоты. Эти требования приведены в табл. 6.3.

Таблица 6.3. Требования к точности частоты генераторов слоев

Strat-	Минимальная	Минимальная стабильность удержа-
1	$\pm 1,0 \times 10^{-11}$	Не применимо
2	$\pm 1,6 \times 10^{-8}$	Дрейф — не более $1,0 \times 10^{-10}$ в сутки
3	$\pm 4,6 \times 10^{-6}$	$\pm 0,37 \times 10^{-6}$ в течение первых 24 часов
4	$\pm 20 \times 10^{-6}$	Не требуется

При потере сигнала синхронизации от генератора более высокого уровня генераторы слоев Stratum 2 и Stratum 3 должны перейти в режим удержания частоты (режим holdover), в котором от них требуется автономно обеспечивать синхросигналы с определенной в таблице точностью в течение, по крайней мере, первых 24 часов после потери связи с эталонным источником более высокого уровня.

Для надежной работы сети у каждого мультиплексора SDH должно быть несколько альтернативных источников синхронизации, но использоваться в каждый момент времени должен только один, наиболее точный. Для выбора такого источника применяют приоритетные (называемые также иерархическими) списки, задаваемые администратором, а также механизм сообщений о статусе син-

хронизации (Synchronization Status Messaging, SSM). Сообщения SSM переносятся по сети в заголовках кадров STM-N, в них указывается уровень качества синхронизации (Quality Level, QL) данного сигнала. Переменная QL может принимать 16 значений, от 0 до 15, причем чем меньше значение QL, тем уровень качества выше (за исключением значения 0, которое обычно интерпретируется мультиплексорами как эквивалент 15). Для кодирования уровня QL чаще всего используются четыре значения кода, соответствующие четырем уровням точности синхросигнала, то есть уровням Stratum I ($QL = 2$), Stratum 2 ($QL = 4$), Stratum 3 ($QL = 8$) и Stratum 4 ($QL = 11$). Значение $QL=15$ означает непригодность для синхронизации, обычно оно используется для того, чтобы ведущий мультиплексор не действовал ведомый в качестве источника синхронизации.

Администратор может задействовать и другие значение кодов, если считает это нужным. В режиме по умолчанию сообщение SSM, поступающее в заголовке кадра STM-N, принимается мультиплексором и используется при выборе источника синхронизации, а далее в неизменном виде передается в составе кадра следующему мультиплексору. У администратора имеется возможность переопределить (override) значение QL в поступившем кадре, так что отправленный следующему мультиплексору кадр будет иметь новое значение QL. Механизм переопределения значений QL применяется также для внешних источников синхронизации, которые не могут поместить сообщение SSM в кадр.

Если у нескольких источников, имеющихся в иерархическом списке, соотношение значений QL противоречат указанным в этом списке приоритетам источников, то предпочтение отдается источнику с лучшим (меньшим) значением QL (а не источнику с более высоким положением в списке).

На рис. 6.16 приведен пример кольца SDH, в котором используются механизмы SSM и приоритетного списка источников. К кольцу подключено два первичных эталонных генератора — PRC1 и PRC2, при этом за счет соответствующего конфигурирования все мультиплексоры кольца синхронизируются от PRC1 либо непосредственно, либо косвенно, а генератор PRC2 является резервным.

Непосредственно от PRC1 синхронизируется мультиплексор M1 — через порт Ext1 внешней синхронизации, так как он занимает верхнюю строчку в иерархическом списке приоритетов источников для этого мультиплексора. Для источника Ext1 задан режим переопределения QL значением 2, отражающим тот факт, что источник имеет уровень точности Stratum 1. Мультиплексор M1 указывает уровень $QL = 2$ в кадрах, которые он передает мультиплексорам M2 и M4.

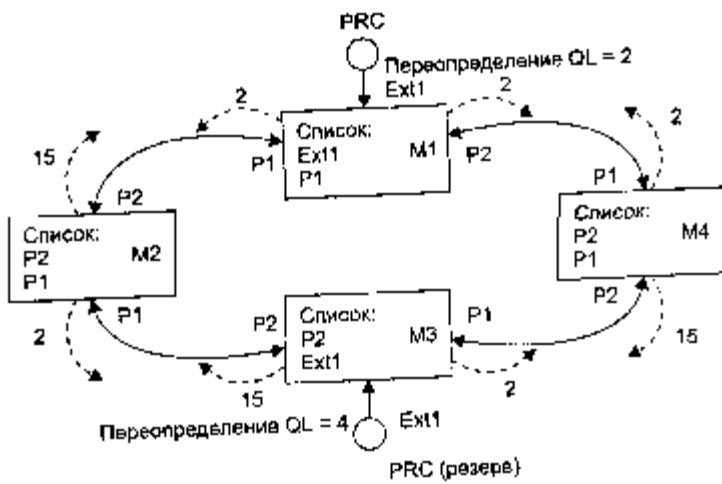


Рис. 8.18. Независимое источники синхронизации

Мультиплексор M2 синхронизируется от потока STM-N, получаемого по порту P2 (имеющего высший приоритет и список) от мультиплексора M1, то есть косвенно — от PRC1. В обратном направлении, то есть в направлении к M1, кадры передаются со значением QL = 15. Мультиплексор M3 выбирает источником синхронизации порт P2 — он и в списке стоит первым, и качество сигнала от него (QL = 2) выше, чем у внешнего порта Ext1 (QL = 4). И наконец, мультиплексор M4 выбирает для синхронизации сигналы с порта P2, потому что этот порт при равенстве качества сигналов с портом P1 стоит выше в иерархическом списке.

При отказе генератора PRC1 мультиплексор переходит в режим удержания частоты внутренним генератором. Точность такого генератора соответствует уровню точности Stratum 4, поэтому в кадрах STM-N мультиплексор M1 указывает значение QL = 11. Кадры с этим значением распространяются по сети, но на мультиплексоре M3 этот процесс заканчивается, так как у него в списке есть источник Ext1 с более высоким качеством QL = 4. Поэтому источником синхронизации для кольца становится генератор PRC2, при этом мультиплексоры M2 и M1 синхронизируются потоками STM-N, идущими по часовой стрелке, а мультиплексор M4 — против часовой стрелки.

При распространении сигналов синхронизации соблюдается определенная иерархия: от сигналов PRC синхронизируется магистральная сеть, от магистральной — внутризоновые, а от магистральной и внутризоновых — местные сети.

Максимальное число промежуточных мультиплексоров SDH, через которые в потоке STM-N передаются синхросигналы от первичного генератора PRC, определено в стандартах G.803 — это последовательность из 20 мультиплексоров. При большем числе промежуточных мультиплексоров нужно использовать внешний ведомый задающий генератор, который будет синхронизироваться от первичного и выполнять роль источника синхронизации для остальной части сети. В примере на рисунке показано несколько ВЗГ, которые синхронизируются от сигналов слоя Stratum 1. Ведомых генераторов, последовательно синхронизирующихся друг от друга, не должно быть более 10.

В России операторы связи используют в качестве источника синхронизации для

своих сетей первичные эталонные генераторы, входящие в систему тактовой сетевой синхронизации (ТСС) ОАО «Ростелеком». Ведомственные и корпоративные сети связи могут пользоваться как ПЭГ «Ростелеком», так и устанавливать свои собственные ПЭГ (в случае специфических особенностей сети).

Выводы

- Разработчики сетей SDH (первоначальное название — SONET, которое осталось за американским вариантом технологии) преследовали несколько целей:
 - расширение иерархии скоростей до мультимегабитных и гигабитных;
 - повышение гибкости механизма мультиплексирования, возможность непосредственной вставки и выделения низкоскоростных потоков в высокоскоростной поток, обеспечение асинхронности низкоскоростного потока относительно высокоскоростного;
 - встраивание механизмов отказоустойчивости в базовую технологию сети.
- Асинхронность вставки абонентских потоков в кадр SDH обеспечивается благодаря концепции виртуальных контейнеров и системы плавающих указателей, отмечающих начало пользовательских данных в виртуальном контейнере.
- Мультиплексоры SDH могут работать в сетях с различной топологией: цепи, кольца и ячеистой топологией. Различают несколько специальных типов мультиплексоров, которые занимают специфическое место в сети: терминальные мультиплексоры, мультиплексоры ввода-вывода и кросс-коннекторы.
- В сетях SDH поддерживается большое количество механизмов отказоустойчивости, которые защищают трафик данных на уровне отдельных блоков, портов или соединений: EPS, CP, MSP, SNC-P и MS-SPKing. Наиболее эффективная схема защиты выбирается в зависимости от логической топологии соединений в сети.

Сети DWDM

Технология плотного волнового (спектрального) мультиплексирования (Dense Wave Division Multiplexing, DWDM) предназначена для создания оптических магистралей нового поколения, работающих на мультигигабитных и терабитных скоростях. Такой качественный скачок производительности обеспечивает принципиально иной, нежели у SDH, метод мультиплексирования — информация в оптическом волокне передается одновременно большим количеством световых волн (лямбд — от традиционного для физики обозначения длины волны). Сети DWDM работают по принципу коммутации каналов, при этом каждая световая волна представляет собой отдельный спектральный канал. Каждая волна несет собственную информацию, при этом оборудование DWDM не занимается непосредственно проблемами передачи данных по каждой волне, то есть способом кодирования информации и протоколом ее передачи. Устройства DWDM занимаются только объединением различных волн в одном световом пучке, а также выделением из общего сигнала информации каждого спектрального ка-

нала.

Принципы работы

Сегодня оборудование DWDM позволяет передавать по одному оптическому волокну 32 и более волн разной длины и окне прозрачности 1550 нм, при этом каждая волна может переносить информацию со скоростью до 10 Гбит/с (при применении протоколов STM-64 или ЮСЕ для передачи информации на каждой волне). В настоящее время ведутся работы по повышению скорости передачи информации на одной длине волны до 40-80 Гбит/с.

У технологии DWDM имеется предшественница — технология WDM, которая использует четыре спектральных канала в окнах прозрачности 1310 нм и 1550 нм, с разносом несущих в 800-400 ГГц (стандартной классификации WDM не существует, встречаются системы WDM и с другими характеристиками).

Мультиплексирование DWDM называется «плотным» из-за того, что в нем используется существенно меньшее расстояние между длинами волн, чем в WDM. На сегодня рекомендацией ITU-T G.692 определены: частотный план с разнесением частот между соседними каналами 100 ГГц (А. * 0,8 нм), в соответствии с которым для передачи данных применяется 41 волна в диапазоне от 1528,77 нм (196,1 ТГц) до 1560,61 нм (192,1 ТГц), и частотный план с шагом в 50 ГГц (ДА. » 0,4 нм), позволяющий передавать в этом же диапазоне 81 длину волны. Некоторыми компаниями выпускается также оборудование (на сегодня — это чаще всего экспериментальные образцы, а не серийная продукция), способное работать с частотной сеткой с шагом 25 ГГц (называемое High-Dense WDM, HDWDM).

Реализация частотных планов с шагом 50 ГГц и 25 ГГц предъявляет гораздо более жесткие требования к оборудованию DWDM, особенно в том случае, если каждая волна переносит сигналы со скоростью модуляции 10 Гбит/с и выше (STM-64, 10GE или STM-256). Необходимо подчеркнуть, что сама технология DWDM (как и WDM) не занимается непосредственно кодированием переносимой по каждой волне информации — это проблема более низкоуровневой технологии, которая пользуется предоставленной ей волной по своему усмотрению и может передавать на этой волне как дискретную, так и аналоговую информацию. Но так как ширина спектра передаваемого сигнала пропорциональна частоте модуляции, а зазор между частотами соседних волн должен быть больше, чем спектр передаваемого сигнала, то при фиксированных значениях шага частотного плана возможность передачи на каждой волне того или иного закодированного сигнала явным образом зависит от его скорости (и принятого метода кодирования). Например, спектр сигнала STM-64 примерно в четыре раза шире спектра сигнала STM-16.

Теоретически зазоры между соседними волнами в 50 ГГц и даже 25 ГГц позволяют передавать данные со скоростями 10 Гбит/с, но при этом нужно обеспечить высокую точность частоты и минимально возможную ширину спектра несущей волны, а также снизить уровень шумов, чтобы минимизировать эффект перекрытия спектра (рис. 6.17).

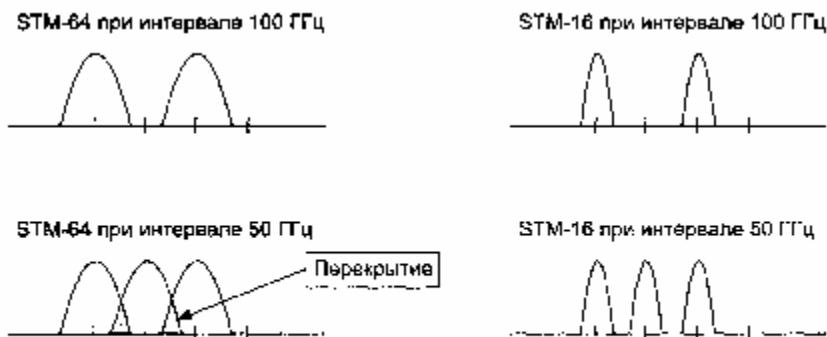


Рис. 6.17. Перекрытие спектра соседних волн для разных частотных планов и скоростей передачи данных

Практический успех технологии DWDM, оборудование которой уже работает на магистралях многих ведущих мировых операторов связи (в том числе, и некоторых российских), во многом определило появление волоконно-оптических усилителей на основе кварца, легированного эрбием (Erbium-Dopped Fiber Amplifier, EDFA). Эти оптические устройства непосредственно усиливают световые сигналы в диапазоне 1550 нм, исключая необходимость промежуточного преобразования их в электрическую форму, как это делают регенераторы, применяемые в сетях SDH. Системы электрической регенерации сигналов являются весьма дорогими и, кроме того, протокольно-зависимыми, так как они должны воспринимать определенный вид кодирования сигнала. Оптические усилители, «прозрачно» передающие информацию, позволяют наращивать скорость магистралей без необходимости модернизации усилительных блоков.

Протяженность участка между оптическими усилителями может достигать 150 км и более, что обеспечивает экономичность создаваемых магистралей DWDM, и которых длина мультиплексной секции составляет на сегодня 600-3000 км при применении от 1 до 7 промежуточных оптических усилителей.

Оптические усилители используются не только для увеличения расстояния между мультиплексорами, но и внутри самих мультиплексоров. Если мультиплексирование и кросс-коммутация выполняются исключительно оптическими средствами, без преобразования в электрическую форму, то сигнал при пассивных оптических преобразованиях теряет мощность и его нужно усиливать перед передачей на линию.

Новые исследования в области EDFA привели к появлению усилителей, работающих в L-диапазоне (4-е окно прозрачности), от 1570 до 1605 нм. Использование этого диапазона, а также сокращение расстояния между волнами до 50 ГГц и 25 ГГц позволяет нарастить количество одновременно передаваемых длин волн до 80-160 и более, то есть обеспечить передачу трафика со скоростями 800 Гбит/с-1,6 Тбит/с в одном направлении по одному оптическому волокну.

С успехами DWDM связано еще одно перспективное технологическое направление — полностью оптические сети (All-Optical Networks). В таких сетях все операции по мультиплексированию/демультиплексированию, вводу-выводу и

кросс-коммутации (маршрутизации) пользовательской информации выполняются без преобразования сигнала из оптической формы в электрическую (такие устройства и сети имеют обозначение «О-О-О», а аббревиатура «О-Е-О» применяется для традиционных оптико-электронных устройств). Исключение преобразований в электрическую форму позволяет существенно удешевить сеть, но возможности оптических технологий пока еще недостаточны для создания полностью оптических сетей нужного масштаба, поэтому практическое применение таких сетей ограничено фрагментами, между которыми выполняется электрическая регенерация сигнала. Тем не менее работы в этом направлении ведутся активно и полностью оптические системы кросс-коммутации уже выпускаются.

Ниже перечислены основные преимущества технологии DWDM.

- Дальнейшее повышение коэффициента использования частотного потенциала оптического волокна (его теоретическая полоса пропускания — 25 000 ГГц) — достижение терабитных скоростей.
- Отличная масштабируемость — повышение суммарной скорости сети за счет добавления новых спектральных каналов без необходимости замены всех магистральных модулей мультиплексоров (что требуется для перехода к новому уровню STM-N в сетях SDH).
- Экономическая эффективность за счет отказа от электрической регенерации на участках сети большой протяженности.
- Независимость от протокола передачи данных — технологическая «прозрачность», позволяющая передавать через магистраль DWDM трафик сетей любого типа.
- Независимость спектральных каналов друг от друга.
- Совместимость с технологией SDH — мультиплексоры DWDM оснащаются интерфейсами STM-N, способными принимать и передавать данные мультиплексоров SDH.
- Совместимость с технологиями семейства Ethernet — Gigabit Ethernet и 10GE.
- Стандартизация на уровне ITU-T.

Типовые топологии

Сети DWDM проходят в своем развитии те же этапы, что и сети SDH.

Цепь «точка-точка». Наиболее популярным применением DWDM на первых этапах жизненного цикла этой технологии является создание сверхдальных высокоскоростных магистралей (Ultralong Haul), в которых применяется топология цепи (на рис. 6.18 показаны 32 волны как типичный, но не единственный вариант).

Для организации такой магистрали достаточно в ее конечных точках установить терминальные мультиплексоры DWDM, а в промежуточных точках — оптические усилители, если этого требует расстояние между конечными точками. Терминальный мультиплексор включает собственно блок мультиплексирования - демультиплексирования, выходной (Booster, B) и предварительный (Pre-ampli-

fier, P) усилители, а также набор транспондеров (Transponder, T), преобразующих входные сигналы от источников, длина волны которых не соответствует частотному плану мультиплексора, в волны требуемой длины. В тех случаях, когда некоторое подключаемое к сети DWDM устройство способно вырабатывать сигнал по волне одной из частот, поддерживаемых мультиплексором DWDM (плана ITU-T G.692 или частного плана производителя), транспондеры не используются. В этом случае говорят, что подключаемое к сети DWDM устройство имеет «окрашенный» интерфейс.

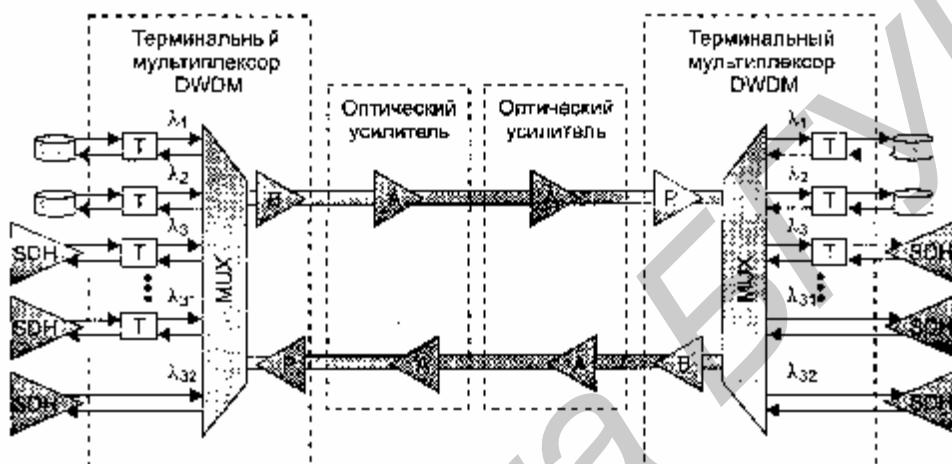


Рис. 6.18. Дальняя связь «точка-точка» на основе терминальных мультиплексоров DWDM

В рекомендации ITU-T G.692 определены три типа усилительных участков, то есть участков между двумя соседними мультиплексорами DWDM;

- L (Long) — участок состоит максимум из 8 пролетов ВОЛС и 7 оптических усилителей, максимальное расстояние между усилителями — до 80 км при общей максимальной протяженности участка 640 км;
- V (Very long) — участок состоит максимум из 5 пролетов ВОЛС и 4 оптических усилителей, максимальное расстояние между усилителями — до 120 км при общей максимальной протяженности участка 600 км;
- U (Ultra long) — участок без промежуточных усилителей длиной до 160 км.

Ограничения на количество пассивных участков и их длину связаны с некоторой деградацией оптического сигнала при его оптическом усилении. Хотя оптический усилитель EDFA восстанавливает мощность сигнала, но он не полностью компенсирует эффект хроматической дисперсии (то есть распространения волн разной длины с разной скоростью, из-за чего сигнал на приемном конце волокна «размазывается»), а также такие нелинейные эффекты, как романовское рассеивание, четырехволновое смешение и нестабильность модуляции. Поэтому для построения более протяженных магистралей необходимо между усилительными участками устанавливать мультиплексоры DWDM, выполняющие регенерацию сигнала за счет его преобразования в электрическую форму и обратно, то есть работающие по принципу О-Е-О. Ситуацию частично улучшают специальные компенсаторы дисперсии, но полностью восстановить качество сигнала они не могут. Для уменьшения нелинейных эффектов в системах

DWDM применяется также ограничение мощности сигнала, передаваемого по длинной линии. В приведенной выше схеме дуплексный обмен между абонентами сети происходит за счет односторонней передачи всего набора волн по двум волокнам. Существует и другой вариант работы сети DWDM, когда для связи узлов сети используется одно волокно. Дуплексный режим достигается за счет двунаправленной передачи оптических сигналов по волокну — половина волн частотного плана передают информацию в одном направлении, а половина — в обратном.

Цепь с промежуточными подключениями. Естественным развитием предыдущей топологии является сеть, в которой промежуточные узлы выполняют функции мультиплексоров ввода-вывода (рис. 6.19).

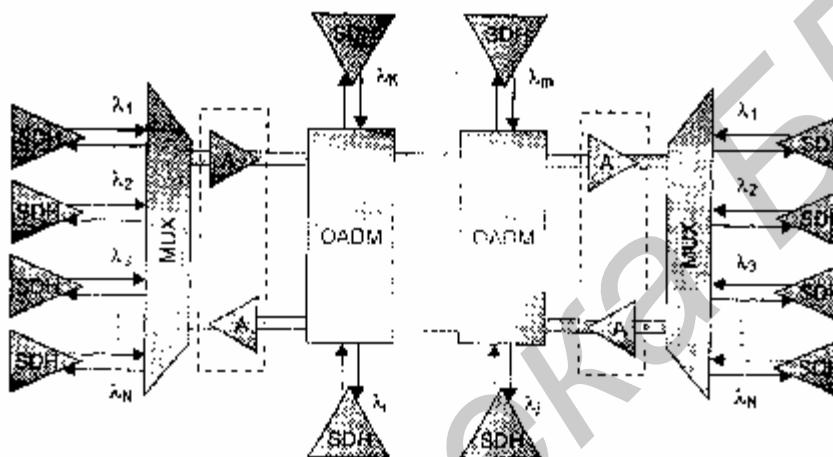


Рис. 6.19. Цепь DWDM с вводом-выводом в промежуточных узлах

Оптические мультиплексоры ввода-вывода (OADM) могут вывести из общего оптического сигнала волну определенной длины и ввести туда сигнал этой же длины волны, так что спектр транзитного сигнала не изменится, а соединение будет выполнено с одним из абонентов, подключенных к промежуточному мультиплексору. OADM может выполнять операции ввода-вывода волн оптическими средствами или путем промежуточного преобразования в электрическую форму. Обычно полностью оптические (пассивные) мультиплексоры ввода-вывода могут отводить небольшое число волн, так как каждая операция вывода требует последовательного прохождения оптического сигнала через оптический фильтр, который вносит дополнительное затухание. Если же мультиплексор использует электрическую регенерацию сигнала, то количество выводимых волн может быть любым в пределах имеющегося набора волн, так как транзитный оптический сигнал предварительно полностью демультиплексируется.

Кольцо. Кольцевая топология (рис. 6.20) обеспечивает живучесть сети DWDM за счет резервных путей. Методы защиты трафика, применяемые в DWDM, аналогичны методам SDH (хотя в DWDM они пока не стандартизованы). Если какое-либо соединение защищается, то между его конечными точками устанав-

ливаются два пути — основной и резервный. Мультиплексор конечной точки сравнивает два сигнала и выбирает сигнал лучшего качества (или сигнал по умолчанию).

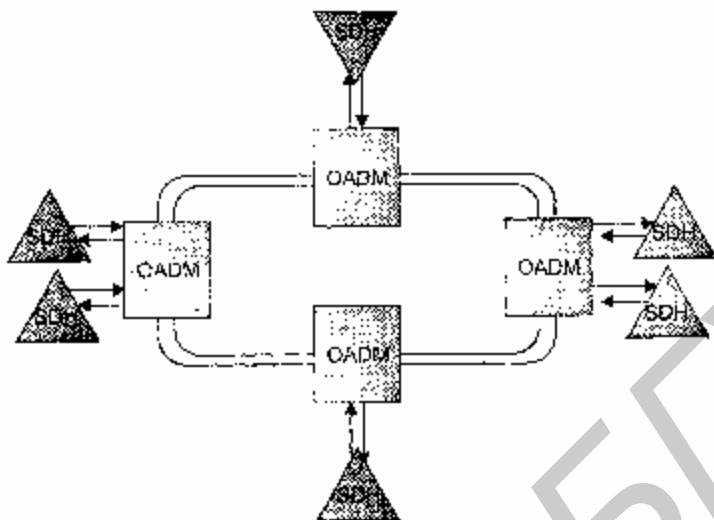


Рис. 6.20. Колыцо мультиплексоров DWDM

Ячеистая топология. По мере развития сетей DWDM в них все чаще будет применяться ячеистая топология (рис. 6.21), которая обеспечивает большую гибкость, производительность и отказоустойчивость, чем остальные топологии. Однако для реализации ячеистой топологии необходимо наличие оптических кросс-коннекторов (OXCs), поддерживающих произвольную коммутацию (а не только добавление волн в общий транзитный сигнал и вывод их оттуда, как это делают мультиплексоры ввода-вывода) между оптическими сигналами, передаваемыми волнами разной длины. Оборудование такого класса сравнительно недавно появилось на рынке, причем пока не все ведущие

производители имеют его в своих линейках продуктов, что затрудняет построение ячеистых сетей DWDM.

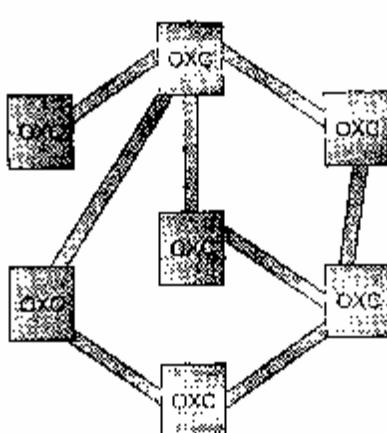


Рис. 6.21. Ячеистая топология сети DWDM

Источники излучения

Сегодня в системах DWDM применяются лазерные диоды с фиксированной волной излучения трех типов:

- с распределенной обратной связью (Distributed FcedBack, DFB);
- с распределенной решеткой Брэгга (Distributed Brag Reflector, DBR);
- на вертикальных резонаторах (Vertical-cavity surface-emitting laser, VCSEL).

В генераторах первых двух типов для получения излучения узкого спектра используются волоконные решетки Брэгга (Fiber Bragg Gratings). Волокно, легированное некоторыми веществами (обычно германием), может изменять свой показатель преломления под воздействием ультрафиолетового света. Если облучить такое волокно ультрафиолетовым излучением с определенной пространственной периодической структурой, то волокно превращается в своеобразную дифракционную решетку. То есть такое волокно будет отражать свет определенного, заранее заданного диапазона длин волн и пропускать свет всех остальных длин волн. Отраженный свет и является выходным сигналом генератора.

В генераторах DFB решетка Брэгга формируется в активной зоне лазера, поэтому технология изготовления таких устройств достаточно сложна. Ширина спектра подобных генераторов обычно составляет менее 100 МГц.

У генераторов DBR решетка расположена в неактивной зоне лазера, поэтому эффективная длина резонатора может быть существенно больше, чем у генераторов DFB, а полоса излучения — уже, до 1 МГц.

В лазерах VCSEL генерация происходит поперек р-п-перехода. Преимуществом таких генераторов является высокая технологичность: на одном чипе можно разместить матрицу лазеров, каждый из которых может излучать заданную длину волны в соответствии с требуемым планом частот.

Кроме фиксированных существуют также перестраиваемые генераторы, которые используются для тестирования систем DWDM, а также для построения оптических кросс-коннекторов. В перестраиваемом генераторе чаще всего применяется внешний резонатор на основе дифракционной решетки, поворот которой вызывает изменение длины выходной волны.

Оптические усилители

Применяемые сегодня повсеместно в оптических сетях усилители EDFA содержат вставку волокна с добавками ионов эрбия. Помимо этого, оптический усилитель (Optical Amplifier) имеет лазер накачки с излучением на определенной частоте и блок сопряжения лазера с легированным волокном. Лазерная накачка ведет к переходу атомов эрбия в возбужденное состояние. Прохождение фотона света вызывает возвращение атома в основное состояние с излучением волны соответствующей длины. Таким способом достигается усиление сигнала.

Для предотвращения опадания на выходное волокно излучения накачки усиленный сигнал пропускают через заграждающий фильтр с частотой накачки.

Частота накачки должна быть больше частоты основного сигнала, так как в противном случае роль накачки будет играть излучение сигнала. В современных усилителях для накачки используется излучение с длинами волн 980 нм и 1480 нм.

Усилители EDFA характеризуются полосой усиления, равномерностью усиления в этой полосе, а также уровнем вносимого шума. Стандартные усилители поддерживают полосу 1530-1565 нм, а для усиления в L-диапазоне волокно легируют не только эрбием, но и ионами туллия.

Оптическое волокно

Сети DWDM могут работать как на стандартном одномодовом волокне, так и на специальных типах волокон, разработанных с учетом требований DWDM.

Стандартное бездисперсионное одномодовое волокно (Non-Dispersion Shifted Fiber, NDSF, или Standard Single Mode Fiber, SSMF) оптимизировано для передачи волны 1310 нм и имеет нулевую дисперсию именно в этом районе спектра. Так как технология DWDM использует с настоящего времени окно прозрачности 1550 нм, то применение стандартного волокна является не самым лучшим вариантом — хроматическая дисперсия в окне 1550 нм в таком кабеле положительна и достигает значительных величин. Это приводит к серьезным ограничениям на дальность передачи без усиления сигнала.

Существует другой тип одномодового волокна — со смешенной нулевой дисперсией (Zero Dispersion Shifted Fiber, ZDSF). У такого волокна нулевое значение дисперсии приходится на диапазон от 1535 до 1565 нм, но применять его имеет смысл только для систем, передающих информацию на волне одной длины из этого диапазона, например для SDH, так как для части доли DWDM в этом окне дисперсия будет ненулевой и отрицательной, а для части — ненулевой и положительной, так что компенсировать ее оказывается непросто.

Наиболее подходящим для применения с современными системами DWDM является волокно со смешенной ненулевой дисперсией (Non-Zero Dispersion Shifted Fiber, NZDSF). У такого волокна нулевая дисперсия приходится не на середину интервала (1550 нм), а смешена за его границы (то есть длина волны, для которой дисперсия нулевая, меньше 1530 нм или больше 1560 нм). Благодаря этому для волн третьего окна прозрачности такое волокно, с одной стороны, обладает малой дисперсией, а с другой — эта дисперсия имеет один и тот же знак для всех волн и слабо подвержена нелинейным эффектам, что облегчает ее компенсацию.

Для компенсации хроматической дисперсии применяются модули, содержащие отрезок оптического волокна с характеристикой дисперсии, противоположной характеристике применяемого в ВОЛС волокна. Суммарный эффект волокна кабеля и компенсатора должен выровнять скорости распространения волн различной длины. Поляризационная дисперсия сигнала, сказывающаяся на его качестве при скоростях передачи данных 10 Гбит/с и выше, таким образом ком-

пенсирована быть не может, для ее уменьшения необходимо уменьшать бюджет линии.

Мультиплексоры ввода-вывода

Терминальный мультиплексор состоит из нескольких блоков (см. рис. 6.18): собственно мультиплексор длин воли (MUX), бустерный усилитель (B), приемный усилитель (P), оптические транспондеры (T).

Оптический мультиплексор выполняет операции смешения нескольких длин волн в общий сигнал, а также выделения волн различной длины из общего сигнала. Для выделения волн в мультиплексоре могут использоваться разнообразные оптические механизмы.

- Тонкопленочные фильтры (Thin-Film Filters) состоят из пластин с многослойным покрытием, в качестве которых на практике используется торец оптического волокна, скошенный под углом 30-45°, с нанесенным на него слоями покрытия. Достоинством этого подхода является применение известной на протяжении десятилетий техники, использующей интерференцию для фильтрации волн определенной длины. Недостаток — ограниченный выбор материала покрытий, сочетающих необходимые оптические и физические свойства. Тонкопленочные фильтры применяются в оптических мультиплексорах, поддерживающих сравнительно небольшое количество длин волн в волокне, обычно 16 или 32. Для систем с большим числом волн требуются другие принципы фильтрации и мультиплексирования.
- Волоконные решетки Брэгга (Fiber Bragg Gratings), принцип действия которых был рассмотрен выше, используются обычно в тех случаях, когда из общего сигнала нужно выделить небольшое число волн — для этого последовательно применяют несколько волокон-решеток, каждая из которых выделяет одну волну. Так как последовательное применение таких пассивных фильтров вносит дополнительное затухание, с их помощью обычно ограничиваются выделением одной-двух волн. Пассивные устройства, позволяющие выделить из общего сигнала одну-две волны, обычно называют ответителем (coupler).
- Принцип действия дифракционных фазовых решеток (Diffraction Phase Gratings), называемых также эшелонами Майкельсона, или дифракционными структурами (Airayed Waveguide Grating, AWG), состоит в том, что свет проpusкается через несколько сложенных строго параллельных пластин разной длины, при этом разность длин постоянна. Сложенные пластины образуют единую прозрачную призму, одна грань которой плоская, а противоположная — ступенчатая, с одинаковыми ступеньками. Пройдя через всю призму, лучи дифрагируют, при этом угол дифракции зависит от длины волны.

В мультиплексорах DWDM применяются интегрально выполненные фазовые решетки. Функции пластин в них выполняют оптические волноводы или волокна. Приходящий мультиплексный сигнал попадает на входной порт (рис. 6.22, а). Затем этот сигнал проходит через волновод-пластины и распределяется по множеству волноводов, представляющих дифракционную структуру AWG. По-прежнему сигнал в каждом из волноводов остается мультиплексным, а каж-

дый канал ($\lambda_1, \lambda_2, \dots, \lambda_N$) остается представленным во всех волноводах. Далее происходит отражение сигналов от зеркальной поверхности, и в итоге световые потоки вновь собираются в волноводе-пластины, где происходит их фокусировка и интерференция — образуются пространственно разнесенные интерференционные максимумы интенсивности, соответствующие разным каналам. Геометрия волновода-пластины, в частности расположение выходных полюсов, и значения длины волноводов структуры AWG рассчитываются таким образом, чтобы интерференционные максимумы совпадали с выходными полюсами. Мультиплексирование происходит обратным путем.

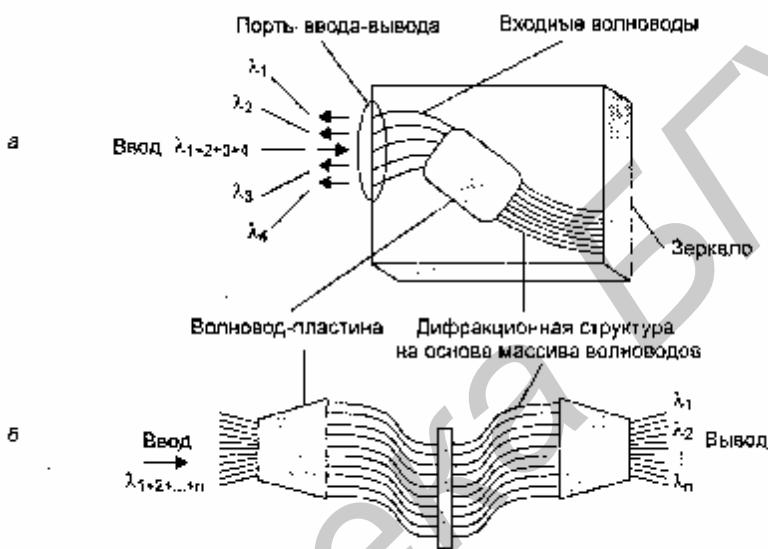


Рис. 6.22. Полное демультиплексирование сигнала с помощью дифракционной фазовой решетки

Другой способ построения мультиплексора базируется не на одной, а на паре волноводов-пластин (рис. 6.22, б). Принцип действия такого устройства аналогичен предыдущему случаю, за исключением того, что здесь для фокусировки и интерференции используется дополнительная пластина.

Интегральные решетки AWG (называемые также фазарами) стали одними из ключевых элементов мультиплексоров DWDM. Они обычно применяются для полного демультиплексирования светового сигнала, так как хорошо масштабируются и потенциально могут успешно работать в системах с сотнями спектральных каналов.

Регенераторы

Регенератор включает следующие элементы:

- блок демультиплексирования комбинированного оптического сигнала;
- блок преобразования оптического сигнала каждой длины волны в электрический сигнал (фотодиоды, чувствительные к определенной длине волны);
- блок электрической регенерации, который восстанавливает форму сигнала и его тактовую частоту;

- блок преобразования электрического сигнала и оптический сигнал (лазерные генераторы);
- блок мультиплексирования, вырабатывающий комбинированный световой сигнал.

Оптические транспондеры

Оптические транспондеры (трансляторы) требуются для изменения длины волны оптического сигнала. Обычно эти устройства применяются в терминальных мультиплексорах для преобразования поступающих от пользовательского оборудования (SDH, коммутаторы/маршрутизаторы пакетных сетей) сигналов в стандартных окнах 850, 1300 или 1550 нм в сигнал одной из волн частотного плана ITU-T (и обратного преобразования). Транспондеры являются также и регенераторами сигналов, так как чаще всего изменение длины волны сигнала происходит за счет его промежуточного преобразования в электрическую форму.

Существуют также полностью оптические преобразователи длин волн, называемые оптическими конверторами (Optical Converter), или оптическими обменниками (Optical Changer, OX), — термин Optical Translator оставлен за оптико-электрическими преобразователями. Полностью оптические преобразователи длин волн могут быть созданы на основе различных принципов, в частности с применением межфазовой модуляции в волоконном усилителе, когда входящая волна используется для модулирования исходящей волны. Полностью оптическое преобразование длины волны существенно удешевляет сеть, так как позволяет отказаться от дорогостоящего промежуточного преобразования сигнала в электрическую форму.

В полностью оптической сети, где применяются оптические преобразователи, различают два вида сервиса: волновой путь (Wavelength Path) и световой путь (Light Path); последний часто называют виртуальным волновым путем (Virtual Wavelength Path). В первом случае информация передается через сеть без изменения длины волны, во втором — длина волны может меняться в зависимости от наличия свободных волн на данном участке сети. При сложной топологии сети проложить волновые пути для всех пользователей может оказаться невозможно даже при самом тщательном планировании, поэтому применение оптических транспондеров или конверторов становится необходимым.

Оптические кросс-коннекторы

В сетях с ячеистой топологией необходимо обеспечить гибкие возможности для изменения маршрута следования волновых соединений между абонентами сети. Такие возможности предоставляют оптические кросс-коннекторы (OXC), позволяющие направить любую из волн входного сигнала каждого порта в любой из выходных портов (конечно, при условии, что никакой другой сигнал этого порта не использует эту волну, иначе необходимо выполнить трансляцию дли-

ны волны).

Существуют оптические кросс-коннекторы двух типов: с промежуточным преобразованием в электрическую форму и полностью оптические. Исторически первыми появились более традиционные оптоэлектронные кросс-коннекторы, за которыми и закрепилось название оптических кросс-коннекторов. Поэтому производители полностью оптических устройств этого типа стараются использовать для них отличающиеся названия — фотонные коммутаторы (Photonic Switches), или маршрутизаторы волн (Wave Routers, или Lambda Routers). У оптических кросс-коммутаторов имеется принципиальное ограничение — они хорошо справляются со своими обязанностями при работе на скоростях до 2,5 Гбит/с, но, начиная со скорости 10 Гбит/с и выше, габариты таких устройств и потребление энергии становятся недопустимыми. Фотонные коммутаторы свободны от такого ограничения.

В фотонных коммутаторах используются различные оптические механизмы, в том числе дифракционные фазовые решетки и микроэлектронные механические системы (Micro-Electro Mechanical Systems, MEMS).

Система MEMS представляет собой набор подвижных зеркал очень маленького размера, с диаметром менее миллиметра (рис. 6.23). Коммутатор MEMS применяется после демультиплексора, когда исходный сигнал уже разделен на составляющие волны. За счет поворота микрозеркала на заданный угол исходный луч определенной волны направляется в соответствующее выходное волокно. Затем все лучи мультиплексируются в общий выходной сигнал.

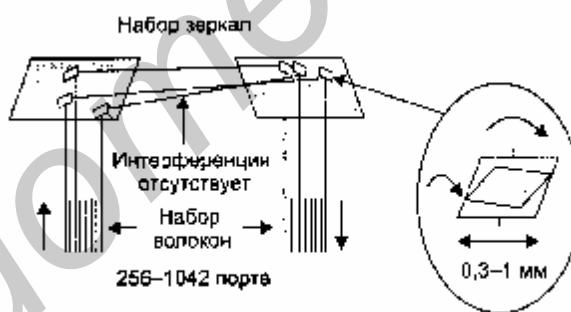


Рис. 6.23. Микроэлектронная механическая система кросс-коммутации

По сравнению с оптоэлектронными кросс-коннекторами, фотонные коммутаторы занимают объем в 30 раз меньше и потребляют примерно в 100 раз меньше энергии. Однако этот тип устройств обладает и недостатками, в первую очередь — это низкое быстродействие и чувствительность к вибрации. Тем не менее системы MEMS находят широкое применение в новых моделях фотонных коммутаторов. Сегодня подобные устройства могут обеспечивать коммутацию 256x256 спектральных каналов, и планируется выпуск устройств, позволяющих коммутировать 1024x1024 каналов и выше.

Разрабатываются также фотонные коммутаторы на других принципах — термооптические и электрооптические. В устройствах этого типа используется эффект изменения показателя преломления под воздействия нагрева или электри-

ческого поля.

Выводы

- Технология мультиплексирования по длине волны WDM/DWDM реализует принципы частотного мультиплексирования для сигналов иной физической природы и на новом уровне иерархии скоростей. Каждый канал WDM/DWDM представляет собой определенный диапазон световых волн, позволяющих переносить данные в аналоговой и цифровой форме, при этом полоса пропускания канала в 25-50-100 ГГц обеспечивает скорости в несколько гигабит в секунду (при передаче дискретных данных).
- В ранних системах WDM использовалось небольшое количество спектральных каналов, от 2 до 16. В системах плотного волнового мультиплексирования используется от 32 до 160 каналов на одном оптическом волокне, что обеспечивает скорости передачи данных для одного волокна до нескольких терабит в секунду.
- Современные эрбьевые оптические усилители позволяют расширить оптический участок линии связи (без преобразования сигнала в электрическую форму) до 700-1000 км,
- Для выделения нескольких каналов из общего светового сигнала разработаны сравнительно недорогие устройства, которые обычно объединяются с оптическими усилителями для организации мультиплексоров ввода-вывода в сетях дальней связи.
- Для взаимодействия с традиционными оптическими сетями (SDH, GE) в сетях DWDM применяются транспондеры и трансляторы длин волн, которые преобразуют длину волны входного сигнала в длину одной из волн стандартного частотного плана DWDM.
- В полностью оптических сетях все операции мультиплексирования и коммутации каналов выполняются над световыми сигналами, без промежуточного преобразования в электрическую форму. Это упрощает и удешевляет сеть.

Вопросы и упражнения к части II

1. Могут ли цифровые линии связи передавать аналоговые данные?
2. Каким будет теоретический предел скорости передачи данных к битах в секунду по каналу с шириной полосы пропускания в 20 кГц, если мощность передатчика составляет 0,01 мВт, а мощность шума в канале равна 0,0001 мВт?
3. Определите пропускную способность канала связи для каждого из направлений дуплексного режима, если известно, что его полоса пропускания равна 600 кГц, а в методе кодирования используется 10 состояний сигнала.
4. Рассчитайте задержку распространения сигнала и задержку передачи данных для случая передачи пакета в 128 байт (считайте скорость распространения сигнала равной скорости света в вакууме 300 000 км/с):

О по кабелю витой пары длиной в 100 м при скорости передачи 100 Мбит/с; О по коаксиальному кабелю длиной в 2 км при скорости передачи в 10 Мбит/с; О по спутниковому геостационарному каналу протяженностью в 72 000 км при скорости передачи 128 кбит/с.

5. Какой кадр передаст на линию передатчик, если он работает с использованием техники бит-стаффинга с флагом 7E, а на вход передатчика поступила последовательность 24 A5 7E 56 8C (все значения — шестнадцатеричные)?
6. Поясните, из каких соображений выбрана пропускная способность 64 кбит/с элементарного канала цифровых телефонных сетей?
7. Назовите методы компрессии, наиболее подходящие для текстовой информации. Почему они неэффективны для сжатия двоичных данных?
8. Предложите коды неравной длины для каждого из символов А, В, С, D, F и О, если нужно передать сообщение BDDACAAFOOOAOOOO. Будет ли достигнута компрессия данных по сравнению с использованием:
О традиционных кодов ASCII;
О кодов равной длины, учитывающих наличие только данных символов?
9. Как передатчик определяет факт потери положительной квитанции и методе скользящего окна?
10. Сеть с коммутацией пакетов испытывает перегрузку. Для устранения этой ситуации размер окна в протоколах компьютеров сети нужно увеличить или уменьшить?
11. Как влияет надежность линий связи в сети на выбор размера окна?
12. В чем проявляется избыточность TDM-технологии?
13. Что является аналогом адреса в технологии TDM?
14. Какова начальная скорость иерархии PDH и с чем она связана?
15. Почему в TDM-сетях невозможно временно отдать пропускную способность одного канала другому?
16. Каковы основные ограничения технологии PDH?
17. В какой из типовых топологий SDH обеспечивается отказоустойчивость?
18. Требуется ли жесткий синхронизм между агрегатным и трибутарным потоками данных в сети SDH?
19. Какие механизмы используются для устранения рассинхронизации агрегатного и трибутарного потоков?
20. В каких случаях механизм защиты MS-SPRing оказывается предпочтительнее механизма защиты SNC-P?
21. Какой протокол используется в сетях SDH для выбора одного источника синхронизации при нескольких возможных?

Часть III
Локальные сети

Библиотека БГУИР

ГЛАВА 7 Базовые технологии локальных сетей

Локальные сети получили распространение при появлении мини-компьютеров, сравнительно низкая стоимость которых позволила многим предприятиям и организациям устанавливать в одном здании несколько таких компьютеров. Локальные сети объединяют компьютеры одного здания или нескольких рядом расположенных зданий в единую сеть, при этом технологии локальных сетей обеспечивают экономичное соединение компьютеров за счет использования стандартных топологий и качественных кабельных систем. В результате селекции, проведенной практикой, в арсенале разработчика осталось несколько базовых технологий, на основе которых работает подавляющее большинство локальных современных сетей: Ethernet, Token Ring, FDDI.

Общая характеристика протоколов локальных сетей

Стандартная топология и разделяемая среда

При организации взаимодействия узлов в локальных сетях основная роль отводится протоколу канального уровня. Однако для того, чтобы канальный уровень мог справиться с этой задачей, структура локальных сетей должна быть вполне определенной, так, например, наиболее популярный протокол канального уровня Ethernet рассчитан на параллельное подключение всех узлов сети к общей для них среде передачи данных — отрезку коаксиального кабеля или иерархической древовидной структуре сегментов, образованных повторителями. Протокол Token Ring также рассчитан на вполне определенную конфигурацию — соединение компьютеров в виде логического кольца.

Подобный подход, заключающийся в использовании простых типовых структур кабельных соединений между компьютерами локальной сети, соответствовал основной цели, которую ставили перед собой разработчики первых локальных сетей во второй половине 70-х годов. Эта цель заключалась в нахождении простого и дешевого решения для объединения в вычислительную сеть нескольких десятков компьютеров, находящихся в пределах одного здания. Решение должно было быть недорогим, поскольку в сеть объединялись недорогие компьютеры — появившиеся и быстро распространившиеся тогда мини-компьютеры стоимостью в 10000-20000 долларов. Количество их в одной организации было небольшим, поэтому предел в несколько десятков (максимум — до сотни) компьютеров представлялся вполне достаточным для практически любой локальной сети.

Для упрощения и, соответственно, удешевления аппаратных и программных решений разработчики первых локальных сетей остановились на совместном использовании кабелей всеми компьютерами сети в режиме разделения времени. Наиболее явным образом режим совместного использования кабеля проявляется в сетях Ethernet шинной топологии, где коаксиальный кабель является

физически неделимым, общим для всех узлов сети. В технологиях Token Ring и FDDI отрезки кабеля, соединяющие компьютеры, в логическое кольцо, также образуют единую разделяемую среду, доступ к которой в каждый момент может получить только один компьютер сети. Использование разделяемых сред (shared media) позволяет упростить логику работы сети. Действительно, поскольку в каждый момент времени выполняется только одна передача, отпадает необходимость в буферизации кадров в транзитных узлах. В сетях других типов, где отрезки кабелей, соединяющих отдельные узлы, не рассматриваются как общий разделяемый во времени ресурс, такая необходимость возникает, и для решения этой проблемы в протоколы обмена информацией вводятся весьма сложные процедуры управления потоком кадров, предотвращающие переполнение каналов связи и буферов узлов сети.

Использование в локальных сетях очень простых конфигураций (звезда, общая шина и кольцо) наряду с положительными имело и отрицательные последствия, из которых наиболее неприятными были ограничения по производительности и надежности. Наличие только одного пути передачи информации, разделяемого всеми узлами сети, в принципе ограничивало пропускную способность сети пропускной способностью этого пути (которая делилась в среднем на число компьютеров сети), а надежность сети — надежностью этого пути. Поэтому по мере повышения популярности локальных сетей и расширения их сфер применения все больше стали применяться специальные коммуникационные устройства — мосты, коммутаторы и маршрутизаторы, — которые в значительной мере снимали ограничения единственной разделяемой среды передачи данных. Базовые конфигурации в форме звезды, общей шины и кольца превратились в элементарные структуры локальных сетей, которые стало возможным соединять друг с другом более сложным образом, образуя параллельные пути между узлами.

Тем не менее внутри крупных локальных сетей по-прежнему часто используются сегменты с разделяемой средой, в которых работают все те же протоколы разделяемых единственных сред передачи данных, которые были разработаны более 15 лет назад. Хорошие скоростные и надежностные характеристики кабелей локальных сетей все еще удовлетворяют пользователей небольших компьютерных сетей и позволяют им строить сети без больших затрат только с помощью сетевых адаптеров и кабеля или концентратора. В последние несколько лет наметилось движение к отказу от разделяемых сред передачи данных в локальных сетях и переходу к применению активных коммутаторов, к которым конечные узлы присоединяются индивидуальными линиями связи. Такой подход предлагается в технологии ATM (когда она используется в локальных сетях), а также в технологиях, носящих традиционные названия с приставкой switched (коммутируемый); switched Ethernet, switched Token Ring, switched FDDI. Применяется и смешанный подход, когда конечные узлы соединяются в небольшие разделяемые сегменты с помощью повторителей, а сегменты соединяются друг с другом с помощью индивидуальных коммутируемых связей.

Существует и достаточно заметная тенденция к использованию в традиционных технологиях так называемой микросегментации, когда конечные узлы сра-

зу соединяются с коммутатором индивидуальными каналами. Такие сети получаются дороже разделяемых или смешанных, но производительность их выше. При использовании коммутаторов у традиционных технологий появился новый режим работы — полнодуплексный (full-duplex). В разделяемом сегменте станции всегда работают в полудуплексном режиме (half-duplex), так как в каждый момент времени сетевой адаптер станции либо передает свои данные, либо принимает чужие, но никогда не делает это одновременно. Это справедливо для всех технологий локальных сетей, так как разделяемые среды поддерживаются не только классическими технологиями локальных сетей Ethernet, Token Ring, FDDI, но и всеми новыми — Fast Ethernet, 100VG-AnyLAN, Gigabit Ethernet. В полнодуплексном режиме сетевой адаптер может одновременно передавать свои данные в сеть и принимать из сети чужие данные. Такой режим несложно обеспечивается при прямом соединении с мостом/коммутатором или маршрутизатором, так как вход и выход каждого порта такого устройства работают независимо друг от друга, каждый со своим буфером кадров.

Сегодня каждая технология локальных сетей приспособлена для работы как в полудуплексном, так и полнодуплексном режимах. В этих режимах ограничения, накладываемые на общую длину сети, существенно отличаются, так что одна и та же технология может позволять строить существенно различные сети в зависимости от выбранного режима работы (который зависит от того, какие устройства используются для соединения узлов — повторители или коммутаторы). Например, технология Fast Ethernet позволяет для полудуплексного режима строить сети диаметром не более 200 м, а для полнодуплексного режима ограничений на диаметр сети не существует. Поэтому при сравнении различных технологий необходимо обязательно принимать во внимание возможность их работы в двух режимах. В данной главе изучается в основном полудуплексный режим работы протоколов, а полнодуплексный режим рассматривается в главе 9.

Несмотря на появление новых технологий, классические протоколы локальных сетей Ethernet и Token Ring по прогнозам специалистов будут повсеместно использоваться еще по крайней мере лет 5-10, в связи с чем знание их деталей необходимо для успешного применения современной коммуникационной аппаратуры. Кроме того, некоторые современные высокопроизводительные технологии, такие как Fast Ethernet, Gigabit Ethernet, рассматриваемые в главе 8, в значительной степени сохраняют преемственность со своими предшественниками. Это еще раз подтверждает важность изучения классических протоколов локальных сетей.

Структура стандартов IEEE 802.x

В 1980 году в институте IEEE был организован комитет 802 по стандартизации локальных сетей, в результате работы которого было принято семейство стандартов IEEE 802.x, которые содержат рекомендации по проектированию нижних уровней локальных сетей. Позже результаты работы этого комитета легли в основу комплекса международных стандартов ISO 8802-1...x. Эти стандарты

были созданы на основе очень распространенных фирменных стандартов сетей Ethernet, ArcNet и Token Ring.

Помимо IEEE в работе по стандартизации протоколов локальных сетей принимали участие и другие организации. Так, для сетей, работающих на оптоволокне, американским институтом по стандартизации ANSI был разработан стандарт FDDI, обеспечивающий скорость передачи данных 100 Мбит/с. Стандарты семейства IEEE 802.x охватывают только два нижних уровня семиуровневой модели OSI — физический и канальный. Это связано с тем, что именно эти уровни в наибольшей степени отражают специфику локальных сетей. Старшие же уровни, начиная с сетевого, в значительной степени имеют общие черты как для локальных, так и для глобальных сетей.

Специфика локальных сетей также нашла свое отражение в разделении канального уровня (Data Link Layer) на два подуровня, которые часто также называют уровнями:

- управление логическим каналом (Logical Link Control, LLC);
- управление доступом к среде (Media Access Control, MAC).

Уровень MAC появился из-за существования в локальных сетях разделяемой среды передачи данных. Именно этот уровень обеспечивает корректное совместное использование общей среды, предоставляя ее в соответствии с определенным алгоритмом в распоряжение того или иного узла сети. В современных локальных сетях получили распространение несколько протоколов уровня MAC, реализующих различные алгоритмы доступа к разделяемой среде. Эти протоколы полностью определяют специфику таких технологий, как Ethernet, Fast Ethernet, Giga-bit Ethernet, Token Ring, FDDI, 100VG-AnyLAN.

После того как доступ к среде получен, ею может пользоваться более высокий уровень — уровень LLC, организующий передачу логических единиц данных, кадров информации, с различным уровнем качества транспортных услуг.

Уровень LLC отвечает за передачу с различной степенью надежности кадров данных между узлами, а также реализует функции интерфейса с прилегающим к нему сетевым уровнем. Именно уровень LLC принимает запрос от сетевого протокола на выполнение транспортной операции канального уровня с тем или иным качеством. Протокол LLC поддерживает несколько режимов работы, отличающихся наличием или отсутствием процедур восстановления кадров в случае их потери или искажения, то есть отличающихся качеством транспортных услуг.

Структуру стандартов IEEE 802 иллюстрирует рис. 7.1.

Эта структура появилась в результате большой работы, проведенной комитетом 802 по выделению в разных фирменных технологиях общих подходов и общих функций, а также согласованию стилей их описания. В результате канальный уровень был разделен на два упомянутых подуровня.

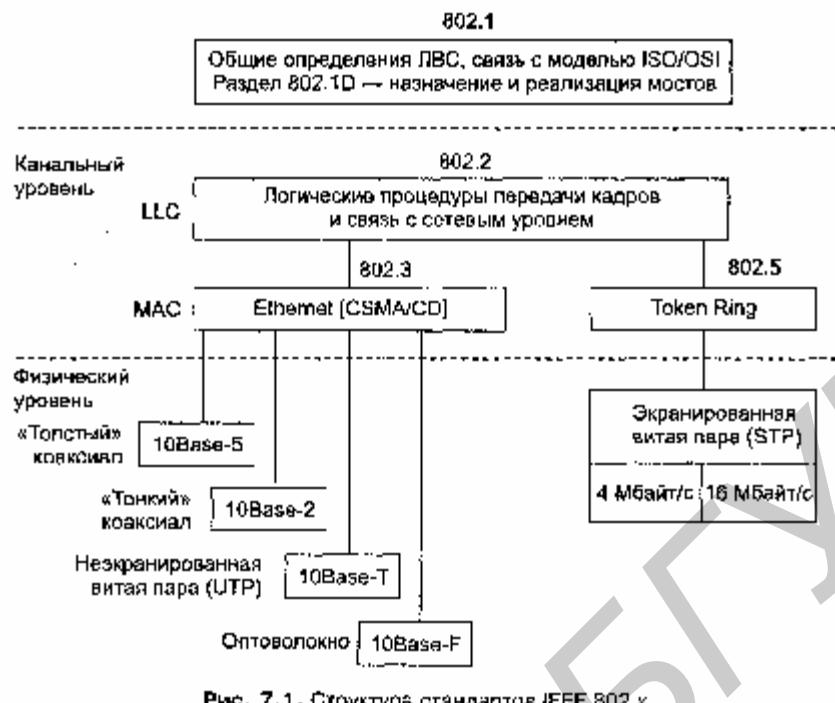


Рис. 7.1. Структура стандартов IEEE 802.x

Над уровнем MAC всех технологий показан общий для всех них протокол LLC, поддерживающий несколько режимов работы, но независимый от конкретной выбранной технологии. Стандарт LLC курирует подкомитет 802.2. Даже технологии, стандартизованные не в рамках комитета 802, ориентируются на использование протокола LLC, определенного стандартом 802.2, например, таким образом устроен протокол FDDI, стандартизованный ANSI.

Описание каждой технологии разделено на две части: описание уровня MAC и описание физического уровня. Как видно из рисунка, практически у каждой технологии единственному протоколу уровня MAC соответствует несколько вариантов протоколов физического уровня (на рисунке, в целях экономии места приведены только технологии Ethernet и Token Ring, но все сказанное справедливо также и для других технологий, таких как ArcNet, FDDI, FastEthernet, Gigabit-Ethernet).

Особняком стоят стандарты, разрабатываемые подкомитетом 802.1. Эти стандарты носят общий для всех технологий характер. В подкомитете 802.1 были разработаны общие определения локальных сетей и их свойств, определена связь трех уровней модели IEEE 802 с моделью OSI. Но наиболее практически важными являются стандарты 802.1, которые описывают взаимодействие между собой различных технологий, а также стандарты по построению более сложных сетей на основе базовых топологий. Эта группа стандартов носит общее название стандартов межсетевого взаимодействия (internetworking). Сюда входят такие важные стандарты, как стандарт 802.1D, описывающий логику работы прозрачного моста/коммутатора, стандарт 802.1H, определяющий функционирование транслирующего моста, который может без маршрутизатора объединять сети Ethernet и FDDI, Ethernet и Token Ring и т. п. Сегодня набор стандартов, разработанных подкомитетом 802.1, продолжает расти. Например,

сравнительно недавно он пополнился двумя важными стандартами: 802.1Q, определяющим способ построения виртуальных локальных сетей VLAN в сетях на основе коммутаторов, и 802.1p, описывающим способ приоритезации трафика на канальном уровне.

Стандарты 802.3, 802.4, 802.5 и 802.12 описывают технологии локальных сетей, которые появились в результате улучшений фирменных технологий, легших в их основу. Так, основу стандарта 802.3 составила технология Ethernet, разработанная компаниями Digital, Intel и Xerox (или Ethernet DIX), стандарт 802.4 появился как обобщение технологии ArcNet компании Datapoint Corporation, а стандарт 802.5 в основном соответствует технологии Token Ring компании IBM.

Комитет 802.11 занимается разработкой локальных радиосетей с методами доступа к среде, близкими к тем, которые используются в сетях Ethernet. Поэтому стандарты 802.11 получили название радио-Ethernet. Исходные фирменные технологии и их модифицированные варианты — стандарты 802.x к ряду случаев долгие годы существовали параллельно. Например, технология ArcNet так до конца не была приведена в соответствие со стандартом 802.4 (теперь это делать поздно, так как где-то примерно с 1993 года производство оборудования ArcNet было свернуто). Расхождения между технологией Token Ring и стандартом 802.5 тоже периодически возникают, так как компания IBM регулярно вносит усовершенствования в свою технологию и комитет 802.5 отражает эти усовершенствования в стандарте с некоторым запозданием. Исключение составляет технология Ethernet. Последний фирменный стандарт Ethernet DIX был принят в 1980 году, и с тех пор никто больше не предпринимал попыток фирменного развития Ethernet. Все новшества в семействе технологий Ethernet вносятся только в результате принятия открытых стандартов комитетом 802.3.

Более поздние стандарты изначально разрабатывались не одной компанией, а группой заинтересованных компаний, а потом передавались в соответствующий подкомитет IEEE 802 для утверждения. Так произошло с технологиями Fast Ethernet, 100VG-AnyLAN, Gigabit Ethernet. Группа заинтересованных компаний образовывала сначала небольшое объединение, а затем по мере развития работ к нему присоединялись другие компании, так что процесс принятия стандарта носил открытый характер.

Сегодня комитет 802 включает следующие подкомитеты (ряд из которых уже прекратили свою активную деятельность):

- 802.1 — Internetworking — объединение сетей;
- 802.2 — Logical Link Control, LLC — управление логической передачей данных;
- 802.3 — Ethernet с методом доступа CSMA/CD;
- 802.4 — Token Bus LAN — локальные сети с методом доступа Token Bus;
- 802.5 — Token Ring LAN — локальные сети с методом доступа Token Ring;
- 802.6 — Metropolitan Area Network, MAN — сети мегаполисов;
- 802.7 — Broadband Technical Advisory Group — техническая консультационная группа по широкополосной передаче;
- 802.8 — Fiber Optic Technical Advisory Group — техническая консультацион-

- ная группа по волоконно-оптическим сетям;
- 802.9 — Integrated Voice and data Networks — интегрированные сети передачи голоса и данных;
 - 802.10 — Network Security — сетевая безопасность; Q 802.11 — Wireless Networks — беспроводные сети;
 - 802.12 — Demand Priority Access LAN, I/OVG-AnyLAN — локальные сети с методом доступа по требованию с приоритетами.

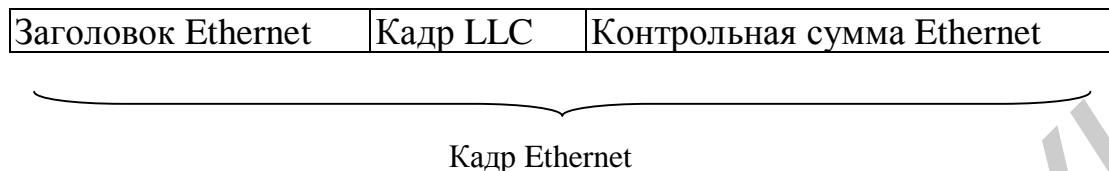
Выводы

- При организации взаимодействия узлов в локальных сетях основная роль отводится классическим технологиям Ethernet, Token Ring, FDDI, разработанным более 15 лет назад и основанным на использовании разделяемых сред.
- Разделяемые среды поддерживаются не только классическими технологиями локальных сетей Ethernet, Token Ring, FDDI, но и новыми — Fast Ethernet, Gigabit Ethernet.
- Современной тенденцией является частичный или полный отказ от разделяемых сред: соединение узлов индивидуальными связями, широкое использование коммутируемых связей и микросегментации. Еще одна важная тенденция — появление полнодуплексного режима работы практически для всех технологий локальных сетей.
- Комитет IEEE 802.x разрабатывает стандарты, которые содержат рекомендации для проектирования нижних уровней локальных сетей — физического и канального. Специфика локальных сетей нашла свое отражение в разделении канального уровня на два подуровня — LLC и MAC.
- Стандарты подкомитета 802.1 носят общий для всех технологий характер, они определяют типы локальных сетей, их свойства, процедуры межсетевого взаимодействия, логику работы мостов/коммутаторов.
- Стандарт 802.2 определяет функции уровня управления логическим каналом LLC.
- Стандарты 802.3, 802.4, 802.5 описывают технологии локальных сетей, которые появились в результате улучшений фирменных технологий, легших в их основу, соответственно Ethernet, ArcNet, Token Ring.

Протокол LLC

Протокол LLC (стандарт подкомитета 802.2) обеспечивает для технологий локальных сетей нужное качество услуг транспортной службы, передавая свои кадры либо дейтаграммным способом, либо с помощью процедур с установлением соединения и восстановлением кадров. Протокол LLC занимает уровень между сетевыми протоколами и протоколами уровня MAC. Протоколы сетевого уровня передают через межуровневый интерфейс данные для протокола LLC — свой пакет (например, пакет IP, IPX или NetBEUI), адресную информацию об узле назначения, а также требования к качеству транспортных услуг, кото-

рое протокол LLC должен обеспечить. Протокол LLC помещает пакет протокола верхнего уровня в свой кадр, который дополняется необходимыми служебными полями, Далее через межуровневый интерфейс протокол LLC передает свой кадр вместе с адресной информацией об узле назначения соответствующему протоколу уровня MAC, который упаковывает кадр LLC в свой кадр (например, кадр Ethernet), имеющий следующий формат:



В основу протокола LLC положен протокол HDLC (High-level Data Link Control), являющийся стандартом ISO. Собственно стандарт HDLC представляет собой обобщение протокола SDLC, разработанного компанией IBM. На основе HDLC в ITU-T разработано несколько стандартов канального уровня: протокол LAP-B сетей X.25, протокол LAP-D, используемый в сетях ISDN, и протокол LAP-M/V.42, работающий в современных модемах. В спецификации IEEE 802.2, как и во всех перечисленных протоколах, имеются некоторые отличия от стандарта HDLC.

Первоначально в фирменных технологиях подуровень LLC не выделялся и его функции растворялись в общих функциях протокола канального уровня. Из-за больших различий в функциях протоколов фирменных технологий, которые можно отнести к уровню LLC, на уровне LLC пришлось свести три типа процедур. Протокол сетевого уровня может обращаться к одной из этих процедур.

Три типа процедур уровня LLC

В соответствии со стандартом 802.2 уровень управления логическим каналом LLC предоставляет верхним уровням три типа процедур:

- LLC1 — процедура без установления соединения и без подтверждения получения данных;
- LLC2 — процедура с установлением соединения и с подтверждением получения данных;
- LLC3 — процедура без установления соединения, но с подтверждением получения данных.

Этот набор процедур является общим для всех методов доступа к среде, определенных стандартами 802.3-802.5, а также стандартом FDDI и стандартом 802.12 на технологию 100VG-AnyLAN.

Процедура без установления соединения и без подтверждения LLC1 даст пользователю средства для передачи данных с минимумом издержек. Это дейтаграммный режим работы. Обычно эта процедура используется, когда такие функции, как восстановление данных после ошибок и упорядочивание данных, выполняются протоколами вышележащих уровней, поэтому нет нужды дублировать их на уровне LLC.

Процедура с установлением соединений и подтверждением LLC2 дает пользователю возможность установить логическое соединение перед началом передачи любого блока данных и, если это требуется, выполнить процедуры восстановления после ошибок и упорядочивание потока этих блоков в рамках установленного соединения. Протокол LLC2 во многом аналогичен протоколам семейства HDLC (LAP-B, LAP-D, LAP-M), которые применяются в глобальных сетях для обеспечения надежной передачи кадров на зашумленных линиях. Протокол LLC2 работает в режиме скользящего окна (см. подраздел «Обнаружение и коррекция ошибок» раздела «Особенности протоколов канального уровня» в главе 5).

В некоторых случаях (например, при использовании сетей в системах реального времени, управляющих промышленными объектами), когда, с одной стороны, временные издержки установления логического соединения перед отправкой данных неприемлемы, а, с другой стороны, подтверждение о корректности приема переданных данных необходимо, оба рассмотренных выше режима не подходят. Для такого рода ситуаций предусмотрена дополнительная процедура LLC3, которая является компромиссом между LLC1 и LLC2, так как она не устанавливает логического соединения, но подтверждает получение данных.

Какой из трех режимов работы уровня LLC будет использован, зависит от стратегии разработчиков конкретного стека протоколов. Например, в стеке TCP/IP, в котором задачу обеспечения надежной доставки выполняет протокол TCP, уровень LLC всегда работает в режиме LLC1, выполняя простую работу извлечения пакета из кадра и направления его одному из вышеперечисленных протоколов — IP, ARP, RARP. Аналогично используется уровень LLC стеком IPX/SPX.

А вот стек Microsoft/IBM, основанный на протоколе NetBIOS/NetBEUI, часто использует режим LLC2 с установлением соединений. Это происходит тогда, когда сам протокол NetBIOS/NetBEUI должен работать в режиме с восстановлением потерянных и искаженных данных. В этом случае эта работа переносится уровню LLC2. Если же протокол NetBIOS/NetBEUI работает в дейтаграммном режиме, то протокол LLC переводится в режим LLC1.

Режим LLC2 используется также стеком протоколов SNA в том случае, когда на нижнем уровне применяется технология Token Ring.

Структура кадров LLC и процедура LLC2

По своему назначению все кадры уровня LLC (называемые в стандарте 802.2 блоками данных — Protocol Data Unit, PDU) подразделяются на три типа — информационные, управляющие и ненумерованные.

- Информационные кадры. (Information) предназначены для передачи информации в процедурах с установлением логического соединения LLC2 и должны обязательно содержать поле информации. В процессе передачи информационных блоков осуществляется их нумерация в режиме скользящего окна.
- Управляющие кадры (Supervisory) предназначены для передачи команд и ответов в процедурах с установлением логического соединения LLC2, в том числе запросов на повторную передачу искаженных информационных блоков.

- Ненумерованные кадры (Unnumbered) предназначены для передачи ненумерованных команд и ответов выполняющих в процедурах без установления логического соединения передачу информации, идентификацию и тестирование уровня LLC, а в процедурах с установлением логического соединения LLC2 — установление и разъединение логического соединения, а также информирование об ошибках.

Все типы кадров уровня LLC имеют единый формат.

Адрес точки входа службы назначения. (DSAP)	Адрес точки входа, службы источника (SSAP)	Управляющее поле (Control)	Данные (Data)
--	---	-------------------------------	---------------

В соответствии с многоуровневой структурой протоколов стандартов IEEE 802, кадр LLC вкладывается в кадр уровня MAC: кадр Ethernet, Token Ring, FDDI и т. д. Кадр LLC содержит поле данных и заголовок, который состоит из трех полей:

- адрес точки входа службы назначения (Destination Service Access Point, DSAP);
- адрес точки входа службы источника (Source Service Access Point, SSAP);
- управляющее поле (Control).

Поле данных кадра LLC предназначено для передачи по сети пакетов протоколов вышележащих уровней — сетевых протоколов IP, IPX, AppleTalk, DECnet, в редких случаях — прикладных протоколов, когда те вкладывают свои сообщения непосредственно в кадры канального уровня. Поле данных может отсутствовать в управляющих кадрах и некоторых ненумерованных кадрах.

Адресные поля DSAP и SSAP занимают по одному байту. Они позволяют указать, какая служба верхнего уровня пересыпает данные с помощью этого кадра. Программному обеспечению узлов сети при получении кадров канального уровня необходимо распознать, какой протокол вложил свой пакет в поле данных поступившего кадра, чтобы передать извлеченный из кадра пакет соответствующему протоколу верхнего уровня для последующей обработки. Для идентификации этих протоколов вводятся так называемые адреса точки входа службы (Service Access Point, SAP). Значения адресов SAP приписываются протоколам в соответствии со стандартом 802.2. Например, для протокола IP значение SAP равно 0x6¹, для протокола NetBIOS — 0xF0. В тех случаях, когда служба имеет единственную точку входа, адреса DSAP и SSAP совпадают. Например, если в кадре LLC значения DSAP и SSAP содержат код протокола IPX, то обмен кадрами осуществляется между двумя IPX-модулями, выполняющими и разных узлах. Но в некоторых случаях в кадре LLC указываются различающиеся значения DSAP и SSAP. Это возможно только в тех случаях, когда служба имеет несколько адресов SAP, что может быть использовано протоколом узла отправителя в специальных целях, например для уведомления узла получателя о переходе протокола-отправителя в некоторый специфический режим работы. Этим свойством протокола LLC часто пользуется протокол NetBEUI.

(Здесь и далее префикс 0x показывает, что число представлено в шестнадцатеричном формате, так, 0x6 означает шестнадцатеричное число G, 0xF0 — шестнадцатеричное число F0.)

Другими словами, с помощью значений DSAP и SSAP в узле осуществляется демультиплексирование кадров, поступающих из сети. То есть общий поток кадров, предназначенных для конкретного узла, распределяется далее по различным протоколам сетевого уровня. Значения DSAP и SSAP представляют собой адреса этих протоколов внутри узла. Функция демультиплексирования выполняется каждым уровнем протоколов при передаче пакетов вверх по стеку внутри узла, и в заголовке протокола каждого уровня имеется адресная информация, аналогичная по назначению полям DSAP и SSAP. Разделение адреса сервиса на две части не является обязательным, скорее это исключение, взятое на вооружение разработчиками протокола LLC. Например, в протоколах IP, UDP и TCP для этих целей используется одно поле.

Поле управления (один или два байта) имеет сложную структуру при работе в режиме LLC2 и достаточно простую структуру при работе в режиме LLC1 (рис. 7.2).

Тип кадра	Разряды поля управления															
	1	2	3	4	5	6	7	8	9..10	11	12	13	14	15	16	
	0			N(S)					P/F					N(R)		
	1	0	S	—	—	—	—							N(R)		
Информационный (Information) управляющий (Supervisory) Ненумерованный (Unnumbered)	1	1	M	P/F	M											

Рис. 7.2. Структура поля управления

В режиме LLC1 используется только один тип кадра — ненумерованный. У этого кадра поле управления имеет длину в один байт. Все подполя поля управления ненумерованных кадров принимают нулевые значения, так что значимыми остаются только первые два бита поля, используемые как признак типа кадра. Учитывая, что в протоколе Ethernet, при записи реализован обратный (начиная со старшего бита) порядок бит в байте, то поле управления кадра LLC1, вложенного в кадр протокола Ethernet, имеет значение 0x03.

В режиме LLC2 используются все три типа кадров. В этом режиме кадры делятся на команды и ответы на эти команды. Бит P/F (Poll/Final) в командах называется битом Poll и требует, чтобы на команду был дан ответ, а в ответах он называется битом Final и говорит о том, что ответ состоит из одного кадра.

Ненумерованные кадры используются на начальной стадии взаимодействия двух узлов, а именно стадии установления соединения по протоколу LLC2. Поле M ненумерованных кадров определяет несколько типов команд, которыми пользуются два узла на этапе установления соединения. Ниже приведены примеры некоторых команд.

- Установить сбалансированный асинхронный расширенный режим (SABME). Эта команда является запросом на установление соединения. Расширенный режим означает использование двухбайтных полей управления для кадров остальных двух типов.

- Ненумерованное подтверждение (UA). Эта команда служит для подтверждения установления или разрыва соединения.
- Сброс соединения (REST). Запрос на разрыв соединения.

После установления соединения данные и положительные квитанции начинают передаваться в информационных кадрах. Логический канал протокола LLC2 является дуплексным, так что данные могут передаваться в обоих направлениях. Если поток дуплексный, то положительные квитанции на кадры также доставляются в информационных кадрах. Если же потока кадров в обратном направлении нет или же нужно передать отрицательную квитанцию, то используются супервизорные кадры.

В информационных кадрах имеется поле N(S) для указания номера отправленного кадра, а также поле N(R) для указания номера кадра, который приемник ожидает получить от передатчика следующим. При работе протокола LLC2 используется скользящее окно размером в 127 кадров, а для их нумерации циклически используется 128 чисел, от 0 до 127.

Приемник всегда помнит номер последнего кадра, принятого от передатчика, и поддерживает переменную с указанным номером кадра, который он ожидает принять от передатчика следующим. Обозначим его через V(R). Именно это значение передается в поле N(R) кадра, посыпанного передатчику. Если в ответ на этот кадр приемник принимает кадр, в котором номер посланного 'кадра N(S) совпадает с номером ожидаемого кадра V(R), то такой кадр считается корректным (если, конечно, корректна его контрольная сумма). Если приемник принимает кадр с номером N(S), неравным V(R), то этот кадр отбрасывается и посыпается отрицательная квитанция Отказ (REJ) с номером V(R). При приеме отрицательной квитанции передатчик обязан повторить передачу кадра с номером V(R), а также всех кадров с большими номерами, которые он уже успел отослать, пользуясь механизмом скользящего окна в 127 кадров.

В состав супервизорных кадров входят следующие:

- отказ (REJacl:);
- приемник не готов (Receiver Not Ready, RNR);
- приемник готов (Receiver Ready, RR~).

Команда RR с номером N(R) часто используется как положительная квитанция, когда поток данных от приемника к передатчику отсутствует, а команда RNR — для замедления потока кадров, поступающих на приемник. Это может быть необходимо, если приемник не успевает обработать поток кадров, присылаемых ему с большой скоростью за счет механизма скользящего окна. Получение кадра RNR требует от передатчика полной приостановки передачи до получения кадра RR. С помощью этих кадров осуществляется управление потоком данных, что особенно важно для коммутируемых сетей, в которых нет разделяемой среды, автоматически тормозящей работу передатчика из-за того, что новый кадр нельзя передать, пока приемник не закончил прием предыдущего.

Нетрудно заметить, что в режиме LLC1 данный протокол выполняет только одну функцию — демультиплексирование потока кадров, поступающих из сети, то есть распределения их по протоколам сетевого уровня и соответствия с адресом точки входа сервиса DSAP.

Выводы

- Протокол LLC обеспечивает для технологий локальных сетей нужное качество транспортной службы, передавая свои кадры либо дейтаграммным способом, либо с помощью процедур с установлением соединения и восстановлением кадров.
- LLC предоставляет верхним уровням три типа процедур; процедуре без установления соединения и без подтверждения (LLC1); процедуре с установлением соединения и подтверждением (LLC2); процедуре без установления соединения, но с подтверждением (LLC3).
- Логический канал протокола LLC2 является дуплексным, так что данные могут передаваться в обоих направлениях.
- Протокол LLC в режиме LLC2 использует алгоритм скользящего окна.
- Протокол LLC с помощью управляющих кадров имеет возможность регулировать поток данных, поступающих от узлов сети. Это особенно важно для коммутируемых сетей, в которых нет разделяемой среды, автоматически тормозящей работу передатчика при высокой загрузке сети.
- В режиме LLC1 единственной функцией протокола LLC является демультиплексирование потока кадров, поступающих из сети, то есть распределения их по протоколам сетевого уровня в соответствии с адресом точки входа сервиса DSAP.

Технология Ethernet

Ethernet — это самый распространенный на сегодняшний день стандарт локальных сетей. Общее количество сетей, работающих по протоколу Ethernet в настоящее время, оценивается в несколько миллионов.

Когда говорят Ethernet, то под этим обычно понимают любой из вариантов этой технологии. В более узком смысле Ethernet — это сетевой стандарт, основанный на экспериментальной сети Ethernet Network, которую фирма Xerox разработала и реализовала в 1975 году. Метод доступа был опробован еще раньше: во второй половине 60-х годов в радиосети Гавайского университета использовались различные варианты случайногo доступа к общей радиосреде, получившие общее название Aloha. В 1980 году фирмы DEC, Intel и Xerox совместно разработали и опубликовали стандарт Ethernet версии II для сети, построенной на основе коаксиального кабеля. Эту последнюю версию фирменного стандарта Ethernet называют стандартом Ethernet DIX, или Ethernet II.

На основе стандарта Ethernet DIX был разработан стандарт IEEE 802.3, который во многом совпадает со своим предшественником, но некоторые различия все же имеются. В то время как в стандарте IEEE 802.3 функции протокола разделены на уровни MAC и LLC, в оригинальном стандарте Ethernet они объединены в единый канальный уровень. В Ethernet DIX определяется протокол тестирования конфигурации (Ethernet Configuration Test Protocol), который отсутствует в IEEE 802.3.

вует в IEEE 802.3. Несколько отличается и формат кадра, хотя минимальные и максимальные размеры кадров в этих стандартах совпадают. Часто для того, чтобы отличить стандарт Ethernet, определенный IEEE, и фирменный стандарт Ethernet DIX, первый называют технологией 802.3, а за фирменным стандартом оставляют название Ethernet без дополнительных обозначений.

В зависимости от типа физической среды стандарт IEEE 802.3 имеет различные модификации — 10Base-5, 10Base-2, 10Base-T, 10Base-FL, 10Base-FB.

В 1995 году был принят стандарт Fast Ethernet, который во многом не является самостоятельным стандартом, о чем говорит и тот факт, что его описание просто является дополнительным разделом к основному стандарту 802.3 — разделом 802.3u. Аналогично, принятый в 1998 году стандарт Gigabit Ethernet описан в разделе 802.3z основного документа.

Для передачи двоичной информации по кабелю для всех вариантов физического уровня технологии Ethernet, обеспечивающих пропускную способность 10 Мбит/с, используется манчестерский код. В более скоростных версиях Ethernet применяются более эффективные в отношении полосы пропускания избыточные логические коды.

Все виды стандартов Ethernet (в том числе Fast Ethernet и Gigabit Ethernet) используют один и тот же метод разделения среды передачи данных — метод CSMA/CD.

Метод доступа CSMA/CD

В сетях Ethernet используется метод доступа к среде передачи данных, называемый методом коллективного доступа с опознаванием несущей и обнаружением коллизий (carrier-sense-multiply-access with collision detection, CSMA/ CD). Этот метод применяется исключительно в сетях с логической общей шиной (к которым относятся и радиосети, породившие этот метод). Все компьютеры такой сети имеют непосредственный доступ к общей среде, поэтому она может быть использована для передачи данных между любыми двумя узлами сети. Одновременно все компьютеры сети имеют возможность немедленно (с учетом задержки распространения сигнала по физической среде) получить данные, которые любой из компьютеров начал передавать в общую среду.

Простота схемы подключения — один из факторов, определивших успех стандарта Ethernet. Говорят, что среда, к которой подключены все станции, работает в режиме коллективного доступа (Multiply Access, MA).

MAC-адреса

На MAC-уровне для идентификации сетевых интерфейсов узлов сети используются регламентированные стандартом IEEE 802.3 уникальные 6-байтовые адреса, называемые MAC-адресами. Обычно MAC-адрес записывают в виде шести пар шестнадцатеричных цифр, разделенных тире или двоеточиями, например 11-A0-17-3D-BC-01. Каждый сетевой адаптер имеет, по крайней мере, один

MAC-адрсс.

Помимо отдельных интерфейсов, MAC-адрес может определять группу интерфейсов или даже все интерфейсы сети. Первый бит (младший.) старшего байта адреса назначения является признаком того, является ли адрес индивидуальным или групповым. Если он равен 0, то адрес является индивидуальным (unicast), а если 1, то это групповой адрес (multicast). Групповой адрес связан только с теми интерфейсами, которые сконфигурированы (например, вручную или автоматически по запросу вышестоящего уровня) как члены группы, номер которой указан в групповом адресе. Если сетевой интерфейс включен в группу, то наряду с уникальным MAC-адресом с ним ассоциируется еще один адрес — групповой. В частном случае, если групповой адрес состоит из всех единиц, то есть имеет шестнадцатеричное представление 0xFFEFFFFFEFFF, он идентифицирует все узлы сети и называется широковещательным адресом, (broadcast). Второй бит старшего байта адреса определяет способ назначения адреса — централизованный или локальный. Если этот бит равен 0 (что бывает почти всегда в стандартной аппаратуре Ethernet), то адрес назначен централизованно, с помощью комитета IEEE. Комитет IEEE распределяет между производителями оборудования так называемые организационно уникальные идентификаторы (Organizationally Unique Identifier, OUI). Каждый производитель помещает выделенный ему идентификатор в три старших байта адреса (например, идентификатор 0x0020AE определяет компанию ЗСОМ, а 0x00000C — Cisco). За уникальность младших трех байтов адреса отвечает производитель оборудования. Двадцать четыре бита, отводимые производителю для адресации интерфейсов его продукции, позволяют выпустить 16 миллионов интерфейсов под одним идентификатором организации. Уникальность централизованно распределенных адресов распространяется на все основные технологии локальных сетей — Ethernet, Token Ring, FDDI и т. д. Локально администрируемые адреса назначаются администратором сети, в обязанности которого входит обеспечение их уникальности.

Этапы доступа к среде

Предполагая для простоты изложения, что каждый узел (станция) имеет только один сетевой интерфейс, рассмотрим алгоритм доступа к разделяемой среде. Чтобы получить возможность передавать кадр, интерфейс-отправитель должен убедиться, что разделяемая среда свободна. Это достигается прослушиванием основной гармоники сигнала, которая также называется несущей частотой (carrier-sense, CS). Признаком незанятости среды является отсутствие на ней несущей частоты, которая при манчестерском способе кодирования равна 5-10 МГц, в зависимости от последовательности единиц и нулей, передаваемых в данный момент.

Если среда свободна, то узел имеет право начать передачу кадра, который показан на рис. 7.3 первым. Узел 1 обнаружил, что среда свободна, и начал передавать свой кадр. В классической сети Ethernet на коаксиальном кабеле сигналы передатчика узла 1 распространяются в обе стороны, так что все узлы сети их

получают. Кадр данных всегда сопровождается преамбулой (preamble), которая состоит из 7 байтов, состоящих из значений 10101010, и 8-го байта, равного 10101011. Последний байт носит название ограничителя начала кадра — Start of Frame. Преамбула нужна для вхождения приемника в побитовый и побайтовый синхронизм с передатчиком.



Рис. 7.3. Метод случайного доступа CSMA/CD

Все станции, подключенные к кабелю, могут распознать факт передачи кадра, и та станция, которая узнает собственный адрес в заголовках кадра, записывает его содержимое в свой внутренний буфер, обрабатывает полученные данные, передаст их вверх по своему стеку, а затем посыпает по кабелю кадр-ответ. Адрес станции-источника содержится в исходном кадре, поэтому станция-получатель знает, кому нужно послать ответ.

Узел 2 во время передачи кадра узлом 1 также пытался начать передачу своего кадра, однако обнаружил, что среда занята — на ней присутствует несущая частота, — поэтому узел 2 вынужден ждать, пока узел 1 не прекратит передачу кадра.

После окончания передачи кадра все узлы сети обязаны выдержать технологическую паузу (Inter Packet Gap) в 9,6 мкс. Эта пауза, называемая также межкадровым интервалом, нужна для приведения сетевых адаптеров в исходное состояние, а также для предотвращения монопольного захвата среды одной станцией. После окончания технологической паузы узлы имеют право начать передачу своего кадра, так как среда свободна. Из-за задержек распространения сигнала по кабелю не все узлы строго одновременно фиксируют факт окончания передачи кадра узлом 1.

В приведенном примере узел 2 дождался окончания передачи кадра узлом 1, сделал паузу и 9,6 мкс и начал передачу своего кадра.

Возникновение коллизии

При описанном подходе возможна ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Механизм прослушивания среды и пауза между кадрами не гарантируют исключения такой ситуации, когда две или более станции одновременно решают, что среда свободна, и начинают передавать свои кадры. Говорят, что при этом происходит коллизия (collision).

sion), так как содержимое обоих кадров сталкивается на общем кабеле и происходит искажение информации — методы кодирования, используемые в Ethernet, не позволяют выделять сигналы каждой станции из общего сигнала.

ПРИМЕЧАНИЕ

Заметим, что этот факт отражен в составляющей «Base(band)», присутствующей в названиях всех физических протоколов технологии Ethernet (например, 10Base-2, 10Base-T и т. п.). Термин «Base band network» означает сеть с немодулированной передачей, в которой сообщения пересылаются в цифровой форме по единственному каналу, без частотного разделения.

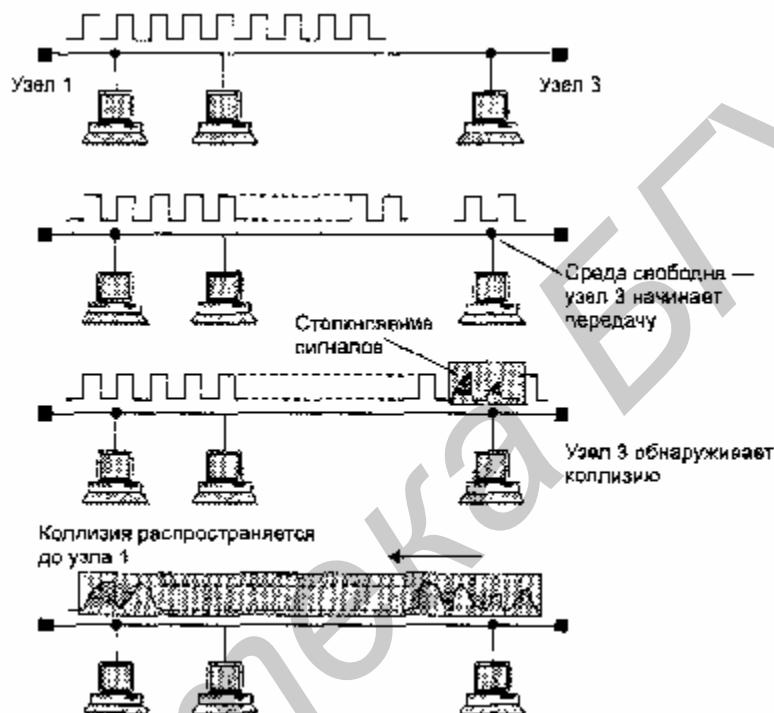


Рис. 7.4. Схема возникновения и распространения коллизии

Коллизия — это нормальная ситуация в работе сетей Ethernet. В примере на рис. 7.4 коллизию породила одновременная передача данных узлами 3 и 1. Для возникновения коллизии не обязательно, чтобы несколько станций начали передачу абсолютно одновременно, такая ситуация маловероятна. Гораздо вероятней, что коллизия возникает из-за того, что один узел начинает передачу раньше другого, но до второго узла сигналы первого просто не успевают дойти к тому времени, когда второй узел решает начать передачу своего кадра. То есть коллизии — это следствие распределенного характера сети.

Чтобы корректно обработать коллизию, все станции одновременно наблюдают за возникающими на кабеле сигналами. Если передаваемые и наблюдаемые сигналы отличаются, то фиксируется обнаружение коллизии (collision detection, CD). Для увеличения вероятности скорейшего обнаружения коллизии всеми станциями сети станция, которая обнаружила коллизию, прерывает передачу своего кадра (в произвольном месте, возможно, и не на границе байта) и усиливает ситуацию коллизии посылкой в сеть специальной последовательности из 32 бит, называемой *jam-последовательностью*.

После этого обнаружившая коллизию передающая станция обязана прекратить передачу и сделать паузу в течение короткого случайного интервала времени.

Затем она может снова предпринять попытку захвата среды и передачи кадра. Случайная пауза выбирается по следующему алгоритму:
Пауза = $L \times$ (интервал отсрочки).

Здесь интервал отсрочки равен 512 битовым интервалам (в технологии Ethernet, принято все интервалы измерять в битовых интервалах; битовый интервал обозначается как b_t и соответствует времени между появлением двух последовательных бит данных на кабеле; для скорости 10 Мбит/с величина битового интервала равна 0,1 мкс или 100 нс). L представляет собой целое число, выбранное с равной вероятностью из диапазона $[0, 2^N]$, где N — номер повторной попытки передачи данного кадра: 1, 2,..., 10.

После 10-й попытки интервал, из которого выбирается пауза, не увеличивается. Таким образом, случайная пауза может принимать значения от 0 до 52,4 мс. Если 16 последовательных попыток передачи кадра вызывают коллизию, то передатчик должен прекратить попытки и отбросить этот кадр.

Из описания метода доступа видно, что он носит вероятностный характер, и вероятность успешного получения в свое распоряжение общей среды зависит от загруженности сети, то есть от интенсивности возникновения в станциях потребности в передаче кадров. При разработке этого метода в конце 70-х годов предполагалось, что скорость передачи данных в 10 Мбит/с очень высока по сравнению с потребностями компьютеров во взаимном обмене данными, поэтому загрузка сети будет всегда небольшой. Это предположение остается иногда справедливым и по сей день, однако уже появились приложения, работающие в реальном масштабе времени с мультимедийной информацией, которые очень загружают сегменты Ethernet. При этом коллизии возникают гораздо чаще. При значительной интенсивности коллизий полезная пропускная способность сети Ethernet резко падает, так как сеть почти постоянно занята повторными попытками передачи кадров. Для уменьшения интенсивности возникновения коллизий нужно либо снизить трафик, сократив, например, количество узлов в сегменте или заменив приложения, либо повысить скорость протокола, например перейти на Fast Ethernet.

Следует отметить, что метод доступа CSMA/CD вообще не гарантирует станции, что она когда-либо сможет получить доступ к среде. Конечно, при небольшой загрузке сети вероятность такой неудачи невелика, но при коэффициенте использования сети, приближающемся к 1, невозможность получения доступа становится очень вероятной. Этот недостаток метода случайного доступа — плата за его чрезвычайную простоту, которая сделала технологию Ethernet самой недорогой. Другие методы доступа — маркерный доступ сетей Token Ring и FDDI, метод Demand Priority сетей 100VG-AnyLAN — свободны от этого недостатка.

Время двойного оборота и распознавание коллизий

Четкое распознавание коллизий всеми станциями сети является необходимым условием корректной работы сети Ethernet. Если какая-либо передающая станция не распознает коллизию и решит, что кадр данных ею передан верно, то

этот кадр данных будет утерян. Из-за наложения сигналов при коллизии информация кадра исказится, и он будет отбракован принимающей станцией (возможно, из-за несовпадения контрольной суммы). Скорее всего, искаженная информация будет повторно передана каким-либо протоколом верхнего уровня, например транспортным или прикладным, работающим с установлением соединения. Но повторная передача сообщения протоколами верхних уровней произойдет через значительно более длительный интервал времени (иногда даже по прошествии нескольких секунд) по сравнению с микросекундными интервалами, которыми оперирует протокол Ethernet. Поэтому если коллизии не будут надежно распознаваться узлами сети Ethernet, то это приведет к заметному снижению полезной пропускной способности данной сети.

Для надежного распознавания коллизий должно выполняться следующее соотношение:

$$T_{\min} \geq PDV$$

Здесь T_{\min} — время передачи кадра минимальной длины, а PDV — время, за которое сигнал коллизии успевает распространиться до самого дальнего узла сети. Так как в худшем случае сигнал должен пройти дважды между наиболее удаленными друг от друга станциями сети (и одну сторону проходит неискаженный сигнал, а на обратном пути распространяется уже искаженный коллизией сигнал), то это время называется временем двойного оборота (Path Delay Value, PDV),

При выполнении этого условия передающая станция должна успеть обнаружить коллизию, которую вызвал переданный ее кадр, еще до того, как она закончит передачу этого кадра.

Очевидно, что выполнение этого условия зависит, с одной стороны, от длины минимального кадра и пропускной способности сети, а с другой стороны, от длины кабельной системы сети и скорости распространения сигнала в кабеле (для разных типов кабеля эта скорость несколько отличается).

Вес параметры протокола Ethernet подобраны таким образом, чтобы при нормальной работе узлов сети коллизии всегда четко распознавались. При выборе параметров, конечно, учитывалось и приведенное выше соотношение, связывающее между собой минимальную длину кадра и максимальное расстояние между станциями в сегменте сети.

В стандарте Ethernet принято, что минимальная длина поля данных кадра составляет 46 байт (что вместе со служебными полями дает минимальную длину кадра 64 байт, а вместе с преамбулой — 72 байт или 576 бит), Отсюда может быть определено ограничение на расстояние между станциями.

Итак, в 10-мегабитном стандарте Ethernet время передачи кадра минимальной длины равно 575 битовых интервалов, следовательно, время двойного оборота должно быть меньше 57,5 мкс. Расстояние, которое сигнал может пройти за это время, зависит от типа кабеля и для толстого коаксиального кабеля равно примерно 13 280 м. Учитывая, что за это время сигнал должен пройти по линии связи дважды, расстояние между двумя узлами не должно быть больше 6635 м. В стандарте величина этого расстояния выбрана существенно меньше, с учетом других, более строгих ограничений.

Одно из таких ограничений связано с предельно допустимым затуханием сигнала. Для обеспечения необходимой мощности сигнала при его прохождении между наиболее удаленными друг от друга станциями сегмента кабеля максимальная длина непрерывного сегмента толстого коаксиального кабеля с учетом вносимого им затухания выбрана в 500 м. Очевидно, что на кабеле в 500 м условия распознавания коллизий будут выполняться с большим запасом для кадров любой стандартной длины, в том числе и 72 байт (время двойного оборота по кабелю 500 м составляет всего 43,3 битовых интервала), Поэтому минимальная длина кадра могла бы быть установлена еще меньше. Однако разработчики технологии не стали уменьшать минимальную длину кадра, имея в виду многосегментные сети, которые строятся из нескольких сегментов, соединенных повторителями.

Повторители увеличивают мощность передаваемых с сегмента на сегмент сигналов, в результате затухание сигналов уменьшается и можно использовать сеть гораздо большей длины, состоящую из нескольких сегментов. В коаксиальных реализациях Ethernet разработчики ограничили максимальное количество сегментов в сети пятью, что, в свою очередь, ограничивает общую длину сети 2500 метрами. Даже в такой многосегментной сети условие обнаружения коллизий по-прежнему выполняется с большим запасом (сравним полученное из условия допустимого затухания расстояние в 2500 м с вычисленным выше максимально возможным по времени распространения сигнала расстоянием 6635 м). Однако в действительности временной запас существенно меньше, поскольку в многосегментных сетях сами повторители вносят в распространение сигнала дополнительную задержку в несколько десятков битовых интервалов. Естественно, небольшой запас был сделан также для компенсации отклонений параметров кабеля и повторителей.

В результате учета всех этих и некоторых других факторов было тщательно подобрано соотношение между минимальной длиной кадра и максимально возможным расстоянием между станциями сети, которое обеспечивает надежное распознавание коллизий. Это расстояние называют также максимальным диаметром сети.

С увеличением скорости передачи кадров, что имеет место в новых стандартах, базирующихся на том же методе доступа CSMA/CD, например Fast Ethernet, максимальное расстояние между станциями сети уменьшается пропорционально увеличению скорости передачи. В стандарте Fast Ethernet оно составляет около 210 м, а в стандарте Gigabit Ethernet оно было бы ограничено 25 м, если бы разработчики стандарта не предприняли некоторых мер по увеличению минимального размера пакета.

В табл. 7.1 приведены значения основных параметров процедуры передачи кадра стандарта 802.3, которые не зависят от реализации физической среды. Важно отметить, что каждый вариант физической среды технологии Ethernet добавляет к этим ограничениям свои, часто более строгие ограничения, которые также должны выполняться и которые будут рассмотрены ниже.

Таблица 7.1. Параметры уровня MAC Ethernet

Параметры	Значения
Битовая скорость	10 Мбит/с
Интервал оторочки	512 битовых интервала
Межкадровый интервал (IPG)	9,6 мкс
Максимальное число попыток передачи	16
Максимальное число возрастания диапазона паузы	50
Длина ям-последовательности	32 бита
Максимальная длина кадра (без преамбулы)	1518 байт
Минимальная длина кадра (без преамбулы)	64 байт (512 бит)
Длина преамбулы	64 бит
Минимальная длина случайной паузы после коллизии	0 битовых интервалов
Максимальная длина случайной паузы после коллизии	524 000 битовых интервалов
Максимальное расстояние между станциями сети	2500 м
Максимальное число станций в сети	1624

Максимальная производительность сети Ethernet

Количество обрабатываемых кадров Ethernet в секунду часто указывается производителями мостов/коммутаторов и маршрутизаторов как основная характеристика производительности этих устройств. В свою очередь, интересно знать чистую максимальную пропускную способность сегмента Ethernet в кадрах в секунду в идеальном случае, когда в сети нет коллизий и нет дополнительных задержек, вносимых мостами и маршрутизаторами. Такой показатель помогает оценить требования к производительности коммуникационных устройств, так как в каждый порт устройства не может поступать больше кадров в единицу времени, чем позволяет это сделать соответствующий протокол. Для коммуникационного оборудования наиболее тяжелым режимом является обработка потока кадров минимальной длины. Это объясняется тем, что на обработку каждого кадра, независимо от его длины, мост, коммутатор или маршрутизатор тратит примерно равное время, связанное с просмотром таблицы продвижения пакета, формированием нового кадра (для маршрутизатора) и т. п. Количество же кадров, поступающих на устройство в единицу времени, естественно, является максимальным при их минимальной длине.

Другая характеристика производительности коммуникационного оборудования — количество передаваемых битов в секунду — используется реже, так как она не говорит о том, какого размера кадры при этом обрабатывало устройство, а на кадрах максимального размера достичь высокой производительности, измеряемой в битах в секунду, гораздо легче.

Используя параметры, приведенные в табл. 7.1, рассчитаем максимальную производительность сегмента Ethernet в таких единицах, как число переданных кадров (пакетов) минимальной длины в секунду.

ПРИМЕЧАНИЕ

При указании пропускной способности сетей термины кадр и пакет обычно используются как синонимы. Соответственно, аналогичными являются и единицы измерения производительности frames-per-second (fps) и packets-per-second (pps).

Для расчета максимального количества кадров минимальной длины, проходящих по сегменту Ethernet, заметим, что размер кадра минимальной длины вместе с преамбулой составляет 72 байт или 576 бит (рис. 7.5), поэтому на его передачу затрачивается 57,5 мкс. Прибавив межкадровый интервал в 9,6 мкс, получаем, что период следования кадров минимальной длины составляет 67,1 мкс. Отсюда максимально возможная пропускная способность сегмента Ethernet составляет 14 880 кадр/с.

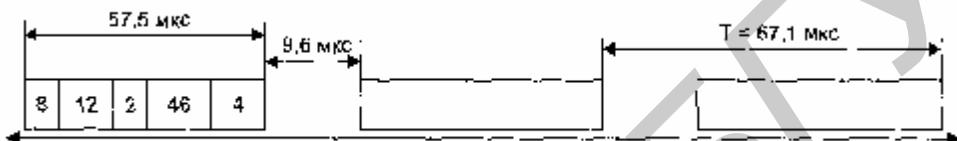


Рис. 7.5. К расчету пропускной способности протокола Ethernet

Естественно, что наличие в сегменте нескольких узлов снижает эту величину за счет ожидания доступа к среде, а также за счет коллизий, приводящих к необходимости повторной передачи кадров.

Кадры максимальной длины технологии Ethernet имеют поле длины 1500 байт, что вместе со служебной информацией дает 1518 байт, а с преамбулой составляет 1526 байт, или 12208 бит. Максимально возможная пропускная способность сегмента Ethernet для кадров максимальной длины составляет 813 кадр/с. Очевидно, что при работе с большими кадрами нагрузка на мосты, коммутаторы и маршрутизаторы довольно ощутимо снижается.

Теперь рассчитаем, какой максимальной полезной пропускной способностью, измеряемой в битах в секунду, обладают сегменты Ethernet при использовании кадров разного размера.

Под полезной пропускной способностью протокола понимается скорость передачи пользовательских данных, которые переносятся полем данных кадра. Эта пропускная способность всегда меньше номинальной битовой скорости протокола Ethernet за счет нескольких факторов:

- служебной информации кадра;
- межкадровых интервалов (IPG); □ ожидания доступа к среде.

Для кадров минимальной длины полезная пропускная способность равна:

$$C_p = 14880 \times 46 \times 8 = 5,48 \text{ Мбит/с.}$$

Это несколько меньше 10 Мбит/с, но следует учесть, что кадры минимальной длины используются в основном для передачи квитанций, так что к передаче собственно данных файлов эта скорость имеет небольшое отношение.

Для кадров максимальной длины полезная пропускная способность равна:

$$C_p - 813 \times 1500 \times 8 = 9,76 \text{ Мбит/с.}$$

Это весьма близко к номинальной скорости протокола.

Еще раз подчеркнем, что такой скорости можно достигнуть только в том слу-

чае, когда двум взаимодействующим узлам в сети Ethernet другие узлы не мешают, что бывает крайне редко.

При использовании кадров среднего размера с полем данных в 512 байт пропускная способность сети составляет 9,29 Мбит/с, что тоже достаточно близко к предельной пропускной способности в 10 Мбит/с.

ВНИМАНИЕ

Отношение текущей пропускной способности сети к ее максимальной пропускной способности называется коэффициентом использования сети (network utilization). При этом при определении текущей пропускной способности принимается во внимание передача по сети любой информации, как пользовательской, так и служебной. Коэффициент использования является важным показателем для технологий разделяемых сред, так как при случайном характере метода доступа высокое значение коэффициента часто говорит о низкой полезной пропускной способности сети (то есть скорости передачи пользовательских данных) — слишком много времени узлы тратят на процедуру получения доступа и повторные передачи кадров после коллизий.

При отсутствии коллизий и ожидания доступа коэффициент использования сети зависит от размера поля данных кадра и имеет максимальное значение 0,976 при передаче кадров максимальной длины. Очевидно, что в реальной сети Ethernet среднее значение коэффициента использования сети может значительно отличаться от этой величины. Более сложные случаи определения пропускной способности сети с учетом ожидания доступа и отработки коллизий будут рассмотрены ниже.

Форматы кадров технологии Ethernet

Стандарт технологии Ethernet, определенный в документе IEEE 802,3, дает описание единственного формата кадра уровня MAC. Так как в кадр уровня MAC должен вкладываться кадр уровня LLC, описанный в документе IEEE 802.2, то по стандартам IEEE в сети Ethernet может использоваться только единственный вариант кадра канального уровня, заголовок которого является комбинацией заголовков подуровней MAC и LLC.

Тем не менее на практике в сетях Ethernet на канальном уровне используются кадры 4 различных форматов (типов). Это связано с длительной историей развития технологии Ethernet, включающей период существования до принятия стандартов IEEE 802, когда подуровень LLC не выделялся из общего протокола и, соответственно, заголовок LLC не применялся.

Консорциум трех фирм Digital, Intel и Xerox в 1980 году представил на рассмотрение комитету 802.3 свою фирменную версию стандарта Ethernet (в которой был, естественно, описан определенный формат кадра) в качестве проекта международного стандарта, но комитет 802.3 принял стандарт, отличающийся — в некоторых деталях от предложения DIX. Отличия касались и формата кадра, что породило существование двух различных типов кадров в сетях Ethernet.

Еще один формат кадра появился в результате усилий компании Novell по ус-

корению работы своего стека протоколов в сетях Ethernet.

И наконец, четвертый формат кадра стал результатом деятельности комитета 802.2 по приведению предыдущих форматов кадров к некоторому общему стандарту и к приданию кадру необходимой гибкости для учета добавления полей или изменения их назначения в будущем.

Различия в форматах кадров могут приводить к несовместимости в работе аппаратуры и сетевого программного обеспечения, рассчитанного на функционирование только с одним стандартом кадра Ethernet. Однако сегодня практически все сетевые адаптеры, драйверы сетевых адаптеров, мосты/коммутаторы и маршрутизаторы умеют работать со всеми используемыми на практике форматами кадров технологии Ethernet, причем распознавание типа кадра выполняется автоматически.

Ниже приводится описание всех четырех типов кадров Ethernet (здесь под кадром понимается весь набор полей, которые относятся к канальному уровню, то есть поля уровня MAC и LLC). Один и тот же тип кадра может иметь разные названия, поэтому ниже для каждого типа кадра приведено по несколько наиболее употребительных названий:

- кадр 802.3/LLC (или кадр 802.3/802.2, или кадр Novell 802.2);
- кадр Raw 802.3 (или кадр Novell 802.3);
- кадр Ethernet DIX (или кадр Ethernet II);
- кадр Ethernet SNAP.

Форматы всех этих четырех типов кадров Ethernet приведены на рис. 7.6.

Кадр 802.3/LLC													
6	6	2	1	1	1(2)	46-1497 (1496)			4				
DA	SA	L	DSAP	SSAP	Control	Data			-CS				
Заголовок LLC													
Кадр Raw 802.3/Novell 802.3													
6	6	2	46-1500				4						
DA	SA	L	Data					FCS					
Кадр Ethernet DIX (II)													
6	6	2	46-1500				4						
DA	SA	T	Data					FCS					
Кадр Ethernet SNAP													
6	6	2	1	1	1	3	2	46-1492	4				
DA	SA	L	DSAP	SSAP	Control	OUI	T	Data	FCS				
AA	AA	03	000000										
Заголовок LLC				Заголовок SNAP									

Рис. 7.6. Форматы кадров Ethernet

Кадр 802.3/LLC

Заголовок кадра 802.3/LLC является результатом объединения полей заголовков кадров, определенных в стандартах IEEE 802.3 и 802.2.

Стандарт 802.3 определяет восемь полей заголовка (на рис. 7.6 поле преамбулы и начальный ограничитель кадра не показаны).

- Поле преамбулы (Preamble) состоит из семи синхронизирующих байтов 10101010. При манчестерском кодировании эта комбинация представляется в физической среде периодическим волновым сигналом с частотой 5 МГц.
- Начальный ограничитель кадра (Start-of-frame-delimiter. SFD) состоит из одного байта 10101011. Появление этой комбинации битов является указанием на то, что следующий байт — это первый байт заголовка кадра.
- Адрес назначения (Destination Address, DA) может быть длиной 2 или 6 байт. На практике всегда используются MAC-адреса из 6 байт.
- Адрес источника (Source Address, SA) — это 2- или 6-байтовое поле, содержащее MAC-адрес узла — отправителя кадра. Первый бит адреса всегда имеет значение 0.
- Длина (Length, L) - 2-байтовое поле, определяющее длину поля данных в кадре.
- Поле данных (Data) может содержать от 0 до 1500 байт. Но если длина поля меньше 46 байт, то используется следующее поле — поле заполнения, дополняющее кадр до минимально допустимого значения в 46 байт.

ВНИМАНИЕ

В стандартах IEEE Ethernet младший бит байта изображается в самой левой позиции поля, а старший бит — в самой правой. Этот нестандартный способ отображения порядка следования битов в байте соответствует порядку передачи битов в линию связи передатчиком Ethernet (первым передается младший бит). В стандартах других организаций, например RFC IETF, JTU-T, ISO, используется традиционное представление байта, когда младший бит считается самым правым битом байта, а старший — самым левым. При этом порядок следования байтов остается традиционным. Поэтому при чтении стандартов, опубликованных этими организациями, а также чтении данных, отображаемых на экране операционной системой или анализатором протоколов, значения каждого байта кадра Ethernet нужно зеркально отобразить, чтобы получить правильное представление о значении разрядов этого байта в соответствии с документами IEEE. Например, групповой адрес, имеющийся в нотации IEEE вид 1000 0000 0000 0000 1010 0111 1111 0000 0000 0000 0000 или в шестнадцатеричной записи 80-00-A7-FO-00-00, будет, скорее всего, отображен анализатором протоколов в традиционном виде как 01-00-5E-OF-00-00.

- Поле заполнения (Padding) состоит из такого количества байтов заполнителей, которое обеспечивает минимальную длину поля данных в 46 байт. Это обеспечивает корректную работу механизма обнаружения коллизии. Если длина поля данных достаточна, то поле заполнения в кадре не появляется.
- Поле контрольной суммы (Frame Check Sequence, FCS) состоит из 4 байт, содержащих контрольную сумму. Это значение вычисляется по алгоритму CRC-32. После получения кадра рабочая станция выполняет собственное вычисление контрольной суммы для этого кадра, сравнивает полученное значение со значением поля контрольной суммы и, таким образом, определяет, не искажен ли полученный кадр.

Кадр 802.3 является кадром подуровня MAC, поэтому в соответствии со стандартом 802.2 в его поле данных вкладывается кадр подуровня LLC с удаленны-

ми флагами начала и конца кадра. Формат кадра LLC был описан выше. Так как кадр LLC имеет заголовок длиной 3 (в режиме LLC1) или 4 байт (и режиме LLC2), то максимальный размер ноля данных уменьшается до 1497 или 1496 байт.

Кадр Raw 802.3/Novell 802.3

Кадр Raw 802.3, называемый также кадром Novell 802.3, также представлен на рис. 7.6. Из рисунка видно, что это кадр подуровня MAC стандарта 802.3, но без вложенного кадра подуровня LLC. Компания Novell долгое время не использовала служебные поля кадра LLC в своей операционной системе NetWare благодаря отсутствию необходимости идентифицировать тип информации, вложенной в поле данных, — там всегда находился пакет протокола IPX, долгое время бывшего единственным протоколом сетевого уровня в ОС NetWare. Теперь, когда необходимость идентификации протокола верхнего уровня помнилась, компания Novell стала использовать возможность инкапсуляции в кадр подуровня MAC кадра LLC, то есть использовать стандартные кадры 802.3/LLC. Такой кадр компания обозначает теперь в своих операционных системах как кадр 802.2, хотя он является комбинацией заголовков 802.3 и 802.2.

Кадр Ethernet DIX/Ethernet II

Кадр Ethernet DIX, называемый также кадром Ethernet II, имеет структуру, совпадающую со структурой кадра Raw 802.3 (см. рис. 7.6). Однако 2-байтовое поле Длина (L) кадра Raw 802.3 в кадре Ethernet DIX используется в качестве поля типа протокола. Это поле, теперь получившее название Type (T) или EtherType, предназначено для тех же целей, что и поля DSAP и SSAP кадра LLC — для указания типа протокола верхнего уровня, вложившего свой пакет в поле данных этого кадра.

В то время как коды протоколов в полях SAP имеют длину в один байт, в поле Type для кода протокола отводится два байта. Поэтому один и тот же протокол в поле SAP и поле Type будет кодироваться в общем случае разными числовыми значениями. Например, протокол IP имеет код 2048_{10} (0x0800) для поля EtherType и значение 6 для поля SAP. Значения кодов протоколов для поля EtherType появились раньше значений 5AP; так как фирменная версия Ethernet. DIX существовала до появления стандарта 802.3, и ко времени распространения оборудования 802.3 эти значения уже стали стандартами де-факто для многих аппаратных и программных продуктов. Так как структуры кадров Ethernet DIX и Raw 802.3 совпадают, то поле длины/типа часто в документации обозначают как поле L/T. При этом числовое значение этого поля определяет его смысл: если значение меньше 1500, то это поле Length, а если больше — то Type.

Кадр Ethernet SNAP

Для устранения разнобоя в кодировках типов протоколов, сообщения которых вложены в поле данных кадров Ethernet, комитетом 802.2 была проведена работа по дальнейшей стандартизации кадров Ethernet. В результате появился кадр Ethernet SNAP (SubNetwork Access Protocol — протокол доступа к подсетям). Кадр Ethernet SNAP (см. рис. 7.6) представляет собой расширение кадра 802.3/LLC за счет введения дополнительного заголовка протокола SNAP, состоящего из двух полей: OUI и Type. Поле Type состоит из 2-х байтов и повторяет по формату и назначению поле Type кадра Ethernet II (то есть в нем используются те же значения кодов протоколов). Поле OUI (Organizationally Unique Identifier) определяет идентификатор организации, которая контролирует коды протоколов в поле Type. С помощью заголовка SNAP достигнута совместимость с кодами протоколов в кадрах Ethernet II, а также создана универсальная схема кодирования протоколов. Коды протоколов для технологий 802 контролирует организация IEEE, которая имеет идентификатор OUI, равный 000000. Если в будущем потребуются другие коды протоколов для какой-либо новой технологии, для этого достаточно указать другой идентификатор организации, назначающей эти коды, а старые значения кодов останутся в силе (в сочетании с другим идентификатором OUI).

Так как SNAP представляет собой протокол, вложенный в протокол LLC, то в полях DSAP и SSAP записывается код 0xAA, отведенный для протокола SNAP. Поле Control заголовка LLC устанавливается в 0x03, что соответствует использованию ненумерованных кадров.

Заголовок SNAP является дополнением к заголовку LLC, поэтому он допустим не только в кадрах Ethernet, но и в кадрах протоколов других технологий 802.

Например, протокол IP всегда использует структуру заголовков LLC/SNAP при инкапсуляции в кадры всех протоколов локальных сетей: FDDI, Token Ring, 100VG-AnyLAN, Ethernet, Fast Ethernet, Gigabit Ethernet.

Правда, при передаче пакетов IP через сети Ethernet, Fast Ethernet и Gigabit Ethernet протокол IP использует кадры Ethernet DIX.

Использование различных типов кадров Ethernet

Автоматическое распознавание типов кадров Ethernet выполняется достаточно просто. Для кодирования типа протокола в поле EtherType указываются значения, превышающие значение максимальной длины поля данных, равное 1500, поэтому кадры Ethernet II легко отличить от других типов кадров по значению поля L/T. Дальнейшее распознавание типа кадра проводится по наличию или отсутствию нолей LLC. Поля LLC могут отсутствовать только в том случае, если за полем длины идет начало пакета IPX, а именно 2-байтовое поле контрольной суммы пакета, которое всегда заполняется единицами, что дает значение в 255 байт. Ситуация, когда поля DSAP и SSAP одновременно содержат такие значения, возникнуть не может, поэтому наличие двух байтов 255 говорит о том, что это кадр Raw 802.3. В остальных случаях дальнейший анализ проводится в зависимости от значений полей DSAP и SSAP. Если они равны 0xAA, то это кадр Ethernet SNAP, а если нет, то 802.3/LLC.

В табл. 7.2 приведены данные о том, какие типы кадров Ethernet обычно поддерживают реализации популярных протоколов сетевого уровня.

Таблица 7.2. Типы кадров Ethernet, поддерживающие реализации популярных протоколов сетевого уровня

Тип кадра	Сетевые протоколы
Ethernet II	IPX, IP, AppleTalk Phase I
Ethernet 802.3	IPX
Ethernet 802.2	IPX, FTAM
Ethernet SNAP	IPX, IP, AppleTalk Phase II

Спецификации физической среды Ethernet

Исторически первые сети технологии Ethernet были созданы на коаксиальном кабеле диаметром 0,5 дюйма. В дальнейшем были определены и другие спецификации физического уровня для стандарта Ethernet, позволяющие использовать различные среды передачи данных. Метод доступа CSMA/CD и все временные параметры остаются одними и теми же для любой спецификации физической среды технологии Ethernet 10 Мбит/с.

Физические спецификации технологии Ethernet на сегодняшний день включают следующие среды передачи данных.

- 10Base-5 — коаксиальный кабель диаметром 0,5 дюйма, называемый «толстым» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 500 м (без повторителей).
- 10Base-2 — коаксиальный кабель диаметром 0,25 дюйма, называемый «тонким» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 185 м (без повторителей).
- 10Base-T — кабель на основе неэкранированной витой пары (Unshielded Twisted Pair, UTP). Образует звездообразную топологию на основе концентратора. Расстояние между концентратором и конечным узлом — не более 100 м.
- 10Base-F — волоконно-оптический кабель. Топология аналогична топологии стандарта 10Base-T. Имеется несколько вариантов этой спецификации •— FOIRL (расстояние до 1000 м), 10Base-FL (расстояние до 2000 м), 10Base-FB (расстояние до 2000 м).

Число 10 в указанных выше названиях обозначает битовую скорость передачи данных этих стандартов — 10 Мбит/с, а слово «Base» — метод передачи на одной базовой частоте 10 МГц (в отличие от методов, использующих несколько несущих частот, которые называются Broadband — широкополосными). Последний символ в названии стандарта физического уровня обозначает тип кабеля.

Стандарт 10Base-5

Стандарт 10Base-5 в основном соответствует экспериментальной сети Ethernet

фирмы Xerox и может считаться классическим стандартом Ethernet. Он использует в качестве среды передачи данных коаксиальный кабель с волновым сопротивлением 50 Ом, диаметром центрального медного провода 2,17 мм и внешним диаметром около 10 мм («толстый» Ethernet). Такими характеристиками обладают кабели марок RG-8 и RG-11.

Различные компоненты сети, выполненной на толстом коаксиале и состоящей из трех сегментов, соединенных повторителями, показаны на рис. 7.7.

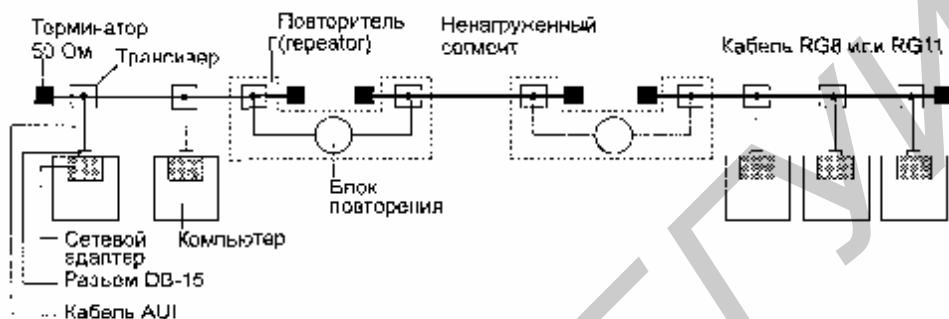


Рис. 7.7. Компоненты физического уровня сети стандарта 10 Base-5, состоящей из трех сегментов

Кабель используется как моноканал для всех станций. Сегмент кабеля максимальной длины 500 м (без повторителей) должен иметь на концах согласующие терминалы («заглушки») сопротивлением 50 Ом, поглощающие распространяющиеся по кабелю сигналы и препятствующие возникновению отраженных сигналов. При отсутствии терминалов в кабеле возникают стоячие волны, так что одни узлы получают мощные сигналы, а другие - настолько слабые, что их прием становится невозможным.

Станция должна подключаться к кабелю при помощи приемопередатчика — трансивера (transmitter + receiver = transceiver). Трансивер устанавливается непосредственно на кабеле и питается от сетевого адаптера компьютера. Трансивер может подсоединяться к кабелю как методом прокалывания, обеспечивающим непосредственный физический контакт, так и бесконтактным методом. Трансивер соединяется с сетевым адаптером интерфейсным кабелем AUI (Attachment Unit Interface) длиной до 50 м, состоящим из 4 витых пар (адаптер должен иметь разъем AUI). Наличие стандартного интерфейса между трансивером и остальной частью сетевого адаптера очень полезно при переходе с одного типа кабеля на другой. Для этого достаточно только заменить трансивер, а остальная часть сетевого адаптера остается неизменной, так как она отрабатывает протокол уровня MAC. При этом необходимо только, чтобы новый трансивер (например, трансивер для витой пары) поддерживал стандартный интерфейс AUI. Для присоединения к интерфейсу AUI используется разъем DB-15.

Допускается подключение к одному сегменту не более 100 трансиверов, причем расстояние между подключениями трансиверов не должно быть меньше 2,5 м. На кабеле имеется разметка через каждые 2,5 м, обозначающая точки подключения трансиверов. При подсоединении компьютеров и соответствии с разметкой влияние стоячих волн в кабеле на сетевые адAPTERы сводится к мини-

муму.

Трансивер — это часть сетевого адаптера, которая выполняет следующие функции:

- прием и передача данных с кабеля на кабель;
- определение коллизий на кабеле;
- электрическая развязка между кабелем и остальной частью адаптера;
- защита кабеля от некорректной работы адаптера.

Последнюю функцию иногда называют контролем «болтливости», что является буквальным переводом соответствующего английского термина (jabber control). При возникновении неисправностей в адаптере может возникнуть ситуация, когда на кабель будет непрерывно выдаваться последовательность случайных сигналов. Так как кабель — это общая среда для всех станций, то работа сети будет заблокирована одним неисправным адаптером. Чтобы этого не случилось, на выходе передатчика ставится схема, которая проверяет время передачи кадра. Если максимально возможное время передачи пакета превышается (с некоторым запасом), то эта схема просто отсоединяет выход передатчика от кабеля. Максимальное время передачи кадра (вместе с преамбулой) равно 1221 мкс, а время jabber-контроля устанавливается равным 4000 мкс (4 мс).

Упрощенная структурная схема трансивера показана на рис. 7.8. Передатчик и приемник присоединяются к одной точке кабеля с помощью специальной схемы, например трансформаторной, позволяющей организовать одновременную передачу и прием сигналов с кабеля.

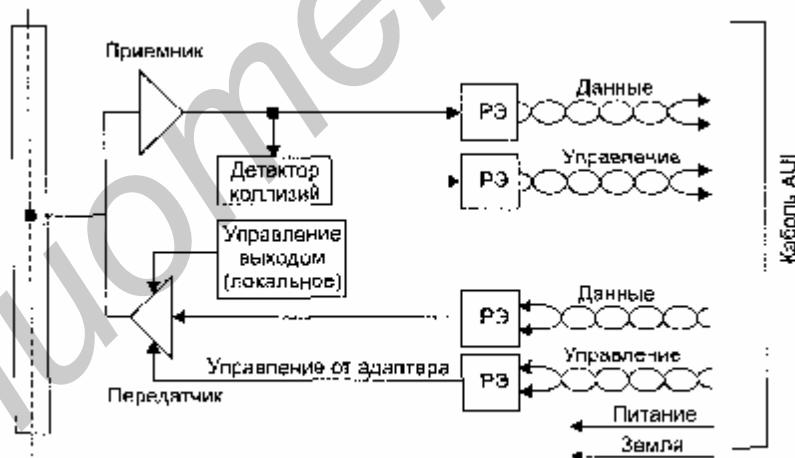


Рис. 7.8. Структурная схема трансивера

Детектор коллизий определяет наличие коллизии в коаксиальном кабеле по повышенному уровню постоянной составляющей сигналов. Если постоянная составляющая превышает определенный порог (около 1,5 В), значит, на кабель работает более одного передатчика. Развязывающие элементы (РЭ) обеспечивают гальваническую развязку трансивера от остальной части сетевого адаптера и тем самым защищают адаптер и компьютер от значительных перепадов напряжения, возникающих на кабеле при его повреждении.

Стандарт 10Base-5 определяет возможность использования в сети специального устройства — повторителя (repeater). Повторитель служит для объединения в

одну сеть нескольких сегментов кабеля и увеличения тем самым общей длины сети. Повторитель принимает сигналы из одного сегмента кабеля и побитно синхронно повторяет их в другом сегменте, улучшая форму и мощность импульсов, а также синхронизируя импульсы. Повторитель состоит из двух (или нескольких) трансиверов, которые присоединяются к сегментам кабеля, а также блока повторения со своим тактовым генератором. Для лучшей синхронизации повторитель задерживает передачу нескольких первых битов преамбулы кадра, за счет чего увеличивается задержка передачи кадра с сегмента на сегмент, а также несколько уменьшается межкадровый интервал IPG.

Стандарт разрешает использование в сети не более 4 повторителей и, соответственно, не более 5 сегментов кабеля. При максимальной длине сегмента кабеля в 500 м это дает максимальную длину сети 10Base-5 в 2500 м. Только 3 сегмента из 5 могут быть нагруженными, то есть такими, к которым подключаются конечные узлы. Между нагруженными сегментами должны быть ненагруженные сегменты, так что максимальная конфигурация сети представляет собой два нагруженных крайних сегмента, которые соединяются ненагруженными сегментами еще с одним центральным нагруженным сегментом. Ранее на рис. 7.7 был приведен пример сети Ethernet, состоящей из трех сегментов, объединенных двумя повторителями. Крайние сегменты являются нагруженными, а промежуточный — ненагруженным.

Правило применения повторителей в сети Ethernet 10Base-5 носит название правила 5.-4-3: 5 сегментов, 4 повторителя, 3 нагруженных сегмента. Ограниченнное число повторителей объясняется дополнительными задержками распространения сигнала, которые они вносят. Применение повторителей увеличивает время двойного распространения сигнала, которое для надежного распознавания коллизий не должно превышать время передачи кадра минимальной длины, то есть кадра в 72 байт или 576 бит.

Каждый повторитель подключается к сегменту одним своим трансивером, поэтому к нагруженным сегментам можно подключить не более 99 узлов. Максимальное число конечных узлов в сети 10Base-5 таким образом составляет $99 \times 3 = 297$ узлов.

К достоинствам стандарта 10Base-5 относятся:

- хорошая защищенность кабеля от внешних воздействий;
- сравнительно большое расстояние между узлами;
- возможность простого перемещения рабочей станции в пределах длины кабеля AUI.

Недостатками 10Base-5 являются:

- высокая стоимость кабеля;
- сложность прокладки кабеля из-за большой жесткости;
- потребность в специальном инструменте для заделки кабеля;
- останов работы всей сети при повреждении кабеля или плохом соединении;
- необходимость заранее предусмотреть подводку кабеля ко всем возможным местам установки компьютеров.

Стандарт 10Base-2

.Стандарт 10Base-2 использует в качестве передающей среды коаксиальный кабель с диаметром центрального медного провода 0,89 мм и внешним диаметром около 5 мм («тонкий» Ethernet). Кабель имеет волновое сопротивление 50 Ом. Такими характеристиками обладают кабели марок RG-58 /U, RG-58 A/U, RG-58 C/U.

Максимальная длина сегмента без повторителей составляет 185 м, сегмент должен иметь на концах согласующие терминалы 50 Ом. Тонкий коаксиальный кабель дешевле толстого, поэтому сети 10Base-2 иногда называют сетями Cheaper-net (от cheaper — более дешевый). Но за дешевизну кабеля приходится расплачиваться качеством — «тонкий» коаксиал обладает худшей помехозащищенностью, худшей механической прочностью и более узкой полосой пропускания.

Станции подключаются к кабелю с помощью высокочастотного (BNC) Т-коннектора, который представляет собой тройник, один отвод которого соединяется с сетевым адаптером, а два других — с двумя концами разрыва кабеля. Максимальное количество станций, подключаемых к одному сегменту, — 30. Минимальное расстояние между станциями — 1 м. Кабель «тонкого» коаксиала имеет разметку для подключения узлов с шагом в 1 м.

Стандарт 10Base-2 также предусматривает использование повторителей по «правилу 5-4-3». В этом случае сеть будет иметь максимальную длину в $5 \times 185 = 925$ м. Очевидно, что это ограничение является более сильным, чем общее ограничение в 2500 м.

ВНИМАНИЕ

Для построения корректной сети Ethernet нужно соблюсти много ограничений, причем некоторые из них относятся к одним и тем же параметрам сети - например, максимальная длина или максимальное количество компьютеров в сети должны удовлетворять одновременно нескольким разным условиям. Корректная сеть Ethernet должна соответствовать всем требованиям, но на практике достаточно соблюсти только наиболее жесткие. Так, если в сети Ethernet не должно быть более 1024 узлов, а стандарт 10Base-2 ограничивает число загруженных сегментов тремя, то общее количество узлов в сети 10Base-2 не должно превышать $29 \times 3 = 87$. Менее жесткое ограничение в 1024 конечных узла и сети 10Base-2 никогда не достигается.

Стандарт 10Base-2 очень близок к стандарту 10Base-5. Но трансиверы в нем объединены с сетевыми адаптерами за счет того, что более гибкий тонкий коаксиальный кабель может быть подведен непосредственно к выходному разъему платы сетевого адаптера, установленной в шасси компьютера. Кабель в данном случае «висит» на сетевом адаптере, что затрудняет физическое перемещение компьютеров.

Типичный состав сети стандарта 10Base-2, состоящей из одного сегмента кабеля, показан на рис. 7.9.

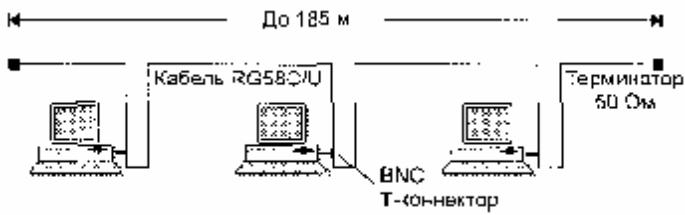


Рис. 7.9. Сеть стандарта 10Base-2

Реализация этого стандарта на практике приводит к наиболее простому решению для кабельной сети, так как для соединения компьютеров требуются только сетевые адAPTERы, Т-коннекторы и терминаторы 50 Ом. Однако этот вид кабельных соединений наиболее сильно подвержен авариям и сбоям: кабель более восприимчив к помехам, чем «толстый» коаксиал, в моноканале имеется большое количество механических соединений (каждый Т-коннектор дает три механических соединения, два из которых имеют жизненно важное значение для всей сети), пользователи имеют доступ к разъемам и могут нарушить целостность моноканала. Кроме того, эстетика и эргономичность этого решения оставляют желать лучшего, так как от каждой станции через Т-коннектор отходят два довольно заметных провода, которые под столом часто образуют моток кабеля — запас, необходимый на случай даже небольшого перемещения рабочего места.

Общим недостатком стандартов 10Base-5 и 10Base-2 является отсутствие оперативной информации о состоянии моноканала. Повреждение кабеля обнаруживается сразу же (сеть перестает работать), но для поиска отказавшего отрезка кабеля необходим специальный прибор — кабельный тестер.

Стандарт 10Base-T

Стандарт 10Base-T принят в 1991 году, как дополнение к существующему набору стандартов Ethernet, и имеет обозначение 802.3i.

Сети 10Base-T используют в качестве среды две неэкранированные витые пары (Unshielded Twisted Pair, UTP). Многопарный кабель на основе неэкранированной витой пары категории 3 (категория определяет полосу пропускания кабеля, величину перекрестных наводок NEXT и некоторые другие параметры его качества, более подробно о категориях кабелей см. подраздел «Стандарты кабелей для зданий и кампусов» раздела «Структурированная кабельная система» в главе 9) телефонные компании уже достаточно давно использовали для подключения телефонных аппаратов внутри зданий. Этот кабель носит также название Voice Grade, говорящее о том, что он предназначен для передачи голоса. Идея приспособить этот популярный вид кабеля для построения локальных сетей оказалась, очень плодотворной, так как многие здания уже были оснащены нужной кабельной системой. Оставалось разработать способ подключения сетевых адAPTERов и прочего коммуникационного оборудования к витой паре таким образом, чтобы изменения в сетевых адAPTERах и программном обеспечении сетевых операционных систем были бы минимальными по сравнению с сетями Ethernet на коаксиале. Это удалось, поэтому переход на витую пару требу-

ет только замены трансивера сетевого адаптера или порта маршрутизатора, а метод доступа и все протоколы канального уровня остались теми же, что и в сетях Ethernet на коаксиале.

Конечные узлы соединяются с помощью двух витых пар по топологии «точка-точка» со специальным устройством — многопортовым повторителем. Одна витая пара требуется для передачи данных от станции к повторителю (выход Tx сетевого адаптера), а другая — для передачи данных от повторителя к станции (вход Rx сетевого адаптера). На рис. 7.10 показан пример трехпортового повторителя. Повторитель принимает сигналы от одного из конечных узлов и синхронно передает их на все свои остальные порты, кроме того, с которого поступили сигналы.

Многопортовые повторители в данном случае обычно называются концентраторами (англоязычный термин — hub, или concentrator). Концентратор осуществляет функции повторителя сигналов на всех отрезках витых пар, подключенных к его портам, так что образуется единая среда передачи данных — логический моноканал (логическая общая шина). Повторитель обнаруживает коллизию в сегменте в случае одновременной передачи сигналов по нескольким своим Rx-входам и посыпает jam-последовательность на все свои Tx-выходы. Стандарт определяет битовую скорость передачи данных 10 Мбит/с и максимальное расстояние отрезка витой пары между двумя непосредственно связанными узлами (станциями и концентраторами) не более 100 м при наличии витой пары качества не ниже категории 3. Это расстояние определяется полосой пропускания витой пары — на длине 100 м она позволяет передавать данные со скоростью 10 Мбит/с при использовании манчестерского кода.

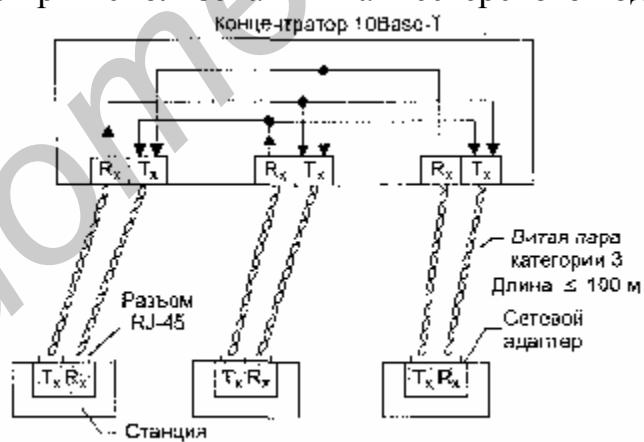


Рис. 7.10. Сеть стандарта 10Base-T: Tx — передатчик; Rx — приемник

Концентраторы 10Base-T можно соединять друг с другом с помощью тех же портов, которые предназначены для подключения конечных узлов. При этом нужно позаботиться о том, чтобы передатчик и приемник одного порта были соединены соответственно с приемником и передатчиком другого порта.

Для обеспечения синхронизации станций при реализации процедур доступа CSMA/CD и надежного распознавания станциями коллизии в стандарте определено максимальное число концентраторов между любыми двумя станциями сети, а именно 4. Это правило носит название «правила 4 хабов» и оно заменяет «правило 5-4-3», применяемое к коаксиальным сетям. При создании сети 10Base-T с большим числом станций концентраторы можно соединять друг с

другом иерархическим способом, образуя древовидную структуру (рис. 7.11).
ВНИМАНИЕ

Петлевидное соединение концентраторов в стандарте 10Base-T запрещено, так как оно приводит к некорректной работе сети. Это требование означает, что в сети 10Base-T не разрешается создавать параллельные каналы связи между критически важными концентраторами для резервирования связей на случай отказа порта, концентратора или кабеля. Резервирование связей возможно только за счет перевода одной из параллельных связей в неактивное (заблокированное) состояние.

Общее количество станций в сети 10Base-T не должно превышать общего предела в 1024, и для данного типа физического уровня это количество действительно достижимо. Для этого достаточно создать двухуровневую иерархию концентраторов, расположив на нижнем уровне достаточное количество концентраторов с общим количеством портов 1024 (рис. 7.12). Конечные узлы нужно подключить к портам концентраторов нижнего уровня. Правило 4 хабов при этом выполняется — между любыми конечными узлами будет ровно три концентратора.

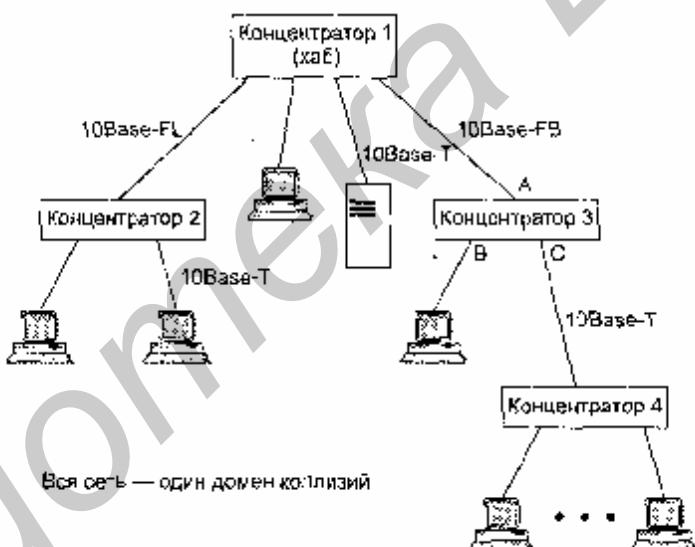


Рис. 7.11. Иерархическое соединение концентраторов Ethernet

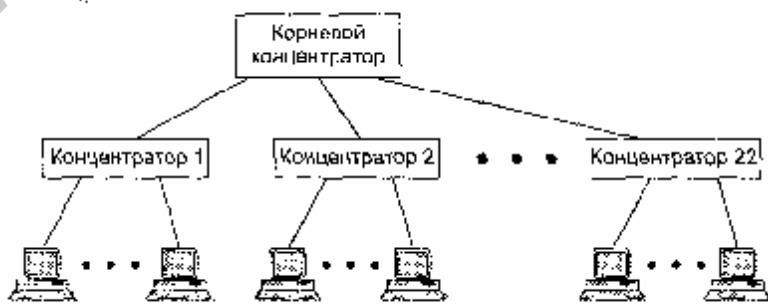


Рис. 7.12. Схема с максимальным количеством станций

Максимальная длина сети в 2500 м здесь понимается как максимальное расстояние между любыми двумя конечными узлами сети (часто применяется также термин «максимальный диаметр сети»). Очевидно, что если между любыми двумя узлами сети не должно быть больше 4 повторителей, то максимальный

диаметр сети 10Base-T составляет $5 \times 100 = 500$ м.

Сети, построенные на основе стандарта 10Base-T, обладают по сравнению с коаксиальными вариантами Ethernet многими преимуществами. Эти преимущества связаны с разделением общего физического кабеля на отдельные кабельные отрезки, подключенные к центральному коммуникационному устройству. И хотя логически эти отрезки по-прежнему образуют общую разделяемую среду, их физическое разделение позволяет контролировать их состояние и отключать в случае обрыва, короткого замыкания или неисправности сетевого адаптера на индивидуальной основе. Это обстоятельство существенно облегчает эксплуатацию больших сетей Ethernet, так как концентратор обычно автоматически выполняет такие функции, уведомляя при этом администратора сети о возникшей проблеме.

В стандарте 10Base-T определена процедура тестирования физической работоспособности двух отрезков витой пары, соединяющих трансивер конечного узла и порт повторителя. Эта процедура называется тестом связности (link test) и основана на передаче каждые 16 мс специальных импульсов J и K манчестерского кода между передатчиком и приемником каждой витой пары. Если тест не проходит, то порт блокируется и отключает проблемный узел от сети. Так как коды J и K являются запрещенными при передаче кадров, то тестовые последовательности не влияют на работу алгоритма доступа к среде.

Появление между конечными узлами активного устройства, которое может контролировать работу узлов и изолировать от сети некорректно работающие, является главным преимуществом технологии 10Base-T по сравнению со сложными в эксплуатации коаксиальными сетями. Благодаря концентраторам сеть Ethernet приобрела некоторые черты отказоустойчивой системы.

Оптоволоконная сеть Ethernet

В качестве среды передачи данных 10-мегабитная сеть Ethernet использует оптическое волокно. Оптоволоконные стандарты в качестве основного типа кабеля рекомендуют достаточно дешевое многомодовое оптическое волокно, обладающее полосой пропускания 500-800 МГц при длине кабеля 1 км. Допустимо и более дорогое одномодовое оптическое волокно с полосой пропускания в несколько гигагерц, но при этом нужно применять специальный тип трансивера.

Функционально сеть Ethernet на оптическом кабеле состоит из тех же элементов, что и сеть стандарта 10Base-T — сетевых адаптеров, многопортового повторителя и отрезков кабеля, соединяющих адаптер с портом повторителя. Как и в случае витой пары, для соединения адаптера с повторителем используется два оптоволокна — одно соединяет выход Tx адаптера с входом Rx повторителя, а другое — вход Rx адаптера с выходом Tx повторителя.

Стандарт FOIRL (Fiber Optic Inter-Repeater Link) представляет собой первый стандарт комитета 802.3 для использования оптоволокна в сетях Ethernet. Он гарантирует длину оптоволоконной связи между повторителями до 1 км при общей длине сети не более 2500 м. Максимальное число повторителей между любыми узлами сети — 4. Максимального диаметра в 2500 м здесь достичь

можно, хотя максимальные отрезки кабеля между всеми 4 повторителями, а также между повторителями и конечными узлами недопустимы — иначе получится сеть длиной 5000 м.

Стандарт 10Base-FL представляет собой незначительное улучшение стандарта FOIRL. Увеличена мощность передатчиков, поэтому максимальное расстояние между узлом и концентратором увеличилось до 2000 м. Максимальное число повторителей между узлами осталось равным 4, а максимальная длина сети — 2500 м.

Стандарт 10Base-FB предназначен только для соединения повторителей. Конечные узлы не могут использовать этот стандарт для присоединения к портам концентратора. Между узлами сети можно установить до 5 повторителей 10Base-FB при максимальной длине одного сегмента 2000 м и максимальной длине сети 2740 м.

Повторители, соединенные по стандарту 10Base-FB, при отсутствии кадров для передачи постоянно обмениваются специальными последовательностями сигналов, отличающимися от сигналов кадров данных, для поддержания синхронизации. Поэтому они вносят меньшие задержки при передаче данных из одного сегмента в другой, и это является главной причиной, по которой количество повторителей удалось увеличить до 5. В качестве специальных сигналов используются манчестерские коды J и K в следующей последовательности: J-J-K-K-J-J-... Эта последовательность порождает импульсы частоты 2,5 МГц, которые и поддерживают синхронизацию приемника одного концентратора с передатчиком другого. Поэтому стандарт 10Base-FB имеет также название синхронный Ethernet.

Как и в стандарте 10Base-T, оптоволоконные стандарты Ethernet разрешают соединять концентраторы только в древовидные иерархические структуры. Любые петли между портами концентраторов не допускаются.

Домен коллизий

В технологии Ethernet, независимо от применяемого стандарта физического уровня, существует понятие домена коллизии.

Домен коллизий (collision domain) — это часть сети Ethernet, вес узлы которой распознают коллизию независимо от того, в какой части этой сети коллизия возникла. Сеть Ethernet, построенная на повторителях, всегда образует один домен коллизий. Домен коллизий соответствует одной разделяемой среде. Мосты, коммутаторы и маршрутизаторы делят сеть Ethernet на несколько доменов коллизий.

Приведенная на рис. 7.11 сеть представляет собой один домен коллизий. Если, например, столкновение кадров произошло в концентраторе 4, то в соответствии с логикой работы концентраторов 10Base-T сигнал коллизии распространится по всем портам всех концентраторов.

Если же вместо концентратора 3 поставить в сеть мост, то его порт С, связанный с концентратором 4, воспримет сигнал коллизии, но не передаст его на свои остальные порты, так как это не входит в его обязанности. Мост просто

отработает ситуацию коллизии средствами порта С, который подключен к общей среде, где эта коллизия возникла. Если коллизия возникла из-за того, что мост пытался передать через порт С кадр в концентратор 4, то, зафиксирован сигнал коллизии, порт С приостановит передачу кадра и попытается передать его повторно через случайный интервал времени. Если порт С принимал в момент возникновения коллизии кадр, то он просто отбросит полученное начало кадра и будет ожидать, когда узел, передававший кадр через концентратор 4, сделает повторную попытку передачи. После успешного принятия данного кадра в свой буфер мост передаст его на другой порт в соответствии с таблицей продвижения, например на порт А. Все события, связанные с обработкой коллизий портом С, для остальных сегментов сети, которые подключены к другим портам моста, просто останутся неизвестными.

Узлы, образующие один домен коллизий, работают синхронно, как единая распределенная электронная схема.

Общие характеристики стандартов Ethernet 10 Мбит/с

В табл. 7.3 и 7.4 сведены основные ограничения и характеристики стандартов Ethernet.

Таблица 7.3. Общие ограничения для всех стандартов Ethernet

Характеристика	Значение
Номинальная пропускная способность	10 Мбит/с
Максимальное число станций в сети	1024
Максимальное расстояние между узлами в сети	2500 м (в 10Base-TB 2750 м)
Максимальное число кратчайших сегментов в сети	5

Таблица 7.4. Параметры спецификаций физического уровня для стандарта Ethernet

Параметр	10Base-5	10Base-2	10Base-T	10Base-F
Кабель	Толстый коаксиальный кабель RG-8 или RG-11	Тонкий коаксиальный кабель RG-58	Неакриптическая витая пара категорий 3, 4, 5	Многомодовый волоконно-оптический кабель
Максимальная длина сегмента, м	500	185	100	2000
Максимальное расстояние между узлами сети (при использовании повторителей), м	2500	925	500	2500 (2740 для 10Base-FB)
Максимальное число станций в сегменте	100	30	1024	1024
Максимальное число повторителей между любыми двумя станциями сети	4	4	4	(5 для 10Base-FB)

Методика расчета конфигурации сети Ethernet

Соблюдение многочисленных ограничений, установленных для различных стандартов физического уровня сетей Ethernet, гарантирует корректную работу сети (естественно, при исправном состоянии всех элементов физического уровня).

Наиболее часто приходится проверять ограничения, связанные с длиной отдельного сегмента кабеля, а также количеством повторителей и общей длиной сети.

Правила «5-4-3» для коаксиальных сетей и «4 хабов» для сетей на основе витой пары и оптоволокна не только дают гарантии работоспособности сети, но и оставляют большой «запас прочности» сети. Например, если посчитать время двойного оборота в сети, состоящей из 4 повторителей 10Base-5 и 5 сегментов максимальный длины 500 м, то окажется, что оно составляет 537 битовых интервалов. А так как время передачи кадра минимальной длины, состоящего вместе с преамбулой, составляющей 72 байт, равно 575 битовым интервалам, то видно, что разработчики стандарта Ethernet оставили 38 битовых интервала в качестве запаса для обеспечения надежности. Тем не менее в документах комитета 802.3 утверждается, что и 4 дополнительных битовых интервала создают достаточный запас надежности.

Комитет IEEE 802.3 приводит исходные данные о задержках, вносимых повторителями и различными средами передачи данных, для тех специалистов, которые хотят самостоятельно рассчитывать максимальное количество повторителей и максимальную общую длину сети, не довольствуясь теми значениями, которые приведены в правилах «5-4-3» и «4 хабов». Особенno такие расчеты полезны для сетей, состоящих из смешанных кабельных систем, например коаксиала и оптоволокна, на которые правила о количестве повторителей не расписаны. При этом максимальная длина каждого отдельного физического сегмента должна строго соответствовать стандарту, то есть 500 м для «толстого» коаксиала, 100 м для витой пары и т. д.

Чтобы сеть Ethernet, состоящая из сегментов различной физической природы, работала корректно, необходимо выполнение четырех основных условий:

- количество станций в сети — не более 1024;
- максимальная длина каждого физического сегмента — не более величины, определенной в соответствующем стандарте физического уровня;
- время двойного оборота сигнала (Path Delay Value, PDV) между двумя самыми удаленными друг от друга станциями сети — не более 575 битовых интервалов;
- сокращение межкадрового интервала IPG (Path Variability Value, PVV) при прохождении последовательности кадров через все повторители — не больше, чем 49 битовых интервалов (так как при отправке кадров конечные узлы обеспечивают начальное межкадровое расстояние в 96 битовых интервалов, то после прохождения повторителя оно должно быть не меньше, чем $96 - 49 = 47$ битовых интервалов).

Соблюдение этих требований обеспечивает корректность работы сети даже в

случаях, когда нарушаются простые правила конфигурирования, определяющие максимальное количество повторителей и общую длину сети в 2500 м.

Расчет PDV

Для упрощения расчетов обычно используются справочные данные IEEE, содержащие значения задержек распространения сигналов в повторителях, приемопередатчиках и различных физических средах. В табл. 7.5 приведены данные, необходимые для расчета значения PDV для всех физических стандартов сетей Ethernet. Битовый интервал обозначен как bt.

Таблица 7.5. Данные для расчета значения PDV

Тип сегмента	База левого сегмента, bt	База промежуточного сегмента, bt	База правого сегмента, bt	Задержка среды на 1 м, bt	Максимальная длина сегмента, м
10Base-5	11,8	46,5	169,5	0,0866	500
10Base-2	11,8	46,5	169,5	0,1026	185
10Base-T	15,3	42,0	165,0	0,113	160
10Base-FB	—	34,0	—	0,1	2000
10Base-FL	12,3	33,5	166,5	0,1	2000
FDDI	7,6	29,0	152,0	0,1	1000
AUI (>2 м)	0	0	0	0,1026	2,48

Комитет 802.3 старался максимально упростить выполнение расчетов, поэтому данные, приведенные в таблице, включают сразу несколько этапов прохождения сигнала. Например, задержки, вносимые повторителем, состоят из задержки входного трансивера, задержки блока повторения и задержки выходного трансивера. Тем не менее в таблице все эти задержки представлены одной величиной, названной базой сегмента.

Чтобы не нужно было два раза складывать задержки, вносимые кабелем, в таблице даются удвоенные величины задержек для каждого типа кабеля.

В таблице используются также такие понятия, как левый сегмент, правый сегмент и промежуточный сегмент. Поясним эти термины на примере сети, приведенной на рис. 7.13. Левым сегментом называется сегмент, в котором начинается путь сигнала от выхода передатчика (выход Tx на рис. 7.10) конечного узла. На примере это сегмент 1. Затем сигнал проходит через промежуточные сегменты 2-5 и доходит до приемника (вход Rx на рис. 7.10) наиболее удаленного узла наиболее удаленного сегмента 6, который называется правым. Именно здесь в худшем случае происходит столкновение кадров и возникает коллизия, что и подразумевается в таблице.

С каждым сегментом связана постоянная задержка, названная базой, которая зависит только от типа сегмента и от положения сегмента на пути сигнала (левый, промежуточный или правый). База правого сегмента, в котором возникает коллизия, намного превышает базу левого и промежуточных сегментов.

Кроме этого, с каждым сегментом связана задержка распространения сигнала вдоль кабеля сегмента, которая зависит от длины сегмента и вычисляется путем умножения времени распространения сигнала по одному метру кабеля (в битовых интервалах) на длину кабеля в метрах.

Расчет заключается в вычислении задержек, вносимых каждым отрезком кабеля (приведенная в таблице задержка сигнала на 1 м кабеля умножается на длину сегмента), а затем суммировании этих задержек с базами левого, промежуточных и правого сегментов. Общее значение PDV не должно превышать 575.

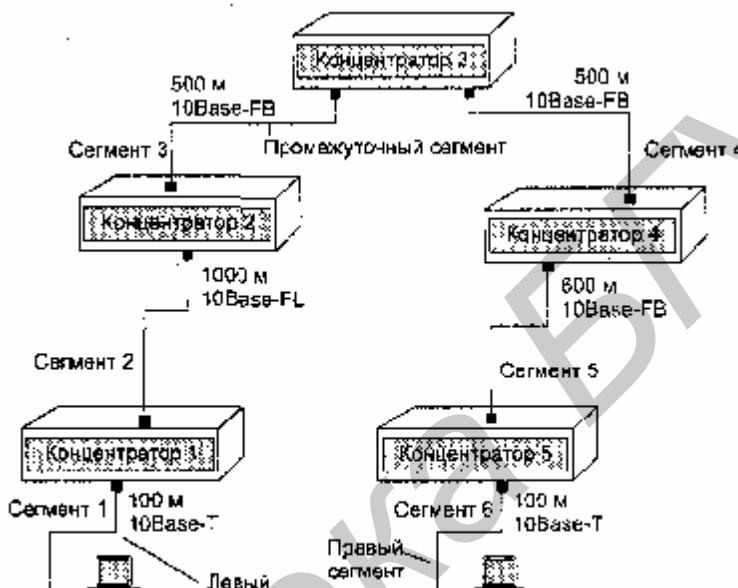


Рис. 7.13. Пример сети Ethernet, состоящей из сегментов различных физических стандартов

Так как левый и правый сегменты имеют разные величины базовой задержки, то в случае различных типов сегментов на удаленных краях сети необходимо выполнить расчеты дважды: одну раз принять в качестве левого сегмента сегмент одного типа, а во второй — сегмент другого типа. Результатом можно считать максимальное значение PDV. В нашем примере крайние сегменты сети принадлежат к одному типу — стандарту 10Base-T, поэтому двойной расчет не требуется, но если бы они были сегментами разного типа, то в первом случае нужно было бы принять в качестве левого сегмент между станцией и концентратором 1, а во втором считать левым сегмент между станцией и концентратором 5.

Приведенная на рисунке сеть в соответствии с правилом 4 хабов не является корректной — в сети между узлами сегментов 1 и 6 имеется 5 хабов, хотя не все сегменты являются сегментами 10Base-FB. Кроме того, общая длина сети равна 2800 м, что нарушает правило 2500 м. Рассчитаем значение PDV для нашего примера.

Левый сегмент 1:

$$15,3 \text{ (база)} + 100 \times 0,113 = 26,6.$$

Промежуточный сегмент 2:

$$33,5 + 1000 \times 0,1 = 133,5.$$

- Промежуточный сегмент 3:

$$24 + 500 \times 0,1 = 74,0.$$
- Промежуточный сегмент 4:

$$24 + 500 \times 0,1 - 74,0.$$
- Промежуточный сегмент 5:

$$24 + 600 \times 0,1 = 84,0.$$
- Правый сегмент 6:

$$165 + 100 \times 0,113 = 176,3.$$

Сумма всех составляющих дает значение PDV, равное 568,4.

Так как значение PDV меньше максимально допустимой величины 575, то эта сеть проходит по критерию времени двойного оборота сигнала несмотря на то, что ее общая длина превышает 2500 м, а количество повторителей больше 4.

Расчет PVV

Чтобы признать конфигурацию сети корректной, нужно рассчитать также уменьшение межкадрового интервала повторителями, то есть величину PVV. Для расчета PVV также можно воспользоваться значениями максимальных величин уменьшения межкадрового интервала при прохождении повторителей различных физических сред, рекомендованными IEEE и приведенными в табл. 7.6.

Таблица 7.6. Уменьшение межкадрового интервала повторителями

Тип сегмента	Передающий сегмент, бт	Промежуточный сегмент, бт
10Base-5 или 10Base-2	16	11
10Base-FB	—	2
10Base-FL	10,5	8
10Base-T	10,5	8

В соответствии с этими данными рассчитаем значение PVV для нашего примера.

- Левый сегмент 1 10Base-T: сокращение в 10,5 бт.
- Промежуточный сегмент 2 10Base-FL: 8.
- Промежуточный сегмент 3 10Base-FB: 2.
- Промежуточный сегмент 4 10Base-FB: 2.
- Промежуточный сегмент 5 10Base-FB: 2.

Сумма этих величин дает значение PVV, равное 24,5, что меньше предельного значения в 49 битовых интервала.

В результате приведенная в примере сеть соответствует стандартам Ethernet по всем параметрам, связанным и с длинами сегментов, и с количеством повторителей.

Выводы

- Ethernet — это самая распространенная на сегодняшний день технология ло-

кальных сетей. В широком смысле Ethernet — это целое семейство технологий, включающее различные фирменные и стандартные варианты, из которых наиболее известны фирменный вариант Ethernet DIX, 10-мегабитные варианты стандарта IEEE 802.3, а также новые высокоскоростные технологии Fast Ethernet и Gigabit Ethernet. Почти все виды технологий Ethernet используют один и тот же метод разделения среды передачи данных — метод случайного доступа CSMA/CD, который определяет облик технологии в целом.

□ В узком смысле Ethernet — это 10-мегабитная технология, описанная в стандарте IEEE 802.3.

□ Важным явлением в сетях Ethernet является коллизия — ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Наличие коллизий — это неотъемлемое свойство сетей Ethernet, являющееся следствием принятого случайного метода доступа. Возможность четкого распознавания коллизий обусловлена правильным выбором параметров сети, в частности соблюдением соотношения между минимальной длиной кадра и максимально возможным диаметром сети.

□ На характеристики производительности сети большое значение оказывает коэффициент использования сети, который отражает ее загруженность. При значениях этого коэффициента свыше 50 % полезная пропускная способность сети резко падает из-за роста интенсивности коллизий и увеличения времени ожидания доступа к среде.

□ Максимально возможная пропускная способность сегмента Ethernet в кадрах в секунду достигается при передаче кадров минимальной длины и составляет 14 880 кадр/с. При этом полезная пропускная способность сети составляет всего 5,48 Мбит/с, что лишь ненамного превышает половину номинальной пропускной способности — 10 Мбит/с.

□ Максимально возможная полезная пропускная способность сети Ethernet составляет 9,75 Мбит/с, что соответствует использованию кадров максимальной длины в 1518 байт, которые передаются по сети со скоростью 513 кадр/с.

□ При отсутствии коллизий и ожидания доступа коэффициент использования сети зависит от размера поля данных кадра и имеет максимальное значение 0,96.

□ Технология Ethernet поддерживает 4 разных типа кадров, которые имеют общий формат адресов узлов. Существуют формальные признаки, по которым сетевые адаптеры автоматически распознают тип кадра.

□ В зависимости от типа физической среды стандарт IEEE 802.3 определяет различные спецификации: 10Base-5, 10Base-2, 10Base-T, FOIRL, 10Base-FL, 10Base-FB. Для каждой спецификации определяются тип кабеля, максимальные длины непрерывных отрезков кабеля, а также правила использования повторителей для увеличения диаметра сети: правило «5-4-3» для коаксиальных вариантов сетей и правило «4 хабов» для витой пары и оптоволокна.

□ Для «смешанной» сети, состоящей из физических сегментов различного типа, полезно проводить расчет общей длины сети и допустимого количества повторителей. Комитет IEEE 802.3 приводит исходные данные для таких расчетов, в которых указываются задержки, вносимые повторителями различных специфи-

каций физической среды, сетевыми адаптерами и сегментами кабеля.

Технология Token Ring

Сети Token Ring (стандарт 802.5), так же как и сети Ethernet, характеризует разделяемая среда передачи данных, которая в данном случае состоит из отрезков кабеля, соединяющих все станции сети в кольцо. Кольцо рассматривается как общий разделяемый ресурс, и для доступа к нему требуется не случайный алгоритм, как в сетях Ethernet, а детерминированный, основанный на передаче станциям права на использование кольца в определенном порядке. Это право передается с помощью кадра специального формата, называемого маркером, или токеном, (token).

Технология Token Ring был разработана компанией IBM в 1984 году, а затем передана в качестве проекта стандарта в комитет IEEE 802, который на ее основе принял в 1985 году стандарт 802.5. Компания IBM в течение долгого времени использовала технологию Token Ring в качестве своей основной сетевой технологии для построения локальных сетей на основе компьютеров различных классов — мейнфреймов, мини-компьютеров и персональных компьютеров. Однако в последнее время даже в продукции компании IBM доминируют представители семейства Ethernet.

Сети Token Ring работают с двумя битовыми скоростями — 4 и 16 Мбит/с. Смешение станций, работающих на различных скоростях, в одном кольце не допускается. Сети Token Ring, работающие со скоростью 16 Мбит/с, имеют некоторые усовершенствования в алгоритме доступа по сравнению со стандартом 4 Мбит/с.

Технология Token Ring является более сложной технологией, чем Ethernet. Она обладает свойствами отказоустойчивости. В сети Token Ring определены процедуры контроля работы сети, которые используют обратную связь кольцеобразной структуры — посланный кадр всегда возвращается в станцию-отправитель. В некоторых случаях обнаруженные ошибки в работе сети устраняются автоматически, например может быть восстановлен потерянный маркер. В других случаях ошибки только фиксируются, а их устранение выполняется вручную обслуживающим персоналом.

Для контроля сети одна из станций выполняет роль так называемого активного монитора. Активный монитор выбирается во время инициализации кольца как станция с максимальным значением MAC-адреса. Если активный монитор выходит из строя, процедура инициализации кольца повторяется и выбирается новый активный монитор. Чтобы сеть могла обнаружить отказ активного монитора, последний в работоспособном состоянии каждые три секунды генерирует специальный кадр своего присутствия. Если этот кадр не появляется в сети более 7 с, то остальные станции сети начинают процедуру выборов нового активного монитора.

Маркерный метод доступа к разделяемой среде

В сетях с маркерным методом доступа (а к ним, кроме сетей Token Ring, относятся сети FDDI, а также сети, близкие к стандарту 802.4, — ArcNet, сети производственного назначения MAP) право на доступ к среде передается циклически от станции к станции по логическому кольцу.

В сети Token Ring кольцо образуется отрезками кабеля, соединяющими соседние станции. Таким образом, каждая станция связана со своей предшествующей и последующей станциями и может непосредственно обмениваться данными только с ними. Для обеспечения доступа станций к физической среде по кольцу циркулирует кадр специального формата и назначения — маркер. В сети Token Ring любая станция всегда непосредственно получает данные только от одной станции — той, которая является предыдущей в кольце. Такая станция называется ближайшим активным соседом, расположенным выше по потоку (данных) — Nearest Active Upstream Neighbor, NAUN. Передачу же данных станция всегда осуществляет своему ближайшему соседу вниз по потоку данных.

Получив маркер, станция анализирует его и при отсутствии у нее данных для передачи обеспечивает его продвижение к следующей станции. Станция, которая имеет данные для передачи, при получении маркера изымает его из кольца, что дает ей право доступа к физической среде для передачи своих данных. Затем эта станция выдает в кольцо кадр данных установленного формата последовательно по битам. Переданные данные проходят по кольцу всегда в одном направлении от одной станции к другой. Кадр снабжен адресом назначения и адресом источника.

Все станции кольца ретранслируют кадр побитно, как повторители. Если кадр проходит через станцию назначения, то, распознав свой адрес, эта станция копирует кадр в свой внутренний буфер и вставляет в кадр признак подтверждения приема. Станция, выдавшая кадр данных в кольцо, при обратном его получении с подтверждением приема изымает этот кадр из кольца и передает в сеть новый маркер, давая другим станциям сети возможность передавать данные. Такой алгоритм доступа применяется в сетях Token Ring со скоростью работы 4 Мбит/с, описанных в стандарте 802.5.

На рис. 7.14 описанный алгоритм доступа к среде иллюстрируется временной диаграммой. Здесь показана передача пакета А в кольце, состоящем из 6 станций, от станции 1 к станции 3. После прохождения станции назначения 3 в пакете А устанавливаются два признака — признак распознавания адреса и признак копирования пакета в буфер (что на рисунке отмечено звездочкой внутри пакета). После возвращения пакета в станцию 1 отправитель распознает свой пакет по адресу источника и удаляет пакет из кольца. Установленные станцией 3 признаки говорят станции-отправителю о том, что пакет дошел до адресата и был успешно скопирован им в свой буфер.

Время владения разделяемой средой в сети Token Ring ограничивается временем, удержания маркера (token holding time), после истечения которого станция обязана прекратить передачу собственных данных (текущий кадр разрешается завершить) и передать маркер далее по кольцу. Станция может успеть передать

за время удержания маркера один или несколько кадров в зависимости от размера кадров и величины времени удержания маркера. Обычно время удержания маркера по умолчанию равно 10 мс, а максимальный размер кадра в стандарте 802.5 не определен. Для сетей 4 Мбит/с он обычно равен 4 Кбайт, а для сетей 16 Мбит/с — 16 Кбайт. Это связано с тем, что за время удержания маркера станция должна успеть передать хотя бы один кадр. При скорости 4 Мбит/с за время 10 мс можно передать 5000 байт, а при скорости 16 Мбит/с — 20 000 байт. Максимальные размеры кадра выбраны с некоторым запасом.

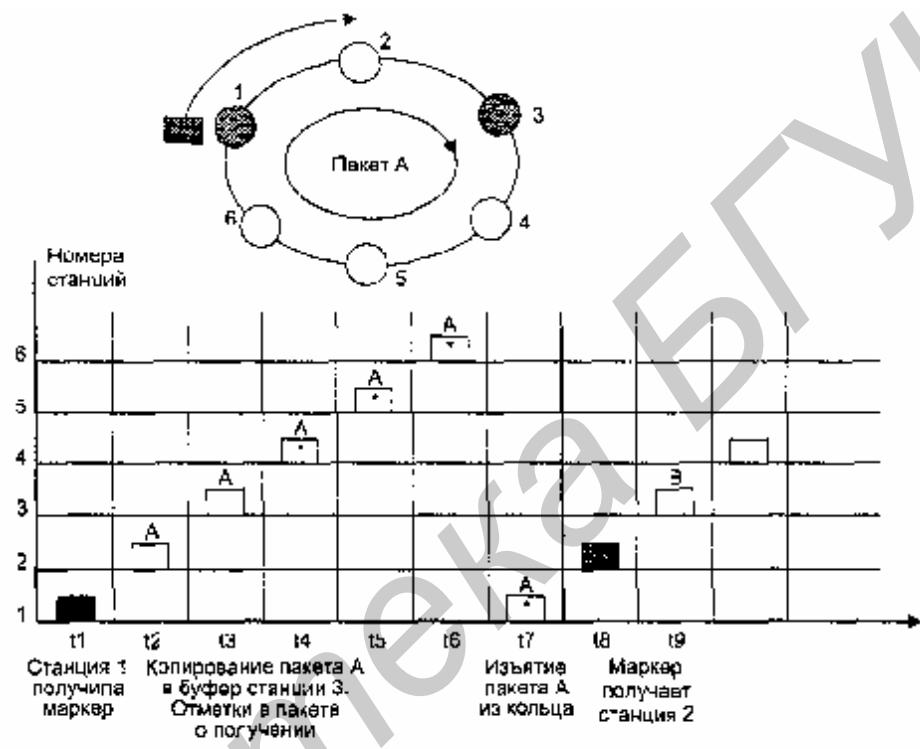


Рис. 7.14. Принцип маркерного доступа

В сетях Token Ring 16 Мбит/с используется также несколько другой алгоритм доступа к кольцу, называемый алгоритмом раннего освобождения маркера (Early Token Release). В соответствии с ним станция передает маркер доступа следующей станции сразу же после окончания передачи последнего бита кадра, не дожидаясь возвращения по кольцу этого кадра с битом подтверждения приема. В этом случае пропускная способность кольца используется более эффективно, так как по кольцу одновременно продвигаются кадры нескольких станций. Тем не менее свои кадры в каждый момент времени может генерировать только одна станция — та, которая в данный момент владеет маркером доступа. Остальные станции в это время только повторяют чужие кадры, так что принцип разделения кольца во времени сохраняется, ускоряется только процедура передачи владения кольцом.

Для различных видов сообщений, передаваемым кадрам, могут назначаться различные приоритеты: от 0 (низший) до 7 (высший). Решение о приоритете конкретного кадра принимает передающая станция (протокол Token Ring получает этот параметр через межуровневые интерфейсы от протоколов верхнего

уровня, например прикладного). Маркер также всегда имеет некоторый уровень текущего приоритета. Станция имеет право захватить переданный ей маркер только в том случае, если приоритет кадра, который она хочет передать, выше (или равен) приоритета маркера. В противном случае станция обязана передать маркер следующей по кольцу станции.

За наличие в сети маркера, причем единственной его копии, отвечает активный монитор. Если активный монитор не получает маркер в течение длительного времени (например, 2,6 с), то он порождает новый маркер.

Форматы кадров Token Ring

В Token Ring существуют три различных формата кадров:

- маркер;
- кадр данных;
- прерывающая последовательность.

Маркер

Кадр маркера состоит из трех полей, каждое длиной в один байт.

- Поле начального ограничителя (Start Delimiter, SD) появляется в начале маркера, а также в начале любого кадра, проходящего по сети. Поле представляет собой следующую уникальную последовательность символов манчестерского кода: JK0JKQOO. Поэтому начальный ограничитель нельзя спутать ни с какой битовой последовательностью внутри кадра.
- Поле управления доступом (Access Control) состоит из четырех подполей: PPP, T, M и RRR, где PPP — биты приоритета, T — бит маркера, M — бит монитора, RRR — резервные биты приоритета. Бит T, установленный в 1, указывает на то, что данный кадр является маркером доступа. Бит монитора устанавливается в 1 активным монитором и в 0 любой другой станцией, передающей маркер или кадр. Если активный монитор видит маркер или кадр, содержащий бит монитора со значением 1, то активный монитор знает, что этот кадр или маркер уже однажды обошел кольцо и не был обработан станциями. Если это кадр, то он удаляется из кольца. Если это маркер, то активный монитор передает его дальше по кольцу. Использование полей приоритетов будет рассмотрено ниже.
- Поле конечного ограничителя (End Delimiter, ED) — последнее поле маркера. Так же как и поле начального ограничителя, это поле содержит уникальную последовательность манчестерских кодов JK1JK1, а также два однобитовых признака: I и E. Признак I (Intermediate) показывает, является ли кадр последним в серии кадров ($I = 0$) или промежуточным ($I = 1$). Признак E (Error) — это признак ошибки. Он устанавливается в 0 станцией-отправителем, и любая станция кольца, через которую проходит кадр, должна установить этот признак в 1, если она обнаружит ошибку по контрольной сумме или другую некорректность кадра.

Кадр данных и прерывающая последовательность

Кадр данных включает те же три поля, что и маркер, и имеет кроме них еще несколько дополнительных полей. Таким образом, кадр данных состоит из следующих полей:

- поля начального ограничителя (Start Delimiter, SD);
- поля управления кадром (Frame Control, FC);
- поля адреса назначения (Destination Address, DA);
- поля адреса источника (Source Address, SA);
- поля данных (INFO);
- поля контрольной суммы (Frame Check Sequence, FCS);
- поля конечного ограничителя (End Delimeter, ED);
- поля статуса кадра (Frame Status, FS).

Кадр данных может переносить либо служебные данные для управления кольцом (данные MAC-уровня), либо пользовательские данные (LLC-уровня). Стандарт Token Ring определяет 6 типов управляющих кадров MAC-уровня. Поле FC определяет тип кадра (MAC или LLC), и если он определен как MAC, то поле также указывает, какой из шести типов кадров представлен данным кадром.

Назначение этих шести типов кадров описано ниже.

- Чтобы удостовериться, что ее адрес уникальный, станция, когда впервые присоединяется к кольцу, посыпает кадр Тест дублирования адреса (Duplicate Address Test, DAT).
- Чтобы сообщить другим станциям, что он работоспособен, активный монитор периодически посыпает в кольцо кадр Существует активный монитор (Active Monitor Present, AMP).
- Кадр Существует резервный монитор (Standby Monitor Present, SMP) отправляется любой станцией, не являющейся активным монитором.
- Резервный монитор отправляет кадр Маркер заявки (Claim Token, CT), когда подозревает, что активный монитор отказал, затем резервные мониторы договариваются между собой, какой из них станет новым активным монитором.
- Станция отправляет кадр Сигнал (Beacon, BCN) в случае возникновения серьезных сетевых проблем, таких как обрыв кабеля, обнаружение станции, передающей кадры без ожидания маркера, выход станции из строя. Определяя, какая станция отправляет кадр сигнала, диагностирующая программа (ее существование и функции не определяются стандартами Token Ring) может локализовать проблему. Каждая станция периодически передает кадры BCN до тех пор, пока не примет кадр BCN от своего предыдущего (NAUN) соседа. В результате в кольце только одна станция продолжает передавать кадры BCN — та, у которой имеются проблемы с предыдущим соседом. В сети Token Ring каждая станция знает MAC-адрес своего предыдущего соседа, поэтому Beacon-процедура приводит к выявлению адреса некорректно работающей станции.
- Кадр Очистка (Purge, PRC) используется новым активным монитором для того, чтобы перевести все станции в исходное состояние и очистить кольцо от

всех ранее посланных кадров.

В стандарте 802.5 используются адреса той же структуры, что и в стандарте 802.3. Адреса назначения и источника могут иметь длину либо 2, либо 6 байт.

Первый бит адреса назначения определяет групповой или индивидуальный адрес как для 2-байтовых, так и для 6-байтовых адресов. Второй бит в 6-байтовых адресах говорит о том, назначен адрес локально или глобально. Адрес, состоящий из всех единиц, является широковещательным.

Адрес источника имеет тот же размер и формат, что и адрес назначения. Однако признак группового адреса используется в нем особым способом. Так как адрес источника не может быть групповым, то наличие единицы в этом разряде говорит о том, что в кадре имеется специальное поле маршрутной информации (Routing Information Field, RIF). Эта информация требуется при работе мостов, связывающих несколько колец Token Ring, в режиме маршрутизации от источника.

Поле данных INFO кадра может содержать данные одного из описанных управляющих кадров уровня MAC или пользовательские данные, упакованные в кадр уровня LLC. Это поле, как уже отмечалось, не имеет определенной стандартом максимальной длины, хотя существуют практические ограничения на его размер, основанные на временных соотношениях между временем удержания маркера и временем передачи кадра.

Поле статуса FS имеет длину 1 байт и содержит 4 резервных бита и 2 под поля: бит распознавания адреса A и бит копирования кадра C. Так как это поле не сопровождается вычисляемой суммой CRC, то используемые биты для надежности дублируются: поле статуса FS имеет вид ACxxACxx. Если бит распознавания адреса не установлен во время получения кадра, это означает, что станция назначения больше не присутствует в сети (возможно, вследствие неполадок, а возможно, станция находится в другом кольце, связанном с данным с помощью моста). Если оба бита опознавания адреса и копирования кадра установлены, а также установлен бит обнаружения ошибки, то исходная станция знает, что ошибка случилась после того, как этот кадр был корректно получен.

Прерывающая последовательность состоит из двух байтов, содержащих начальный и конечный ограничители. Прерывающая последовательность может появиться в любом месте потока битов и сигнализирует о том, что текущая передача кадра или маркера отменяется.

Приоритетный доступ к кольцу

Каждый кадр данных или маркер имеет приоритет, устанавливаемый битами приоритета (значение от 0 до 7, причем 7 — наивысший приоритет), Станция может воспользоваться маркером, только если у нее есть кадры для передачи с приоритетом равным или большим, чем приоритет маркера. Сетевой адаптер станции с кадрами, у которых приоритет ниже, чем приоритет маркера, не может захватить маркер, но может поместить наибольший приоритет своих ожидающих передачи кадров в резервные биты маркера, но только в том случае, если записанный в резервных битах приоритет ниже его собственного. В ре-

зультате в резервных битах приоритета устанавливается наивысший приоритет станции, которая пытается получить доступ к кольцу, но не может этого сделать из-за высокого приоритета маркера.

Станция, сумевшая захватить маркер, передает свои кадры с приоритетом маркера, а затем передает маркер следующему соседу. При этом она переписывает значение резервного приоритета в поле приоритета маркера, а резервный приоритет обнуляется. Поэтому при следующем проходе маркера по кольцу его захватит станция, имеющая наивысший приоритет.

При инициализации кольца основной и резервный приоритеты маркера устанавливаются в 0.

Хотя механизм приоритетов в технологии Token Ring имеется, но он начинает работать только в том случае, когда приложение или прикладной протокол решает его использовать. Иначе все станции будут иметь равные права доступа к кольцу, что в основном и происходит на практике, так как большая часть приложений этим механизмом не пользуется. Это связано с тем, что приоритеты кадров поддерживаются не во всех технологиях, например в сетях Ethernet они отсутствуют, поэтому приложение будет вести себя по-разному, в зависимости от технологии нижнего уровня, что нежелательно. В современных сетях приоритетность обработки кадров обычно обеспечивается коммутаторами или маршрутизаторами, которые поддерживают их независимо от используемых протоколов канального уровня.

Физический уровень технологии Token Ring

Стандарт Token Ring фирмы IBM изначально предусматривал построение связей в сети с помощью концентраторов, называемых MAU (Multistation Access Unit), или MSAU (Multi-Station Access Unit), то есть устройствами многостанционного доступа (рис. 7.15). Сеть Token Ring может включать до 260 узлов. Использование концентраторов приводит к тому, что сети Token Ring имеют физическую топологию звезда, а логическую — кольцо.

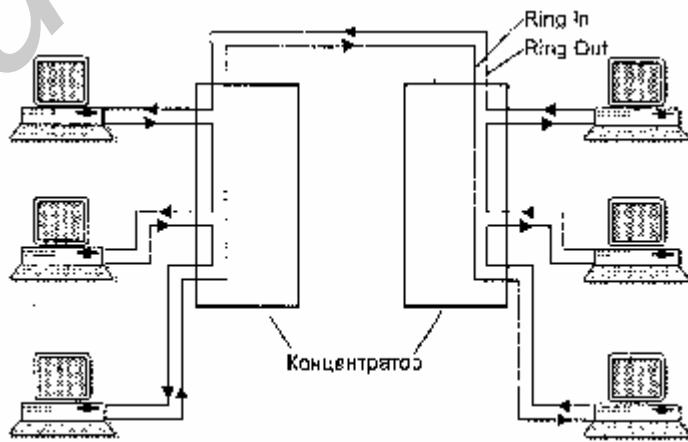


Рис. 7.15. Физическая конфигурация сети Token Ring

Концентратор Token Ring может быть активным или пассивным. Пассивный концентратор просто соединяет порты внутренними связями так, чтобы станции, подключаемые к этим портам, образовали кольцо. Ни усиление сигналов,

ни их ресинхронизацию пассивный концентратор MSAU не выполняет. Такое устройство можно считать простым кроссовым блоком за одним исключением — MSAU обеспечивает обход какого-либо порта, когда присоединенный к этому порту компьютер выключают. Такая функция необходима для обеспечения связности кольца вне зависимости от состояния подключенных компьютеров. Обычно обход порта выполняется за счет релейных схем, которые питаются постоянным током от сетевого адаптера, а при выключении сетевого адаптера нормально замкнутые контакты реле соединяют вход порта с его выходом.

Активный концентратор выполняет функции регенерации сигналов и поэтому иногда называется повторителем, как в стандарте Ethernet.

Возникает вопрос — если концентратор является пассивным устройством, то каким образом обеспечивается качественная передача сигналов на большие расстояния, которые возникают при включении в сеть нескольких сот компьютеров? Ответ состоит в том, что роль усилителя сигналов в этом случае берет на себя каждый сетевой адаптер, а роль ресинхронизирующего блока выполняет сетевой адаптер активного монитора кольца. Каждый сетевой адаптер Token Ring имеет блок повторения, который умеет регенерировать и ресинхронизировать сигналы, однако последнюю функцию выполняет в кольце только блок повторения активного монитора.

Блок ресинхронизации состоит из 30-битного буфера, который принимает манчестерские сигналы с несколько искаженными за время оборота по кольцу интервалами следования. При максимальном количестве станций в кольце (260) вариация задержки циркуляции бита по кольцу может достигать 3-битовых интервалов. Активный монитор «вставляет» свой буфер в кольцо и синхронизирует битовые сигналы, выдавая их на выход с требуемой частотой.

В общем случае сеть Token Ring имеет комбинированную звездно-кольцевую конфигурацию. Конечные узлы подключаются к MSAU по топологии звезды, а сами концентраторы MSAU объединяются через специальные порты Ring In (RI) и Ring Out (RO) для образования магистрального физического кольца.

Все станции в кольце должны работать на одной скорости — либо 4 Мбит/с, либо 16 Мбит/с. Кабели, соединяющие станцию с концентратором, называются ответвительными (lobe cable), а кабели, соединяющие концентраторы, — магистральными (trunk cable).

Технология Token Ring позволяет использовать для соединения конечных станций и концентраторов различные типы кабеля: STP Type 1, UTP Type 3, UTP Type 6, а также волоконно-оптический кабель.

При использовании экранированной витой пары STP Type 1 из номенклатуры кабельной системы IBM в кольцо допускается объединять до 260 станций при длине ответвительных кабелей до 100 м, а при использовании неэкранированной витой пары максимальное количество станций сокращается до 72 при длине ответвительных кабелей до 45 м.

Расстояние между пассивными концентраторами MSAU может достигать 100 м при использовании кабеля STP Type 1 и 45 м при использовании кабеля UTP Type 3. Между активными концентраторами MSAU максимальное расстояние увеличивается соответственно до 730 м или 365 м в зависимости от типа кабе-

ля.

Максимальная длина кольца Token Ring составляет 4000 м. Ограничения на максимальную длину кольца и количество станций в кольце в технологии Token Ring не являются такими жесткими, как в технологии Ethernet. Здесь эти ограничения во многом связаны со временем оборота маркера, по кольцу (но не только — есть и другие соображения, диктующие выбор ограничений). Так, если кольцо состоит из 260 станций, то при времени удержания маркера в 10 мс маркер вернется в активный монитор в худшем случае через 2,6 с, а это время как раз составляет тайм-аут контроля оборота маркера. В принципе все значения тайм-аутов в сетевых адаптерах узлов сети Token Ring можно настраивать, поэтому можно построить сеть Token Ring с большим количеством станций и с большей длиной кольца.

Существует большое количество аппаратуры для сетей Token Ring, которая улучшает некоторые стандартные характеристики этих сетей: максимальную длину сети, расстояние между концентраторами, надежность (путем использования двойных колец).

Выводы

- Технология Token Ring развивается в основном компанией IBM и имеет также статус стандарта IEEE 802.5, который отражает наиболее важные усовершенствования, вносимые в эту технологию.
- В сетях Token Ring используется маркерный метод доступа, который гарантирует каждой станции доступ к разделяемому кольцу в течение времени оборота маркера. Из-за этого свойства маркерный метод иногда называют детерминированным.
- Метод доступа основан на приоритетах: от 0 (низший) до 7 (высший)- Станция сама определяет приоритет текущего кадра и может захватить кольцо только в том случае, когда в кольце нет более приоритетных кадров.
- Сети Token Ring работают на двух скоростях (4 и 16 Мбит/с) и могут использовать в качестве физической среды экранированную витую пару, неэкранированную витую пару, а также волоконно-оптический кабель. Максимальное количество станций в кольце — 260, а максимальная длина кольца — 4 км.
- Технология Token Ring обладает элементами отказоустойчивости. За счет обратной связи кольца одна из станций — активный монитор — непрерывно контролирует наличие маркера, а также время оборота маркера и кадров данных. При некорректной работе кольца запускается процедура его повторной инициализации, а если она не помогает, то для локализации неисправного участка кабеля или неисправной станции используется процедура beaconing.
- Максимальный размер поля данных кадра Token Ring зависит от скорости работы кольца. Для скорости 4 Мбит/с он равен около 5000 байт, а при скорости 16 Мбит/с — около 16 Кбайт. Минимальный размер поля данных кадра не определен, то есть может быть равен 0.
- В сети Token Ring станции в кольцо объединяют с помощью концентраторов, называемых MSAU. Пассивный концентратор MSAU выполняет роль крос-

свой панели, которая соединяет выход предыдущей станции в кольце с входом последующей. Максимальное расстояние от станции до MSAU — 100 м для STP и 45 м для UTP.

- Активный монитор выполняет в кольце также роль повторителя — он ресинхронизирует сигналы, проходящие по кольцу.
- Кольцо может быть построено на основе активного концентратора MSAU, который в этом случае называют повторителем. •
- Сеть Token Ring может строиться на основе нескольких колец, разделенных мостами, маршрутизирующими кадры по принципу «от источника», для чего в кадр Token Ring добавляется специальное поле с маршрутом прохождения колец.

Технология FDDI

Технология FDDI (Fiber Distributed Data Interface — оптоволоконный интерфейс распределенных данных) — это первая технология локальных сетей, в которой средой передачи данных является волоконно-оптический кабель. Работы по созданию технологий и устройств для использования волоконно-оптических каналов в локальных сетях начались в 80-х годах, вскоре после начала промышленной эксплуатации подобных каналов в территориальных сетях. Проблемная группа X3T9.5 института ANSI разработала в период с 1986 по 1988 год начальные версии стандарта FDDI, который обеспечивает передачу кадров со скоростью 100 Мбит/с по двойному волоконно-оптическому кольцу длиной до 100 км.

Основные характеристики технологии

Технология FDDI во многом основывается на технологии Token Ring, развивая и совершенствуя ее основные идеи. Разработчики технологии FDDI ставили перед собой в качестве наиболее приоритетных следующие цели:

- повысить битовую скорость передачи данных до 100 Мбит/с;
- повысить отказоустойчивость сети за счет стандартных процедур восстановления ее после отказов различного рода — повреждения кабеля, некорректной работы узла, концентратора, возникновения высокого уровня помех на линии и т. п.;
- максимально эффективно использовать потенциальную пропускную способность сети как для асинхронного, так и для синхронного (чувствительного к задержкам) трафиков.

Сеть FDDI строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи данных между узлами сети. Наличие двух колец — это основной способ повышения отказоустойчивости в сети FDDI, и узлы, которые хотят воспользоваться этим повышенным потенциалом надежности, должны быть подключены к обоим кольцам.

В нормальном режиме работы сети данные проходят через все узлы и все уча-

стки кабеля только первичного (Primary) кольца, этот режим назван режимом Thru, то есть «сквозным», или «транзитным». Вторичное кольцо (Secondary) в этом режиме не используется.

В случае какого-либо вида отказа, когда часть первичного кольца не может передавать данные (например, обрыв кабеля или отказ узла), первичное кольцо объединяется со вторичным (рис. 7.16), вновь образуя единое кольцо. Этот режим работы сети называется Wrap, то есть «свертывание», или «сворачивание», колец. Операция свертывания производится средствами концентраторов и/или сетевых адаптеров FDDI. Для упрощения этой процедуры данные по первично-му кольцу всегда передаются в одном направлении (на диаграммах это направление изображается против часовой стрелки), а по вторичному — в обратном (изображается по часовой стрелке). Поэтому при образовании общего кольца из двух колец передатчики станций по-прежнему остаются подключенными к приемникам соседних станций, что позволяет правильно передавать и принимать информацию соседними станциями.

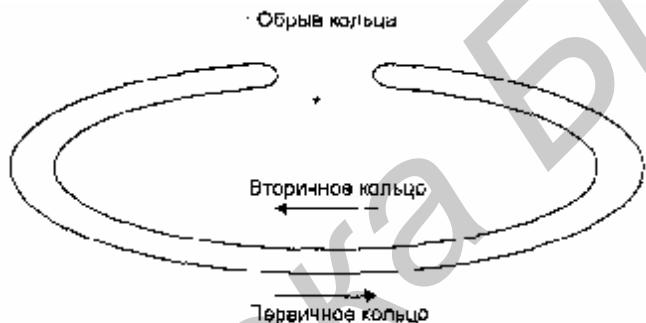


Рис. 7.16. Реконфигурация колец FDDI при отказе

В стандартах FDDI много внимания отводится различным процедурам, которые позволяют определить наличие отказа в сети, а затем произвести необходимую реконфигурацию. Сеть FDDI может полностью восстанавливать свою работоспособность в случае единичных отказов ее элементов. При множественных отказах сеть распадается на несколько не связанных сетей. Технология FDDI дополняет механизмы обнаружения отказов технологии Token Ring механизмами реконфигурации путем передачи данных в сети, основанными на наличии резервных связей, обеспечиваемых вторым кольцом.

Кольца в сетях FDDI рассматриваются как общая разделяемая среда передачи данных, поэтому для нее определен специальный метод доступа. Этот метод очень близок к методу доступа сетей Token Ring и также называется методом маркерного (или токенного) кольца — token ring.

Отличия метода доступа заключаются в том, что время удержания маркера в сети FDDI не является постоянной величиной, как в сети Token Ring. Это время зависит от загрузки кольца — при небольшой загрузке оно увеличивается, а при перегрузках может уменьшаться до нуля. Эти изменения в методе доступа касаются только асинхронного трафика, который не критичен к небольшим задержкам передачи кадров. Для синхронного трафика время удержания маркера по-прежнему остается фиксированной величиной. Механизм приоритетов кадров, аналогичный принятому в технологии Token Ring, в технологии FDDI отсутствует. Разработчики технологии решили, что деление трафика на 8 уровней

приоритетов избыточно и достаточно разделить трафик на два класса — асинхронный и синхронный, последний из которых обслуживается всегда, даже при перегрузках кольца.

В остальном пересылка кадров между станциями кольца на уровне MAC полностью соответствует технологии Token Ring. Станции FDDI применяют алгоритм раннего освобождения маркера, как и сети Token Ring со скоростью 16 Мбит/с.

Адреса уровня MAC имеют стандартный для технологий IEEE 802 формат. Формат кадра FDDI близок к формату кадра Token Ring, основные отличия заключаются в отсутствии полей приоритетов. Признаки распознавания адреса, копирования кадра и ошибки позволяют сохранить имеющиеся в сетях Token Ring процедуры обработки кадров станцией-отправителем, промежуточными станциями и станцией-получателем.

Рисунок 7.17 иллюстрирует соответствие структуры протоколов технологии FDDI семиуровневой модели OSI. FDDI определяет протокол физического уровня и протокол подуровня доступа к среде (MAC) канального уровня. Как и во многих других технологиях локальных сетей, в технологии FDDI используется протокол подуровня управления каналом данных LLC, определенный в стандарте IEEE 802.2. Таким образом, несмотря на то, что технология FDDI была разработана и стандартизована институтом ANSI, а не IEEE, она полностью вписывается в структуру стандартов 802.

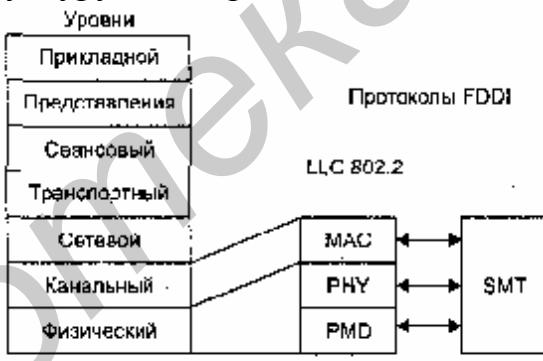


Рис. 7.17. Структура протоколов технологии FDDI

Отличительной особенностью технологии FDDI является уровень управления станцией (Station Management, SMT). Именно уровень SMT выполняет все функции по управлению и мониторингу всех остальных уровней стека протоколов FDDI. В управлении кольцом принимает участие каждый узел сети FDDI. Поэтому все узлы обмениваются специальными кадрами SMT для управления сетью.

Отказоустойчивость сетей FDDI обеспечивается протоколами и других уровней: с помощью физического уровня устраняются отказы сети по физическим причинам, например из-за обрыва кабеля, а с помощью уровня MAC — логические отказы сети, например потеря нужного внутреннего пути передачи маркера и кадров данных между портами концентратора.

Особенности метода доступа FDDI

Для передачи синхронных кадров станция всегда имеет право захватить маркер при его поступлении. При этом время удержания маркера имеет заранее заданную фиксированную величину.

Если же станции кольца FDDI нужно передать асинхронный кадр (тип кадра определяется протоколами верхних уровней), то для выяснения возможности захвата маркера при его очередном появлении станция должна измерить интервал времени, который прошел с момента предыдущего прихода маркера. Этот интервал называется временем оборота маркера (Token Rotation Time, TRT). Интервал TRT сравнивается с другой величиной — максимально допустимым временем оборота маркера по кольцу T_{0pr} . Если о технологии Token Ring максимально допустимое время оборота маркера является фиксированной величиной (2,6 с из расчета 260 станций в кольце), то в технологии FDDI станции договариваются о величине T_{0pr} во время инициализации кольца. Каждая станция может предложить свое значение T_{0pr} , в результате для кольца устанавливается минимальное из предложенных станциями времен. Это позволяет учитывать потребности приложений, работающих на станциях. Обычно синхронным приложениям (приложениям реального времени) нужно чаще передавать данные в сеть небольшими порциями, а асинхронным приложениям лучше получать доступ к сети реже, но большими порциями. Предпочтение отдается станциям, передающим синхронный трафик.

Таким образом, при очередном поступлении маркера для передачи асинхронного кадра сравнивается фактическое время оборота маркера TRT с максимально возможным значением T_{0pr} . Если кольцо не перегружено, то маркер приходит раньше, чем истекает интервал T_{0pr} , то есть $TRT < T_{0pr}$. В этом случае станции разрешается захватить маркер и передать свой кадр (или кадры) в кольцо. Время удержания маркера ТНГ равно разности $T_{0pr} - TRT$, и в течение этого времени станция передает в кольцо столько асинхронных кадров, сколько успеет.

Если же кольцо перегружено и маркер опоздал, то интервал TRT будет больше T_{0pr} . В этом случае станция не имеет права захватить маркер для асинхронного кадра. Если все станции в сети хотят передавать только асинхронные кадры, а маркер сделал оборот по кольцу слишком медленно, то все станции пропускают маркер в режиме повторения, маркер быстро делает очередной оборот, и на следующем цикле работы станции уже имеют право захватить маркер и передать свои кадры.

Метод доступа FDDI для асинхронного трафика является адаптивным и хорошо регулирует временные перегрузки сети.

Отказоустойчивость технологии FDDI

Для обеспечения отказоустойчивости в стандарте FDDI предусмотрено создание двух оптоволоконных колец — первичного и вторичного. В стандарте FDDI допускается два вида подсоединения станций к сети. Одновременное подключение к первичному и вторичному кольцам называется двойным подключением (Dual Attachment, DA). Подключение только к первичному кольцу

называется одиночным подключением (Single Attachment, SA).

В стандарте FDDI предусмотрено наличие в сети конечных узлов — станций (Station), а также концентраторов (Concentrator). Для станций и концентраторов допустим любой вид подключения к сети — как одиночный, так и двойной. Соответственно такие устройства имеют соответствующие названия: SAS (Single Attachment Station), DAS (Dual Attachment Station), SAC (Single Attachment Concentrator) и DAC (Dual Attachment Concentrator).

Обычно, хотя это и не обязательно, концентраторы имеют двойное подключение, а станции — одиночное, как показано на рис. 7.18. Чтобы устройства легче было правильно присоединять к сети, их разъемы маркируются. Разъемы типа А и В должны быть у устройств с двойным подключением, разъем М (Master) имеется у концентратора для одиночного подключения станции, у которой ответный разъем должен иметь тип S (Slave).

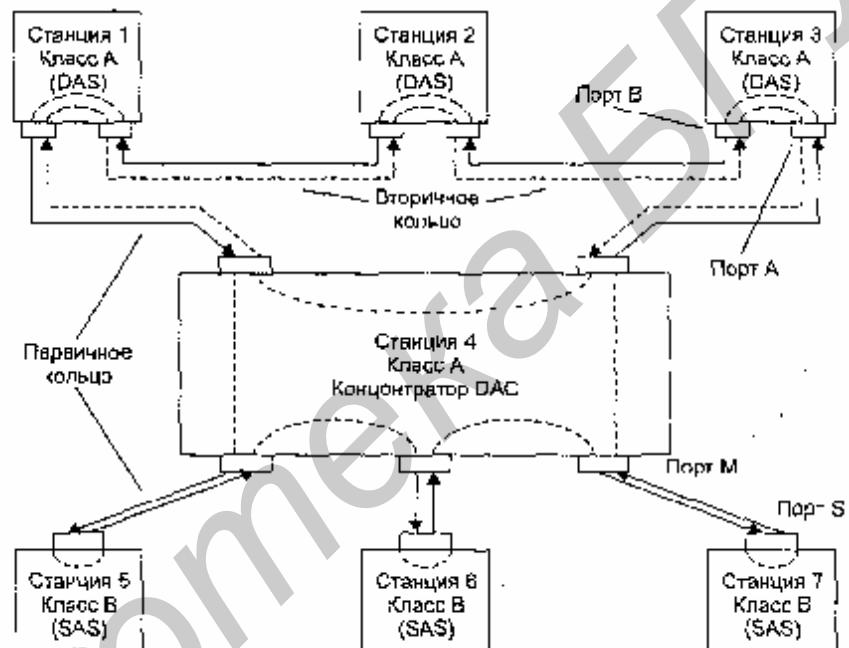


Рис. 7.18. Подключение узлов к кольцам FDDI

В случае однократного обрыва кабеля между устройствами с двойным подключением сеть FDDI сможет продолжить нормальную работу за счет автоматической реконфигурации внутренних путей передачи кадров между портами концентратора (рис. 7.19). Двукратный обрыв кабеля приведет к образованию двух изолированных сетей FDDI. При обрыве кабеля, идущего к станции с одиночным подключением, она оказывается отрезанной от сети,

а кольцо продолжает работать за счет реконфигурации внутреннего пути в концентраторе — порт М, к которому была подключена данная станция, будет исключен из общего пути.

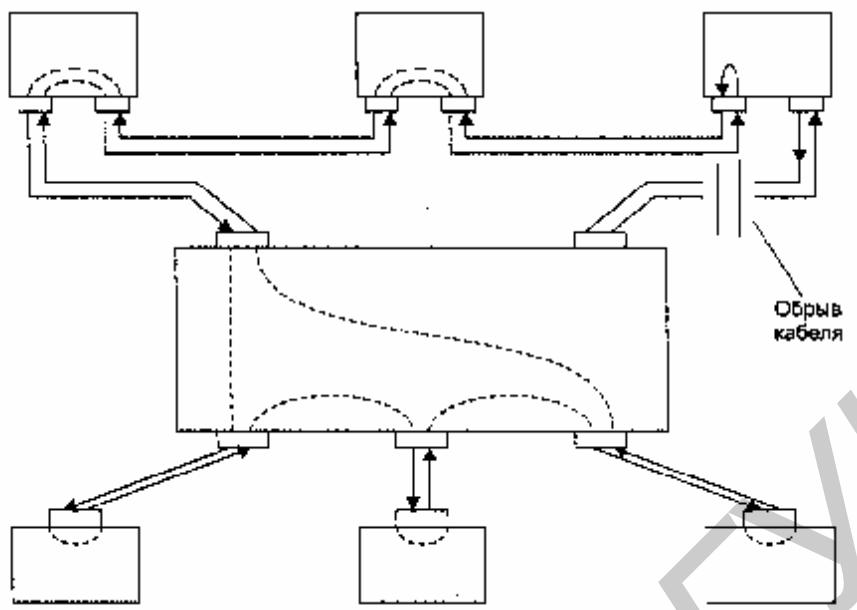


Рис. 7.19. Реконфигурация сети FDDI при обрыве провода

Для сохранения работоспособности сети при отключении питания в станциях с двойным подключением, то есть станциях DAS, последние должны быть оснащены оптическими обходными переключателями (Optical Bypass Switch), создающими обходной путь для световых потоков при исчезновении питания, которое они получают от станции.

И, наконец, станции DAS или концентраторы DAC можно подключать к двум портам М одного или двух концентраторов, создавая древовидную структуру с основными и резервными связями. По умолчанию порт В поддерживает основную связь, а порт А — резервную. Такая конфигурация называется подключением Dual Homing.

Отказоустойчивость поддерживается за счет постоянного слежения уровня SMT концентраторов и станций за временными интервалами циркуляции маркера и кадров, а также за наличием физического соединения между соседними портами в сети. В сети FDDI нет выделенного активного монитора — все станции и концентраторы равноправны, и при обнаружении отклонений от нормы они начинают процесс повторной инициализации сети, а затем и ее реконфигурации.

Реконфигурация внутренних путей в концентраторах и сетевых адаптерах выполняется специальными оптическими переключателями, которые перенаправляют световой луч и имеют достаточно сложную конструкцию.

Физический уровень технологии FDDI

В технологии FDDI для передачи световых сигналов по оптическим волокнам реализовано логическое кодирование 4B/5B в сочетании с физическим кодированием NRZI. Эта схема приводит к передаче по линии связи сигналов с тактовой частотой 125 МГц.

Так как из 32 комбинаций 5-битных символов для кодирования исходных 4-битных символов нужно только 16 комбинаций, то из оставшихся 16 выбрано

несколько кодов, которые используются как служебные. К наиболее важным служебным символам относится символ Idle (простой), который постоянно передается между портами в течение пауз между передачей кадров данных. За счет этого станции и концентраторы сети FDDI имеют постоянную информацию о состоянии физических соединений своих портов. В случае отсутствия потока символов Idle фиксируется отказ физической связи и производится реконфигурация внутреннего пути концентратора или станции, если это возможно.

При первоначальном соединении кабелем двух узлов их порты в первую очередь выполняют процедуру установления физического соединения. В этой процедуре используются последовательности служебных символов кода 4B/5B, с помощью которых создается некоторый язык команд физического уровня. Эти команды позволяют портам выяснить друг у друга типы портов (A, B, M или S) и решить, корректно ли данное соединение (например, соединение S-S является некорректным и т. п.). Если соединение корректно, то далее выполняется тест качества канала при передаче символов кодов 4B/5B, а затем проверяется работоспособность уровня MAC соединенных устройств путем передачи нескольких кадров MAC. Если все тесты прошли успешно, то физическое соединение считается установленным. Работу по установлению физического соединения контролирует протокол управления станцией SMT.

Физический уровень разделен на два подуровня: независимый от среды подуровень PHY (Physical) и зависящий от среды подуровень PMD (Physical Media Dependent) (см. рис. 7.17).

Технология FDDI в настоящее время поддерживает два подуровня PMD: для волоконно-оптического кабеля и для неэкранированной витой пары категории 5. Последний стандарт появился позже оптического и носит название TP-PMD.

Оптоволоконный подуровень PMD обеспечивает необходимые средства для передачи данных от одной станции к другой по оптическому волокну. Его спецификация определяет:

- использование в качестве основной физической среды многомодового волоконно-оптического кабеля 62,5/125 мкм;
- требования к мощности оптических сигналов и максимальному затуханию между узлами сети (для стандартного многомодового кабеля эти требования приводят к предельному расстоянию между узлами в 2 км, а для одномодового кабеля расстояние увеличивается до 10-40 км в зависимости от качества кабеля);
- требования к оптическим обходным переключателям (optical bypass switches) и оптическим приемопередатчикам;
- параметры оптических разъемов MIC (Media Interface Connector), их маркировку;
- использование для передачи света с длиной волны в 1300 нм;
- представление сигналов в оптических волокнах в соответствии с методом NRZI.

Подуровень TP-PMD определяет возможность передачи данных между станциями по витой паре в соответствии с методом физического кодирования MLT-

3, использующего два уровня потенциала: +V и -V для представления данных в кабеле. Для получения равномерного по мощности спектра сигнала данные перед физическим кодированием проходят через скрэмблер. Максимальное расстояние между узлами в соответствии со стандартом TP-PMD равно 100 м. Максимальная общая длина кольца FDDI составляет 100 км, максимальное число станций с двойным подключением в кольце — 500.

Сравнение FDDI с Ethernet и Token Ring

В табл. 7.7 представлены результаты сравнения технологии FDDI с технологиями Ethernet и Token Ring.

Таблица 7.7. Характеристики технологий FDDI, Ethernet, Token Ring

Характеристика	FDDI	Ethernet	Token Ring
Битовая скорость	100 Мбит/с	10 Мбит/с	16 Мбит/с
Топология	Двойное кольцо дерева	Шина/звезда	Звезда/кольцо
Метод доступа	Доля от времени оборота кольца	CSMA/CD	Проритетная система резервирования
Среда передачи данных	Онголовокно, неэкранированная витая пара категории 5	Толстый коаксиал, тонкий коаксиал, витая пара категории 3, оптоволокно	Экранированная или неэкранированная витая пара, оптоволокно
Максимальная длина сети (без мостов)	200 км (100 км на кольцо)	2500 м	4000 м
Максимальное расстояние между узлами	2 км (не больше 11 дБ потерь между узлами)	2500 м	100 м
Максимальное количество узлов	500 (1000 соединений)	1024	260 для первичной витой пары, 72 для неэкранированной витой пары
Тактирование и восстановление после отказов	Распределенная реализация тактирования и восстановления после отказов	Не определено	Активный монитор

Технология FDDI разрабатывалась для применения в ответственных участках сетей — на магистральных соединениях между крупными сетями, например сетями зданий, а также для подключения к сети высокопроизводительных серверов. Поэтому главным для разработчиков было обеспечить высокую скорость передачи данных, отказоустойчивость на уровне протокола и большие расстояния между узлами сети. Все эти цели были достигнуты. В результате технология FDDI получилась качественной, но весьма дорогой. Даже появление более дешевого варианта для витой пары не намного снизило стоимость подключения одного узла к сети FDDI. Поэтому практика показала, что основной областью применения технологии FDDI стали магистрали сетей, состоящих из нескольких зданий, а также сети масштаба крупного города, то есть класса MAN. Для подключения клиентских компьютеров и даже небольших серверов технология

оказалась слишком дорогой. А поскольку оборудование FDDI выпускается уже около 10 лет, значительного снижения его стоимости ожидать не приходится. В результате сетевые специалисты с начала 90-х годов стали искать пути создания сравнительно недорогих и в то же время высокоскоростных технологий, которые бы так же успешно работали на всех этажах корпоративной сети, как это делали в 80-е годы технологии Ethernet и Token Ring.

Выводы

- Технология FDDI первой использовала волоконно-оптический кабель в локальных сетях, а также работу на скорости 100 Мбит/с.
- Существует значительная преемственность между технологиями Token Ring и FDDI: для обеих характерны кольцевая топология и маркерный метод доступа.
- Технология FDDI является наиболее отказоустойчивой технологией локальных сетей. При однократных отказах кабельной системы или станции сеть за счет «сворачивания» двойного кольца в одинарное остается вполне работоспособной.
- Маркерный метод доступа FDDI работает по-разному для синхронных и асинхронных кадров (тип кадра определяет станция). Для передачи синхронного кадра станция всегда может захватить пришедший маркер на фиксированное время. Для передачи асинхронного кадра станция может захватить маркер только в том случае, если маркер выполнил оборот по кольцу достаточно быстро, что говорит об отсутствии перегрузок кольца. Такой метод доступа, во-первых, отдает предпочтение синхронным кадрам, а во-вторых, регулирует загрузку кольца, притормаживая передачу несрочных асинхронных кадров,
- В качестве физической среды технология FDDI использует волоконно-оптические кабели и UTP категории 5 (этот вариант физического уровня называется TP-PMD).
- Максимальное количество станций двойного подключения в кольце — 500, максимальный диаметр двойного кольца — 100 км. Максимальные расстояния между соседними узлами для многомодового кабеля равны 2 км, для витой пары UPT категории 5 — 100 м, а для одномодового оптоволокна зависят от его качества.

ГЛАВА 8 Развитие технологии Ethernet

Классическая 10-мегабитная сеть Ethernet устраивала большинство пользователей на протяжении около 15 лет. Однако в начале 90-х годов начала ощущаться ее недостаточная пропускная способность. Для компьютеров на процессорах Intel 80286 или 80386 с шинами ISA (8 Мбайт/с) или EISA (32 Мбайт/с) пропускная способность сегмента Ethernet составляла 1/8 или 1/32 канала «память-диск», и это хорошо согласовывалось с соотношением объемов данных, обрабатываемых локально, и данных, передаваемых по сети. Для более мощных клиентских станций с шиной PCI (133 Мбайт/с) эта доля упала до 1/133, что было явно недостаточно. Поэтому многие сегменты 10-мегабитной сети Ethernet стали перегруженными, реакция серверов в них значительно упала, а частота возникновения коллизий существенно возросла, еще более снижая полезную пропускную способность.

Назрела необходимость в разработке «новой» технологии Ethernet, то есть технологии, которая была бы такой же эффективной по соотношению цена/качество при производительности 100 Мбит/с, в результате поисков и исследований специалисты разделились на два лагеря, что, в конце концов, привело к появлению двух новых технологий — Fast Ethernet и 100VG-AnyLAN. Они отличаются степенью преемственности с классической технологией Ethernet.

Fast Ethernet

В 1992 году группа производителей сетевого оборудования, включая таких лидеров технологии Ethernet, как SynOptics, 3Com и ряд других, образовали некоммерческое объединение Fast Ethernet Alliance для разработки стандарта новой технологии, которая должна была в максимально возможной степени сохранить особенности технологии Ethernet.

Второй лагерь возглавили компании Hewlett-Packard и AT&T, которые предложили воспользоваться удобным случаем для устранения некоторых известных недостатков технологии Ethernet. Через некоторое время к этим компаниям присоединилась компания IBM, которая внесла свой вклад предложением обеспечить в новой технологии некоторую совместимость с сетями Token Ring.

В комитете 802 института IEEE в это же время была сформирована исследовательская группа для изучения технического потенциала новых высокоскоростных технологий. За период с конца 1992 года и по конец 1993 года группа IEEE изучила 100-мегабитные решения, предложенные различными производителями. Наряду с предложениями Fast Ethernet Alliance группа рассмотрела также и высокоскоростную технологию, предложенную компаниями Hewlett-Packard и AT&T.

В центре дискуссий была проблема сохранения случайного метода доступа CSMA/CD. Предложение Fast Ethernet Alliance сохраняло этот метод и тем самым обеспечивало преемственность и согласованность сетей 10 Мбит/с и 100

Мбит/с. Коалиция HP и AT&T, которая имела поддержку значительно меньшего числа производителей в сетевой индустрии, чем Fast Ethernet Alliance, предложила совершенно новый метод доступа, названный Demand Priority — приоритетный доступ по требованию. Он существенно менял картину поведения узлов в сети, поэтому не смог вписаться в технологию Ethernet и стандарт 802.3, и для его стандартизации был организован новый комитет IEEE 802.12. Осенью 1995 года обе технологии стали стандартами IEEE. Комитет IEEE 802.3 принял спецификацию Fast Ethernet в качестве стандарта 802.3и, который не является самостоятельным стандартом, а представляет собой дополнение к существующему стандарту 802.3 в виде глав с 21 по 30. Комитет 802.12 принял технологию 100VG-AnyLAN, которая использует новый метод доступа Demand Priority и поддерживает кадры двух форматов — Ethernet и Token Ring.

Физический уровень технологии Fast Ethernet

Все отличия технологии Fast Ethernet от Ethernet сосредоточены на физическом уровне (рис. 8.1). Уровни MAC и LLC в Fast Ethernet остались абсолютно теми же, и их описывают прежние главы стандартов 802.3 и 802.2. Поэтому, рассматривая технологию Fast Ethernet, мы будем изучать только несколько вариантов ее физического уровня.

Более сложная структура физического уровня технологии Fast Ethernet вызвана тем, что в ней используются три варианта кабельных систем:

- волоконно-оптический многомодовый кабель, используются два волокна;
- витая пара категории 5, используются две пары;
- витая пара категории 3, используются четыре пары.

Коаксиальный кабель, давший миру первую сеть Ethernet, в число разрешенных сред передачи данных новой технологии Fast Ethernet не попал. Это общая тенденция многих новых технологий, поскольку на небольших расстояниях витая пара категории 5 позволяет передавать данные с той же скоростью, что и коаксиальный кабель, но сеть получается более дешевой и удобной в эксплуатации. На больших расстояниях оптическое волокно обладает гораздо более широкой полосой пропускания, чем коаксиал, а стоимость сети получается ненамного выше, особенно если учесть высокие затраты на поиск и устранение неисправностей в крупной кабельной коаксиальной системе.

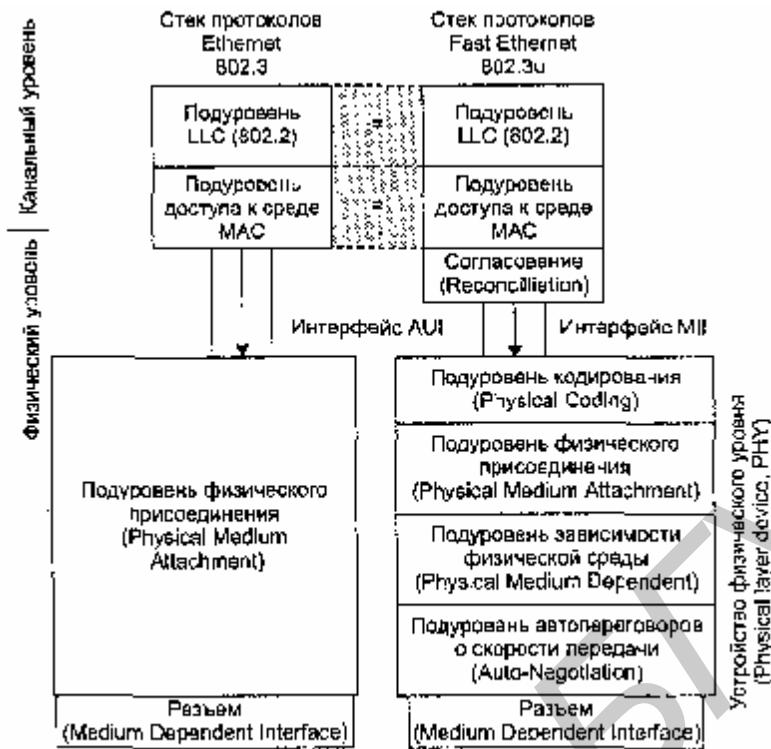


Рис. 8.1. Отличия технологии Fast Ethernet от технологии Ethernet

Отказ от коаксиального кабеля привел к тому, что сети Fast Ethernet всегда имеют иерархическую древовидную структуру, построенную на концентраторах, как и сети 10Base-T/10Base-F. Основным отличием конфигураций сетей Fast Ethernet является сокращение диаметра сети примерно до 200 м, что объясняется уменьшением времени передачи кадра минимальной длины в 10 раз за счет увеличения скорости передачи в 10-раз по сравнению с 10-мегабитной сетью Ethernet.

Тем не менее это обстоятельство не очень препятствует построению крупных сетей на технологии Fast Ethernet. Дело в том, что середина 90-х годов отмечена не только широким распространением недорогих высокоскоростных технологий, но и бурным развитием локальных сетей на основе коммутаторов. При использовании коммутаторов протокол Fast Ethernet может работать в полнодуплексном режиме, в котором нет ограничений на общую длину сети, а остаются только ограничения на длину физических сегментов, соединяющих соседние устройства (адаптер—коммутатор или коммутатор—коммутатор). Поэтому при создании магистралей локальных сетей большой протяженности технология Fast Ethernet; также активно применяется, но только в полнодуплексном варианте, совместно с коммутаторами.

В данном разделе рассматривается полудуплексный вариант работы технологии Fast Ethernet, который полностью соответствует определению метода доступа, описанному в стандарте 802.3. Особенности полнодуплексного режима Fast Ethernet описаны в разделе «Коммутаторы и полнодуплексные протоколы локальных сетей» главы 9.

По сравнению с вариантами физической реализации Ethernet (а их насчитываются шесть), в Fast Ethernet отличия каждого варианта от других глубже — ме-

няется как количество проводников, так и методы кодирования. А так как физические варианты Fast Ethernet создавались одновременно, а не эволюционно, как для сетей Ethernet, то имелась возможность детально определить те подуровни физического уровня, которые не изменяются от варианта к варианту, и те подуровни, которые специфичны для каждого варианта физической среды.

Официальный стандарт 802.3u установил три различных спецификации для физического уровня Fast Ethernet и дал им следующие названия (рис. 8.2):

- 100Base-TX для двухпарного кабеля на неэкранированной витой паре UTP категории 5 или экранированной витой паре STP Type 1;
- 100Base-T4 для четырехпарного кабеля на неэкранированной витой паре UTP категории 3, 4 или 5;
- 100Base-FX для многомодового оптоволоконного кабеля, используются два волокна.

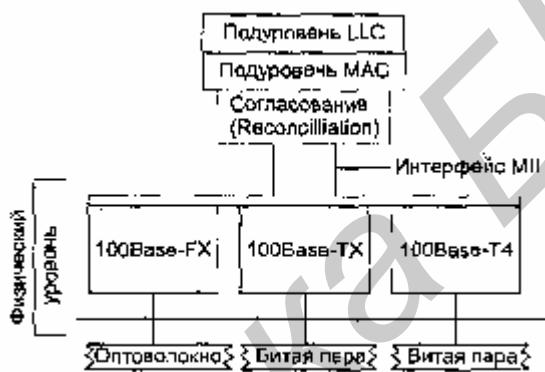


Рис. 8.2. Структура физического уровня Fast Ethernet

Для всех трех стандартов справедливы перечисленные ниже утверждения и характеристики.

- Форматы кадров технологии Fast Ethernet не отличаются от форматов кадров технологий 10-мегабитной сети Ethernet.
- Межкадровый интервал (IPG) равен 0,96 мкс, а битовый интервал — 10 нс. Все временные параметры алгоритма доступа (интервал отсрочки, время передачи кадра минимальной длины и т. п.), измеренные в битовых интервалах, остались прежними, поэтому изменения в разделы стандарта, касающиеся уровня MAC, не вносились.
- Признаком свободного состояния среды является передача по ней символа Idle соответствующего избыточного кода (а не отсутствие сигналов, как в стандартах Ethernet 10 Мбит/с).

Физический уровень включает три элемента:

- уровень согласования (reconciliation sublayer);
- независимый от среды интерфейс (Media Independent Interface, MII);
- устройство физического уровня (Physical layer device, PHY).

Уровень согласования нужен для того, чтобы уровень MAC, рассчитанный на интерфейс AUI, смог работать с физическим уровнем через интерфейс МП.

Устройство физического уровня (PHY) состоит, в свою очередь, из нескольких подуровней (см. рис. 8.2):

- подуровня логического кодирования данных, преобразующего поступающие

от уровня MAC байты в символы кода 4B/5B или 8B/6T (оба кода используются в технологии Fast Ethernet);

□ подуровней физического присоединения и подуровня зависимости от физической среды (PMD), которые обеспечивают формирование сигналов в соответствии с методом физического кодирования, например NRZI или MLT-3;

□ подуровня автопереговоров, который позволяет двум взаимодействующим портам автоматически выбрать наиболее эффективный режим работы, например полудуплексный или полнодуплексный (этот подуровень является факультативным).

Интерфейс MP поддерживает независимый от физической среды способ обмена данными между подуровнем MAC и подуровнем PHY. Этот интерфейс аналогичен по назначению интерфейсу AUI классического Ethernet за исключением того, что интерфейс AUI располагался между подуровнем физического кодирования сигнала (для любых вариантов кабеля использовался одинаковый метод физического кодирования — манчестерский код) и подуровнем физического присоединения к среде, а интерфейс MP располагается между подуровнем MAC и подуровнями кодирования сигнала, которых в стандарте Fast Ethernet три — FX, TX и T4.

Разъем MP в отличие от разъема AUI имеет 40 контактов, максимальная длина кабеля MP составляет 1 м. Сигналы, передаваемые по интерфейсу MP, имеют амплитуду 5 В.

Физический уровень 100Base-FX

Спецификация 100Base-FX (многомодовое оптоволокно, два волокна) определяет работу протокола Fast Ethernet по многомодовому оптоволокну в полудуплексном и полнодуплексном режимах на основе хорошо проверенной схемы кодирования FDDI. Как и в стандарте FDDI, каждый узел соединяется с сетью двумя оптическими волокнами, идущими от приемника (Rx) и от передатчика (Tx).

Между спецификациями 100Base-FX и 100Base-TX есть много общего, поэтому общие для двух спецификаций свойства будут даваться под обобщенным названием 100Base-FX/TX.

В то время как Ethernet со скоростью передачи 10 Мбит/с использует манчестерское кодирование для представления данных при передаче по кабелю, в стандарте Fast Ethernet определен другой метод кодирования — 4B/5B. Этот метод уже показал свою эффективность в стандарте FDDI и без изменений перенесен в спецификацию 100Base-FX/TX. При этом методе каждые 4 бита данных подуровня MAC (называемых символами) представляются 5 битами. Избыточный бит позволяет применить потенциальные коды при представлении каждого из пяти битов в виде электрических или оптических импульсов. Существование запрещенных комбинаций символов позволяет отбраковывать ошибочные символы, что повышает устойчивость работы сетей с 100Base-FX/TX.

Для отделения кадра Ethernet от символов Idle используется комбинация символов Start Delimiter — пара символов J (11000) и K (10001) кода 4B/5B, а после

завершения кадра перед первым символом Idle вставляется символ Т (рис. 8.3).



Рис. 8.3. Непрерывный поток данных спецификаций 100Base-FX/TX

После преобразования 4-битовых порций кодов MAC в 5-битовые порции физического уровня их необходимо представить в виде оптических или электрических сигналов в кабеле, соединяющем узлы сети. Спецификации 100Base-FX и 100Base-TX используют для этого различные методы физического кодирования — NRZI и MLT-3 соответственно (как и в технологии FDDI при работе через оптоволокно и витую пару).

Физический уровень 100Base-TX

В качестве среды передачи данных спецификация 100Base-TX (витая пара UTP Cat 5 или STP Type 1, две пары) использует кабель UTP категории 5 или кабель STP Type 1. Максимальная длина кабеля в обоих случаях - 100 м.

Основные отличия от спецификации 100Base-FX — использование метода MLT-3 для передачи сигналов 5-битовых порций кода 4B/5B по витой паре, а также наличие функции автопереговоров (Auto-negotiation) для выбора режима работы порта. Схема автопереговоров позволяет двум соединенным физически устройствам, которые поддерживают несколько стандартов физического уровня, отличающихся битовой скоростью и количеством витых пар, выбрать наиболее выгодный режим работы. Обычно процедура автопереговоров происходит при подсоединении сетевого адаптера, который может работать на скоростях 10 и 100 Мбит/с, к концентратору или коммутатору.

Описанная ниже схема автопереговоров сегодня является стандартом технологии 100Base-T. До этого производители применяли собственные схемы автоматического определения скорости работы взаимодействующих портов, которые не были совместимы. Принятую в качестве стандарта схему автопереговоров предложила первоначально компания National Semiconductor под названием NWay.

Всего в настоящее время определено 5 различных режимов работы, которые могут поддерживать устройства 100Base-TX или 100Base-T4 на витых парах:

- 10Base-T — 2 пары категории 3;
- 10Base-T full-duplex — 2 пары категории 3;
- 100Base-TX — 2 пары категории 5 (или Type 1A STP);
- 100Base-T4 — 4 пары категории 3;
- 100Base-TX full-duplex — 2 пары категории 5 (или Type 1A STP).

Режим 10Base-T имеет самый низкий приоритет при переговорном процессе, а полнодуплексный режим 100Base-T4 — самый высокий. Переговорный процесс происходит при включении питания устройства, а также может быть инициирован в любой момент модулем управления устройства.

Устройство, начавшее процесс автопереговоров, посыпает своему партнеру

пачку специальных импульсов Fast Link Pulse burst (FLP), в которой содержится 8-битное слово, кодирующее предлагаемый режим взаимодействия, начиная с самого приоритетного, поддерживаемого данным узлом.

Если узел-партнер поддерживает функцию автопереговоров и также может поддерживать предложенный режим, он отвечает пачкой импульсов FLP, в которой подтверждает данный режим, и на этом переговоры заканчиваются. Если же узел-партнер может поддерживать менее приоритетный режим, то он указывает его в ответе, и этот режим выбирается в качестве рабочего. Таким образом, всегда выбирается наиболее приоритетный общий режим узлов.

Узел, который поддерживает только технологию 10Base-T, каждые 16 мс посыпает манчестерские импульсы для проверки целостности линии, связывающей его с соседним узлом. Такой узел не понимает запрос FLP, который делает ему узел с функцией автопереговоров, и продолжает слать свои импульсы. Узел, получивший в ответ на запрос FLP только импульсы проверки целостности линии, понимает, что его партнер может работать только по стандарту 10Base-T, и устанавливает этот режим работы и для себя.

Физический уровень 100Base-T4

Спецификация 100Base-T4 (витая пара UTP Cat 3, четыре пары) была разработана для того, чтобы можно было использовать для высокоскоростного соединения Ethernet имеющуюся проводку на витой паре категории 3. Эта спецификация позволяет повысить общую пропускную способность за счет одновременной передачи потоков битов по всем 4 парам кабеля.

Спецификация 100Basc-T4 появилась позже других спецификаций физического уровня Fast Ethernet. Разработчики этой технологии в первую очередь хотели создать физические спецификации, наиболее близкие к спецификациям 10Base-T и 10Base-F, которые работали на двух линиях передачи данных: двух парах или двух волокнах. Для реализации работы по двум витым парам пришлось перейти на более качественный кабель категории 5.

В то же время разработчики конкурирующей технологии 100VG-AnyLAN изначально сделали ставку на работу по витой паре категории 3; самое главное преимущество состояло не столько в стоимости, а в том, что она была уже проложена в подавляющем числе зданий. Поэтому после выпуска спецификаций 100Base-TX и 100Base-FX разработчики технологии Fast Ethernet реализовали свой вариант физического уровня для витой пары категории 3.

Вместо кодирования 4B/5B в этом методе используется кодирование 8B/6T, которое обладает более узким спектром сигнала и при скорости 33 Мбит/с укладывается в полосу 16 МГц витой пары категории 3 (при кодировании 4B/5B спектр сигнала в эту полосу не укладывается). Каждые 8 бит информации уровня MAC кодируются 6-ю троичными цифрами (ternary symbols), то есть цифрами, имеющими три состояния. Каждая троичная цифра имеет длительность 40 нс. Группа из шести троичных цифр затем передается на одну из трех передающих витых пар, независимо и последовательно.

Четвертая пара всегда используется для прослушивания несущей частоты в це-

лях обнаружения коллизий. Скорость передачи данных по каждой из трех передающих пар равна 33,3 Мбит/с, поэтому общая скорость протокола 100Base-T4 составляет 100 Мбит/с. В то же время из-за принятого способа кодирования скорость изменения сигнала на каждой паре равна всего 25 Мбод, что и позволяет использовать витую пару категории 3.

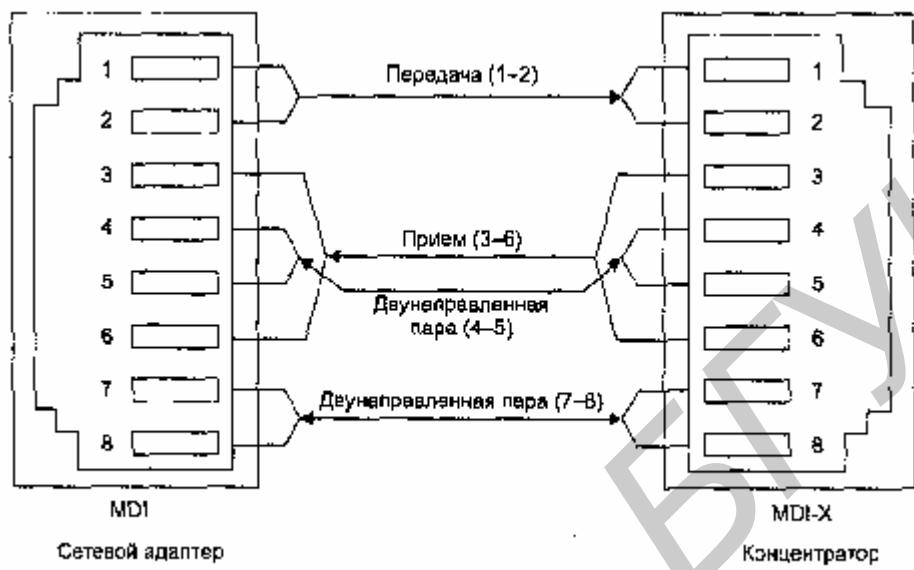


Рис. 8.4. Соединение узлов по спецификации 100Base-T4

На рис. 8.4 показано соединение порта MDI сетевого адаптера 100Base-T4 с портом MDI-X концентратора (приставка X говорит о том, что у этого разъема присоединения приемника и передатчика меняются парами кабели по сравнению с разъемом сетевого адаптера, что позволяет проще соединять пары проводов в кабеле — без перекрещивания). Пара 1-2 всегда требуется для передачи данных от порта MDI к порту MDI-X, пара 3-6 — для приема данных портом MDI от порта MDI-X, а пары 4-5 и 7-8 являются двунаправленными и используются как для приема, так и для передачи, в зависимости от потребности.

Правила построения сегментов Fast Ethernet при наличии повторителей

Технология Fast Ethernet, как и все некоаксиальные варианты Ethernet, рассчитана на использование концентраторов-повторителей для образования связей в сети.

Правила корректного построения сегментов сетей Fast Ethernet включают:

- ограничения на максимальную длину сегментов, соединяющих DTE с DTE;
- ограничения на максимальную длину сегментов, соединяющих DTE с портом повторителя;
- ограничения на максимальный диаметр сети;
- ограничения на максимальное число повторителей и максимальную длину сегмента, соединяющего повторители.

Ограничения длины сегментов DTE-DTE

В качестве DTE (Data Terminal Equipment) может выступать любой источник кадров данных для сети: сетевой адаптер, порт моста, порт маршрутизатора, модуль управления сетью и другие подобные устройства. Отличительной особенностью устройства DTE является то, что оно вырабатывает новый кадр для разделяемого сегмента (мост или коммутатор, хотя и передают через выходной порт кадр, выработанный в свое время сетевым адаптером, но для сегмента сети, к которому подключен выходной порт, этот кадр является новым). Порт повторителя не является устройством DTE, так как он побитно повторяет уже появившийся в сегменте кадр.

В типичной конфигурации сети Fast Ethernet несколько устройств DTE подключаются к портам повторителя, образуя сеть звездообразной топологии. Соединения DTE-DTE в разделяемых сегментах не встречаются (если исключить экзотическую конфигурацию, когда сетевые адаптеры двух компьютеров соединены прямо друг с другом кабелем), а вот для мостов/коммутаторов и маршрутизаторов такие соединения являются нормой — когда сетевой адаптер прямо соединен с портом одного из этих устройств либо эти устройства соединяются друг с другом.

Максимальные значения длины сегментов DTE-DTE в соответствие со спецификацией IEEE 802.3u приведены в табл. 8.1.

Таблица 8.1. Максимальные значения длины сегментов DTE-DTE

Стандарт	Тип кабеля	Максимальная длина сегмента
100Base-TX	Категория 5 UTP	100 м
100Base-FX	Мономодовое оптоволокно 62,5/125 м см	412 м (полудуплекс) 2 км (полный дуплекс)
100Base-T4	Категория 3, 4 или 5 UTP	100 м

Ограничения сетей Fast Ethernet, построенных на повторителях

Повторители Fast Ethernet делятся на два класса. Повторители класса I поддерживают все типы логического кодирования данных: как 4B/5B, так и 8B/6T. Повторители класса II поддерживают только какой-либо один тип логического кодирования — либо 4B/5B, либо 8B/6T. То есть повторители класса I позволяют выполнять трансляцию логических кодов с битовой скоростью 100 Мбит/с, а повторителям класса II эта операция недоступна.

Поэтому повторители класса I могут иметь порты всех трех типов физического уровня: 100Base-TX, 100Base-FX и 100Base-T4. Повторители класса II имеют либо все порты 100Base-T4, либо порты 100Base-TX и 100Base-FX, так как последние используют один логический код 4B/5B.

В одном домене коллизий допускается наличие только одного повторителя класса I. Это связано с тем, что такой повторитель вносит большую задержку при распространении сигналов из-за необходимости трансляции различных систем сигнализации — 70 bt.

Повторители класса II вносят меньшую задержку при передаче сигналов: 46 bt

для портов TX/FX и 33,5 м для портов T4. Поэтому максимальное число повторителей класса II в домене коллизий — 2, причем они должны быть соединены между собой кабелем не длиннее 5 м.

Небольшое количество повторителей Fast Ethernet не является серьезным препятствием при построении больших сетей, так как применение коммутаторов и маршрутизаторов делит сеть на несколько доменов коллизий, каждый из которых может строиться на одном или двух повторителях. Общая длина сети не будет иметь в этом случае ограничений.

В табл. 8.2 приведены правила построения сети на основе повторителей класса I.

Таблица 8.2. Параметры сетей на основе повторителей класса I

Тип кабелей	Максимальный диаметр сети, м	Максимальная длина сегмента, м
Только витая пара (TX)	200	100
Только оптоволокно (FX)	272	136
Несколько сегментов на витой паре и один на оптоволокне	260	100 (TX) 160 (FX)
Несколько сегментов на витой паре и несколько сегментов на оптоволокне	272	100 (TX) 136 (FX)

Эти ограничения проиллюстрированы типовыми конфигурациями сетей, показанными на рис. 8.5.

Таким образом, правило 4 хабов превратилось для технологии Fast Ethernet в правило одного или двух хабов, в зависимости от класса хаба.

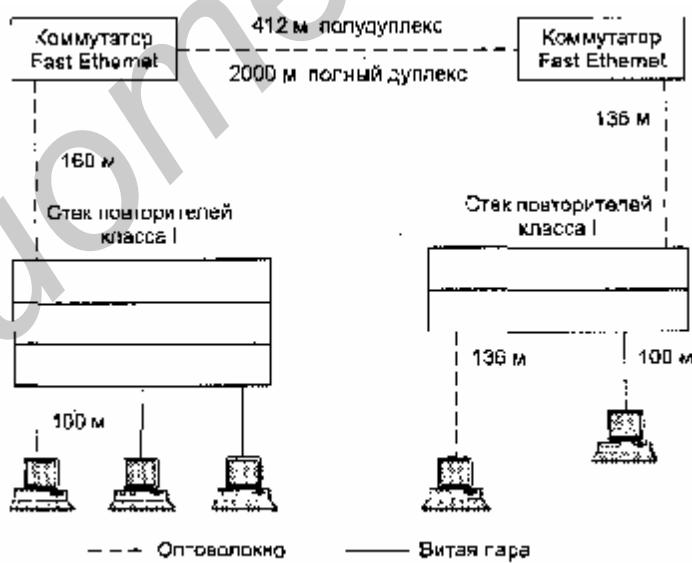


Рис. 8.5. Примеры построения сети Fast Ethernet с помощью повторителей класса I

При определении корректности конфигурации сети можно не руководствоваться правилами одного, или двух хабов, а рассчитывать время двойного оборота сети, как это было показано выше для сети Ethernet 10 Мбит/с.

Как и для технологии Ethernet 10 Мбит/с, комитет 802.3 дает исходные данные для расчета времени двойного оборота сигнала. Однако при этом сама форма

представления этих данных и методика расчета несколько изменились. Комитет предоставляет данные об удвоенных задержках, вносимых каждым элементом сети, не разделяя сегменты сети на левый, правый и промежуточный. Кроме того, задержки, вносимые сетевыми адаптерами, учитывают преамбулы кадров, поэтому время двойного оборота нужно сравнивать с величиной 512 битовых интервала (bt), то есть со временем передачи кадра минимальной длины без преамбулы.

Для повторителей класса I время двойного оборота можно рассчитать следующим образом.

Задержки, вносимые прохождением сигналов по кабелю, рассчитываются на основании данных табл. 8.3, где учтено удвоенное прохождение сигнала по кабелю.

Таблица 8.3. Задержки, вносимые кабелем

Тип кабеля	Удвоенная задержка в bt на 1 м	Удвоенная задержка на кабеле максимальной длины
UTP Cat 3	1,14 bt	114 bt (100 м)
UTP Cat 4	1,14 bt	114 bt (100 м)
UTP Cat 5	1,112 bt	111,2 bt (100 м)
STP	1,112 bt	111,2 bt (100 м)
Оптоволокно	1,0 bt	412 bt (412 м)

Задержки, которые вносят два взаимодействующих через повторитель сетевых адаптера (или порта коммутатора), берутся из табл. 8.4

Таблица 8.4. Задержки, вносимые сетевыми адаптерами

Тип сетевых адаптеров	Максимальная задержка при двойном обороте
Два адаптера TX/FX	100 bt
Два адаптера T4	136 bt
Один адаптер TX/FX и один адаптер T4	127 bt

Учитывая, что удвоенная задержка, вносимая повторителем класса I, равна 140 bt можно рассчитать время двойного оборота для произвольной конфигурации сети, естественно, учитывая максимально возможные длины непрерывных сегментов кабелей, приведенные в табл. 8.3. Если получившееся значение меньше 512, значит, по критерию распознавания коллизий сеть является корректной. Комитет 802.3 рекомендует оставлять запас в 4 bt для устойчиво работающей сети, но разрешает выбирать эту величину из диапазона от 0 до 5 bt.

Рассчитаем для примера рекомендуемую в таблице конфигурацию сети, состоящую из одного повторителя и двух оптоволоконных сегментов длиной по 136 м.

Каждый сегмент вносит задержку по 136 bt, пара сетевых адаптеров FX дает задержку в 100 bt, а сам повторитель вносит задержку в 140 bt. Сумма задержек

раина 512 бт, что говорит о том, что сеть корректна, но запас принят равным 0.

Особенности технологии 100VG-AnyLAN

Технология JOOVG-AnyLAN отличается от классической технологии Ethernet в значительно большей степени, чем Fast Ethernet. Главные отличия перечислены ниже.

- Используется метод доступа Demand Priority, который обеспечивает более справедливое распределение пропускной способности сети по сравнению с методом CSMA/CD. Кроме того, этот метод поддерживает приоритетный доступ для синхронных приложений.
- Кадры передаются не всем станциям сети, а только станции назначения.
- В сети есть выделенный арбитр доступа — концентратор, и это заметно отличает данную технологию от других, в которых применяется распределенный между станциями сети алгоритм доступа.
- Поддерживаются кадры двух технологий — Ethernet и Token Ring (именно это обстоятельство дало добавку AnyLAN в название технологии).
- Данные передаются одновременно по 4 парам кабеля UTP категории 3. По каждой паре данные передаются со скоростью 25 Мбит/с, что в сумме даст 100 Мбит/с. В отличие от Fast Ethernet в сетях 100VG-AnyLAN нет коллизий, поэтому можно использовать для передачи все четыре пары стандартного кабеля категории 3. Для кодирования данных применяется код 5Н/6В, который обеспечивает спектр сигнала в диапазоне до 16 МГц (полоса пропускания UTP категории 3) при скорости передачи данных 25 Мбит/с.

Метод доступа Demand Priority основан на передаче концентратору функций арбитра, решающего проблему доступа к разделяемой среде. Сеть 100VG-AnyLAN состоит из центрального концентратора, называемого также корневым, и соединенных с ним конечных узлов и других концентраторов (рис. 8.6).

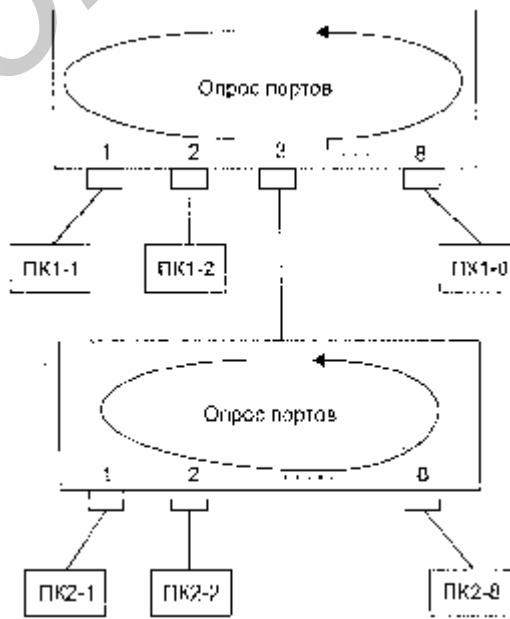


Рис. 8.6. Сеть 100VG-AnyLAN

Допускаются три уровня каскадирования. Каждый концентратор и сетевой

адаптер 100VG-AnyLAN должен быть настроен либо на работу с кадрами Ethernet, либо с кадрами Token Ring, причем одновременно циркуляция обоих типов кадров не допускается.

Концентратор циклически выполняет опрос портов. Станция, желающая передать пакет, посыпает специальный низкочастотный сигнал концентратору, запрашивая передачу кадра и указывая его приоритет. В сети 100VG-AnyLAN используется два уровня приоритетов — низкий и высокий. Низкий уровень приоритета соответствует обычным данным (файловая служба, служба печати и т. п.), а высокий приоритет соответствует данным, чувствительным к временным задержкам (например, мультимедиа). Приоритеты запросов имеют статическую и динамическую составляющие, то есть станция с низким уровнем приоритета, долго не имеющая доступа к сети, получает высокий приоритет.

Если сеть свободна, то концентратор разрешает передачу пакета. После анализа адреса получателя в принятом пакете концентратор автоматически отправляет пакет станции назначения. Если сеть занята, концентратор ставит полученный запрос в очередь, которая обрабатывается в соответствии с порядком поступления запросов и с учетом приоритетов. Если к порту подключен другой концентратор, то опрос приостанавливается до завершения опроса концентратором нижнего уровня. Станции, подключенные к концентраторам различного уровня иерархии, не имеют преимуществ по доступу к разделяемой среде, так как решение о предоставлении доступа принимается после проведения опроса всеми концентраторами опроса всех своих портов.

Остается неясным вопрос — каким образом концентратор узнает, к какому порту подключена станция назначения? Во всех других технологиях кадр просто передавался всем станциям сети, а станция назначения, распознав свой адрес, копировала кадр и буфер. Для решения этой задачи концентратор узнает MAC-адрес станции в момент физического присоединения ее к сети кабелем. Если в других технологиях процедура физического соединения выясняет связность кабеля (link test в технологии 10Base-T), тип порта (технология FDDI), скорость работы порта (процедура автопротоколов в Fast Ethernet), то в технологии 100VG-AnyLAN концентратор при установлении физического соединения выясняет MAC-адрес станции и запоминает его в таблице MAC-адресов, аналогичной таблице моста/коммутатора. Отличие концентратора 100VG-AnyLAN от моста/коммутатора в том, что у него нет внутреннего буфера для хранения кадров. Поэтому он принимает от станции сети только один кадр, отправляет его на порт назначения, и пока этот кадр не будет полностью принят станцией назначения, новые кадры концентратор не принимает. Так что эффект разделяемой среды сохраняется. Улучшается только безопасность сети - кадры не попадают на чужие порты и их труднее перехватить.

Технология 100VG-AnyLAN поддерживает несколько спецификаций физического уровня. Первоначальный вариант был рассчитан на четыре неэкранированные витые пары категорий 3, 4, 5. Позже появились варианты физического уровня, рассчитанные на две неэкранированные витые пары категории 5, две экранированные витые пары типа 1 или же два оптических многомодовых оптоволокна.

Несмотря на много хороших технических решений, технология 100VG-AnyLAN не нашла большого количества сторонников и прекратила свое существование. Она не нашла своей ниши, оказавшись слишком сложной по сравнению с более традиционной и привычной технологией Fast Ethernet. Тем более что для поддержки очень требовательных к скорости передачи данных приложений имеется технология Gigabit Ethernet, которая, сохраняя преемственность с Ethernet и Fast Ethernet, обеспечивает скорость передачи данных 1000 Мбит/с.

Выводы

- Потребности в высокоскоростной и в то же время недорогой технологии для подключения к сети мощных рабочих станций привели в начале 90-х годов к созданию инициативной группы, которая занялась поисками такой же простой и эффективной технологии, как Ethernet, но работающей на скорости 100 Мбит/с.
- Специалисты разбились на два лагеря, что, в конце концов, привело к появлению двух стандартов, принятых осенью 1995 года: комитет 802.3 утвердил стандарт Fast Ethernet, почти полностью повторяющий технологию Ethernet 10 Мбит/с, а специально созданный комитет 802.12 утвердил стандарт технологии 100VG-AnyLAN, которая сохраняла формат кадра Ethernet, но существенно изменяла метод доступа.
- Технология Fast Ethernet сохранила в неприкосновенности метод доступа CSMA/CD, оставив в нем тот же алгоритм и те же временные параметры в битовых интервалах (сам битовый интервал уменьшился в 10 раз). Все отличия Fast Ethernet от Ethernet проявляются на физическом уровне.
- В стандарте Fast Ethernet определены три спецификации физического уровня: 100Base-TX для двух пар UTP категории 5 или двух пар STP Type 1 (метод кодирования 4B/5B), 100Base-FX для многомодового волоконно-оптического кабеля с двумя оптическими волокнами (метод кодирования 4B/5B) и 100Base-T4, работающую на четырех парах UTP категории 3, но использующую одновременно только три пары для передачи, а оставшуюся — для обнаружения коллизий (метод кодирования 8B/6T).
- Стандарты 100Base-TX/FX могут работать в полнодуплексном режиме.
- Максимальный диаметр сети Fast Ethernet равен приблизительно 200 м, а более точные значения зависят от спецификации физической среды. В домене коллизий Fast Ethernet допускается не более одного повторителя класса I (позволяющего транслировать коды 4B/5B в коды 8B/6T и обратно) и не более двух повторителей класса II (не позволяющих выполнять трансляцию кодов).
- Технология Fast Ethernet при работе на витой паре позволяет за счет процедуры автопереговоров двум портам выбирать наиболее эффективный режим работы — скорость 10 Мбит/с или 100 Мбит/с, а также полудуплексный или полнодуплексный режим.
- В технологии 100VG-AnyLAN арбитром, решающим вопрос о предоставлении станциям доступа к разделяемой среде, является концентратор, поддерживающий метод Demand Priority — приоритетные требования. Метод Demand

Priority оперирует с двумя уровнями приоритетов, выставляемыми станциями, причем приоритет станции, долго не получающей обслуживания, повышается динамически.

Gigabit Ethernet

Достаточно быстро после появления на рынке продуктов Fast Ethernet сетевые интеграторы и администраторы почувствовали определенные ограничения при построении корпоративных сетей. Во многих случаях серверы, подключенные по 100-мегабитному каналу, перегружали магистрали сетей, работающие также на скорости 100 Мбит/с — магистрали FDDI и Fast Ethernet. Ощущалась потребность в следующем уровне иерархии скоростей. В 1995 году более высокий уровень скорости могли предоставить только коммутаторы ATM, а при отсутствии в то время удобных средств миграции этой технологии в локальные сети (хотя спецификация LAN Emulation — LANE и была примята в начале 1995 года, практическая ее реализация была впереди) внедрять их в локальную сеть почти никто не решался. Кроме того, технология ATM отличалась очень высокой стоимостью.

Поэтому логичным выглядел следующий шаг, сделанный IEEE, — через 5 месяцев после окончательного принятия стандарта Fast Ethernet в июне 1995 года исследовательской группе по изучению высокоскоростных технологий IEEE было предписано заняться рассмотрением возможности выработки стандарта Ethernet с еще более высокой битовой скоростью.

Летом 1996 года было объявлено о создании группы 802.3z для разработки протокола, максимально подобного Ethernet, но с битовой скоростью 1000 Мбит/с. Как и в случае Fast Ethernet, сообщение было воспринято сторонниками Ethernet с большим энтузиазмом.

Основной причиной энтузиазма была перспектива такого же плавного перевода магистралей сетей на Gigabit Ethernet, подобно тому, как были переведены на Fast Ethernet перегруженные сегменты Ethernet, расположенные на нижних уровнях иерархии сети. К тому же опыт передачи данных на гигабитных скоростях уже имелся как в территориальных сетях (технология SDH), так и в локальных — технология Fibre Channel, которая используется в основном для подключения высокоскоростной периферии к большим компьютерам и передает данные по волоконно-оптическому кабелю со скоростью, близкой к гигабитной, посредством избыточного кода 8B/10B.

В образованный для согласования усилии в этой области комитет Gigabit Ethernet Alliance с самого начала вошли такие флагманы отрасли, как Bay Networks (в настоящее время компания вошла в состав Nortel Networks), Cisco Systems и 3Com. За год своего существования количество участников Gigabit Ethernet Alliance существенно выросло и насчитывает сейчас более 100. В качестве первого варианта физического уровня был принят уровень технологии Fiber Channel, с ее кодом 8B/10B (как и в случае Fast Ethernet, когда для ускорения работ был принят отработанный физический уровень FDDI).

Первая версия стандарта была рассмотрена в январе 1997 года, а окончательно

стандарт 802.3z был принят 29 июня 1998 года на заседании комитета IEEE 802.3. Работы по реализации Gigabit Ethernet на витой паре категории 5 были переданы специальному комитету 802.3a, который уже рассмотрел, несколько вариантов проекта этого стандарта, причем с июля 1998 года проект приобрел достаточно стабильный характер. Окончательное принятие стандарта 802.3a ожидается в сентябре 1999 года.

Не дожидаясь принятия стандарта, некоторые компании выпустили первое оборудование Gigabit Ethernet на оптоволоконном кабеле уже к лету 1997 года.

Основная идея разработчиков стандарта Gigabit Ethernet состоит в максимальном сохранении идей классической технологии Ethernet при достижении битовой скорости в 1000 Мбит/с.

Так как при разработке новой технологии естественно ожидать некоторых технических новинок, идущих в общем русле развития сетевых технологий, то важно отметить, что Gigabit Ethernet, так же как и его менее скоростные собратья, на уровне протокола не будет поддерживать:

- качество обслуживания;
- избыточные связи;
- тестирование работоспособности узлом и оборудования (в последнем случае — за исключением тестирования связи порт—порт, как это делается для Ethernet, 10Base-T, 10Base-F и Fast Ethernet).

Все три названных свойства считаются весьма перспективными и полезными в современных сетях, а особенно в сетях ближайшего будущего. Почему же авторы Gigabit Ethernet отказываются от них?

По поводу качества обслуживания коротко можно отметить так: «сила есть — ума не надо». Если магистраль сети будет работать со скоростью, превышающей в 20000 раз среднюю скорость сетевой активности клиентского компьютера и в 100 раз среднюю сетевую активность сервера с сетевым адаптером 100 Мбит/с, то о задержках пакетах на магистрали во многих случаях можно небеспокоиться вообще. При небольшом коэффициенте загрузки магистрали 1000 Мбит/с очереди, в коммутаторах Gigabit, Ethernet будут небольшими, а время буферизации и коммутации на такой скорости составляет единицы и даже доли микросекунд.

Ну а если все же магистраль загрузится на достаточную величину, то приоритет чувствительному к задержкам или требовательному к средней скорости трафику можно предоставить с помощью техники приоритетов в коммутаторах — соответствующие стандарты для коммутаторов уже приняты (они будут рассматриваться в главе 10). Зато можно будет пользоваться весьма простой (почти как Ethernet) технологией, принципы работы которой известны практически всем сетевым специалистам.

Главная идея разработчиков технологии Gigabit, Ethernet состоит в том, что существует и будет существовать, весьма много сетей, в которых благодаря высокой скорости магистрали и возможности назначения пакетам приоритетов в коммутаторах можно обеспечить качество транспортного обслуживания для всех клиентов сети. И только в тех редких случаях, когда и магистраль достаточно загружена, и требования к качеству обслуживания очень жесткие, нужно

применять технологию ATM, которая действительно за счет высокой технической сложности дает гарантии качества обслуживания для всех основных видов трафика.

Избыточные связи и тестирование оборудования не будут поддерживаться технологией Gigabit Ethernet из-за того, что с этими задачами хорошо справляются протоколы более высоких уровней, например Spanning Tree, протоколы маршрутизации и т. п. Поэтому разработчики технологии решили, что нижний уровень просто должен быстро передавать данные, а более сложные и более редко встречающиеся задачи (например, приоритизация трафика) должны передаваться верхним уровням.

Что же общего имеется в технологии Gigabit Ethernet по сравнению с технологиями Ethernet и Fast Ethernet?

- Сохраняются все форматы кадров Ethernet.
- По-прежнему будут существовать полудуплексная версия протокола, поддерживающая метод доступа CSMA/CD, и полнодуплексная версия, работающая с коммутаторами. По поводу сохранения полудуплексной версии протокола сомнения были еще у разработчиков Fast Ethernet, так как сложно заставить работать алгоритм CSMA/CD на высоких скоростях. Однако метод доступа остался неизменным в технологии Fast Ethernet и его решили оставить в новой технологии Gigabit Ethernet. Сохранение недорогого решения для разделяемых сред позволит применить Gigabit Ethernet в небольших рабочих группах, имеющих быстрые серверы и рабочие станции.
- Поддерживаются все основные виды кабелей, используемых в Ethernet и Fast Ethernet: волоконно-оптический, витая пара категории 5, коаксиал.

Тем не менее разработчикам технологии Gigabit Ethernet для сохранения приведенных выше свойств пришлось внести изменения не только в физический уровень, как это было в случае Fast Ethernet, но и в уровень MAC.

Перед разработчиками стандарта Gigabit Ethernet стояло несколько трудно разрешимых проблем. Одной из них была задача обеспечения приемлемого диаметра сети для полудуплексного режима работы. В связи с ограничениями, накладываемыми методом CSMA/CD на длину кабеля, версия Gigabit Ethernet для разделяемой среды допускала бы длину сегмента всего в 25 м при сохранении размера кадров и всех параметров метода CSMA/CD неизменными. Так как существует большое количество применений, требующих диаметра сети хотя бы 200 м, необходимо было каким-то образом решить эту задачу за счет минимальных изменений в технологии Fast Ethernet.

Другой сложнейшей задачей было достижение битовой скорости 1000 Мбит/с на основных типах кабелей. Даже для оптоволокна достижение такой скорости представляет некоторые проблемы, так как технология Fibre Channel, физический уровень которой был взят за основу для оптоволоконной версии Gigabit Ethernet, обеспечивает скорость передачи данных всего в 800 Мбит/с (битовая скорость на линии равна в этом случае примерно 1000 Мбит/с, но при методе кодирования 8B/10B полезная битовая скорость на 25 % меньше скорости импульсов на линии).

И, наконец, самая сложная задача — поддержка кабеля на витой паре. Такая за-

дача на первый взгляд кажется неразрешимой — ведь даже для 100-мегабитных протоколов требуются достаточно сложные методы кодирования, чтобы уложить спектр сигнала в полосу пропускания кабеля. Однако успехи специалистов по кодированию, проявившиеся в последнее время в новых стандартах на модемы, показали, что задача имеет шансы на решение. Чтобы не тормозить принятие основной версии стандарта Gigabit Ethernet, использующего оптоволокно и коаксиал, был создан отдельный комитет 802.3az, который занимается разработкой стандарта Gigabit Ethernet на витой паре категории 5.

Все эти задачи были успешно решены.

Средства обеспечения диаметра сети в 200 м на разделяемой среде

Для расширения максимального диаметра сети Gigabit Ethernet, в полудуплексном режиме до 200 м разработчики технологии предприняли достаточно естественные меры, основывающиеся на известном соотношении времени передачи кадра минимальной длины и временем двойного оборота.

Минимальный размер кадра был увеличен (без учета преамбулы) с 64 до 512 байт или до 4096 бт. Соответственно, время двойного оборота теперь также можно было увеличить до 4095 бт, что делает допустимым диаметр сети около 200 м при использовании одного повторителя. При двойной задержке сигнала в 10 бт/м оптоволоконные кабели длиной 100 м вносят вклад во время двойного оборота по 1000 бт, и если повторитель и сетевые адаптеры будут вносить такие же задержки, как в технологии Fast Ethernet (соответствующие данные см. в предыдущем разделе), то задержка повторителя в 1000 бт и пары сетевых адаптеров в 1000 бт. дадут в сумме время двойного оборота 4000 бт, что удовлетворяет условию распознавания коллизий. Для увеличения длины кадра до требуемой в новой технологии величины сетевой адаптер должен дополнить поле данных до длины 448 байт так называемым расширением (exlention), представляющим собой поле, заполненное нулями.

Формально минимальный размер кадра не изменился, он по-прежнему равен 64 байтам или 512 битам, но это объясняется тем, что поле расширения помещается после поля контрольной суммы кадра FCS, его значение не включается в контрольную сумму и не учитывается значением поля Length/Tуре (в том случае, когда это поле переносит значения именно длины кадра). Поле Extention является просто расширением несущей, необходимым для корректного обнаружения коллизий.

Для сокращения накладных расходов при использовании слишком длинных кадров для передачи коротких квитанций разработчики стандарта разрешили конечным узлам передавать несколько кадров подряд, без передачи среды другим станциям. Такой режим получил название Burst Mode — монопольный пакетный режим. Станция может передать подряд несколько кадров с общей длиной не более 65536 бит или 8192 байт. Если станции нужно передать несколько небольших кадров, то она может не дополнять первый кадр до размера в 512

байт за счет поля Extension, а передавать несколько кадров подряд до исчерпания предела в 8192 байт (в этот предел входят все байты кадра, в том числе преамбула, заголовок, данные и контрольная сумма). Предел 8192 байт называется Burst Length. Если станция начала передавать кадр и предел Burst-Length был достигнут в середине кадра, то кадр разрешается передать до конца. Увеличение «совмещенного» кадра до 8192 байт несколько задерживает доступ к разделяемой среде других станций, но при скорости 1000 Мбит/с эта задержка не столь существенна. В полнодуплексном режиме поля Extension и Burst Mode не используются, так как распознавать коллизии в этом режиме нет необходимости, такое понятие здесь просто отсутствует.

Спецификации физической среды стандарта 802.3z

В стандарте 802.3z определены следующие типы физической среды:

- одномодовый волоконно-оптический кабель;
- многомодовый волоконно-оптический кабель 62,5/125;
- многомодовый волоконно-оптический кабель 50/125;
- двойной коаксиал с волновым сопротивлением 75 Ом.

Многомодовый кабель

Для передачи данных по традиционному для компьютерных сетей многомодовому волоконно-оптическому кабелю стандарт определяет применение излучателей, работающих на двух длинах волн: 1300 и 850 нм. Применение светодиодов с длиной волны 850 нм объясняется тем, что они намного дешевле, чем светодиоды, работающие на волне 1300 нм, хотя при этом максимальная длина кабеля уменьшается, так как затухание многомодового оптоволокна на волне 850 нм более чем в два раза выше, чем на волне 1300 нм. Однако возможность удешевления чрезвычайно важна для такой в целом дорогой технологии, как Gigabit Ethernet.

Для многомодового оптоволокна стандарт 802.3z определил спецификации 100Base-SX и 1000Base-LX.

В первом случае используется длина волны 850 нм (S означает Short Wavelength — короткая волна), а на втором — 1300 нм (L — от Long Wavelength, то есть длинная волна).

В табл. 8.5 приведены предельные значения длины сегментов волоконно-оптического кабеля разного типа для стандарта 1000Base-SX.

Таблица 8.5. Диапазоны значений длины сегментов кабеля стандарта 1000Base-SX

Тип кабеля	Погонная пропускная способность, МГц/км	Диапазон, м
62,5 мкм MMF	160	2-220
62,5 мкм MMF	900	2-275
50 мкм MMF	400	2-500
50 мкм MMF	500	2-550
10 мкм SMF	Не определено	Не поддерживается

Очевидно, что эти максимальные значения могут достигаться только для полнодуплексной передачи данных, так как время двойного оборота сигнала на двух отрезках 220 м равно 4400 бт, что превосходит предел 4095 бт: даже без учета повторителя и сетевых адаптеров. Для полудуплексной передачи максимальные значения сегментов оптоволоконного кабеля всегда должны быть меньше 100 м. Приведенные в таблице расстояния рассчитаны для худшего по стандарту случая полосы пропускания многомодового кабеля, находящегося в пределах от 160 до 500 МГц/км. Реальные кабели обычно обладают значительно лучшими характеристиками, находящимися между 600 и 1000 МГц/км. В этом случае можно увеличить длину кабеля примерно до 800 м.

Одномодовый кабель

Для спецификации 1000Base-LX в качестве источника излучения всегда применяется полупроводниковый лазер-диод с длиной волны 1300 нм.

Спецификация 1000Base-LX может работать как с многомодовым, так и с одномодовым кабелем. Предельные значения сегментов для кабелей разного типа приведены в табл. 8.6.

Таблица 8.6. Диапазоны значений длины сегментов кабеля стандарта 1000Base-LX

Тип кабеля	Погонная пропускная способность, МГц/км	Диапазон, м
62,5 мкм MMF	500	2-550
50 мкм MMF	400	2-550
50 мкм MMF	500	2-550
10 мкм SMF	Не определено	2-5000

Твинаксиальный кабель

В качестве среды передачи данных используется высококачественный твинаксиальный кабель (Twinaх) с волновым сопротивлением 150 Ом (2x75 Ом). Данные посылаются одновременно по паре проводников, каждый из которых окружен экранирующей оплеткой. При этом получается режим полудуплексной передачи. Для обеспечения полудуплексной передачи необходимы еще две пары коаксиальных проводников. Начал выпускаться специальный кабель, который содержит четыре коаксиальных проводника, — так называемый Quad-кабель. Он внешне напоминает кабель категории 5 и имеет близкие к нему внешний диаметр и гибкость. Максимальная длина твинаксиального сегмента составляет всего 25 м, поэтому это решение подходит для оборудования, расположенного в одной комнате.

Gigabit Ethernet на витой паре категории 5

Как известно, каждая пара кабеля категории 5 имеет гарантированную полосу пропускания до 100 МГц. Для передачи по такому кабелю данных со скоростью 1000 Мбит/с было решено организовать параллельную передачу одновременно по всем 4 парам кабеля (так же, как и в технологии 100VG-AnyLAN).

Это сразу уменьшило скорость передачи данных по каждой паре до 250 Мбит/с. Однако и для такой скорости необходимо было придумать метод кодирования, который имел бы спектр не выше 100 МГц. Кроме того, одновременное использование четырех пар на первый взгляд лишает сеть возможности распознавания коллизий.

На оба эти вопроса комитет 802.3аб нашел ответы.

Для кодирования данных был применен код РАМ5, использующий 5 уровней потенциала: -2, -1, 0, +1, +2. Поэтому за один такт по одной паре передается 2,322 бит информации. Следовательно, тактовую частоту вместо 250 МГц можно снизить до 125 МГц. При этом если использовать не все коды, а передавать 8 бит за такт (по 4 парам), то выдерживается требуемая скорость передачи в 1000 Мбит/с и еще остается запас неиспользуемых кодов, так как код РАМ5 содержит $5^4 = 625$ комбинаций, а если передавать за один такт по всем четырем парам 8 бит данных, то для этого требуется всего $2^6 = 256$ комбинаций. Оставшиеся комбинации приемник использует для контроля принимаемой информации и выделения правильных комбинаций на фоне шума. Код РЛМ5 на тактовой частоте 125 МГц укладывается в полосу 100 МГц кабеля категории 5.

Для распознавания коллизии и организации полнодуплексного режима разработчики спецификации 802.3аб применили технику, используемую при организации дуплексного режима на одной паре проводов в современных модемах и аппаратуре передачи данных абонентских окончаний ISDN. Вместо передачи по разным парам проводов или разнесения сигналов двух одновременно работающих навстречу друг другу передатчиков по диапазону частот оба передатчика работают навстречу друг другу по каждой из 4 пар в одном и том же диапазоне частот, так как используют один и тот же потенциальный код РАМ5 (рис. 8.7). Схема гибридной развязки H позволяет приемнику и передатчику одного и того же узла использовать одновременно витую пару и для приема и для

передачи (так же, как и в трансиверах Ethernet, на коаксиале).

Для отделения принимаемого сигнала от своего собственного приемник вычитает из результирующего сигнала известный ему свой сигнал. Естественно, что это не простая операция и для ее выполнения используются специальные цифровые сигнальные процессоры — DSP (Digital Signal Processor). Такая техника уже прошла проверку практикой, но в модемах и сетях ISDN она применялась совсем на других скоростях.

При полудуплексном режиме работы получение встречного потока данных считается коллизией, а для полнодуплексного режима работы — нормальной ситуацией.

Ввиду того что работы по стандартизации спецификации Gigabit Ethernet на некранированной витой паре категории 5 подходят к концу, многие производители и потребители надеются на положительный исход этой работы, так как в этом случае для поддержки технологии Gigabit Ethernet не нужно будет заменять уже установленную проводку категории 5 на оптоволокно или проводку категорий 6 или 7.

Основная область применения протокола Gigabit Ethernet - магистрали локальных (LAN) и городских (MAN) сетей. Наиболее часто используемый режим — полнодуплексный, обеспечивающий максимальные пропускную способность и расстояние между узлами.

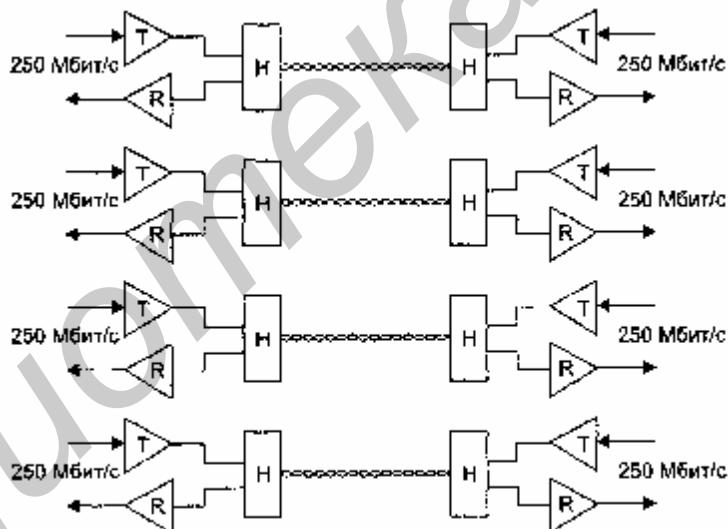


Рис. 8.7. Двунаправленная передача по четырем парам UTP категории 5

Во время написания второй редакции данной книги проводились интенсивные работы над версией стандарта Ethernet, работающего на скорости 10 Гбит/с, так называемого стандарта 10GE. Этот протокол предназначается для создания магистралей крупных локальных и городских сетей, а также магистралей глобальных сетей. Стандарт 10GE был принят комитетом IEEE 802 в июне 2002 года.

Выводы

- Технология Gigabit Ethernet добавляет новую ступень в иерархии скоростей

семейства Ethernet величиной в 1000 Мбит/с. Эта ступень позволяет эффективно строить крупные локальные сети, в которых мощные серверы и магистрали нижних уровней сети работают на скорости 100 Мбит/с, а магистраль Gigabit Ethernet объединяет их, обеспечивая достаточно большой запас пропускной способности.

- Разработчики технологии Gigabit Ethernet сохранили большую степень преемственности с технологиями Ethernet и Fast Ethernet. Gigabit Ethernet использует те же форматы кадров, что и предыдущие версии Ethernet, работает в полнодуплексном и полудуплексном режимах, поддерживая на разделяемой среде тот же метод доступа CSMA/CD с минимальными изменениями.
- Для обеспечения приемлемого максимального диаметра сети в 200 м в полу duplexном режиме разработчики технологии пошли на увеличение минимального размера кадра с 64 до 512 байт. Разрешается также передавать несколько кадров подряд, не освобождая среду, на интервале 8096 байт, тогда кадры не обязательно дополнять до 512 байт. Остальные параметры метода доступа и максимального размера кадра остались неизменными.
- Летом 1998 года был принят стандарт 802.3z, который определяет использование в качестве физической среды трех типов кабеля: многомодового оптоволоконного (расстояние до 500 м), одномодового оптоволоконного (расстояние до 5000 м) и двойного коаксиального (twinax), по которому данные передаются одновременно по двум медным экранированным проводникам на расстояние до 25 м.
- Специальная рабочая группа 802.3z разработала вариант Gigabit Ethernet на UTP категории 5. Для обеспечения скорости в 1000 Мбит/с используется одновременная передача данных по четырем неэкранированным витым парам, скорость в 250 Мбит/с обеспечивает метод кодирования РАМ-5, полнодуплексный режим поддерживается за счет встречной одновременной передачи информации по каждой паре с выделением принимаемого сигнала из общего с помощью сигнальных процессоров DSP.

ГЛАВА 9 Структуризация LAN на физическом и канальном уровнях

Крупные локальные сети не могут строиться без специального структурообразующего оборудования: кабелей, повторителей, концентраторов, мостов и коммутаторов. Некоторые из этих устройств, такие как повторители и концентраторы, предназначены для изменения физической конфигурации сети, при которой пути информационных потоков не изменяются, другие — мосты и коммутаторы влияют на структуру логических связей.

Структурированная кабельная система

Кабельная система является фундаментом любой сети: локальной, глобальной, компьютерной, телефонной или первичной. Как при строительстве нельзя создать хороший дом на плохо построенном фундаменте, так и сеть, отлично работающая на плохой кабельной системе, — это явление из области ненаучной фантастики. Если в кабелях ежедневно происходят короткие замыкания, контакты разъемов то отходят, то возвращаются на место, добавление новой станции приводит к необходимости тестирования десятка контактов разъемов из-за того, что документация на физические соединения не ведется, то ясно, что на основе такой кабельной системы любое, самое современное и производительное оборудование будет работать из рук вон плохо. Пользователи будут недовольны большими периодами простоев и низкой производительностью сети, а обслуживающий персонал пребывать в постоянном напряжении, разыскивая места коротких замыканий, обрывов и плохих контактов. Причем проблем с кабельной системой становится намного больше при увеличении размеров сети. Наиболее детально на сегодня разработаны стандарты кабельных систем зданий, при этом иерархический подход к процессу создания такой кабельной системы позволил назвать ее структурированной. На основе структурированной кабельной системы здания работает одна или несколько локальных сетей организации или подразделений одной организации, размещенной в этом здании. Процесс стандартизации кабелей территориальных сетей менее детализирован, и для этого есть весомые причины — здание является гораздо более определенным объектом, нежели город с его запутанной кабельной системой или сельская или гористая местность, где какая бы то ни было кабельная система иногда вообще отсутствует.

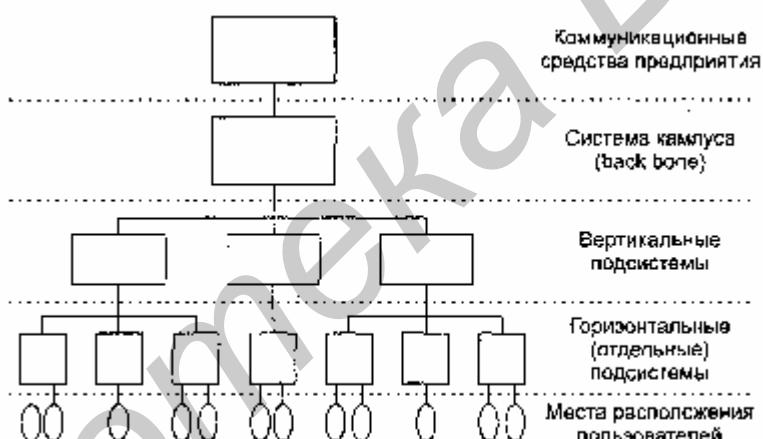
Иерархия в кабельной системе

Структурированная кабельная система (Structured Cabling System, SCS) — это набор коммутационных элементов (кабелей, разъемов, коннекторов, кроссовых панелей и шкафов), а также методика их совместного использования, которая позволяет создавать регулярные, легко расширяемые структуры связей в вычислительных сетях.

Структурированная кабельная система представляет своего рода «конструктор», с помощью которого проектировщик сети строит нужную ему конфигурацию из стандартных кабелей, соединенных стандартными разъемами и коммутируемых на стандартных кроссовых панелях. При необходимости конфигурацию связей можно легко изменить — добавить компьютер, сегмент, коммутатор, изъять ненужное оборудование, а также менять соединения между компьютерами и концентраторами.

При построении структурированной кабельной системы подразумевается, что каждое рабочее место на предприятии должно быть оснащено розетками для подключения телефона и компьютера, даже если в данный момент этого не требуется. То есть хорошая структурированная кабельная система строится избыточной. В будущем это может сэкономить средства, так как изменения и подключении новых устройств можно производить за счет перекоммутации уже проложенных кабелей.

Структурированная кабельная система планируется и строится иерархически, с главной магистралью и многочисленными ответвлениями от нее (рис. 9.1).



Эта система может быть построена на базе уже существующих современных телефонных кабельных систем, в которых кабели, представляющие собой набор витых пар, прокладываются в каждом здании, разводятся между этажами, на каждом этаже используется специальный кроссовый шкаф, от которого провода в трубах и коробах подводятся к каждой комнате и разводятся по розеткам.

К сожалению, в нашей стране далеко не во всех зданиях телефонные линии прокладываются витыми парами, поэтому они непригодны для создания компьютерных сетей, и кабельную систему в таком случае нужно строить заново.

Типичная иерархическая структура структурированной кабельной системы (рис. 9.2) включает:

- горизонтальные подсистемы (в пределах этажа);
- вертикальные подсистемы (внутри здания);
- подсистему кампуса (в пределах одной территории с несколькими зданиями).

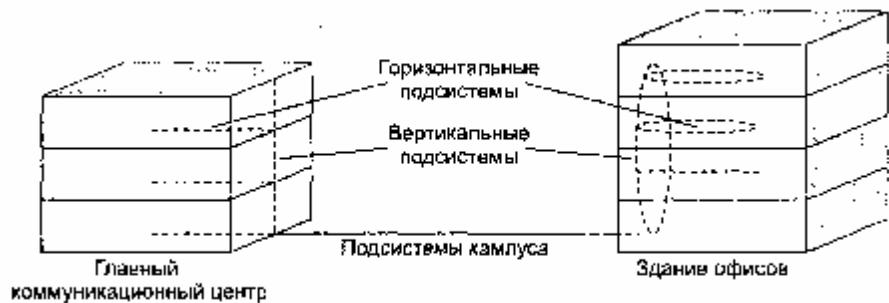


Рис. 9.2. Структура кабельных подсистем

Горизонтальная подсистема соединяет кроссовый шкаф этажа с розетками пользователей. Подсистемы этого типа соответствуют этажам здания. Вертикальная подсистема, соединяет кроссовые шкафы каждого этажа с центральной аппаратной здания. Следующим шагом иерархии является подсистема кампуса, которая соединяет несколько зданий с главной аппаратной всего кампуса. Эта часть кабельной системы обычно называется магистралью (backbone).

Использование структурированной кабельной системы вместо хаотически проложенных кабелей дает предприятию много преимуществ.

- Универсальность. Структурированная кабельная система при продуманной организации может стать единой средой для передачи компьютерных данных в локальной вычислительной сети, организации локальной телефонной сети, передачи видеинформации и даже для передачи сигналов от датчиков пожарной безопасности или охранных систем. Это позволяет автоматизировать многие процессы контроля, мониторинга и управления хозяйственными службами и системами жизнеобеспечения предприятия.
- Увеличение срока службы. Срок морального старения хорошо структурированной кабельной системы может составлять 10-15 лет.
- Уменьшение стоимости добавления, новых пользователей и изменения их мест размещения. Известно, что стоимость кабельной системы значительна и определяется в основном не стоимостью кабеля, а стоимостью работ по его прокладке. Поэтому выгоднее провести однократную работу по прокладке кабеля, возможно, с большим запасом по длине, чем несколько раз выполнять прокладку, наращивая длину кабеля. При таком подходе все работы по добавлению или перемещению пользователя сводятся к подключению компьютера к уже имеющейся розетке.
- Возможность легкого расширения сети. Структурированная кабельная система является модульной, поэтому ее легко расширять. Например, к магистрали можно добавить новую подсеть, не оказывая никакого влияния на существующие подсети. Можно заменить в отдельной подсети тип кабеля независимо от остальной части сети. Структурированная кабельная система является основой для деления сети на легко управляемые логические сегменты, так как она сама уже разделена на физические сегменты.
- Более эффективное обслуживание. Структурированная кабельная система облегчает обслуживание и поиск неисправностей по сравнению с шинной кабель-

ной системой. При шинной организации кабельной системы отказ одного из устройств или соединительных элементов приводит к трудно локализуемому отказу всей сети. В структурированных кабельных системах отказ одного сегмента не действует на другие, так как объединение сегментов осуществляется с помощью концентраторов. Концентраторы, диагностируют и локализуют неисправный участок.

□ Надежность. Структурированная кабельная система имеет повышенную надежность, поскольку производитель такой системы гарантирует не только качество ее отдельных компонентов, но и их совместимость.

Первой структурированной кабельной системой, имеющей все современные черты систем такого типа, была система SYSTIMAX SCS компании Lucent Technologies (ранее — подразделение AT&T). И сегодня компания Lucent Technologies принадлежит основная доля мирового рынка. Многие другие компании также выпускают качественные структурированные кабельные системы, например AMP, BICC Brand-Rex, Siemens, Alcatel, MOD-TAP.

Стандарты кабелей для зданий и кампусов

В компьютерных сетях применяются кабели, удовлетворяющие определенным стандартам, что позволяет строить кабельную систему сети из кабелей и соединительных устройств разных производителей. Ниже перечислены наиболее употребительные на сегодня в мировой практике стандарты.

□ Американский стандарт EIA/TIA-568A был разработан совместными усилиями нескольких организаций: ANSI, EIA/TIA и лабораторией Underwriters Labs (UL). Стандарт EIA/TIA-568A объединил предыдущую версию стандарта — EIA/TTA-568 и дополнения к нему - TSB-36 и TSB-40A.

□ Международный стандарт ISO/IEC 11801. □ Европейский стандарт EN50173. Эти стандарты близки между собой и по многим позициям предъявляют к кабелям идентичные требования. Однако есть и различия, например, в международный стандарт 11801 и европейский EN50173 пошли некоторые типы кабелей, которые отсутствуют в стандарте EIA/TIA-568A.

До появления стандарта EIA/TIA большую роль играл американский стандарт системы категорий кабелей Underwriters Labs, разработанный совместно с компанией Anixter. Позже этот стандарт вошел в стандарт EIA/TIA-568.

Кроме этих открытых стандартов многие компании в свое время разработали свои фирменные стандарты, из которых до сих пор имеет практическое значение только один — стандарт компании IBM.

При стандартизации кабелем принят протокольно-независимый подход. Это означает, что в стандарте оговариваются электрические, оптические и механические характеристики, которым должен удовлетворять тот или иной тип кабеля или соединительного изделия — разъема, кроссовой коробки и т. п. Однако для какого протокола предназначен данный кабель, стандарт не оговаривает. Поэтому нельзя приобрести кабель для протокола Ethernet или FDDI, нужно просто знать, какие типы стандартных кабелем поддерживают протоколы Ethernet и FDDI.

В ранних версиях стандартов определялись только характеристики кабелем без соединителей. В последних версиях стандартов появились требования к соединительным элементам (документы TSB-36 и TSB-40A, вошедшие затем в стандарт 568А), а также к линиям (каналам), представляющим типовую сборку элементов кабельной системы, состоящую из шнура от рабочей станции до розетки, самой розетки, основного кабеля (длиной до 90 м для витой пары), точки перехода (например, еще одной розетки или жесткого кроссового соединения) и шнура до активного оборудования, например концентратора или коммутатора. К сожалению, различные стандарты и даже различные версии одного и того же стандарта оперируют отличающимися понятиями линии и канала.

Мы остановимся только на основных требованиях к самим кабелям, не рассматривая характеристик соединительных элементов и собранных линий (это тема для отдельного пособия по кабельным системам).

В стандартах кабелей оговаривается достаточно много характеристик, из которых наиболее важные перечислены ниже (первые две из них уже были достаточно детально рассмотрены).

Основное внимание в современных стандартах уделяется кабелям на основе витой пары и волоконно-оптическим кабелям (о характеристиках оптоволокна уже было сказано в главе 4 в разделе «Типы линий связи»).

Кабели на основе неэкранированной витой пары

Медный неэкранированный кабель UTP в зависимости от электрических и механических характеристик разделяется на категории (Category I - Category 7). Кабели категорий 1 и 2 были определены в стандарте EIA/TIA-568, но в стандарт 568А уже не вошли как устаревшие. В стандарте 568А термин «категория» применяется как для компонентов кабельной системы — кабелей и разъемов, так и для линий и каналов. В международном стандарте 11801 термин «категория» употребляется только для компонентов, а линии и каналы, собранные из компонентов соответствующей категории, делятся на классы. Далее используется общий термин «категория» для обозначения соответствующего компонента — кабеля.

Кабели категории 1 применяются там, где требования к скорости передачи минимальны. Обычно это кабель для цифровой и аналоговой передачи голоса и низкоскоростной (до 20 кбит/с) передачи данных. До 1983 года это был основной тип кабеля для телефонной разводки.

Кабели категории 2 были впервые применены фирмой IBM при построении собственной кабельной системы. Главное требование к кабелям этой категории — способность передавать сигналы со спектром до 1 МГц.

Кабели категории 3 были стандартизованы в 1991 году, когда был разработан стандарт телекоммуникационных кабельных систем для коммерческих зданий (ЕТА-568), на основе которого затем был создан действующий стандарт EIA-568А. Стандарт EIA-568 определил электрические характеристики кабелей категории 3 для частот в диапазоне до 16 МГц, поддерживающих, таким образом,

высокоскоростные сетевые приложения. Кабель категории 3 предназначен как для передачи данных, так и для передачи голоса. Шаг скрутки проводов равен примерно три витка на один фут (30,5 см). Кабели категории 3 сейчас составляют основу многих кабельных систем зданий, в которых они используются для передачи и голоса, и данных.

Кабели категории 4 представляют собой несколько улучшенный вариант кабелей категории 3. Кабели категории 4 обязаны выдерживать тесты на частоте передачи сигнала 20 МГц и обеспечивать повышенную помехоустойчивость и низкие потери сигнала. Кабели категории 4 хорошо подходят для применения в системах с увеличенными расстояниями (до 135 м) и в сетях Token Ring с пропускной способностью 16 Мбит/с. На практике используются редко.

Кабели категории 5 были специально разработаны для поддержки высокоскоростных протоколов. Поэтому их характеристики определяются в диапазоне до 100 МГц. Большинство новых высокоскоростных стандартов ориентируются на использование витой пары категории 5. На этом кабеле работают протоколы со скоростью передачи данных 100 Мбит/с — FDDI (с физическим стандартом TR-PMD), Fast Ethernet, 100VG-AnyLAN, а также более скоростные протоколы — ATM на скорости 155 Мбит/с и Gigabit Ethernet на скорости 1000 Мбит/с (вариант Gigabit Ethernet на витой паре категории 5 стал стандартом в июне 1999 года). Кабель категории 5 пришел на замену кабелю категории 3, и сегодня все новые кабельные системы крупных зданий строятся именно на этом типе кабеля (в сочетании с волоконно-оптическим).

Наиболее важные электромагнитные характеристики кабеля категории 5 имеют следующие значения:

- полное волновое сопротивление в диапазоне частот до 100 МГц равно 100 Ом (стандарт ISO 11801 допускает также кабель с волновым сопротивлением 120 Ом);
- величина перекрестных наводок NEXT в зависимости от частоты сигнала должна принимать значения не менее 74 дБ на частоте 150 кГц и не менее 32 дБ на частоте 100 МГц;
- затухание имеет предельные значения от 0,8 дБ (на частоте 64 кГц) до 22 дБ (на частоте 100 МГц);
- суммарное переходное затухание на ближнем конце (PS NEXT) — не менее 27,1 дБ;
- защищенность на дальнем конце (ELFEXT) — не менее 17 дБ;
- активное сопротивление не должно превышать 9,4 Ом на 100 м;
- емкость кабеля не должна превышать 5,6 нФ на 100 м.

Существует так называемая «улучшенная» (enhanced) категория 5е, разработанная специально для поддержки протокола Gigabit Ethernet, передающего данные одновременно по всем четырем парам. Эта категория отличается несколько более жесткими требованиями к затуханию на дальнем конце, а также к некоторым другим параметрам. Однако диапазон частот, для которого предназначены кабели категории 5е, остается тем же, что и для категории 5 — до 100 МГц.

Все кабели UTP независимо от их категории выпускаются в 4-парном исполне-

нии. Каждая из четырех пар кабеля имеет определенный цвет и шаг скрутки. Обычно две пары предназначены для передачи данных, а две - для передачи голоса.

Для соединения кабелей с оборудованием используются вилки и розетки RJ-45, представляющие собой 8-контактные разъемы, похожие на обычные телефонные разъемы RJ-11.

Особое место занимают кабели категорий 6 и 7, которые промышленность начала выпускать сравнительно недавно. Для кабеля категории 6 характеристики определяются до частоты 250 МГц, а для кабелей категории 7 — до 600 МГц. Кабели категории 7 обязательно экранируются, причем как каждая пара, так и весь кабель в целом. Кабель категории 6 может быть как экранированным, так и не экранированным. Основное назначение этих кабелей — поддержка высокоскоростных протоколов на отрезках кабеля большей длины, чем кабель UTP категории 5. Некоторые специалисты сомневаются в необходимости применения кабелей категории 7, так как стоимость кабельной системы при их использовании получается соизмеримой со стоимостью сети на волоконно-оптических кабелях, характеристики которых лучше.

Кабели категорий 6 и 7 стандартизованы во втором издании спецификации ISO 11801, которое вступило в силу с третьего квартала 2002 года. В новой версии американского стандарта 568-В вводятся только кабели категории G.

Кабели на основе экранированной витой пары

Экранированная витая пара STP хорошо защищает передаваемые сигналы от внешних помех, а также меньше излучает электромагнитные колебания вовне, что, в свою очередь, защищает пользователей сетей от вредного для здоровья излучения. Наличие заземляемого экрана удорожает кабель и усложняет его прокладку, так как требует качественного заземления. Экранированный кабель применяется только для передачи данных, голос по нему не передают.

Основным стандартом, определяющим параметры экранированной витой пары, является фирменный стандарт IBM. В этом стандарте кабели делятся не на категории, а на типы: Type 1, Type 2, ..., Type 9.

Основным типом экранированного кабели является кабель Type 1 стандарта IBM. Он состоит из двух пар скрученных проводов, экранированных проводящей оплеткой, которая заземляется. Электрические параметры кабеля Type 1 примерно соответствуют параметрам кабеля UTP категории 5. Однако волновое сопротивление кабеля Type 1 равно 150 Ом (UTP категории 5 имеет полновое сопротивление 100 Ом), поэтому простое «улучшение» кабельной проводки сети путем замены неэкранированной пары UTP на STP Type 1 невозможно. Трансиверы, рассчитанные на работу с кабелем, имеющим волновое сопротивление 100 Ом, будут плохо работать на волновое сопротивление 150 Ом. Поэтому при использовании STP Type 1 необходимы соответствующие трансиверы. Такие трансиверы имеются в сетевых адаптерах Token Ring, так как эти сети разрабатывались для работы на экранированной витой паре. Некоторые другие стандарты также поддерживают кабель STP Type 1, например 100VG-

AnyLAN, а (также Fast Ethernet (хотя основным типом кабеля для Fast Ethernet является UTP категории 5). В случае если технология может использовать и UTP, и STP, нужно убедиться, что приобретаемые трансиверы соответствуют имеющемуся типу кабеля. Сегодня кабель STP Type 1 включен в стандарты EIA/TIA-568A, ISO 11801 и EN50173, то есть приобрел международный статус. Экранированные витые пары используются также в кабеле IBM Type 2, который представляет собой кабель Type 1 с добавленными двумя парами неэкранированного провода для передачи голоса.

Для присоединения экранированных кабелей к оборудованию используются разъемы конструкции IBM.

Не все типы кабелей стандарта IBM относятся к экранированным кабелям — некоторые определяют характеристики неэкранированного телефонного кабеля (Type 3) и оптоволоконного кабеля (Type 5).

Коаксиальные кабели

Существует большое количество типов коаксиальных кабелей, используемых в сетях различного типа — телефонных, телевизионных и компьютерных. Ниже приводятся основные типы и характеристики этих кабелей.

- RG-8 и RG-11 — «толстый» коаксиальный кабель, разработанный для сетей Ethernet 10Base-5. Имеет волновое сопротивление 50 Ом и внешний диаметр 0,5 дюйма (около 12 мм). Этот кабель имеет достаточно толстый внутренний проводник диаметром 2,17 мм, который обеспечивает хорошие механические и электрические характеристики (затухание на частоте 10 МГц — не хуже 18 дБ/км). Зато этот кабель сложно монтировать — он плохо гнется.
- RG-58/U. RG-58 A/U и RG-58 C/U — разновидности «тонкого» коаксиального кабеля для сетей Ethernet 10Base-2. Кабель RG-58/U имеет сплошной внутренний проводник, а кабель RG-58 A/U — многожильный. Кабель RG-58 C/U проходит «военную приемку». Все эти разновидности кабеля имеют волновое сопротивление 50 Ом, но обладают худшими механическими и электрическими характеристиками по сравнению с «толстым» коаксиальным кабелем. Тонкий внутренний проводник 0,89 мм не так прочен, зато обладает гораздо большей гибкостью, что удобно при монтаже. Затухание в этом типе кабеля выше, чем в «толстом» коаксиальном кабеле, что приводит к необходимости уменьшать длину кабеля для получения одинакового затухания в сегменте. Для соединения кабелей с оборудованием используется разъем типа BNC.
- RG-59 — телевизионный кабель с волновым сопротивлением 75 Ом. Широко применяется в кабельном телевидении.
- RG-59 — кабель с волновым сопротивлением 93 Ом, использовался в сетях ArcNet, оборудование которых сегодня практически не выпускается.

Коаксиальные кабели с волновым сопротивлением 50 Ом («тонкий» и «толстый») описаны в стандарте EIA/TIA-568. Новый стандарт EIA/TIA-568A коаксиальные кабели не описывает как морально устаревшие, для кабельной системы зданий.

Выбор типа кабеля для горизонтальных подсистем

Большинство проектировщиков начинает разработку структурированной кабельной системы с горизонтальных подсистем, так как именно к ним подключаются конечные пользователи. При этом они могут выпирать между экранированной витой парой, неэкранированной витой парой, коаксиальным кабелем и волоконно-оптическим кабелем. Возможно использование и беспроводных линий связи.

Горизонтальная подсистема характеризуется очень большим количеством ответвлений кабеля (рис. 9.3), так как его нужно провести к каждой пользовательской розетке, причем и в тех комнатах, где пока компьютеры в сеть не объединяются. Поэтому к кабелю, используемому в горизонтальной проводке, предъявляются повышенные требования к удобству выполнения ответвлений, а также удобству его прокладки в помещениях. На этаже обычно устанавливается кроссовая панель, которая позволяет с помощью коротких отрезков кабеля, оснащенного разъемами, провести перекоммутацию соединений между пользовательским оборудованием и концентраторами/коммутаторами.

Медный провод, в частности неэкранированная витая пара, является предпочтительной средой для горизонтально и кабельной подсистемы, хотя, если пользователям нужна очень высокая пропускная способность или кабельная система прокладывается в агрессивной среде, для нее подойдет и волоконно-оптический кабель. Применение коаксиального кабеля в локальной сети — это устаревшая технология, которой следует избегать, если только она уже широко не используется на предприятии.

При выборе кабеля принимаются во внимание следующие характеристики: полоса пропускания, расстояние, физическая защищенность, электромагнитная помехозащищенность, стоимость. Кроме того, при выборе кабеля нужно учитывать, какая кабельная система уже установлена на предприятии, а также какие тенденции и перспективы существуют на рынке в данный момент.

Экранированная витая пара, STP, позволяет передавать данные на большее расстояние и поддерживать больше узлов, чем неэкранированная. Наличие экрана делает ее более дорогой и не даст возможности передавать голос. Экранированная витая пара используется в основном в сетях, базирующихся на продуктах IBM и Token Ring, и редко пригодна для остального оборудования локальных сетей.

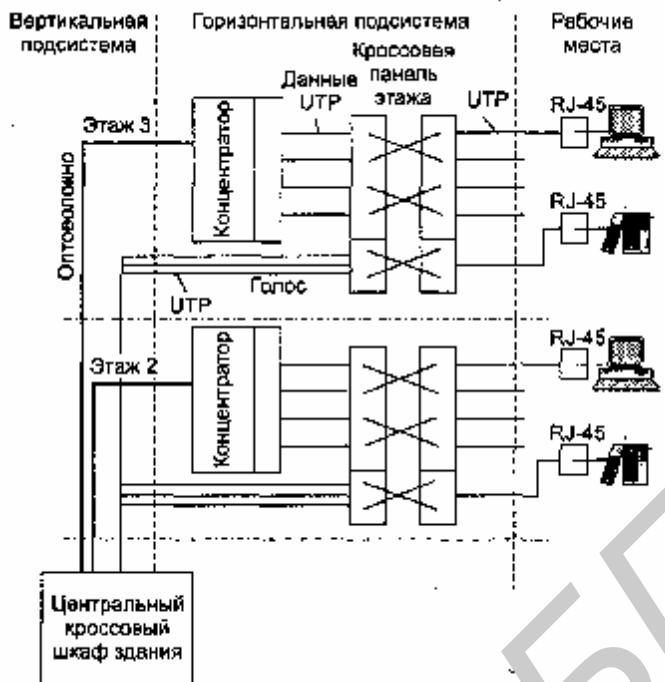


Рис. 9.3. Структура кабельной системы этажа и здания.

Неэкранированная витая пара UTP по характеристикам полосы пропускания и поддерживаемым расстояниям также подходит для создания горизонтальных подсистем. Но так как она может передавать данные и голос, она используется чаще.

Однако и коаксиальный кабель все еще остается одним из возможных вариантов кабеля для горизонтальных подсистем. Особенно в случаях, когда высокий уровень электромагнитных помех не позволяет использовать витую пару или же небольшие размеры сети не создают больших проблем с эксплуатацией кабельной системы.

Толстый Ethernet обладает по сравнению с тонким большей полосой пропускания, он более стоек к повреждениям и передаст данные на большие расстояния, однако к нему сложнее подсоединиться и он менее гибок, то есть с ним сложнее работать, и он мало подходит для горизонтальных подсистем. Однако его можно использовать в вертикальной подсистеме в качестве магистрали, если оптоволоконный кабель по каким-то причинам не подходит.

Тонкий Ethernet — это кабель, который должен был решить проблемы, связанные с применением толстого Ethernet. До появления стандарта 10Basc-T тонкий Ethernet был основным кабелем для горизонтальных подсистем. Тонкий Ethernet проще монтировать, чем толстый. Сети на тонком Ethernet можно быстро собрать, так как компьютеры соединяются друг с другом непосредственно. Главный недостаток тонкого Ethernet — сложность обслуживания. Каждый конец кабеля должен завершаться терминатором 50 Ом. При отсутствии терминатора или утере им своих рабочих свойств (например, из-за отсутствия контакта) перестает работать весь сегмент сети, подключенный к этому кабелю. Аналогичные последствия имеет плохое соединение любой рабочей станции (осуществляемое через Т-коннектор). Неисправности в сетях на тонком Ethernet слож-

но локализовать. Часто приходится отсоединять Т-коннектор от сетевого адаптера, тестируировать кабельный сегмент и затем последовательно повторять эту процедуру для всех присоединенных узлов. Поэтому стоимость эксплуатации сети на тонком Ethernet обычно значительно превосходит стоимость эксплуатации аналогичной сети на витой паре, хотя капитальные затраты на кабельную систему для тонкого Ethernet, обычно ниже.

Основные области применения оптоволоконного кабеля — вертикальная подсистема и подсистемы кампусов. Однако если нужна высокая степень защищенности данных, высокая пропускная способность или устойчивость к электромагнитным помехам, волоконно-оптический кабель может использоваться и в горизонтальных подсистемах. С волоконно-оптическим кабелем работают протоколы Apple-Talk, ArcNet, Ethernet, FDDI и Token Ring, а также новые протоколы 100VG-AnyLAN, Fast Ethernet, ATM.

Стоимость установки сетей на оптоволоконном кабеле для горизонтальной подсистемы оказывается весьма высокой. Эта стоимость складывается из стоимости сетевых адаптеров (около тысячи долларов каждый) и стоимости монтажных работ, которая в случае оптоволокна гораздо выше, чем при работе с другими видами кабеля.

МТК кабеля.

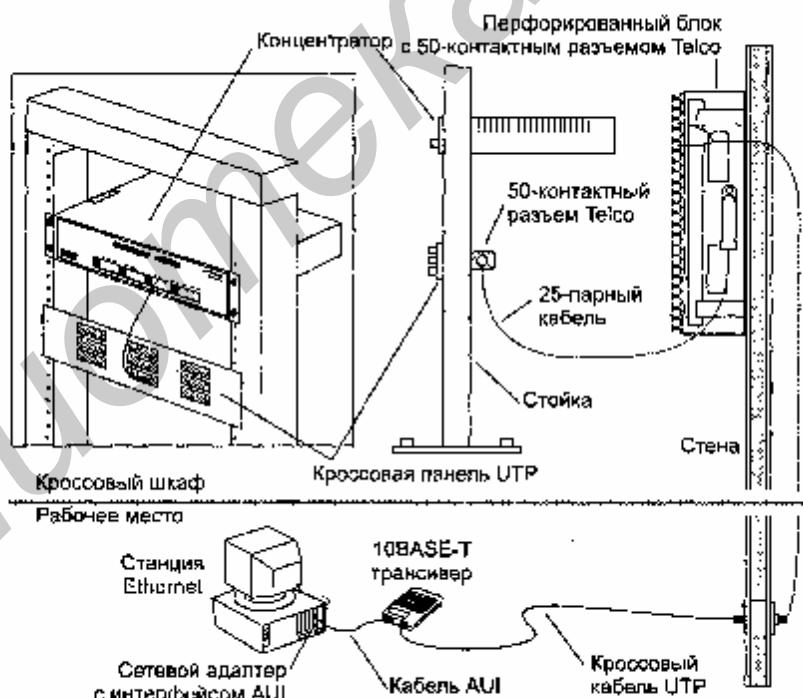


Рис. 9.4. Коммутационные элементы горизонтальной кабельной подсистемы для UTP

Преобладающим кабелем для горизонтальной подсистемы является неэкранированная витая пара категории 5 и 5e. Ее позиции еще более укрепятся с принятием спецификации 802.3az для применения на этом виде кабеля технологии Gigabit Ethernet.

На рис. 9.4 показаны типовые коммутационные элементы структурированной кабельной системы, применяемые на этаже при прокладке неэкранированной витой пары. Для сокращения количества кабелей здесь установлен 25-парный кабель и разъем для такого типа кабеля Telco, имеющий 50 контактов.

Выбор типа кабеля для вертикальных подсистем

Кабель вертикальной (или магистральной) подсистемы, которая соединяет этажи здания, должен передавать данные на большие расстояния и с большей скоростью по сравнению с кабелем горизонтальной подсистемы. В прошлом основным видом кабеля для вертикальных подсистем был коаксиал. Теперь для этой цели все чаще используется оптоволоконный кабель.

Для вертикальной подсистемы выбор кабеля в настоящее время ограничивается тремя вариантами.

- Оптоволокно — отличные характеристики пропускной способности, расстояния и защиты данных; устойчивость к электромагнитным помехам; может передавать голос, видеоизображение и данные. Но сравнительно дорого, сложно выполнять ответвления.
- Толстый коаксиал — хорошие характеристики пропускной способности, расстояния и защиты данных; может передавать данные. Но с ним сложно работать, хотя специалистов, имеющих подобный опыт работы, достаточно много.
- Широкополосный кабель, используемый в кабельном телевидении, — хорошие показатели пропускной способности и расстояния; может передавать голос, видео и данные. Но с ним очень сложно работать, требует больших затрат во время эксплуатации.

Применение волоконно-оптического кабеля в вертикальной подсистеме имеет ряд преимуществ. Он передает данные на значительно большие расстояния без необходимости регенерации сигнала. Он имеет сердечник меньшего диаметра, поэтому может быть проложен и более узких местах. Так как передаваемые по нему сигналы являются световыми, а не электрическими, оптоволоконный кабель не чувствителен к электромагнитным и радиочастотным помехам, в отличие от медного коаксиального кабеля. Это делает оптоволоконный кабель идеальной средой передачи данных для промышленных сетей. Оптоволоконному кабелю не страшна молния, поэтому он хорош для внешней прокладки. Он обеспечивает более высокую степень защиты от несанкционированного доступа, так как ответвление гораздо легче обнаружить, чем в случае медного кабеля (при ответвлении резко уменьшается интенсивность света).

Оптоволоконный кабель имеет и недостатки. Он дороже медного кабеля, дороже обходится и его прокладка. Оптоволоконный кабель менее прочный, чем коаксиальный. Инструменты, применяемые при прокладке и тестировании оптоволоконного кабеля, имеют высокую стоимость и сложны в работе. Присоединение коннекторов к оптоволоконному кабелю требует большого искусства и времени, а следовательно, и денег.

Для уменьшения стоимости построения межэтажной магистрали на оптоволокне некоторые компании, например AMP, предлагают кабельную систему с од-

ним коммутационным центром. Обычно коммутационный центр есть на каждом этаже, а в здании имеется общий коммутационный центр, соединяющий между собой коммутационные центры этажей. При такой традиционной схеме и использовании волоконно-оптического кабеля между этажами требуется выполнять достаточное большое число оптоволоконных соединений в коммутационных центрах этажей. Если же коммутационный центр в здании один, то все оптические кабели расходятся из единого кроссового шкафа прямо к разъемам конечного оборудования — коммутаторов, концентраторов или сетевых адаптеров с оптоволоконными трансиверами.

Толстый коаксиальный кабель также допустим в качестве магистрали сети, однако для новых кабельных систем более рационально использовать оптоволоконный кабель, так как он имеет больший срок службы и сможет в будущем поддерживать высокоскоростные и мультимедийные приложения. Но для уже существующих систем толстый коаксиальный кабель служил магистралью системы многие годы, и с этим нужно считаться. Причинами его повсеместного применения были широкая полоса пропускания, хорошая защищенность от электромагнитных помех и низкое радиоизлучение.

Хотя толстый коаксиальный кабель и дешевле, чем оптоволокно, с ним гораздо сложнее работать. Он особенно чувствителен к различным уровням напряжения заземления, что часто бывает при переходе от одного этажа к другому. Эту проблему сложно разрешить. Поэтому кабелем номер 1 для горизонтальной подсистемы сегодня является волоконно-оптический кабель.

Выбор типа кабеля для подсистемы кампуса

Как и для вертикальных подсистем, оптоволоконный кабель является наилучшим выбором для подсистем нескольких зданий, расположенных в радиусе нескольких километров. Для этих подсистем также подходит толстый коаксиальный кабель.

При выборе кабеля для кампуса нужно учитывать воздействие среды на кабель вне помещения. Для предотвращения поражения молнией лучше выбрать для внешней проводки неметаллический оптоволоконный кабель. По многим причинам внешний кабель производится из полиэтиленовой защитной оболочки высокой плотности. При подземной прокладке кабель должен иметь специальную влагозащитную оболочку (от дождя и подземной влаги), а также металлический защитный слой от грызунов и вандалов. Влагозащитный кабель имеет прослойку из инертного газа между диэлектриком, экраном и внешней оболочкой.

Кабель для внешней прокладки не подходит для прокладки внутри зданий, так как он выделяет при сгорании большое количество дыма.

Выводы

- Кабельная система составляет фундамент любой компьютерной сети. От ее качества зависят ее основные свойства сети.

- Структурированная кабельная система представляет собой набор коммуникационных элементов — кабелей, разъемов, коннекторов, кроссовых панелей и шкафов, которые удовлетворяют стандартам и позволяют создавать регулярные, легко расширяемые структуры связей.
- Структурированная кабельная система состоит из трех подсистем: горизонтальной (в пределах этажа), вертикальной (между этажами) и подсистемы кампуса (в пределах одной территории с несколькими зданиями).
- Для горизонтальной подсистемы характерно наличие большого количества ответвлений и перекрестных связей. Наиболее подходящий тип кабеля - неэкранированная витая пара категории 5.
- Вертикальная подсистема состоит из более протяженных отрезков кабеля, количество ответвлений намного меньше, чем в горизонтальной подсистеме. Предпочтительный тип кабеля — волоконно-оптический.
- Для подсистемы кампуса характерна нерегулярная структура связей с центральным зданием. Предпочтительный тип кабеля — волоконно-оптический и специальной изоляции.
- Кабельная система здания строится избыточной, так как стоимость последующего расширения кабельной системы превосходит стоимость установки избыточных элементов.

Сетевые адAPTERЫ И КОНЦЕНТРАТОРЫ

Концентраторы вместе с сетевыми адаптерами, и также кабельной системой представляют тот минимум оборудования, с помощью которого можно создать локальную сеть. Такая сеть будет представлять собой разделяемую среду. Понятно, что сеть не может быть слишком большой, так как при большом количестве узлов общая среда передачи данных быстро становится узким местом, снижающим производительность сети. Поэтому концентраторы и сетевые адAPTERЫ позволяют строить небольшие базовые фрагменты сетей, которые затем должны объединяться друг с другом с помощью мостов, коммутаторов и маршрутизаторов.

ФУНКЦИИ И ХАРАКТЕРИСТИКИ СЕТЕВЫХ АДAPTERОВ

Сетевой адаптер (Network Interface Card, NIC) вместе со своим драйвером реализует второй, канальный уровень модели открытых систем в конечном узле сети — компьютере. Более точно, в сетевой операционной системе пара адаптер и драйвер выполняет только функции физического и MAC-уровней, в то время как LLC-уровень обычно реализуется модулем операционной системы, единым для всех драйверов и сетевых адаптеров. Собственно так оно и должно быть в соответствии с моделью стека протоколов IEEE 802. Например, в ОС Windows NT уровень LLC реализуется в модуле NDIS, общем для всех драйверов сетевых адаптеров, независимо от того, какую технологию поддерживает драйвер.

Сетевой адаптер совместно с драйвером выполняют две операции: передачу и прием кадра.

Передача кадра из компьютера в кабель состоит из перечисленных ниже этапов (в зависимости от принятых методов кодирования некоторые могут отсутствовать).

- Прием кадра данных LLC через межуровневый интерфейс вместе с адресной информацией MAC-уровня. Обычно взаимодействие между протоколами внутри компьютера происходит через буферы, расположенные в оперативной памяти. Данные для передачи в сеть помещаются в эти буферы протоколами верхних уровней, которые извлекают их из дисковой памяти либо из файлового кэша с помощью подсистемы ввода-вывода операционной системы.
- Оформление кадра данных MAC-уровня, в который инкапсулируется кадр LLC (с отброшенными флагами 0111110). Заполнение адресов назначения и источника, вычисление контрольной суммы.
- Формирование символов кодов при использовании избыточных кодов типа 4B/5B. Скрэмблирование кодов для получения более равномерного спектра сигналов. Этот этап используется не во всех протоколах — например, технология Ethernet 10 Мбит/с обходится без него.
- Выдача сигналов в кабель в соответствии с принятым линейным кодом — манчестерским, NRZI, MLT-3 и т. п.

Прием кадра из кабеля в компьютер включает следующие действия.

- Прием из кабеля сигналов, кодирующих битовый поток.
- Выделение сигналов на фоне шума. Эту операцию могут выполнять различные специализированные микросхемы или сигнальные процессоры DSP. В результате в приемнике адаптера образуется некоторая битовая последовательность, с большой степенью вероятности совпадающая с той, которая была послана передатчиком.
- Если данные перед отправкой в кабель подвергались скрэмблированию, то они пропускаются через дескремблер, после чего в адаптере восстанавливаются символы кода, посланные передатчиком.
- Проверка контрольной суммы кадра. Если она неверна, то кадр отбрасывается, а через межуровневый интерфейс наверх, протоколу LLC передается соответствующий код ошибки. Если контрольная сумма верна, то из MAC-кадра извлекается кадр LLC и передается через межуровневый интерфейс наверх, протоколу LLC. Кадр LLC помещается в буфер оперативной памяти.

Распределение обязанностей между сетевым адаптером и его драйвером стандартами не определяется, поэтому каждый производитель решает этот вопрос самостоятельно. Обычно сетевые адаптеры делятся на адаптеры для клиентских компьютеров и адаптеры для серверов.

В адаптерах для клиентских компьютеров значительная часть работы перекладывается на драйвер, тем самым адаптер оказывается проще и дешевле. Недостатком такого подхода является высокая степень загрузки центрального процессора компьютера рутинными работами по передаче кадров из оперативной памяти компьютера в сеть. Центральный процессор вынужден заниматься этой работой вместо выполнения прикладных задач пользователя.

АдAPTERы, предназначенные для серверов, обычно снабжаются собственными процессорами, которые самостоятельно выполняют большую часть работы по передаче кадров из оперативной памяти в сеть и в обратном направлении. Примером такого адаптера может служить сетевой адаптер SMS EtherPower со встроенным процессором Intel i960.

В зависимости от того, какой протокол реализует адаптер, адAPTERы делятся на Ethernet-адAPTERы, Token Ring-адAPTERы, FDDI-адAPTERы и т. д. Так как протокол Fast Ethernet позволяет за счет процедуры автопереговоров автоматически выбрать скорость работы сетевого адаптера и зависимости от возможностей концентратора, то многие адAPTERы Ethernet сегодня поддерживают две скорости работы и имеют в своем названии приставку 10/100. Это свойство некоторые производители называют автономностью.

Сетевой адаптер перед установкой в компьютер необходимо конфигурировать. При конфигурировании адаптера обычно задаются номер прерывания IRQ, используемого адаптером, номер канала прямого доступа к памяти DMA (если адаптер поддерживает режим DMA) и базовый адрес портов ввода-вывода.

Если сетевой адаптер, аппаратура компьютера и операционная система поддерживают стандарт Plug-and-Play, то конфигурирование адаптера и его драйвера осуществляется автоматически. В противном случае нужно сначала сконфигурировать сетевой адаптер, а затем повторить параметры его конфигурации для драйвера. В общем случае, детали процедуры конфигурирования сетевого адаптера и его драйвера во многом зависят от производителя адаптера, а также от возможностей шины, для которой разработан адаптер.

Классификация сетевых адAPTERов

В качестве примера классификации адAPTERов используем подход фирмы 3Com, имеющей репутацию лидера в области адAPTERов Ethernet. Фирма 3Com считает, что сетевые адAPTERы Ethernet прошли в своем развитии три поколения.

АдAPTERы первого поколения были выполнены на дискретных логических микросхемах, в результате чего обладали низкой надежностью. Они имели буферную память только на один кадр, что приводило к низкой производительности адаптера, так как все кадры передавались из компьютера в сеть или из сети в компьютер последовательно. Кроме этого, конфигурирование адаптера первого поколения происходило вручную, с помощью перемычек. Для каждого типа адAPTERов использовался свой драйвер, причем интерфейс между драйвером и сетевой операционной системой не был стандартизирован.

В сетевых адAPTERах второго поколения для повышения производительности стали применять метод многокадровой буферизации. При этом следующий кадр загружается из памяти компьютера в буфер адаптера одновременно с передачей предыдущего кадра в сеть. В режиме приема, после того как адаптер полностью принял один кадр, он может начать передавать этот кадр из буфера в память компьютера одновременно с приемом другого кадра из сети.

В сетевых адAPTERах второго поколения широко используются микросхемы с

высокой степенью интеграции, что попытает надежность адаптеров. Кроме того, драйверы этих адаптеров основаны на стандартных спецификациях. АдAPTERЫ второго поколения обычно поставляются с драйверами, работающими как в стандарте NDIS (спецификация интерфейса сетевого драйвера), разработанном фирмами 3Com и Microsoft и одобренном IBM, так и в стандарте ODI (интерфейс открытого драйвера), разработанном фирмой Novell.

В сетевых адаптерах третьего поколения реализована конвейерная схема обработки кадров. Она заключается в том, что процессы приема кадра из оперативной памяти компьютера и передачи его в сеть совмещаются во времени. Таким образом, после приема нескольких первых байтов кадра начинается их передача. Это существенно (на 25-55 %) повышает производительность цепочки оперативная память — адаптер — физический канал — адаптер — оперативная память. Такая схема очень чувствительна к порогу начала передачи, то есть к количеству байтов кадра, которое загружается в буфер адаптера перед началом передачи в сеть. Сетевой адаптер третьего поколения осуществляет самонастройку этого параметра путем анализа рабочей среды, а также методом расчета, без участия администратора сети. Самонастройка обеспечивает максимально возможную производительность для конкретного сочетания производительности внутренней шины компьютера, его системы прерываний и системы прямого доступа к памяти.

Адаптеры третьего поколения базируются на специализированных интегральных схемах (ASIC), что повышает производительность и надежность адаптера при одновременном снижении его стоимости. Компания 3Com назвала свою технологию конвейерной обработкой кадров Parallel Tasking, другие компании также реализовали похожие схемы в своих адаптерах. Повышение производительности канала «адаптер—память» очень важно для повышения производительности сети в целом, так как производительность сложного маршрута обработки кадров, включающего, например, концентраторы, коммутаторы, маршрутизаторы, глобальные каналы связи и т. п., всегда определяется производительностью самого медленного элемента этого маршрута. Следовательно, если сетевой адаптер сервера или клиентского компьютера работает медленно, никакие быстрые коммутаторы не смогут повысить скорость работы сети.

Выпускаемые сегодня сетевые адаптеры можно отнести к четвертому поколению. В эти адаптеры обязательно входит интегральная схема ASIC, выполняющая функции MAC-уровня, а также большое количество высокоуровневых функций. В набор таких функций может входить поддержка агента удаленного мониторинга RMON, схема приоритезации кадров, функции дистанционного управления компьютером и т. п. В серверных вариантах адаптеров почти обязательно наличие мощного процессора, разгружающего центральный процессор.

Основные и дополнительные функции концентраторов

Практически во всех современных технологиях локальных сетей определено устройство, которое имеет несколько равноправных названий — концентратор

(concentrator), хаб (hub), повторитель (repeater). В зависимости от области применения этого устройства в значительной степени изменяется состав его функций и конструктивное исполнение. Неизменной остается только основная функция — повторение кадра либо на всех портах (как определено в стандарте Ethernet), либо только на некоторых портах, в соответствии с алгоритмом, определенным соответствующим стандартом.

Концентратор обычно имеет несколько портов, к которым с помощью отдельных физических сегментов кабеля подключаются конечные узлы сети — компьютеры. Концентратор объединяет отдельные физические сегменты сети в единую разделяемую среду, доступ к которой осуществляется в соответствии с одним из рассмотренных протоколов локальных сетей — Ethernet, Token Ring и т. д. Так как логика доступа к разделяемой среде существенно зависит от технологии, то для каждого типа технологии выпускаются свои концентраторы — Ethernet; Token Ring; FDDI и 100VG-AnyLAN. Для конкретного протокола иногда используется свое, узкоспециализированное название этого устройства, более точно отражающее его функции или же использующееся в силу традиции, например, для концентраторов Token Ring характерно название MSAU.

Каждый концентратор выполняет некоторую основную функцию, определенную в соответствующем протоколе той технологии, которую он поддерживает. Хотя эта функция достаточно детально определена в стандарте технологии, при ее реализации концентраторы разных производителей могут отличаться такими деталями, как количество портов, поддержка нескольких типов кабелей и т. п.

Кроме основной функции концентратор может выполнять некоторое количество дополнительных функций, которые либо в стандарте вообще не определены, либо являются факультативными. Например, концентратор Token Ring может выполнять функцию отключения некорректно работающих портов и перехода на резервное кольцо, хотя в стандарте такие его возможности не описаны. Концентратор оказался удобным устройством для выполнения дополнительных функций, облегчающих контроль и эксплуатацию сети.

Рассмотрим особенности реализации основной функции концентратора на примере концентраторов Ethernet:

В технологии Ethernet устройства, объединяющие несколько физических сегментов коаксиального кабеля в единую разделяемую среду, использовались давно и получили название «повторителей» по своей основной функции — повторению на всех своих портах сигналов, полученных на входе одного из портов. В сетях на основе коаксиального кабеля обычными являлись двухпортовые повторители, соединяющие только два сегмента кабеля, поэтому термин концентратор к ним обычно не применялся.

С появлением спецификации 10Base-T для витой пары повторитель стал неотъемлемой частью сети Ethernet, так как без него связь можно было организовать только между двумя узлами сети. Многопортовые повторители Ethernet на витой паре стали называть концентраторами или хабами, так как в одном устройстве действительно концентрировались связи между большим количеством узлов сети. Концентратор Ethernet обычно имеет от 8 до 72 портов, причем основная часть портов предназначена для подключения кабелей на витой паре. На

рис. 9.5 показан типичный концентратор Ethernet, рассчитанный на образование небольших сегментов разделяемой среды. Он имеет 16 портов стандарта 10Base-T с разъемами RJ-45, а также один порт AUI для подключения внешнего трансивера. Обычно к этому порту подключается трансивер, работающий на коаксиал или оптоволокно. С помощью этого трансивера концентратор подключается к магистральному кабелю, соединяющему несколько концентраторов между собой. Таким же образом обеспечивается подключение станции, удаленной от концентратора более чем на 100 м.

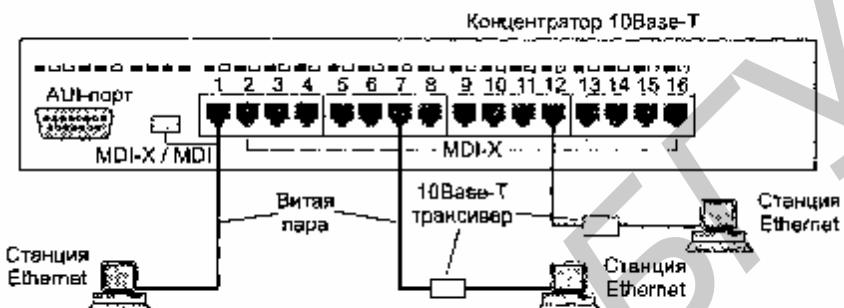


Рис. 9.5. Концентратор Ethernet

Для соединения концентратором технологии 10Base-T между собой в иерархическую систему коаксиальный или оптоволоконный кабель не обязателен, можно применять те же порты, что и для подключения конечных станций, с учетом одного обстоятельства. Дело в том, что обычный порт RJ-45, предназначенный для подключения сетевого адаптера и называемый MDI-X (кроссированный MDI), имеет инвертированную разводку контактов разъема, чтобы сетевой адаптер можно было подключить к концентратору с помощью стандартного соединительного кабеля, не кроссирующего контакты (рис. 9.6). В случае соединения концентраторов через стандартный порт MDI-X приходится использовать нестандартный кабель с перекрестным соединением пар. Поэтому некоторые изготовители снабжают концентратор выделенным портом MDI, в котором нет кrossирования пар. Таким образом, два концентратора можно соединить обычным некросированным кабелем, если это делать через порт MDI-X одного концентратора и порт MDI второго. Чаще один и тот же порт концентратора может работать и как порт MDI-X, и как порт MDI, в зависимости от положения кнопочного переключателя, как это показано в нижней части рис. 9.6.

Многопортовый повторитель-концентратор Ethernet может по-разному рассматриваться при использовании правила 4 хабов. В большинстве моделей все порты связаны с единственным блоком повторения, и при прохождении сигнала между двумя портами повторителя блок повторения вносит задержку всего один раз. Поэтому такой концентратор нужно считать одним повторителем с ограничениями, накладываемыми правилом 4 хабов. Но существуют и другие модели повторителей, в которых на нескольких портах имеется свой блок повторения.

рения. В таком случае каждый блок повторения нужно считать отдельным повторителем и учитывать его отдельно в правиле 4 хабов.

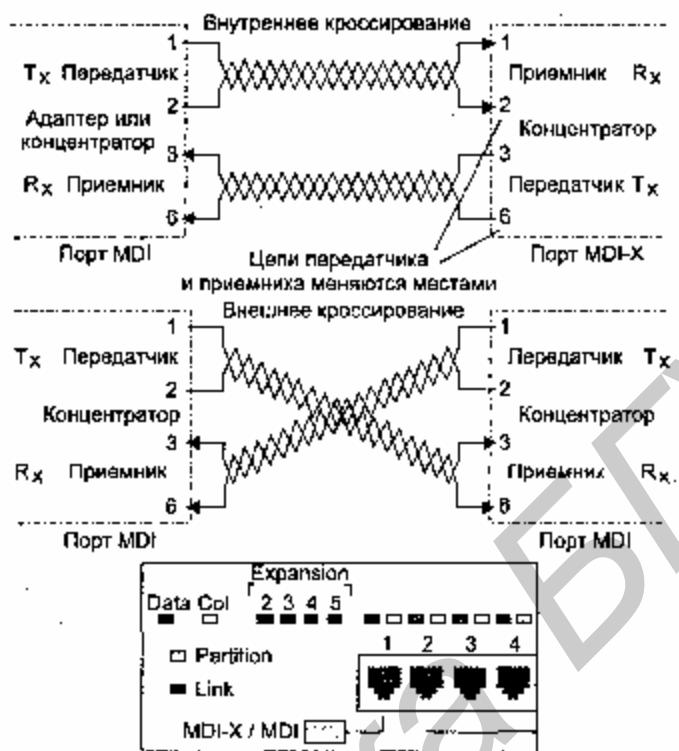


Рис. 9.6. Соединения типа «станция—концентратор» и «концентратор—концентратор» на витой паре

Некоторые отличия могут иметь место в моделях концентраторов, работающих на одномодовый волоконно-оптический кабель. Дальность сегмента кабеля, поддерживаемого концентратором FDDI, на таком кабеле может значительно отличаться в зависимости от мощности лазерного излучателя — от 10 до 40 км. Однако если существующие различия при выполнении основной функции концентраторов не столь велики, то их намного превосходит разброс в возможностях реализации концентраторами дополнительных функций.

Отключение портов

Очень полезной при эксплуатации сети является способность концентратора отключать некорректно работающие порты, изолируя тем самым остальную часть сети от возникших в узле проблем. Эту функцию называют автосегментацией (autopartitioning). Для концентратора FDDI эта функция для многих ошибочных ситуаций является основной, так как определена в протоколе. В то же время для концентратора Ethernet или Token Ring функция автосегментации для многих ситуаций является дополнительной, так как стандарт не описывает реакцию концентратора на эту ситуацию. Основной причиной отключения порта в стандартах Ethernet и Fast Ethernet является отсутствие ответа на последовательность импульсов link test, посыпаемых во все порты каждые 16 мс. В этом случае неисправный порт переводится в состояние «отключен», но импульсы

link test будут продолжать посыпаться в порт с тем, чтобы при восстановлении устройства работа с ним была продолжена автоматически.

Рассмотрим ситуации, в которых концентраторы Ethernet и Fast Ethernet выполняют отключение порта,

□ Ошибки на уровне кадра. Если интенсивность прохождения через порт кадров, имеющих ошибки, превышает заданный порог, то порт отключается, а затем, при отсутствии ошибок в течение заданного времени, включается снова. Такими ошибками могут быть: неверная контрольная сумма, неверная длина кадра (больше 1518 байт или меньше 64 байт), неоформленный заголовок кадра.

□ Множественные коллизии. Если концентратор фиксирует, что источником коллизии был один и тот же порт 60 раз подряд, то порт отключается. Через некоторое время порт снова будет включен.

□ Затянувшаяся передача (jabber). Как и сетевой адаптер, концентратор контролирует время прохождения одного кадра через порт. Если это время превышает время передачи кадра максимальной длины в три раза, то порт отключается.

Поддержка резервных связей

Так как использование резервных связей в концентраторах определено только в стандарте FDDI, то для остальных стандартов разработчики концентраторов поддерживают такую функцию с помощью своих частных решений. Например, концентраторы Ethernet/Fast Ethernet могут образовывать только иерархические связи без петель. Поэтому резервные связи всегда должны соединять отключенные порты, чтобы не нарушать логику работы сети. Обычно при конфигурировании концентратора администратор должен определить, какие порты являются основными, а какие по отношению к ним — резервными (рис. 9.7). Если по какой-либо причине порт отключается (срабатывает механизм автосегментации), концентратор делает активным его резервный порт.

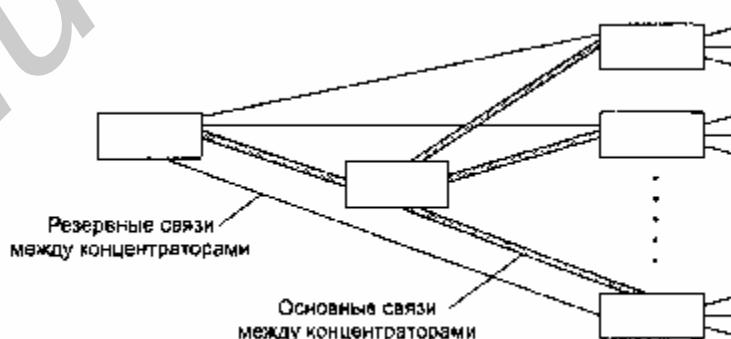


Рис. 9.7. Резервные связи между концентраторами Ethernet

В некоторых моделях концентраторов разрешается использовать механизм назначения резервных портов только для оптоволоконных портов, разработчики устройства исходили из того, что нужно резервировать только наиболее важные связи, которые обычно выполняются на оптическом кабеле. В других, же моделях резервным можно сделать любой порт.

Защита от несанкционированного доступа

Разделяемая среда предоставляет очень удобную возможность для несанкционированного прослушивания сети и получения доступа к передаваемым данным. Для этого достаточно подключить компьютер с программным анализатором протоколов к свободному разъему концентратора, записать на диск весь проходящий по сети трафик, а затем выделить из него нужную информацию. Разработчики концентраторов предоставляют некоторый способ защиты данных в разделяемых средах.

Наиболее простой способ — назначение разрешенных MAC-адресов портам концентратора. В стандартном концентраторе Ethernet порты MAC-адресов не имеют. Защита заключается в том, что администратор вручную связывает с каждым портом концентратора некоторый MAC-адрес. Этот MAC-адрес является адресом станции, которой разрешается подключаться к данному порту. Например, на рис. 9.8 первому порту концентратора назначен MAC-адрес 123 (условная запись). Компьютер с MAC-адресом 123 нормально работает с сетью через данный порт. Если злоумышленник отсоединяет этот компьютер и присоединяется вместо него свой, концентратор заметит, что при старте нового компьютера в сеть начали поступать кадры с адресом источника 789. Так как этот адрес является недопустимым для первого порта, то эти кадры фильтруются, порт отключается, а факт нарушения прав доступа может быть зафиксирован.

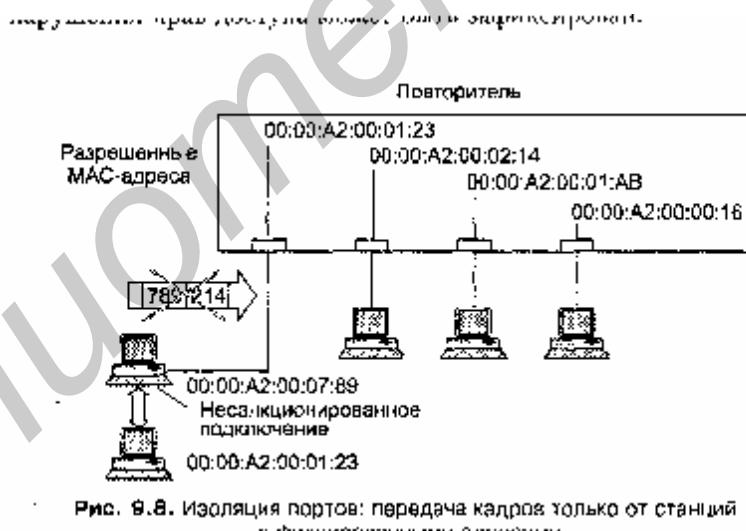


Рис. 9.8. Изоляция портов: передача кадров только от станций с фиксированными адресами

Заметим, что для реализации описанного метода защиты данных концентратор нужно предварительно сконфигурировать. Для этого концентратор должен иметь блок управления. Такие концентраторы обычно называют интеллектуальными. Блок управления представляет собой компактный вычислительный блок со встроенным программным обеспечением. Для взаимодействия администратора с блоком управления концентратор имеет консольный порт (чаще всего RS-232), к которому подключается терминал или персональный компьютер с программой эмуляции терминала. При присоединении терминала блок управ-

ления организует на его экране диалог, с помощью которого администратор вводит значения MAC-адресов. Блок управления может поддерживать и другие операции конфигурирования, например ручное отключение или включение портов и т. д. Для этого при подключении терминала блок управления выдает на экран некоторое меню, с помощью которого администратор выбирает нужное действие.

Другим способом защиты данных от несанкционированного доступа является их шифрация. Однако процесс истинной шифрации требует большой вычислительной мощности, и для повторителя, не буферизующего кадр, выполнить шифрацию «на лету» весьма сложно. Вместо этого в концентраторах применяется метод случайного искажения поля данных в пакетах, передаваемых портам с адресом, отличным от адреса назначения пакета. Этот метод сохраняет логику случайного доступа к среде, так как все станции видят занятость среды кадром информации, но только станция, которой послан этот кадр, может понять содержание поля данных кадра (рис. 9.9). Для реализации этого метода концентратор также нужно снабдить информацией о том, какие MAC-адреса имеют станции, подключенные к его портам. Обычно поле данных в кадрах, направляемых станциям, отличным от адресата, заполняется нулями.

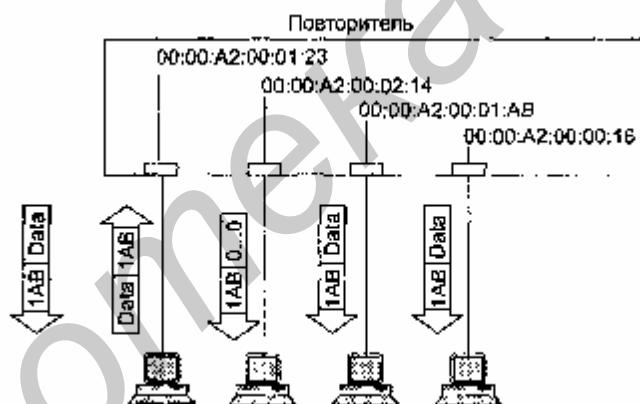


Рис. 9.9. Искажение поля данных в кадрах, не предназначенных для приема станциями

Многосегментные концентраторы

При рассмотрении некоторых моделей концентраторов возникает вопрос — зачем в них имеется такое большое количество портов, например 192 или 240? Имеет ли смысл разделять среду в 10 или 16 Мбит/с между таким большим количеством станций? Возможно, 10-15 лет назад ответ в некоторых случаях мог бы быть и положительным, например, для тех сетей, в которых компьютеры пользовались сетью только для отправки небольших почтовых сообщений или для переписывания небольшого текстового файла. Сегодня таких сетей осталось крайне мало, и далее 5 компьютеров могут полностью загрузить сегмент Ethernet или Token Ring, а в некоторых случаях — и сегмент Fast Ethernet. Для чего же тогда нужен концентратор с большим количеством портов, если ими практически нельзя воспользоваться из-за ограничений по пропускной способ-

ности, приходящейся на одну станцию? Ответ состоит в том, что в таких концентраторах имеется несколько несвязанных внутренних шин, которые предназначены для создания нескольких разделяемых сред. Так, концентратор, представленный на рис. 9.10, имеет три внутренние шины Ethernet. Если, например, в таком концентраторе 72 порта, то каждый из этих портов может быть связан с любой из трех внутренних шин. На рисунке первые два компьютера связаны с шиной Ethernet 3, а третий и четвертый компьютеры — с шиной Ethernet 1. Первые два компьютера образуют один разделяемый сегмент, а третий и четвертый — другой разделяемый сегмент.

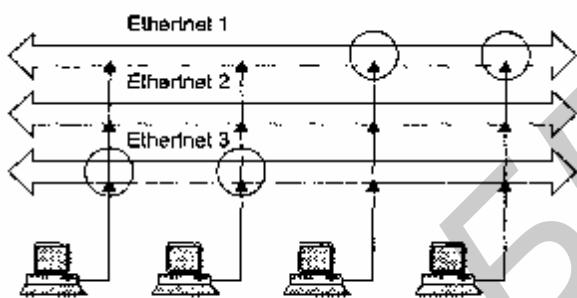


Рис. 9.10. Многосегментный концентратор

Между собой компьютеры, подключенные к разным сегментам, общаться через концентратор не могут, так как шины внутри концентратора никак не связаны. Многосегментные концентраторы нужны для создания разделяемых сегментов, состав которых может легко изменяться. Большинство многосегментных концентраторов позволяют выполнять операцию соединения порта с одной из внутренних шин чисто программным способом, например путем локального конфигурирования через консольный порт. В результате администратор сети может присоединять компьютеры пользователей к любым портам концентратора, а затем с помощью программы конфигурирования концентратора управлять составом каждого сегмента. Если завтра сегмент 1 окажется перегруженным, то его компьютеры можно распределить между оставшимися сегментами концентратора.

Возможность многосегментного концентратора программно изменять связи портов с внутренними шинами называется конфигурационной коммутацией (configuration switching).

ВНИМАНИЕ

Конфигурационная коммутация не имеет ничего общего с коммутацией кадров, которую выполняют мосты и коммутаторы.

Многосегментные концентраторы — это программируемая основа больших сетей. Для соединения сегментов между собой нужны устройства другого типа — мосты/коммутаторы или маршрутизаторы. Такое межсетевое устройство должно подключаться к нескольким портам многосегментного концентратора, подсоединенными к разным внутренним шинам, и выполнять передачу кадров или

пакетов между сегментами точно так же, как если бы они были образованы отдельными устройствами-концентраторами.

Для крупных сетей многосегментный концентратор играет роль интеллектуального кроссового шкафа, который выполняет новое соединение не за счет механического перемещения вилки кабеля в новый порт, а за счет программного изменения внутренней конфигурации устройства.

Управление концентратором по протоколу SNMP

Как видно из описания дополнительных функций, многие из них требуют конфигурирования концентратора. Это конфигурирование может производиться локально, через интерфейс RS-232C, который имеется у любого концентратора, имеющего блок управления. Кроме конфигурирования в большой сети очень полезна функция наблюдения за состоянием концентратора: работоспособен ли он, в каком состоянии находятся его порты.

При большом количестве концентраторов и других коммуникационных устройств в сети постоянное наблюдение за состоянием многочисленных портов и изменением их параметров становится очень обременительным занятием, если оно должно выполняться путем локального подключения терминала. Поэтому большинство концентраторов, поддерживающих интеллектуальные дополнительные функции, могут управляться централизованно по сети с помощью популярного протокола управления SNMP (Simple Network Management Protocol) стека TCP/IP. Упрощенная структура системы управления показана на рис. 9.11.



Рис. 9.11. Структура системы управления на основе протокола SNMP

Подробно системы управления и протокол SNMP рассматриваются в части VI. В блок управления концентратором встраивается так называемый SNMP-агент. Этот агент собирает информацию о состоянии контролируемого устройства и хранит ее в так называемой базе данных управляющей информации — Management Information Base, MIB. Эта база данных имеет стандартную структуру, что позволяет одному из компьютеров сети, играющему роль центральной станции управления, запрашивать у агента значения стандартных переменных базы MIB. В базе MIB хранятся не только данные о состоянии устройства, но и управляющая информация, воздействующая на это устройство. Например, в

МIB есть переменная, ответственная за управление состоянием порта и имеющая значения «включить» и «выключить». Если станция управления меняет значение данной переменной, то агент должен выполнить это указание и воздействовать на устройство соответствующим образом, например выключить порт или изменить связь порта с внутренними шинами концентратора.

Взаимодействие между станцией управления (по-другому - менеджером системы управления) и встроенными в коммуникационные устройства агентами происходит по протоколу SNMP. Концентратор, который управляется по протоколу SNMP, должен поддерживать основные протоколы стека TCP/IP и иметь IP-и MAC-адреса. Точнее, эти адреса относятся к агенту концентратора. Поэтому администратор, который хочет воспользоваться преимуществами централизованного управления концентраторами по сети, должен знать стеки протоколов TCP/IP и сконфигурировать IP-адреса их агентов.

Конструктивное исполнение концентраторов

На конструктивное устройство концентраторов большое влияние оказывает их область применения. Концентраторы рабочих групп чаще всего выпускаются как устройства с фиксированным количеством портов, корпоративные концентраторы — как модульные устройства на основе шасси, а концентраторы отделов могут иметь стековую конструкцию. Такое деление не является жестким, и в качестве корпоративного концентратора может использоваться, например, модульный концентратор.

Концентратор с фиксированным количеством портов — это наиболее простое конструктивное исполнение, когда устройство представляет собой отдельный корпус со всеми необходимыми элементами (портами, органами индикации и управления, блоком питания), и эти элементы заменять нельзя. Обычно все порты такого концентратора поддерживают одну среду передачи, общее количество портов изменяется от 4-8 до 24. Один порт может быть специально выделен для подключения концентратора к магистрали сети или же для объединения концентраторов (в качестве такого порта часто используется порт с интерфейсом AUI, в этом случае применение соответствующего трансивера позволяет подключить концентратор практически к любой физической среде передачи данных).

Модульный концентратор выполняется в виде отдельных модулей с фиксированным количеством портов, устанавливаемых на общее шасси. Шасси имеет внутреннюю шину для объединения отдельных модулей в единый повторитель. Часто такие концентраторы являются многосегментными, тогда в пределах одного модульного концентратора работает несколько несвязанных между собой повторителей. Для модульного концентратора могут существовать различные типы модулей, отличающиеся количеством портов и типом поддерживаемой физической среды. Часто агент протокола SNMP выполняется в виде отдельного модуля, при установке которого концентратор превращается в интеллектуальное устройство. Модульные концентраторы позволяют более точно подобрать необходимую для конкретного применения конфигурацию концентратора,

а также гибко и с минимальными затратами реагировать на изменения конфигурации сети.

Ввиду ответственной работы, которую выполняют корпоративные модульные концентраторы, они снабжаются модулем управления, системой терморегулирования, избыточными источниками питания и возможностью замены модулей «на ходу».

Недостатком концентратора на основе шасси является высокая начальная стоимость такого устройства для случая, когда предприятию на первом этапе создания сети нужно установить всего 1-2 модуля. Высокая стоимость шасси вызвана тем, что оно поставляется вместе со всеми общими устройствами, такими как избыточные источники питания и т. п. Поэтому для сетей средних размеров большую популярность завоевали стековые концентраторы.

Стековый концентратор, как и концентратор с фиксированным числом портов, выполнен в виде отдельного корпуса без возможности замены отдельных его модулей. Типичный вид нескольких стековых концентраторов Ethernet показан на рис. 9.12. Стековыми эти концентраторы называются не потому, что устанавливаются один на другой. Такая чисто конструктивная деталь вряд ли удостоилась бы особого внимания, так как установка нескольких устройств одинаковых габаритов в общую стойку практикуется очень давно. Стековые концентраторы имеют специальные порты и кабели для объединения нескольких таких корпусов в единый повторитель (рис. 9.13), который имеет общий блок повторения, обеспечивает общую ресинхронизацию сигналов для всех своих портов и поэтому с точки зрения правила 4 хабов считается одним повторителем.



Рис. 9.12. Стековые концентраторы Ethernet

Если стековые концентраторы имеют несколько внутренних шин, то при соединении в стек эти шины объединяются и становятся общими для всех устройств стека. Число объединяемых в стек корпусов может быть достаточно большим (обычно до 8, но бывает и больше). Стековые концентраторы могут поддерживать различные физические среды передачи, что делает их почти такими же гибкими, как и модульные концентраторы, но при этом стоимость этих устройств в расчете на один порт получается обычно ниже, так как сначала предприятие может купить одно устройство без избыточного шасси, а потом нарастить стек еще несколькими аналогичными устройствами.

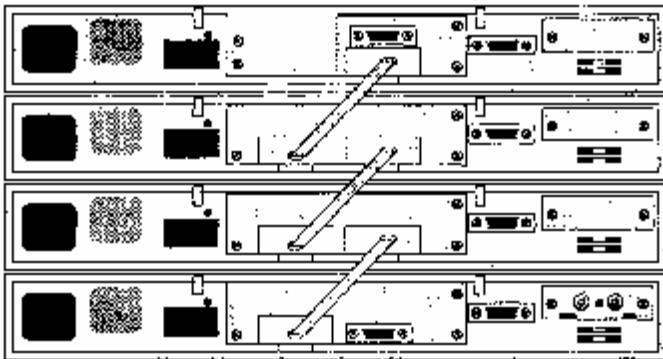


Рис. 9.13. Объединение стековых концентраторов в единое устройство с помощью специальных разъемов на задней панели

Стековые концентраторы, выпускаемые одним производителем, выполняются в едином конструктивном стандарте, что позволяет легко устанавливать их друг на друга, образуя единое настольное устройство, или помещать их в общую стойку. Экономия при организации стека происходит еще и за счет единого для всех устройств стека модуля SNMP-управления (который вставляется в один из корпусов стека как дополнительный модуль), а также общего избыточного источника питания.

Модульно-стековые концентраторы представляют собой модульные концентраторы, объединенные специальными связями в стек. Как правило, корпуса таких концентраторов рассчитаны на небольшое количество модулей (1-3). Эти концентраторы сочетают достоинства концентраторов обоих типов.

Приведенная выше классификация конструктивного исполнения справедлива не только для концентраторов, но и для коммуникационных устройств всех типов — мостов и коммутаторов локальных сетей, коммутаторов глобальных сетей и маршрутизаторов. Нужно только отметить, что не для всех типов устройств стековая конструкция подразумевает столь тесное взаимодействие элементов стека, как у концентраторов. Часто устройства стека объединяют только общие блоки питания и управления, а основные функции устройства стека могут выполнять достаточно автономно.

Выводы

- От производительности сетевых адаптеров зависит производительность любой сложной сети, так как данные всегда проходят не только через коммутаторы и маршрутизаторы сети, но и через адAPTERЫ компьютеров, а результирующая производительность последовательно соединенных устройств определяется производительностью самого медленного устройства.
- Сетевые адAPTERЫ характеризуются типом поддерживаемого протокола, производительностью, шиной компьютера, к которой они могут присоединяться, типом приемопередатчика, а также наличием собственного процессора, разгружающего центральный процессор компьютера от рутинной работы.

- Сетевые адаптеры для серверов обычно имеют собственный процессор, а клиентские сетевые адаптеры — нет.
- Современные адаптеры умеют адаптироваться к временным параметрам шины и оперативной памяти компьютера для повышения производительности обмена системы «сеть—компьютер».
- Концентраторы, кроме основной функции протокола (побитного повторения кадра на всех портах или на следующем порту), всегда выполняют ряд полезных дополнительных функций, определяемых производителем концентратора.
- Автосегментация — одна из важнейших дополнительных функций, с помощью которой концентратор отключает порт при обнаружении разнообразных проблем с кабелем и конечным узлом, подключенным к данному порту.
- В число дополнительных функций входят функции защиты сети от несанкционированного доступа, запрещающие подключение к концентратору компьютеров с неизвестными MAC-адресами, а также заполняющие нулями поля данных кадров, поступающих не к станции назначения.
- Стековые концентраторы сочетают преимущества модульных концентраторов и концентраторов с фиксированным количеством портов.
- Многосегментные концентраторы позволяют делить сеть на сегменты программным способом, без физической перекоммутации устройств.
- Сложные концентраторы, выполняющие дополнительные функции, обычно могут управляться централизованно по сети по протоколу SNMP.

Логическая структуризация сети с помощью мостов и коммутаторов

Под логической структуризацией сети понимается разбиение общей разделяемой среды на логические сегменты, которые представляют собой самостоятельные разделяемые среды с меньшим количеством узлов. Сеть, разделенная на логические сегменты, обладает более высокой производительностью и надежностью. Взаимодействие между логическими сегментами организуется с помощью мостов и коммутаторов

Недостатки сети на одной разделяемой среде

При построении небольших сетей, состоящих из 10-30 узлов, использование стандартных технологий на разделяемых средах передачи данных приводит к экономичным и эффективным решениям. Во всяком случае, это утверждение справедливо для очень большого числа сегодняшних сетей, даже тех, в которых передаются большие объемы мультимедийной информации, — появление высокоскоростных технологий со скоростями обмена 100 и 1000 Мбит/с решает проблему качества транспортного обслуживания таких сетей.

Эффективность разделяемой среды для небольшой сети проявляется в первую очередь в следующих свойствах:

- простой топологии сети, допускающей легкое наращивание числа узлов (в

небольших пределах);

- отсутствии потерь кадров из-за переполнения буферов коммуникационных устройств, так как новый кадр не передается в сеть, пока не принят предыдущий — сама логика разделения среды регулирует поток кадров и приостанавливает станции, слишком часто генерирующие кадры, заставляя их ждать доступа;
- простоте протоколов, обеспечившей низкую стоимость сетевых адаптеров, повторителей и концентраторов.

Однако справедливым является и другое утверждение — крупные сети, насчитывающие сотни и тысячи узлов, не могут быть построены на основе одной разделяемой среды даже при такой скоростной технологии, как Gigabit Ethernet. И не только потому, что практически все технологии ограничивают количество узлов в разделяемой среде: вес виды семейства Ethernet — 1024 узлами, Token Ring — 260 узлами, а FDDI — 500 узлами. Даже сеть средних размеров, состоящая из 50-100 компьютеров и укладывающаяся в разрешенный максимум количества узлов, чаще всего будет плохо работать на одной разделяемой среде.

Основные недостатки сети на одной разделяемой среде начинают проявляться при превышении некоторого порога количества узлов, подключенных к разделяемой среде, и состоят в следующем. Даже та доля пропускной способности разделяемого сегмента, которая должна в среднем доставаться одному узлу (то есть, например, $10/N$ Мбит/с для сегмента Ethernet с N компьютерами), очень часто узлу не достается. Причина заключается в случайному характере метода доступа к среде, используемом во всех технологиях локальных сетей. Наиболее тяжелые условия для узлов сети создает метод доступа CSMA/CD технологии Ethernet, но и в других технологиях, таких как Token Ring или FDDI, где метод доступа носит менее случайный характер и даже часто называется детерминированным, случайный фактор доступа к среде все равно присутствует и оказывает свое негативное влияние на пропускную способность, достающуюся отдельному узлу.

На рис. 9.14 показана зависимость задержек доступа к среде передачи данных в сетях Ethernet, Token Ring и FDDI от коэффициента использования сети r , который также часто называют коэффициентом нагрузки сети. Напомним, что коэффициент использования сети равен отношению трафика, который должна передать сеть, к ее максимальной пропускной способности. Для сети Ethernet максимальная пропускная способность равна 10 Мбит/с, а трафик, который она должна передать, равен сумме интенсивностей трафика, генерируемого каждым узлом сети. Коэффициент использования обычно измеряют в относительных единицах или процентах.

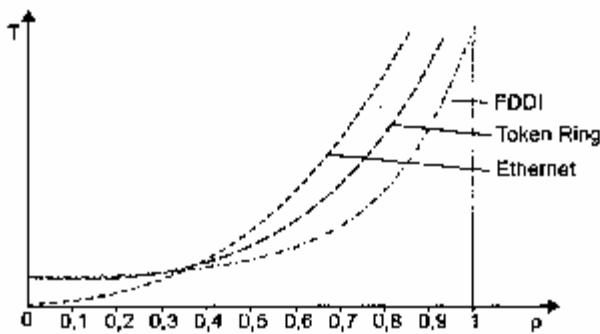


Рис. 9.14. Задержки доступа к среде передачи данных для технологий Ethernet, Token Ring и FDDI

Как видно из рисунка, всем технологиям присущ экспоненциальный рост величины задержек доступа при увеличении коэффициента использования сети, отличается только порог, при котором наступает резкий перелом в поведении сети, когда почти прямолинейная зависимость переходит в крутую экспоненту. Для всего семейства технологий Ethernet — это 40-50%, для технологии Token Ring - 50 %, а технологии FDDI - 70 %.

Количество узлов, при которых коэффициент использования сети начинает приближаться к опасной границе, зависит от типа функционирующих в узлах приложений. Если раньше для сетей Ethernet считалось, что 30 узлов — это вполне приемлемое число для одного разделяемого сегмента, то сегодня для мультимедийных приложений, перекачивающих большие файлы данных, эту цифру нужно уточнять с помощью натурных или имитационных экспериментов.

Влияние задержек и коллизий на полезную пропускную способность сети Ethernet хорошо отражает график, представленный на рис. 9.15.



Рис. 9.15. Зависимость полезной пропускной способности сети Ethernet от коэффициента использования

При загрузке сети до 50 % технология Ethernet на разделяемом сегменте хорошо справляется с передачей трафика, генерируемого конечными узлами. Однако при повышении интенсивности генерируемого узлами трафика сеть все больше времени начинает проводить неэффективно, повторно передавая кадры,

которые вызвали коллизию. При возрастании интенсивности генерируемого трафика до такой величины, когда коэффициент использования сети приближается к 1, вероятность столкновения кадров настолько увеличивается, что практически любой кадр, который какая-либо станция пытается передать, сталкивается с другими кадрами, вызывая коллизию. Сеть перестает передавать полезную пользовательскую информацию и работает «на себя», обрабатывая коллизии.

Этот эффект хорошо известен на практике и исследован путем имитационного моделирования, поэтому сегменты Ethernet не рекомендуется загружать так, чтобы среднее значение коэффициента использования превосходило 30 %. Именно поэтому во многих системах управления сетями пороговая граница для индикатора коэффициента загрузки сети Ethernet по умолчанию устанавливается на величину 30 %.

Технология Ethernet наиболее чувствительна к перегрузкам разделяемого сегмента, но и другие технологии также весьма страдают от этого эффекта, поэтому ограничения, связанные с возникающими коллизиями и большим временем ожидания доступа при значительной загрузке разделяемого сегмента, чаще всего оказываются более серьезными, чем ограничение на максимальное количество узлов, определенное в стандарте из соображений устойчивой передачи электрических сигналов в кабелях.

В результате даже сеть средних размеров трудно построить на одном разделяемом сегменте так, чтобы она работала эффективно при изменении интенсивности генерируемого станциями трафика. Кроме того, при использовании разделяемой среды проектировщик сети сталкивается с жесткими ограничениями максимальной длины сети, которые для всех технологий лежат в пределах нескольких километров, и только технология FDDI позволяет строить локальные сети, длина которых измеряется десятками километров.

Преимущества логической структуризации сети

Ограничения, возникающие из-за использования одной разделяемой среды, можно преодолеть, разделив сеть на несколько разделяемых сред и соединив отдельные сегменты сети такими устройствами, как мосты, коммутаторы или маршрутизаторы (рис. 9.16).

Перечисленные устройства передают кадры с одного своего порта на другой, анализируя адрес назначения, помещенный в этих кадрах. (В отличие от концентраторов, которые повторяют кадры на всех своих портах, передавая их во все подсоединеные к ним сегменты, независимо от того, в каком из них находится станция назначения.) Мосты и коммутаторы выполняют операцию передачи кадров на основе плоских адресов канального уровня, то есть MAC-адресов, а маршрутизаторы — на основе номера сети. При этом единая разделяемая среда, созданная концентраторами (или в предельном случае — одним сегментом кабеля), делится на несколько частей, каждая из которых присоединена к порту моста, коммутатора или маршрутизатора.

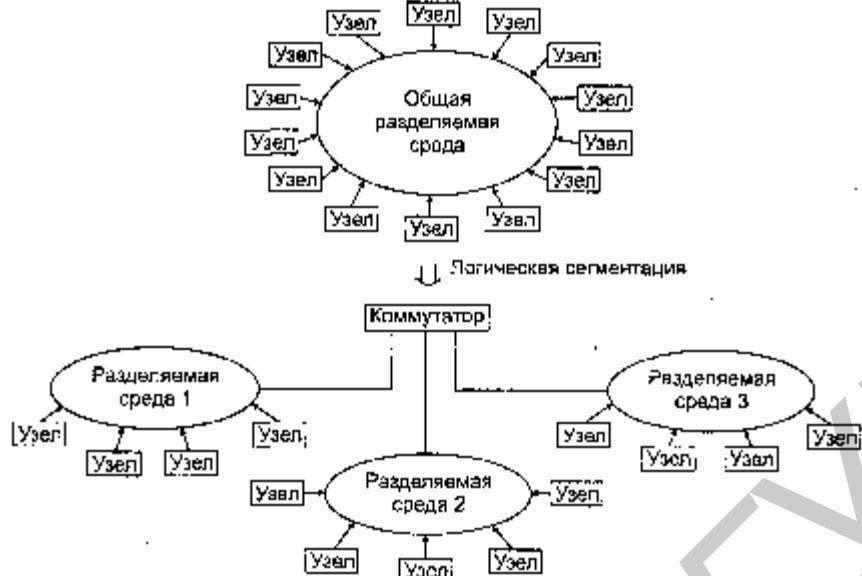


Рис. 9.16. Логическая структуризация сети

Говорят, что при этом сеть делится на логические сегменты или сеть подвергается логической структуризации. Логический сегмент представляет собой единую разделяемую среду. Деление сети на логические сегменты приводит к тому, что нагрузка, приходящаяся на каждый из вновь образованных сегментов, почти всегда оказывается меньше, чем нагрузка, которую испытывала исходная сеть. Следовательно, уменьшаются вредные эффекты от разделения среды: снижается время ожидания доступа, а в сетях Ethernet — и интенсивность коллизий.

Для иллюстрации этого эффекта рассмотрим рис. 9.17. На нем показаны два сегмента, соединенные мостом. Внутри сегментов имеются повторители. До деления сети на сегменты весь трафик, генерируемый узлами сети, был общим (представим, что место межсетевого устройства также занимал повторитель) и учитывался при определении коэффициента использования сети. Если обозначить среднюю интенсивность трафика, идущего от узла i к узлу j через C_{ij} , то суммарный трафик, который должна была передавать сеть до деления на сегменты, равен $C_{\Sigma} = \sum C_{ij}$ (считаем, что суммирование проводится по всем узлам).

После разделения сети на сегменты нагрузка каждого сегмента изменилась. При ее вычислении теперь нужно учитывать только внутрисегментный трафик, то есть трафик кадров, которые циркулируют между узлами одного сегмента, а также межсегментный трафик, который либо направляется от узла данного сегмента узлу другого сегмента, либо приходит от узла другого сегмента о узел данного сегмента. Внутренний трафик другого сегмента нагрузку на данный сегмент больше не создает.

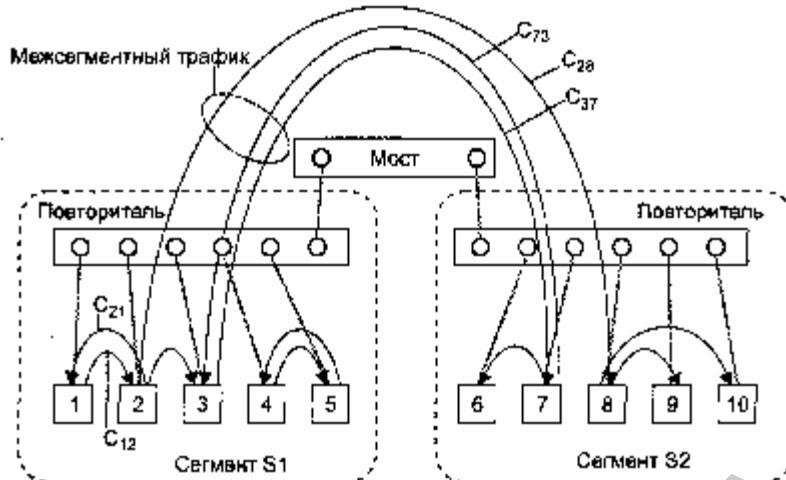


Рис. 9.17. Изменение нагрузки при делении сети на сегменты

Поэтому нагрузка, например, сегмента St стала равна $C_{S1} + C_{S1-S2}$, где C_{S1} — внутренний трафик сегмента S1, а C_{S1-S2} — межсегментный трафик. Чтобы показать, что нагрузка сегмента S1 уменьшилась, заметим, что общую нагрузку сети до разделения на сегменты можно записать в такой форме: $C_{\Sigma} = C_{S1} + C_{S2} + C_{S3}$, а значит, нагрузка сегмента S1 после разделения стала равной $C_{\Sigma} - C_{S2}$, то есть уменьшилась на величину внутреннего трафика сегмента S2. А раз нагрузка на сегмент уменьшилась, то в соответствии с графиками, приведенными на рис. 9.14 и 9.15, задержки в сегментах также уменьшились, а полезная пропускная способность сегмента в целом и полезная пропускная способность, приходящаяся на один узел, увеличились.

Выше было сказано, что деление сети на логические сегменты почти всегда уменьшает нагрузку в новых сегментах. Слово «почти» учитывает очень редкий случай, когда сеть разбита на сегменты так, что внутренний трафик каждого сегмента равен нулю, то есть весь трафик является межсегментным. Для примера, представленного на рис. 9.17, это означало бы, что все компьютеры сегмента S1 обмениваются данными только с компьютерами сегмента S2, и наоборот.

Такой случай является, естественно, экзотическим. На практике на предприятии всегда можно выделить группу компьютеров, которые принадлежат сотрудникам, выполняющим общую задачу. Это могут быть сотрудники одной рабочей группы, отдела, другого структурного подразделения предприятия. В большинстве случаев им нужен доступ к ресурсам сети их отдела и только изредка — доступ к удаленным ресурсам. И хотя уже упомянутое эмпирическое правило, говорящее о том, что можно разделить сеть на сегменты так, что 80 % трафика составят обращения к локальным ресурсам и только 20 % — к удаленным, сегодня трансформируется в правило 50 на 50 % и даже 20 на 80 %, все равно внутрисегментный трафик существует. Если его нет, значит, сеть разбита на логические подсети неверно.

Большинство крупных сетей разрабатывается на основе структуры с общей магистралью, к которой через мосты и маршрутизаторы присоединяются подсети.

Эти подсети обслуживают различные отделы. Подсети могут делиться и далее на сегменты, предназначенные для обслуживания рабочих групп.

В общем случае деление сети на логические сегменты повышает производительность (за счет разгрузки сегментов) и гибкость сети, увеличивая степень защиты данных, а также облегчает управление сетью.

Сегментация увеличивает гибкость сети. При построении сети как совокупности подсетей каждая подсеть может быть адаптирована к специфическим потребностям рабочей группы или отдела. Например, в одной подсети может использоваться технология Ethernet и ОС NetWare, а в другой Token Ring и OS-400, в соответствии с традициями того или иного отдела или потребностями имеющихся приложений. Вместе с тем у пользователей обеих подсетей есть возможность обмениваться данными через межсетевые устройства, такие как мосты, коммутаторы, маршрутизаторы. Процесс разбиения сети на логические сегменты можно рассматривать и в обратном направлении, как процесс создания большой сети из модулей — уже имеющихся подсетей.

Подсети повышают безопасность данных. При подключении пользователей к различным физическим сегментам сети можно запретить доступ определенных пользователей к ресурсам других сегментов. Устанавливая различные логические фильтры на мостах, коммутаторах и маршрутизаторах, можно контролировать доступ к ресурсам, чего не позволяют сделать повторители.

Подсети упрощают управление сетью. Побочным эффектом снижения трафика и повышения безопасности данных является упрощение управления сетью. Проблемы очень часто локализуются внутри сегмента. Как и в случае структурированной кабельной системы, проблемы одной подсети не оказывают влияния на другие подсети. Подсети образуют логические домены управления сетью.

Сети должны проектироваться на двух уровнях: физическом и логическом. Логическое проектирование определяет места расположения ресурсов, приложений и способы группировки этих ресурсов в логические сегменты.

Структуризация с помощью мостов и коммутаторов

В данной главе рассматриваются устройства логической структуризации сетей, работающие на канальном уровне стека протоколов, а именно — мосты и коммутаторы. Структуризация сети возможна также на основе маршрутизаторов, которые для выполнения этой задачи привлекают протоколы сетевого уровня. Каждый способ структуризации — с помощью канального протокола и сетевого протокола — имеет свои преимущества и недостатки. В современных сетях часто используют комбинированный способ логической структуризации — небольшие сегменты объединяются устройствами канального уровня в более крупные подсети, которые, в свою очередь, соединяются маршрутизаторами.

Итак, сеть можно разделить на логические сегменты с помощью устройств двух типов — мостов (bridge) и/или коммутаторов (switch). Сразу после появления коммутаторов в начале 90-х годов сложилось мнение, что мост и коммутатор — это принципиально разные устройства. И хотя постепенно представление о

коммутаторах изменилось, это мнение можно услышать и сегодня. Тем не менее мост и коммутатор — это функциональные близнецы. Оба эти устройства продвигают кадры на основании одних и тех же алгоритмов. Мосты и коммутаторы используют два типа алгоритмов: алгоритм прозрачного моста (*transparent bridge*), описанного в стандарте IEEE 802.1D, либо алгоритм моста с маршрутизацией от источника (*source routing bridge*) компании IBM для сетей Token Ring. Эти стандарты были разработаны задолго до появления первого коммутатора, поэтому для них характерен термин «мост». Когда же на свет появилась первая промышленная модель коммутатора для технологии Ethernet, то она выполняла тот же алгоритм продвижения кадров IEEE 802.1D, который был с десяток лет отработан мостами локальных и глобальных сетей. Точно так же поступают и все современные коммутаторы. Коммутаторы, которые продвигают кадры протокола Token Ring, работают по алгоритму *Source Routing*, характерному для мостов IBM.

Основное отличие коммутатора от моста заключается в том, что мост обрабатывает кадры последовательно, а коммутатор — параллельно. Это обстоятельство связано с тем, что мосты появились в те времена, когда сеть делили на небольшое количество сегментов, а межсегментный трафик был небольшим (он подчинялся правилу 80 на 20 %). Сеть чаще всего делили на два сегмента, поэтому и термин был выбран соответствующий — «мост». Для обработки потока данных со средней интенсивностью 1 Мбит/с мосту вполне хватало производительности одного процессорного блока.

При изменении ситуации в конце 80-х — начале 90-х годов — появлении быстрых протоколов, производительных персональных компьютеров, мультимедийной информации, разделении сети на большое количество сегментов — классические мосты перестали справляться с работой. Обслуживание потоков кадров между теперь уже несколькими портами с помощью одного процессорного блока требовало значительного повышения быстродействия процессора, а это довольно дорогостоящее решение.

Более эффективным оказалось решение, которое и «породило» коммутаторы: для обслуживания потока, поступающего на каждый порт, в устройство ставился отдельный специализированный процессор, который реализовывал алгоритм моста. По сути, коммутатор — это мультипроцессорный мост, способный параллельно продвигать кадры сразу между всеми парами своих портов. Но если при добавлении процессорных блоков компьютер не перестали называть компьютером, а добавили только прилагательное «мультипроцессорный», то с мультипроцессорными мостами произошла метаморфоза — они превратились в коммутаторы. Этому способствовал способ связи между отдельными процессорами коммутатора — они связывались коммутационной матрицей, похожей на матрицы мультипроцессорных компьютеров, связывающие процессоры с блоками памяти.

Постепенно коммутаторы вытеснили из локальных сетей классические однопроцессорные мосты. Основная причина этого — очень высокая производительность, с которой коммутаторы передают кадры между сегментами сети. Если мосты могли даже замедлять работу сети, когда их производительность ока-

зывалась меньше интенсивности межсегментного потока кадров, то коммутаторы всегда выпускаются с процессорами портов, которые могут передавать кадры с той максимальной скоростью, на которую рассчитан протокол. Добавление к этому возможности параллельной передачи кадров между портами сделало производительность коммутаторов на несколько порядков выше, чем мостов — коммутаторы могут передавать до нескольких миллионов кадров в секунду, в то время как мосты обычно обрабатывали 3-5 тысяч кадров в секунду. Это и предопределило судьбу мостов и коммутаторов.

Процесс вытеснения мостов начал протекать достаточно быстро с 1994 года, и сегодня локальные мосты практически не производятся сетевой индустрией. За время своего существования уже без конкурентов-мостов коммутаторы вобрали в себя многие дополнительные функции, которые появлялись в результате естественного развития сетевых технологий. К этим функциям относятся, например, поддержка виртуальных сетей (VLAN), приоритезация трафика, использование магистрального порта по умолчанию и т. п.

Сегодня мосты по-прежнему работают в сетях, но только на достаточно медленных глобальных связях между двумя удаленными локальными сетями. Такие мосты называются удаленными (remote bridge), и алгоритм их работы ничем не отличается от стандарта 802.1D или Source Routing.

Прозрачные мосты помимо передачи кадров в рамках одной технологии умеют транслировать протоколы локальных сетей, например Ethernet в Token Ring, FDDI в Ethernet и т. п. Это свойство прозрачных мостов описано в стандарте IEEE802.1H.

В дальнейшем будем называть устройство, которое продвигает кадры по алгоритму моста и работает в локальной сети, современным термином «коммутатор». При описании же самих алгоритмов 802.1H и Source Routing в следующем разделе будем по традиции называть устройство мостом, как собственно оно в этих стандартах и называется.

Алгоритм работы прозрачного моста

Прозрачные мосты незаметны для сетевых адаптеров конечных узлов, так как они самостоятельно строят специальную адресную таблицу, на основании которой можно решить, нужно передавать пришедший кадр в какой-либо другой сегмент или нет. Сетевые адаптеры при использовании прозрачных мостов работают точно так же, как и в случае их отсутствия, то есть не предпринимают никаких дополнительных действий, чтобы кадр прошел через мост. Алгоритм прозрачного моста не зависит от технологии локальной сети, в которой устанавливается мост, поэтому прозрачные мосты Ethernet работают точно так же, как прозрачные мосты FDDI.

Прозрачный мост строит свою адресную таблицу на основании пассивного наблюдения за трафиком, циркулирующим в подключенных к его портам сегментах. При этом мост учитывает адреса источников кадров данных, поступающих на порты моста. По адресу источника кадра мост делает вывод о принадлежно-

сти этого узла тому или иному сегменту сети.

Рассмотрим процесс автоматического создания адресной таблицы моста и ее использования на примере простой сети, представленной на рис. 9.18.

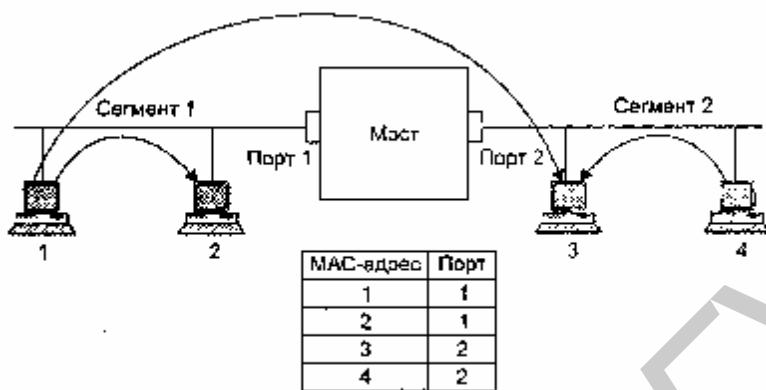


Рис. 9.18. Принцип работы прозрачного моста

Мост соединяет два логических сегмента. Сегмент 1 составляют компьютеры, подключенные с помощью одного отрезка коаксиального кабеля к порту 1 моста, а сегмент 2 — компьютеры, подключенные с помощью другого отрезка коаксиального кабеля к порту 2 моста.

Каждый порт моста работает как конечный узел своего сегмента за одним исключением — порт моста не имеет собственного MAC-адреса. Порт моста работает в так называемом неразборчивом (promiscuous) режиме захвата пакетов, когда все поступающие на порт пакеты запоминаются в буферной памяти. С помощью такого режима мост следит за всем трафиком, передаваемым в присоединенных к нему сегментах, и использует проходящие через него пакеты для изучения состава сети. Так как в буфер записываются все пакеты, то адрес порта мосту не нужен.

В исходном состоянии мост ничего не знает о том, компьютеры с какими MAC-адресами подключены к каждому из его портов. Поэтому в этом случае мост просто передает любой захваченный и буферизованный кадр на все свои порты за исключением того, от которого этот кадр получен. В нашем примере у моста только два порта, поэтому он передаст кадры с порта 1 на порт 2, и наоборот. Отличие работы моста в этом режиме от повторителя заключается в том, что он передает кадр не побитые, а с буферизацией. Буферизация разрывает логику работы всех сегментов как единой разделяемой среды. Когда мост собирается передать кадр с сегмента на сегмент, например с сегмента 1 на сегмент 2, он заново пытается получить доступ к сегменту 2 как конечный узел по правилам алгоритма доступа, в данном примере — по правилам алгоритма CSMA/CD.

Одновременно с передачей кадра на все порты мост изучает адрес источника кадра и делает новую запись о его принадлежности в своей адресной таблице, которую также называют таблицей фильтрации или таблицей маршрутизации. Например, получив на свой порт 1 кадр от компьютера 1, мост делает первую запись в своей адресной таблице: MAC-адрес 1 — порт 1. Если все четыре ком-

пьютера данной сети проявляют активность и посылают друг другу кадры, то скоро мост построит полную адресную таблицу сети, состоящую из 4 записей — по одной записи на узел,

После того как мост прошел этап обучения, он может работать более рационально. При получении кадра, направленного, например, от компьютера 1 компьютеру 3, он просматривает адресную таблицу на предмет совпадения ее адресов с адресом назначения 3. Поскольку такая запись есть, то мост выполняет второй этап анализа таблицы — проверяет, находятся ли компьютеры с адресами источника (в нашем случае — это адрес 1) и адресом назначения (адрес 3) в одном сегменте. Так как в нашем примере они находятся в разных сегментах, то мост выполняет операцию продвижения (forwarding) кадра — передает кадр на другой порт, предварительно получив доступ к другому сегменту.

Если бы оказалось, что компьютеры принадлежат одному сегменту, то кадр просто был бы удален из буфера и работа с ним на этом закончилась. Такая операция называется фильтрацией (filtering).

Если же адрес назначения неизвестен, то мост передает кадр на все свои порты, кроме порта — источника кадра, как и на начальной стадии процесса обучения. На самом деле мы несколько упростили алгоритм работы моста. Процесс его обучения никогда не заканчивается. Мост постоянно следит за адресами источника буферизуемых кадров, чтобы быть в состоянии автоматически приспособливаться к изменениям, происходящим в сети, — перемещениям компьютеров из одного сегмента сети в другой, появлению новых компьютеров. С другой стороны, мост не ждет, когда адресная таблица заполнится полностью (да это и невозможно, поскольку заранее не известно, сколько компьютеров и адресов будут находиться в сегментах моста). Как только в таблице появляется первый адрес, мост пытается его использовать, проверяя совпадение с ним адресов назначения всех поступающих пакетов.

Входы адресной таблицы могут быть динамическими, создаваемыми в процессе самообучения моста, и статическими, создаваемыми вручную администратором сети. Динамические входы имеют срок жизни — при создании или обновлении записи в адресной таблице с ней связывается отметка времени. По истечении определенного тайм-аута запись помечается как недействительная, если за это время мост не принял ни одного кадра с данным адресом в поле адреса источника. Это дает возможность автоматически реагировать на перемещения компьютера из сегмента в сегмент — при его отключении от старого сегмента запись о его принадлежности к нему со временем вычеркивается из адресной таблицы. После включения этого компьютера в работу в другом сегменте его кадры начнут попадать в буфер моста через другой порт, и в адресной таблице появится новая запись, соответствующая текущему состоянию сети.

Статические записи не имеют срока жизни, что дает администратору возможность подправлять работу моста, если это необходимо.

Кадры с широковещательными MAC-адресами передаются мостом на все его порты, как и кадры с неизвестным адресом назначения. Такой режим распространения кадров называется затоплением сети (flood). Наличие мостов в сети не препятствует распространению широковещательных кадров по всем сегмен-

там сети, сохраняя ее прозрачность. Однако это является достоинством только в том случае, когда широковещательный адрес выработан корректно работающим узлом. Однако часто случается так, что в результате каких-либо программных или аппаратных сбоев протокол верхнего уровня или сам сетевой адаптер начинают работать некорректно и постоянно с высокой интенсивностью генерировать кадры с широковещательным адресом в течение длительного промежутка времени. Мост в этом случае передаст эти кадры во все сегменты, затапливая сеть ошибочным трафиком. Такая ситуация называется широковещательным штормом (broadcast storm).

К сожалению, мосты не защищают сети от широковещательного шторма, во всяком случае, по умолчанию, как это делают маршрутизаторы. Максимум, что может сделать администратор с помощью моста для борьбы с широковещательным штормом — установить для каждого узла предельно допустимую интенсивность генерации кадров с широковещательным адресом. Но при этом нужно точно знать, какая интенсивность является нормальной, а какая — ошибочной. При смене протоколов ситуация в сети может измениться, и то, что вчера считалось ошибочным, сегодня может оказаться нормой. Таким образом, мосты располагают весьма грубыми средствами борьбы с широковещательным штормом.

На рис. 9.19 показана типичная структура моста. Функции доступа к среде при приеме и передаче кадров выполняют микросхемы MAC, которые идентичны микросхемам сетевого адаптера.

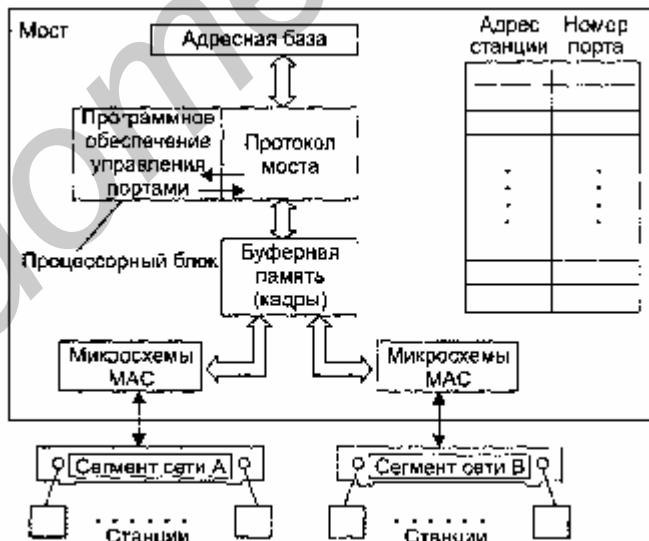


Рис. 9.19. Структура моста

На рис. 9.20 показана копия экрана терминала с адресной таблицей модуля локального моста концентратора System 3000 компании SynOptics (сам концентратор уже не выпускается, но в свое время он сыграл роль пионера в становлении многосегментных концентраторов Ethernet на витой паре, причем концентратор имел модуль моста, который мог соединять внутренние сегменты без

привлечения внешнего моста). Терминал подключен к консольному порту, и информация на его экране выведена модулем управления моста.

Forwarding Table						Page 1 of 1
Address	Dispн	Address	Dispн	Address	Dispн	
00608CB17E58	LAN B	0000810298D6	LAN A	02070188ACA	LAN A	
00008101C4DF	LAN B	* 000081016A52	LAN A	* 010081000100	Flood	
* 010081000101	Discard	* 0180C2000000	Discard	* 000081FFD166	Flood	

Статус адреса:
срок жизни записи истек

Exit Next Page Prev Page Edit Table Search Item Go Page
+ Unlearned * Static Total Entries = 9 Static Entries = 4
Use cursor keys to choose option. Press <RETURN> to select.
Press <CTRL> <P> to return to Main Menu

Рис. 9.20. Адресная таблица моста System 3000 Local Bridge

Из присутствующей на экране адресной таблицы (Forwarding Table) видно, что сеть состоит из двух сегментов — LAN A и LAN B. В сегменте LAN A имеются, по крайней мере, три станции, а в сегменте LAN B — две станции. Четыре адреса, помеченные звездочками, являются статическими, то есть назначенными администратором вручную. Адрес, помеченный знаком + (плюс), является динамическим адресом с истекшим сроком жизни.

Таблица имеет столбец Dispн — «распоряжение», которое говорит мосту, какую операцию нужно проделать с кадром, имеющим данный адрес назначения. Обычно при автоматическом составлении таблицы в этом поле ставится условное обозначение порта назначения, но при ручном задании адреса в это поле можно внести нестандартную операцию обработки кадра. Например, операция Flood (затопление) заставляет мост распространять кадр в широковещательном режиме, несмотря на то что его адрес назначения не является широковещательным. Операция Discard (отбросить) говорит мосту, что кадр с таким адресом не нужно передавать на порт назначения.

Собственно операции, задаваемые в столбце Dispн, определяют особые условия фильтрации кадров, дополняющие стандартные условия их распространения. Такие условия обычно называют пользовательскими фильтрами.

Мосты с маршрутизацией от источника

Мосты с маршрутизацией от источника применяются для соединения колец Token Ring и FDDI, хотя для этих же целей могут использоваться и прозрачные мосты. Маршрутизация от источника (Source Routing, SR) основана на том, что станция- отправитель помещает в посылаемый в другое кольцо кадр всю адресную информацию о промежуточных мостах и кольцах, которые должен пройти кадр перед тем, как попасть в кольцо станции- получателя. Хотя в название этого способа входит термин «маршрутизация», настоящей маршрутизации в строгом понимании этого термина здесь нет, так как мосты и станции по-прежнему используют для передачи кадров данных только информацию MAC- уровня, а заголовки сетевого уровня для мостов данного типа остаются неразличимой ча-

стью поля данных кадра.

Рассмотрим принципы работы мостов Source Routing (в дальнейшем, SR-мосты) на примере сети, представленной на рис. 9.21. Сеть состоит из трех колец, соединенных тремя мостами. Для задания маршрута кольца и мосты имеют идентификаторы. SR-мосты не строят адресную таблицу, а при продвижении кадров пользуются информацией, имеющейся в соответствующих полях кадра данных.

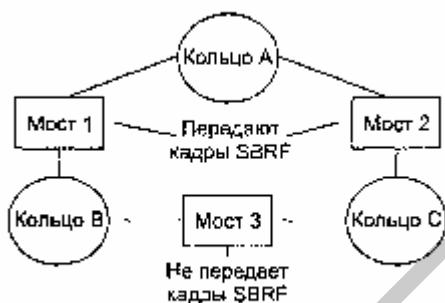


Рис. 9.21. Мости типа Source Routing

При получении каждого пакета SR-мосту нужно только просмотреть поле маршрутной информации (поле Routing information Field, RIF, в кадре Token Ring или FDDI) на предмет наличия в нем своего идентификатора. И если он там присутствует и сопровождается идентификатором кольца, которое подключено к данному мосту, то в этом случае мост копирует поступивший кадр в указанное кольцо. В противном случае кадр в другое кольцо не копируется. В любом случае исходная копия кадра возвращается по исходному кольцу станции-отправителю, и если он был передан в другое кольцо, то бит А (адрес распознан) и бит С (кадр скопирован) поля статуса кадра устанавливаются в 1, чтобы сообщить станции-отправителю, что кадр был получен станцией назначения (в данном случае передан мостом в другое кольцо).

Так как маршрутная информация в кадре нужна не всегда, а только для передачи кадра между станциями, подключенными к разным кольцам, то наличие в кадре поля RIF обозначается установкой в 1 бита индивидуального/группового адреса (I/G), при этом данный бит используется не по назначению, так как адрес источника всегда индивидуальный.

Поле RIF имеет управляющее подполе, состоящее из трех частей.

- Тип кадра определяет тип поля RIF. Существуют различные типы полей RIF, использующиеся для нахождения маршрута и для отправки кадра по известному маршруту.
- Поле максимальной длины кадра используется мостом для связи колец, в которых установлено различное значение MTU. С помощью этого поля мост уведомляет станцию о максимально возможной длине кадра (то есть минимальном значении MTU на протяжении всего составного маршрута).
- Длина поля RIF необходима, так как заранее неизвестно количество описателей маршрута, задающих идентификаторы пересекаемых колец и мостов.

Для работы алгоритма маршрутизации от источника используются два дополнительных типа кадра — одномаршрутный широковещательный кадр-исследователь (Single-Route Broadcast Frame, SRBF) и многомаршрутный широковещательный кадр-исследователь (All-Route Broadcast Frame, ARBF).

Вес SR-мосты должны быть сконфигурированы администратором вручную, чтобы передавать кадры ARBF на все порты, кроме порта-источника кадра, а для кадров SRBF некоторые порты мостов нужно заблокировать, чтобы в сети не было петель. В примере сети на рис. 4.21 для исключения петли администратор заблокировал оба порта моста 3 для передачи кадров SRBF.

Кадр первого типа отправляется станцией, когда она, во-первых, определяет, что станция назначения находится в другом кольце, а во-вторых, ей неизвестно, через какие мосты и кольца пролегает путь к этой станции назначения, то есть неизвестен маршрут до этой станции. Первое обстоятельство выясняется, если кадр, отправленный по кольцу, возвращается в станцию-источник с неустановленными признаками распознавания адреса и копирования. Значит, ни одна из станций исходного кольца не является станцией назначения и кадр надо передавать по некоторому составному маршруту. Отсутствие маршрута к станции назначения в таблице моста является вторым обстоятельством, которое и вызывает отправку одномаршрутного кадра-исследователя SRBF.

В кадре SRBF станция задает длину поля RIF, равную пулью. Как и прозрачные мосты, SR-мосты работают в режиме «неразборчивого» захвата, буферизуя и анализируя все кадры. При получении кадра SRBF SR-мост передает его в исходном виде на все незаблокированные для этого типа кадров порты. Необходимость в конфигурировании топологии без петель для кадров-исследователей SRBF вызвана тем, что таким способом предотвращается возможность бесконечного зацикливания этих кадров.

В конце концов кадр-исследователь SRBF, распространяясь по всем кольцам сети, доходит до станции назначения. В ответ станция назначения отправляет многомаршрутный широковещательный кадр-исследователь ARBF станции- отправителю. В отличие от кадра SRBF этот кадр передается мостами через все порты. При приеме такого кадра каждый промежуточный мост добавляет в поле маршрутной информации RIF новый описатель маршрута (свой идентификатор и идентификатор сегмента, с которого получен кадр), наращивает длину поля маршрутной информации и широковещательно его распространяет.

Для предотвращения зацикливания кадров ARBF мосты обрабатывают их следующим образом. Перед передачей кадра на какой-либо сегмент мост проверяет, нет ли идентификатора этого сегмента в списке маршрутов кадра. Если такой сегмент уже был пройден кадром, то кадр в данный сегмент не направляется.

Станция-источник получает в общем случае несколько кадров-ответов, прошедших по всем возможным маршрутам составной сети, и выбирает наилучший маршрут (обычно по количеству пересечений промежуточных мостов). Именно для получения информации обо всех возможных маршрутах кадр ARBF передается по всем возможным направлениям.

Затем маршрутная информация помещается в таблицу маршрутизации станции

и используется для отправки кадров данных станции назначения по наилучшему маршруту за счет помещения последовательности номеров сетей и мостов в заголовке каждого такого кадра.

Мосты с маршрутизацией от источника имеют по сравнению с прозрачными мостами как преимущества, так и недостатки, отраженные в табл. 0.1.

Таблица 9.1. Преимущества и недостатки мостов с маршрутизацией от источника

Преимущества	Недостатки
Более рациональные маршруты	Более дорогие сетевые адаптеры, принимающие участие в маршрутизации
Проще и дешевле – не нужно строить таблицы фильтрации	Сеть непрозрачна – кольца имеют номера
Более высокая скорость – не нужно просматривать таблицы фильтрации	Увеличивается трафик за счёт широковещательных пакетов

Наличие двух возможных алгоритмов работы мостов — от источника и в прозрачном режиме — создает трудности для построения сложных сетей Token Ring. Мосты, работающие от источника, не могут поддерживать сегменты, рассчитанные на работу в прозрачном режиме, и наоборот.

До некоторого времени эта проблема решалась двумя способами. Один способ заключался в использовании во всех сегментах либо только маршрутизации от источника, либо только прозрачных мостов. Другим способом была установка маршрутизаторов. Сегодня имеется третье решение. Оно основано на стандарте, который позволяет объединить обе технологии работы моста в одном устройстве. Этот стандарт, называемый SRT (Source Route Transparent), позволяет мосту работать в любом режиме. Мост просматривает специальные флаги в заголовке кадров Token Ring и автоматически определяет, какой из алгоритмов нужно применить.

Недостатки топологии сети на мостах

Слабая защита от широковещательного шторма — одно из главных ограничений моста, но не единственное. Другим серьезным ограничением их функциональных возможностей является невозможность поддержки петлеобразных конфигураций сети. Рассмотрим это ограничение на примере сети, показанной на рис. 9.22.

Два сегмента параллельно соединены двумя мостами, так что образовалась активная петля. Пусть новая станция с адресом 10 впервые начинает работу в данной сети. Обычно начало работы любой операционной системы сопровождается рассылкой широковещательных кадров, в которых станция заявляет о своем существовании и одновременно ищет серверы сети.

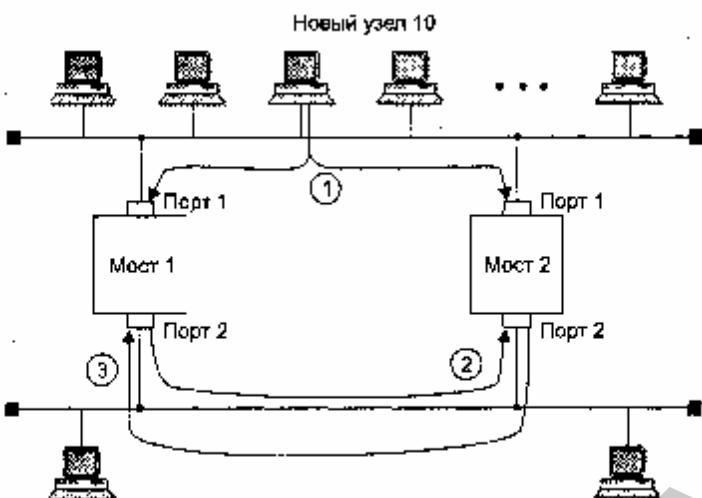


Рис. 9.22. Влияние замкнутых маршрутов на работу мостов

На этапе 1 станция посылает первый кадр с широковещательным адресом назначения и адресом источника 10 в свой сегмент. Кадр попадает как в мост 1, так и в мост 2. В обоих мостах новый адрес источника 10 заносится в адресную таблицу с пометкой о его принадлежности сегменту 1, то есть создается новая запись вида.

MAC-адрес	Порт
10	1

Так как адрес назначения широковещательный, то каждый мост должен передать кадр на сегмент 2. Эта передача происходит поочередно, в соответствии с методом случайного доступа технологии Ethernet. Пусть первым доступ к сегменту 2 получил мост 1 (этап 2 на рис. 9.22). При появлении пакета на сегменте 2 мост 2 принимает его в свой буфер и обрабатывает. Он видит, что адрес 10 уже есть в его адресной таблице, но пришедший кадр является более свежим, и он утверждает, что адрес 10 принадлежит сегменту 2, а не 1. Поэтому мост 2 корректирует содержимое базы и делает запись о том, что адрес 10 принадлежит сегменту 2.

Теперь адресная таблица моста 2 будет иметь уже другую запись о станции с адресом 10.

MAC-адрес	Порт
10	1
10	2

Аналогично поступает мост 1, когда мост 2 передает свою копию кадра на сегмент 2.

Результаты наличия петли перечислены ниже.

- «Размножение» кадра, то есть появление нескольких его копий (в данном случае — двух, но если бы сегменты были соединены тремя мостами — то трех и т. д.).

□ Бесконечная циркуляция обеих копий кадра по петле в противоположных направлениях, а значит, засорение сети ненужным трафиком.

□ Постоянная перестройка мостами своих адресных таблиц, так как кадр с адресом источника 10 будет появляться то на одном порту, то на другом.

Чтобы исключить все эти нежелательные эффекты, мосты нужно применять так, чтобы между логическими сегментами не было петель, то есть строить с помощью мостов только древовидные структуры, гарантирующие наличие только одного пути между любыми двумя сегментами. Тогда кадры от каждой станции будут поступать в мост всегда с одного и того же порта, и мост сможет правильно решать задачу выбора рационального маршрута в сети.

Ограничение топологии структурированной сети древовидной структурой вытекает из самого принципа построения адресной таблицы мостом, а поэтому точно так же это ограничение действует и на коммутаторы.

В простых сетях сравнительно легко гарантировать существование одного и только одного пути между двумя сегментами. Но когда количество соединений возрастает и сеть становится сложной, то вероятность непреднамеренного образования петли оказывается высокой. Кроме того, желательно для повышения надежности иметь между мостами резервные связи, которые не участвуют при нормальной работе основных связей в передаче информационных пакетов станций, но при отказе какой-либо основной связи образуют новую связную рабочую конфигурацию без петель.

Поэтому в сложных сетях между логическими сегментами прокладывают избыточные связи, которые образуют петли, но для исключения активных петель блокируют некоторые порты мостов. Проще всего эта задача решается вручную, но существуют и алгоритмы, которые позволяют решать ее автоматически. Наиболее известным является стандартный алгоритм покрывающего дерева (Spanning Tree Algorithm, STA), который будет детально рассмотрен ниже. Кроме того, имеются фирменные алгоритмы, решающие ту же задачу, по некоторыми улучшениями для конкретных моделей коммутаторов.

Выводы

□ Логическая структуризация сети необходима при построении сетей средних и крупных размеров. Использование общей разделяемой среды приемлемо только для сети, состоящей из 5-10 компьютеров.

□ Деление сети на логические сегменты повышает производительность, надежность, гибкость построения и управляемость сети.

□ Для логической структуризации сети применяются мосты и их современные преемники — коммутаторы и маршрутизаторы. Первые два типа устройств позволяют разделить сеть на логические сегменты с помощью минимума средств — только на основе протоколов канального уровня. Кроме того; эти устройства не требуют конфигурирования.

□ Логические сегменты, построенные на основе коммутаторов, являются строительными элементами более крупных сетей, объединяемых маршрутизаторами.

Коммутаторы и полнодуплексные протоколы локальных сетей

Технология коммутации сегментов Ethernet была предложена фирмой Kalpana в 1990 году в ответ на растущие потребности в повышении пропускной способности связей высокопроизводительных серверов с сегментами рабочих станций.

Структурная схема коммутатора Ether Switch, предложенного фирмой Kalpana, представлена на рис. 9.23.

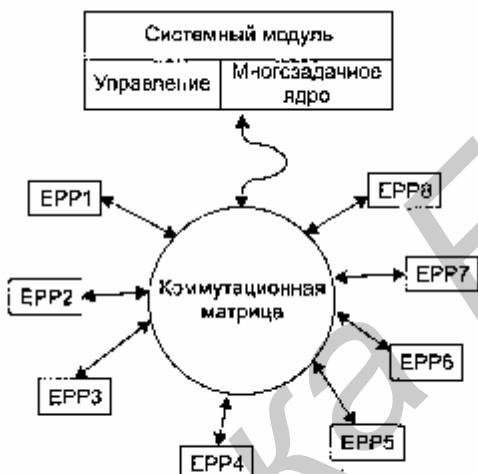


Рис. 9.23. Структура коммутатора EtherSwitch компании Kalpana

Каждый из 8 портов 10Base-T обслуживается одним процессором пакетов Ethernet (Ethernet Packet Processor, EPP). Кроме того, коммутатор имеет системный модуль, который координирует работу всех процессоров EPP. Системный модуль ведет общую адресную таблицу коммутатора и обеспечивает управление коммутатором по протоколу SNMP. Для передачи кадров между портами используется коммутационная матрица, подобная тем, которые работают в телефонных коммутаторах или мультипроцессорных компьютерах, соединяя несколько процессоров с несколькими модулями памяти.

Коммутационная матрица функционирует по принципу коммутации каналов. Для 8 портов матрица может обеспечить 8 одновременных внутренних каналов при полудуплексном режиме работы портов и 16 — при полнодуплексном, когда передатчик и приемник каждого порта работают независимо друг от друга. При поступлении кадра в какой-либо порт процессор EPP буферизует несколько первых байтов кадра, чтобы прочитать адрес назначения. После получения адреса назначения процессор сразу же принимает решение о передаче пакета, не дожидаясь прихода остальных байтов кадра. Для этого он просматривает свой собственный кэш адресной таблицы, а если не находит там нужного адреса, обращается к системному модулю, который работает в многозадачном режиме, параллельно обслуживая запросы всех процессоров EPP. Системный модуль производит просмотр общей адресной таблицы и возвращает процессору

найденную строку, которую тот буферизует в своем кэше для последующего использования.

После нахождения адреса назначения процессор ЕРР знает, что нужно дальше делать с поступающим кадром (во время просмотра адресной таблицы процессор продолжал буферизацию поступающих в порт байтов кадра). Если кадр нужно отфильтровать, процессор просто прекращает записывать в буфер байты кадра, очищает буфер и ждет поступления нового кадра.

Если же кадр нужно передать на другой порт, то процессор обращается к коммутационной матрице и пытается установить в ней путь, связывающий его порт с портом, через который идет маршрут к адресу назначения. Коммутационная матрица может это сделать только в том случае, когда порт адреса назначения в этот момент свободен, то есть не соединен с другим портом.

Если же порт занят, то, как и в любом устройстве с коммутацией каналов, матрица в соединении отказывает. В этом случае кадр полностью буферизуется процессором входного порта, после чего процессор ожидает освобождения выходного порта и образования коммутационной матрицей нужного пути.

После того как нужный путь установлен, в него направляются буферизованные байты кадра, которые принимаются процессором выходного порта. Как только процессор выходного порта получает доступ к подключенному к нему сегменту Ethernet по алгоритму CSMA/CD, байты кадра сразу же начинают передаваться в сеть. Процессор входного порта постоянно хранит несколько байтов принятого кадра в своем буфере, что позволяет ему независимо и асинхронно принимать и передавать байты кадра (рис. 9.24).

При свободном в момент приема кадра состоянии выходного порта задержка между приемом первого байта кадра коммутатором и появлением этого же байта на выходе порта адреса назначения составляла у коммутатора компании Kalpana всего 40 мкс, что было гораздо меньше задержки кадра при его передаче мостом.

Описанный способ передачи кадра без его полной буферизации получил название коммутации «на лету» («on-the-fly») или «напролет» («cut-through»). Этот способ представляет, по сути, конвейерную обработку кадра, когда частично совмещаются во времени несколько этапов его передачи (рис. 9.25).

1. Прием первых байтов кадра процессором входного порта, включая прием байтов адреса назначения.
2. Поиск адреса назначения в адресной таблице коммутатора (в кэше процессора или в общей таблице системного модуля).
3. Коммутация матрицы.
4. Прием остальных байтов кадра процессором входного порта.
5. Прием байтов кадра (включая первые) процессором выходного порта через коммутационную матрицу.
6. Получение доступа к среде процессором выходного порта.
7. Передача байтов кадра процессором выходного порта в сеть.

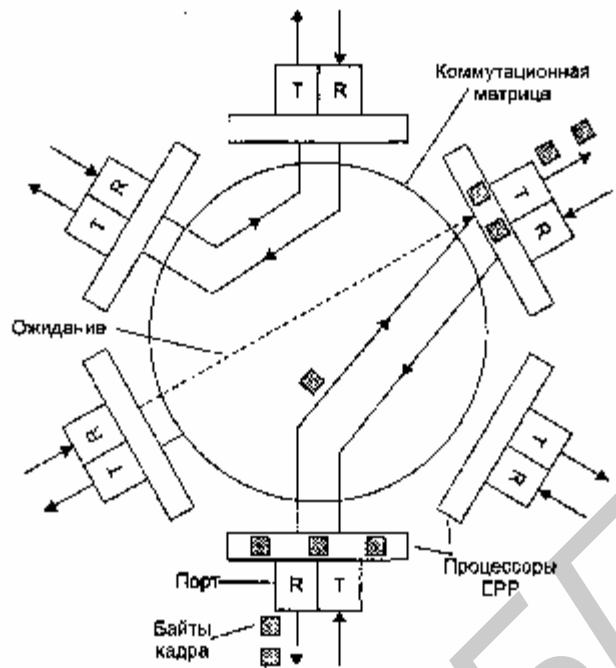


Рис. 9.24. Передача кадра через коммутационную матрицу

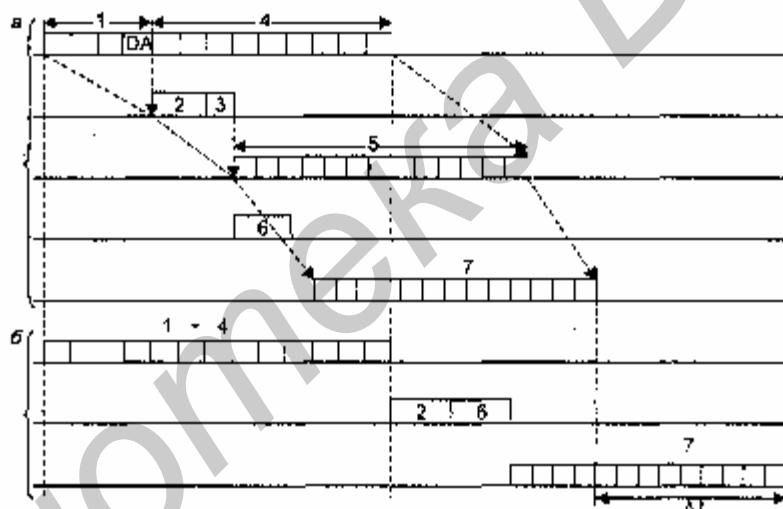


Рис. 9.25. Экономия времени при конвейерной обработке кадра: а — конвейерная обработка; б — обычная обработка с полной буферизацией

Этапы 2 и 3 совместить во времени нельзя, так как без знания номера выходного порта операция коммутации матрицы не имеет смысла.

По сравнению с режимом полной буферизации кадра, также представленном на рис. 9.25, экономия от конвейеризации получается ощутимой.

Однако главной причиной повышения производительности сети при использовании коммутатора является параллельная обработка нескольких кадров.

Этот эффект иллюстрирует рис. 9.26, на котором показана идеальная в отношении повышения производительности ситуация, когда четыре порта из восьми передают данные с максимальной для протокола Ethernet скоростью 10 Мбит/с, причем они передают эти данные на остальные четыре порта коммутатора не конфликтую — потоки данных между узлами сети распределились так, что для каждого принимающего кадры порта есть свой выходной порт. Если коммута-

тор успевает обрабатывать входной трафик даже при максимальной интенсивности поступления кадров на входные порты, то общая производительность коммутатора в приведенном примере составит $4 \times 10 - 40$ Мбит/с, а при обобщении примера для Апортов — $(JV/2) \times 10$ Мбит/с. Говорят, что коммутатор предоставляет каждой станции или сегменту, подключенным к его портам, выделенную пропускную способность протокола.

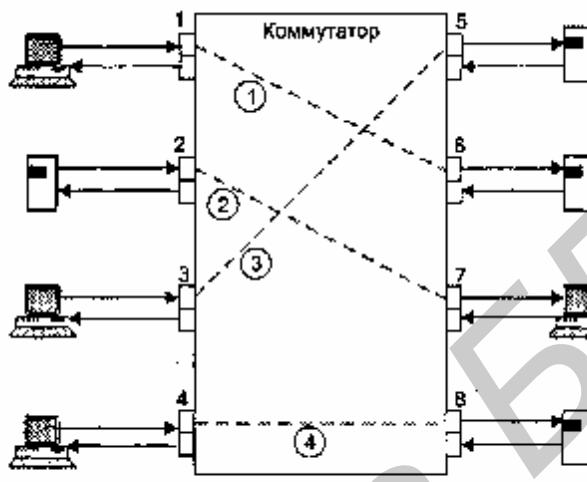


Рис. 9.26. Параллельная передача кадров коммутатором

Естественно, что в сети не всегда складывается описанная ситуация. Если двум станциям, например станциям, подключенным к портам 3 и 4, одновременно нужно записывать данные на один и тот же сервер, подключенный к порту 8, то коммутатор не сможет выделить каждой станции поток данных по 10 Мбит/с, так как порт 8 не может передавать данные со скоростью 20 Мбит/с. Кадры станций будут ожидать во внутренних очередях входных портов 3 и 4, когда освободится порт 8 для передачи очередного кадра. Очевидно, хорошим решением для такого распределения потоков данных было бы подключение сервера к более высокоскоростному порту, например Fast Ethernet.

Так как главное достоинство коммутатора, благодаря которому он завоевал очень хорошие позиции в локальных сетях, — его высокая производительность, то разработчики коммутаторов стараются выпускать так называемые неблокирующие (nonblocking) модели коммутаторов.

Неблокирующий коммутатор — это такой коммутатор, который может передавать кадры через свои порты с той же скоростью, с которой они на них поступают. Естественно, что даже неблокирующий коммутатор не может разрешить в течение долгого промежутка времени ситуации, подобные описанной выше, когда блокировка кадров происходит из-за ограниченной скорости выходного порта.

Обычно имеют в виду устойчивый неблокирующий режим работы коммутатора, когда коммутатор передает кадры со скоростью их поступления в течение произвольного промежутка времени. Для обеспечения такого режима нужно,

естественно, такое распределение потоков кадров по выходным портам, чтобы они справлялись с нагрузкой и коммутатор мог всегда в среднем передать на выходы столько кадров, сколько их поступило на входы. Если же входной поток кадров (просуммированный по всем портам) в среднем будет превышать выходной поток кадров (также просуммированный по всем портам), то кадры будут накапливаться в буферной памяти коммутатора, а при превышении ее объема — просто отбрасываться. Для обеспечения неблокирующего режима коммутатора необходимо выполнение достаточно простого условия:

$$C_k = (\sum C_{pi})/2.$$

Здесь C_k — производительность коммутатора, C_{pi} — максимальная производительность протокола, поддерживаемого i -м портом коммутатора. Суммарная производительность портов учитывает каждый проходящий кадр дважды — как входящий кадр и как выходящий, а так как в устойчивом режиме входной трафик равен выходному, то минимально достаточная производительность коммутатора для поддержки неблокирующего режима равна половине суммарной производительности портов. Если порт работает в полудуплексном режиме, например Ethernet 10 Мбит/с, то производительность порта C_{pi} равна 10 Мбит/с, а если в полнодуплексном — 20 Мбит/с.

Иногда говорят, что коммутатор поддерживает мгновенный неблокирующий режим. Это означает, что он может принимать и обрабатывать кадры от всех своих портов на максимальной скорости протоколов, независимо от того, обеспечиваются ли условия устойчивого равновесия между входным и выходным трафиком. Правда, обработка некоторых кадров при этом может быть неполной — при занятости выходного порта кадр помещается в буфер коммутатора. Для поддержки неблокирующего мгновенного режима коммутатор должен обладать большей собственной производительностью, а именно, она должна быть равна суммарной производительности его портов:

$$C_k = \sum C_{pi}.$$

Первый коммутатор для локальных сетей не случайно появился для технологии Ethernet. Кроме очевидной причины, связанной с наибольшей популярностью сетей Ethernet, существовала и другая, не менее важная причина — эта технология больше других страдает от увеличения времени ожидания доступа к среде при повышении загрузки сегмента. Поэтому сегменты Ethernet в крупных сетях в первую очередь нуждались в средстве разгрузки узких мест сети, и этим средством стали коммутаторы фирмы Kalpana, а затем и других компаний.

Некоторые компании стали развивать технологию коммутации для повышения производительности других технологий локальных сетей, таких как Token Ring и FDDI. Коммутаторы этих компаний поддерживали как алгоритм работы прозрачного моста, так и алгоритм моста с маршрутизацией от источника. Внутренняя организация коммутаторов различных производителей иногда очень отличалась от структуры первого коммутатора Ether Switch, однако принцип параллельной обработки кадров по каждому порту оставался неизменным.

Широкому применению коммутаторов, безусловно, способствовало то обстоятельство, что внедрение технологии коммутации не требовало замены установленного в сетях оборудования — сетевых адаптеров, концентраторов, кабель-

ной системы. Порты коммутаторов работали в обычном полудуплексном режиме, поэтому к ним прозрачно можно было подключить как конечный узел, так и концентратор, организующий целый логический сегмент.

Так как коммутаторы и мосты прозрачны для протоколов сетевого уровня, то их появление в сети не оказало никакого влияния на маршрутизаторы сети, если они там имелись.

Удобство использования коммутатора состоит еще и в том, что это самообучающееся устройство и, если администратор не нагружает его дополнительными функциями, конфигурировать его не обязательно — нужно только правильно подключить разъемы кабелей к портам коммутатора, а дальше он будет работать самостоятельно и эффективно выполнять поставленную перед ним задачу повышения производительности сети.

Изменения в работе MAC-уровня при полнодуплексной работе

Технология коммутации сама по себе не имеет непосредственного отношения к методу доступа к среде, который используется портами коммутатора. При подключении сегментов, представляющих собой разделяемую среду, порт коммутатора должен поддерживать полудуплексный режим, так как является одним из узлов этого сегмента.

Однако когда к каждому порту коммутатора подключен не сегмент, а только один компьютер, причем по двум раздельным каналам, как это происходит почти во всех стандартах физического уровня, кроме коаксиальных версий Ethernet, ситуация становится не такой однозначной. Порт может работать как в обычном полудуплексном режиме, так и в полнодуплексном. Подключение к портам коммутатора не сегментов, а отдельных компьютеров называется микросегментацией.

В обычном режиме работы порт коммутатора по-прежнему распознает коллизии. Доменом коллизий в этом случае является участок сети, включающий передатчик коммутатора, приемник коммутатора, передатчик сетевого адаптера компьютера, приемник сетевого адаптера компьютера и две витые пары, соединяющие передатчики с приемниками (рис. 9.27).

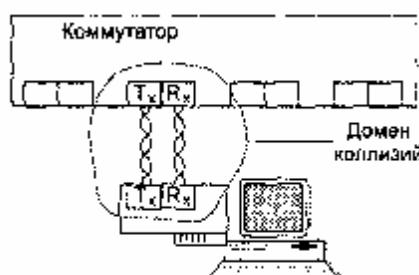


Рис. 9.27. Домен коллизий, образуемый коммутатором и портом коммутатора

Коллизия возникает, когда передатчики порта коммутатора и сетевого адаптера одновременно или почти одновременно начинают передачу своих кадров, считая, что сегмент свободен (см. рисунок). Правда, вероятность коллизии в таком сегменте гораздо меньше, чем в сегменте, состоящем из 20-30 узлов, но она не нулевая. При этом максимальная производительность сегмента Ethernet в 14 880 кадров в секунду при минимальной длине кадра делится между передатчиком порта коммутатора и передатчиком сетевого адаптера. Если считать, что она делится пополам, то каждому предоставляется возможность передавать примерно по 7440 кадров в секунду.

В полнодуплексном режиме одновременная передача данных передатчиком порта коммутатора и сетевого адаптера коллизией не считается. В принципе это достаточно естественный режим работы для индивидуальных полнодуплексных каналов связи и он часто используется в протоколах территориальных сетей. При полнодуплексной связи порты Ethernet могут передавать данные со скоростью 20 Мбит/с — по 10 Мбит/с в каждом направлении.

Естественно, необходимо, чтобы МАС-узлы взаимодействующих устройств поддерживали этот специальный режим. В случае, когда только один узел поддерживает полнодуплексный режим, второй узел будет постоянно фиксировать коллизии и приостанавливать свою работу, в то время как другой узел будет продолжать передавать данные, которые никто в этот момент не принимает. Изменения, которые нужно сделать в логике МАС-узла, чтобы он мог работать в полнодуплексном режиме, минимальны — нужно просто отменить фиксацию и отработку коллизий в сетях Ethernet, а в сетях Token Ring и FDDI — посыпать кадры в коммутатор, не дожидаясь прихода маркера доступа, а тогда, когда это нужно конечному узлу. Фактически при работе в полнодуплексном режиме МАС-узел не использует метод доступа к среде, разработанный для данной технологии.

Так как переход на полнодуплексный режим работы требует изменения логики работы МАС-узлов и драйверов сетевых адаптеров, то он сначала был опробован при соединении двух коммутаторов. Уже первые модели коммутатора EtherSwitch компании Kalpana поддерживали полнодуплексный режим при взаимном соединении, обеспечивая скорость взаимного обмена 20 Мбит/с.

Позже появились версии полнодуплексного соединения FDDI-коммутаторов, которые при одновременном использовании двух колец FDDI обеспечивали скорость обмена в 200 Мбит/с.

Сейчас для каждой технологии можно найти модели коммутаторов, которые поддерживают полнодуплексный обмен при соединении коммутатор—коммутатор.

После опробования полнодуплексной технологии на соединениях коммутатор—коммутатор разработчики реализовали ее и в сетевых адаптерах, в основном адаптерах Ethernet и Fast Ethernet. При разработке технологий Fast Ethernet и Gigabit Ethernet полнодуплексный режим стал одним из двух полноправных стандартных режимов работы узлов сети. Многие сетевые адаптеры сейчас могут поддерживать оба режима работы, отрабатывая логику алгоритма доступа CSMA/CD при подключении к порту концентратора и работая в полнодуплекс-

ном режиме при подключении к порту коммутатора.

При использовании полнодуплексных версий протоколов происходит некоторое сближение различных технологий, так как лицо каждой технологии во многом определял метод доступа. Различие технологий остается в разных форматах кадров, а также в процедурах контроля корректности работы сети на физическом и канальном уровнях.

Полнодуплексные версии протоколов могли бы быть реализованы и в мостах. Принципиальных препятствий для этого не было, просто в период применения локальных мостов потребности в высокоскоростной передаче межсегментного трафика не возникало.

Проблема управления потоком данных при полнодуплексной работе

Простой отказ от поддержки алгоритма доступа к разделяемой среде без какой-либо модификации протокола ведет к повышению вероятности потерь кадров коммутаторами, так как при этом теряется контроль за потоками кадров, направляемых конечными узлами в сеть. Раньше поток кадров регулировался методом доступа к разделяемой среде, так что слишком часто генерирующий кадры узел вынужден был ждать своей очереди к среде и фактическая интенсивность потока данных, который направлял в сеть этот узел, была заметно меньше той интенсивности, которую узел хотел бы отправить и сеть. При переходе на полнодуплексный режим узлу разрешается отправлять кадры в коммутатор всегда, когда это ему нужно, поэтому коммутаторы сети могут в этом режиме сталкиваться с перегрузками, не имея при этом никаких средств регулирования («притормаживания») потока кадров.

Причина перегрузок обычно кроется не в том, что коммутатор является блокирующим, то есть ему не хватает производительности процессоров для обслуживания потоков кадров, а в ограниченной пропускной способности отдельного порта, которая определяется временными параметрами протокола. Например, порт Ethernet без нарушения временных соотношений, установленных стандартом, не может передавать больше 14 880 кадров/с.

Причина перегрузок обычно кроется не в том, что коммутатор является блокирующим, то есть ему не хватает производительности процессоров для обслуживания потоков кадров, а в ограниченной пропускной способности отдельного порта, которая определяется временными параметрами протокола. Например, порт Ethernet без нарушения временных соотношений, установленных стандартом, не может передавать больше 14 880 кадров/с.

Поэтому, если входной трафик неравномерно распределяется между выходными портами, легко представить ситуацию, когда в какой-либо выходной порт коммутатора будет направляться трафик с суммарной средней интенсивностью большей, чем протокольный максимум. На рис. 9.28 показана как раз такая ситуация, когда в порт 3 коммутатора направляется трафик от портов 1, 2, 4 и 6 с суммарной интенсивностью в 22 100 кадров/с. Порт 3 оказывается загружен-

ным на 150 %. Естественно, что когда кадры поступают в буфер порта со скоростью 20 100 кадров/с, а уходят со скоростью 14 880 кадров/с, то внутренний буфер выходного порта начинает неуклонно заполняться необработанными кадрами.

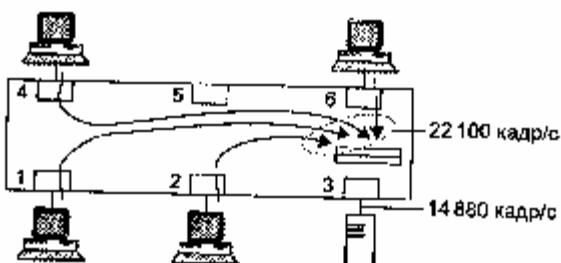


Рис. 9.28. Переполнение буфера порта из-за несбалансированности трафика

Какой бы ни был объем буфера порта, он в какой-то момент времени обязательно переполнится. Нетрудно подсчитать, что при размере буфера в 100 Кбайт в приведенном примере полное заполнение буфера произойдет через 0,22 с после начала его работы (буфер такого размера может хранить до 1600 кадров размером в 64 байт). Увеличение буфера до 1 Мбайт даст увеличение времени заполнения буфера до 2,2 с, что также неприемлемо. А потери кадров всегда очень нежелательны, так как снижают полезную производительность сети, и коммутатор, теряющий кадры, может значительно ухудшить производительность сети вместо ее улучшения.

Коммутаторы локальных сетей — не первые устройства, которые сталкиваются с такой проблемой. Мосты также могут испытывать перегрузки, однако такие ситуации при использовании мостов встречались редко из-за небольшой интенсивности межсегментного трафика, поэтому разработчики мостов не стали встраивать в протоколы локальных сетей или в сами мосты механизмы регулирования потока. В глобальных сетях коммутаторы технологии X.25 поддерживают протокол канального уровня LAP-B, который имеет специальные кадры управления потоком «приемник готов» (RR) и «приемник не готов» (RNR), аналогичные по назначению кадрам протокола LLC2 (это не удивительно, так как оба протокола принадлежат семейству протоколов HDLC). Протокол LAP-B работает между соседними коммутаторами сети X.25 и в том случае, когда очередь коммутатора доходит до опасной границы, запрещает своим ближайшим соседям с помощью кадра «приемник не готов» передавать ему кадры, пока очередь не уменьшится до нормального уровня. В сетях X.25 такой протокол необходим, так как эти сети никогда не использовали разделяемые среды передачи данных, а работали по индивидуальным каналам связи в полнодуплексном режиме.

При разработке коммутаторов локальных сетей ситуация коренным образом отличалась от ситуации, при которой создавались коммутаторы территориальных сетей. Основной задачей было сохранение конечных узлов в неизменном виде, что исключало корректировку протоколов локальных сетей. А в этих про-

токолах процедур управления потоком не было — общая среда передачи данных в режиме разделения времени исключала возникновение ситуаций, когда сеть переполнялась бы необработанными кадрами. Сеть не накапливала данных в каких-либо промежуточных буферах и при использовании только повторителей или концентраторов.

ПРИМЕЧАНИЕ

Здесь речь идет о протоколах MAC-уровня (Ethernet, Token Ring и т. п.), так как мосты и коммутаторы имеют дело только с ними. Протокол LLC2, который умеет управлять потоком данных, для целей управления потоком кадров в коммутаторах использовать нельзя. Для коммутаторов протокол LLC (все его процедуры: 1, 2 и 3) прозрачен, как и все остальные протоколы верхних уровней, — коммутатор не анализирует заголовок LLC, считая его просто полем данных кадра MAC-уровня.

Применение коммутаторов без изменения протокола работы оборудования всегда порождает опасность потери кадров. Если порты коммутатора работают в обычном, то есть в полудуплексном режиме, то у коммутатора имеется возможностьказать некоторое воздействие на конечный узел и заставить его приостановить передачу кадров, пока у коммутатора не разгрусятся внутренние буферы. Нестандартные методы управления потоком в коммутаторах при сохранении протокола доступа в неизменном виде будут рассмотрены ниже.

Если же коммутатор работает в полнодуплексном режиме, то протокол работы конечных узлов, да и его портов все равно меняется. Поэтому имело смысл для поддержки полнодуплексного режима работы коммутаторов несколько модифицировать протокол взаимодействия узлов, встроив в него явный механизм управления потоком кадров.

Работа над выработкой стандарта для управления потоком кадров в полнодуплексных версиях Ethernet и Fast Ethernet продолжалась несколько лет. Такой длительный период объясняется разногласиями членов соответствующих комитетов по стандартизации, отстаивающих подходы фирм, которые реализовали в своих коммутаторах собственные методы управления потоком.

В марте 1997 года принят стандарт IEEE 802.3x на управление потоком в полнодуплексных версиях протокола Ethernet. Он определяет весьма простую процедуру управления потоком, подобную той, которая используется в протоколах LLC2 и LAP-B. Эта процедура подразумевает две команды — «приостановить передачу» и «возобновить передачу», которые направляются соседнему узлу. Отличие от протоколов типа LLC2 в том, что эти команды реализуются на уровне символов кодов физического уровня, таких как 4B/5B, а не на уровне команд, оформленных в специальные управляющие кадры. Сетевой адаптер или порт коммутатора, поддерживающий стандарт 802.3x и получивший команду «приостановить передачу», должен прекратить передавать кадры впредь до получения команды «возобновить передачу».

Некоторые специалисты высказывают опасение, что такая простая процедура управления потоком окажется непригодной в сетях Gigabit Ethernet. Полная приостановка приема кадров от соседа при такой большой скорости передачи

кадров (1 488 090 кадр/с) может быстро вызвать переполнение внутреннего буфера теперь у этого соседа, который, в свою очередь, полностью заблокирует прием кадров у своих ближайших соседей. Таким образом, перегрузка просто распространится по сети, вместо того чтобы постепенно исчезнуть. Для работы с такими скоростными протоколами необходим более тонкий механизм регулирования потока, который бы указывал, на какую величину нужно уменьшить интенсивность потока входящих кадров в перегруженный коммутатор, а не приостанавливал этот поток до нуля. Подобный плавный механизм регулирования потока появился у коммутаторов ATM через несколько лет после их появления. Поэтому существует мнение, что стандарт 802.3x — это временное решение, которое просто закрепило существующие фирменные простые механизмы управления потоком ведущих производителей коммутаторов. Пройдет некоторое время, и этот стандарт сменит другой стандарт — более сложный и более гибкий. Но возможен и иной вариант, ведь нет ничего более постоянного, чем временное.

Управление потоком кадров при полудуплексной работе

При работе порта в полудуплексном режиме коммутатор не может изменять протокол и пользоваться для управления потоком новыми командами, такими как «приостановить передачу» и «возобновить передачу».

Зато у коммутатора появляется возможность воздействовать на конечный узел с помощью механизмов алгоритма доступа к среде, который конечный узел обязан отрабатывать. Эти приемы основаны на том, что конечные узлы строго соблюдают все параметры алгоритма доступа к среде, а порты коммутатора — нет. Обычно применяются два основных способа управления потоком кадров — обратное давление на конечный узел и агрессивный захват среды.

Метод обратного давления (backpressure) состоит в создании искусственных коллизий в сегменте, который чересчур интенсивно посыпает кадры в коммутатор. Для этого коммутатор обычно использует юм-последовательность, отправляемую на выход порта, к которому подключен сегмент (или узел), чтобы приостановить его активность. Кроме того, метод обратного давления может применяться в тех случаях, когда процессор порта не рассчитан на поддержку максимально возможного для данного протокола трафика. Один из первых примеров применения метода обратного давления как раз связан с таким случаем — метод был использован компанией LANNET к 1994 году в модулях LSE-1 и LSE-2, рассчитанных на коммутацию трафика Ethernet с максимальной интенсивностью соответственно 1 Мбит/с и 2 Мбит/с.

Второй метод «торможения» конечного узла в условиях перегрузки внутренних буферов коммутатора основан на так называемом агрессивном поведении порта коммутатора при захвате среды либо после окончания передачи очередного пакета, либо после коллизии. Эти два случая иллюстрирует рис. 9.29.

В первом случае (рис. 9.29, а) коммутатор окончил передачу очередного кадра и вместо технологической паузы в 9,6 мкс сделал паузу в 9,1 мкс и начал передачу нового кадра. Компьютер не смог захватить среду, так как он выдержал

стандартную паузу в 9,6 мкс и обнаружил после этого, что среда уже занята.

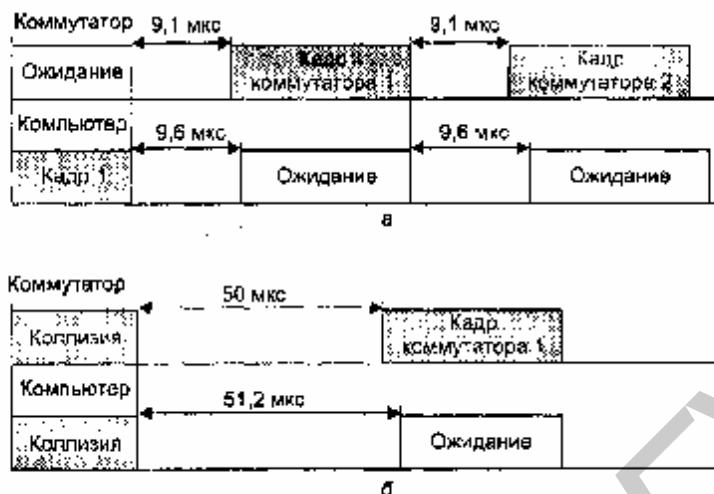


Рис. 9.29. Агрессивное поведение коммутатора при перегрузках буферов

Во втором случае (рис. 9.29, б) кадры коммутатора и компьютера столкнулись, и была зафиксирована коллизия. Так как компьютер сделал паузу после коллизии в 51,2 мкс, как это положено по стандарту (интервал отсрочки равен 512 битовых интервалов), а коммутатор — 50 мкс, то и в этом случае компьютеру не удалось передать свой кадр.

Коммутатор может пользоваться этим механизмом адаптивно, увеличивая степень своей агрессивности по мере необходимости.

Многие производители путем сочетания этих двух методов реализуют достаточно тонкие механизмы управления потоком кадров при перегрузках. Эти методы используют алгоритмы чередования передаваемых и принимаемых кадров (frame interleave). Алгоритм чередования должен быть гибким и позволять компьютеру в критических ситуациях на каждый принимаемый: кадр передавать несколько своих, разгружая внутренний буфер кадров, причем не обязательно снижая при этом интенсивность приема кадров до нуля, а просто уменьшая ее до необходимого уровня.

Практически во всех моделях коммутаторов, кроме самых простых моделей для рабочих групп, реализуют тот или иной алгоритм управления потоком кадров при полудуплексном режиме работы портов. Этот алгоритм, как правило, обеспечивает более тонкое управление потоком, чем стандарт 802.3x, не приостанавливая до нуля прием кадров от соседнего узла и тем самым не способствуя переносу перегрузки в соседний коммутатор, если к порту подключен не конечный узел, а другой коммутатор.

Выводы

- Коммутаторы — наиболее быстродействующие современные коммуникационные устройства, позволяющие соединять высокоскоростные сегменты без блокирования (уменьшения пропускной способности) межсегментного трафи-

ка.

- Пассивный способ построения адресной таблицы коммутаторами — путем слежения за проходящим трафиком — приводит к невозможности работы в сетях с петлевидными связями. Другим недостатком сетей, построенных на коммутаторах, является отсутствие защиты от широковещательного шторма, который эти устройства обязаны передавать в соответствии с алгоритмом работы.
- Применение коммутаторов позволяет сетевым адаптерам использовать полнодуплексный режим работы протоколов локальных сетей (Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI). В этом режиме отсутствует этап доступа к разделяемой среде, а общая скорость передачи данных удваивается.
- В полнодуплексном режиме для борьбы с перегрузками коммутаторов используется метод управления потоком, описанный в стандарте 802.3x. Он повторяет алгоритмы полной приостановки трафика по специальной команде, известной из технологий глобальных сетей.
- При полудуплексном режиме работы коммутаторы используют для управления потоком при перегрузках два метода: агрессивный захват среды и обратное давление на конечный узел. Применение этих методов позволяет достаточно гибко управлять потоком, чередуя несколько передаваемых кадров с одним принимаемым.

Особенности реализации коммутаторов

Несмотря на то, что в коммутаторах работают известные и хорошо отработанные алгоритмы прозрачных мостов и мостов с маршрутизацией от источника, существует большое разнообразие моделей коммутаторов. Они отличаются как внутренней организацией, так и набором выполняемых дополнительных функций, таких как трансляция протоколов, поддержка алгоритма покрывающего дерева, образование виртуальных логических сетей и ряда других.

После того как технология коммутации привлекла общее внимание и получила высокие оценки специалистов, многие компании занялись реализацией этой технологии в своих устройствах, применяя для этого различные технические решения. Многие коммутаторы первого поколения были похожи на маршрутизаторы, то есть основывались на центральном процессоре общего назначения, связанном с интерфейсными портами по внутренней скоростнойшине (рис. 9.30). Однако это были скорее пробные устройства, предназначенные для освоения самой компанией технологии коммутации, а не для завоевания рынка.

Основным недостатком таких коммутаторов была их низкая скорость. Универсальный процессор никак не мог справиться с большим объемом специализированных операций по пересылке кадров между интерфейсными модулями.

Для ускорения операций коммутации нужны были специализированные процессоры со специализированными средствами обмена данными, как в первом коммутаторе Kalpana, и они вскоре появились. Сегодня все коммутаторы используют заказные специализированные БИС — ASIC, которые оптимизированы для выполнения основных операций коммутации. Часто в одном коммута-

торе, имеется несколько специализированных БИС, каждая из которых выполняет функционально законченную часть операций. Сравнительно низкая стоимость современных коммутаторов по сравнению с их предшественниками 3-5-летней давности объясняется массовым характером производства основных БИС, на которых каждая компания строит свои коммутаторы.

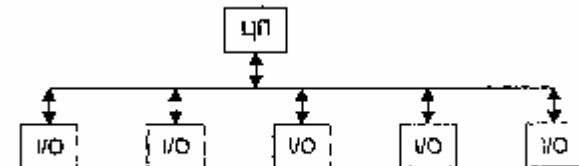


Рис. 9.30. Коммутатор на процессоре общего назначения

Кроме процессорных микросхем для успешной неблокирующей работы коммутатору нужно также иметь быстродействующий узел для передачи кадров между процессорными микросхемами портов.

В настоящее время коммутаторы используют в качестве базовой одну из трех схем, на которой строится такой узел обмена;

- коммутационная матрица;
- разделяемая многовходовая память;
- общая шина.

Часто эти три схемы взаимодействия комбинируются в одном коммутаторе.

Архитектура и конструктивное исполнение коммутаторов

Коммутаторы на основе коммутационной матрицы

Коммутационная матрица обеспечивает основной и самый быстрый способ взаимодействия процессоров портов, именно он был реализован в первом промышленном коммутаторе локальных сетей. Однако реализация матрицы возможна только для определенного числа портов, причем сложность схемы возрастает пропорционально квадрату количества портов коммутатора (рис. 9.31).

Более детальное представление одного из возможных вариантов реализации коммутационной матрицы для 8 портов дано на рис. 9.32. Входные блоки процессоров портов на основании просмотра адресной таблицы коммутатора определяют по адресу назначения номер выходного порта. Эту информацию они добавляют к байтам исходного кадра в виде специального ярлыка — тега (tag). Для данного примера тег представляет собой просто 3-разрядное двоичное число, соответствующее номеру выходного порта.

Матрица состоит из трех уровней двоичных переключателей, которые соединяют спой вход с одним из двух выходов в зависимости от значения бита тега. Переключатели первого уровня управляются первым битом тега, второго — вторым, а третьего — третьим.

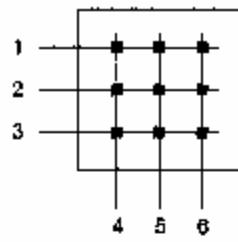


Рис. 9.31. Коммутационная матрица

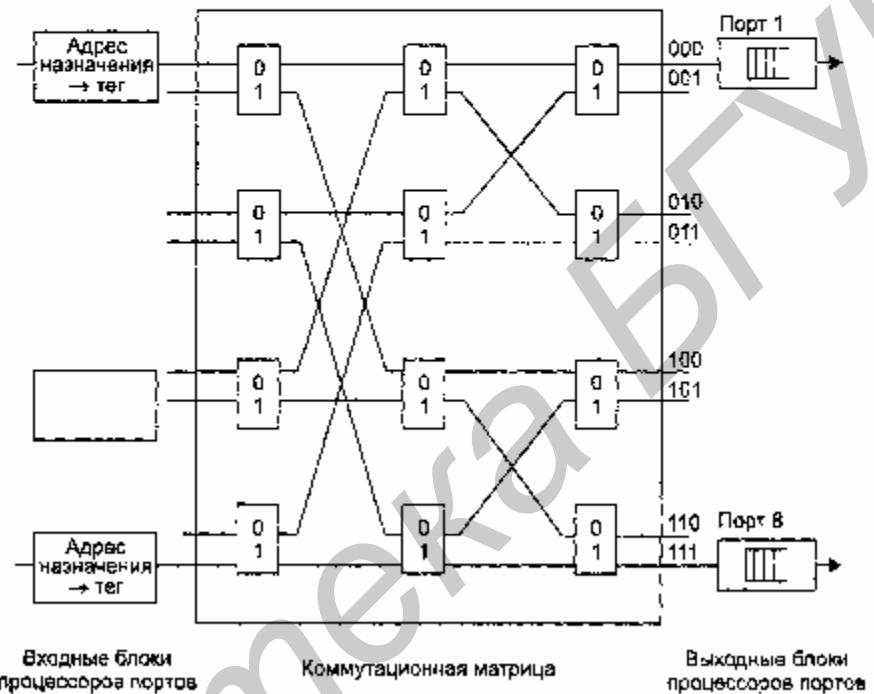


Рис. 9.32. Реализация коммутационной матрицы 8×8 с лампющими двоичными переключателями

Матрица может быть реализована и иначе, на основании комбинационных схем другого типа, но ее особенностью все равно остается технология коммутации физических каналов. Известным недостатком этой технологии является отсутствие буферизации данных внутри коммутационной матрицы — если составной канал невозможно построить из-за занятости выходного порта или промежуточного коммутационного элемента, то данные должны накапливаться в их источнике, в данном случае — во входном блоке порта, принявшего кадр. Основные достоинства таких матриц — высокая скорость коммутации и регулярная структура, которую удобно реализовывать в интегральных микросхемах. Зато после реализации матрицы NxN в составе БИС проявляется еще один ее недостаток — сложность наращивания числа коммутируемых портов.

Коммутаторы с общей шиной

В коммутаторах с общей шиной процессоры портов связывают высокоскорост-

ной шиной, используемой в режиме разделения времени.

Пример такой архитектуры приведен на рис. 9.33. Чтобы шина не блокировала работу коммутатора, ее производительность должна равняться по крайней мере сумме производительности всех портов коммутатора. Для модульных коммутаторов некоторые сочетания модулей с низкоскоростными портами могут приводить к неблокирующей работе, а установка модулей с высокоскоростными портами может приводить к тому, что блокирующим элементом станет, например, общая шина.

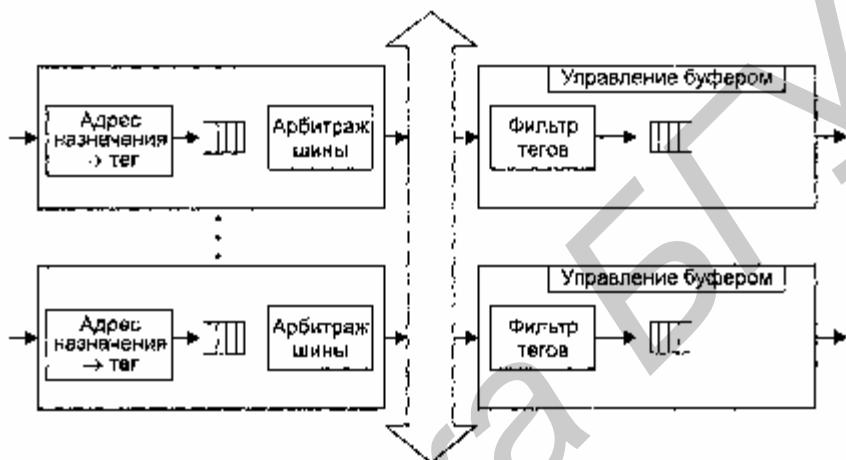


Рис. 9.33. Архитектура коммутатора с общей шиной

Кадр должен передаваться по шине небольшими частями, по несколько байтов, чтобы передача кадров между портами происходила в псевдопараллельном режиме, не внося задержек в передачу кадра в целом. Размер такой ячейки данных определяется производителем коммутатора. Некоторые производители, например LANNET или Ccntillion, выбрали в качестве порции данных, переносимых за одну операцию по шине, ячейку ATM с ее полем данных в 48 байт. Такой подход облегчает трансляцию протоколов локальных сетей в протокол ATM, если коммутатор поддерживает эти технологии.

Входной блок процессора помещает в ячейку, переносимую по шине, тег, в котором указывает номер порта назначения. Каждый выходной блок процессора порта содержит фильтр тегов, который выбирает теги, предназначенные данному порту.

Шина, так же как и коммутационная матрица, не может осуществлять промежуточную буферизацию, но так как данные кадра разбиваются на небольшие ячейки, то задержек с начальным ожиданием доступности выходного порта в такой схеме нет — здесь работает принцип коммутации пакетов, а не каналов.

Коммутаторы с разделяемой памятью

Третья базовая архитектура взаимодействия портов — двухвходовая разделяемая память. Пример такой архитектуры приведен на рис. 9.34.

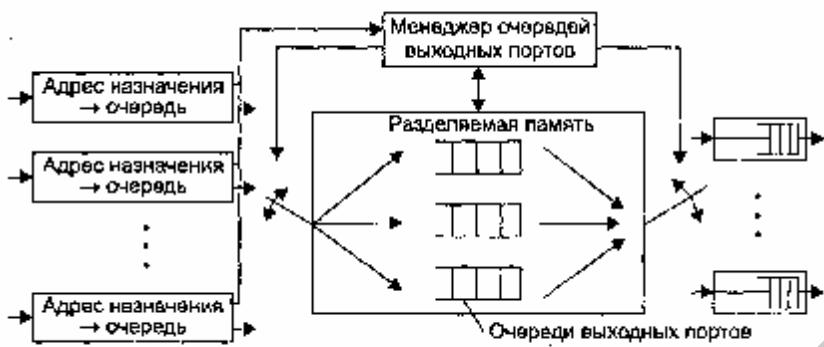


Рис. 9.34. Архитектура разделяемой памяти

Входные блоки процессоров портов соединяются с переключаемым входом разделяемой памяти, а выходные блоки этих же процессоров соединяются с переключаемым выходом этой памяти. Переключением входа и выхода разделяемой памяти управляет менеджер очередей выходных портов. В разделяемой памяти менеджер организует несколько очередей данных, по одной для каждого выходного порта. Входные блоки процессоров передают менеджеру портов запросы на запись данных в очередь того порта, который соответствует адресу назначения пакета. Менеджер по очереди подключает вход памяти к одному из входных блоков процессоров и тот переписывает часть данных кадра в очередь определенного выходного порта. По мере заполнения очередей менеджер производит также поочередное подключение выхода разделяемой памяти к выходным блокам процессоров портов, и данные из очереди переписываются в выходной буфер процессора.

Память должна быть достаточно быстродействующей для поддержания скорости переписи данных между N портами коммутатора. Применение общей буферной памяти, гибко распределяемой менеджером между отдельными портами, снижает требования к размеру буферной памяти процессора порта.

Комбинированные коммутаторы

У каждой из описанных архитектур есть свои преимущества и недостатки, поэтому часто в сложных коммутаторах эти архитектуры применяются в комбинации друг с другом. Пример такого комбинирования приведен на рис. 9.35.



Рис. 9.35. Комбинированное архитектурное решение коммутационной матрицы и общей шины

Коммутатор состоит из модулей с фиксированным количеством портов (2-12), выполненных на основе специализированной БИС, реализующей архитектуру коммутационной матрицы. Если порты, между которыми нужно передать кадр данных, принадлежат одному модулю, то передача кадра осуществляется процессорами модуля на основе имеющейся в модуле коммутационной матрицы. Если же порты принадлежат разным модулям, то процессоры общаются по общей шине. При такой архитектуре передача кадров внутри модуля будет происходить быстрее, чем при межмодульной передаче, так как коммутационная матрица — наиболее быстрый, хотя и наименее масштабируемый способ взаимодействия портов. Скорость внутренней шины коммутаторов может достигать нескольких гигабит в секунду, а у наиболее мощных моделей — до 20-30 Гбит/с.

Можно представить и другие способы комбинирования архитектур, например использование разделяемой памяти для взаимодействия модулей.

Конструктивное исполнение коммутаторов

В конструктивном отношении коммутаторы, как и все коммуникационные устройства, могут быть автономными, стековыми и выполненными на основе шасси.

Первый тип коммутаторов обычно предназначен для организации небольших рабочих групп.

Модульные коммутаторы на основе шасси чаще всего предназначены для применения на магистрали сети. Поэтому они выполняются на основе какой-либо комбинированной схемы, в которой взаимодействие модулей организуется по быстродействующей шине или же на основе быстрой разделяемой памяти большого объема. Модули такого коммутатора выполняются на основе технологии «hot swap», то есть допускают замену на ходу, без выключения коммутатора, так как центральное коммуникационное устройство сети не должно иметь перерывов в работе. Шасси обычно снабжается резервированными источниками питания и резервированными вентиляторами в тех же целях.

С технической точки зрения определенный интерес представляют стековые коммутаторы. Эти устройства могут работать автономно, так как выполнены в отдельном корпусе, но имеют специальные интерфейсы, которые позволяют их объединять в общую систему, работающую как единый коммутатор. Говорят, что в этом случае отдельные коммутаторы образуют стек.

Обычно такой специальный интерфейс представляет собой высокоскоростную шину, которая позволяет объединить отдельные корпуса подобно модулям в коммутаторе на основе шасси. Так как расстояния между корпусами больше, чем между модулями на шасси, скорость обмена по шине обычно ниже, чем у модульных коммутаторов. Не очень высокие скорости обмена между коммутаторами стека обусловлены также тем, что стековые коммутаторы обычно занимают промежуточное положение между коммутаторами с фиксированным ко-

личеством портов и коммутаторами на основе шасси. Стековые коммутаторы применяются для создания сетей рабочих групп и отделов, поэтому сверхвысокие скорости шин обмена им не очень нужны и не соответствуют их ценовому диапазону.

Структура стека коммутаторов, соединяемых по скоростным специальным портам, показана на рис. 9.36

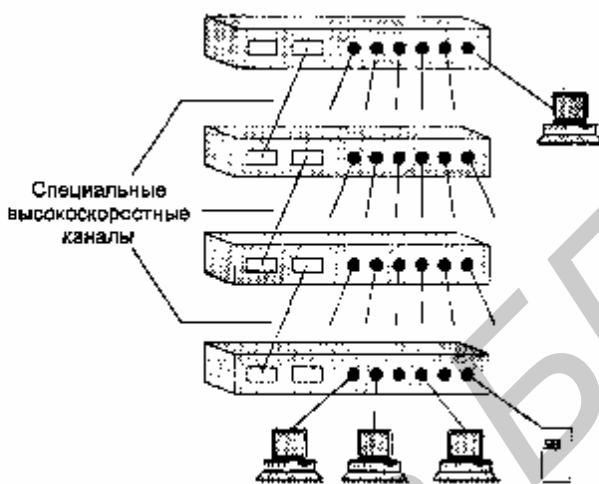


Рис. 9.36. Стек коммутаторов, объединяемых по высокоскоростным каналам

Характеристики производительности коммутаторов

Производительность — это то, что сетевые интеграторы и администраторы ждут от коммутатора в первую очередь.

Основными показателями коммутатора, характеризующими его производительность, являются:

- скорость фильтрации кадров;
- скорость продвижения кадров;
- пропускная способность;
- задержка передачи кадра.

Кроме того, существует несколько характеристик коммутатора, которые в наибольшей степени влияют на указанные характеристики производительности. К ним относятся:

- тип коммутации — «на лету» или с полной буферизацией;
- размер буфера (буферов) кадров;
- производительность внутренней шины;
- производительность процессора или процессоров;
- размер внутренней адресной таблицы.

Скорость фильтрации и скорость продвижения

Скорость фильтрации и продвижения кадров — две основные характеристики производительности коммутатора. Эти характеристики являются интегральны-

ми показателями, они не зависят от того, каким образом технически реализован коммутатор,

Скорость фильтрации (filtering) определяет скорость, с которой коммутатор выполняет перечисленные ниже этапы обработки кадров.

1. Прием кадра в свой буфер.

2. Просмотр адресной таблицы с целью нахождения порта для адреса назначения кадра.

3. Уничтожение кадра, так как его порт назначения и порт источника принадлежат одному логическому сегменту.

Скорость фильтрации практически у всех коммутаторов является неблокирующей — коммутатор успевает отбрасывать кадры в темпе их поступления.

Скорость продвижения (forwarding) определяет скорость, с которой коммутатор выполняет следующие этапы обработки кадров.

1. Прием кадра в свой буфер.

2. Просмотр адресной таблицы с целью нахождения порта для адреса назначения кадра.

3. Передача кадра в сеть через найденный по адресной таблице порт назначения.

Как скорость фильтрации, так и скорость продвижения измеряются обычно в кадрах в секунду. Если в характеристиках коммутатора не уточняется, для какого протокола и для какого размера кадра приведены значения скоростей фильтрации и продвижения, то по умолчанию считается, что эти показатели даются для протокола Ethernet и кадров минимального размера, то есть кадров длиной 64 байт (без преамбулы) с полем данных в 46 байт. Если скорости указаны для какого-либо определенного протокола, например Token Ring или FDDI, то они также даны для кадров минимальной длины этого протокола (например, кадров длины 29 байт для протокола FDDI). Применение в качестве основного показателя скорости работы коммутатора кадров минимальной длины объясняется тем, что такие кадры всегда создаются для коммутатора наиболее тяжелый режим работы по сравнению с кадрами другого формата при равной пропускной способности переносимых пользовательских данных. Поэтому при проведении тестирования коммутатора режим передачи кадров минимальной длины используется как наиболее сложный тест, который должен проверить способность коммутатора работать при наихудшем сочетании параметров трафика. Кроме того, для пакетов минимальной длины скорость фильтрации и продвижения максимальна, что имеет немаловажное значение при рекламе коммутатора.

Пропускная способность коммутатора измеряется количеством пользовательских данных, переданных в единицу времени через его порты (измеряется в мегабитах в секунду). Так как коммутатор работает на канальном уровне, для него пользовательскими данными являются те данные, которые переносятся в поле данных кадров протоколов канального уровня — Ethernet, Token Ring, FDDI и т. п. Максимальное значение пропускной способности коммутатора всегда достигается на кадрах максимальной длины, так как при этом доля накладных расходов на служебную информацию кадра гораздо ниже, чем для кадров мини-

мальной длины, а время выполнения коммутатором операций по обработке кадра, приходящееся на один байт пользовательской информации, существенно меньше. Поэтому коммутатор может быть блокирующим для кадров минимальной длины, но при этом иметь очень хорошие показатели пропускной способности.

Задержка передачи кадра измеряется как время, прошедшее с момента прихода первого байта кадра на входной порт коммутатора до момента появления этого байта на его выходном порту. Задержка складывается из времени, затрачиваемого на буферизацию байтов кадра, а также времени, затрачиваемого на обработку кадра коммутатором, — просмотр адресной таблицы, принятие решения о фильтрации или продвижении и получение доступа к среде выходного порта. Величина вносимой коммутатором задержки зависит от режима его работы. Если коммутация осуществляется «на лету», то задержки обычно невелики и составляют от 5 до 40 мкс, а при полной буферизации кадров — от 50 до 200 мкс (для кадров минимальной длины).

Коммутатор — это многопортовое устройство, поэтому для него принято все приведенные выше характеристики (кроме задержки передачи кадра) давать в двух вариантах. Первый вариант — суммарная производительность коммутатора при одновременной передаче трафика по всем его портам, второй вариант — производительность в расчете на один порт. Обычно производители коммутаторов указывают общую максимальную пропускную способность устройства.

Коммутация «на лету» и коммутация с буферизацией

На производительности коммутатора оказывается способ передачи пакетов — «на лету» или с буферизацией. Коммутаторы, передающие пакеты «на лету», вносят меньшие задержки передачи кадров на каждом промежуточном коммутаторе, поэтому общее уменьшение задержки доставки данных может быть значительным, что важно для мультимедийного трафика. Кроме того, выбранный способ коммутации оказывает влияние на возможности реализации некоторых полезных дополнительных функций, например трансляцию протоколов канального уровня.

В табл. 9.2 дается сравнение возможностей двух способов коммутации.

Таблица 9.2. Возможности коммутаторов при коммутации «на лету» и с полной буферизацией

Функция	«На лету»	С буферизацией
Защита от плохих кадров	Нет	Да
Поддержка разнородных сетей (Ethernet, Token Ring, FDDI, ATM)	Нет	Да
Задержка передачи пакетов	Низкая (5-40 мкс) при низкой нагрузке, средняя при высокой нагрузке	Средняя при любой нагрузке
Поддержка резервных связей	Нет	Да

Функция анализа трафика	Нет	Да
-------------------------	-----	----

Средняя величина задержки коммутаторов, работающих «на лету», при высокой нагрузке объясняется тем, что в этом случае выходной порт часто бывает занят приемом другого пакета, поэтому вновь поступивший пакет для данного порта все равно приходится буферизовать.

Коммутатор, работающий «на лету», может выполнять проверку некорректности передаваемых кадров, но не может изъять плохой кадр из сети, так как часть его байтов (и, как правило, большая часть) уже переданы в сеть.

Так как каждый способ имеет свои достоинства и недостатки, в тех моделях коммутаторов, которым не нужно транслировать протоколы, иногда применяется механизм адаптивной смены режима работы коммутатора. Основной режим такого коммутатора — коммутация «на лету», но коммутатор постоянно контролирует трафик и при превышении интенсивности появления плохих кадров некоторого порога переходит на режим полной буферизации. Затем коммутатор может вернуться к коммутации «на лету».

Размер адресной таблицы

Максимальная емкость адресной таблицы определяет предельное количество MAC-адресов, с которыми может одновременно оперировать коммутатор. Так как коммутаторы чаще всего используют для выполнения операций каждого порта выделенный процессорный блок со своей памятью для хранения экземпляра адресной таблицы, то размер адресной таблицы для коммутаторов обычно приводится в расчете на один порт. Экземпляры адресной таблицы разных процессорных модулей не обязательно содержат одну и ту же адресную информацию — скорее всего, повторяющихся адресов будет не так много, если только распределение трафика каждого порта между остальными портами не полностью равновероятно. Каждый порт хранит только те наборы адресов, с которыми он работал в последнее время.

Значение максимального числа MAC-адресов, которое может запомнить процессор порта, зависит от области применения коммутатора. Коммутаторы рабочих групп обычно поддерживают всего несколько адресов на порт, так как они предназначены для образования микросегментов. Коммутаторы отделов должны поддерживать несколько сотен адресов, а коммутаторы магистралей сетей — до нескольких тысяч, обычно 4000-8000 адресов.

Недостаточная емкость адресной таблицы может служить причиной замедления работы коммутатора и засорения сети избыточным трафиком. Если адресная таблица процессора порта полностью заполнена, а он встречает новый адрес источника о поступившем пакете, процессор должен вытеснить из таблицы какой-либо старый адрес и поместить на его место новый. Эта операция сама по себе отнимет у процессора часть времени, но главные потери производительности будут наблюдаться при поступлении кадра с адресом назначения, который пришлось удалить из адресной таблицы. Так как адрес назначения кадра неиз-

вестен, то коммутатор должен передать этот кадр на все остальные порты. Эта операция создаст лишнюю работу для многих процессоров портов, кроме того, копии этого кадра попадут и на те сегменты сети, где они совсем не обязательны.

Некоторые производители коммутаторов решают эту проблему за счет изменения алгоритма обработки кадров с неизвестным адресом назначения. Один из портов коммутатора конфигурируется как магистральный порт, на который по умолчанию передаются все кадры с неизвестным адресом. В маршрутизаторах такой прием применяется давно, позволяя сократить размеры адресных таблиц в сетях, организованных по иерархическому принципу.

Передача кадра на магистральный порт производится в расчете на то, что этот порт подключен к вышестоящему коммутатору (при иерархическом соединении коммутаторов в крупной сети), который имеет достаточную емкость адресной таблицы и знает, куда нужно передать любой кадр.

Объем буфера кадров

Внутренняя буферная память коммутатора нужна для временного хранения кадров данных в тех случаях, когда их невозможно немедленно передать на выходной порт. Буфер предназначен для сглаживания кратковременных пульсаций трафика. Ведь даже если трафик хорошо сбалансирован и производительность процессоров портов, а также других обрабатывающих элементов коммутатора достаточна для передачи средних значений трафика, это не гарантирует, что их производительности хватит при пиковых значениях нагрузок. Например, трафик может в течение нескольких десятков миллисекунд поступать одновременно на все входы коммутатора, не давая ему возможности передавать принимаемые кадры на выходные порты.

Для предотвращения потерь кадров при кратковременном многократном превышении среднего значения интенсивности трафика (а для локальных сетей часто встречаются значения коэффициента пульсации трафика в диапазоне 50-100) единственным средством служит буфер большого объема. Как и в случае адресных таблиц, каждый процессорный модуль порта обычно имеет свою буферную память для хранения кадров. Чем больше объем этой памяти, тем менее вероятны потери кадров при перегрузках, хотя при несбалансированности средних значений трафика буфер все равно рано или поздно переполнится. Обычно коммутаторы, предназначенные для работы в ответственных частях сети, имеют буферную память, в несколько десятков или сотен килобайт на порт. Хорошо, когда эту буферную память можно перераспределять между несколькими портами, так как одновременные перегрузки по нескольким портам маловероятны. Дополнительным средством защиты может служить общий для всех портов буфер в модуле управления коммутатором. Такой буфер обычно имеет объем в несколько мегабайт.

Трансляция протоколов канального уровня

Коммутаторы могут выполнять трансляцию одного протокола канального уровня в другой, например Ethernet в FDDI, Fast Ethernet в Token Ring и т. п. При этом они работают по тем же алгоритмам, что и транслирующие мосты, то есть в соответствии со спецификациями IEEE 802.1H и RFC 1042, определяющими правила преобразования полей кадров разных протоколов.

Трансляцию протоколов локальных сетей облегчает тот факт, что наиболее сложную работу, которую при объединении гетерогенных сетей часто выполняют маршрутизаторы и шлюзы, а именно работу по трансляции адресной информации, в данном случае выполнять не нужно. Все конечные узлы локальных сетей имеют уникальные адреса одного и того же формата независимо от поддерживаемого протокола. Поэтому адрес сетевого адаптера Ethernet понятен сетевому адаптеру FDDI, и они оба могут использовать эти адреса в полях своих кадров, не задумываясь о том, что узел, с которым они взаимодействуют, принадлежит сети, работающей по другой технологии.

Поэтому при согласовании протоколов локальных сетей коммутаторы не строят таблиц соответствия адресов узлов, а переносят адреса назначения и источника из кадра одного протокола в кадр другого.

Кроме изменения порядка следования битов при передаче байтов адреса, трансляция протокола Ethernet (и Fast Ethernet, который использует формат кадров Ethernet) в протоколы FDDI и Token Ring включает выполнение перечисленных ниже (возможно, не всех) операций.

- Вычисление длины поля данных кадра и помещение этого значения в поле Length при передаче кадра из сети FDDI или Token Ring в сеть Ethernet 802.3 (в кадрах FDDI и Token Ring поле длины отсутствует).
- Заполнение полей статуса кадра при передаче кадров из сети FDDI или Token Ring в сеть Ethernet. Кадры FDDI и Token Ring имеют два бита, устанавливаемые станцией, которой предназначался кадр, — бит распознавания адреса A и бит копирования кадра С. При получении кадра станция должна установить эти два бита, чтобы кадр, вернувшийся по кольцу к сгенерировавшей его станции, принес данные обратной связи. При передаче коммутатором кадра в другую сеть нет стандартных правил для установки битов А и С в кадре, который возвращается по кольцу к станции-источнику. Поэтому производители коммутаторов решают эту проблему по своему усмотрению.
- Отбрасывание кадров, передаваемых из сетей FDDI или Token Ring в сеть Ethernet с размером поля данных большим, чем 1500 байт, так как это максимально возможное значение поля данных для сетей Ethernet. В дальнейшем, не дождавшись ответа от станции назначения из сети Ethernet, протокол верхнего уровня станции из сети FDDI, возможно, уменьшит размер передаваемых в одном кадре данных, и тогда коммутатор сможет передавать кадры между этими

станциями. Другим вариантом решения проблемы является поддержка коммутатором IP-фрагментации, но это требует, во-первых, реализации в коммутаторе протокола сетевого уровня, а во-вторых, поддержки протокола IP взаимодействующими узлами транслируемых сетей.

- Заполнение поля Type (тип протокола в поле данных) кадра Ethernet II при приходе кадров из сетей, поддерживающих кадры FDDI или Token Ring, в которых это поле отсутствует, зато имеются поля DSAP и SSAP, выполняющие то же назначение, но с другими кодами для обозначения протоколов. Для упрощения трансляции спецификация RFC 1042 предлагает всегда использовать в сетях FDDI и Token Ring кадры с заголовками LLC/SNAP, которые имеют то же поле Type и с теми же значениями, что и кадры Ethernet II. При преобразовании кадров значение из поля Type заголовка LLC/SNAP переносится в поле Type кадра Ethernet II, и наоборот. Если и сети Ethernet имеются форматы кадров, отличные от Ethernet II, то они также должны иметь заголовок LLC/SNAP.
- Пересчет контрольной суммы кадра в соответствии со сформированными значениями служебных полей кадра.

Возможности коммутаторов по фильтрации трафика

Многие коммутаторы позволяют администраторам задавать дополнительные условия фильтрации кадров наряду со стандартными условиями их фильтрации в соответствии с информацией адресной таблицы. Пользовательские фильтры предназначены для создания дополнительных барьеров на пути кадров, которые ограничивают доступ определенных групп пользователей к определенным службам сети.

Наиболее простыми являются пользовательские фильтры на основе MAC-адресов станций. Так как MAC-адреса — это та информация, с которой работает коммутатор, то он позволяет создавать такие фильтры в удобной для администратора форме, возможно, проставляя некоторые условия в дополнительном поле адресной таблицы, подобно тем, которые были указаны в адресной таблице моста System 3000 на рис. 9.20 (например, отбрасывать кадры с определенным адресом). При этом пользователю, работающему на компьютере с данным MAC-адресом, полностью запрещается доступ к ресурсам другого сегмента сети.

Часто администратору требуется задать более тонкие условия фильтрации, например запретить некоторому пользователю печатать свои документы на определенном сервере печати NetWare чужого сегмента, а остальные ресурсы этого сегмента сделать доступными. Для реализации подобного фильтра нужно запретить передачу кадров с определенным MAC-адресом, в которых вложены пакеты IPX, если в поле «номер пакета» этих кадров указано значение, соответствующее службе печати NetWare. Коммутаторы не анализируют протоколы верхних уровней, такие как IPX, поэтому администратору приходится для задания условий фильтрации вручную определять поле, по значению которого нужно осуществлять фильтрацию, в виде пары «смещение—размер» относи-

тельно начала поля данных кадра канального уровня, а затем еще указать в шестнадцатеричном формате значение этого поля для службы печати.

Обычно условия фильтрации записываются в виде булевых выражений, формируемых с помощью логических операторов AND и OR.

Наложение дополнительных условий фильтрации может снизить производительность коммутатора, так как вычисление булевых выражений требует проведения дополнительных вычислений процессорами портов.

Приоритетная обработка кадров

Построение сетей на основе коммутаторов позволяет задействовать приоритезацию трафика для повышения качества обслуживания пользовательского трафика, причем независимо от технологии сети. Эта новая возможность (по сравнению с сетями, построенными целиком на концентраторах) является следствием того, что коммутаторы буферизуют кадры перед их отправкой на другой порт. Коммутатор обычно ведет для каждого входного и выходного порта не одну, а несколько очередей, причем каждая очередь имеет собственный приоритет обработки. При этом коммутатор может быть сконфигурирован, например, так, чтобы передавать один низкоприоритетный пакет на каждые 10 высокоприоритетных пакетов.

Поддержка приоритетной обработки может особенно пригодиться для приложений, предъявляющих различные требования к допустимым задержкам кадров и пропускной способности сети для потока кадров.

Приоритезация трафика коммутаторами сегодня является одним из основных механизмов обеспечения качества транспортного обслуживания в локальных сетях. К каким уровням задержек приводит приписывание того или иного уровня приоритета кадру, какую пропускную способность обеспечивает приоритет потоку кадров — схема приоритезации не говорит. Выяснить последствия ее применения можно только путем проведения натурных экспериментов или имитационного моделирования. Ясно только одно — более приоритетные кадры будут обрабатываться раньше менее приоритетных, поэтому все показатели качества обслуживания у них будут выше, чем у менее приоритетных. Остается вопрос — насколько? Гарантии качества обслуживания дают другие схемы, которые основаны на предварительном резервировании качества обслуживания. Например, такие схемы используются в технологиях глобальных сетей frame relay и ATM, а также в протоколе RSVP для сетей TCP/IP. Однако для коммутаторов такого рода протоколов нет, так что гарантий качества обслуживания они пока дать не могут.

Основным вопросом при приоритетной обработке кадров коммутаторами является вопрос назначения кадру приоритета. Так как не все протоколы канального уровня поддерживают поле приоритета кадра, например у кадров Ethernet оно отсутствует, то коммутатор должен использовать какой-либо дополнительный механизм для связывания кадра с его приоритетом. Наиболее распространенный способ — приписывание приоритета портам коммутатора. При этом способе коммутатор помещает кадр в очередь кадров соответствующего приоритета

та в зависимости от того, через какой порт поступил кадр в коммутатор. Способ несложный, но недостаточно гибкий — если к порту коммутатора подключен не отдельный узел, а сегмент, то иск узлы сегмента получают одинаковый приоритет.

Многие компании, выпускающие коммутаторы, реализовали в них ту или иную схему приоритетной обработки кадров. Примером фирменного подхода к назначению приоритетов на основе портов является технология PACE компании 3Com.

Более гибким является назначение приоритетов кадрам в соответствии с достаточно новым стандартом IEEE 802.1p. Этот стандарт разрабатывался совместно со стандартом 802.1Q, который рассматривается в разделе «Виртуальные локальные сети» главы 10. В обоих стандартах предусмотрен общий дополнительный заголовок для кадров Ethernet, состоящий из двух байтов. В этом дополнительном заголовке, который вставляется перед полем данных кадра, три бита используются для указания приоритета кадра. Существует протокол, по которому конечный узел может запросить у коммутатора один из восьми уровней приоритета кадра. Если сетевой адаптер не поддерживает стандарт 802.1p, то коммутатор может назначать приоритеты кадрам на основе порта поступления кадра. Такие помеченные кадры будут обслуживаться в соответствии с их приоритетом всеми коммутаторами сети, а не только тем коммутатором, который непосредственно принял кадр от конечного узла. При передаче кадра сетевому адаптеру, не поддерживающему стандарт 802.1p, дополнительный заголовок должен быть удален.

Более подробно вопросы поддержки качества обслуживания, QoS, рассматриваются в части V, посвященной глобальным сетям. Это связано с тем, что пропускная способность глобальных сетей всегда ценилась дороже, так как территориальные каналы долгое время были более низкоскоростными, чем локальные, к тому же пользование ими чаще всего было построено на коммерческой основе, когда за каждый дополнительный килобит в секунду средней скорости приходилось платить. Л раз платить, то нужно точно знать, за что именно, для чего и были разработаны достаточно эффективные процедуры обеспечения качества обслуживания. В локальных сетях с развитием приложений, чувствительных к задержкам, например приложений IP-телефонии, также повысились требования к качеству обслуживания. Методы обеспечения QoS одинаковы для глобальных и локальных сетей, тем не менее авторы посчитали, что более естественно отнести их к той части, в которой рассматриваются сети, «породившие» эти методы, а именно глобальные сети.

Выводы

-
- Коммутаторы связывают процессоры портов по трем основным схемам — коммутационной матрицы, общей шины и разделяемой памяти. В коммутаторах с фиксированным количеством портов обычно используется коммутационная матрица, а в модульных коммутаторах — сочетание коммутационной мат-

рицы в отдельных модулях с общей шиной и разделяемой памятью для связи модулей.

□ Для поддержания неблокирующего режима работы коммутатора общая шина или разделяемая память должны обладать производительностью, превышающей сумму производительностей всех портов максимально высокоскоростного набора модулей, которые устанавливаются в шасси.

□ Основными характеристиками производительности коммутатора являются; скорость фильтрации кадров, скорость продвижения кадров, общая пропускная способность по всем портам в мегабитах в секунду, задержка передачи кадра.

□ На характеристики производительности коммутатора влияют: тип коммутации — «на лету» или с полной буферизацией, размер адресной таблицы, размер буфера кадров.

а Коммутаторы умеют фильтровать передаваемый трафик по различным критериям, учитывающим адреса получателя и отправителя, а также значения произвольных полей. Однако способ задания фильтров на канальном уровне достаточно сложный, требующий от администратора хорошего знания протоколов и кропотливой работы по определению местоположения нужного признака внутри кадра.

□ Для приоритетной обработки трафика в коммутаторах организуется несколько очередей. Выбор очереди происходит, исходя из различных условий, в том числе и на основании значения трехбитного поля приоритета, добавляемого в соответствии со стандартом 802.1p.

ГЛАВА 10 Дополнительные функции мостов и коммутаторов

Производительность и надежность каналов связи — важнейшие характеристики любой сети, самым непосредственным образом влияющие на ее полноценную работу. Какие же средства повышения надежности и производительности каналов связи имеются в распоряжении сетевого интегратора или администратора? Этот арсенал достаточно обширен и зависит от типа сети. Если говорить о локальных сетях, то нужной производительности можно добиться в первую очередь за счет рационального выбора для каждого канала той или иной технологии с необходимой пропускной способностью. Диапазон вариантов достаточно широк. Семейство Ethernet, которое сегодня практически вытеснило все остальные технологии локальных сетей, предлагает шкалу скоростей от 10 Мбит/с (классический стандарт Ethernet) и 100 Мбит/с (Fast Ethernet) до 1000 Мбит/с (Gigabit Ethernet) и даже 10 Гбит/с.

Однако десятикратное увеличение скорости при переходе от одного уровня Ethernet к другому не всегда удобно. Например, если в установленных в сети коммутаторах отсутствует возможность добавления модуля с портом Gigabit Ethernet, то повышение скорости на некоторых каналах до 1000 Мбит/с потребует их полной замены. В то же время вполне возможно, что у таких коммутаторов имеются свободные порты East Ethernet, поэтому скорость передачи данных можно было бы повысить, включив в работу не занятые ресурсы. При этом в сети появляются альтернативные пути следования трафика.

Таким образом, повышение уровня связности сети, то есть создание альтернативных маршрутов, — еще одно средство (причем достаточно мощное и масштабируемое) повышения эксплуатационных характеристик сети. Неизбежные дополнительные затраты могут оказаться меньше, чем при повышении производительности «в лоб», за счет применения более скоростной технологии канала, а для повышения надежности введение избыточных соединений чаще всего становится единственным практически осуществимым способом достижения цели.

Алгоритм Spanning Tree

Альтернативные связи

Альтернативные связи в сетях обычно используются двумя способами (рис. 10.1):

- в режиме резервирования, когда одно из них функционирует, а остальные находятся в «горячем» резерве для замены отказавшей связи (например, канал 2 зарезервирован для канала 7, а канал 7 — для канала 6);
- в режиме баланса нагрузки; при этом данные передаются параллельно по

всем альтернативным соединениям (например, для передачи данных от устройства S2 к устройству S3 можно воспользоваться путем через канал 10 или транзитным альтернативным путем: канал 1 — устройство S1 — канал 3).

В первом случае улучшается только одна характеристика сети — надежность, а производительность остается неизменной. Второй способ позволяет повысить как производительность, так и надежность. Когда одно из альтернативных соединений оказывается неработоспособным, связь между узлами не разрывается, а просто уменьшается обеспечиваемая производительность — на величину пропускной способности отказавшего канала связи. Обычно такие системы называют «живучими» или отказоустойчивыми: их работоспособность снижается постепенно, по мере отказа составляющих их компонентов.

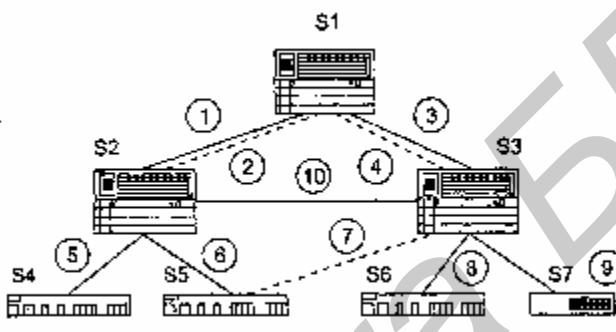


Рис. 10.1. Агрегирование физических каналов

Рассмотрим сначала первый подход. В тех локальных сетях, где технологии и оборудование реализуют функции только первого и второго уровней модели ISO/OSI, проблема использования альтернативных путей имеет свою специфику: базовые протоколы поддерживают только древовидные, то есть не содержащие замкнутых контуров топологии связей. А это означает, что для организации альтернативных каналов требуются особые протоколы и технологии, выходящие за рамки базовых, к которым относится Ethernet или алгоритм прозрачного моста (Transparent Bridge), описанный в стандарте 802.1D и применяемый и современных коммутаторах второго уровня.

Для автоматического перевода в резервное состояние всех альтернативных связей, не вписывающихся в топологию дерева, в локальных сетях используются алгоритм покрывающего дерева (Spanning Tree Algorithm, STA) и реализующий его протокол (Spanning Tree Protocol, STP).

Алгоритм Spanning Tree был разработан достаточно давно — в 1983 году сотрудниками компании DEC Радией Перлман. Он должен был решить проблему прозрачных мостов, популярность которых в то время быстро росла вместе с распространением сетей Ethernet. Прозрачные мосты по причине своей простоты не могут правильно работать в сетях с петлями — широковещательные пакеты при такой топологии размножаются и зацикливаются, да и таблица продвижения пакетов не всегда может быть построена корректно. Вместе с тем производители оборудования для локальных сетей не хотели усложнять свои устройства, пытаясь сохранить низкую стоимость и простоту установки и экс-

плутации сети за счет отказа от поддержки функций сетевого уровня. Найденное Радией Перлман решением позволило достичь желанного компромисса. Прозрачные мосты с поддержкой алгоритма STA остались простыми самообучающимися устройствами, не требующими предварительного конфигурирования. При этом появилась возможность без применения маршрутизаторов строить крупные локальные сети, обладающие высокой надежностью за счет наличия избыточных связей в сети с произвольной топологией, в том числе и петлевидной. Алгоритм Spanning Tree был признан IEEE удачным и включен в ту же спецификацию 802.1D, где описывается и сам алгоритм прозрачного моста.

Алгоритм STA по-прежнему остается очень важной составляющей локальных сетей. Хотя мосты, для которых был разработан алгоритм, сегодня уже относятся к практически «вымершему» виду коммуникационных устройств, STA широко применяется в наиболее массовых устройствах современных локальных сетей — коммутаторах второго уровня. Это неудивительно, поскольку последние функционально идентичны прозрачным мостам, а вот производительность у них существенно выше за счет параллельного продвижения пакетов процессорами портов. Поэтому производители оборудования, как правило, реализуют алгоритм STA в коммутаторах, предлагаемых для тех участков сети, к которым предъявляются повышенные требования по надежности, — в магистральных коммутаторах, а также в коммутаторах отделов и крупных рабочих групп.

Далее при описании алгоритма будет использоваться термин «коммутатор», хотя замена его «мостом» только придала бы тексту некоторый налет архаичности, но никоим образом не изменила бы суть алгоритма.

Сеть и ее граф

Алгоритм STA формализует сеть (рис. 10.2, а) в виде графа (рис. 10.2, б), вершинами которого являются коммутаторы и сегменты сети. Сегмент — связная часть сети, не содержащая коммутаторов (и маршрутизаторов). Он может быть разделяемым (во время создания алгоритма STA это был единственный тип сегмента) и включать устройства физического уровня — повторители и концентраторы, существование которых коммутатор, будучи устройством канального уровня, не замечает. Сегодня сегмент часто представляет собой дуплексный канал «точка-точка» между смежными портами двух коммутаторов.

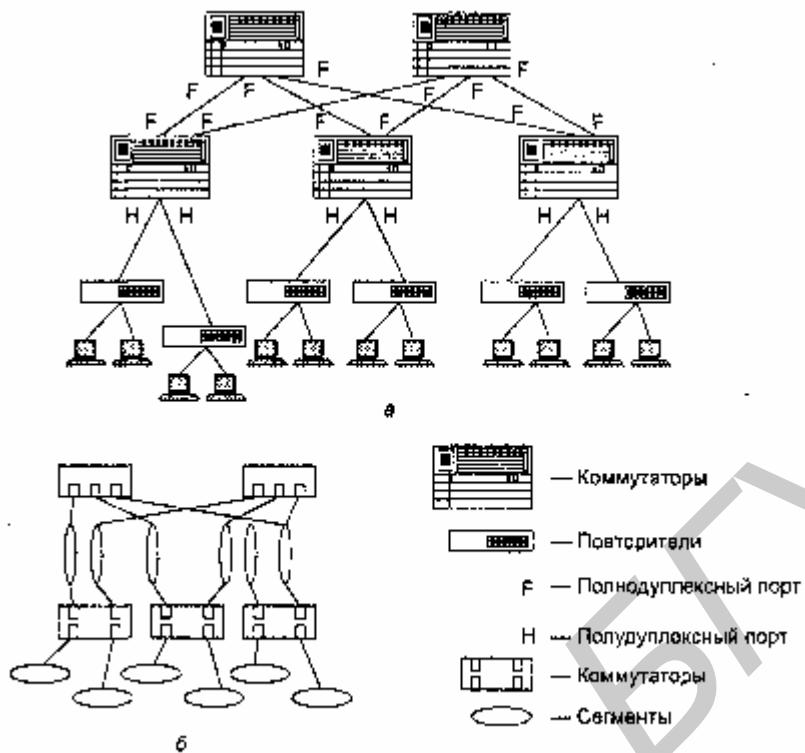


Рис. 10.2. Формализованное представление сети по STA

Алгоритм Spanning Tree обеспечивает поиск древовидной топологии связей с единственным путем от каждого коммутатора и от каждого сегмента до некоторого выделенного коммутатора (корня дерева) при минимально возможном расстоянии. Единственность пути гарантирует отсутствие петель, а минимальность расстояния — рациональность маршрутов следования трафика от периферии сети к ее магистрали, роль которой выполняет корневой коммутатор.

В качестве расстояния в STA используется метрика, традиционная для протоколов маршрутизации, — величина, обратно пропорциональная пропускной способности сегмента. Более формально, метрика — это так называемое «условное время сегмента», которое рассчитывается как время передачи одного бита информации и измеряется в 10-наносекундных единицах. Так, для сегмента Ethernet 10 Мбит/с условное время равно 10 условным единицам, Ethernet 100 Мбит/с — 1, а для сегмента Token Ring 16 Мбит/с - 6,25. Учитывая, что сети становятся все более и более скоростными, существует так называемая пересмотренная версия шкалы условных единиц; 10 Мбит/с - 100, 100 Мбит/с — 19, 1 Гбит/с — 4, 10 Гбит/с - 2.

В качестве корневого коммутатора выбирается коммутатор с наименьшим значением идентификатора. Идентификатор коммутатора — это число длиной восемь байт, шесть младших байтов которого составляет MAC-адрес его блока управления, отрабатывающего алгоритм STA (напомним, что портам коммутаторов и мостов для выполнения своей основной функции MAC-адресов не требуется), а два старших байта конфигурируются вручную, что позволяет администратору сети влиять на процесс выбора корневого коммутатора.

Идентификатором порта служит число длиной в два байта. Младший байт со-

держит порядковый номер данного порта в коммутаторе, а значение старшего байта задается администратором. На практике для целей управления порты коммутатора часто имеют собственные MAC-адреса, младшая часть которых используется в качестве идентификатора порта.

На рис. 10.3 приведен пример сети, для которой коммутаторы уже нашли покрывающее дерево. Предположим, что идентификаторы коммутаторов совпадают с их порядковыми номерами. При равных значениях метрики для разрешения неоднозначности к процедуре выбора минимального расстояния привлекаются значения идентификаторов коммутаторов и портов. Предпочтение отдается коммутатору с наименьшим идентификатором. Например, для сегмента 3 существует два равноценных в отношении метрики пути к корневому коммутатору 1 — через коммутатор 3 и через коммутатор 4. Выбранный путь проходит через коммутатор, с меньшим значением идентификатора, а именно 3 (номера портов внутри коммутатора в данном случае совпадают, но при сравнении сначала принимается во внимание идентификатор коммутатора, а потом уже номер порта). Все порты, не вошедшие в покрывающее дерево, блокируются для пользовательского трафика (на рисунке они перечеркнуты).

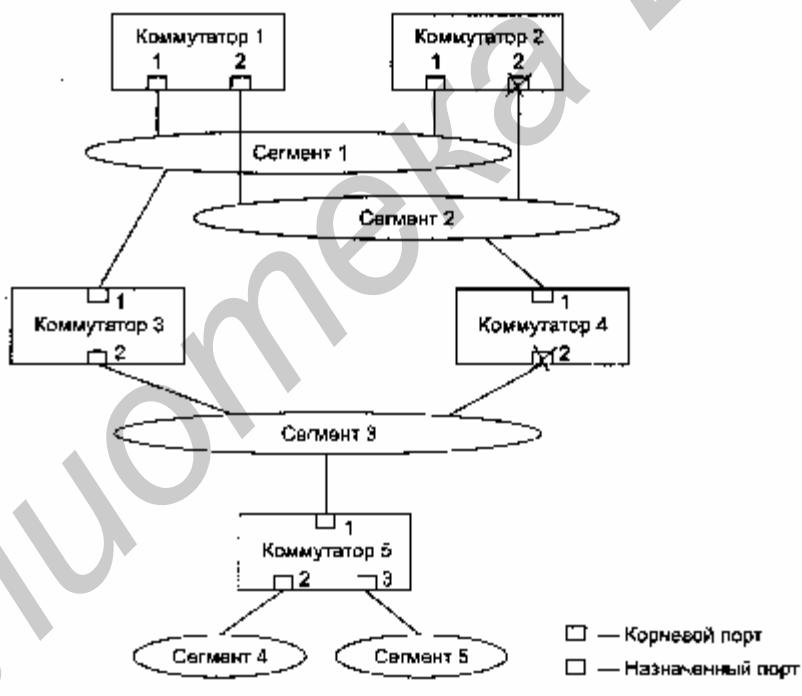


Рис. 10.3. Пример покрывающего дерева STA

Отметим, что выбранная по алгоритму STA древовидная топология в общем случае не оптимальна для всех путей передачи трафика. Так, в описываемом примере при передаче пакетов из сегмента 3 в сегмент 2 трафик проходит путь: коммутатор 3 — сегмент 1 — коммутатор 1 — сегмент 2. Общая метрика этого пути — 30.

Если бы порт 2 коммутатора 4 не был заблокирован, то путь мог быть короче — через коммутатор 4. При этом метрика пути была бы равна 20 — лучше, чем в предыдущем случае. Такой вариант возможен при выборе кратчайшего пути к

корневому коммутатору для сегмента 4 через коммутатор 4, а не 3, например, за счет соответствующего назначения старших частей идентификаторов коммутаторов. Однако при таком варианте путь из сегмента 4 в сегмент 1 уже не окажется оптимальным.

Дело в том, что алгоритм STA для упрощения работы коммутатора создает единое дерево, не зависящее от адресов назначения пакетов, что принципиально отличается от подхода, принятого в протоколах маршрутизации при выборе маршрутов для трафика сетевого уровня. В последнем случае кратчайший маршрут выбирается для каждого сетевого адреса назначения, так что трафик всегда идет по оптимальному для принятой метрики маршруту.

Построение дерева в три приема

Для автоматического определения начальной конфигурации дерева все коммутаторы сети после их инициализации начинают периодически обмениваться специальными пакетами, называемыми протокольными блоками данных моста (Bridge Protocol Data Unit, BPDU). Пакеты BPDU переносят данные об идентификаторах коммутаторов и портов, а также о расстоянии до корневого коммутатора. Интервал генерации пакетов BPDU, называемый в алгоритме интервалом hello, настраивается администратором и обычно составляет от 1 до 4 с.

Алгоритм STA определяет активную конфигурацию сети за три этапа.

Сначала в сети устанавливается корневой коммутатор (Root Switch), от которого строится дерево. Если администратор не вмешается в этот процесс, корневой коммутатор будет выбран достаточно случайным образом — им станет устройство с минимальным MAC-адресом блока управления. Очевидно, что такой выбор может оказаться далеко не рациональным. Примером мог бы служить выбор корневого коммутатора 5 в сети, показанной на рис. 10.3. При этом значительная часть трафика проходила бы через большое количество транзитных сегментов и коммутаторов. Поэтому ни в коем случае нельзя пускать данный процесс на самотек — лучше в него вмешаться и назначить корневой коммутатор осознанно (за счет соответствующего конфигурирования старших байтов идентификаторов коммутатора), чтобы выбранный коммутатор действительно занимал центральное место в соединениях сегментов.

После выбора корневого коммутатора все остальные коммутаторы сети перестают генерировать пакеты BPDU и продолжают только ретранслировать пакеты BPDU корневого моста — для того чтобы каждый коммутатор определил минимальные расстояния от всех своих портов до корневого коммутатора. Эти данные используются на следующих этапах работы STA.

Выбор корневого порта для каждого коммутатора — второй этап создания покрывающего дерева. Корневой порт (Root Port) коммутатора должен иметь кратчайшее расстояние до корневого коммутатора (точнее, до любого из портов корневого коммутатора). Расстояние определяется по пакетам BPDU, ретранслируемым коммутатором.

И, наконец, на третьем этапе из всех портов всех коммутаторов каждого сег-

мента сети выбирается так называемый назначенный порт (Designated Port) — порт с минимальным расстоянием до корневого коммутатора. Коммутатор, которому принадлежит назначенный порт для данного сегмента, объявляется назначенным коммутатором (Designated Switch) сегмента.

Все остальные порты, кроме корневых и назначенных, переводятся в заблокированное состояние. Математически доказано, что при таком выборе активных портов в сети исключаются петли и оставшиеся связи образуют покрывающее дерево (если оно вообще может быть построено при существующих связях в сети).

Пакеты BPDU и выбор корневого коммутатора

Пакеты BPDU помещаются в поле данных кадров канального уровня; например кадров Ethernet или FDDI. Желательно, чтобы все коммутаторы поддерживали общий групповой адрес, с помощью которого кадры, содержащие пакеты BPDU, могли одновременно передаваться всем коммутаторам сети. Иначе пакеты BPDU рассылаются широковещательно.

Пакет BPDU имеет следующие поля.

- Идентификатор версии протокола STA (2 байта).
- Тип BPDU (1 байт). Пакеты BPDU бывают двух типов — конфигурационный пакет BPDU, то есть заявка на статус корневого коммутатора, на основании которой происходит определение активной конфигурации, и пакет BPDU, служащий для уведомления о реконфигурации. Последний посыпается коммутатором при обнаружении им события, наступление которого требует проведения реконфигурирования. Это может быть, например, отказ соединения, отказ порта, изменение приоритетов коммутатора или портов,
- Флаги (1 байт). Один бит содержит флаг изменения конфигурации, второй — флаг подтверждения изменения конфигурации.
- Идентификатор корневого коммутатора (8 байт).
- Расстояние до корня (2 байта).
- Идентификатор данного коммутатора (8 байт).
- Идентификатор порта (2 байта).
- Время жизни сообщения (2 байта). Оно измеряется в единицах по 0,5 с и служит для выявления устаревших сообщений.
- Максимальное время жизни сообщения (2 байта).
- Интервал hello (2 байта), через который посыпаются пакеты BPDU.
- Задержка смены состояний (2 байта).

У пакета BPDU с уведомлением о реконфигурации отсутствуют вес поля, кроме двух первых.

После инициализации каждый коммутатор по умолчанию считает себя корневым. Поэтому он начинает периодически с интервалом hello генерировать через все свои порты сообщения BPDU конфигурационного типа. В них он указывает свой идентификатор в качестве идентификатора корневого коммутатора (и в качестве идентификатора данного коммутатора также), расстояние до корня ус-

становливается равным 0, а в поле идентификатора порта помещается идентификатор того порта, через который передается данный пакет BPDU. Как только коммутатор получает пакет BPDU, в котором значение идентификатора корневого коммутатора меньше его собственного, он перестает генерировать пакеты BPDU и переходит в режим ретрансляции пакетов BPDU нового претендента на звание корневого коммутатора. На рис. 10.3 наименьшее значение идентификатора имеет коммутатор 1, поэтому именно он становится корневым.

Выбор корневых и назначенных портов

При ретрансляции пакетов каждый коммутатор увеличивает расстояние до корня, указанное в полученном пакете BPDU, на условное время того сегмента, из которого принят данный пакет. Тем самым в пакете BPDU по мере прохождения через коммутаторы наращивается расстояние до корневого коммутатора. Например, если считать, что все сегменты в рассматриваемом примере являются сегментами Ethernet на 10 Мбит/с, то коммутатор 2, приняв из сегмента 1 сообщение BPDU с расстоянием, равным 0, увеличивает его на 10 единиц.

Ретранслируя пакеты, каждый коммутатор для каждого своего порта запоминает минимальное расстояние до корня, встретившееся во всех принятых этим портом пакетах BPDU. По завершении (по времени) процедуры определения конфигурации покрывающего дерева каждый коммутатор находит свой корневой порт (с минимальным расстоянием до корня).

Так, в примере, приведенном на рис. 10.3, коммутатор 3 выбирает порт 1 в качестве корневого, так как для него минимальное расстояние до корня равно 10 (пакет BPDU с таким расстоянием принят от корневого коммутатора через сегмент 1). Порт 2 коммутатора 3 установил на основании принятых пакетов, что минимальное расстояние равно 20 единицам — это соответствовало случаю прохождения пакета от порта 2 корневого моста через сегмент 2, затем через коммутатор 4 и сегмент 3. Коммутатор 2 при выборе корневого порта «столкнулся» с ситуацией, когда у его портов 1 и 2 равное расстояние до корня — по 10 единиц (порт 1 принимает пакеты от порта 1 корневого коммутатора через один промежуточный сегмент — сегмент 1, так же как порт 2 получает пакеты от порта 2 корневого коммутатора через сегмент 2). Поскольку числовое значение идентификатора 1 меньше, чем идентификатора 2, то корневым и был выбран порт 1.

Кроме корневого порта коммутаторы распределенным образом выбирают назначенный порт для каждого сегмента сети. Каждый коммутатор сегмента прежде всего исключает из рассмотрения свой корневой порт (для сегмента, к которому он подключен, всегда существует другой коммутатор, который расположен ближе к корню). Для каждого из оставшихся портов выполняется сравнение принятых по ним минимальных расстояний до корня (еще до наращивания па условное время сегмента) с расстоянием до корня корневого порта данного коммутатора.

Если все принятые на этом порту расстояния оказываются больше, чем рас-

стояние от собственного корневого порта, значит, для сегмента, к которому подключен порт, кратчайший путь к корневому коммутатору проходит через него, и он становится назначенным. Коммутатор делает все свои порты, для которых такое условие выполняется, назначенными.

Когда имеется несколько портов с одинаковым кратчайшим расстоянием до корневого коммутатора, выбирается порт с наименьшим идентификатором. В рассматриваемом примере коммутатор 2 при проверке порта 2 обнаружил, что через этот порт принимались пакеты с минимальным расстоянием 0 (это были пакеты от порта 2 корневого коммутатора 1). Так как собственный корневой порт у коммутатора 2 имеет расстояние до корня равное 10, то порт 2 этого коммутатора не является назначенным для сегмента 2.

Требуемое время

Для корректной передачи пользовательских пакетов сетью коммутаторов алгоритм STA формализует состояния портов и определяет правила их смены только после того, как конфигурация покрывающего дерева установится во всех коммутаторах сети и для всех портов.

Каждый порт коммутатора, поддерживающего алгоритм STA, может находиться в одном из пяти состояний (рис. 10.4):

- отключен (disabled);
- заблокирован (blocked);
- прослушивание (listening);
- обучение (learning);
- продвижение (forwarding).

В состояние «отключен» порт может быть переведен только по команде конфигурирования, поданной администратором локально или удаленно по протоколу управления. В таком режиме порт не участвует ни в работе протокола STA, ни в продвижении пакетов данных. Вывести порт из этого состояния можно тоже только командой конфигурирования.

При инициализации коммутатора, поддерживающего STA, все порты (за исключением отключенных) автоматически переводятся в состояние «заблокирован». В этом случае порт генерирует, принимает, обрабатывает и ретранслирует пакеты BPDU. Коммутатор, которому принадлежит порт, принимает участие в выборе корневого коммутатора, а также решает, какой из его портов является корневым, а какой — назначенным. Однако пакеты данных порт пока не передает — этим полезным делом он занимается только в состоянии «продвижение».

Если порт претендует на роль корневого или назначенного, он переходит в состояние «прослушивание», что происходит обычно в начальный момент работы алгоритма STA, когда пакеты BPDU от других мостов еще не получены и коммутатор считает себя корневым, а все свои порты — назначенными. Режим «прослушивание» устанавливается на время, определяемое таймером смены состояний (Forwarding Timer). Интервал, выдерживаемый с помощью этого таймера и равный обычно 15 с (его значение может конфигурироваться), необхо-

дим для гарантированного получения всех пакетов BPDU, сгенерированных коммутаторами сети. Порт продолжает генерировать (если он принадлежит коммутатору, который считает себя корневым), принимать, обрабатывать и ретранслировать BPDU. Если в течение этого времени порт получит BPDU с лучшими параметрами, чем его собственные (расстояние, идентификатор коммутатора или порта), то он перейдет в состояние «заблокирован». В противном случае порт переводится в состояние «обучение».

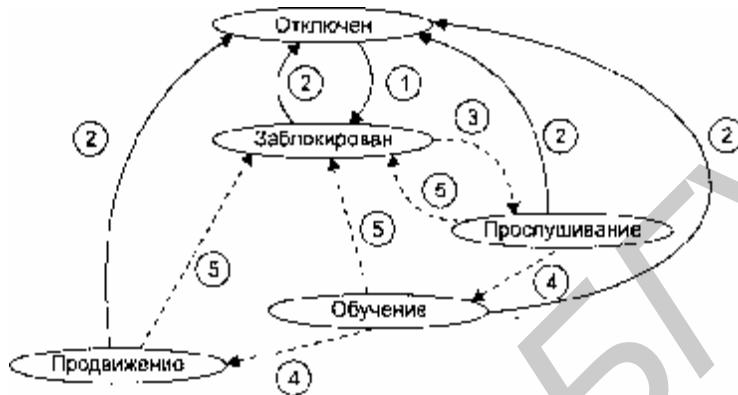


Рис. 10.4. Состояния порта

Теперь коммутатор начинает принимать (но не продвигать) пакеты данных и на основе адресов их источников строить таблицу продвижения. Это обычный режим обучения прозрачного моста, который ранее нельзя было активизировать, так как порт не был уверен в том, что он останется корневым или назначенным и будет передавать пакеты данных. Состояние «обучение» также выдерживается о течение интервала таймера смены состояний, то есть 15 с по умолчанию. При этом порт продолжает участвовать в работе STA, так что поступление BPDU с лучшими параметрами переводит его в состояние «заблокирован».

И только после двукратной выдержки по таймеру порт переходит в состояние «продвижение» и обрабатывает пакеты данных в соответствии с построенной таблицей (которая продолжает модифицироваться, отражая изменения в структуре сети).

В процессе нормальной работы корневой коммутатор продолжает генерировать конфигурационные пакеты BPDU с интервалом hello, а остальные коммутаторы получают их через свои корневые порты и ретранслируют назначенными портами. У коммутатора могут отсутствовать назначенные порты, как у коммутаторов 2 и 4, но он все равно участвует в работе протокола STA, так как корневой порт принимает служебные пакеты BPDU.

Если по истечении максимального времени жизни сообщения (по умолчанию — 20 с) корневой порт любого коммутатора сети не получит служебный пакет BPDU, то он инициализирует новую процедуру построения покрывающего дерева. При этом на все порты генерируется и передается пакет BPDU, в котором коммутатор указывает себя в качестве корневого. Аналогичным образом поступают и другие коммутаторы сети с истекшим таймером максимального времени жизни сообщения, в результате чего выбирается новая активная конфигурация.

Недостатки и достоинства STA

Одним из основных достоинств алгоритма Spanning Tree является то, что в отличие от многих упрощенных алгоритмом, где переход и резервное соединение осуществляется исключительно при отказе непосредственного соседа-устройства, он принимает решение о реконфигурировании с учетом не только связей с соседями, но и связей в отдаленных сегментах сети.

А вот то, что в сетях с большим количеством коммутаторов время определения новой активной конфигурации может оказаться слишком большим, можно отнести к недостаткам алгоритма. Если в сети используются значения тайм-аутов по умолчанию, переход на новую конфигурацию может занять свыше 50 с: 20 с понадобится на констатацию факта потери связи с корневым коммутатором (истечение таймера — единственный способ узнать об этом событии в стандартном варианте STA) и еще 2x15 с потребуется для перехода портов в состояние «продвижение».

Имеющиеся многочисленные нестандартные версии STA позволяют сократить время реконфигурирования за счет усложнения алгоритма, например добавления новых типов служебных сообщений. Недавно разработана новая версия Spanning Tree — спецификация IEEE 802.1w — для ускорения работы протокола, но уже стандартным способом.

Еще быстрее работают процедуры реконфигурирования сети при отказах, когда избыточные связи образованы за счет агрегирования нескольких каналов между двумя коммуникационными устройствами. В этом случае исключение из общей конфигурации неисправного канала происходит практически мгновенно — в течение нескольких микросекунд, требуемых на распознавание отказа на физическом уровне и реконфигурацию.

Выводы

- Для автоматического поддержания резервных связей в сложных сетях в коммутаторах реализуется алгоритм покрывающего дерева - Spanning Tree Algorithm. Этот алгоритм описан в документе IEEE 802.1D и основан на периодической генерации служебных кадров, с помощью которых выявляются и блокируются петлевидные связи и сети.
- Протокол STA находит конфигурацию покрывающего дерева за три этапа. На первом этапе определяется корневой коммутатор, на втором — корневые порты и на третьем — назначенные порты сегментов.
- Каждый порт может находиться в одном из пяти состояний:
 - отключен (disabled);
 - заблокирован (blocked);
 - прослушивание (listening);
 - обучение (learning);
 - продвижение (forwarding).
- Переход из состояния в состояние происходит либо при поступлении слу-

жебного кадра с новой информацией, либо по времени.

- Недостатком протокола STA 802.1D является сравнительно большое время установления новой активной конфигурации — около 50 с. Новый стандарт 802.1w устраняет этот недостаток.

Агрегирование каналов в локальных сетях

Агрегирование физических каналов (Link Aggregation) между двумя коммуникационными устройствами в один логический канал является еще одной формой использования избыточных альтернативных связей в локальных сетях. В данном контексте термин «локальные сети» используется как эквивалент технологий канального уровня, таких как Ethernet, и устройств, работающих на этом уровне, — коммутаторов, сетевых адаптеров и маршрутизатором (в последнем случае имеется в виду соответствующий слой аппаратного и программного обеспечения). Именно на канальном уровне протоколов локальных сетей проявляются особенности использования избыточных связей, связанные с дублированием и зацикливанием кадров в петлевидных маршрутах. И хотя эти проблемы могут быть решены на более высоком сетевом уровне маршрутизаторами IP/IPX и программными средствами операционных систем, поддержка избыточных связей такими экономичными и широко распространенными устройствами локальных сетей, как коммутаторы второго уровня, обладает рядом преимуществ. Прежде всего, снижается стоимость сети, так как многие ее участки строятся без привлечения относительно дорогих маршрутизаторов (коммутаторов 3-го уровня), кроме того, во многих случаях повышается скорость реакции на обрыв связи.

Транки и логические каналы

Отличие техники агрегирования связей от описанного выше алгоритма покрывающего дерева достаточно принципиально. Алгоритм покрывающего дерева переводит избыточные связи в горячий резерв, оставляя в рабочем состоянии только минимальный набор каналов, необходимый для обеспечения связности сегментов сети. В этом случае повышается надежность сети, так как при отказе соединения в строю автоматически вводятся резервные связи и через небольшой промежуток времени восстанавливается работоспособность сети.

При агрегировании физических каналов нее избыточные связи остаются в рабочем состоянии, а существующий трафик распределяется между ними для достижения баланса загрузки (рис. 10.5). При отказе одного из составляющих такого логического канала, который часто называется также транком, трафик распределяется между оставшимися каналами. На рисунке примером такой ситуации

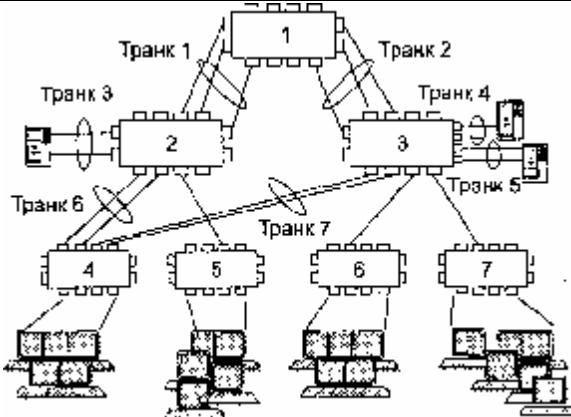


Рис. 10.5. Агрегирование физических каналов

Агрегирование каналов применяется как для связей между двумя коммутаторами локальной сети, так и для связей между компьютером и коммутатором (чаще всего этот вариант применяется для высокоскоростных и ответственных серверов) или между портами маршрутизатора. В двух последних случаях все сетевые адаптеры или порты маршрутизатора, которые входят в транк, разделяют один и тот же сетевой адрес (или одни и те же адреса, так как транк, как и обычный порт/адаптер, может иметь их несколько). Поэтому для протокола IP или другого протокола сетевого уровня порты транка неразличимы, что соответствует концепции единого логического канала, лежащей в основе агрегирования.

Очевидно, что агрегирование каналов приводит как к повышению производительности (вместо одного канала одновременно используется несколько), так и к повышению надежности. Но все методы агрегирования, применяемые в настоящее время, обладают существенным ограничением — в них учитываются только связи между двумя соседними коммутаторами сети и полностью игнорируется все, что происходит вне этого участка сети. Например, работа транка 1 никак не координируется с работой транка 2, и наличие обычной связи между коммутаторами 2 и 3, которая создает вместе с транками 1 и 2 петлю, не учитывается. Поэтому агрегирование каналов нужно применять одновременно с алгоритмом покрывающего дерева — если администратор сети хочет использовать все топологические возможности объединения узлов сети. Для алгоритма STA транк должен выглядеть как одна связь, тогда все его вычисления останутся в силе.

Существует большое количество фирменных реализаций механизма агрегирования каналов. Наиболее популярные принадлежат, естественно, лидерам в секторе оборудования для локальных сетей. Это такие реализации, как Fast EtherChannel и Gigabit EtherChannel компании Cisco, MultiLink Trunking компании Nortel, Adaptive Load Balancing компании Intel и ряд других. Стандарт IEEE 802.3ad Link Aggregation обобщает эти подходы.

Проблемы транковых соединений

Рассмотрим теперь подробней, в чем состоят особенности работы коммутатора в случае, когда его порты образуют транк. Во фрагменте сети, приведенном на рис. 10.6, два коммутатора — 1 и 2 — связаны четырьмя физическими каналами. Необходимо отметить, что транк может быть односторонним или двусторонним. Каждый коммутатор контролирует только отправку кадра, принимая решение, на какой из выходных портов его нужно передать. Поэтому, если оба коммутатора считают связывающие их каналы храпком, то он будет двусторонним, в противном случае — односторонним.

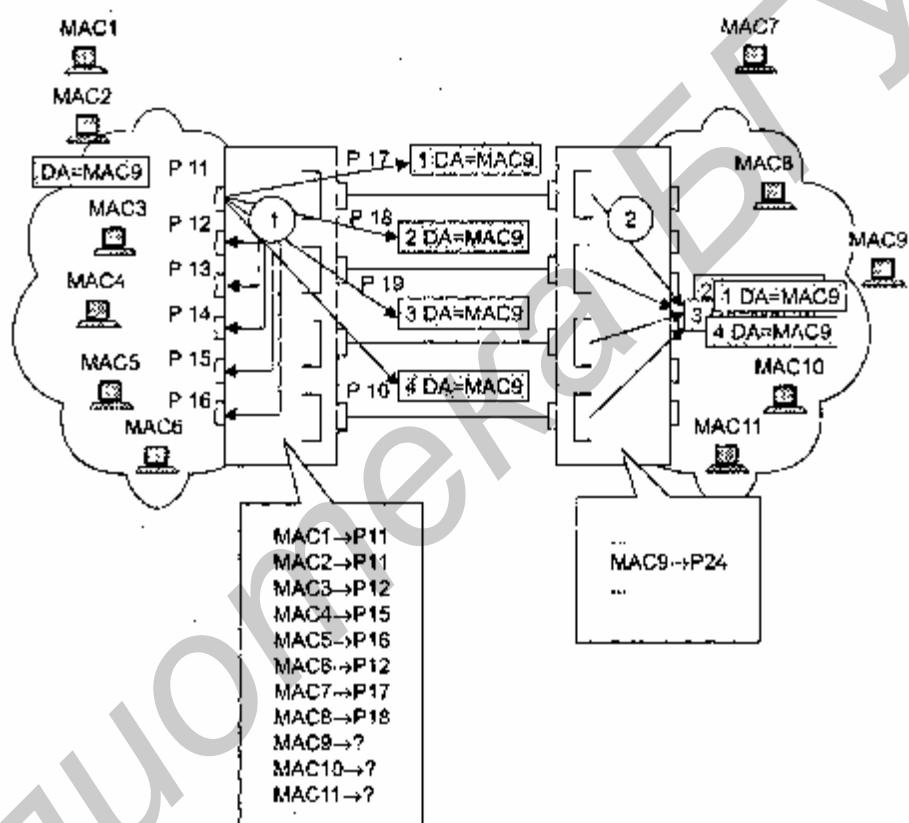


Рис. 10.6. Размножение пакетов с неизученным адресом при использовании параллельных каналов между коммутаторами

Рисунок иллюстрирует поведение коммутатора 1 по отношению к параллельным связям. В том случае, когда они не рассматриваются данным коммутатором как агрегированный канал, возникают проблемы с кадрами двух типов — во-первых, с кадрами, несущими в себе еще не изученные коммутатором уникальные адреса, и, во-вторых, с кадрами, в которых указан широковещательный или групповой адрес. Алгоритм прозрачного моста требует от коммутатора передавать кадр с неизученным (отсутствующим в таблице продвижения) адресом на все порты, кроме того, с которого кадр был принят. При наличии параллельных каналов такой кадр будет «размножен» в количестве, равном количеству каналов — в приведенном примере коммутатор 2 примет четыре копии ориги-

нального кадра. Если коммутатор 2 уже изучил адрес назначения MAC9, то он передаст каждую копию кадра узлу назначения через один порт, а если нет — то через все, в том числе и через порты параллельных каналов (естественно, кроме того, с которого кадр получен, так что кадр 1 в таком случае поступит на порты P18, P19 и P10 коммутатора 1).

При этом произойдет также зацикливание кадров — они будут постоянно циркулировать между двумя коммутаторами, причем удалить их из сети невозможно, так как в кадрах канального уровня отсутствует поле «срок жизни», часто используемое в протоколах верхних уровней, например в IP и ТРХ.

В любом случае кадр с неизученным адресом повысит нагрузку на сеть за счет увеличения числа кадров, что чревато возникновением заторов, задержек и потерь данных. Кроме роста нагрузки дублирование кадров может привести также к неэффективной работе многих протоколов верхнего уровня. Примером может служить узел, работающий по протоколу TCP. При дублировании квитанций, подтверждающих факт доставки данных адресату, некоторые реализации этого протокола резко уменьшают размер окна приема, снижая тем самым скорость отправки данных в сеть передатчиком (дублирование квитанции интерпретируется как признак потери данных — раз приемник повторно присыпает квитацию на одни и те же данные, значит, с большой степенью вероятности можно считать, что последующие кадры утеряны).

Еще больше проблем создают кадры с широковещательным адресом — они всегда должны передаваться на все порты, кроме исходного, так что в любом случае «засорение» сети посторонним трафиком окажется значительным и кадры будут зацикливаться.

С кадрами, у которых адрес назначения уникален и изучен, проблем у коммутаторов, связанных параллельными каналами, не возникает — коммутатор передаст такой кадр па единственный порт, по которому пришел в свое время кадр с данным адресом источника.

Борьба с «размножением» пакетов

Разработчики механизмов агрегирования учили проблемы, возникающие при обработке кадров с неизученными, широковещательными и групповыми адресами. Решение достаточно простое — все порты, связанные с параллельными каналами, считаются одним логическим портом, который и фигурирует в таблице продвижения вместо нескольких физических. В примере, представленном на рис. 10.6, в таблице продвижения вместо портов P17, P18, P19 и P10 фигурирует логический порт AL11. С этим портом связаны адреса всех узлов, путь к которым лежит через коммутатор 2. При этом изучение нового адреса по любому из физических портов, входящих в транк, приводит к появлению в таблице продвижения коммутатора новой записи с идентификатором логического порта. Поступающий в коммутатор кадр, адрес назначения которого изучен и связан с идентификатором логического порта, передается на один (и только один!) выходной физический порт, входящий в состав транка. Точно так же коммутатор

поступает с неизученными, широковещательными и групповыми адресами — для передачи кадра используется только одна из связей. На порты коммутатора, не входящие в транк, это изменение в логике обработки кадров не распространяется. Так, коммутатор 1 всегда передает кадр с неизученным или широковещательным адресом на порты P11-P16.

Благодаря такому решению кадры не дублируются и описанные проблемы не возникают. Правда, это справедливо только, если оба коммутатора рассматривают каналы как транк, в противном случае часть проблем остается. Например, если коммутатор 2 не считает связи транком, то кадр с неизвестным для коммутатора 2 адресом назначения будет им размножен и отправлен в обратном направлении коммутатору 1 в трех копиях. Поэтому полностью использовать свойства транка можно только тогда, когда он таким образом сконфигурирован с двух сторон.

Выбор порта

Остается открытым вопрос — какой из портов коммутатора нужно использовать для продвижения кадра через транк? Можно предложить несколько вариантов ответов. Так как одной из целей агрегирования каналов является повышение суммарной производительности участка сети между двумя коммутаторами (или коммутатором и сервером), то желательно кадры распределять по портам транка динамически, учитывая текущую загрузку каждого порта и направляя кадры в наименее загруженные (с меньшей длиной очереди) порты. Динамический способ распределения кадров, учитывающий текущую загрузку портов и обеспечивающий баланс нагрузки между всеми связями транка, должен приводить, казалось бы, к максимальной пропускной способности транка. Однако такое утверждение справедливо не всегда, так как в нем не учитывается поведение протоколов верхнего уровня, а только интенсивность общего потока кадров канального уровня. В то же время существует ряд протоколов, производительность которых может существенно уменьшиться, если пакеты сеанса связи между двумя конечными узлами будут приходить не в том порядке, в котором они отправлены узлом-источником. А такая ситуация может возникнуть, если два или более смежных по отношению к сеансу кадра будут передаваться через разные порты транка — из-за того, что очереди в буферах этих портов имеют разную длину, а значит, и задержка передачи кадра может быть разной, так что более поздний кадр обгонит более ранний.

Если протоколы стека IP (например, TCP) толерантны к неупорядоченности приходящих пакетов, то протоколы стека IPX и DLCnet резко теряют полезную производительность, так как считают такие события потерей пакета и запрашивают его повторную передачу. Поэтому в большинстве реализаций механизмов агрегирования используются методы статического, а не динамического распределения кадров по портам.

Статическое распределение подразумевает закрепление за определенным портом транка потока кадров определенного сеанса между двумя узлами, так что

все кадры будут проходить через одну и ту же очередь и последовательность их не изменится. Обычно при статическом распределении выбор порта для некоторого сеанса выполняется на основании определенных признаков, имеющихся в поступающих пакетах. Чаще всего такими признаками являются MAC-адреса источника или назначения, или оба вместе. В популярной реализации Fast Ether Channel компании Cisco коммутаторов семейства Catalyst 5000/6000 для выбора номера порта транка используется операция исключающего ИЛИ (XOR) над двумя последними битами MAC-адресов источника и приемника. Результат этой операции имеет четыре значения: 00, 01, 10 и 11, которые и являются условными номерами портов транка. На рис. 10.7 приведен пример сети, в которой работает механизм Fast EtherChannel (аналогично работает и Gigabit Ether-Channel) для коммутаторов Catalyst 5000/GO00. Распределение потоков для сеансов между конечными узлами получается при этом достаточно случайным. Так как распределение не учитывает реальной нагрузки, которую создает каждый сеанс, общая пропускная способность транка может использоваться нерационально, особенно если интенсивности сеансов намного отличаются друг от друга. Кроме того, алгоритм распределения не гарантирует даже равномерного в количественном отношении распределения сеансов по портам. Случайный набор MAC-адресов в сети может привести к тому, что через один порт будут проходить несколько десятков сеансов, а через другой — только два-три. Выравнивания нагрузки портов можно при данном алгоритме достигнуть только при большом количестве компьютеров и сеансов связи между ними, что характерно для магистральных коммутаторов, которыми и являются коммутаторы Catalyst 5000/6000.

Библиомегаполис

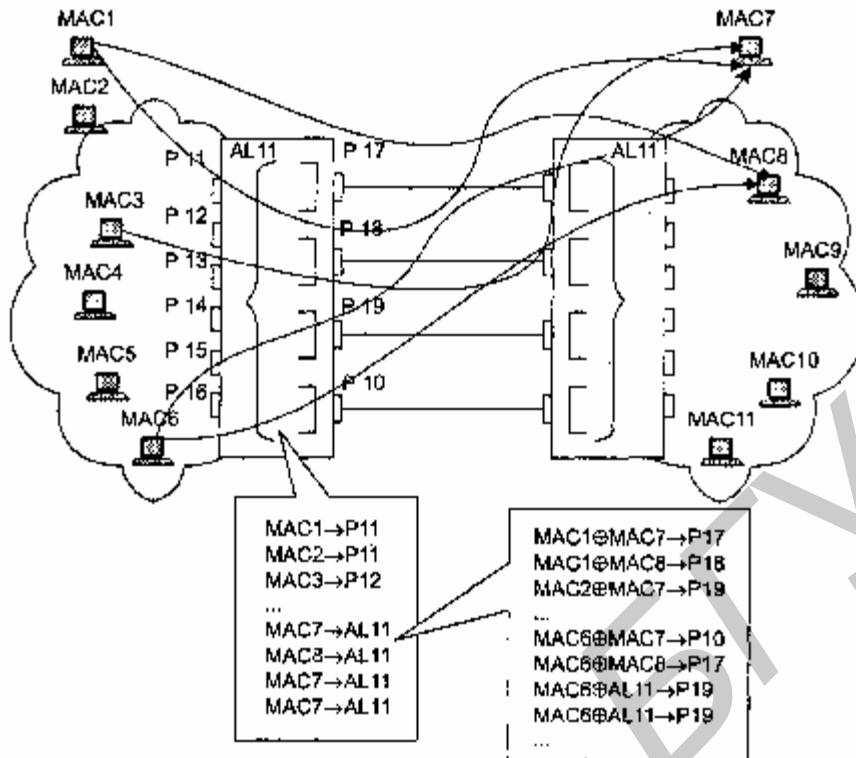


Рис. 10.7. Пример сети с механизмом Fast EtherChannel

Можно предложить и другие способы распределения сеансов по портам. Например, в соответствии с IP-адресами пакетов, которые переносятся кадрами канального уровня, номерами виртуальных локальных сетей (VLAN), тинами прикладных протоколов (почта по одному порту, web-трафик по другому и т. д.), полезным оказывается назначение порту сеансов с MAC-адресами, которые были изучены именно через этот порт — чтобы трафик сеанса проходил через один и тот же порт и обоих направлениях.

Чем больше признаков учитывает механизм распределения, тем более равномерно нагружаются порты транка и, следовательно, более производительно работает транк. Некоторые реализации могут также выполнять распределение с учетом текущей загрузки портов, но не для каждого отдельного пакета, а для всех пакетов сеанса. Такой способ наряду с учетом номера порта, по которому изучен MAC-адрес назначения, применяется в механизмах Fast EtherChannel в коммутаторах Catalyst 2900X.

Стандартный способ создания агрегированных каналов, описанный в спецификации 802.3ad, предполагает возможность распределения физических портов, объединяемых в логический порт, по различным коммутаторам. Для того чтобы коммутаторы могли автоматически обеспечиваться информацией о принадлежности какого-либо физического порта к определенному логическому порту, в спецификации предложен служебный протокол Link Control Aggregation Protocol (LCAP). Поэтому возможны такие конфигурации агрегированных каналов, которые увеличивают отказоустойчивость сети не только на участках между двумя коммутаторами, но и в более сложных топологиях (рис. 10.8).

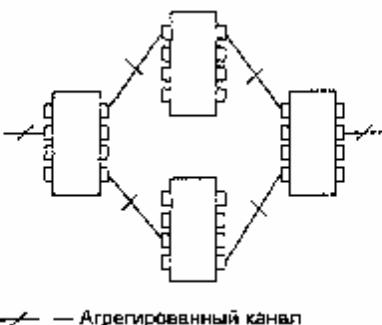


Рис. 10.8. Распределение агрегированных каналов

Восстановление при отказе

При отказе какого-либо канала транка все пакеты сеансов, назначенные для соответствующего порта, будут направляться на один из оставшихся портов. Обычно такое восстановление связности занимает от единиц до десятков миллисекунд. Это объясняется тем, что во многих реализациях транка после отказа канала все MAC-адреса, которые были с ним связаны, принудительно помечаются как неизученные. Затем коммутатор повторяет процедуру изучения этих адресов, дожидаясь, когда один из них придет на какой-либо порт транка, после чего они снова становятся изученными. После этого процедура назначения сеанса портам выполняется заново, естественно, учитывая только работающие порты. Так как тайм-ауты в сеансах протоколов локальных сетей обычно небольшие, коротким оказывается и время восстановления соединения.

Выводы

- Агрегирование нескольких физических каналов в один логический является одной из форм использования альтернативных маршрутов в локальных сетях на коммутаторах.
- Агрегирование каналов повышает как производительность, так и надежность сети. Стандартный способ агрегирования описан в спецификации 802.3ad.
- Основной сложностью при реализации агрегирования является обработка кадров с широковещательными и неизученными адресами — они не должны передаваться на физические порты, принадлежащие одному и тому же логическому порту.
- Агрегирование может учитывать наличие сеансов трафика, для которых пересортировка последовательности кадров может оказаться нежелательным. В таком случае все пакеты одного сеанса всегда передаются через один и тот же физический порт транка и распределяются по портам сеанса целиком.
- Агрегированный канал может быть образован не только между двумя соседними коммутаторами, но и распределяться между портами нескольких комму-

таторов. Для автоматического уведомления о принадлежности физического порта к определенному агрегированному порту разработан протокол Link Control Aggregation Protocol (LCAP).

Виртуальные локальные сети

Кроме своего основного назначения — повышения пропускной способности связей в сети — коммутатор позволяет локализовывать потоки информации в сети, а также контролировать эти потоки и управлять ими, опираясь на механизм пользовательских фильтров. Однако пользовательский фильтр может запретить передачи кадров только по конкретным адресам, а широковещательный трафик он передает всем сегментам сети. Так требует алгоритм работы моста, который реализован в коммутаторе, поэтому сети, созданные на основе мостов и коммутаторов, иногда называют плоскими — из-за отсутствия барьеров на пути широковещательного трафика.

Технология виртуальных локальных сетей (Virtual LAN, VLAN), которая появилась несколько лет тому назад в коммутаторах, позволяет преодолеть указанное ограничение. Виртуальной сетью называется группа узлов сети, трафик которой, в том числе и широковещательный, на канальном уровне полностью изолирован от других узлов сети (рис. 10.9). Это означает, что передача кадров между разными виртуальными сетями па основании адреса канального уровня невозможна, независимо от типа адреса — уникального, группового или широковещательного.

В то же время внутри виртуальной сети кадры передаются по технологии коммутации, то есть только на тот порт, который связан с адресом назначения кадра.

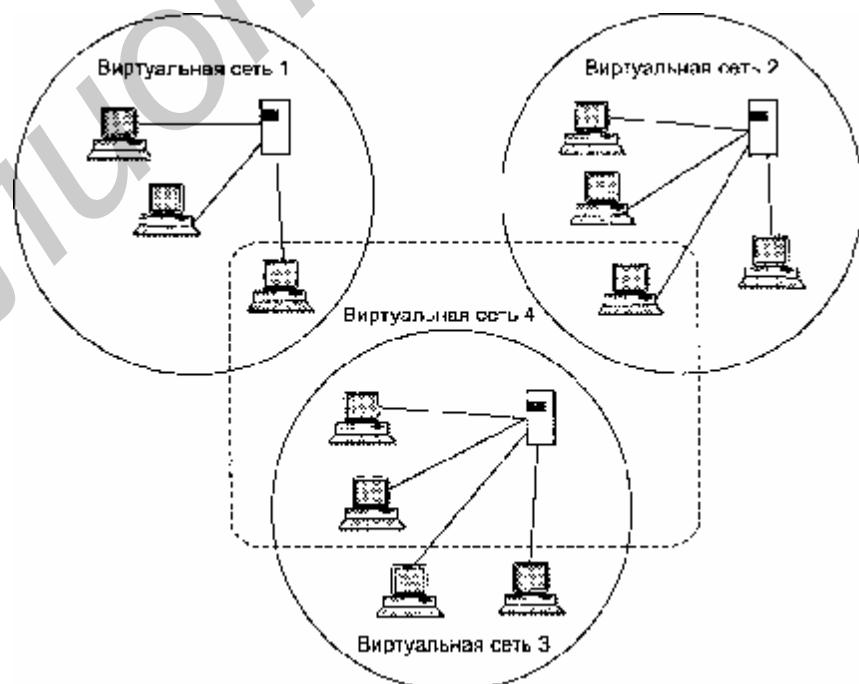


Рис. 10.9. Виртуальные сети

Виртуальные сети могут пересекаться, если один или несколько компьютеров входят в состав более чем одной виртуальной сети. На рис. 10.9 сервер электронной почты входит в состав виртуальных сетей 3 и 4. Это значит, что его кадры передаются коммутаторами всем компьютерам, входящим в эти сети. Если же какой-то компьютер входит в состав только виртуальной сети 3, то его кадры до сети 4 доходить не будут, но он может взаимодействовать с компьютерами сети 4 через общий почтовый сервер. Такая схема не полностью защищает виртуальные сети друг от друга — так, широковещательный шторм, возникший на сервере электронной почты, захлестнет и сеть 3, и сеть 4. Говорят, что виртуальная сеть образует домен широковещательного трафика (broadcast domain), по аналогии с доменом коллизий, который образуется повторителями сетей Ethernet.

Назначение VLAN

Назначение технологии VLAN состоит в облегчении процесса создания изолированных сетей, которые затем должны связываться с помощью маршрутизаторов, реализующих какой-либо протокол сетевого уровня, например IP. Такое построение сети создает гораздо более мощные барьеры на пути ошибочного трафика из одной сети в другую. Сегодня считается, что любая крупная сеть должна включать маршрутизаторы, иначе потоки ошибочных кадров, например широковещательных, будут периодически «затапливать» нею сеть через прозрачные для них коммутаторы, приводя ее в неработоспособное состояние. Технология виртуальных сетей создает гибкую основу для построения крупной сети, соединенной маршрутизаторами, так как коммутаторы позволяют создавать полностью изолированные сегменты программным путем, не прибегая к физической коммутации.

До появления технологии VLAN для создания отдельной сети использовались либо физически изолированные сегменты коаксиального кабеля, либо несвязанные между собой сегменты, построенные на повторителях и мостах. Затем эти сети связывались маршрутизаторами в единую составную сеть (рис. 10.10).

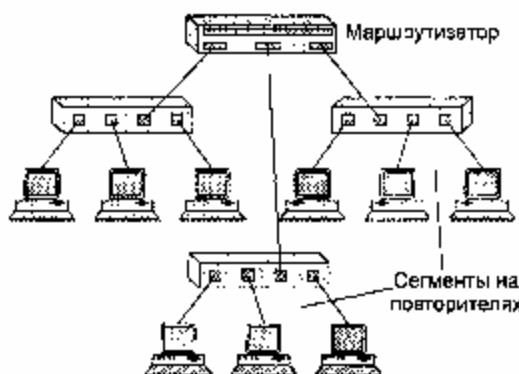


Рис. 10.10. Интерсеть, состоящая из сетей, построенных на основе повторителей

Изменение состава сегментов (переход пользователя в другую сеть, дробление крупных сегментов) при таком подходе подразумевает физическую перекоммутацию разъемов на передних панелях повторителей или в кроссовых панелях, что не очень удобно в больших сетях — много физической работы, к тому же высока вероятность ошибки.

Поэтому для устранения необходимости физической перекоммутации узлов стали применять многосегментные концентраторы, что дало возможность программировать состав разделяемого сегмента без физической перекоммутации.

Однако решение задачи изменения состава сегментов с помощью концентраторов накладывает большие ограничения на структуру сети — количество сегментов такого повторителя обычно невелико, поэтому выделить каждому узлу свой сегмент, как это можно сделать с помощью коммутатора, нереально. Кроме того, при таком подходе вся работа по передаче данных между сегментами ложится на маршрутизаторы, а коммутаторы со своей высокой производительностью остаются «не у дел». Поэтому сети, построенные на основе повторителей с конфигурационной коммутацией, по-прежнему основаны па разделении среды передачи данных между большим количеством узлов и, следовательно, обладают гораздо меньшей производительностью по сравнению с сетями, построенными на основе коммутаторов.

При использовании технологии виртуальных сетей в коммутаторах одновременно решаются две задачи:

- повышение производительности в каждой из виртуальных сетей, так как коммутатор передает кадры в такой сети только узлу назначения;
- изоляция сетей друг от друга для управления правами доступа пользователей и создания защитных барьеров на пути широковещательных штормов.

Для связи виртуальных сетей в общую сеть требуется привлечение сетевого уровня. Он может быть реализован в отдельном маршрутизаторе, а может работать и в составе программного обеспечения коммутатора, который тогда становится комбинированным устройством — так называемым коммутатором 3-го уровня. Коммутаторы 3-го уровня рассматриваются в части IV книги.

Технология образования и работы виртуальных сетей с помощью коммутаторов долгое время не стандартизовалась, хотя и была реализована в очень широком спектре моделей коммутаторов разных производителей. Такое положение изменилось после принятия в 1998 году стандарта IEEE 802.1Q, который определяет базовые правила построения виртуальных локальных сетей, не зависящие от протокола канального уровня, поддерживаемого коммутатором.

Ввиду долгого отсутствия стандарта на VLAN каждый крупный производитель коммутаторов разработал свою технологию виртуальных сетей, которая, как правило, была несовместима с технологиями других производителей. Поэтому, несмотря на появление стандарта, не так уж редко встречается ситуация, когда виртуальные сети, созданные на коммутаторах одного производителя, не распознаются и, соответственно, не поддерживаются коммутаторами другого производителя.

Создание VLAN на основе одного коммутатора

При создании виртуальных сетей на основе одного коммутатора обычно используется механизм группирования в сети портов коммутатора (рис. 10.11). При этом каждый порт приписывается той или иной виртуальной сети. Кадр, пришедший от порта, принадлежащего, например, виртуальной сети 1, никогда не будет передан порту, который не принадлежит этой виртуальной сети. Порт можно присоединить нескольким виртуальным сетям, хотя на практике так делают редко — пропадает эффект полной изоляции сетей.

Группирование портов для одного коммутатора — наиболее логичный способ образования VLAN, так как виртуальных сетей, построенных на основе одного коммутатора, не может быть больше, чем портов. Если к одному порту подключен сегмент, построенный на основе повторителя, то узлы такого сегмента не имеет смысла включать в разные виртуальные сети — все равно трафик этих узлов будет общим.

Создание виртуальных сетей на основе группирования портов не требует от администратора большого объема ручной работы — достаточно каждый порт присоединить к одной из нескольких заранее поименованных виртуальных сетей. Обычно такая операция выполняется с помощью специальной программы, прилагаемой к коммутатору. Администратор создает виртуальные сети путем перетаскивания мышью графических символов портов на графические символы сетей.

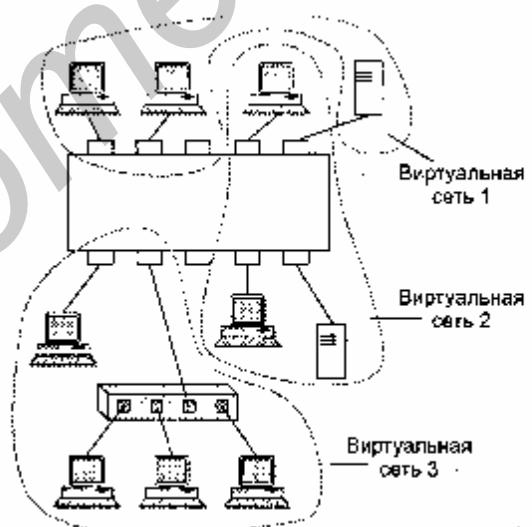


Рис. 10.11. Виртуальные сети, построенные на одном коммутаторе

Второй способ образования виртуальных сетей основан на группировании MAC-адресов. Каждый MAC-адрес, который изучен коммутатором, присоединяется той или иной виртуальной сети. При существовании в сети множества узлов этот способ требует выполнения большого количества ручных операций от администратора. Однако при построении виртуальных сетей на основе нескольких коммутаторов он оказывается более гибким, чем способ группирова-

ния портов.

Создание VLAN на основе нескольких коммутаторов

Рисунок 10.12 иллюстрирует проблему, возникающую при создании виртуальных сетей на основе нескольких коммутаторов, поддерживающих технику группирования портов. Если узлы какой-либо виртуальной сети подключены к разным коммутаторам, то для соединения коммутаторов каждой такой сети должна быть выделена своя пара портов. В противном случае, если коммутаторы будут связаны только одной парой портов, информация о принадлежности кадра той или иной виртуальной сети при передаче из коммутатора в коммутатор будет потеряна. Таким образом, коммутаторы с группированием портов требуют для своего соединения столько портов, сколько виртуальных сетей они поддерживают. Порты и кабели используются при таком способе очень расточительно. Кроме того, при соединении виртуальных сетей через маршрутизатор для каждой виртуальной сети выделяется отдельный кабель и отдельный порт маршрутизатора, что также приводит к большим накладным расходам.

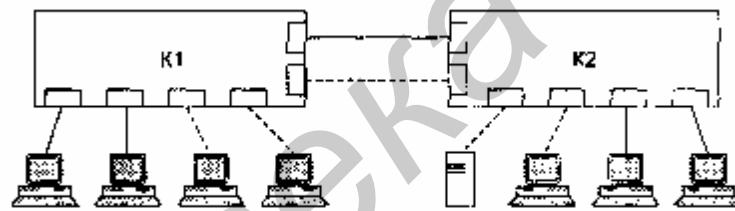


Рис. 10.12. Построение виртуальных сетей на нескольких коммутаторах с группированием портов

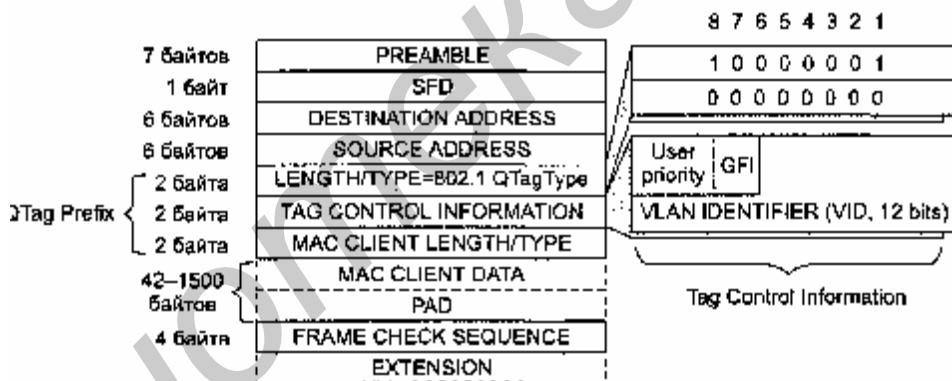
Группирование MAC-адресов в виртуальную сеть на каждом коммутаторе добавляет от необходимости их связи по нескольким портам, поскольку в этом случае MAC-адрес является меткой виртуальной сети. Однако этот способ требует выполнения большого количества ручных операций по маркировке MAC-адресов на каждом коммутаторе сети.

Описанные дна подхода основаны только на добавлении дополнительной информации к адресным таблицам моста, и в них отсутствует возможность встраивания в передаваемый кадр информации о принадлежности кадра к виртуальной сети. Остальные подходы используют имеющиеся или дополнительные поля кадра для сохранения информации о принадлежности кадра при его перемещениях между коммутаторами сети. При этом нет необходимости запоминать в каждом коммутаторе принадлежность всех MAC-адресов интэрсети виртуальным сетям.

Дополнительное поле с пометкой о номере виртуальной сети используется только тогда, когда кадр передается от коммутатора к коммутатору, а при передаче кадра конечному узлу оно обычно удаляется. При этом модифицируется протокол взаимодействия «коммутатор—коммутатор», а программное и аппаратное обеспечение конечных узлов остается неизменным. Примеров таких

фирменных протоколов много, но общий недостаток у них один — они не поддерживаются другими производителями. Компания Cisco предложила в качестве стандартной добавки к кадрам любых протоколов локальных сетей заголовок протокола 802.10, предназначенный для поддержки функций безопасности вычислительных сетей. Сама компания использует этот метод в тех случаях, когда коммутаторы объединяются между собой по протоколу FDDI. Однако эта инициатива не была поддержана другими ведущими производителями коммутаторов.

Для хранения номера виртуальной сети в стандарте IEEE 802.1Q предусмотрен дополнительный заголовок в два байта, который этот протокол делит с протоколом 802.1p. Помимо 3 бит для хранения приоритета кадра, описанных стандартом 802.1p, в этом заголовке 12 бит используются для хранения номера виртуальной сети, к которой принадлежит кадр. Эта дополнительная информация называется тегом виртуальной сети (VLAN TAG) и позволяет коммутаторам разных производителей создавать до 4096 общих виртуальных сетей. Такой кадр называют «помеченным» (tagged). Длина помеченного кадра Ethernet увеличивается на 4 байта, так как помимо двух байтов собственно тега добавляются еще два байта. Структура помеченного кадра Ethernet показана на рис. 10.13. При добавлении заголовка 802.1p/Q поле данных уменьшается на четыре байта.



Введение стандарта 802.1Q позволило производителям оборудования преодолеть различия в фирменных реализациях VLAN и добиться совместимости при построении виртуальных локальных сетей. Поддерживают технику VLAN производители как коммутаторов, так и сетевых адаптеров. В последнем случае сетевой адаптер может генерировать и принимать помеченные кадры Ethernet, содержащие поле VLAN TAG. Если сетевой адаптер генерирует помеченные кадры, то тем самым он определяет их принадлежность к той или другой виртуальной локальной сети, поэтому коммутатор должен обрабатывать их соответствующим образом, то есть передавать или не передавать на выходной порт в зависимости от принадлежности порта. Драйвер сетевого адаптера может получить номер своей (или своих) виртуальной локальной сети путем ручного конфигурирования его администратором сети, либо от некоторого приложения, работающего на данном узле. Такое приложение может работать и централизованно.

ванно на одном из серверов сети и управлять структурой всей сети. При поддержке VLAN сетевыми адаптерами можно обойтись без статического конфигурирования путем приписывания порта определенной виртуальной сети. Тем не менее техника статического конфигурирования VLAN остается популярной, так как она позволяет создать структурированную сеть без привлечения программного обеспечения конечных узлов.

Выводы

- Технология виртуальных локальных сетей (VLAN) позволяет в сети, построенной на коммутаторах, создать изолированные группы узлов, между которыми отсутствует любой тип трафика, в том числе и широковещательный.
- Виртуальные сети являются основой для создания крупных маршрутизируемых сетей и имеют преимущество перед физически изолированными сегментами гибкостью состава, изменяемого программным путем.
- Конфигурирование VLAN обычно ведется путем группирования порта». Для построение виртуальной локальной сети на основе нескольких коммутаторов желательно помечать передаваемые кадры специальной меткой — тегом, - идентифицирующей номер сети, к которой принадлежит отправитель кадра.
- Существует стандартный формат тега VLAN, он определен в спецификации 802.1Q. Под номер VLAN отводится 12 бит, что позволяет разбить есть на 4096 виртуальных сетей.

Типовые схемы применения коммутаторов в локальных сетях

Сочетание коммутаторов и концентраторов

При построении небольших сетей, составляющих нижний уровень иерархии корпоративной сети, вопрос о применении того или иного коммуникационного устройства сводится к вопросу о выборе между концентратором и коммутатором.

При ответе на этот вопрос нужно принимать во внимание несколько факторов. Безусловно, немаловажное значение имеет стоимости в пересчете за порт, которую нужно заплатить при выборе того или иного устройства. Из технических соображений в первую очередь нужно принять во внимание существующее распределение трафика между узлами сети. Кроме того, нужно учитывать перспективы развития сети: будут ли в скором времени применяться мультимедийные приложения, будет ли модернизироваться компьютерная база. Если да, то нужно заранее обеспечить резервы по пропускной способности применяемого коммуникационного оборудования. Использование технологии intranet также ведет к увеличению трафика, циркулирующего в сети, и это еще одно обстоятельство, которое необходимо учитывать при выборе устройства.

При выборе типа устройства — концентратор или коммутатор — нужно еще определить и тип протокола, который будут поддерживать его порты (или протоколов, если идет речь о коммутаторе, так как каждый порт может поддерживать отдельный протокол).

Сегодня выбор делается между протоколами трех скоростей — 10, 100 и 1000 Мбит/с. Поэтому, выбирая между концентратором и коммутатором, необходимо рассмотреть варианты концентратора с портами на 10, 100 и 1000 Мбит/с, а также несколько вариантов коммутаторов с различными комбинациями скоростей на портах.

Рассмотрим для примера вопрос о применимости коммутатора в сети с одним сервером и несколькими рабочими станциями, взаимодействующими только с сервером (рис. 10.14). Такая конфигурация сети часто встречается в сетях масштаба рабочей группы, особенно в сетях NetWare, где стандартные клиентские оболочки не могут взаимодействовать друг с другом.

Если коммутатор имеет все порты с одинаковой пропускной способностью, например 10 Мбит/с, то эта пропускная способность будет распределяться между всеми компьютерами сети. Возможности коммутатора по повышению общей пропускной способности сети оказываются для такой конфигурации невостребованными. Несмотря на микросегментацию сети, ее пропускная способность ограничивается пропускной способностью протокола одного порта, как и в случае применения концентратора с портами 10 Мбит/с. Небольшой выигрыш при использовании коммутатора будет достигаться лишь за счет уменьшения количества коллизий — вместо коллизий кадры будут просто попадать в очередь к передатчику порта коммутатора, к которому подключен сервер.

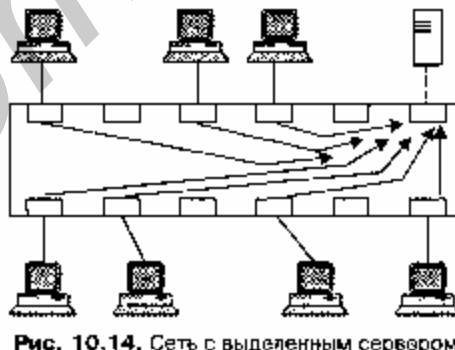


Рис. 10.14. Сеть с выделенным сервером

Чтобы коммутатор работал в сетях с выделенным сервером более эффективно, производители коммутаторов выпускают модели с одним высокоскоростным портом на 100 Мбит/с для подключения сервера и несколькими низкоскоростными портами на 10 Мбит/с для подключения рабочих станций. В этом случае между рабочими станциями распределяется уже 100 Мбит/с, что позволяет обслуживать в неблокирующем режиме 10-30 станций в зависимости от интенсивности создаваемого ими трафика.

Однако с таким коммутатором может конкурировать концентратор, поддерживающий протокол с пропускной способностью 100 Мбит/с, например Fast Ethernet. Его стоимость в пересчете за порт будет несколько ниже стоимости за

порт коммутатора с одним высокоскоростным портом, а производительность сети примерно та же.

Очевидно, что выбор коммуникационного устройства для сети с выделенным сервером достаточно сложен. Для принятия окончательного решения нужно принимать во внимание перспективы развития сети и отношении движения к сбалансированному трафику. Если в сети вскоре может потребоваться взаимодействие между рабочими станциями или же появиться второй сервер, то выбор необходимо делать в пользу коммутатора, который сможет поддержать дополнительный трафик без ущерба по отношению к основному.

В пользу коммутатора может сыграть и фактор расстояний — применение коммутаторов не ограничивает максимальный диаметр сети величинами в 2500 м или 210 м, которые определяют размеры домена коллизий при использовании концентраторов Ethernet и Fast Ethernet.

В целом же существует явная тенденция вытеснения концентраторов коммутаторами.

Стянутая в точку магистраль на коммутаторе

При всем разнообразии структурных схем сетей, построенных на коммутаторах, все они используют две базовые структуры — стянутую в точку магистраль и распределенную магистраль. На основе этих базовых структур затем строятся разнообразные структуры конкретных сетей.

Стянутая в точку, или коллапсированная, магистраль (*collapsed backbone*) — это структура, при которой объединение узлов, сегментов или сетей происходит па внутренней магистрали коммутатора. Преимуществом такой структуры является высокая производительность магистрали. Так как для коммутатора производительность внутренней шины или схемы общей памяти, объединяющей модули порта», в несколько десятков гигабит в секунду не является редкостью, то магистраль сети может быть весьма быстродействующей, причем ее скорость не зависит от применяемых в сети протоколов и может быть повышена путем замены одной модели коммутатора на другую.

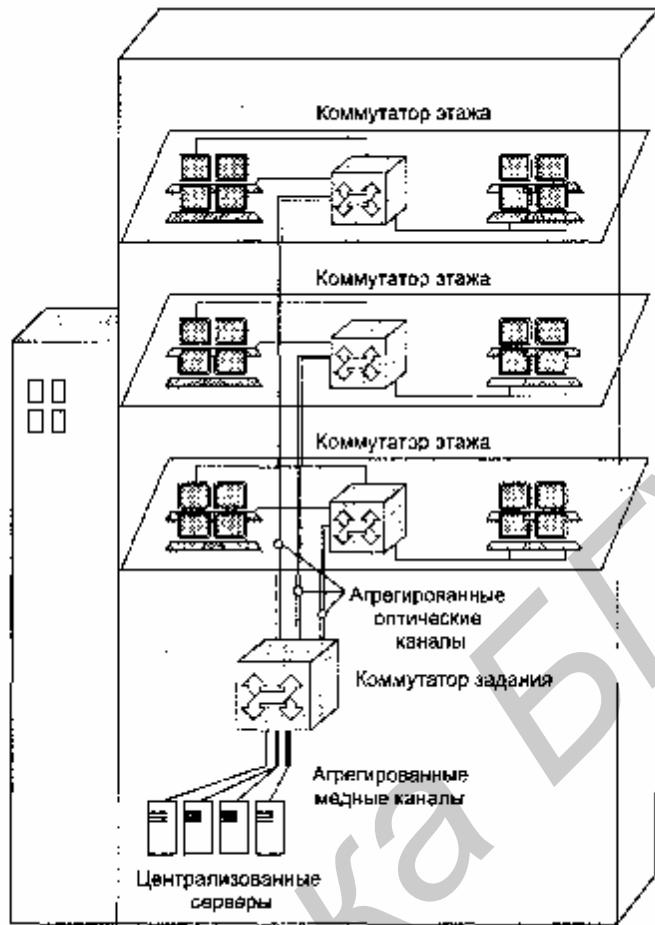


Рис. 10.15. Сеть здания со стянутой в точку магистралью

Пример сети такого типа показан на рис. 10.15. Здесь стянутая в точку магистраль реализована на центральном коммутаторе здания, который связан агрегированными каналами с коммутаторами этажей. На каждом этаже также установлен коммутатор, который образует стянутую в точку магистраль этажа. В том случае, когда к каждому порту коммутатора в такой схеме подключен только один узел, эта схема будет соответствовать микросегментированной сети.

Распределенная магистраль на коммутаторах

В сетях больших зданий или кампусов структура с коллапсированной магистралью не всегда рациональна или возможна. Такая структура приводит к протяженным кабельным системам, связывающим конечные узлы или коммутаторы сетей рабочих групп с центральным коммутатором, шина которого и является магистралью сети. Высокая плотность кабелей и их высокая стоимость ограничивают применение стянутой в точку магистрали в таких сетях. Иногда, особенно в сетях кампусов, просто невозможно стянуть все кабели в одно помещение из-за ограничений на длину связей, накладываемых технологией (например, все реализации технологий локальных сетей на витой паре ограничивают протяженность кабелей значением 100 м).

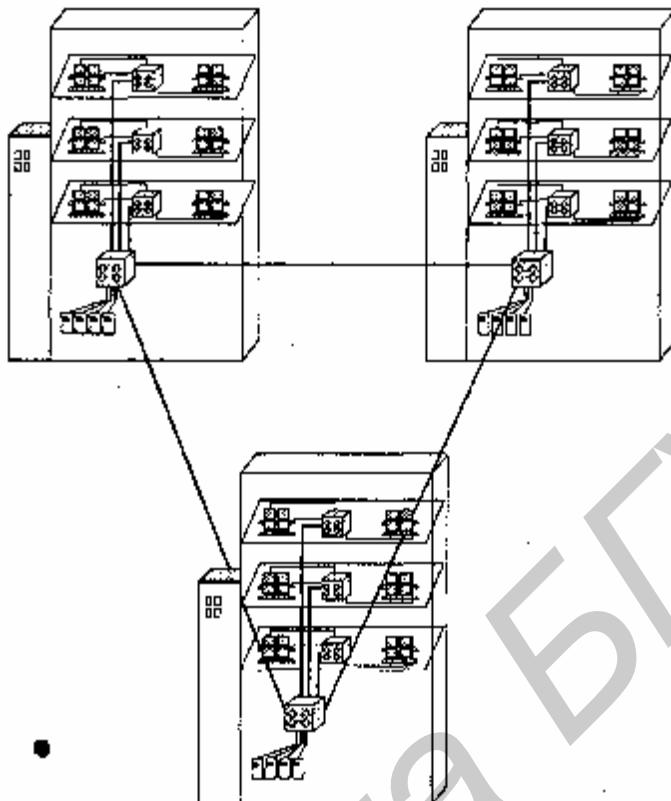


Рис. 10.16. Сеть с распределенной магистралью

Поэтому и локальных сетях, покрывающих большие территории, часто используется другой вариант построения сети - с распределенной магистралью. Пример такой сети показан на рис. 10.16. Здесь коммутаторы зданий соединены по схеме «каждый с каждым», образуя структуру с избыточными связями. Для нормальной работы в такой топологии необходимо, чтобы центральные коммутаторы зданий поддерживали протокол STA, в этом случае одна из связей между зданиями будет находиться в резерве, а две другие - передавать пользовательский трафик. В каждом из зданий по-прежнему используется коллапсированная магистраль, так что в целом сеть имеет комбинированную топологию. Распределенная магистраль сокращает стоимость кабельной системы и преодолевает ограничения на расстояния.

Однако скорость магистрали в этом случае будет, скорее всего, ниже скорости магистрали на внутреннейшине коммутатора. Причем скорость эта фиксированная и в настоящее время чаще всего не превышает 100 или 1000 Мбит/с. Агрегирование магистральных каналов частично решает эту проблему, так как скорость можно повысить до $N \times 1000$ Мбит/с, причем в соответствии со стандартом 802.3ad максимальное число каналов в транке — 8.

Выводы

- В последнее время наблюдается отчетливая тенденция вытеснения коммути-

торами концентраторов с нижних уровней крупных сетей.

□ Существуют две основные схемы применения коммутаторов: со стянутой в точку магистралью и с распределенной магистралью. В больших сетях эти схемы применяют комбинированно.

□ Сеть, построенная на коммутаторах без применения концентраторов, называется микросегментированной.

Вопросы и упражнения к части III

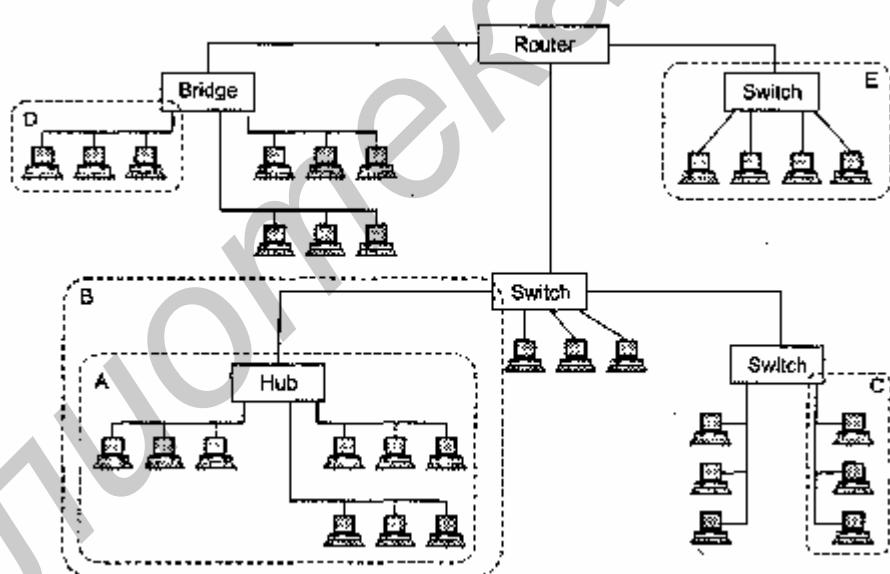
1. Поясните разницу между расширяемостью и масштабируемостью на примере технологии Ethernet.

2. Что такое коллизия:

○ ситуация, когда станция, желающая передать пакет, обнаруживает, что в данный момент другая станция уже заняла передающую среду (A);

○ ситуация, когда две рабочие станции одновременно передают данные в разделяемую передающую среду (B).

3. Что такое домен коллизий? Являются ли доменами коллизий показанные ниже фрагменты сети?



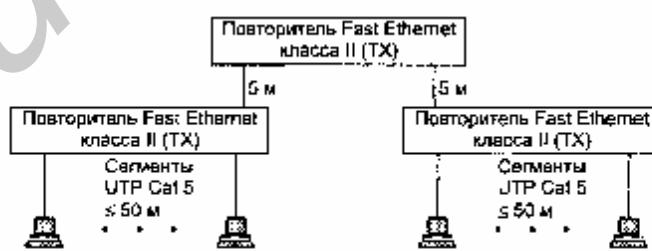
4. В чем состоят функции преамбулы и начального ограничителя кадра в стандарте Ethernet?

5. Какие сетевые средства осуществляют jabber-контроль?

6. Чему равны значения следующих характеристик стандарта 10Base-5:

- номинальная пропускная способность (бит/с);
- эффективная пропускная способность (бит/с);
- пропускная способность (кадр/с);
- внутрипакетная скорость передачи (бит/с);
- межбитовый интервал (с).

7. Чем объясняется, что минимальный размер кадра в стандарте 10Base-5 был выбран равным 64 байт?
8. Поясните смысл каждого поля кадра Ethernet.
9. Как известно, имеются 4 стандарта на формат кадров Ethernet. Выберите из нижеприведенного списка названия для каждого из этих стандартов. Учтите, что некоторые стандарты имеют несколько названий:
- Novell 802.2;
 - Ethernet II;
 - 802.3/802.2;
 - Novell 802.3;
 - Raw 802.3;
 - Ethernet DIX;
 - 802.3/LLC;
 - Ethernet SNAP.
10. Что может произойти в сети, в которой передаются кадры Ethernet разных форматов?
11. При каких типах ошибок в сети Ethernet концентратор обычно отключает порт?
12. Как величина MTU влияет на работу сети? Какие проблемы связаны со слишком длинными кадрами? В чем состоит неэффективность коротких кадров?
13. Как коэффициент использования влияет на производительность сети Ethernet?
14. Если один вариант технологии Ethernet имеет более высокую скорость передачи данных, чем другой (например, Fast Ethernet и Ethernet), то какая из скоростей поддерживает большую максимальную длину сети?
15. Из каких соображений выбрана максимальная длина физического сегмента в стандартах Ethernet?
16. Проверьте корректность конфигурации сети Fast Ethernet (см. рисунок).



17. Укажите максимально допустимые значения MTU для:

- Ethernet;
- Token Ring;
- FDDI;
- ATM.

18. Опишите алгоритм доступа к среде для технологии Token Ring.

19. Из каких соображений выбирается максимальное время оборота маркера по кольцу?
20. Если бы вам пришлось выбирать, какую из технологий — Ethernet или Token Ring — использовать в сети вашего предприятия, какое решение вы бы приняли? Какие соображения привели бы в качестве обоснования этого решения?
21. В чем состоит сходство и различие технологий FDDI и Token Ring?
22. Какие элементы сети FDDI обеспечивают отказоустойчивость?
23. Технология FDDI является отказоустойчивой. Означает ли это, что при любом однократном обрыве кабеля сеть FDDI будет продолжать нормально работать?
24. К каким последствиям может привести двукратный обрыв кабеля в кольце FDDI?
25. Что общего в работе концентратора 100VG-AnyLAN и обычного моста?
26. Какие из ниже перечисленных пар сетевых технологий совместимы по форматам кадров и, следовательно, позволяют образовывать составную сеть без необходимости транслирования кадров;
- FDDI - Ethernet (A);
 - Token Ring — Fast Ethernet (B);
 - Token Ring - 100VG-AnyLAN (C);
 - Ethernet - Fast Ethernet (D);
 - Ethernet - 100VG-AnyLAN (E);
 - Token Ring - FDDI (F).
27. Из-за увеличения пропускной способности минимальный размер кадра в Gigabit Ethernet пришлось увеличить до 512 байт. В тех случаях, когда передаваемые данные не могут полностью заполнить поле данных кадра, оно дополняется до необходимой длины неким «заполнителем», который не несет полезной информации. Что предпринято в Gigabit Ethernet для сокращения накладных расходов, возникающих при передаче коротких данных?
28. С чем связано ограничение, известное как «правило 4 хабов»?
29. Как влияет па производительность сети пропускная способность сетевого адаптера и пропускная способность порта концентратора?
30. Имеются ли отличия в работе сетевых адаптеров, соединяющих компьютер с коммутатором, мостом, концентратором?
31. Как концентратор поддерживает резервные связи?
-
32. В соответствии с основной функцией концентратора — повторением сигнала — его относят к устройствам, работающим на физическом уровне модели OSI. Приведите примеры дополнительных функций концентратора, для выполнения которых концентратору требуется информация протоколов более высоких уровней.
33. Чем модульный концентратор отличается от стекового?
34. Почему для соединения концентраторов между собой используются специальные порты?
35. Каким образом мост или коммутатор строит свою внутреннюю таблицу?
36. Что случится, если во время работы моста или коммутатора произойдет ре-

конфигурирование сети, например будут подключены новые компьютеры?

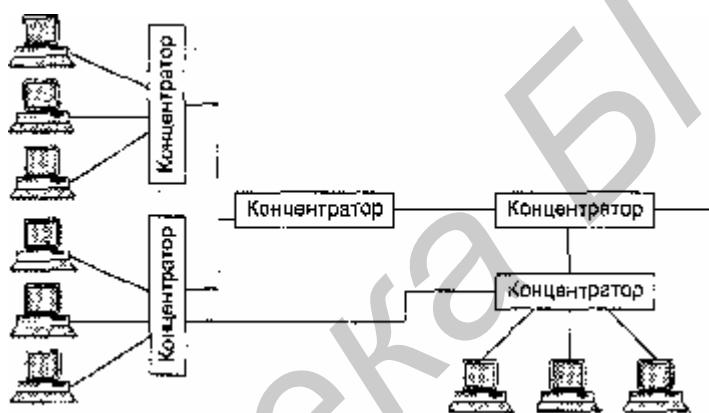
37. О чём говорит размер внутренней адресной таблицы моста? Что произойдет, если таблица переполнится?

38. Можно ли утверждать, что у любого моста скорость продвижения не выше скорости фильтрации?

39. Что нужно сделать администратору сети, чтобы мосты, не поддерживающие алгоритм Spanning Tree, правильно работали в сети с петлями?

40. Что произойдет, если в сети, построенной на концентраторах, имеются замкнутые контуры (например, такие, как на представленном ниже рисунке)?

- Сеть будет работать нормально (Л).
- Кадры не будут доходить до адресата (Б).
- В сети при передаче любого кадра будет возникать коллизия (С).
- Произойдет зацикливание кадров (Д).



41. Какие дополнительные возможности имеют мосты, поддерживающие алгоритм Spanning Tree?

42. В чём отличие между резервированием связей маршрутизаторами, с одной стороны, и мостами, поддерживающими алгоритм Spanning Tree, с другой стороны?

43. Пусть на предприятии имеются две изолированные рабочие группы, в каждой из которых имеется свой сервер. В каких случаях лучше использовать:

- два отдельных концентратора;
- два концентратора, объединенные в стек;
- один общий концентратор с большим количеством Нортон?

44. Пусть на предприятии в одном отделе установлена одноранговая сеть Windows 98, а в другом отделе — сеть NetWare с одним выделенным сервером. Каждая из сетей построена на основе одного концентратора. Как вы считаете, в каком отделе замена концентратора коммутатором может привести к существенному росту производительности? Рассмотрите следующие варианты замены концентратора коммутатором:

- концентратор имеет порты 10 Мбит/с, коммутатор имеет все порты 10 Мбит/с (А);
- концентратор имеет порты 10 Мбит/с, коммутатор имеет порты 10 Мбит/с и один порт 100 Мбит/с (В);

○ концентратор имеет порты 100 Мбит/с, коммутатор имеет все порты 100 Мбит/с (С).

45. В области сетевых технологий явно наметилась тенденция к использованию индивидуальных связей компьютеров с коммуникационными устройствами (в отличие от подключения к портам сегментов)- С чем это связано?

46. Почему полнодуплексный канал Ethernet не поддерживается в концентраторах?

47. Каким образом коммутатор может управлять потоком пакетов, поступающих от сетевых адаптеров станций сети?

48. Существуют маршрутизаторы, работающие в режиме моста на некоторых портах. Как вы думаете, можно ли создать маршрутизатор или коммутатор, который способен работать в режиме концентратора на тех же портах, на которых выполняется маршрутизация?

49. Можно ли соединить транслирующим коммутатором сегменты, в которых установлено разное максимальное значение поля данных?

50. Имеется ли специфика в использовании мостов и коммутаторов? Приведите примеры, когда замена моста коммутатором не повышает производительности сети.

51. Почему недорогие коммутаторы, выполняющие ограниченное число функций, обычно работают по быстрому алгоритму обработки пакетов «на лету», а дорогие коммутаторы, с большим числом функций, — по более медленному алгоритму буферизации пакетов?

52. Какая информация содержится в таблицах мостов/коммутаторов и маршрутизаторов?

53. Поясните определение: «Виртуальная локальная сеть - это домен распространения широковещательных сообщений».

54. В каких случаях появляется необходимость в создании виртуальных сегментов? Приведите примеры.

Часть IV
Составные сети

Библиотека БГУИР

ГЛАВА 11 Объединение сетей на основе сетевого уровня

В стандартной модели взаимодействия открытых систем ISO/OSI к сетевому уровню отнесены функции обеспечения передачи данных через сложные сети, объединенные произвольными связями и построенные на основе различных локальных и глобальных сетевых технологий.

Протоколы сетевого уровня реализуются, как правило, в виде программных модулей и выполняются па конечных узлах-компьютерах, а также на промежуточных узлах-маршрутизаторах.

Понятие «internetworking»

Ограничения мостов и коммутаторов

Создание сложной, структурированной сети, интегрирующей различные базовые технологии, может осуществляться и средствами канального уровня: для этого могут быть использованы некоторые типы мостов и коммутаторов. Мост или коммутатор разделяет сеть на сегменты, локализуя трафик внутри сегмента, что делает линии связи разделяемыми преимущественно между станциями данного сегмента. Тем самым сеть распадается на отдельные подсети, из которых могут быть построены составные сети достаточно крупных размеров.

Однако построение сложных сетей только на основе повторителей, мостов и коммутаторов имеет существенные ограничения и недостатки.

- Во-первых, в топологии получившейся сети должны отсутствовать петли. Действительно, мост/коммутатор может решать задачу доставки пакета адресату только тогда, когда между отправителем и получателем существует единственный путь. В то же время наличие избыточных связей, которые и образуют петли, часто необходимо для лучшей балансировки нагрузки, а также для повышения надежности сети за счет образования резервных путей.
- Во-вторых, логические сегменты сети, расположенные между мостами или коммутаторами, слабо изолированы друг от друга, а именно не защищены от так называемых широковещательных штормов. Если какая-либо станция посыпает широковещательное сообщение, то это сообщение передается всем станциям всех логических сегментов сети. Защита от широковещательных штормов в сетях, построенных на основе мостов и коммутаторов, имеет количественный, а не качественный характер: администратор просто ограничивает количество широковещательных пакетов, которое разрешается генерировать некоторому узлу в единицу времени. Использование же механизма виртуальных сетей, реализованного во многих коммутаторах, хотя и позволяет достаточно гибко создавать изолированные по трафику группы станций, но при этом изолирует их полностью, то есть так, что узлы одной виртуальной сети не могут взаимодействовать с узлами другой виртуальной сети.

- В-третьих, в сетях, построенных на основе мостов и коммутаторов, достаточно сложно решается задача управления трафиком на основе значения данных, содержащихся в пакете. В таких сетях это возможно только с помощью пользовательских фильтров, для создания которых администратору приходится иметь дело с двоичным представлением содержимого пакетов.
- В-четвертых, реализация транспортной подсистемы только средствами физического и канального уровней, к которым относятся мосты и коммутаторы, приводит к недостаточно гибкой, одноуровневой системе адресации: в качестве адреса назначения используется MAC-адрес, жестко связанный с сетевым адаптером.
- Наконец, возможностью трансляции протоколов канального уровня обладают далеко не все типы мостов и коммутаторов, к тому же эти возможности ограничены. В частности, в объединяемых сетях должны совпадать максимально допустимые размеры полей данных в кадрах, так как мостами и коммутаторами не поддерживается функция фрагментации кадров, Наличие серьезных ограничений у протоколов канального уровня показывает, что построение на основе средств этого уровня больших неоднородных сетей является весьма проблематичным. Естественное решение в этих случаях — это привлечение средств более высокого, сетевого уровня.

Архитектура составной сети

Основная идея введения сетевого уровня состоит в следующем. Сеть в общем случае рассматривается как совокупность нескольких сетей и называется составной сетью, или интерсетью (internetwork, или internet). Сети, входящие в составную сеть, называются подсетями (subnet), составляющими сетями или просто сетями (рис. 11.1).

Подсети соединяются между собой маршрутизаторами. Компонентами составной сети могут являться как локальные, так и глобальные сети. Внутренняя структура каждой сети на рисунке не показана, тук как она не имеет значения при рассмотрении сетевого протокола. Все узлы в пределах одной подсети взаимодействуют, используя единую для них технологию. Так, в составную сеть, показанную на рисунке, входит несколько сетей разных технологий: локальные сети Ethernet, Fast Ethernet, Token Ring, FDDI и глобальные сети frame relay, X.25, ISDN. Каждая из этих технологий способна обеспечить взаимодействие всех узлов в своей подсети, но не способна построить информационную связь между произвольно выбранными узлами, принадлежащими разным подсетям, например между узлом А и узлом В на рис. 11.1. Следовательно, для организации взаимодействия между любой произвольной парой узлов этой «большой» составной сети требуются дополнительные средства. Такие средства и предоставляет сетевой уровень.

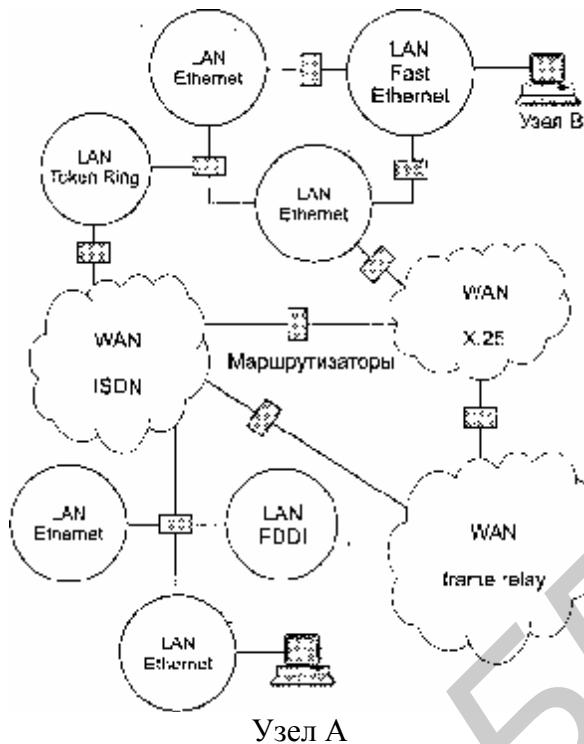


Рис. 11.1. Архитектура составной сети

Сетевой уровень выступает в качестве координатора, организующего работу всех подсетей, лежащих па пути продвижения пакета по составной сети. Для перемещения данных в пределах подсетей сетевой уровень обращается к используемым в этих подсетях технологиям.

Хотя многие технологии локальных сетей (Ethernet, Token Ring, FDDI, Fast Ethernet и др.) используют одну и ту же систему адресации узлов на основе MAC-адресов, существует немало технологий (X.25, ATM, frame relay), в которых применяются другие схемы адресации. Адреса, присвоенные узлам в соответствии с локальными технологиями подсетей, называют локальными¹. Чтобы сетевой уровень мог выполнить свою задачу, ему необходима собственная глобальная система адресации, не зависящая от способов адресации узлов в отдельных подсетях, которая позволила бы па сетевом уровне универсальным и однозначным способом идентифицировать любой узел составной сети.

(При обсуждении составной сети в таких терминах, как «локальная технология», «локальный адрес» и т. п., слово «локальный» показывает, что характеризуемый этим словом объект относится к части большой составной (глобальной) сети. При этом совсем не обязательно, что частью составной сети является непременно LAN.)

Естественным способом формирования сетевого адреса является уникальная нумерация всех подсетей составной сети и нумерация всех узлов в пределах каждой подсети. Таким образом, сетевой адрес представляет собой пару: номер сети (подсети) и номер узла.

В качестве номера узла может выступать либо локальный адрес этого узла (такая схема принята в стеке IPX/SPX), либо некоторое число, никак не связанное с локальной технологией, которое однозначно идентифицирует узел в пределах данной подсети. В первом случае сетевой адрес становится зависимым от ло-

кальных технологий, что ограничивает его применение. Например, сетевые адреса IPX/SPX рассчитаны на работу в составных сетях, объединяющих сети, в которых используются только MAC-адреса или адреса аналогичного формата. Второй подход более универсален, он характерен для стека TCP/IP. В том и другом случае каждый узел составной сети имеет наряду со своим локальным адресом еще один - универсальный сетевой адрес.

Данные, которые поступают на сетевой уровень и которые необходимо передать через составную сеть, снабжаются заголовком сетевого уровня. Данные вместе с заголовком образуют пакет. Заголовок пакета сетевого уровня имеет унифицированный формат, не зависящий от форматов кадров канального уровня тех сетей, которые могут входить в составную сеть, и несет наряду с другой служебной информацией данные о номере сети назначения этого пакета. Сетевой уровень определяет маршрут и перемещает пакет между подсетями.

Каждый раз, когда пакет сетевого уровня передается из одной сети в другую, он извлекается из кадра первой подсети (освобождается от канального заголовка этой сети) и упаковывается в кадр (снабжается новым заголовком) канального уровня следующей подсети. Информацию, на основе которой делается эта замена, предоставляют служебные поля пакета сетевого уровня. В поле адреса назначения нового кадра указывается локальный адрес следующего маршрутизатора.

ПРИМЕЧАНИЕ

Если в подсети доставка данных осуществляется средствами канального и физического уровней (как, например, в стандартных локальных сетях), то пакеты сетевого уровня упаковываются в кадры канального уровня. Если же в какой-либо подсети для транспортировки сообщений используется технология, основанная на стеках с большим числом уровней, то пакеты сетевого уровня упаковываются в блоки передаваемых данных самого высокого уровня подсети.

Если проводить аналогию между взаимодействием, разнородных сетей и перепиской людей из разных стран, то сетевая информация — это общепринятый индекс страны, добавленный к адресу письма, написанному на одном из сотни языков земного тара, например на санскрите. И даже если это письмо должно пройти через множество стран, почтовые работники которых не знают санскрита, понятный им индекс страны-адресата подскажет, через какие промежуточные страны лучше передать письмо, чтобы оно кратчайшим путем попало в Индию. А уже там работники местных почтовых отделений смогут прочитать точный адрес, идентифицирующий город, улицу, дом и индивидуума, и доставить письмо адресату, так как адрес написан на языке и в форме, принятой в данной стране.

Основным полем заголовка сетевого уровня является номер сети-адресата. В рассмотренных нами ранее протоколах локальных сетей такого поля в кадрах предусмотрено не было — предполагалось, что все узлы принадлежат одной сети. Явная нумерация сетей позволяет протоколам сетевого уровня составлять точную карту межсетевых связей и выбирать рациональные маршруты при любой их топологии, в том числе альтернативные маршруты, если они имеются, что не умеют делать мосты и коммутаторы.

Кроме номера сети заголовок сетевого уровня должен содержать и другую информацию, необходимую для успешного перехода пакета из сети одного типа в сеть другого типа. К такой информации может относиться, например:

- номер фрагмента макета, необходимый для успешного проведения операций сборки-разборки фрагментов при соединении сетей с разными максимальными размерами пакетов;
- время жизни пакета, указывающее, как долго он путешествует по интерсети, это время может использоваться для уничтожения «заблудившихся» пакетов;
- качество услуги — критерий выбора маршрута при межсетевых передачах — например, узел-отправитель может потребовать передать пакет с максимальной надежностью, возможно, в ущерб времени доставки.

Когда две или более сети организуют совместную транспортную службу, то такой режим взаимодействия обычно называют межсетевым взаимодействием (internetworking).

Принципы маршрутизации

Важнейшей задачей сетевого уровня является маршрутизация — передача пакетов между двумя конечными узлами в составной сети.

Рассмотрим принципы маршрутизации на примере составной сети, показанной на рис. 11.2. В этой сети 20 маршрутизаторов объединяют 18 сетей в общую сеть; S1, S2, …, S20 — это номера сетей. Маршрутизаторы имеют по несколько портов (минимум по два), к которым присоединяются сети. Каждый порт маршрутизатора можно рассматривать как отдельный узел сети: он имеет собственный сетевой адрес и собственный локальный адрес в той подсети, которая к нему подключена. Например, маршрутизатор под номером 1 имеет три порта, к которым подключены сети S1, S2, S3. На рисунке сетевые адреса этих портов обозначены как M1(1), M1(2) и M1(3). Порт M1(1) имеет локальный адрес в сети с номером S1, порт M1(2) — в сети S2, а порт M1(3) — в сети S3. Таким образом, маршрутизатор можно рассматривать как совокупность нескольких узлов, каждый из которых входит и свою сеть. Как единое устройство маршрутизатор не имеет ни отдельного сетевого адреса, ни какого-либо локального адреса.

ПРИМЕЧАНИЕ

Если маршрутизатор имеет блок управления (например, SNMP-управления), то этот блок обладает собственными локальным и сетевым адресами, но которым к нему обращается центральная станция управления, находящаяся где-то в составной сети.

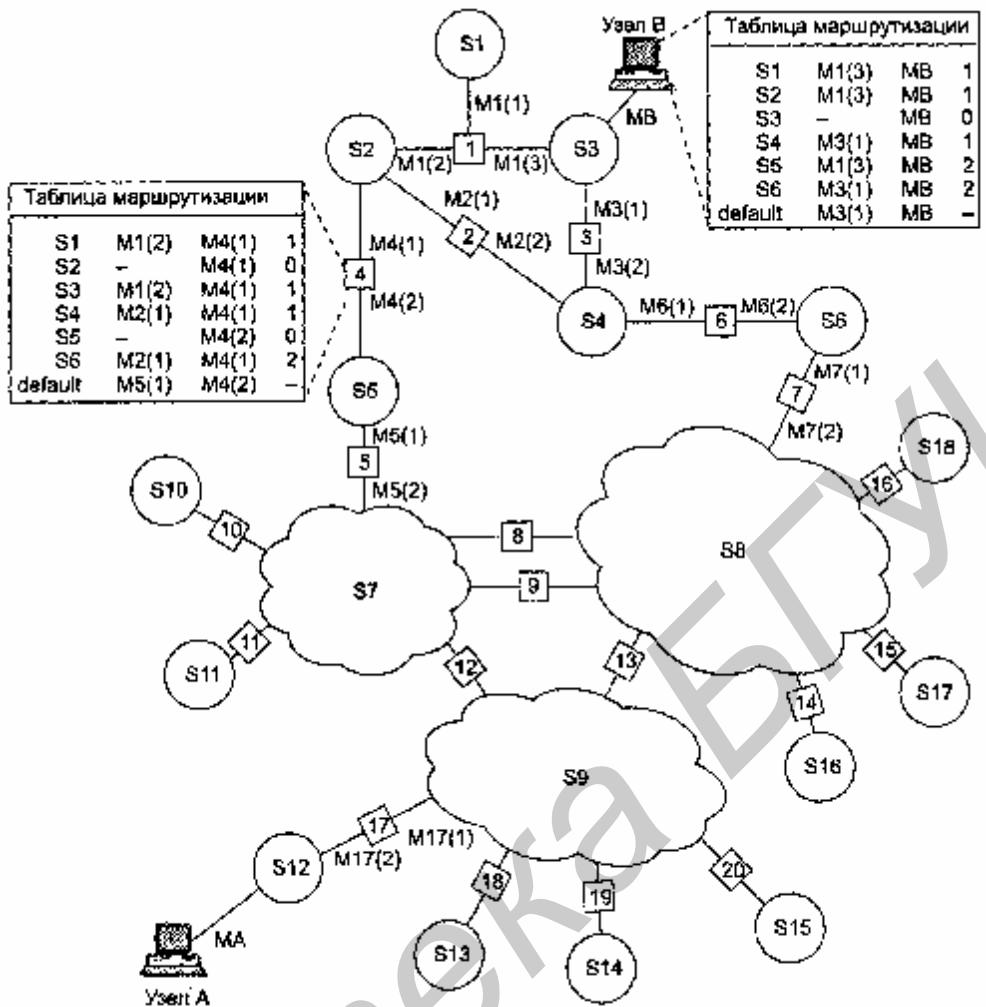


Рис. 11.2. Принципы маршрутизации в составной сети

В сложных составных сетях почти всегда существует несколько альтернативных маршрутов для передачи пакетов между двумя конечными узлами. Маршрут — это последовательность маршрутизаторов, которые должен пройти пакет от отправителя до получателя. Так, пакет, отправленный из узла А в узел В, может пройти через маршрутизаторы 17, 12, 5, 4 и 1 или маршрутизаторы 17, 13, 7, 6 и 3. Нетрудно найти еще несколько маршрутов между узлами А и В.

Задачу выбора маршрута из нескольких возможных решают маршрутизаторы, а также конечные узлы. Маршрут выбирается на основании имеющейся у этих устройств информации о текущей конфигурации сети, а также на основании указанного критерия выбора маршрута. Обычно в качестве критерия выступает задержка прохождения маршрута отдельным пакетом или средняя пропускная способность маршрута для последовательности пакетов. Часто также используется весьма простой критерий, учитывающий только количество пройденных в маршруте промежуточных маршрутизаторов (хопов).

Чтобы по адресу сети назначения можно было выбрать рациональный маршрут дальнейшего следования пакета, каждый конечный узел и маршрутизатор анализируют специальную информационную структуру, которая называется таблицей маршрутизации. Используя условные обозначения для сетевых адресов

маршрутизаторов и номеров сетей в том виде, как они приведены на рис. 11.2, посмотрим, как могла бы выглядеть таблица маршрутизации, например, в маршрутизаторе 4 (табл. 11.1).

Таблица 11.1. Таблица маршрутизации маршрутизатора 4

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
S1	M1(2)	M4(1)	1
S2		M4(1)	0 (подсоединенна)
S3	MЦ2)	M4(1)	1
S4	M2(1)	M4(1)	1
S5	—	M4(2)	0 (подсоединенна)
S6	M2(1)	M4(1)	>
Default	M5(1)	M4(2)	-

ПРИМЕЧАНИЕ

Таблица 11.1 значительно упрощена по сравнению с реальными таблицами маршрутизации, например, отсутствуют столбцы с масками, признаками состояния маршрута, временем, и течение которого действительны записи данной таблицы (их применение будет рассмотрено позже). Вместо номера сети назначения может быть указан полный сетевой адрес отдельного узла назначения. Кроме того, как уже было сказано, здесь указаны адреса сетей условного формата, не соответствующие какому-либо определенному сетевому протоколу. Тем не менее эта таблица содержит основные поля, имеющиеся в реальных таблицах при использовании конкретных сетевых протоколов, таких как IP, IPX или X.25.

В первом столбце приведенной выше таблицы перечисляются номера сетей, входящих в интэрсет. В каждой строке таблицы следом за номером сети указывается сетевой адрес следующего маршрутизатора (более точно, сетевой адрес соответствующего порта следующего маршрутизатора), на который надо направить пакет, чтобы тот передвигался по направлению к сети с данным номером по рациональному маршруту.

Когда па маршрутизатор поступает новый пакет, номер сети назначения, извлеченный из поступившего кадра, последовательно сравнивается с номерами сетей из каждой строки таблицы. Стока с совпавшим номером сети указывает, на какой ближайший маршрутизатор следует направить пакет. Например, если на какой-либо порт маршрутизатора 4 поступает пакет, адресованный в сеть S6, то из таблицы маршрутизации следует, что адрес следующего маршрутизатора — M2(1), то есть очередным этапом движения данного пакета будет движение к порту 1 маршрутизатора 2.

Поскольку пакет может быть адресован и любую сеть составной сети, может показаться, что каждая таблица маршрутизации должна иметь записи обо всех сетях, входящих в составную сеть. Но при таком подходе в случае крупной сети

объем таблиц маршрутизации может оказаться очень большим, что повлияет на время ее просмотра, потребует много места для хранения и т. п. Поэтому на практике число записей в таблице маршрутизации стараются уменьшить за счет использования специальной записи — «маршрутизатор по умолчанию» {default). Действительно, если принять во внимание топологию составной сети, то в таблицах маршрутизаторов, находящихся на периферии составной сети, достаточно записать номера сетей, непосредственно подсоединенных к данному маршрутизатору или расположенных поблизости, на тупиковых маршрутах. Обо всех же остальных сетях можно сделать в таблице единственную запись, указывающую на маршрутизатор, через который пролегает путь ко всем этим сетям. Такой маршрутизатор называется маршрутизатором по умолчанию, а вместо номера сети в соответствующей строке помещается особая запись, например Default. В нашем примере таким маршрутизатором по умолчанию для сети S5 является маршрутизатор 5, точнее его порт M5(1). Это означает, что путь из сети S5 почти ко всем сетям большой составной сети пролегает через этот порт маршрутизатора.

Перед тем как передать пакет следующему маршрутизатору, текущий маршрутизатор должен определить, на какой из нескольких собственных портов он должен поместить данный пакет. Для этого служит третий столбец таблицы маршрутизации. Еще раз подчеркнем, что каждый порт идентифицируется собственным сетевым адресом.

Некоторые реализации сетевых протоколов допускают наличие в таблице маршрутизации сразу нескольких строк, соответствующих одному и тому же адресу сети назначения. В этом случае при выборе маршрута принимается во внимание столбец «Расстояние до сети назначения». При этом под расстоянием понимается любая метрика, используемая в соответствии с заданным в сетевом пакете критерием (часто называемым классом сервиса). Расстояние может измеряться хопами, временем прохождения пакета по линиям связи, какой-либо характеристикой надежности линий связи на данном маршруте или другой величиной, отражающей качество данного маршрута по отношению к заданному критерию. Если маршрутизатор поддерживает несколько классов сервиса пакетов, то таблица маршрутов составляется и применяется отдельно для каждого вида сервиса (критерия выбора маршрута).

В табл. 11.1 расстояние между сетями измерялось хопами. Расстояние для сетей, непосредственно подключенных к портам маршрутизатора, здесь принимается равным 0, однако в некоторых реализациях отсчет расстояний начинается с 1.

Наличие нескольких маршрутов к одному узлу делают возможным передачу трафика к этому узлу параллельно по нескольким каналам связи, что повышает пропускную способность и надежность сети.

Задачу маршрутизации решают не только промежуточные узлы-маршрутизаторы, но и конечные узлы — компьютеры. Средства сетевого уровня, установленные на конечном узле, при обработке пакета должны, прежде всего, определить, направляется ли он в другую сеть или адресован какому-нибудь узлу данной сети. Если номер сети назначения совпадает с номером

данной сети, то пакету не требуется решать задачу маршрутизации. Если же номера сетей отправления и назначения не совпадают, то маршрутизация нужна. Таблицы маршрутизации конечных узлов полностью аналогичны таблицам маршрутизации, хранящимся на маршрутизаторах.

Обратимся снова к сети, представленной на рис. 11.2. Таблица маршрутизации для конечного узла В могла бы выглядеть так, как показано в табл. 11.2. Здесь МБ — сетевой адрес порта компьютера В. На основании этой таблицы конечный узел В выбирает, на какой из двух имеющихся в локальной сети S3 маршрутизаторов следует посыпать тот или иной пакет.

Таблица 11.2. Таблица маршрутизации конечного узла В

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
S1	M 1 (3)	МБ	i
S2	M1(3)	МВ	1
со		МВ	0
S4	M3(1)	МВ	1
S5	M1(3)	МВ	2
S6	M3(1)	МВ	2
Default	M3(1)	МВ	-

Конечные узлы в еще большей степени, чем маршрутизаторы, пользуются приемом маршрутизации по умолчанию. Хотя они также в общем случае имеют в своем распоряжении таблицу маршрутизации, ее объем обычно незначителен, что объясняется периферийным расположением всех конечных узлов. Конечный узел часто вообще работает без таблицы маршрутизации, имея только сведения об адресе маршрутизатора по умолчанию. При наличии одного маршрутизатора в локальной сети этот вариант — единственно возможный для всех конечных узлов. Но даже при наличии нескольких маршрутизаторов в локальной сети, когда перед конечным узлом стоит проблема их выбора, задание маршрута по умолчанию часто используется в компьютерах для сокращения объема их таблицы маршрутизации.

Ниже помещена таблица маршрутизации другого конечного узла составной сети — узла А (табл. 11.3). Компактный вид таблицы маршрутизации отражает тот факт, что все пакеты, направляемые из узла А, либо не выходят за пределы сети S12, либо непременно проходят через порт 1 маршрутизатора 17. Этот маршрутизатор и определен в таблице маршрутизации в качестве маршрутизатора по умолчанию.

Таблица 11.3. Таблица маршрутизации конечного узла А

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
S12	-	МА	0
Default	M17(1)	МА	-

Еще одним отличием работы маршрутизатора и конечного узла при выборе маршрута является способ построения таблицы маршрутизации. Если маршрутизаторы обычно автоматически создают таблицы маршрутизации, обмениваясь служебной информацией, то для конечных узлов таблицы маршрутизации часто создаются вручную администраторами и хранятся в виде постоянных файлов на дисках.

Протоколы маршрутизации

Задача маршрутизации решается на основе анализа таблиц маршрутизации, размещенных во всех маршрутизаторах и конечных узлах сети. Каким же образом происходит формирование этих таблиц? Какими средствами обеспечивается адекватность содержащейся в них информации постоянно изменяющейся структуре сети? Основная работа по созданию таблиц маршрутизации выполняется автоматически, но и возможность вручную скорректировать или дополнить таблицу тоже, как правило, предусматривается.

Для автоматического построения таблиц маршрутизации маршрутизаторы обмениваются информацией о топологии составной сети в соответствии со специальным служебным протоколом. Протоколы этого типа называются протоколами маршрутизации (или маршрутизирующими протоколами). Протоколы маршрутизации (например, RIP, OSPF, NLSP) следует отличать от собственно сетевых протоколов (например, IP, IPX). Те и другие выполняют функции сетевого уровня модели OSI — участвуют в доставке пакетов адресату через разнородную составную сеть. Но в то время как первые собирают и передают по сети чисто служебную информацию, вторые предназначены для передачи пользовательских данных, как это делают протоколы канального уровня. Протоколы маршрутизации используют сетевые протоколы как транспортное средство. При обмене маршрутной информацией пакеты протокола маршрутизации помещаются в поле данных пакетов сетевого уровня или даже транспортного уровня, поэтому с точки зрения вложенности пакетов протоколы маршрутизации формально следовало бы отнести к более высокому уровню, чем сетевой.

В том, что маршрутизаторы для принятия решения о продвижении пакета обращаются к адресным таблицам, можно увидеть их некоторое сходство с мостами и коммутаторами. Однако природа используемых ими адресных таблиц значительно различается. Вместо MAC-адресов в таблицах маршрутизации указываются номера сетей, которые соединяются в интерсеть. Другим отличием таблиц маршрутизации от адресных таблиц мостов является способ их создания. В то время как мост строит таблицу, пассивно наблюдая за проходящими через него информационными кадрами, посыпаемыми конечными узлами сети друг другу, маршрутизаторы по своей инициативе обмениваются специальными служебными пакетами, сообщая соседям об известных им сетях в интерсети, маршрутизаторах и о связях этих сетей с маршрутизаторами. Обычно учитывается не только топология связей, но и их пропускная способность и состояние. Это позволяет маршрутизаторам быстрее адаптироваться к изменениям конфигурации сети, а также правильно передавать пакеты в сетях с произвольной то-

пологией, допускающей наличие замкнутых контуров.

С помощью протоколов маршрутизации маршрутизаторы составляют карту связей сети той или иной степени подробности. На основании этой информации для каждого номера сети принимается решение о том, какому следующему маршрутизатору надо передавать пакеты, направляемые в эту сеть, чтобы маршрут оказался рациональным. Результаты этих решений заносятся в таблицу маршрутизации. При изменении конфигурации сети некоторые записи в таблице становятся недействительными. В таких случаях пакеты, отправленные по ложным маршрутам, могут «зацикливаться» и теряться. От того, насколько быстро протокол маршрутизации приводит к соответствие содержимое таблицы реальному состоянию сети, зависит качество работы всей сети.

Выводы

- Составная сеть (*internetwork*, или *internet*) — это совокупность нескольких сетей, называемых также подсетями (*subnet*), которые соединяются между собой маршрутизаторами. Организация совместной транспортной службы в составной сети называется межсетевым взаимодействием (*internetworking*).
- В функции сетевого уровня входят: передача пакетов между конечными узлами в составных сетях, выбор маршрута, согласование локальных технологий отдельных подсетей.
- Маршрут — это последовательность маршрутизаторов, которые должен пройти пакет от отправителя до получателя. Задачу выбора маршрута из нескольких возможных решают маршрутизаторы и конечные узлы на основе таблиц маршрутизации. Записи в таблицу могут вноситься вручную администратором и автоматически протоколами маршрутизации.
- Протоколы маршрутизации (например, RIP или OSPF) следует отличать от собственно сетевых протоколов (например, I? или IPX). В то время как первые собирают и передают по сети служебную информацию о возможных маршрутах, вторые предназначены для передачи пользовательских данных.
- Сетевые протоколы и протоколы маршрутизации реализуются в виде программных модулей на конечных узлах-компьютерах и на промежуточных узлах-маршрутизаторах.
- Маршрутизатор представляет собой сложное многофункциональное устройство, в задачи которого входит: построение таблицы маршрутизации, определение на ее основе маршрута, буферизация, фрагментация и фильтрация поступающих пакетов, поддержка сетевых интерфейсов. Функции маршрутизаторов могут выполнять как специализированные устройства, так и универсальные компьютеры с соответствующим программным обеспечением.

Реализация межсетевого взаимодействия средствами TCP/IP

В настоящее время стек TCP/IP является самым популярным средством организации составных сетей. На рис. 11.3 показана доля, которую составляет тот или

иной стек протоколов в общемировой сетевой базе. До 1996 года бесспорным лидером был стек IPX/SPX компании Novell, но затем картина резко изменилась — стек TCP/IP по темпам роста числа установок намного стало опережать другие стеки, а с 1998 года вышел в лидеры и в абсолютном выражении. Именно поэтому дальнейшее изучение функций сетевого уровня будет проводиться на примере стека TCP/IP.

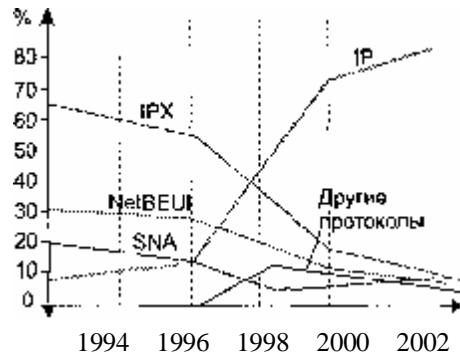


Рис. 11.3. Стек TCP/IP становится основным средством построения составных сетей

Многоуровневая структура стека TCP/IP

Так как стек TCP/IP был разработан до появления модели взаимодействия открытых систем ISO/OSI, то, хотя он также имеет многоуровневую структуру, соответствие уровней стека TCP/IP уровням модели OSI достаточно условно (рис. 11.4).

Уровни OSI	Уровни стека TCP/IP	
Прикладной (Application)	Прикладной (Application)	FTP, telnet, SMTP, E-Mail, HTTP, TFTP
Представительный (Presentation)		
Сессионный (Session)	Сессионный (Session)	
Транспортный (Transport)	Транспортный (Transport)	TCP, UDP
Сетевой (Network)	Сетевой (Network)	IP, RIP, OSPF, ICMP
Канальный (Link)	Уровень сетевого интерфейса (Network interface)	Протоколы канального уровня
Физический (Physical)		Изменение адресов

Рис. 11.4. Многоуровневая архитектура стека TCP/IP

В стеке TCP/IP определены 4 уровня. Каждый из этих уровней несет на себе некоторую нагрузку по решению основной задачи — организации надежной и производительной работы составной сети, части которой построены на основе разных сетевых технологий.

Прикладной уровень

Прикладной (application) уровень стека TCP/IP соответствует трем верхним

уровням модели OSI: прикладному, представительному и сеансовому. Он объединяет службы, предоставляемые системой пользовательским приложениям. За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и служб прикладного уровня. К ним относятся такие широко используемые протоколы, как протокол копирования файлов (File Transfer Protocol, FTP), протокол эмуляции терминала (telnet), простой протокол передачи электронной почты (Simple Mail Transfer Protocol, SMTP), протокол передачи гипертекстовой информации (Hypertext Transfer Protocol» HTTP) и многие другие.

Протоколы прикладного уровня устанавливаются на хостах¹. Прикладной уровень реализуется программными системами, построенными в архитектуре клиент-сервер. В отличие от протоколов остальных трех уровней, протоколы прикладного уровня отрабатывают логику приложений и «не интересуются» способами передачи данных по сети, они обращаются к протоколам нижних уровней как к некоторому набору инструментов. Так, клиентская часть протокола прикладного уровня для обмена сообщениями со своей серверной частью, установленной на удаленном узле составной сети, должна обратиться с запросом к нижележащему транспортному уровню.

Транспортный уровень

Транспортный (transport) уровень стека TCP/IP, называемый также основным уровнем, может предоставлять вышележащему уровню два типа сервиса:

- гарантированная доставка — протокол управления передачей (Transmission Control Protocol, TCP);
- доставка «по возможности» («best effort») — протокол пользовательских дейтаграмм (User Datagram Protocol, UDP).

Для того чтобы обеспечить надежную доставку данных, протокол TCP предусматривает установление логического соединения, это позволяет ему нумеровать пакеты, подтверждать их прием квитанциями, в случае потери организовывать повторные передачи, распознавать и уничтожать дубликаты, доставлять прикладному уровню пакеты в том порядке, в котором они были отправлены. Этот протокол позволяет равноранговым объектам на компьютере-отправителе и компьютере-получателе поддерживать обмен данными в дуплексном режиме. (В Интернете конечный узел традиционно называют хостом, а маршрутизатор — шлюзом. В этой главе мы будем также придерживаться такой терминологии.)

TCP позволяет без ошибок доставить сформированный на одном из компьютеров поток байтов в любой другой компьютер, входящий в составную сеть. TCP делит поток байтов на части — сегменты и передает их нижележащему уровню межсетевого взаимодействия. После того как эти сегменты будут доставлены средствами уровня межсетевого взаимодействия в пункт назначения, протокол TCP снова соберет их в непрерывный поток байтов.

Второй протокол этого уровня — UDP является простейшим дейтаграммным протоколом, который используется в том случае, когда задача надежного обме-

на данными либо вообще не ставится, либо решается средствами более высокого уровня — системными прикладными службами или пользовательскими приложениями.

В функции протоколов транспортного уровня TCP и UDP входит также выполнение роли связующего звена между прилегающими к ним прикладным уровнем и уровнем межсетевого взаимодействия. От прикладного протокола транспортный уровень принимает задание на передачу данных с тем или иным качеством, а после выполнения рапортует об этом прикладному уровню. К нижележащему уровню межсетевого взаимодействия протоколы TCP и UDP обращаются как к своего рода инструменту, не очень надежному, но способному перемещать пакет в свободном и рискованном путешествии по составной сети. Протоколы TCP и UDP, так же как и протоколы прикладного уровня, устанавливаются на хостах.

Уровень межсетевого взаимодействия

Уровень межсетевого взаимодействия (internet), называемый также сетевым (network) уровнем, является стержнем всей архитектуры TCP/IP. Именно этот уровень, функции которого соответствуют сетевому уровню модели OSI, обеспечивает перемещение пакетов в пределах всей составной сети. Протоколы уровня межсетевого взаимодействия поддерживают интерфейсы с вышележащим транспортным уровнем, получая от него запросы на передачу данных по составной сети.

Основным протоколом межсетевого уровня является межсетевой протокол (Internet Protocol, IP). В его задачу входит продвижение пакета между подсетями — от одного пограничного маршрутизатора до другого, до тех пор, пока пакет не попадет в сеть назначения. В отличие от протоколов прикладного и основного уровней протокол IP устанавливается не только на хостах, но и на всех шлюзах.

Протокол IP — это дейтаграммный протокол, работающий без установления соединений по принципу «по возможности», в соответствии с которым он не берет на себя ответственность за доставку пакета до узла назначения. Если же по каким-то причинам пакет теряется (например, из-за переполнения буфера), протокол IP не пытается повторить его передачу. Максимум на что он способен — послать уведомление о потере пакета узлу-отправителю.

Протокол IP изначально проектировался как хорошо масштабируемое средство передачи пакетов в составных сетях, состоящих из большого количества сетей, объединенных как локальными, так и глобальными связями. Учитывая, что между двумя узлами сети может пролегать несколько возможных путей, задача перемещения данных в составной сети включает задачу прокладки и выбора маршрутов. Протоколы, связанные с составлением и модификацией таблиц маршрутизации, такие как протоколы сбора маршрутной информации RIP (Routing Internet Protocol) и OSPF (Open Shortest Path First), также относятся к уровню межсетевого взаимодействия.

На этом же уровне работает и протокол межсетевых управляющих сообщений

(Internet Control Message Protocol, ICMP), предназначенный для обмена информацией об ошибках между маршрутизаторами сети и узлом-источником пакета. С помощью специальных пакетов ICMP сообщает о невозможности доставки пакета, о превышении времени жизни или продолжительности сборки пакета из фрагментов, об аномальных величинах параметров, об изменении маршрута пересылки и типа обслуживания, о состоянии системы и т. п.

Выбор последовательности шлюзов, через которые надо передавать пакет, чтобы он дошел до места назначения, — это задача протоколов уровня межсетевого взаимодействия. А вот перемещение пакета между соседними шлюзами в пределах каждой из встречающихся на пути пакета подсетей — это уже задача локальной (то есть использующейся в каждой из подсетей) технологии. Каждый раз, когда требуется воспользоваться локальными средствами доставки пакета в пределах подсети, протокол IP обращается к нижележащему уровню межсетевых интерфейсов.

Уровень сетевых интерфейсов

Идеологическим отличием архитектуры стека TCP/IP от многоуровневой организации других стеков является интерпретация функций самого нижнего уровня — уровня сетевых интерфейсов (network interface). Так, например, нижние уровни модели OSI (канальный и физический) нагружены функциями по доступу к среде передачи, формированию кадров и согласованию уровней электрических сигналов, кодированию и синхронизации и другими весьма конкретными действиями, составляющими суть таких протоколов обмена данными, как Ethernet, Token Ring и многих других.

У нижнего уровня стека TCP/IP задача существенно проще — он отвечает только за организацию интерфейса с частными технологиями подсетей. Перемещение пакета IP можно рассматривать как последовательность «прыжков» от одного шлюза до другого. Каждый раз на очередном шлюзе в результате работы протоколов межсетевого уровня определяется сетевой адрес следующего по маршруту шлюза. Чтобы добраться до него, надо пересечь некоторую подсеть, для этого протоколы TCP/IP должны обратиться к транспортным средствам данной промежуточной подсети. Упрощенно задача обеспечения интерфейса между двумя технологиями сводится, во-первых, к определению способа упаковки (инкапсуляции) пакета IP в единицу передаваемых данных промежуточной сети, а во-вторых, к определению способа преобразования сетевого адреса следующего шлюза в новый тип адреса, который принят для адресации узлов в технологии данной промежуточной сети.

Такой подход делает составную сеть TCP/IP открытой для включения в себя любой сети, какую бы внутреннюю технологию передачи данных эта сеть не использовала. Для каждой технологии, включаемой в составную сеть подсети, должны быть разработаны собственные интерфейсные средства. Отсюда следует, что этот уровень нельзя определить раз и навсегда.

К таким интерфейсным средствам относятся протоколы инкапсуляции пакетов

IP уровня межсетевого взаимодействия в кадры технологий LAN. Например, документ RFC 1042 определяет способы инкапсуляции пакетов IP в кадры технологий IEEE 802. Для этих целей должен использоваться заголовок LLC/SNAP, причем в поле Type заголовка SNAP должен быть указан код 0x0800. Только для протокола Ethernet в RFC 1042 сделано исключение — помимо заголовка LLC/SNAP разрешается использовать кадр Ethernet DIX, не имеющий заголовка LLC, зато имеющий поле Type. В сетях Ethernet предпочтительным является инкапсуляция пакета IP в кадр Ethernet DIX. Интерфейс также обеспечивается протоколом ARP, который служит для преобразования сетевых адресов в MAC-адреса.

Заметим, что сеть может быть включена в составную сеть независимо от того, каким количеством уровней описывается используемая в ней технология. Так, например, хотя перемещение данных в сети X.25 обеспечивают протоколы физического, канального и сетевого уровней (в терминологии OSI), стек TCP/IP рассматривает сеть X.25 наравне с другими технологиями в качестве средства транспортировки пакетов IP между двумя пограничными шлюзами. Уровень сетевых интерфейсов обычным образом предоставляет для этой технологии способ инкапсуляции пакета IP в пакет X.25, а также средства преобразования сетевых адресов IP в адреса X.25.

Уровень сетевых интерфейсов в стеке TCP/IP не регламентируется, но он поддерживает все популярные технологии физического и канального уровней: для локальных сетей это Ethernet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, для глобальных сетей — протоколы соединений «точка-точка» SLIP и PPP, протоколы территориальных сетей с коммутацией пакетов X.25, frame relay. Разработана также специальная спецификация, определяющая использование технологии ATM.

Обычно при появлении новой технологии локальных или глобальных сетей она быстро включается в стек TCP/IP путем разработки соответствующего документа RFC, определяющего метод инкапсуляции пакетов IP в ее кадры (спецификация RFC 1577, определяющая работу IP через сети ATM, появилась в 1994 году вскоре после принятия основных стандартов этой технологии).

Сетезависимые и сетенезависимые уровни стека TCP/IP

Рассматривая многоуровневую архитектуру TCP/IP, можно выделить в ней, подобно архитектуре OSI, уровни, функции которых зависят от конкретной технической реализации сети, и уровни, функции которых ориентированы на работу с приложениями (рис. 11.5).

Протоколы прикладного уровня стека TCP/IP работают па компьютерах, выполняющих приложения пользователей. Даже полная смена сетевого оборудования в общем случае не должна влиять на работу приложений, если они получают доступ к сетевым возможностям через протоколы прикладного уровня.

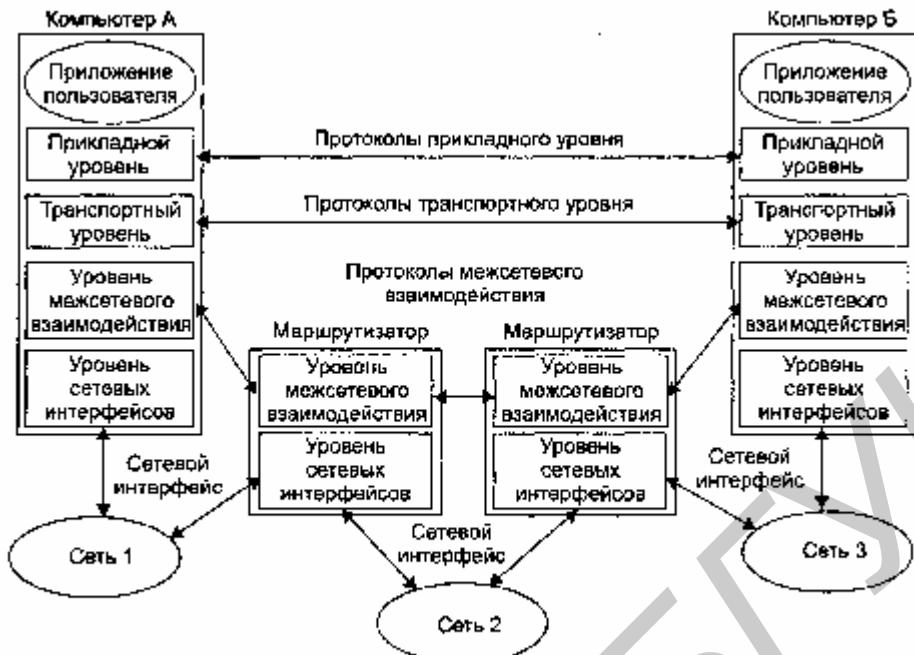


Рис. 11.5. Сетезависимые и сетенезависимые уровни стека TCP/IP

Протоколы транспортного уровня зависят от сети уже в большей степени, так как они реализуют интерфейс к уровням, непосредственно организующим передачу данных по сети. Однако подобно протоколам прикладного уровня, протоколы транспортного уровня устанавливаются только на конечных узлах. Протоколы двух нижних уровней являются сетезависимыми, программные модули протоколов межсетевого уровня и уровня сетевых интерфейсов устанавливаются на всех хостах и шлюзах.

Каждый коммуникационный протокол оперирует с некоторой единицей передаваемых данных. Названия этих единиц иногда закрепляются стандартом, а чаще просто определяются традицией. В стеке TCP/IP за многие годы его существования образовалась устоявшаяся терминология в этой области (рис. 11.6). Потоком (stream) называют данные, поступающие от приложений на вход протоколов транспортного уровня TCP и UDP.

Протокол TCP «нарезает» из потока данных сегменты (segment).

Единицу данных протокола UDP часто называют дейтаграммой, или дейтаграммой (datagram).

Дейтаграмма — это общее название для единиц данных, которыми оперируют протоколы без установления соединений. К таким протоколам относится и протокол межсетевого взаимодействия IP, поэтому его единицу данных также называют дейтаграммой. Однако очень часто используется и другой термин — пакет (packet) IP.

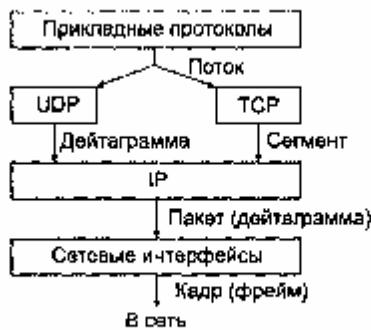


Рис. 11.6. Терминология в TCP/IP

В стеке TCP/IP принято называть кадрами, или фреймами (frame), единицы данных любых технологий, в которые упаковываются пакеты IP для последующей переноски их через подсети составной сети. При этом не имеет значения, какое название используется для этой единицы данных в локальной технологии подсети. Для TCP/IP фреймом является и кадр Ethernet, и ячейка ATM, и пакет X.25, так как все они выступают в качестве контейнера, в котором пакет IP переносится через составную сеть.

Выводы

- Наибольшее распространение при построении составных сетей в последнее время получил стек TCP/IP. Стек TCP/IP имеет 4 уровня: прикладной, транспортный, уровень межсетевого взаимодействия и уровень сетевых интерфейсов. Соответствие уровней стека TCP/IP уровням модели OSI достаточно условно.
- Прикладной уровень объединяет все службы, предоставляемые системой пользовательским приложениям: традиционные сетевые службы типа telnet, FTP, TFTP, DNS, SNMP, а также сравнительно новые, такие, например, как протокол передачи гипертекстовой информации HTTP.
- На транспортном уровне стека TCP/IP, называемом также основным, функционируют протоколы TCP и UDP. Протокол управления передачей TCP решает задачу обеспечения надежной информационной связи между двумя конечными узлами. Дейтаграммный протокол UDP используется как экономичное средство связи уровня межсетевого взаимодействия с прикладным уровнем.
- Уровень межсетевого взаимодействия реализует концепцию коммутации пакетов в режиме без установления соединений. Основными протоколами этого уровня являются дейтаграммный протокол IP и протоколы маршрутизации (RIP, OSPF, BGP и др.). Вспомогательную роль выполняют протокол межсетевых управляющих сообщений ICMP и протокол группового управления IGMP.
- Протоколы уровня сетевых интерфейсов обеспечивают интеграцию в составную сеть других сетей. Этот уровень не регламентируется, но поддерживает все популярные стандарты физического и канального уровней: для локальных сетей — Ethernet, Token Ring, FDDI и т. д., для глобальных сетей — X.25, frame relay, PPP, ISDN и т. д. К этому уровню часто относят и протокол разрешения

адресов ARP.

□ В стеке TCP/IP для именования единиц передаваемых данных на разных уровнях используют разные названия: поток, сегмент, дейтаграмма, пакет, кадр.

Библиотека БГУИР

ГЛАВА 12 Адресация в IP-сетях

Принятый в IP-сетях способ адресации узлов в немалой степени способствует масштабируемости данной технологии, которая позволяет однозначно идентифицировать миллионы сетевых интерфейсов (вспомним хотя бы Интернет с его многомиллионной армией пользователей). Однако чтобы обеспечить такую возможность в технологию TCP/IP, пришлось включить целый ряд специальных механизмов и протоколов.

Типы адресов стека TCP/IP

В стеке TCP/IP используются три типа адресов:

- локальные, или аппаратные, адреса, используемые для адресации узлов в пределах подсети;
- сетевые, или IP-адреса, используемые для однозначном идентификации узлов в пределах всей составной сети;
- доменные имена - символьные идентификаторы узлов, к которым часто обращаются пользователи.

В общем случае сетевой интерфейс может иметь одновременно один или несколько локальных адресов и один или несколько сетевых адресов, а также одно или несколько доменных имен.

Итак, аппаратный (локальный) адрес идентифицирует узел в пределах подсети. Если подсеть использует одну из базовых технологий LAN — Ethernet, FDDI, Token Ring, — то для доставки данных любому узлу такой подсети достаточно указать MAC-адрес. Таким образом, в этом случае аппаратным адресом является MAC-адрес.

В составную сеть TCP/IP могут входить подсети, построенные на основе более сложных технологий, к примеру, технологии IPX/SPX. Эта сеть сама может быть разделена на подсети, и, так же как IP-сеть, она идентифицирует свои узлы аппаратными и сетевыми IPX-адресами. Но поскольку для составной сети TCP/IP составная сеть IPX/SPX является обычной подсетью, в качестве аппаратных адресов узлов этой подсети выступают те адреса, которые однозначно определяют узлы в данной подсети, а такими адресами являются IPX-адреса. Аналогично, если в составную сеть включена сеть X.25, то локальными адресами для протокола IP соответственно будут адреса X.25.

ВНИМАНИЕ

Используемое здесь определение «локальный» может пониматься неоднозначно. Как уже было сказано, «локальный» в контексте TCP/IP означает «действующий не во всей составной сети, а лишь в пределах подсети». Именно в таком смысле понимается здесь «локальная технология» (технология, на основе которой построена подсеть), «локальный адрес» (адрес, который используется некоторой локальной технологией для адресации узлов в пределах подсети). Напомним, что в качестве подсети («локальной сети») может выступать есть, построенная как на основе глобальной (WAN) технологии, например, X.25,

Frame Relay и др., так и на основе локальной (LAN) технологии, например Ethernet, FDDI и др. В аббревиатуре LAN определение «local» характеризует особенности технологии, ограничивающие ее небольшими расстояниями, и никак не связано с архитектурой составной сети.

Сложности могут возникнуть и при интерпретации определения «аппаратный». В данном случае термин «аппаратный» подчеркивает концептуальное пред ста плен и е разработчиков стека TCP/IP о подсети как о некотором вспомогательном аппаратном средстве, единственной функцией которого является перемещение пакета IP через подсеть до ближайшего шлюза. И не важно, что нижележащая локальная технология может быть достаточно сложной, все ее сложности игнорируются технологией TCP/IP.

IP-адреса представляют собой основной тип адресов, на основании которых сетевой уровень передает пакеты между сетями. Эти адреса состоят из 4 байт, например 109.26.17.100, IP-адрес назначается администратором при конфигурировании компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно либо назначен по рекомендации специального подразделения Интернета (Internet Network Information Center, InterNIC), если сеть должна работать как составная часть Интернета. Обычно поставщики услуг Интернета получают диапазоны адресов у подразделений InterNIC, а затем распределяют их между своими абонентами. Номер узла в протоколе IP назначается независимо от локального адреса узла. Маршрутизатор по определению входит сразу в несколько сетей, поэтому каждый порт маршрутизатора имеет собственный IP-адрес. Конечный узел также может входить в несколько IP-сетей. В этом случае компьютер должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Символьные имена в IP-сетях называются доменными и строятся по иерархическому признаку. Составляющие полного символьного имени в IP-сетях разделяются точкой и перечисляются в следующем порядке: сначала простое имя хоста, затем имя группы узлов (например, имя организации), затем имя более крупной группы (поддомена) и так до имени домена самого высокого уровня (например, домена, объединяющего организации по географическому принципу: RU — Россия, UK — Великобритания, SU — США). Примером доменного имени может служить имя base2.sales.zil.ru. Между доменным именем и IP-адресом узла нет никакой функциональной зависимости, поэтому единственный способ установления соответствия — это таблица. В сетях TCP/IP используется специальная распределенная служба доменных имен (Domain Name System, DNS), которая устанавливает это соответствие на основании создаваемых администраторами сети таблиц соответствия. Поэтому доменные имена называют также DNS-именами.

Формы записи IP-адреса

IP-адрес имеет длину 4 байта (32 бита) и состоит из двух логических частей —

номера сети и номера узла в сети.

Наиболее употребляемой формой представления IP-адреса является запись в виде четырех чисел, представляющих значения каждого байта в десятичной форме и разделенных точками, например:

128.10.2.30

Этот же адрес может быть представлен в двоичном формате;

10000000 00001010 00000010 00011110

А также в шестнадцатеричном формате:

80.0A.02.ID

Заметим, что запись адреса не предусматривает специального разграничительного знака между номером сети и номером узла. Каким образом маршрутизаторы, на которые поступают пакеты, выделяют из адреса назначения номер сети, чтобы по нему определить дальнейший маршрут? Какая часть из 32 бит, отведенных под IP-адрес, относится к номеру сети, а какая — к номеру узла? Можно предложить несколько вариантов решения этой проблемы. Простейший из них состоит в том, что все 32-битовое поле адреса заранее делится на две части не обязательно равной, но фиксированной длины, в одной из которых всегда будет размещаться номер сети, а в другой — номер узла. Решение очень простое, но хорошее ли? Поскольку поле, которое отводится для хранения номера узла, имеет фиксированную длину, все сети будут иметь одинаковое максимальное число узлов. Если, например, под номер сети отвести один первый байт, то все адресное пространство распадется на сравнительно небольшое (2^8) число сетей огромного размера (2^{24} узлов). Если границу передвинуть дальше вправо, то сетей станет больше, но все равно все они будут одинакового размера. Очевидно, что такой жесткий подход не позволяет дифференцированно подходить к потребностям отдельных предприятий и организаций. Именно поэтому такой способ структуризации адреса и не нашел применения.

Второй подход основан на использовании маски, которая позволяет максимально гибко устанавливать границу между номером сети и номером узла. В данном случае маска — это число, которое используется в паре с IP-адресом; двоичная запись маски содержит последовательность единиц в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети. Поскольку номер сети является цельной частью адреса, единицы в маске также должны представлять непрерывную последовательность. Граница между последовательностью единиц и последовательностью нулей в маске соответствует границе между номером сети и номером узла в IP-адресе. При таком подходе адресное пространство можно представить как совокупность множества сетей разного размера.

И, наконец, традиционный способ решения данной проблемы заключается в использовании классов. Этот способ представляет собой компромисс по отношению к двум выше описанным: размеры сетей хотя и не являются произвольными, как при использовании масок, но и не являются одинаковыми, как при установлении фиксированных границ. Вводится несколько классов сетей, и для каждого класса определены свои размеры.

Классы IP-адресов

Традиционная схема деления IP-адреса на номер сети и номер узла основана на понятии класса, который определяется значениями нескольких первых битов адреса. Именно потому, что первый байт адреса 185.23.4-1.206 попадает в диапазон 128-191, мы можем сказать, что этот адрес относится к классу В, а значит, номером сети являются первые два байта IP-адреса, дополненные двумя нулевыми байтами — 185.23.0.0, а номером узла — дна младшие байты, дополненные с начала двумя нулевыми байтами — 0.0.44.206.

Принадлежность IP-адреса к классу определяется значениями первых битов адреса. На рис. 12.1 показана структура IP-адресов разных классов.

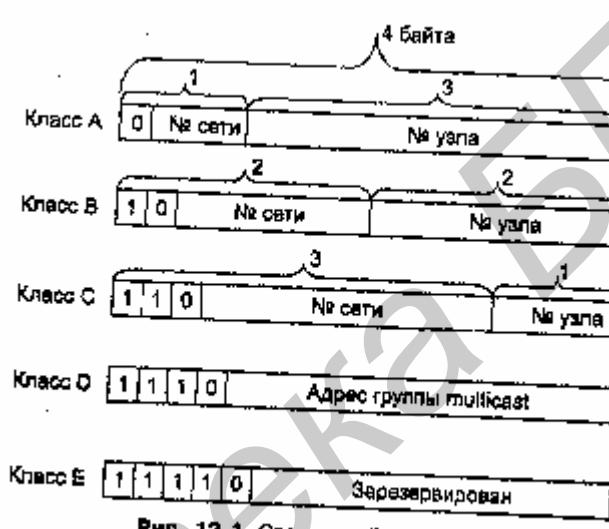


Рис. 12.1. Структура IP-адресов

Если адрес начинается с 0, то этот адрес относится к классу А, в котором под номер сети отводится один байт, а остальные три байта интерпретируются как номер узла в сети. Сети, имеющие поморы и диапазоне от 1 (00000001) до 126 (01111110), называются сетями класса А. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем будет сказано ниже.) Сетей класса А немного, зато количество узлов в них может достигать 2^{24} то есть 16777216 узлов.

Если первые два бита адреса равны 10, то адрес относится к классу В. В адресах класса В под номер сети и под номер узла отводится по два байта. Сети, имеющие номера в диапазоне от 128.0 (1000000000000000) до 191.255(10111111 11111111), называются сетями класса В. Таким образом, сетей класса В больше, чем сетей класса А, по размеры их меньше, максимальное количество узлов в них составляет 2^{16} (65536).

Если адрес начинается с последовательности битов ПО, то это адрес класса С. В этом случае под номер сети отводится 24 бита, а под номер узла — 8 бит. Сети класса С наиболее распространены, но число узлов к ним ограничено значением 2^8 (256) узлов.

Еще два класса адресов D и E не связаны непосредственно с сетями.

Если адрес начинается с последовательности 1110, то он является адресом

класса D и обозначает особый, групповой адрес (multicast). Групповой адрес идентифицирует группу узлов (сетевых интерфейсов), которые в общем случае могут принадлежать разным сетям. Интерфейс, входящий в группу, получает наряду с обычным индивидуальным IP-адресом еще один групповой адрес. Если при отправке пакета в качестве адреса назначения указан адрес класса О, то такой пакет должен быть доставлен всем узлам, которые входят в группу. Если адрес начинается с последовательности 11110, то это значит, что данный адрес относится к классу Е. Адреса этого класса зарезервированы для будущих применений.

В табл. 12.1 приведены диапазоны номеров сетей и максимальное число узлов, соответствующих каждому классу сетей.

Таблица 12.1. Характеристики адресов разного класса

Класс	Первые биты	Наименьший номер сети	Наибольший номер сети	Максимальное число узлов в сети
Л	0	1.0.0.0	126.0.0.0	2^{24}
В	10	128.0.0.0	191.255.0.0	2^{16}
С	110	192.0.1.0	223.255.255.0	2^8
П	1110	224.0.0.0	239.255.255.255	Multicast
Е	11110	240.0.0.0	247.255.255.255	Зарезервирован

Большие сети получают адреса класса А, средние — класса В, а небольшие — класса С.

Особые IP-адреса

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов.

- Если весь IP-адрес состоит только из двоичных нулей, то он обозначает адрес того узла, который сгенерировал этот пакет (этот режим используется только в некоторых сообщениях ICMP).
- Если в поле номера сети стоят только пули, то по умолчанию считается, что узел назначения принадлежит той же самой сети, что и узел, который отправил пакет.
- Если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассыпаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast). Ограничность в данном случае означает, что пакет не выйдет за границы маршрутизатора ни при каких условиях.
- Если в поле номера узла назначения стоят только единицы, то пакет, имеющий такой адрес, рассыпается всем узлам сети с заданным номером сети. Например, пакет с адресом 192.190.21.255 доставляется всем узлам сети 192.190.21.0. Такая рассылка называется широковещательным сообщением (broadcast).

Специальные адреса, состоящие из последовательностей нулей, могут быть ис-

пользованы только в качестве адреса отправителя, а адреса, состоящие из последовательностей единиц, — только в качестве адреса получателя.

При назначении адресов конечным узлам и маршрутизаторам необходимо учитывать те ограничения, которые вносятся особым назначением некоторых IP-адресов. Так, ни номер сети, ни номер узла не может состоять только из одних двоичных единиц или только из одних двоичных нулей. Отсюда следует, что максимальное количество узлов, приведенное в таблице для сетей каждого класса, на практике должно быть уменьшено на 2. Например, в адресах класса С под номер узла отводится 8 бит, которые позволяют задать 256 номеров: от 0 до 255. Однако на практике максимальное число узлов в сети класса С не может превышать 254, так как адреса 0 и 255 имеют специальное назначение. Из этих же соображений следует, что конечный узел не может иметь адрес типа 98.255.255.255, поскольку номер узла в этом адресе класса А состоит из одних двоичных единиц.

Особый смысл имеет IP-адрес, первый октет которого равен 127. Он используется для тестирования программ и взаимодействия процессов в пределах одной машины. Когда программа посыпает данные по IP-адресу 127.0.0.1, то образуется как бы «петля». Данные не передаются по сети, а возвращаются модулям верхнего уровня, как только что принятые. Поэтому в IP-сети запрещается присваивать сетевым интерфейсам IP-адреса, начинающиеся с числа 127. Этот адрес имеет название *loopback*. Можно отнести адрес 127.0.0.0 к внутренней сети модуля маршрутизации узла, а адрес 127.0.0.1 — к адресу этого модуля на внутренней сети. На самом деле любой адрес сети 127.0.0.0 служит для обозначения своего модуля маршрутизации, а не только 127.0.0.1, например 127.0.0.3.

ПРИМЕЧАНИЕ

Необходимость в адресе *loopback* возникает, например, когда на одном компьютере работает и клиентская, и серверная части некоторого сетевого приложения. Обе программные части данного приложения спроектированы в расчете на то, что они будут обмениваться сообщениями по сети. Но какой же IP-адрес они должны использовать для этого? Адрес сетевого интерфейса компьютера, на котором они установлены? Но это приводит к избыточным передачам пакетов в сеть. Экономичным решением является использование внутреннего адреса 127.0.0.0.

В протоколе IP нет понятия широковещательности в том смысле, в котором оно используется в протоколах канального уровня локальных сетей, когда данные

должны быть доставлены абсолютно всем узлам. Как ограниченный широковещательный IP-адрес, так и широковещательный IP-адрес имеют пределы распространения в интэрнете — они ограничены либо сетью, к которой принадлежит узел-источник пакета, либо сетью, номер которой указан в адресе назначения. Поэтому деление сети с помощью маршрутизаторов на части локализует широковещательный шторм пределами одной из подсетей просто потому, что нет способа адресовать пакет одновременно всем узлам всех сетей составной сети.

Уже упоминавшаяся форма группового IP-адреса — multicast — означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Один и тот же узел может входить в несколько групп. Члены какой-либо группы multicast не обязательно должны принадлежать одной сети. В общем случае они могут распределяться по различным сетям, расстояние между которыми измеряется произвольным количеством хопов. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом.

Основное назначение адресов multicast — распространение информации по схеме «один ко многим». Хост, желающий передать одну и ту же информацию многим абонентам, с помощью специального протокола IGMP (Internet Group Management Protocol) сообщает о создании в сети новой мультивещательной группы с определенным адресом. Маршрутизаторы, поддерживающие мультивещательность, распространяют информацию о создании новой группы в сетях, подключенных к портам этого маршрутизатора. Хосты, которые хотят присоединиться к вновь создаваемой мультивещательной группе, сообщают об этом своим локальным маршрутизаторам и те передают эту информацию хосту, инициатору создания новой группы.

Чтобы маршрутизаторы могли автоматически распространять пакеты с адресом multicast по составной сети, необходимо использовать в конечных маршрутизаторах модифицированные протоколы обмена маршрутной информацией, такие как, например, MOSPF (Multicast OSPF), multicast-аналог OSPF.

Групповая адресация предназначена для экономичного распространения в Интернете или крупной корпоративной сети аудио- или видеопрограмм, предназначенных сразу большой аудитории слушателей или зрителей. Если такие средства найдут широкое применение (сейчас они представляют в основном небольшие экспериментальные островки в общем Интернете), то Интернет сможет создать серьезную конкуренцию радио и телевидению.

Использование масок при IP-адресации

Снабжая каждый IP-адрес маской, можно отказаться от понятий классов адресов и сделать систему адресации более гибкой. Например, если приведенный в качестве примера в разделе «Классы IP-адресов» адрес 185.23.44.206 ассоциировать с маской 255.255.255.0, то номером сети будет 185.23.44.0, а не 185.23.0.0, как это определено системой классов.

В масках количество единиц в последовательности, определяющей границу но-

мера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты. Пусть, например, для IP-адреса 129.64.134.5 указана маска 255.255.128.0, то есть в двоичном виде IP-адрес 129.64.134.5 выглядит так:

10000001. 01000000. 10800110. 00000101

А маска 255.255.128.0 так:

11111111. 11111111. 10000000. 00000000

Если игнорировать маску, то в соответствии с системой классов адрес 129.64.134.5 относится к классу В, а значит, номером сети являются первые два байта — 129.64.0.0, а номером узла - 0.0.134.5.

Если же использовать для определения границы номера сети маску, то 17 последовательных двоичных единиц в маске 255.255.128.0, «наложенные» на IP-адрес, делят его на следующие две части.

	№ сети	№ узла
IP-адрес 129.64.134.5	10000001.01000000. 1	0000110.00000101
Маска 255.255.128.0	11111111. 11111111. 1	0000000. 00000000

В десятичной форме записи номер сети — 129.64.128.0, а номер узла — 0.0.6.5.

Для стандартных классов сетей маски имеют следующие значения:

- класс А - 1111111. 00000000. 00000000. 00000000 (255.0.0.0);
- класс В - 11111111. 11111111. 00000000. 00000000 (255.255.0.0);
- класс С - 11111111. 11111111. 11111111. 00000000 (255.255.255.0).

ПРИМЕЧАНИЕ

Для записи масок используются и другие форматы, например, удобно интерпретировать значение маски, записанной в шестнадцатеричном коде: FF.FF.00.00 — маска для адресов класса В. Часто встречается и такое обозначение 185.23.44.206/16 — эта запись говорит о том, что маска для этого адреса содержит 16 единиц или что в указанном IP-адресе под номер сети отведено 16 двоичных разрядов.

Механизм масок широко распространен в IP-маршрутизации, причем маски могут использоваться для самых разных целей. С их помощью администратор может разбивать свою сеть на несколько других, не требуя от поставщика услуг дополнительных номеров сетей (операция subnetting). На основе этого же механизма поставщики услуг могут объединять адресные пространства нескольких сетей путем введения так называемых « префиксов » с целью уменьшения объема таблиц маршрутизации и повышения за счет этого производительности маршрутизаторов — такая операция называется supemetting.

Выводы

- В стеке TCP/IP используются три типа адресов: локальные (называемые также аппаратными), IP-адреса и символьные доменные имена. Все эти типы адресов присваиваются узлам составной сети независимо друг от друга.

- IP-адрес имеет длину 4 байта и состоит из номера сети и номера узла. Для определения границы, отделяющей номер сети от номера узла, реализуются два подхода. Первый основан на понятии класса адреса, второй — на использовании масок.
- Класс адреса определяется значениями нескольких первых битов адреса. В адресах класса А под номер сети отводится один байт, а остальные три байта — под номер узла, поэтому они используются в самых больших сетях. Для небольших сетей больше подходят адреса класса С, в которых номер сети занимает три байта, а для нумерации узлов может быть использован только один байт. Промежуточное положение занимают адреса класса В.
- Другой способ определения, какая часть адреса является номером сети, а какая номером узла, основан на использовании маски. Маска — это число, которое используется в паре с IP-адресом; двоичная запись маски содержит единицы в тех разрядах, которые в IP-адресе должны интерпретироваться как номер сети.

Порядок назначения IP-адресов

У каждой подсети в пределах составной сети должен быть собственный уникальный номер, следовательно, процедура распределения номеров должна быть централизованной. Аналогично, узлы должны быть однозначно пронумерованы в пределах каждой из подсетей, отсюда следует, что централизованный характер должна иметь и процедура распределения номеров узлов в пределах каждой подсети. Если сеть небольшая, то уникальность адресов может быть обеспечена вручную администратором.

Централизованное распределение адресов

В больших сетях, подобных Интернету, уникальность сетевых адресов гарантируется централизованной, иерархически организованной системой их распределения. Главным органом регистрации глобальных адресов в Интернете с 1998 года является ICANN (Internet Corporation for Assigned Names and Numbers) — неправительственная некоммерческая организация, управляемая советом директоров. Эта организация координирует работу региональных отделов, деятельность которых охватывает большие географические площади: ARIN (Америка), RIPE (Европа), APNIC (Азия и Тихоокеанский регион). Региональные отделы выделяют блоки адресов сетей крупным поставщикам услуг, те, в свою очередь присваивают их своим клиентам, среди которых могут быть и более мелкие поставщики услуг.

Понятно, что в любой автономной сети могут быть использованы произвольные IP-адреса, лишь бы они были синтаксически правильными и уникальными в пределах этой сети. Совпадение адресов в не связанных между собой сетях не вызовет никаких отрицательных последствий. Однако во всех сетях, входящих в единую составную сеть (например, Интернет), должны использоваться гло-

бально уникальные IP-адреса, то есть адреса, однозначно определяющие сетевые интерфейсы в пределах всей составной сети.

Уже сравнительно давно наблюдается дефицит IP-адресов. Очень трудно получить адрес класса В и практически невозможно стать обладателем адреса класса А. При этом надо отметить, что дефицит обусловлен не только ростом сетей, но и тем, что имеющееся адресное пространство используется нерационально. Очень часто владельцы сетей класса С расходуют лишь небольшую часть из имеющихся у них 25-1 адресов. Рассмотрим пример, когда две сети необходимо соединить глобальной связью. В таких случаях в качестве канала связи используют два маршрутизатора, соединенных по схеме «точка-точка» (рис. 12.2). Для вырожденной сети, образованной каналом, связывающим порты двух смежных маршрутизаторов, приходится выделять отдельный номер сети, хотя в этой сети имеются всего два узла.

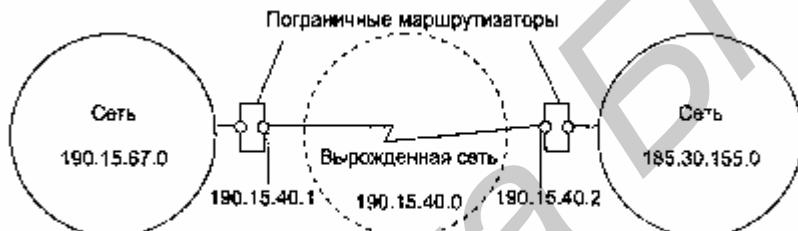


Рис. 12.2. Нерациональное использование пространства IP-адресов

Если же некоторая IP-сеть создана для работы в «автономном режиме», без связи с Интернетом, тогда администратор этой сети волен назначить ей произвольно выбранный номер. Но и в этой ситуации для того, чтобы избежать каких-либо коллизий, в стандартах Интернета определено несколько адресов, рекомендуемых для автономного использования: в классе А — это сеть 10.0.0.0, в классе В — это диапазон из 16 номеров сетей 172.16.0.0-172.31.0.0, в классе С — это диапазон из 255 сетей 192.168.0.0-192.168.255.0. Подробнее о том, как использование данных зарезервированных адресов позволяет справляться с дефицитом сетевых адресов, читайте в разделе «Трансляция сетевых адресов» главы 14.

Для смягчения проблемы дефицита адресов разработчики стека TCP/IP предлагают разные подходы. Принципиальным решением является переход на новую версию IPv6, в которой резко расширяется адресное пространство за счет использования 16-байтных адресов. Однако и текущая версия IPv4 поддерживает некоторые технологии, направленные на более экономное расходование IP-адресов. Одной из таких технологий является технология бесклассовой междоменной маршрутизации (Classless Inter-Domain Routing, CIDR). Технология CIDR основана на масках, она отказывается от традиционной концепции разделения адресов протокола IP па классы, что позволяет выдавать в пользование столько адресов, сколько реально необходимо потребителю. Благодаря CIDR поставщик услуг получает возможность «нарезать» блоки из выделенного ему адресного пространства в точном соответствии с требованиями каждого клиен-

та.

Автоматизация процесса назначения IP-адресов

Для нормальной работы сети каждому сетевому интерфейсу компьютера и маршрутизатора, на которые будут приходить и с которых будут отправляться IP-пакеты, должен быть назначен IP-адрес. Назначение IP-адресов может происходить вручную в результате выполнения процедуры конфигурирования интерфейса, которая для компьютера сводится, например, к заполнению системы экранных форм. При этом администратор должен помнить, какие адреса из имеющегося множества он уже использовал для других интерфейсов, а какие — еще свободны. Даже при не очень большом размере сети эта работа представляет для администратора рутинную, а временами и утомительную процедуру. Протокол Dynamic Host Configuration Protocol (DHCP) освобождает администратора от этих проблем, автоматизируя процесс назначения IP-адресов.

DHCP может поддерживать автоматическое динамическое распределение адресов, а также более простые способы ручного и автоматического статического назначения адресов. Протокол DHCP работает в соответствии с моделью клиент-сервер. Во время старта системы компьютер, являющийся DHCP-клиентом, посыпает в сеть широковещательный запрос на получение IP-адреса. DHCP-сервер откликается и посыпает сообщение-ответ, содержащее IP-адрес и некоторые другие конфигурационные параметры. Предполагается, что DHCP-клиент и DHCP-сервер находятся в одной IP-сети.

При ручной процедуре назначения статических адресов активное участие принимает администратор, который сообщает DHCP-серверу информацию о жестком соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. DHCP-сервер, пользуясь этой информацией, всегда выдает определенному клиенту один и тот же назначенный ему администратором адрес.

При автоматическом статическом способе DHCP-сервер самостоятельно, без вмешательства администратора, выбирает клиенту произвольный IP-адрес из пула наличных IP-адресов. Границы пула задает администратор при конфигурировании DHCP-сервера. Адресдается клиенту из пула в постоянное пользование, то есть с неограниченным сроком аренды. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первого назначения DHCP-сервером IP-адреса клиенту. При всех последующих запросах сервер возвращает клиенту тот же самый IP-адрес.

При динамическом распределении адресов DHCP-сервер выдаст адрес клиенту на ограниченное время, называемое временем аренды (lease duration), что дает возможность впоследствии повторно использовать этот IP-адрес для назначения другому компьютеру. Таким образом, помимо основного преимущества DHCP — автоматизации рутинной работы администратора по конфигурированию стека TCP/IP на каждом компьютере, динамическое разделение адресов в принципе позволяет строить IP-сеть, количество узлов в которой превышает количество имеющихся в распоряжении администратора IP-адресов.

DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие дублирования адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра «продолжительность аренды», который определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от DHCP-сервера в аренду.

Рассмотрим работу протокола DHCP в ситуации, когда компьютер, являющийся DHCP-клиентом, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

DHCP-сервер может назначить клиенту не только IP-адрес, но и другие параметры стека TCP/IP, необходимые для его эффективной работы, например, маску, IP-адрес маршрутизатора по умолчанию, IP-адрес сервера DNS, доменное имя компьютера и т. п.

Однако непостоянство IP-адресов влечет за собой и некоторые проблемы.

- Снижается надежность системы из-за наличия центрального элемента — сервера DHCP.
- Возникают сложности при преобразовании символьного доменного имени в IP-адрес.
- Трудно осуществлять удаленное управление и автоматический мониторинг (например, сбор статистики) интерфейса, если в качестве его идентификатора выступает динамически изменяемый IP-адрес.
- Для обеспечения безопасности сети многие сетевые устройства могут блокировать (фильтровать) пакеты, значения полей которых имеют некоторые заранее заданные значения. При динамическом назначении адресов усложняется фильтрация по IP-адресам.

Все эти проблемы в той или иной степени находят решение. Так, для снижения риска выхода сети из строя из-за отказа сервера DHCP в сети ставят резервный сервер DHCP. А для серверов, к которым пользователи часто обращаются по символьному имени, назначают статические IP-адреса, оставляя динамические только для клиентских компьютеров. Есть примеры успешных подходов и к решению остальных задач.

Выводы

- IP-адреса уникально идентифицируют узел в пределах составной сети, поэтому они должны назначаться централизовано.
- Если сеть небольшая и автономная, то уникальность IP-адресов в пределах этой сети может быть обеспечена администратором сети. При этом он может выбирать для нумерации сетей и узлов любые синтаксически правильные IP-адреса.
- Если сеть очень велика, как, например, Интернет, то процесс назначения IP-

адресов становится сложным и разбивается на два этапа. Первый — распределение номеров сетей — регулируется специальным административным органом (в Интернете это ICAN), обеспечивающим однозначность нумерации сетей. После того как сеть получила номер, наступает второй этап — назначение номеров узлам сети.

□ Назначение IP-адресов узлам сети может происходить вручную — администратор сам ведет списки свободных и занятых адресов и конфигурирует сетевой интерфейс, либо автоматически — с использованием протокола DHCP. В этом случае администратор заранее определяет для сервера DHCP диапазон свободных для распределения адресов, из которого последний автоматически выделяет адреса узлам в ответ на поступившие от них запросы.

Протоколы разрешения адресов

Отображение IP-адресов на локальные адреса

Одной из главных задач, которая ставилась при создании протокола IP, являлось обеспечение совместной согласованной работы в сети, состоящей из подсетей, в общем случае использующих разные сетевые технологии.

Взаимодействие технологии TCP/IP с частными технологиями подсетей происходит многократно при перемещении пакета IP по составной сети. На каждом маршрутизаторе протокол IP определяет, в какую следующую подсеть и какому пограничному узлу в этой подсети надо направить пакет. Таким пограничным узлом является маршрутизатор, и протоколу IP известен его IP-адрес. Очевидно, что для того, чтобы частная технология подсети смогла доставить пакет на следующий маршрутизатор, необходимо:

- во-первых, упаковать пакет в кадр соответствующего для данной подсети формата (например, Ethernet);
- во-вторых, снабдить кадр адресом, формат которого был бы понятен локальной технологии подсети (преобразовать, например, IP-адрес в MAC-адрес).

Решением этих задач, как уже было сказано, занимается уровень сетевых интерфейсов стека TCP/IP.

Для определения локального адреса по IP-адресу используется протокол разрешения адресов (Address Resolution Protocol, ARP).

Протокол разрешения адресов реализуется различным образом в зависимости от того, какой протокол канального уровня работает в данной сети — протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещательного доступа одновременно ко всем узлам сети или же какой-либо из протоколов глобальной сети (X.25, frame relay), которые, как правило, не поддерживают широковещательный доступ.

Рассмотрим работу протокола ARP в сетях с широковещанием. В результате конфигурирования сети каждый интерфейс знает свои IP-адрес и MAC-адрес. Кроме того, на каждом интерфейсе (сетевом адаптере или порту маршрутизатора) поддерживается отдельная ARP-таблица, определяющая соответствие между

ду IP-адресами и MAC-адресами других узлов данной подсети. Первоначально, при включении компьютера или маршрутизатора в сеть все его ARP-таблицы пусты.

Пусть в какой-то момент модуль IP передаст пакет на уровень сетевых интерфейсом, например драйверу Ethernet, и ему требуется найти на основе известного IP-адреса MAC-адрес узла назначения. Для этого протокол IP обращается к протоколу ARP. Работа ARP начинается с просмотра ARP-таблицы соответствующего интерфейса. Как мы предположили, обращение к ARP произошло в начале работы, и таблица оказалась пуста. Те же самые действия последовали бы, если бы таблица содержала некоторые записи, но нужный IP-адрес в ARP-таблице отсутствовал. В обоих этих случаях исходящий IP-пакет, для которого оказалось невозможным определить локальный адрес из ARP-таблицы, запоминается в буфере, а протокол ARP формирует запрос (ARP-запрос), вкладывает его в кадр протокола канального уровня и рассыпает широковещательно.

Все интерфейсы подсети получают ARP-запрос и сравнивают указанный там IP-адрес с собственным. Интерфейс, который констатировал совпадение, формирует ARP-ответ, указывая в нем свой IP-адрес и свой локальный адрес, а затем отправляет его уже направленно, так как в ARP-запросе имеется локальный адрес отправителя. ARP-запросы и ответы используют один и тот же формат пакета. В табл. 12.2 приведены значения полей примера ARP-запроса для передачи по сети Ethernet.

Таблица 12.2. Пример ARP-запроса

Поле	Значение
Тип сети	1 (0x1)
Тип протокола	2048 (0x800)
Длина локального адреса	6 (0x6)
Длина сетевого адреса	4 (0x4)
Опция	1 (0x1)
Локальный адрес отправителя	008048EB7E60
Сетевой адрес отправителя	194.85.135.75
Локальный (искомый) адрес получателя	000000000000
Сетевой адрес получателя	194.85.135.65

В поле «Тип сети» для сетей Ethernet указывается значение 1.

Поле «Тип протокола» позволяет использовать протокол ARP не только для протокола IP, но и для других сетевых протоколов. Для IP значение этого поля равно 0x0800.

Длина локального адреса для протокола Ethernet равна 6 байт, а длина IP-адреса — 4 байт. В поле операции для ARP-запросов указывается значение 1, если это запрос, и 2, если это ответ.

Из этого запроса видно, что в сети Ethernet узел с IP-адресом 194.85.135.75 пытается определить, какой MAC-адрес имеет другой узел той же сети, сетевой адрес которого 194.85.135.65. Поле искомого локального адреса заполнено ну-

лями.

Ответ присыпает узел, опознавший свой IP-адрес. Если в сети нет машины с искомым IP-адресом, то ARP-ответа не будет. Протокол IP уничтожает IP-пакеты, направляемые по этому адресу. (Заметим, что протоколы верхнего уровня не могут отличить случай повреждения сети Ethernet от случая отсутствия машины с искомым IP-адресом.) В табл. 12.3 помещены значения полей ARP-ответа, который мог бы поступить на приведенный выше пример ARP-запроса.

Таблица 12.3. Пример ARP-ответа

Поле	Значение
Тип сети	1 (0x1)
Тип протокола	2048 (0x800)
Длина локального адреса	6 (0x6)
Длина сетевого адреса	4 (0x4)
Опция	1 (0x1)
Локальный адрес отправителя	OOEOF77F1920
Сетевой адрес отправителя	194.85.135.65
Локальный (искомый) адрес полу-	008048EB7E60
Сетевой адрес получателя	194.85.135.75

Этот ответ получает машина, отправившая ARP-запрос. В результате обмена этими двумя ARP-сообщениями модуль IP, имеющий адрес 194.85.135.75, определяет, что IP-адресу 194.85.135.65 соответствует MAC-адрес OOEOF77F1920. Этот адрес будет затем помещен в заголовок кадра Ethernet, ожидавшего отправления IP-пакета.

Чтобы уменьшить число ARP-обращений в сети, найденное соответствие между IP-адресом и MAC-адресом записывается в ARP-таблицу соответствующего интерфейса. Новая запись в ARP-таблице появляется автоматически, спустя несколько миллисекунд после того, как модуль ARP проанализировал ARP-ответ. Теперь, если вдруг вновь возникнет необходимость послать пакет по адресу 194.85.135.65, то протокол, прежде чем послать широковещательный запрос, проверит, нет ли уже такого адреса в ARP-таблице. Ниже показана ARP-таблица, в которую в ходе работы сети были добавлены еще две записи (табл. 12.4).

Поле «Тип записи» может содержать одно из двух значений — «динамический» или «статический». Статические записи создаются вручную с помощью утилиты арг и не имеют срока устаревания, точнее, они существуют до тех пор, пока компьютер или маршрутизатор не будут выключены.

Таблица 12.4. Пример ARP-таблицы

IP-адрес	MAC -адрес	Тип записи
194.85.135.75	008048EB7E60	Динамический
194.85.135.70	08005A21A722	Динамический
194.85.60.21	008048EB7567	Статический

Динамические же записи создаются модулем протокола ARP, использующим широковещательные возможности локальных сетевых технологий. ARP-таблица пополняется не только за счет поступающих на данный интерфейс ARP-ответов, но и в результате извлечения полезной информации из широковещательных ARP-запросов. Действительно, в каждом запросе, как это видно из табл. 12.2, наряду с прочими данными содержаться IP-адрес отправителя и его MAC-адрес. Даже если нет совпадения с запрашиваемым адресом, почему бы не поместить эту ценную информацию об отправителе в собственную ARP-таблицу.

Динамические записи должны периодически обновляться. Если запись не обновлялась в течение определенного времени (порядка нескольких минут), то она исключается из таблицы. Таким образом, в ARP-таблице содержатся записи не обо всех узлах сети, а только о тех, которые активно участвуют в сетевых операциях. Поскольку такой способ хранения информации называют кэшированием, ARP-таблицы иногда называют ARP-кэшем.

ПРИМЕЧАНИЕ

Некоторые реализации протоколов IP и ARP не ставят IP-пакеты в очередь на время ожидания ARP-ответов. Вместо этого IP-пакет просто уничтожается, а его восстановление возлагается на модуль TCP или прикладной процесс, работающий через UDP. Такое восстановление выполняется с помощью тайм-аутов и повторных передач. Повторная передача сообщения проходит успешно, так как первая попытка уже вызнала заполнение ARP-таблицы.

Совсем другой способ разрешения адресов используется в глобальных сетях, в которых не поддерживаются широковещательные сообщения. Здесь администратору сети чаще всего приходится вручную формировать и помещать на какой-либо сервер ARP-таблицы, в которых он задает, например, соответствие IP-адресов адресам X.25, которые имеют для протокола IP смысл локальных адресов. В то же время сегодня наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети.

При таком централизованном подходе для всех узлов и маршрутизаторов вручную нужно задать только IP-адрес и локальный адрес выделенного маршрутизатора. Затем каждый узел и маршрутизатор регистрируют свои адреса в выделенном маршрутизаторе, а при необходимости установления соответствия между IP-адресом и локальным адресом протокольный модуль обращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора. Работающий таким образом маршрутизатор называют ARP-сервером.

В некоторых случаях возникает обратная задача — нахождение IP-адреса по известному локальному адресу. Тогда в действие вступает реверсивный протокол ARP (Reverse Address Resolution Protocol, RARP). Этот протокол используется, например, при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих MAC-адрес своего сетевого адаптера.

Организация доменов и доменных имен

Для идентификации компьютеров аппаратное и программное обеспечение в сетях TCP/IP полагается на IP-адреса, поэтому для доступа к сетевому ресурсу в параметрах программы вполне достаточно указать IP-адрес, чтобы программа правильно поняла, к какому хосту ей нужно обратиться. Например, команда ftp://192.45.66.17 будет устанавливать сеанс связи с нужным ftp-сервером, а команда http://203.23.106.33 откроет начальную страницу на корпоративном web-сервере. Однако пользователи обычно предпочитают работать с символьными именами компьютеров, и операционные системы локальных сетей приучили их к этому удобному способу. Следовательно, в сетях TCP/IP должны существовать символьные имена хостов и механизм для установления соответствия между символьными именами и IP-адресами.

В операционных системах, которые первоначально разрабатывались для работы в локальных сетях, таких как Novell NetWare, Microsoft Windows или IBM OS/2, пользователи всегда работали с символьными именами компьютеров. Так как локальные сети состояли из небольшого числа компьютеров, то использовались так называемые плоские имена, состоящие из последовательности символов, не разделенных на части. Примерами таких имен являются: NW1_1, mail2, MOSCOW_ SALES_2. Для установления соответствия между символьными именами и MAC-адресами в этих операционных системах применялся механизм широковещательных запросов, подобный механизму запросов протокола ARP. Так, широковещательный способ разрешения имен реализован в протоколе NetBIOS, на котором были построены многие локальные ОС. Так называемые NetBIOS-имена стали на долгие годы одним из основных типов плоских имен в локальных сетях.

Для стека TCP/IP, рассчитанного в общем случае на работу в больших территориально распределенных сетях, подобный подход оказывается неэффективным по некоторым причинам.

Плоские имена не дают возможности разработать единый алгоритм обеспечения уникальности имен в пределах большой сети. В небольших сетях уникальность имен компьютеров обеспечивает администратор сети, записывая несколько десятков имен в журнале или файле. При росте сети задачу решают уже несколько администраторов, согласовывая имена между собой неформальным способом. Однако если сеть расположена в разных городах или странах, то администраторам каждой части сети нужно придумать способ именования, который позволил бы им давать имена новым компьютерам независимо от других администраторов, обеспечивая в то же время уникальность имен для всей сети. Самый надежный способ решения этой задачи — отказ от плоских имен в принципе.

Широковещательный способ установления соответствия между символьными именами и локальными адресами хорошо работает только в небольшой локальной сети, не разделенной на подсети. В крупных сетях, где общая широковеща-

тельность не поддерживается, нужен другой способ разрешения символьных имен. Обычно хорошей альтернативой широковещательности является применение централизованной службы, поддерживающей соответствие между различными типами адресов всех компьютеров сети. Компания Microsoft для своей корпоративной операционной системы Windows NT разработала централизованную службу WINS, которая поддерживает базу данных NetBIOS-имен и соответствующих им IP-адресов.

Для эффективной организации именования компьютеров в больших сетях естественным является применение иерархических составных имен.

В стеке TCP/IP применяется доменная система имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей (рис. 12.3).

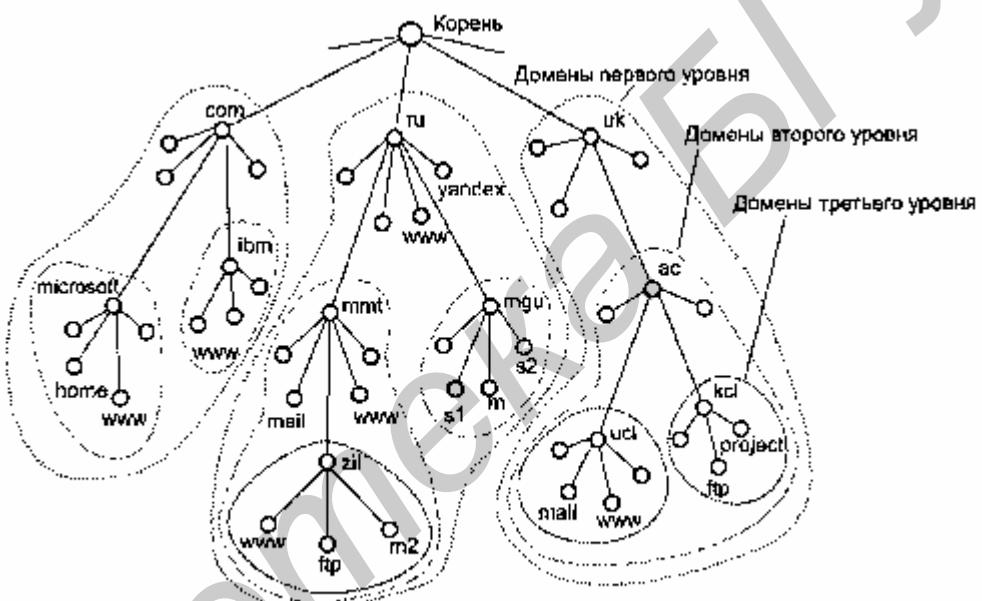


Рис. 12.3. Пространство доменных имен

Иерархия доменных имен аналогична иерархии имен файлов, принятой во многих популярных файловых системах. Дерево имен начинается с корня, обозначеного здесь точкой (.). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть имени и т. д. Младшая часть имени соответствует конечному узлу сети. В отличие от имен файлов, при записи которых сначала указывается самая старшая составляющая, затем составляющая более низкого уровня и т. д., запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяются друг от друга точкой. Например, в имени partnering.microsoft.com составляющая partnering является именем одного из компьютеров в домене microsoft.com.

Разделение имени на части позволяет разделить административную ответственность за назначение уникальных имен между различными людьми или организациями в пределах своего уровня иерархии. Так, для примера, приведенного на рис. 12.3, один человек может нести ответственность за то, чтобы все имена,

которые имеют окончание «ru», имели уникальную следующую вниз по иерархии часть. Если этот человек справляется со своими обязанностями, то все имена типа www.ru, mail.mmt.ru или m2.zil.mmt.ru будут отличаться второй по старшинству частью.

Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций между организациями, отвечающими за имена одного уровня иерархии. Очевидно, что должна существовать одна организация, отвечающая за назначение имен верхнего уровня иерархии.

Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен (domain) имен. Например, имена www1.zil.mmt.ru, ftp.zil.mmt.ru, yandex.ru и s1.mgu.ru входят в домен ru, так как все эти имена имеют одну общую старшую часть — имя ru. Другим примером является домен mgu.ru. Из представленных на рис. 12.3 имен в него входят имена s1.mgu.ru, s2.mgu.ru и m.mgu.ru. Этот домен образуют имена, у которых две старшие части всегда равны mgu.ru. Имя www.mmt.ru в домене mgu.ru не входит, так как имеет отличающуюся составляющую mmt.

ВНИМАНИЕ

Термин «домен» очень многозначен, поэтому его нужно трактовать в рамках определенного контекста. Кроме доменов имен стека TCP/IP в компьютерной литературе также часто упоминаются домены Windows NT, домены коллизий и некоторые другие. Общим у всех этих терминов является то, что они описывают некоторое множество компьютеров, обладающее каким-либо определенным свойством.

Если один домен входит в другой домен как его составная часть, то такой домен могут называть поддоменом (subdomain), хотя название домен за ним также остается. Обычно поддомен называют по имени той его старшей составляющей, которая отличает его от других поддоменов. Например, поддомен mmt.ru обычно называют поддоменом (или доменом) mmt. Имя поддомену назначает администратор вышестоящего домена. Хорошей аналогией домена является каталог файловой системы.

Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой в доменной системе имен различают краткие имена, относительные имена и полные доменные имена. Краткое имя — это имя конечного узла сети: хоста или порта маршрутизатора. Краткое имя — это лист дерева имен. Относительное имя — это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Например, www1.zil — это относительное имя. Полное доменное имя (fully qualified domain name, FQDN) включает составляющие всех уровней иерархии, начиная от краткого имени и кончая корневой точкой: www1.zil.mmt.ru.

Корневой домен управляет центральными органами Интернета; IANA и InterNIC. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международ-

ному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, например, ru (Россия), uk (Великобритания), fin (Финляндия), us (Соединенные Штаты), а для различных типов организаций — следующие обозначения:

- com — коммерческие организации (например, microsoft.com);
- edu — образовательные организации (например, mit.edu);
- gov — правительственные организации (например, nsf.gov);
- org — некоммерческие организации (например, fidonet.org);
- net — организации поддержки сетей (например, nsf.net).

Каждый домен администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Чтобы получить доменное имя, необходимо зарегистрироваться в какой-либо организации, которой организация InterNIC делегировала свои полномочия по распределению имен доменов. В России такой организацией является РосНИИРОС, которая отвечает за делегирование имен поддоменов в домене ru.

Необходимо подчеркнуть, что компьютеры входят в домен в соответствии со своими составными именами, при этом они могут иметь абсолютно независимые друг от друга IP-адреса, принадлежащие к различным сетям и подсетям. Например, в домен mgu.ru могут входить хосты с адресами 132.13.34.15, 201.22.100.33 и 14.0.0.6.

Доменная система имен реализована в Интернете, но она может работать и как автономная система имен в любой крупной корпоративной сети, которая также использует стек TCP/IP, но никак не связана с Интернетом.

Система доменных имен DNS

Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста, так и средствами централизованной службы. На раннем этапе развития Интернета на каждом хосте вручную создавался текстовый файл с известным именем hosts.txt. Этот файл состоял из некоторого количества строк, каждая из которых содержала одну пару «IP-адрес — доменное имя», например 102.54.94.97 — rhino.acme.com.

По мере роста Интернета файлы hosts также росли, и создание масштабируемого решения для разрешения имен вышло на первый план.

Таким решением стала специальная служба — система доменных имен (Domain Name System, DNS). DNS — это централизованная служба, основанная на распределенной базе отображений «доменное имя — IP-адрес». Служба DNS использует в своей работе протокол типа «клиент-сервер». В нем определены DNS-серверы и DNS-клиенты. DNS-серверы поддерживают распределенную базу отображений, а DNS-клиенты обращаются к серверам с запросами о разрешении доменного имени в IP-адрес.

Служба DNS использует текстовые файлы почти такого формата, как и файл hosts, и эти файлы администратор также подготовливает вручную. Однако служба DNS опирается на иерархию доменов, и каждый сервер службы DNS

храпит только часть имен сети, а не все имена, как это происходит при использовании файлов hosts. При росте количества узлов в сети проблема масштабирования решается созданием новых доменов и поддоменов имен и добавлением в службу DNS новых серверов.

Для каждого домена имен создается свой DNS-сервер. Имеются две системы распределения имен па серверах. В первом случае сервер может хранить отображения «доменное имя — IP-адрес» для всего домена, включая все его поддомены. Однако такое решение оказывается плохо масштабируемым, так как при добавлении новых поддоменов нагрузка на этот сервер может превысить его возможности. Чаще используется другой подход, когда сервер домена хранит только имена, которые заканчиваются на следующем ниже уровне иерархии по сравнению с именем домена. (Аналогично каталогу файловой системы, который содержит записи о файлах и подкаталогах, непосредственно в него «входящих».) Именно при такой организации службы DNS нагрузка по разрешению имен распределяется более-менее равномерно между всеми DNS-серверами сети. Например, в первом случае DNS-сервер домена mmt.ru будет хранить отображения для всех имен, заканчивающихся на mmt.ru: www1.zil.mmt.ru, ftp.zil.mmt.ru, mail.mmt.ru и т. д. Во втором случае этот сервер хранит отображения только имен типа mail.mmt.ru, www.mmt.ru, а все остальные отображения должны храниться на DNS-сервере поддомена zil.

Каждый DNS-сервер кроме таблицы отображений имен содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой IP-адреса соответствующих серверов. Для обслуживания корневого домена выделено несколько дублирующих друг друга DNS-серверов, IP-адреса которых являются широко известными (их можно узнать, например, в InterNIC).

Процедура разрешения DNS-имени во многом аналогична процедуре поиска файловой системой адреса файла по его символьному имени. Действительно, в обоих случаях составное имя отражает иерархическую структуру организации соответствующих справочников — каталогов файлов или таблиц DNS. Здесь домен и доменный DNS-сервер являются аналогом каталога файловой системы. Для доменных имен, так же как и для символьных имен файлов, характерна независимость именования от физического местоположения.

Процедура поиска адреса файла по символьному имени заключается в последовательном просмотре каталогов, начиная с корневого. При этом предварительно проверяется кэш и текущий каталог. Для определения IP-адреса по доменному имени также необходимо просмотреть все DNS-серверы, обслуживающие цепочку поддоменов, входящих в имя хоста, начиная с корневого домена. Существенным же отличием является то, что файловая система расположена на одном компьютере, а служба DNS по своей природе является распределенной.

Существует две основные схемы разрешения DNS-имен. В первом варианте работу по поиску IP-адреса координирует DNS-клиент.

1. DNS-клиент обращается к корневому DNS-серверу с указанием полного доменного имени.
2. DNS-сервер отвечает, указывая адрес следующего DNS-сервера, обслужи-

вающего домен верхнего уровня, заданный в старшей части запрошенного имени.

3. DNS-клиент делает запрос следующего DNS-сервера, который отсылает его к DNS-серверу нужного поддомена и т. д., пока не будет найден DNS-сервер, в котором хранится соответствие запрошенного имени IP-адресу. Этот сервер дает окончательный ответ клиенту.

Такая схема взаимодействия называется нерекурсивной, или итеративной, когда клиент сам итеративно выполняет последовательность запросов к разным серверам имен. Так как эта схема загружает клиента достаточно сложной работой, то она применяется редко.

Во втором варианте реализуется рекурсивная процедура.

1. DNS-клиент запрашивает локальный DNS-сервер, то есть тот сервер, обслуживающий поддомен, к которому принадлежит имя клиента.

2. Если локальный DNS-сервер знает ответ, то он сразу же возвращает его клиенту; это может соответствовать случаю, когда запрошенное имя входит в тот же поддомен, что и имя клиента, а также случаю, когда сервер уже узнавал данное соответствие для другого клиента и сохранил его в своем кэше.

3. Если локальный сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу и т. д. точно так же, как это делает клиент в первом варианте. Получив ответ, он передаст его клиенту, который все это время просто ожидает его от своего локального DNS-сервера.

В этой схеме клиент перепоручает работу своему серверу, поэтому схема называется косвенной, или рекурсивной. Практически все DNS-клиенты используют рекурсивную процедуру.

Для ускорения поиска IP-адресов DNS-серверы широко применяют процедуру кэширования проходящих через них ответов. Чтобы служба DNS могла оперативно отрабатывать изменения, происходящие в сети, ответы кэшируются на определенное время — обычно от нескольких часов до нескольких дней.

Выводы

- Установление соответствия между IP-адресом и аппаратным адресом осуществляется протоколом разрешения адресов.
- Существует два принципиально отличных подхода к разрешению адресов; в сетях, поддерживающих широковещание, и в сетях, его не поддерживающих.
- Протокол ARP, работающий в сетях Ethernet, Token Ring, FDDI, для трансляции IP-адреса в MAC-адрес выполняет широковещательный ARP-запрос. Для ускорения процедуры преобразования адресов протокол ARP кэширует полученные ответы в ARP-таблицах.
- В сетях, в которых не поддерживаются широковещательные сообщения, ARP-таблицы хранятся централизовано на выделенном ARP-сервере. Таблицы составляются либо вручную администратором, либо автоматически — при включении каждый узел регистрирует в них свои адреса. При необходимости установления соответствия между IP-адресом и локальным адресом узел обращается к ARP-серверу с запросом и автоматически получает ответ без участия

администратора.

- В стеке TCP/IP применяется доменная система символьных имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен. Доменные имена назначаются централизованно, если сеть является частью Интернета, в противном случае — локально.
- Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста с использованием файла hosts, так и с помощью централизованной службы DNS, основанной на распределенной базе отображений «доменное имя — IP-адрес».

ГЛАВА 13 Протоколы межсетевого и транспортного уровней TCP/IP

Основу транспортных средств стека TCP/IP составляют протоколы межсетевого и транспортного уровней. Именно они определяют маршрут, обеспечивают передачу данных от отправителя к получателям через объединенную систему компьютерных сетей и, если это требуется, гарантируют надежность доставки пакетов. Остальные средства стека TCP/IP выполняют в некотором смысле вспомогательную роль — сервисы верхнего уровня поддерживают интерфейс с пользователями и приложениями, а нижний уровень — интерфейс с технологиями составляющих сетей.

Протокол IP

Основные функции протокола IP

Название протокола IP — протокол межсетевого взаимодействия (Internet Protocol, IP) — отражает его суть: он должен передавать пакеты между сетями. В каждой очередной сети, лежащей на пути перемещения пакета, протокол IP вызывает средства транспортировки, принятые в этой сети, чтобы с их помощью передать этот пакет на маршрутизатор, ведущий к следующей сети, или непосредственно к узлу-получателю.

Протокол IP относится к протоколам без установления соединения. Перед IP не ставится задача надежной доставки сообщений от отправителя к получателю. Протокол IP обрабатывает каждый IP-пакет как независимую единицу, не имеющую связи ни с какими другими IP-пакетами. В протоколе IP нет механизмов, обычно применяемых для увеличения достоверности конечных данных: отсутствует квитирование — обмен подтверждениями между отправителем и получателем, нет процедуры упорядочивания, повторных передач или других подобных функций. Если во время продвижения пакета произошла какая-либо ошибка, то протокол IP по своей инициативе ничего не предпринимает для исправления этой ошибки. Например, если на промежуточном маршрутизаторе пакет был отброшен по причине истечения времени жизни или из-за ошибки в контрольной сумме, то модуль IP не пытается заново послать испорченный или потерянный пакет. Все вопросы обеспечения надежности при доставке данных по составной сети в стеке TCP/IP решает протокол TCP, работающий непосредственно над протоколом IP. Именно TCP организует повторную передачу пакетов, когда в этом возникает необходимость.

Важной особенностью протокола IP, отличающей его от других сетевых протоколов (например, от сетевого протокола IPX), является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными максимально допустимыми значениями длины поля данных кадров (MTU). Способность к фрагментации во многом способствовала тому, что протокол IP смог занять доминирующие позиции в сложных составных сетях.

Имеется прямая связь между многообразием полей заголовка пакета и функциональной сложностью протокола, который работает с этим заголовком. Чем проще заголовок — тем проще соответствующий протокол. Большая часть действий протокола связана с обработкой той служебной информации, которая переносится в полях заголовка пакета. Поэтому очень полезно изучить назначение каждого тюля заголовка IP-пакета, это даст не только формальные знания о структуре пакета, но и объяснит все основные режимы работы протокола IP.

Структура IP-пакета

IP-пакет состоит из заголовка и поля данных. Заголовок, как правило, имеющий длину 20 байт, имеет следующую структуру (рис. 13.1).

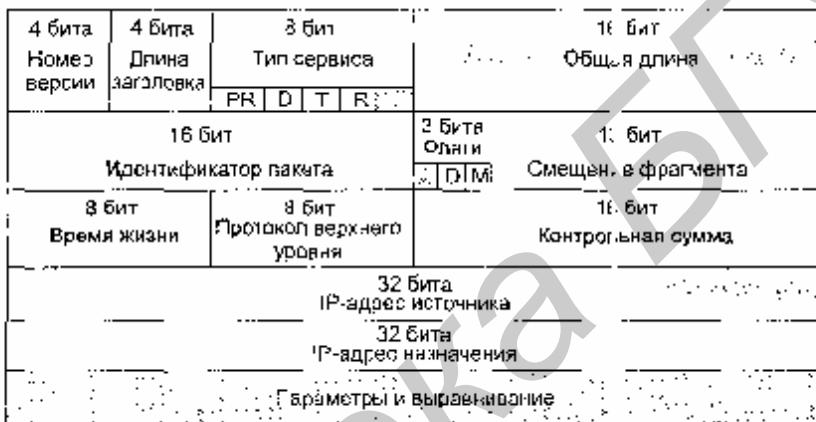


Рис. 13.1. Структура заголовка IP-пакета

Поле Номер версии (Version), занимающее 4 бита, указывает версию протокола IP. Сейчас повсеместно используется версия 4 (IPv4) и готовится переход на версию 6 (IPv6).

Поле Длина заголовка (IHL) IP-пакета занимает 4 бита и указывает значение длины заголовка, измеренное в 32-битовых словах. Обычно заголовок имеет длину в 20 байт (пять 32-битовых слов), но при увеличении объема служебной информации эта длина может быть увеличена за счет использования дополнительных байтов в поле Параметры (IP Options). Наибольший заголовок занимает 60 байт.

Поле Тип сервиса (Type of Service, ToS) занимает один байт. Оно используется пока очень редко и обычно в нем устанавливается 0. Это поле позволяет приложениям влиять на качество обслуживания, задавая приоритетность пакета и критерий выбора маршрута. Первые три бита этого поля образуют подполе приоритета пакета (Precedence). Приоритет может иметь значения от самого низкого — 0 (нормальный пакет) до самого высокого — 7 (пакет управляющей информации). Маршрутизаторы и компьютеры могут принимать во внимание приоритет пакета и обрабатывать более важные пакеты в первую очередь. Поле Тип сервиса содержит также три бита, определяющие критерий выбора маршрута. Реально выбор осуществляется между тремя альтернативами: малой задержкой, высокой достоверностью и высокой пропускной способностью. Уста-

новленный бит D (delay) говорит о том, что маршрут должен выбираться для минимизации задержки доставки данного пакета, бит T (throughput) — для максимизации пропускной способности, а бит R (reliability) — для максимизации надежности доставки. Во многих сетях улучшение одного из этих параметров связано с ухудшением другого, кроме того, обработка каждого из них требует дополнительных вычислительных затрат. Поэтому редко, когда имеет смысл устанавливать одновременно хотя бы два из этих трех критериев выбора маршрута. Зарезервированные биты имеют нулевое значение.

Поле Общая длина (Total Length) занимает два байта и означает общую длину пакета с учетом заголовка и поля данных. Максимальная длина пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65 535 байт, однако в большинстве компьютеров и сетей столь большие пакеты не используются. При передаче по сетям различного типа длина пакета выбирается с учетом максимальной длины пакета протокола нижнего уровня, несущего IP-пакеты. Если это кадры Ethernet, то выбираются пакеты с максимальной длиной в 1500 байт, умещающиеся в поле данных кадра Ethernet. В стандарте предусматривается, что все хосты должны быть готовы принимать пакеты вплоть до 576 байт длиной (приходят ли они целиком или по фрагментам), Хостам рекомендуется отправлять пакеты размером более чем 576 байт, только если они уверены, что принимающий хост или промежуточная сеть готовы обслуживать пакеты такого размера.

Поле Идентификатор пакета (Identification) занимает два байта и используется для распознавания пакетов, образовавшихся путем фрагментации исходного пакета. Все фрагменты должны иметь одинаковое значение этого поля.

Поле Флаги (Flags) занимает три бита и содержит признаки, связанные с фрагментацией. Установленный в 1 бит DF (Do not Fragment) запрещает маршрутизатору фрагментировать данный пакет, а установленный в 1 бит MF (More Fragments) говорит о том, что данный пакет является промежуточным (не последним) фрагментом. Оставшийся бит зарезервирован.

Поле Смещение фрагмента (Fragment Offset) занимает 13 бит и задает смещение в байтах поля данных этого пакета от начала общего поля данных исходного пакета, подвергнутого фрагментации. Используется при сборке/разборке фрагментов пакетов при передачах их между сетями с различными величинами MTU. Смещение должно быть кратно 8 байт.

Поле Время жизни (Time to Live) занимает один байт и означает предельный срок, в течение которого пакет может перемещаться по сети. Время жизни данного пакета измеряется d секундах и задается источником передачи. На маршрутизаторах и в других узлах сети, в которые попадает пакет, по истечении каждой секунды из его текущего времени жизни вычитается единица; единица вычитается и в том случае, когда время задержки оказалось меньше секунды. Поскольку современные маршрутизаторы редко обрабатывают пакет дольше, чем за одну секунду, то время жизни можно считать равным максимальному числу узлов, которые разрешено пройти данному пакету до того, как он достигнет места назначения. Если параметр времени жизни станет нулевым до того, как пакет достигнет получателя, этот пакет будет уничтожен. Время жизни

можно рассматривать как часовой механизм самоуничтожения. Значение этого поля изменяется при обработке заголовка IP-пакета.

Идентификатор Протокол верхнего уровня (Protocol) занимает один байт и указывает, какому протоколу верхнего уровня принадлежит информация, размещенная в поле данных пакета. Значения идентификаторов для различных протоколов приводятся в документе RFC «Assigned Numbers», например, 6 означает, что в пакете находится сегмент TCP, 17 — дейтаграмма UDP, 1 — пакет ICMP.

Контрольная сумма (Header Checksum) занимает два байта и рассчитывается только по заголовку. Поскольку некоторые поля заголовка меняют свое значение в процессе передачи пакета по сети (например, время жизни), контрольная сумма проверяется и повторно рассчитывается при каждой обработке IP-заголовка. Контрольная сумма (16 бит) подсчитывается как дополнение к сумме всех 16-битных слов заголовка. При вычислении контрольной суммы значение самого поля «контрольная сумма» устанавливается в нуль. Если контрольная сумма неверна, то пакет будет отброшен, как только ошибка обнаружится.

Поля IP-адрес источника (Source IP Address) и IP-адрес назначения (Destination IP Address) имеют одинаковую длину (32 бита) и одинаковую структуру.

Поле Параметры (IP Options) является необязательным и используется обычно только при отладке сети. Механизм задания параметров предоставляет функции управления, которые необходимы или просто полезны в определенных ситуациях, однако он не нужен при обычных коммуникациях. Это поле состоит из нескольких подполей, каждое из которых может быть одного из восьми предопределенных типов. В этих под полях можно указывать точный маршрут прохождения маршрутизаторов, регистрировать проходимые пакетом маршрутизаторы, помещать данные системы безопасности, а также временные отметки. Так как число подполей может быть произвольным, то в конце поля Параметры должно быть добавлено несколько байтов для выравнивания заголовка пакета по 32-битной границе.

Поле Выравнивание (Padding) заполняется нулями и используется для того, чтобы гарантировать завершение IP-заголовка на 32-битной границе.

Ниже приведена распечатка значений полей заголовка одного из реальных IP-пакетов, захваченных в сети Ethernet средствами анализатора протоколов Microsoft Network Monitor.

IP: Version = 4 (0x4)

IP: Header Length = 20 (0x14)

IP: Service Type = 0 (0x0)

IP: Precedence = Routine

IP: ...0.... = Normal Delay

IP:0... = Normal Throughput

IP:0.. = Normal Reliability

IP: Total Length = 54 (0x36)

"IP: Identification = 31746 (0x7C02)

IP: Flags Summary = 2 (0x2)

IP:0 = Last fragment in datagram

IP:1. = Cannot fragment datagram
IP: Fragment Offset = 0 (0x0) bytes
IP: Time to Live - 128 (0x80)
IP: Protocol = TCP - Transmission Control
IP: Checksum = 0xEB86
IP: Source Address - 194.85.135'.75
IP: Destination Address = 194.85.135.66
IP: Data: Number of data bytes remaining = 34 (0x0022)

Выводы

- Протокол IP решает задачу доставки сообщений между узлами составной сети.
- Протокол IP относится к протоколам без установления соединения, поэтому он не дает никаких гарантий надежной доставки сообщений. Все вопросы обеспечения надежности доставки данных в составной сети в стеке TCP/IP решает протокол TCP, основанный на установлении логических соединений между взаимодействующими процессами.
- Важной особенностью протокола IP, отличающей его от других сетевых протоколов (например, от сетевого протокола IPX), является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными максимально допустимыми значениями длины поля данных кадров (MTU).
- IP-пакет состоит из заголовка и поля данных. Максимальная длина пакета — 65 535 байт. Заголовок обычно имеет длину 20 байт и содержит информацию о сетевых адресах отправителя и получателя, о параметрах фрагментации, о времени жизни пакета, о контрольной сумме и некоторых других.
- В поле данных IP-пакета находятся сообщения более высокого уровня, например TCP или UDP.

Таблицы маршрутизации в IP-сетях

Программные модули протокола IP устанавливаются на всех конечных станциях и маршрутизаторах сети. Для продвижения пакетов они используют таблицы маршрутизации.

Примеры таблиц для маршрутизаторов различных типов

Структура таблицы маршрутизации стека TCP/IP соответствует общим принципам построения таблиц маршрутизации, рассмотренным выше. Однако важно отметить, что вид таблицы IP-маршрутизации зависит от конкретной реализации стека TCP/IP. Приведем пример трех вариантов таблицы маршрутизации, с которыми мог бы работать маршрутизатор M1 в сети, представленной на рис. 13.2.

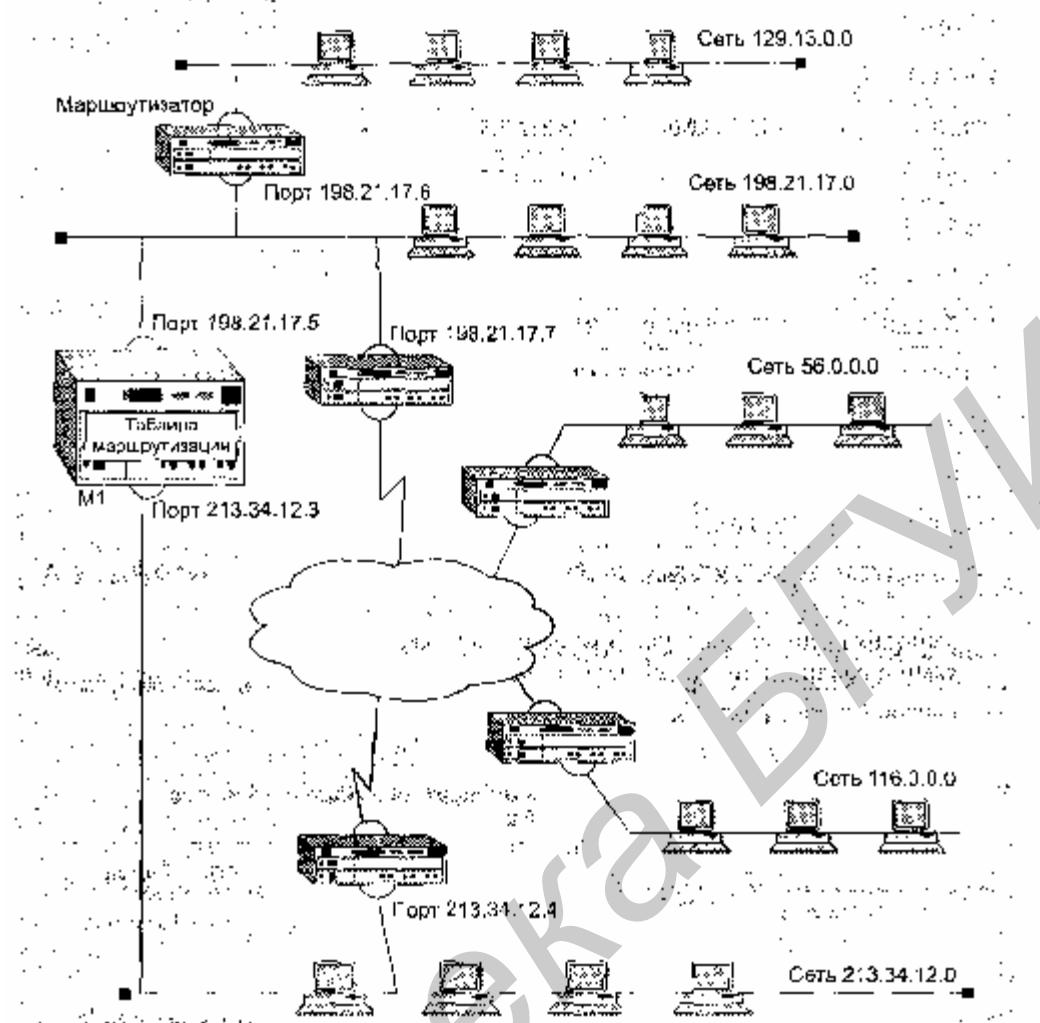


Рис. 13.2. Пример маршрутизуемой сети

Если представить, что в качестве маршрутизатора M1 в данной сети работает штатный программный маршрутизатор MPR операционной системы Microsoft Windows NT, то его таблица маршрутизации могла бы выглядеть так же, как табл. 13.1.

Таблица 13.1. Таблица программного маршрутизатора MPR Windows NT

Network address	Ad- dress	Netmask	Gateway Ad- dress	Ad- dress	Interface	Metric
127.0.0.0		255.0.0.0	127.0.0.1	127.0.0.1		1
0.0.0.0		0.0.0.0	198.21.17.7	198.21.17.5		1
56.0.0.0		255.0.0.0	213.34.12.4	213.34.12.3		15
116.0.0.0		255.0.0.0	213.34.12.4	213.34.12.3		13
129.13.0.0		255.255.0.0	198.21.17.6	198.21.17.5		2
198.21.17.0		255.255.255.0	198.21.17.5	198.21.17.5		1
198.21.17.5		255.255.255.2	127.0.0.1	127.0.0.1		1
198.21.17.255		255.255.255.2	198.21.17.5	198.21.17.5		1
213.34.12.0		255.255.255.0	213.34.12.3	213.34.12.3		1

213.34.12.3	255.255.255.2	127.0.0.1	127.0.0.1	1
213.34.12.255	255.255.255.2	213.34.12.3	213.34.12.3	1
224.0.0.0	224.0.0.0	198.21.17.6	198.21.17.6	1
224.0.0.0	224.0.0.0	213.34.12.3	213.34.12.3	1
255.255.255.255	255.255.255.2	198.21.17.6	198.21.17.6	1

Если на месте маршрутизатора M1 установить аппаратный маршрутизатор NetBuilder II компании 3Com, то его таблица маршрутизации для этой же сети может выглядеть как табл. 13.2.

Таблица 13.2. Таблица маршрутизации аппаратного маршрутизатора NetBuilder II компании 3Com

Destination	Mask	Gateway	Metric	Status	TT	Source
198.21.17.0	255.255.255	198.21.17.5	0	Up	-	Connected
213.34.12.0	255.255.255	213.34.12.3	0	Up	-	Connected
56.0.0.0	255.0.0.0	213.34.12.4	14	Up	--	Static
116.0.0.0	255.0.0.0	213.34.12.4	12	Up	--	Static
129.13.0.0	255.255.0.0	198.21.17.6	1	Up	160	RIP

Таблица 13.3 представляет собой таблицу маршрутизации для маршрутизатора M1, реализованного в виде программного маршрутизатора одной из версий операционной системы UNIX.

Таблица 13.3. Таблица маршрутизации Unix-маршрутизатора

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.0	127.0.01	UH	1	154	Lo0
Default	198.21.17.7	UG	5	43270	Le0
198.21.17.0	198.21.17.5	U	35	246876	Le0
213.34.12.0	213.34.12.3	и	44	132435	lei
129.13.0.0	198.21.1.7.6	UG	6	16450	Le0
56.0.0.0	213.34.12.4	UG	12	5764	lei
116.0.0.0	213.34.12.4	UG	21	23544	lei

ПРИМЕЧАНИЕ

Заметим, что поскольку между структурой сети и таблицей маршрутизации в принципе нет однозначного соответствия, то и для каждого из приведенных вариантов таблицы можно предложить свои «подварианты», отличающиеся выбранным маршрутом к той или иной сети. В данном случае внимание концентрируется на существенных различиях в форме представления маршрутной информации разными реализациями маршрутизаторов.

Назначение полей таблицы маршрутизации

Несмотря на достаточно заметные внешние различия, во всех трех таблицах есть все ключевые параметры, необходимые для работы маршрутизатора и рассмотренные ранее при обсуждении концепции маршрутизации.

К таким параметрам, во-первых, относится адрес сети назначения (столбцы «Destination» в маршрутизаторах NetBuilder и UNIX или «Network Address» в маршрутизаторе MPR). Заметим, что в некоторых случаях, когда маршрут к одному из узлов сети отличается от маршрута ко всем остальным узлам данной сети, в этом столбце указывается адрес данного конкретного узла назначения.

Вторым обязательным полем таблицы маршрутизации является адрес следующего маршрутизатора (столбцы «Gateway» в маршрутизаторах NetBuilder и UNIX или «Gateway Address» в маршрутизаторе MPR).

Третий ключевой параметр — адрес порта, на который нужно направить пакет, в некоторых таблицах указывается прямо (поле «Interface» в таблице Windows NT), а в некоторых — косвенно. Так, в таблице UNIX-маршрутизатора вместо адреса порта задается его условное наименование — 1o0 для порта с адресом 198.21.17.5, lei для порта с адресом 213.34.12.3 и 1o0 для внутреннего порта с адресом 127.0.0.1.

В маршрутизаторе NetBuilder II поле, обозначающее выходной порт в какой-либо форме, вообще отсутствует. Это объясняется тем, что адрес выходного порта всегда можно косвенно определить по адресу следующего маршрутизатора. Например, попробуем определить по табл. 13.2 адрес выходного порта для сети 56.0.0.0. Из таблицы следует, что следующим маршрутизатором для этой сети будет маршрутизатор с адресом 213.34.12.4. Адрес следующего маршрутизатора должен принадлежать одной из непосредственно присоединенных к маршрутизатору сетей, и в данном случае это есть 213.34.12.0. Маршрутизатор имеет порт, присоединенный к этой сети, и адрес этого порта 213.34.12.3 мы находим в поле «Gateway» второй строки таблицы маршрутизации, которая описывает непосредственно присоединенную сеть 213.34.12.0. Для непосредственно присоединенных сетей адресом следующего маршрутизатора всегда является адрес собственного порта маршрутизатора. Таким образом, адрес выходного порта для сети 56.0.0 - это адрес 213.34.12.3.

Остальные параметры, которые можно найти в представленных версиях таблицы маршрутизации, для принятия решения о пути следования пакета являются необязательными.

Наличие или отсутствие поля маски в таблице говорит о том, насколько современен данный маршрутизатор. Стандартным решением сегодня является использование поля маски в каждой записи таблицы, как это сделано в таблицах маршрутизаторов MPR Windows NT (поле «Netmask») и NetBuilder (поле «Mask»). Обработка масок при принятии решения маршрутизаторами будет рассмотрена ниже. Отсутствие поля маски говорит о том, что либо маршрутизатор рассчитан на работу только с тремя стандартными классами адресов, либо он использует для всех записей одну и ту же маску, что снижает гибкость маршрутизации.

Метрика, как видно из примера таблицы UNIX-маршрутизатора, является необязательным параметром. В остальных двух таблицах это поле имеется, одна-

ко оно используется только в качестве признака непосредственно подключенной сети. Действительно, если в таблице маршрутизации каждая сеть назначения упомянута только один раз, то поле метрики не будет приниматься во внимание при выборе маршрута, так как выбор отсутствует. А вот признак непосредственно подключенной сети маршрутизатору нужен, поскольку пакет для этой сети обрабатывается особым способом — он не передается следующему маршрутизатору, а отправляется узлу назначения. Поэтому метрика 0 для маршрутизатора Net-Builder или 1 для маршрутизатора MPR просто говорит маршрутизатору, что эта сеть непосредственно подключена к его порту, а другое значение метрики соответствует удаленной сети. Выбор значения метрики для непосредственно подключенной сети (1 или 0) является произвольным, главное, чтобы метрика удаленной сети отсчитывалась с учетом этого выбранного начального значения. В UNIX-маршрутизаторе используется поле признаков, где флаг G отмечает удаленную сеть, а его отсутствие — непосредственно подключенную.

Однако существуют ситуации, когда маршрутизатор должен обязательно хранить значение метрики для записи о каждой удаленной сети. Эти ситуации возникают, когда записи в таблице маршрутизации являются результатом работы некоторых протоколов маршрутизации, например протокола RIP. В таких протоколах новая информация о какой-либо удаленной сети сравнивается с имеющейся в таблице, и если метрика новой информации лучше имеющейся, то новая запись вытесняет имеющуюся. В таблице UNIX-маршрутизатора поле метрики отсутствует, и это значит, что он не использует протокол RIP.

Флаги записей присутствуют только в таблице UNIX-маршрутизатора. Они описывают характеристики записи.

- U — показывает, что маршрут активен и работоспособен. Аналогичный смысл имеет поле «Status» в маршрутизаторе NetBuilder.
- H — признак специфического маршрута к определенному хосту. Маршрут ко всей сети, к которой принадлежит данный хост, может отличаться от данного маршрута.
- G — означает, что маршрут пакета проходит через промежуточный маршрутизатор (gateway). Отсутствие этого флага отмечает непосредственно подключенную сеть.
- D — означает, что маршрут получен из сообщения Redirect (перенаправление) протокола ICMP. Этот признак может присутствовать только в таблице маршрутизации конечного узла. Признак означает, что конечный узел в какой-то предыдущей передаче пакета выбрал не самый рациональный следующий маршрутизатор на пути к данной сети, и этот маршрутизатор с помощью протокола ICMP сообщил, что все последующие пакеты к данной сети нужно отправлять через другой следующий маршрутизатор. Протокол ICMP может посыпать сообщения только узлу-отправителю, поэтому у промежуточного маршрутизатора этот признак встретиться не может. Признак никак не влияет на процесс маршрутизации, он только указывает администратору источник появления записи.

В таблице UNIX-маршрутизатора используется еще два поля, имеющих спра-

вочное значение. Поле «*Refcnt*» показывает, сколько раз на данный маршрут ссылались при продвижении пакетов. Поле «*Use*» отражает количество байтов, переданных по данному маршруту.

В таблице маршрутизатора NetBuilder также имеется два справочных поля. Поле времени жизни «*TTL*» (Time To Live) в данном случае никак не связано с временем жизни пакета. Здесь, как и во многих других таблицах, это поле предотвращает использование записей, содержимое которых может устаревать. Текущее значение поля *TTL* показывает оставшийся срок жизни записи в секундах, то есть время, в течение которого ее значение еще действительно. Поле «*Source*» отражает источник появления записи в таблице маршрутизации. Хотя это поле имеется не во всех маршрутизаторах, но практически для всех маршрутизаторов существуют три основных источника появления записи в таблице.

Источники и типы записей в таблице маршрутизации

Одним из источников записей в таблицу маршрутизации является программное обеспечение стека TCP/IP, которое при инициализации маршрутизатора автоматически заносит в таблицу несколько записей, в результате чего создается так называемая минимальная таблица маршрутизации.

Это, во-первых, записи о непосредственно подключенных сетях и маршрутизаторах по умолчанию, информация о которых появляется в стеке при ручном конфигурировании интерфейсов компьютера или маршрутизатора. К таким записям в приведенных примерах относятся записи о сетях 213.34.12.0 и 198.21.17.0, а также запись о маршрутизаторе по умолчанию — Default в UNIX-маршрутизаторе и 0.0.0.0 в маршрутизаторе MPR Windows NT. В приведенном примере таблицы для маршрутизатора NetBuilder маршрутизатор по умолчанию не используется, следовательно, при поступлении пакета с адресом назначения, отсутствующим в таблице маршрутизации, этот пакет будет отброшен.

Во-вторых, программное обеспечение автоматически заносит в таблицу маршрутизации записи об адресах особого назначения. В приведенных примерах таблица маршрутизатора MPR Windows NT содержит наиболее полный набор записей такого рода. Несколько записей в этой таблице связаны с особым адресом 127.0.0.0 (loopback), который используется для локального тестирования стека TCP/IP. Пакеты, направленные в сеть с номером 127.0.0.0, не передаются протоколом IP на канальный уровень для последующей передачи в сеть, а возвращаются в источник — локальный модуль IP. Записи с адресом 224.0.0.0 требуются для обработки групповых адресов (multicast address). Кроме того, в таблицу могут быть занесены адреса, предназначенные для обработки широковещательных рассылок (например, записи 8 и 11 содержат адрес отправки широковещательного сообщения в соответствующих подсетях, а последняя запись в таблице — адрес ограниченной широковещательной рассылки сообщения). Заметим, что в некоторых таблицах записи об особых адресах вообще отсутствуют.

Вторым источником появления записи в таблице является администратор, не-

посредственно формирующий запись с помощью некоторой системной утилиты, например программы route, имеющейся в операционных системах UNIX и Windows NT. В аппаратных маршрутизаторах также всегда имеется команда для ручного задания записей таблицы маршрутизации. Заданные вручную записи всегда являются статическими, то есть они не имеют срока жизни. Эти записи могут быть как постоянными, то есть сохраняющимися при перезагрузке маршрутизатора, так и временными, хранящимися в таблице только до выключения устройства. Часто администратор вручную заносит запись Default о маршрутизаторе по умолчанию. Таким же образом в таблицу маршрутизации может быть внесена запись о специфичном для узла маршруте. Специфичный для узла маршрут содержит вместо номера сети полный IP-адрес, то есть адрес, имеющий ненулевую информацию не только в поле номера сети, но и в поле номера узла. Предполагается, что для такого конечного узла маршрут должен выбираться не так, как для всех остальных узлов сети, к которой он относится. В случае когда в таблице есть разные записи о продвижении пакетов для всей сети и ее отдельного узла, при поступлении пакета, адресованного узлу, маршрутизатор отдаст предпочтение записи с полным адресом узла.

И, наконец, третьим источником записей могут быть протоколы маршрутизации, такие как RIP или OSPF. Подобные записи всегда являются динамическими, то есть имеют ограниченный срок жизни. Программные маршрутизаторы Windows NT и UNIX не показывают источник появления той или иной записи в таблице, а маршрутизатор NetBuilder использует для этой цели поле «Source».

В приведенном в табл. 13.2 примере первые две записи созданы программным обеспечением стека на основании данных о конфигурации портов маршрутизатора — об этом свидетельствует признак «Connected». Следующие две записи обозначены как «Static», что указывает на то, что их ввел вручную администратор. Последняя запись является следствием работы протокола RIP, поэтому в ее поле «TTL» имеется значение 160.

Маршрутизация без использования масок

Рассмотрим на примере IP-сети алгоритм работы средств сетевого уровня по продвижению пакета в составной сети (рис. 13.3). При этом будем считать, что все узлы сети, рассматриваемой в примере, имеют адреса, основанные на классах, а маски не используются. Особое внимание будет уделено взаимодействию протокола IP с протоколами разрешения адресов ARP и DNS.

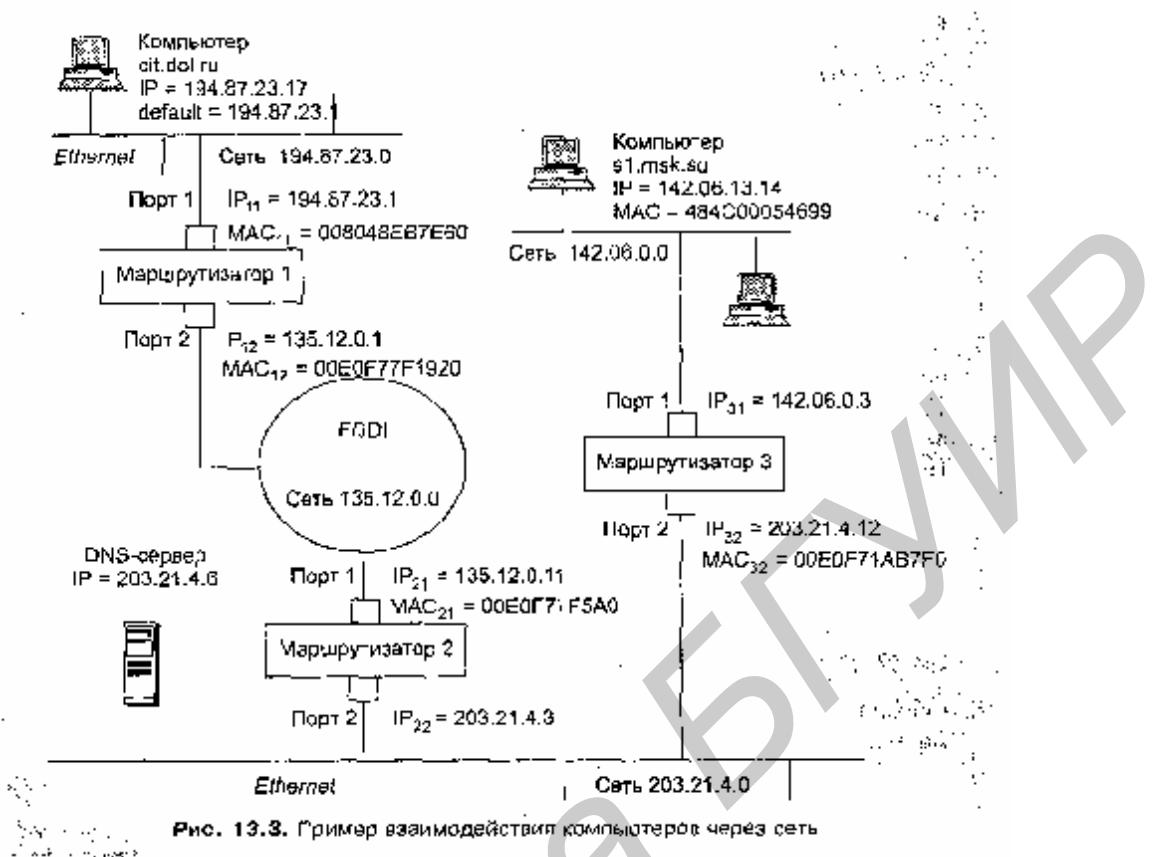


Рис. 13.3. Пример взаимодействия компьютеров через сеть

1. Итак, пусть пользователь компьютера cit.dol.ru, находящегося в сети Ethernet и имеющего IP-адрес 194.87.23.17 (адрес класса C), обращается по протоколу FTP к компьютеру s1.msk.su, принадлежащему другой сети Ethernet и имеющему IP-адрес 142.06.13.14 (адрес класса B):

> ftp s1.msk.su

2. Модуль FTP упаковывает свое сообщение в сегмент транспортного протокола TCP, который, в свою очередь, помещает свой сегмент в пакет протокола IP. В заголовке IP-пакета должен быть указан IP-адрес узла назначения. Так как пользователь компьютера cit.dol.ru указывает символьное имя компьютера s1.msk.su, то стек TCP/IP должен определить IP-адрес узла назначения самостоятельно.

3. При конфигурировании стека TCP/IP в компьютере cit.dol.ru был задан его собственный IP-адрес, IP-адрес маршрутизатора по умолчанию и IP-адрес DNS-сервера. Модуль IP может сделать запрос к серверу DNS, но обычно сначала просматривается локальная таблица соответствия символьных имен и IP-адресов. Такая таблица хранится чаще всего в виде текстового файла простой структуры — каждая его строка содержит запись об одном символьном имени и его IP-адресе. В ОС UNIX такой файл традиционно носит имя hosts и находится в каталоге /etc.

4. Будем считать, что компьютер cit.dol.ru имеет файл hosts, а в нем есть строка 142.06.13.14 s1.msk.su

5. Таким образом, разрешение имени выполняется локально, и протокол IP может теперь формировать IP-пакеты с адресом назначения 142.06.13.14 для

взаимодействия с компьютером s1.msk.su.

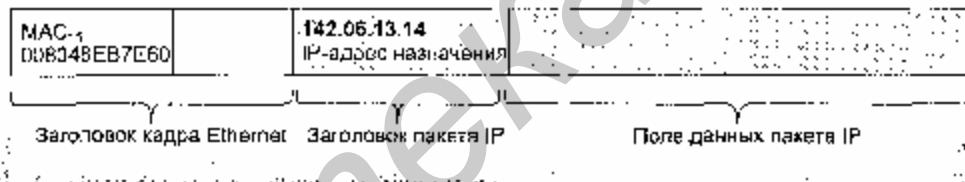
6. Модуль IP компьютера cit.dol.ru проверяет, нужно ли маршрутизировать пакеты с адресом 142.06.13.14. Так как адрес сети назначения (142.06.0.0) не совпадает с адресом (194.87.23.0) сети, которой принадлежит компьютер-отправитель, то маршрутизация необходима.

7. Компьютер cit.dol.ru начинает формировать кадр Ethernet для отправки IP-пакета маршрутизатору по умолчанию, IP-адрес которого известен — 194.87.23.1, но неизвестен MAC-адрес, необходимый для перемещения кадра в локальной сети. Для определения MAC-адреса маршрутизатора протокол IP обращается к протоколу ARP, который просматривает ARP-таблицу. Если в последнее время компьютер cit.dol.ru выполнял какие-либо межсетевые обмены, то скорее всего искомая запись, содержащая соответствие между IP- и MAC-адресами маршрутизатора по умолчанию, уже находится в кэш-таблице протокола ARP. Пусть в данном случае нужная запись была найдена именно в кэш-таблице:

194.87.23.1 008048EB7E60

8. Обозначим найденный MAC-адрес 008048EB7E60 в соответствии с номером маршрутизатора и его порта через MAC_{11} .

9. В результате компьютер cit.dol.ru отправляет по локальной сети пакет, упакованный в кадр Ethernet, имеющий следующие поля:

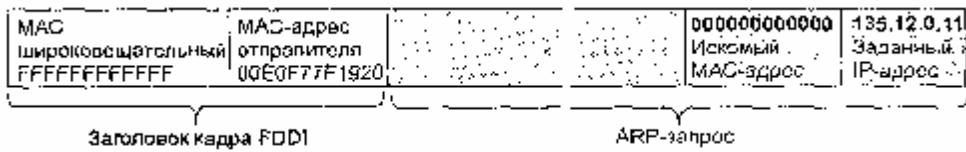


10. Кадр принимается портом 1 маршрутизатора 1 и соответствует протоколом Ethernet, так как MAC-узел этого порта распознает спой адрес MAC_{11} . Протокол Ethernet извлекает из этого кадра IP-пакет и передает его программному обеспечению маршрутизатора, реализующему протокол IP. Протокол IP извлекает из пакета адрес назначения 142.06.13.14 и просматривает записи своей таблицы маршрутизации. Пусть маршрутизатор 1 имеет в своей таблице маршрутизации следующую запись:

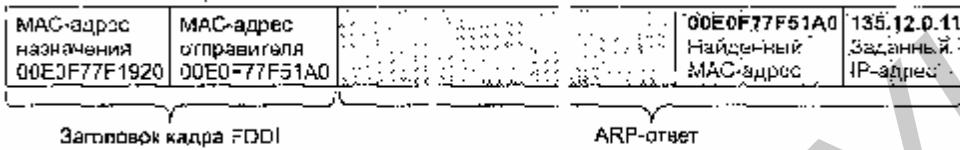
142.06.0.0 135.12.0.11 2

11. Эта запись говорит о том, что пакеты для сети 142.06.0.0 нужно передавать маршрутизатору 135.12.0.11, находящемуся в сети, подключенной к порту 2 маршрутизатора 1.

12. Маршрутизатор 1 просматривает параметры порта 2 и находит, что к нему подключена сеть FDDI. Так как сеть FDDI имеет значение MTU большее, чем сеть Ethernet, то фрагментация IP-пакета не требуется. Поэтому маршрутизатор 1 формирует кадр формата FDDI. На этом этапе модуль IP должен определить MAC-адрес следующего маршрутизатора по известному IP-адресу 135.12.0.11. Для этого он обращается к протоколу ARP. Допустим, что нужной записи в кэш-таблице не оказалось, тогда в сеть FDDI отправляется широковещательный ARP-запрос, содержащий наряду с прочей следующую информацию:



13. Порт 1 маршрутизатора 2 распознает свой IP-адрес и посыпает ARP-ответ по адресу запросившего узла:



14. Теперь, зная MAC-адрес следующего маршрутизатора 00EOF77F51AO, маршрутизатор 1 отсылает кадр FDDI по направлению к маршрутизатору 2. Заметим, что в поле IP-адреса назначения никаких изменений не произошло:



15. Аналогично действует модуль IP на маршрутизаторе 2. Получив кадр FDDI, он отбрасывает его заголовок, а из заголовка IP извлекает IP-адрес сети назначения и просматривает свою таблицу маршрутизации. Там он может найти запись о конкретной сети назначения:

142.06.0.0 203.21.4.12 2

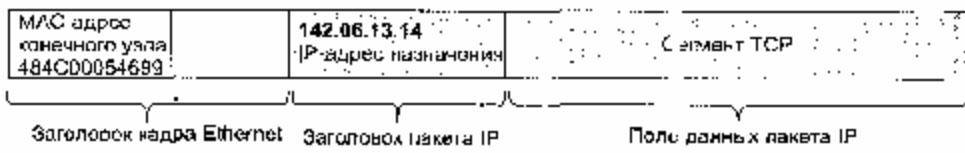
16. При отсутствии такой записи будет использована запись о маршрутизаторе по умолчанию:

default 203.21.4.12 2

17. Определив IP-адрес следующего маршрутизатора 203.21.4.12, модуль IP формирует кадр Ethernet для передачи пакета маршрутизатору 3 по сети Ethernet. С помощью протокола ARP он находит MAC-адрес этого маршрутизатора и помещает его в заголовок кадра. IP-адрес узла назначения, естественно, остается неизменным:



18. Наконец, после того как пакет поступил в маршрутизатор сети назначения (маршрутизатор 3), появляется возможность передачи этого пакета компьютеру назначения. Маршрутизатор 3 определяет, что пакет нужно передать в сеть 142.06.0.0, которая непосредственно подключена к его первому порту. Поэтому он посыпает ARP-запрос по сети Ethernet с IP-адресом компьютера s1.msk.su. ARP-ответ содержит MAC-адрес конечного узла, который модуль IP передает Канальному Протоколу для формирования кадра Ethernet:



19. Сетевой адаптер компьютера s1.msk.su захватывает кадр Ethernet, обнаруживает совпадение MAC-адреса, содержащегося в заголовке, со своим собственным адресом и направляет его модулю IP. После анализа полей IP-заголовка из пакета извлекаются данные, которые, в свою очередь, содержат сообщение вышеперечисленного протокола. Поскольку в данном примере рассматривается обмен данными по протоколу FTP, который использует в качестве транспортного протокола TCP, то в поле данных IP-пакета находится TCP-сегмент. Определив из TCP-заголовка номер порта, модуль IP переправляет сегмент в соответствующую очередь, из которой данный сегмент попадает программному модулю FTP-сервера.

Выводы

- Вид таблицы IP-маршрутизации зависит от конкретной реализации маршрутизатора, но, несмотря на достаточно сильные внешние различия, в таблицах всех типов маршрутизаторов есть все ключевые поля, необходимые для выполнения маршрутизации.
- К ключевым параметрам таблицы маршрутизации относятся адрес сети назначения и адрес следующего маршрутизатора. Еще один параметр — адрес порта маршрутизатора, на который нужно направить пакет, — в некоторых таблицах указывается прямо, а в некоторых — косвенно.
- Существует несколько источников, поставляющих записи в таблицу маршрутизации. Во-первых, при инициализации программное обеспечение стека TCP/IP заносит в таблицу записи о непосредственно подключенных сетях и маршрутизаторах по умолчанию, а также записи об особых адресах типа 127.0.0.0. Во-вторых, администратор вручную заносит статические записи о специфичных маршрутах или о маршрутизаторе по умолчанию. В-третьих, протоколы маршрутизации автоматически заносят в таблицу динамические записи об имеющихся маршрутах.
- Маршрутизация пакета очень упрощенно сводится к следующим действиям. Из кадра, поступившего на входной интерфейс маршрутизатора, извлекают IP-пакет и анализируют его IP-адрес назначения. По этому адресу из таблицы маршрутизации определяют IP-адрес следующего маршрутизатора и транслируют его в локальный адрес. Затем исходный IP-пакет упаковывают и кадр низлежащей технологии и отправляют на следующий маршрутизатор. И так происходит до тех пор, пока пакет не попадет в сеть назначения.

Маршрутизация с использованием масок

Использование масок для структуризации сети

Алгоритм маршрутизации усложняется, когда в систему адресации узлов вносятся дополнительные элементы — маски. В чем же причины отказа от хорошо себя зарекомендовавшего в течение многих лет метода адресации, основанного на классах? Основная из них — потребность в структуризации сетей в условиях дефицита нераспределенных номеров сетей.

Часто администраторы сетей испытывают неудобства из-за того, что количество централизованно выделенных им номеров сетей недостаточно для того, чтобы структурировать сеть надлежащим образом, например, разместить все слабо взаимодействующие компьютеры в разных сетях. В такой ситуации возможны два пути. Первый из них связан с получением от InterNIC или поставщика услуг Интернета дополнительных номеров сетей. Второй способ, употребляющийся чаще, связан с использованием технологии масок, которая позволяет разделять одну сеть на несколько сетей.

Допустим, администратор получил в свое распоряжение адрес класса В: 129.44.0.0. Он может организовать сеть с большим числом узлов, номера которых выбрать из диапазона 0.0.0.1-0.0.255.254 (с учетом того, что адреса из одних нулей и одних единиц имеют специальное назначение и не годятся для адресации узлов).

Однако ему не нужна одна большая неструктурированная сеть, производственная необходимость диктует администратору другое решение, в соответствии с которым сеть должна быть разделена на три отдельных подсети, при этом трафик в каждой подсети должен быть надежно локализован. Это позволит легче диагностировать сеть и проводить в каждой из подсетей особую политику безопасности.

Посмотрим, как решается эта проблема с помощью механизма масок. Вначале остановимся на случае, когда в сети используются маски одинаковой длины.

Итак, номер сети, который администратор получил от поставщика услуг, — 129.44.0.0 (10000001 00101100 00000000 00000000). В качестве маски было выбрано значение 255.255.192.0 (1111111111111111000000 00000000). После наложения маски на этот адрес число разрядов, интерпретируемых как номер сети, увеличилось с 16 (стандартная длина ноля номера сети для класса В) до 18 (число единиц в маске), то есть администратор получил возможность использовать два дополнительных бита для нумерации подсетей. Это позволяет ему сделать из одного централизованно заданного ему номера сети четыре:

129.44.0.0 (10000001 00101100 00000000 00000000)

129.44.64.0 (10000001 00101100 01000000 00000000)

129.44.128.0 (10000001 00101100 10000000 00000000)

129.44.192.0 (10000001 00101100 11000000 00000000)

Два дополнительных бита в номере сети часто интерпретируются как номера подсетей (subnet), и тогда четыре перечисленные выше подсети имеют номера 0 (00), 1 (01), 2 (10) и 3 (11) соответственно.

ПРИМЕЧАНИЕ

Некоторые программные и аппаратные маршрутизаторы не поддерживают номера подсетей, которые состоят либо только из одних нулей, либо только из

одних единиц. Например, для некоторых типов оборудования номер сети 129.44.0.0 с маской 255.255.192.0, использованный в нашем примере, окажется недопустимым, поскольку в этом случае разряды в поле номера подсети имеют значение 00. По аналогичным соображениям недопустимым может оказаться и номер сети 129.44.192,0 с тем же значением маски. Здесь номер подсети состоит только из единиц. Однако современные маршрутизаторы свободны от этих ограничений. Поэтому, принимая решение об использовании механизма масок, необходимо выяснить характеристики того оборудования, которым вы располагаете, чтобы соответствующим образом сконфигурировать маршрутизаторы и компьютеры сети.

В результате использования масок была предложена схема распределения адресного пространства, показанная на рис. 13.4.

Сеть, получившаяся в результате проведенной структуризации, показана на рис. 13.5. Весь трафик во внутреннюю сеть 129.44.0.0, направляемый из внешней сети, поступает через маршрутизатор M1. В целях структуризации информационных потоков во внутренней сети установлен дополнительный маршрутизатор M2.

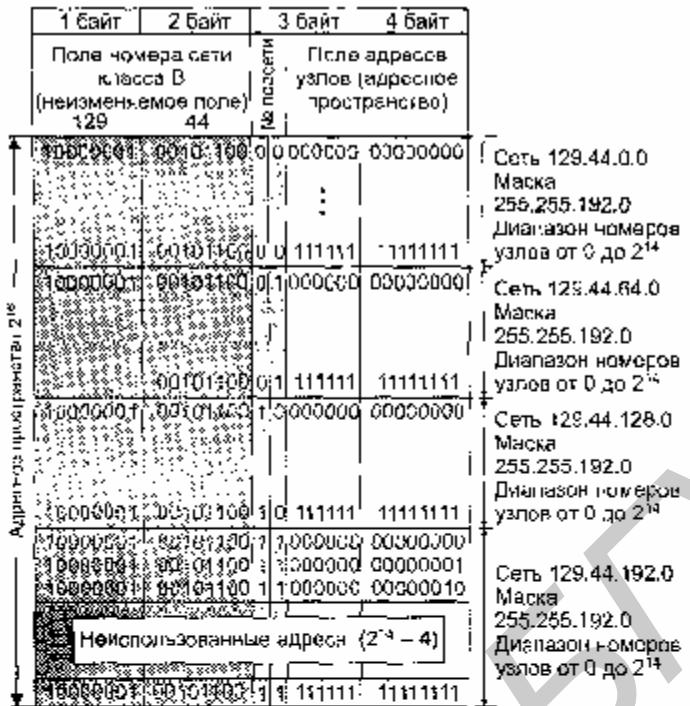


Рис. 13.4. Разделение адресов о гранж-сегментов на четыре равные части путем использования масок одинаковой длины

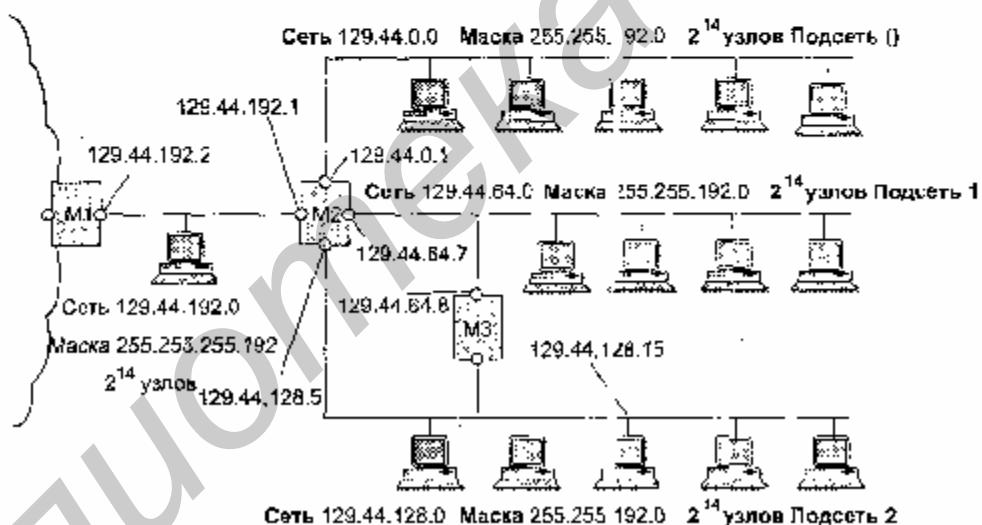


Рис. 13.5. Маршрутизация с использованием масок одинаковой длины

Все узлы были распределены по трем разным сетям, которым были присвоены номера 129.44.0.0, 129.44.64.0 и 129.44.128.0 и маски одинаковой длины — 255.255.192.0. Каждая из вновь образованных сетей была подключена к соответственно сконфигурированным портам внутреннего маршрутизатора M2. Кроме того, еще одна сеть (номер 129.44.192.0, маска 255.255.192.0) была выделена для создания соединения между внешним и внутренним маршрутизаторами. Особо отметим, что в этой сети для адресации узлов заняты всего два адреса 129.44.192.1 (порт маршрутизатора M2) и 129.44.192.2 (порт маршрутизатора M1), еще два адреса 129.44.192.0 и 129.44.192.255 являются особыми адресами. Следовательно, огромное число узлов (214-4) в этой подсети никак не ис-

пользуются.

Извне сеть по-прежнему выглядит, как единая сеть класса В, а на местном уровне это полноценная составная сеть, в которую входят три отдельные сети. Приходящий общий трафик разделяется местным маршрутизатором М2 между этими сетями в соответствии с таблицей маршрутизации. (Заметим, что разделение большой сети, имеющей один адрес старшего класса, например Л или В, с помощью масок обладает еще одним преимуществом по сравнению с использованием нескольких адресов стандартных классов для сетей меньшего размера, например С. Оно позволяет скрыть внутреннюю структуру сети предприятия от внешнего наблюдения и тем самым повысить ее безопасность.) Рассмотрим, как изменяется работа модуля IP, когда становится необходимым учитывать наличие масок. Во-первых, в каждой записи таблицы маршрутизации появляется новое поле — поле маски. Во-вторых, меняется алгоритм определения маршрута по таблице маршрутизации.

Напомним, что в столбце «Адрес назначения» могут быть помещены записи двух типов: номера сетей (в том случае, когда для всех узлов данной сети определен один и тот же маршрут) или IP-адрес узла, включающий и номер сети, и номер узла (в том случае, когда для данного узла определен особый маршрут, отличающийся от того, который ведет ко всем остальным узлам данной сети). Например, в табл. 13.4 в одной из строк указан адрес 129.44.128.15, для которого определен специфический маршрут, через маршрутизатор 129.44.64.8.

В тех строках таблицы, в которых в качестве адреса назначения указан полный IP-адрес узла, маска всегда имеет значение 255.255.255.255, то есть она состоит из одних двоичных единиц. Это несколько нарушает принятые нами условие одинаковой длины масок.

Итак, после того как IP-адрес извлекается из очередного полученного IP-пакета, необходимо определить адрес следующего маршрутизатора, на который надо передать пакет с этим адресом.

Для этого прежде всего определяется, нет ли в таблице маршрутизации специфического маршрута для этого адреса. С этой целью из каждой записи таблицы, в которой маска имеет значение 255.255.255.255, извлекается адрес назначения и сравнивается с адресом из пакета. Если в какой-либо строке совпадение произошло, то адрес следующего маршрутизатора для данного пакета берется из данной строки.

Если же была просмотрена вся таблица, но не произошло ни одного полного совпадения адресов, то начинается следующий цикл просмотра таблицы маршрутизации — поиск неспецифического маршрута, общего для группы узлов, относящихся к одной сети. Модуль IP последовательно просматривает все записи таблицы маршрутизации. С каждой записью производятся следующие действия.

1. Маска M, содержащаяся в данной записи, накладывается на IP-адрес узла назначения, извлеченный из пакета.
2. Полученное в результате число является номером сети назначения обрабатываемого пакета.
3. Полученный результат (номер сети назначения пакета) сравнивается со зна-

чением, которое помещено в поле «Адрес назначения» данной записи таблицы маршрутизации.

4. Если номера сетей совпадают, то пакет передается маршрутизатору, адрес которого помещен в соответствующем поле данной записи.

Теперь рассмотрим этот алгоритм на примере маршрутизации пакетов в сети, изображенной на рис. 13.5. Все маршрутизаторы внешней сети, встретив пакеты с адресами, начинающимися с 129.44, интерпретируют их как адреса класса В и направляют по маршрутам, ведущим к маршрутизатору M1. Маршрутизатор M1, в свою очередь, направляет весь входной трафик сети 129.44.0.0 на маршрутизатор M2, а именно на его порт 129.44.192.1.

Маршрутизатор M2 обрабатывает все поступившие на него пакеты в соответствии с таблицей маршрутизации (табл. 13.4).

Таблица 13.4. Таблица маршрутизатора M2 в сети с масками одинаковой длины

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.192.0	129.44.0.1	129.44.192.2	Подключена
129.44.64.0	255.255.192.0	129.44.64.7	129.44.64.7	Подключена
129.44.128.0	255.255.192.0	129.44.128.5	129.44.128.5	Подключена
129.44.192.0	255.255.192.0	129.44.192.1	129.44.192.1	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-
129.44.128.15	255.255.255.255	129.44.64.8	129.44.64.7	-

Первые четыре записи в таблице соответствуют внутренним подсетям, непосредственно подключенными к портам маршрутизатора M2.

Запись 0.0.0.0 с маской 0.0.0.0 соответствует маршруту по умолчанию. Действительно, любой адрес в пришедшем пакете после наложения на него маски 0.0.0.0 даст адрес сети 0.0.0.0, что совпадает с адресом, указанным в записи. Маршрутизатор выполняет сравнение с адресом 0.0.0.0 в последнюю очередь, в том случае когда пришедший адрес не дал совпадения ни с одной записью в таблице, отличающейся от 0.0.0.0. Записей с адресом 0.0.0.0 в таблице маршрутизации может быть несколько. В этом случае маршрутизатор передает пакет по всем таким маршрутам.

Последняя запись определяет специфический маршрут к узлу 129.44.128.15. В отличие от всех других узлов сети 129.44.128.0, к которым пакеты поступают с интерфейса 129.44.128.5 маршрутизатора M2, к данному узлу они будут приходить через маршрутизатор M3.

Пусть, например, с маршрутизатора M1 на порт 129.44.192.1 маршрутизатора M2 поступает пакет с адресом назначения 129.44.78.200. Модуль IP прежде всего сравнивает этот адрес с адресом 129.44.128.15, для которого определен специфический маршрут. Совпадения не произошло, поэтому модуль ТР начинает последовательно обрабатывать все строки таблицы, накладывая маски и сравнивая результаты до тех пор, пока не найдет совпадения номера сети в адресе назначения и в строке таблицы. Мaska из первой строки 255.255.192.0 наклады-

вается на адрес 129.44.78.200, в результате чего получается номер сети 129.44.64.0.

В двоичном виде эта операция выглядит следующим образом:

```
10000001.00101100.01001110. 11001000  
11111111.11111111.11000000.00000000  
10000001.00101100.01000000.00000000
```

Полученный номер 129.44.64.0 сравнивается с номером сети в первой строке таблицы — 129.44.0.0. Поскольку они не совпадают, то происходит переход к следующей строке. Теперь извлекается маска из второй строки (в данном случае она имеет такое же значение, но в общем случае это совсем не обязательно) и накладывается на адрес назначения пакета 129.44.78.200. Понятно, что из-за совпадения длины масок будет получен тот же номер сети 129.44.64.0. Этот номер совпадает с номером сети во второй строке таблицы, а значит, найден маршрут для данного пакета — он должен быть отправлен на порт маршрутизатора 129.44.64.7 в сеть, непосредственно подключенную к данному маршрутизатору.

Использование масок переменной длины

В предыдущем примере использования масок все подсети имеют одинаковую длину поля номера сети — 18 двоичных разрядов, и, следовательно, для нумерации узлов в каждой из них отводится по 14 разрядов (см. рис. 13.4 и 13.5). То есть все сети являются очень большими и имеют одинаковый размер. Однако в этом случае, как и во многих других, более эффективным явилось бы разбиение сети на подсети разного размера. В частности, большое число узлов, вполне желательное для пользовательской подсети, явно является избыточным для подсети, которая связывает два маршрутизатора по схеме «точка-точка». В этом случае требуется всего два адреса для адресации двух портом соседних маршрутизаторов. В предыдущем же примере для этой вспомогательной сети M1—M2 был использован номер, позволяющий адресовать 2^{14} узлов, что делает такое решение неприемлемо избыточным. Администратор может более рационально распределить имеющееся в его распоряжении адресное пространство с помощью масок переменной длины.

На рис. 13.6 приведен пример распределения адресного пространства, при котором избыточность имеющегося множества IP-адресов может быть сведена к минимуму. Половина из имеющихся адресов (2^{15}) отведена для сети с адресом 129.44.0.0 и маской 255.255.128.0. Следующая порция адресов, составляющая четверть всего адресного пространства (2^{14}), назначена для сети 129.44.128.0 с маской 255.255.192.0. Далее в пространстве адресов был «вырезан» небольшой фрагмент для создания сети, предназначеннной для связывания внутреннего маршрутизатора M2 с внешним маршрутизатором M1.

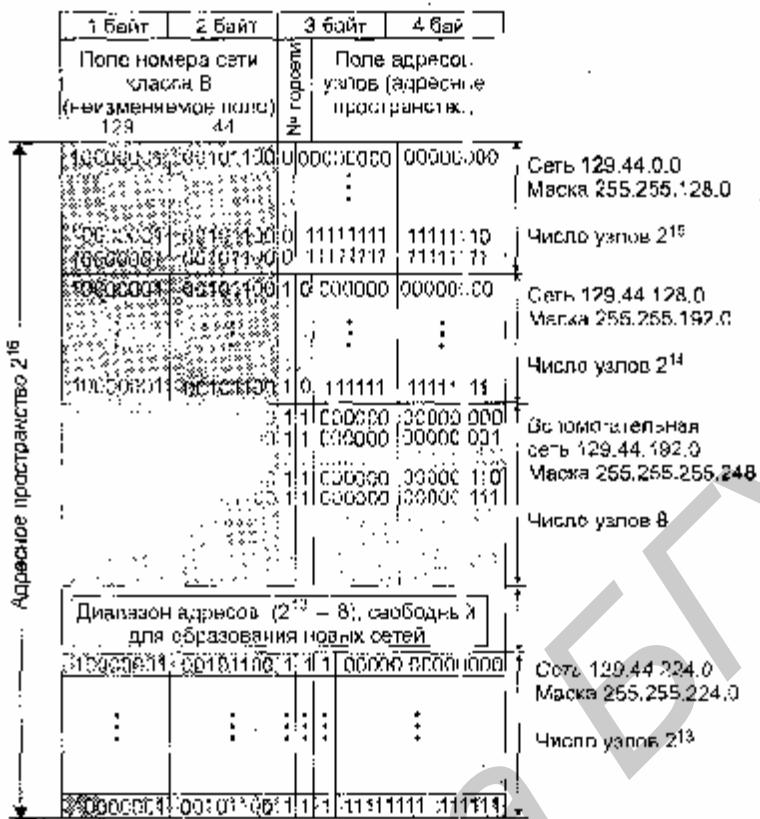


Рис. 13.6. Разделение адресного пространства сети класса В на сети разного размера путем использования масок переменной длины

В IP-адресе такой вырожденной сети для поля номера узла как минимум должны быть отведены два двоичных разряда. Из четырех возможных комбинаций номеров узлов: 00, 01, 10 и 11 два номера имеют специальное назначение и не могут быть присвоены узлам, но оставшиеся два 10 и 01 позволяют адресовать порты маршрутизаторов. В нашем примере сеть выбрана с некоторым запасом — на 8 узлов. Поле номера узла в таком случае имеет три двоичных разряда, маска в десятичной нотации имеет вид 255.255.255.248, а номер сети, как видно из рисунка, равен в данном конкретном случае 129.44.192.0. Если эта есть является локальной, то на ней могут быть расположены четыре узла помимо двух портов маршрутизаторов.

ПРИМЕЧАНИЕ

Заметим, что глобальным связям между маршрутизаторами типа «точка-точка» не обязательно давать IP-адреса, так как к такой сети не могут подключаться никакие другие узлы, кроме двух портов маршрутизаторов. Однако чаще всего подобной вырожденной сети все же дают IP-адрес. Это делается, например, для того, чтобы скрыть внутреннюю структуру сети и обращаться к ней по одному адресу входного порта маршрутизатора, в данном примере — по адресу 129.44.192.1. Кроме того, этот адрес может понадобиться при туннелировании немаршрутизуемых протоколов в IP-пакеты, что будет рассмотрено ниже.

Оставшееся адресное пространство администратор может «нарезать» на разное

количество сетей разного объема в зависимости от своих потребностей. Из оставшегося пула ($2^{14} - 4$) адресов администратор может образовать еще одну достаточно большую сеть с числом узлов 2^{13} . При этом свободными останутся почти столько же адресов ($2^{13} - 4$), которые также могут быть использованы для создания новых сетей. К примеру, из этого «остатка» можно образовать 31 сеть, каждая из которых равна размеру стандартной сети класса С, и к тому же еще несколько сетей меньшего размера. Ясно, что разбиение может быть другим, но в любом случае с помощью масок переменного размера администратор всегда имеет возможность гораздо рациональнее использовать все имеющиеся у него адреса.

На рис. 13.7 показана схема сети, структурированной с помощью масок переменной длины.

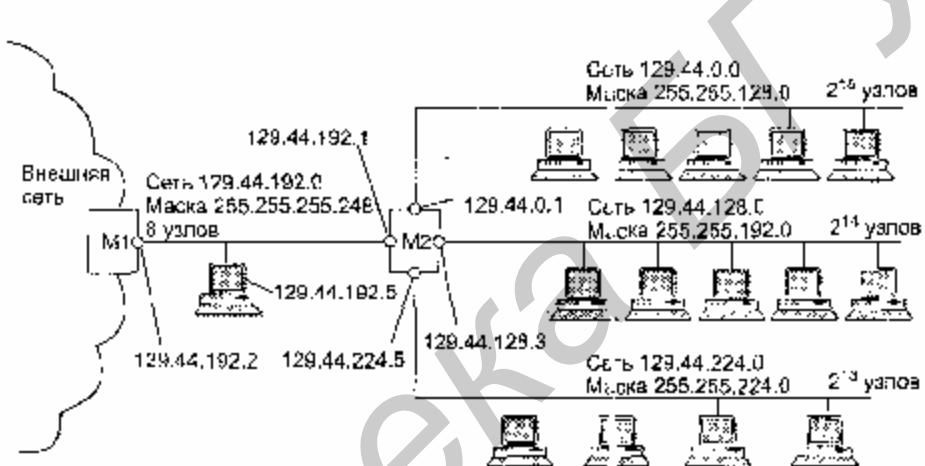


Рис. 13.7. Сеть, структурированная с помощью масок переменной длины

Таблица маршрутизации М2, соответствующая структуре сети, показанной на рисунке, содержит записи о четырех непосредственно подключенных сетях и запись о маршрутизаторе по умолчанию (табл. 13.5). Процедура поиска маршрута при использовании масок переменной аналогична подобной процедуре, описанной ранее для масок одинаковой длины.

Некоторые особенности масок переменной длины проявляются при наличии так называемых «перекрытий». Под перекрытием понимается наличие нескольких маршрутов к одной и той же сети или одному и тому же узлу. В этом случае адрес сети в пришедшем пакете может совпасть с адресами сетей, содержащихся сразу в нескольких записях таблицы маршрутизации.

Таблица 13.5. Таблица маршрутизатора М2 в сети с масками переменной длины

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.128.0	129.44.0.1	129.44.0.1	Подключена
129.44.128.0	255.255.192.0	129.44.128.3	129.44.128.3	Подключена
129.44.192.0	255.255.255.248	129.44.192.1	129.44.191.1	Подключена

129.44.224.0	255.255.224.0	129.44.224.5	129.44.224.5	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-

Рассмотрим пример. Пусть пакет, поступивший из внешней сети на маршрутизатор M1, имеет адрес назначения 129.44.192.5. Ниже приведен фрагмент таблицы маршрутизации маршрутизатора M1 (табл. 13.6). Первая из приведенных двух записей говорит о том, что все пакеты, адреса которых начинаются на 129.44, должны быть переданы на маршрутизатор M2. Эта запись выполняет агрегирование адресов всех подсетей, созданных на базе одной сети 129.44.0.0. Вторая строка говорит о том, что среди всех возможных подсетей сети 129.44.0.0 есть одна, 129.44.192.0, для которой пакеты можно направлять непосредственно, а не через маршрутизатор M2.

Таблица 13.6. Фрагмент таблицы маршрутизации маршрутизатора M1

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.0.0	129.44.192.1	129.44.191.2	2
129.44.192.0	255.255.255.248	129.44.192.2	129.44.192.2	Подключена

Если следовать стандартному алгоритму поиска маршрута по таблице, то сначала на адрес назначения 129.44.192.5 накладывается маска из первой строки 255.255.0.0 и получается результат 129.44.0.0, который совпадает с номером сети в этой строке. Но и при наложении на адрес 129.44.192.5 маски из второй строки 255.255.255.248 полученный результат 129.44.192.0 также совпадает с номером сети во второй строке. В таких случаях должно быть применено следующее правило: «Если адрес принадлежит нескольким подсетям в базе данных маршрутов, то продвигающий пакет маршрутизатор использует наиболее специфический маршрут, то есть выбирается адрес подсети, дающий большее совпадение разрядов».

В данном примере будет выбран второй маршрут, то есть пакет будет передан в непосредственно подключенную сеть, а не пойдет кружным путем через маршрутизатор M2.

Механизм выбора самого специфического маршрута является обобщением понятия «маршрут по умолчанию». Поскольку в традиционной записи для маршрута по умолчанию 0.0.0.0 маска 0.0.0.0 имеет нулевую длину, то этот маршрут считается самым неспецифическим и используется только при отсутствии совпадений со всеми остальными записями из таблицы маршрутизации.

ПРИМЕЧАНИЕ

В IP-пакетах при использовании механизма масок передается только IP-адрес назначения, а маска сети назначения не передается. Поэтому из IP-адреса при-

шедшего пакета невозможно выяснить, какая часть адреса относится к номеру сети, а какая — к номеру узла. Если маски во всех подсетях имеют один размер, то это не создает проблем. Если же для образования подсетей применяют маски переменной длины, то маршрутизатор должен как-то узнавать, каким адресам сетей какие маски соответствуют. Для этого используются протоколы маршрутизации, переносящие между маршрутизаторами не только служебную информацию об адресах сетей, но и о масках, соответствующих этим номерам. К таким протоколам относятся протоколы RIPv2 и OSPF, а вот, например, протокол RIP маски не распространяет и для использования масок переменной длины не подходит.

Технология бесклассовой междоменной маршрутизации

За последние несколько лет в Интернете многое изменилось: резко возросло число узлов и сетей, повысилась интенсивность трафика, изменился характер передаваемых данных. Из-за несовершенства протоколов маршрутизации обмен сообщениями об обновлении таблиц стал иногда приводить к сбоям магистральных маршрутизаторов из-за перегрузки при обработке большого объема служебной информации. Так, сегодня таблицы магистральных маршрутизаторов в Интернете могут содержать до нескольких сотен маршрута».

На решение этой проблемы была направлена, в частности, и технология бесклассовой междоменной маршрутизации (Classless Inter-Domain Routing, CIDR), впервые о которой было официально объявлено в 1993 году, когда были опубликованы документы RFC 1517, RFC 1518, RFC 1519 и RFC 1520.

Суть технологии CIDR заключается в следующем. Каждому поставщику услуг Интернета должен назначаться непрерывный диапазон в пространстве IP-адресов. При таком подходе адреса всех сетей каждого поставщика услуг имеют общую старшую часть — префикс, поэтому маршрутизация на магистралях Интернета может осуществляться на основе префиксов, а не полных адресов сетей. Агрегирование адресов позволит уменьшить объем таблиц в маршрутизаторах всех уровней, а следовательно, ускорить работу маршрутизаторов и повысить пропускную способность Интернета.

Деление IP-адреса на номер сети и номер узла в технологии CIDR происходит не на основе нескольких старших битов, определяющих класс сети (A, B или C), а на основе маски переменной длины, называемой поставщиком услуг. На рис. 13.8 показан пример некоторого пространства IP-адресов, которое имеется в распоряжении гипотетического поставщика услуг. Все адреса имеют одинаковые цифры в k старших разрядах — префикс. Оставшиеся p разрядов используются для дополнения неизменяемого префикса переменной частью адреса. Диапазон имеющихся у поставщика услуг адресов в таком случае составляет 2^p . Когда потребитель обращается к поставщику услуг с просьбой о выделении ему некоторого количества адресов, то в имеющемся пуле адресов «вырезается» непрерывная область S1, S2, S3 или S4 соответствующего размера. Причем границы каждой из областей выбираются не произвольно, а такими, чтобы значения всех старших разрядов (оставшихся после нумерации требуемого числа уз-

лов) было одинаковыми у всех адресов данного диапазона. Таким условиям могут удовлетворять только области, размер которых кратен степени двойки. А границы выделяемого участка должны быть кратны требуемому количеству узлов. Рассмотрим пример. Пусть поставщик услуг Интернета располагает пулом адресов в диапазоне 193.20.0.0-193.23.255.255 (НОО 0001.0001 0100.0000 0000.0000 0000-11000001. 0001 0111.1111 1111.1111 1111) с общим префиксом 193.20(11000001.0001 01) и маской, соответствующей этому префиксу 255.252.0.0.

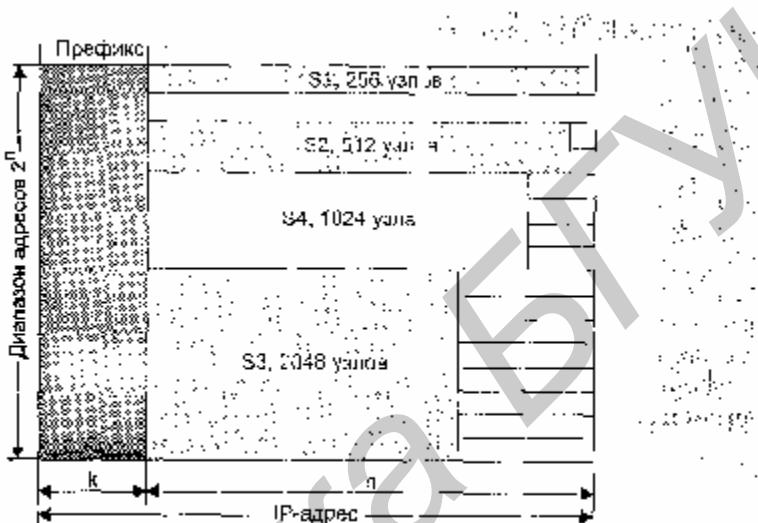


Рис. 13.8. Равномерное представление адресов на оси с сетью технологиях CIDR

Если абоненту этого поставщика услуг требуется совсем немного адресов, например 13, то поставщик мог бы предложить ему различные варианты: сеть 193.20.30.0, сеть 193.20.30.16 или сеть 193.21.204.48, все с одним и тем же значением маски 255.255.255.2⁰. Во всех случаях в распоряжении абонента для нумерации узлов имеются 4 младших бита.

Рассмотрим другой вариант, когда к поставщику услуг обратился крупный заказчик, сам, возможно, собирающийся оказывать услуги по доступу в Интернет. Ему требуется блок адресов в 4000 узлов. На нумерацию такого количества узлов пойдет 12 двоичных разрядов, следовательно, маска будет иметь значение 255.255.240.0 и размер выделенного пула адресов окажется несколько больше требуемого — 4096. Граница, с которой должен начинаться выделяемый участок, должна быть кратна размеру участка, то есть это могут быть любые адреса из следующих: 193.22.0.0, 193.22.16.0, 193.22.17.0, 193.22.18.0 и другие числа, оканчивающиеся на 12 нулей. В частности, поставщик услуг мог бы предложить потребителю, например, диапазон адресов 193.22.16.0-193.22.31.255 с маской 255.255.240.0. Агрегированный номер сети (префикс) имеет длину 20 двоичных разрядов и равен 193.22.16.0.20.

Администратор маршрутизатора М2 поместит в таблицу маршрутизации только по одной записи на каждого клиента, которому был выделен пул адресов, независимо от количества подсетей, организованных клиентом (рис. 13.9). Если клиент, получивший сеть 193.22.16.0, через некоторое время разделит ее адрес-

ное пространство в 4096 адресов на 8 подсетей, то в маршрутизаторе M2 первоначальная информация о выделенной ему сети не изменится.

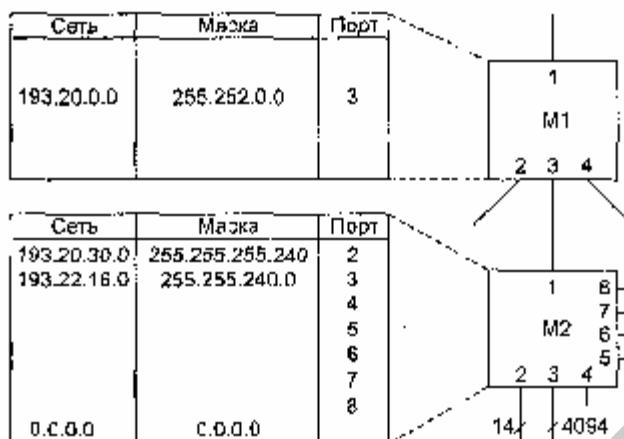


Рис. 13.9. Выигрыш в количестве записей маршрутизатора при использовании технологии CIDR

Для поставщика услуг верхнего уровня, поддерживающего клиентов через маршрутизатор M1, усилия поставщика услуг нижнего уровня по разделению его адресного пространства также будут незаметны. Запись 193.20.0.0 с маской 255.252.0.0 полностью описывает сети поставщика услуг нижнего уровня в маршрутизаторе M1.

Итак, внедрение технологии CIDR позволяет решить две основные задачи.

□ Более экономное расходование адресного пространства. Действительно, получая в свое распоряжение адрес сети, например, класса C, некоторые организации не используют весь возможный диапазон адресов просто потому, что в их сети гораздо меньше 255 узлов. Технология CIDR отказывается от традиционной концепции разделения адресов протокола IP на классы, что позволяет ограничивать потребителей-тем количеством адресов, которые им реально необходимы. Благодаря технологии CIDR поставщики услуг получают возможность «нарезать» блоки из выделенного им адресного пространства в точном соответствии с требованиями каждого клиента, при этом у клиента остается пространство для маневра на случай его будущего роста.

□ Уменьшение числа записей в таблицах маршрутизаторов за счет объединения маршрутов — одна запись в таблице маршрутизации может представлять большое количество сетей. Действительно, для всех сетей, номера которых начинаются с одинаковой последовательности цифр, в таблице маршрутизации может быть предусмотрена одна запись (см. рис. 13.9). Так, маршрутизатор M2 установленный в организации, которая использует технику CIDR для выделения адресов своим клиентам, должен поддерживать в своей таблице маршрутизации все 8 записей о сетях клиентов. А маршрутизатору M1 достаточно иметь одну запись обо всех этих сетях, на основании которой он передает пакеты с префиксом 193.20 маршрутизатору M2, распределяющему их по нужным портам.

Если все поставщики услуг Интернета будут придерживаться стратегии CIDR, то особенно заметный выигрыш получат магистральные маршрутизаторы.

Технология CIDR уже успешно используется в текущей версии IPv4 и поддерживается такими протоколами маршрутизации, как OSPF, RIP-2, BGP4. Предполагается, что эти же протоколы будут работать и с новой версией протокола IPv6. Следует отметить, что в настоящее время технология CIDR поддерживается магистральными маршрутизаторами Интернета, а не обычными хостами локальных сетей.

Необходимым условием эффективного использования CIDR является локализация адресов, то есть назначение адресов, имеющих совпадающие префиксы, сетям, располагающимся территориально по соседству. Только в таком случае трафик может быть агрегирован. К сожалению, сейчас распределение адресов носит во многом случайный характер. Кардинальный путь решения проблемы — перенумерация сетей. Однако эта процедура сопряжена с определенными временными и материальными затратами, и для ее проведения пользователей нужно каким-либо образом стимулировать. В качестве таких стимулов рассматривается, например, введение оплаты за строку в таблице маршрутизации или же за количество узлов в сети. Первое требование подводит потребителя к мысли получить у поставщика услуг такой адрес, чтобы маршрутизация трафика в его сеть шла на основании префикса и номер его сети не фигурировал больше в магистральных маршрутизаторах. Требование оплаты каждого адреса узла также может подтолкнуть пользователя решиться на перенумерацию с тем, чтобы получить ровно столько адресов, сколько ему нужно.

Выводы

- Эффективным средством структуризации IP-сетей являются маски. Маски позволяют разделить одну сеть на несколько подсетей (subnetting) или агрегировать несколько сетей в одну более крупную (supernetting).
- Маски одинаковой длины используются для деления сети на подсети равного размера, а маски переменной длины — для деления сети на подсети разного размера.
- Использование масок модифицирует алгоритм маршрутизации, что предъявляет особые требования к протоколам маршрутизации в сети, к техническим характеристикам маршрутизаторов и процедурам их конфигурирования.
- Значительная роль в будущем IP-сетей отводится технологии бесклассовой междоменной маршрутизации (CIDR), которая решает две основные задачи. Первая состоит в более экономном расходовании адресного пространства — благодаря CIDR поставщики услуг получают возможность «нарезать» блоки разных размеров из выделенного им адресного пространства в точном соответствии с требованиями каждого клиента. Вторая задача заключается в уменьшении числа записей в таблицах маршрутизации за счет объединения маршрутов — одна запись в таблице маршрутизации может представлять большое количество сетей с общим префиксом.

Фрагментация IP-пакетов

MTU — параметр технологии

Прежде всего отметим разницу между фрагментацией сообщений в узле-отправителе и динамическую фрагментацию сообщений в транзитных узлах сети — маршрутизаторах. Практически во всех стеках протоколов есть протоколы, отвечающие за фрагментацию сообщений прикладного уровня на такие части, которые укладывались бы в кадры канального уровня. Для этого они анализируют тип технологии нижнего уровня и определяют ее MTU (табл. 13.7). Напомним, что MTU — это максимально возможная длина дейтаграммы, которую та или иная технология может поместить в поле данных своей единицы передачи.

В стеке TCP/IP эту задачу решает протокол TCP, который разбивает поток байтов, передаваемый ему с прикладного уровня, на сегменты нужного размера (например, на 1460 байт, если на нижнем уровне данной сети работает протокол Ethernet). Поэтому протокол IP в узле-отправителе, как правило, не использует свои возможности по фрагментации пакетов. А вот на маршрутизаторе, когда пакет необходимо передать из сети с большим значением MTU в сеть с меньшим значением MTU, способности протокола IP выполнять фрагментацию становятся востребованными.

Значения MTU для различных технологий перечислены в табл. 13.7.

Таблица 13.7. Значения MTU различных технологий

Технология	MTU
DIX Ethernet	1 500 байт
Ethernet 802.3	1492 байт
Token Ring (IBM, 16 Мбит/с)	17914 байт
Token Ring (802.5, 4 Мбит/с)	4464 байт
FDDI	4352 байт
X.25	576 байт

Из таблицы видно, что значения MTU для наиболее популярных технологий существенно отличаются, а это значит, что в современной сети, которой свойственна гетерогенность, фрагментация не является редким явлением.

Параметры фрагментации

В функции протокола IP входит разбиение ставшего при передаче недопустимо длинным пакета на более короткие пакеты-фрагменты, каждый из которых должен быть снабжен полноценным IP-заголовком. Некоторые из полей заголовка — идентификатор, TTL, флаги DF и MF, смещение — непосредственно предназначены для проведения последующей сборки фрагментов в исходное сообщение.

Процедуры фрагментации и сборки протокола IP рассчитаны на то, чтобы пакет можно было разбить на практически любое количество частей, которые впо-

следствии могли бы быть вновь собраны. Фрагмент, путешествуя по сети, может вторично подвергнуться фрагментации. Получатель фрагмента использует поле идентификатор для того, чтобы опознать все фрагменты одного и того же пакета. Модуль IP, отправляющий пакет, устанавливает в поле идентификатор значение, которое должно быть уникальным для данной пары отправитель — получатель, а также время 771, в течение которого пакет может быть активным в сети.

'Поле смещения фрагмента сообщает получателю положение фрагмента в исходном пакете. Смещение фрагмента и длина определяют часть исходного пакета, принесенную этим фрагментом.

Флаг MF показывает появление последнего фрагмента. Модуль протокола IP, отправляющий неразбитый на фрагменты пакет, устанавливает в нуль флаг MF и смещение во фрагменте.

IP-пакет может быть помечен как не фрагментируемый (DF= 1), в таком случае модулю IP запрещено разделять его на части, даже если в этом случае он будет потерян. Если пакет, помеченный как не фрагментируемый, не может достигнуть получателя без фрагментации, то этот пакет просто уничтожается, а узлу-отправителю посыпается диагностическое ICMP-сообщение. Наличие такой возможности (запрета фрагментации) позволяет в некоторых случаях ускорить работу приложений. Для этого вначале необходимо исследовать сеть, определить максимальный размер пакета, который сможет пройти весь путь без фрагментации, а затем использовать пакеты такого или меньшего размера для обмена данными. Данное средство позволяет также предотвращать фрагментацию в тех случаях, когда хост-получатель не имеет достаточных ресурсов для сборки фрагментов.

Эти поля дают достаточное количество информации для разборки-сборки пакета.

Порядок сборки фрагментов

Итак, чтобы разделить пакет на фрагменты, модуль протокола IP, установленный, например, на маршрутизаторе, отводит несколько буферов для новых пакетов и копирует содержимое некоторых полей IP-заголовка из большого пакета в IP-заголовки всех новых пакетов-фрагментов. Данные из старого пакета делятся на соответствующее число частей, причем размер каждой, кроме самой последней, обязательно должен быть кратным 8 байт. Размер последней части данных равен полученному остатку.

Каждая из полученных частей данных помещается в новый пакет. Когда происходит фрагментация, то некоторые параметры IP-заголовка копируются в заголовки всех фрагментов, а другие остаются лишь в заголовке первого фрагмента. Процесс фрагментации может изменить значения некоторых полей, например контрольной суммы заголовка, флага MF, смещения фрагмента, длины IP-заголовка и общей длины пакета. В заголовок каждого пакета-фрагмента заносятся особые, соответствующие положению этого фрагмента значения смещения. Первый фрагмент будет иметь в поле смещение нулевое значение. Во всех

пакетах, кроме последнего, флаг MF устанавливается в единицу, а в последнем фрагменте—в нуль.

Сбор пакета происходит на хосте назначения. Для каждого фрагментированного пакета отводится отдельный буфер, в который принимающий протокол IP помещает IP-пакеты, имеющие одинаковые значения в полях идентификатора, IP-адресов отправителя и получателя, а также в поле номера порта (идентификатора) протокола, сообщение которого вложено в данный IP-пакет. Все эти признаки говорят протоколу IP, что данные пакеты являются фрагментами одного исходного пакета.

Отсюда следует, что идентификатор пакета должен быть выбран таким образом, чтобы он был уникален для данной пары отправитель—получатель, для вложенного протокола и в течение того времени, пока данный пакет (или любой его фрагмент) может существовать в составной IP-сети.

Очевидно, что модуль протокола IP, отправляющий пакеты, должен иметь таблицу идентификаторов, где каждая запись соотносится с каждым отдельным получателем, с которым осуществлялась связь, и указывает последнее значение максимального времени жизни пакета в IP-сети. Однако поскольку поле идентификатора допускает 65 536 различных значений, некоторые хосты просто могут использовать уникальные идентификаторы, не зависящие от адреса получателя.

В некоторых случаях целесообразно, чтобы идентификаторы IP-пакетов выбирались протоколами более высокого, чем IP, уровня. Например, в протоколе TCP предусмотрена повторная передача TCP-сегментов, по каким-либо причинам не дошедшими до адресата. Вероятность правильного приема увеличивалась бы, если бы при повторной передаче идентификатор для IP-пакета был бы тем же, что и в исходном IP-пакете, поскольку его фрагменты обоих пакетов могли бы использоваться для сборки правильного TCP-сегмента.

Собственно сборка заключается в помещении данных из каждого фрагмента в позицию, указанную в поле смещение заголовка пакета. После того как данные собраны, их можно передавать вышележащему протоколу, например TCP.

Пример фрагментации

Рассмотрим пример, в котором фрагментация осуществляется и на маршрутизаторе (рис. 13.10).

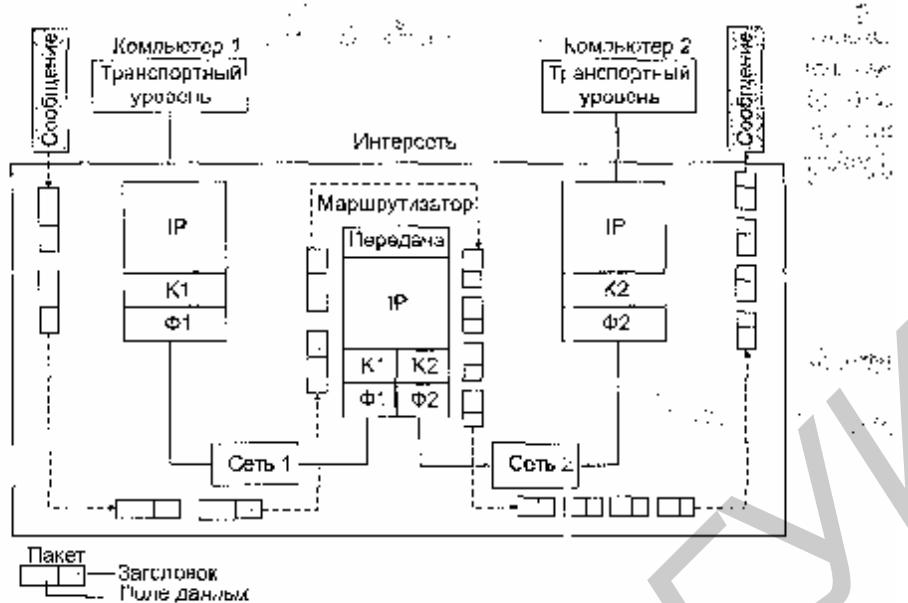


Рис. 13.10. Пример фрагментации IP-пакетов при передаче между системами с разным максимальным размером IP-кадра

Пусть компьютер 1 связан с сетью, имеющей значение MTU, равное 4352 байт, например с сетью FDDI. При поступлении на IP-уровень компьютера 1 сообщения от транспортного уровня размером в 5600 байт протокол IP делит его на два IP-пакета, устанавливая в первом пакете признак MF = 1 (не последний фрагмент) и присваивая пакету уникальный идентификатор, например 486. В первом пакете значение в поле смещения равно 0, а во втором — 2800. Признак MF во втором пакете равен нулю, показывая, что это последний фрагмент пакета. Общая величина каждого пакета-фрагмента составляет 2800 плюс 20 (размер IP-заголовка), то есть 2820 байт, что умещается в поле данных кадра FDDI. Далее модуль IP компьютера 1 передает эти пакеты своему сетевому интерфейсу (образуемому протоколами канального уровня К1 и физического уровня Ф1). Сетевой интерфейс отправляет кадры следующему маршрутизатору.

После того как кадры пройдут уровень сетевого интерфейса маршрутизатора (К1 и Ф1) и освободятся от заголовков FDDI, модуль IP по сетевому адресу определяет, что прибывшие два пакета нужно передать в сеть 2, которая является сетью Ethernet и имеет значение MTU, равное 1500. Следовательно, прибывшие IP-пакеты необходимо снова фрагментировать. Маршрутизатор извлекает поле данных из каждого пакета и делит его еще пополам, чтобы каждая часть уместилась в поле данных кадра Ethernet. Затем он формирует новые IP-пакеты, каждый из которых имеет длину $1400 + 20 = 1420$ байт, что меньше 1500 байт, поэтому они нормально помещаются в поле данных кадров Ethernet.

В результате в компьютер 2 по сети Ethernet приходят четыре IP-пакета с общим идентификатором 486, что позволяет протоколу IP, работающему в компьютере 2, правильно собрать исходное сообщение. Если пакеты пришли не в том порядке, в котором были посланы, смещение укажет правильный порядок их объединения.

Отметим, что IP-маршрутизаторы не собирают фрагменты пакетов в более крупные пакеты, даже если на пути встречается сеть, допускающая такое ук-

рупнение. Это связано с тем, что отдельные фрагменты сообщения могут перемещаться по интерсети по различным маршрутам, поэтому нет гарантии, что все фрагменты пройдут через какой-то один определенный маршрутизатор на их пути.

При приходе первого фрагмента пакета узел назначения запускает таймер, который определяет максимально допустимое время ожидания прихода остальных фрагментов этого пакета. Таймер устанавливается па максимальное из двух значений: первоначальное установочное время ожидания и время жизни, указанное в принятом фрагменте. Действительно, зачем ждать остальные фрагменты после того, как время жизни одного из фрагментов истечет?

ВНИМАНИЕ

Если хотя бы один фрагмент пакета не успеет прийти па хост назначения к моменту истечения таймера, то никаких действий по дублированию этого фрагмента не предпринимается, а все полученные к этому времени фрагменты пакета отбрасываются. Узлу, пославшему исходный пакет, направляется сообщение об ошибке с помощью протокола ICMP. Такому поведению протокола IP вполне соответствует его кredo «best effort» — стараться по возможности, но никаких гарантий не давать.

Протокол IP допускает возможность использования в пределах отдельной подсети ее собственных средств фрагментации, невидимых для протокола IP. Например, технология ATM делит поступающие IP-пакеты на ячейки с полем данных в 48 байт с помощью своего уровня сегментирования, а затем собирает ячейки в исходные пакеты на выходе из сети ATM. Но такие технологии, как ATM, являются скорее исключением, чем правилом.

Выводы

- Фрагментация — это деление поля данных исходного пакета на части и оформление этих частей в виде пакетов меньшего размера — фрагментов.
- Фрагментация выполняется при невозможности передать пакет в следующую по маршруту сеть из-за того, что его размер превосходит максимально допустимый размер единицы передаваемых данных в этой сети.
- Все фрагменты перемещаются по составной сети независимо друг от друга, поэтому некоторые из них могут прийти не в том порядке, в котором их отправляли, или вовсе потеряться.
- Сбор исходного пакета из фрагментов осуществляется модулем IP в узле назначения (и никогда на промежуточных маршрутизаторах).
- При сборке устанавливается таймер, ограничивающий время ожидания фрагментов. Если ко времени истечения таймера хотя бы один фрагмент не дойдет до узла назначения, то все фрагменты отбрасываются. Причем протокол IP не берет на себя задачу повторной передачи отдельных фрагментов или пакета целиком, оставляя эту проблему протоколам верхнего уровня.

Протоколы транспортного уровня TCP и UDP

Порты, мультиплексирование и демультиплексирование

В то время как задачей уровня межсетевого взаимодействия, к которому относится протокол IP, является передача данных между любой парой сетевых интерфейсов в составной сети, задача транспортного уровня, которую решают протоколы TCP и UDP, заключается в передаче данных между любой парой прикладных процессов, выполняющихся в сети.

После того как пакет средствами протокола IP доставлен на сетевой интерфейс компьютера-получателя, данные необходимо каким-то образом переправить конкретному процессу-получателю. Каждый компьютер может выполнять несколько процессов, более того, прикладной процесс тоже может иметь несколько точек входа, выступающих в качестве адресов назначения для пакетов данных.

Пакеты, поступающие на транспортный уровень, организуются операционной системой в виде множества очередей к точкам входа различных прикладных процессов. В терминологии TCP/IP такие системные очереди называются портами. Порт однозначно определяет приложение в пределах компьютера.

ВНИМАНИЕ

Порты приложений не стоит путать с портами (интерфейсами) оборудования. Существует два способа присвоения порта приложению — централизованный и локальный. За каждым из этих способов закреплен свой диапазон номеров портов: для централизованного от 0 до 1023, для локального — от 1023 до 65 535. Если процессы представляют собой популярные общедоступные службы, такие как FTP, telnet, HTTP, TFTP, DNS и т. п., то за ними закрепляются стандартные присвоенные (assigned) номера, иногда также называемые хорошо известными (well-known) номерами. Централизованное присвоение службам номеров портов выполняется организацией Internet Assigned Numbers Authority (IANA). Эти номера закрепляются и публикуются в стандартах Интернета (RFC 1700). Так, например, номер 21 закреплен за службой удаленного доступа к файлам FTP, а 23 — за службой удаленного управления telnet.

Для тех служб, которые еще не стали столь распространенными, чтобы закреплять за ними стандартные номера, номера портов выделяются локально. На каждом компьютере операционная система ведет список занятых и свободных номеров портов. При поступлении запроса от приложения, выполняемого на данном компьютере, операционная система выделяет ему первый свободный номер. Такие номера называют динамическими (dynamic). В дальнейшем все сетевые приложения должны адресоваться к данному приложению с указанием назначенного ему номера порта. После того как приложение завершит работу, выделенный ему локальный номер порта возвращается в список свободных и

может быть назначен другому приложению.

Все, что было сказано выше о портах, в равной степени относится к обоим протоколам транспортного уровня. В принципе нет никакой зависимости между назначением номеров для приложений, использующих протокол TCP, и приложений, работающих с протоколом UDP. Приложения, которые передают данные на уровень IP, используя протокол UDP, получают номера, называемые портами UDP. Аналогично, приложениям, обращающимся к протоколу TCP, выделяются порты TCP. В том и другом случаях это могут быть как назначенные, так и динамические номера. Диапазоны чисел, из которых выделяются номера портов TCP и UDP, совпадают: от 0 до 1023 для присвоенных и от 1024 до 65 535 для динамических номеров. Однако никакой связи между назначенными номерами нет. Например, одному приложению может быть назначен порт TCP 1750, а другому приложению — порт UDP 1750. В некоторых случаях, когда приложение может обращаться по выбору к протоколу TCP или UDP (например, таким приложением является DNS), ему, исходя из удобства запоминания, назначаются совпадающие номера портов TCP и UDP (в данном примере это номер 53).

Протоколы TCP и UDP ведут для каждого номера порта две очереди: очередь пакетов, поступающих в данный порт из сети, и очередь пакетов, отправляемых данным портом в сеть. Процедура приема данных протоколом TCP (или UDP), поступающих от нескольких прикладных служб, называется мультиплексированием. Обратная процедура распределения протоколом TCP (или UDP) поступающих от сетевого уровня пакетов между набором высоковысоконвейных служб, идентифицированных номерами портов, называется демультиплексированием (рис. 13.11).

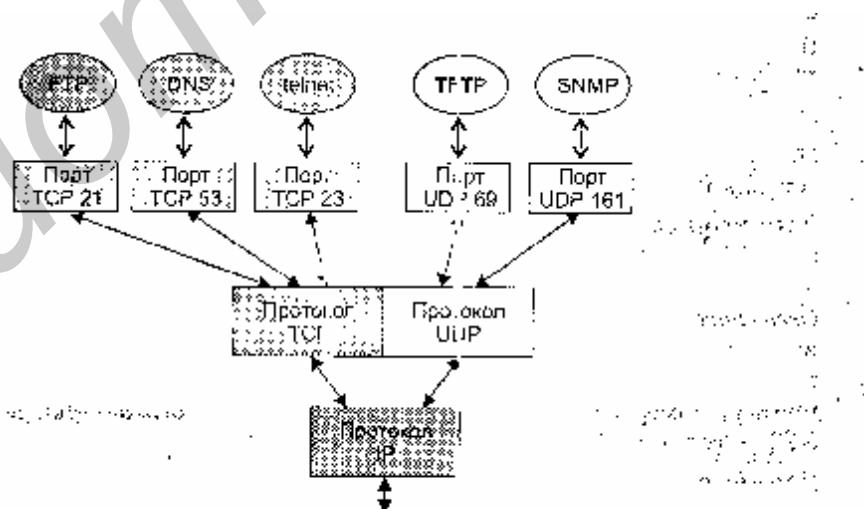


Рис. 13.11. Мультиплексирование и демультиплексирование на транспортном уровне

Надежный протокол доставки сообщений TCP

Если роль связующего звена между сетью и приложениями берут на себя оба протокола транспортного уровня (TCP и UDP), то задачу обеспечения надежно-

го канала обмена данными между прикладными процессами в составной сети решает только протокол TCP.

Протокол TCP работает непосредственно над протоколом IP и использует для транспортировки своих блоков данных этот дейтаграммный и поэтому потенциально ненадежный протокол. Надежность передачи данных протоколом TCP достигается за счет того, что он основан на установлении логических соединений между взаимодействующими процессами. До тех пор пока модули протокола TCP на обоих окончных хостах продолжают функционировать корректно, а составная сеть не распалась на несвязные части, ошибки в передаче данных на уровне протокола IP не будут влиять на правильность обмена хостов сегментами.

Информация, поступающая к протоколу TCP в рамках логического соединения от протоколов более высокого уровня, рассматривается протоколом TCP как неструктурированный поток байтов. Поступающие данные буферизуются средствами TCP. Для передачи на сетевой уровень из буфера «вырезается» некоторая непрерывная часть данных, которая называется сегментом (рис. 13.12). Сегменты могут быть разного размера, однако об их максимальных размерах участники соединения должны договориться. Заметим, что сегментом называют как единицу передаваемых данных в целом (поле данных и заголовок TCP), так и отдельно поле данных.

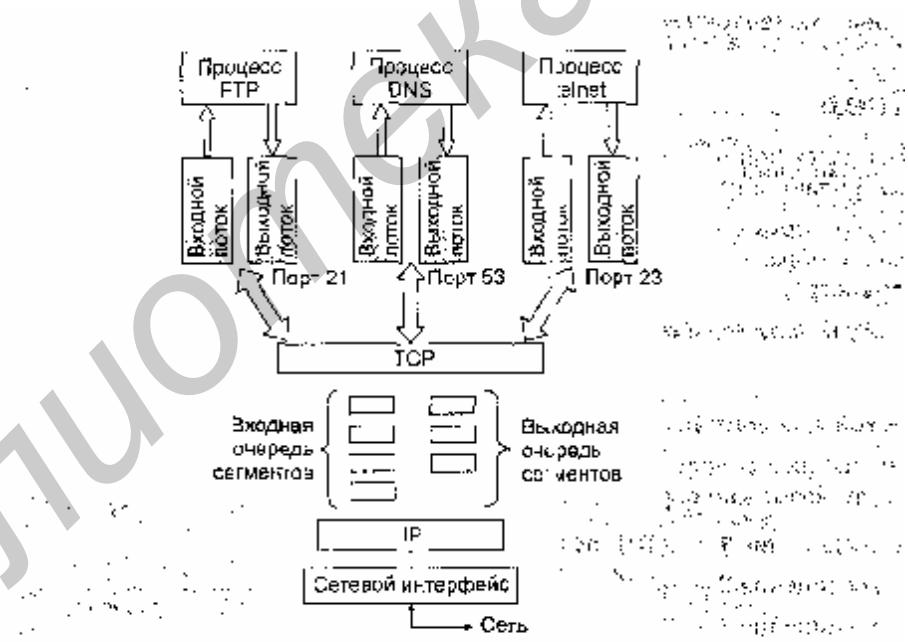


Рис. 13.12. Работа протокола TCP с буферами приложений

Протокол IP используется протоколом TCP в качестве транспортного средства. Перед отправкой сегментов протокол TCP помещает их в оболочку IP-пакета. В заголовок сегмента протокол TCP помещает номер порта назначения и номер порта отправителя.

На рис. 13.13 показано, как процессы, выполняющиеся на двух конечных узлах, устанавливают с помощью протокола TCP надежную связь через составную сеть, все узлы которой используют для передачи сообщений дейтаграммный

протокол IP. При необходимости протокол IP осуществляет любую фрагментацию и сборку сегментов TCP, требующуюся для осуществления передачи и доставки через множество сетей и промежуточных шлюзов.

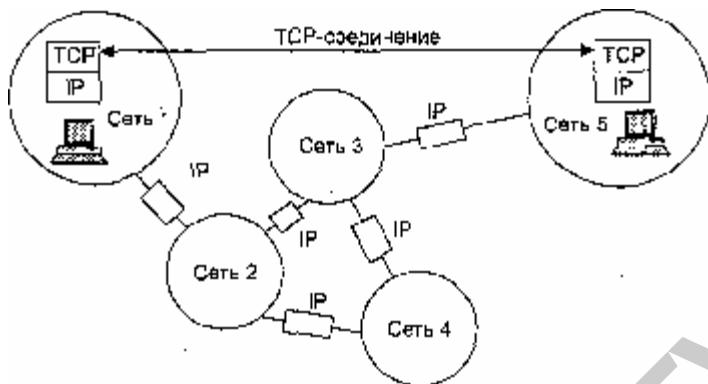


Рис. 13.13. TCP-соединение создает надежный канал связи между конечными узлами

Соединения

Для надежной передачи данных между двумя прикладными процессами предусматривается установление логического соединения.

Номер порта в совокупности с номером сети и номером конечного узла однозначно определяют прикладной процесс в сети. Этот набор идентифицирующих параметров {IP-адрес, номер порта} имеет название сокет (socket).

Каждый взаимодействующий процесс идентифицируется сокетом — парой (IP-адрес интерфейса, номер порта), а каждое соединение — парой сокетов взаимодействующих процессов. Каждый процесс одновременно может участвовать в нескольких соединениях,

Так, например, если (IP1, n1), (IP2, n2), (IP3, n3) — сокеты трех разных процессов, то возможно образование следующих соединений:

- соединение 1 - {(IP1, n1), (IP2, n2)};
- соединение 2 - {(IP1, n1), (IP3, n3)};
- соединение 3 - {(IP2, n2), (IP3, n3)}.

Каждая такая пара однозначно идентифицирует соединение. Сутью же понятия «соединение» является договоренность о параметрах, характеризующих процедуру обмена данными между двумя процессами. В протоколе TCP каждая сторона соединения посыпает противоположной стороне следующие параметры:

- максимальный размер сегмента, который она готова принимать;
- максимальный объем данных (возможно несколько сегментов), которые она разрешает другой стороне передавать в свою сторону, даже если та еще не получила подтверждения на предыдущую порцию данных;
- начальный порядковый номер байта, с которого она начинает отсчет потока данных в рамках данного соединения.

После того как в результате переговорного процесса модулей TCP с двух сторон соединения параметры процедуры обмена определены, одни из них остаются постоянными в течение всего сеанса связи, а некоторые адаптивно изменяются.

Поскольку соединения устанавливаются через ненадежную коммуникационную систему, основанную на протоколе IP, то во избежание ошибочной инициализации соединений используется специальная многошаговая процедура подтверждения связи.

Соединение в TCP позволяет вести передачу данных одновременно в обе стороны, то есть полнодуплексную передачу.

Последовательный номер и номер подтверждения

В рамках установленного соединения правильность передачи каждого сегмента должна подтверждаться квитанцией получателя. Квитирование — это один из традиционных методов обеспечения надежной связи. В протоколе TCP используется частный случай квитирования — алгоритм скользящего окна (идея этого алгоритма была изложена в главе 5).

Особенность использования алгоритма скользящего окна в протоколе TCP состоит в том, что, хотя единицей передаваемых данных является сегмент, окно определено на множестве нумерованных байтов неструктурированного потока данных, поступающих с верхнего уровня и буферизуемых протоколом TCP. При установлении соединения обе стороны договариваются о начальном номере байта, с которого будет вестись отсчет в течение всего данного соединения. У каждой стороны свой начальный номер. Сегменты могут иметь разную величину, но все равно идентификатором каждого сегмента является номер его первого байта. Нумерация байтов в пределах сегмента осуществляется так, чтобы первый байт данных сразу вслед за заголовком имел наименьший номер, а следующие за ним байты имели возрастающие номера (рис. 13.14).



Рис.13.14. Инкапсуляция сегмента TCP в пакете IP

Когда отправитель посыпает сегмент, он помещает в одно из полей заголовка TCP в качестве идентификатора сегмента номер первого байта данного сегмента, который называют последовательным номером (sequence number). На основании этого номера модуль TCP-получатель не только отличает данный сегмент от других, но и позиционирует полученный фрагмент относительно общего потока байтов, он может сделать также вывод, не является ли этот сегмент дубликатом и не было ли пропусков между полученным сегментом и предыдущим и т. д.

В качестве квитанции получатель сегмента отсылает ответное сообщение (сегмент), в которое помещает число, на единицу превышающее максимальный номер байта в полученном сегменте. Это число называют номером подтверждения (acknowledgement number). Номер подтверждения также является номером для следующего ожидаемого байта данных, передаваемого в обратном на-

правлении. Квитанция (подтверждение) в протоколе TCP посыпается только в случае правильного приема данных, отрицательные квитанции не посыпаются. Таким образом, отсутствие квитанции означает либо прием искаженного сегмента, либо потерю сегмента, либо потерю квитанции.

В протоколе TCP в одном и том же сегменте могут быть помещены и данные, которые приложение посыпает другой стороне, и квитанция, которой модуль TCP подтверждает получение данных.

Окно приема

Протокол TCP является дуплексным, то есть в рамках одного соединения регламентируется процедура обмена данными в обе стороны. Каждая сторона одновременно выступает и как отправитель, и как получатель. У каждой стороны есть пара буферов: один — для хранения принятых сегментов, другой — для сегментов, которые только еще предстоит отправить. При установлении соединения, а затем и в ходе передачи обе стороны, выступая в роли получателя, посыпают друг другу так называемые окна приема (receiver window). Каждая из сторон, получив окно приема, «понимает», сколько байтов ей разрешается отправить с момента получения последней квитанции. Другими словами, посылая окна приема, обе стороны пытаются регулировать поток байтов в свою сторону, сообщая своему «визави», какое количество байтов (начиная с номера байта, о котором уже была выслана квитанция) они готовы в настоящий момент принять.

На рис. 13.15 показан поток байтов, поступающий с верхнего уровня в выходной буфер протокола TCP. Из потока байтов модуль TCP «нарезает» последовательность сегментов и готовит их для отправки другому сокету. Для определенности на рисунке принято направление перемещения данных справа налево. В этом потоке можно указать несколько логических границ. Первая граница отделяет сегменты, которые уже были отправлены и на которые уже пришли квитанции. По другую сторону этой границы располагается окно размером W байт. Часть байтов, входящих в окно, составляют сегменты, которые также уже отправлены, но квитанции на них пока не получены. Оставшаяся часть окна — это сегменты, которые пока не отправлены, но могут быть отправлены, так как входят в пределы окна. И наконец, последняя граница указывает на начало последовательности сегментов, ни один из которых не может быть отправлен до тех пор, пока не придет очередная квитанция и окно не сдвинется вправо.

Если размер окна равен W , а последняя по времени квитанция содержала значение N , то отправитель может посыпать новые сегменты до тех пор, пока в очередной сегмент не попадет байт с номером $N + W$. Этот сегмент выходит за рамки окна, и передачу в таком случае необходимо приостановить до прихода следующей квитанции.

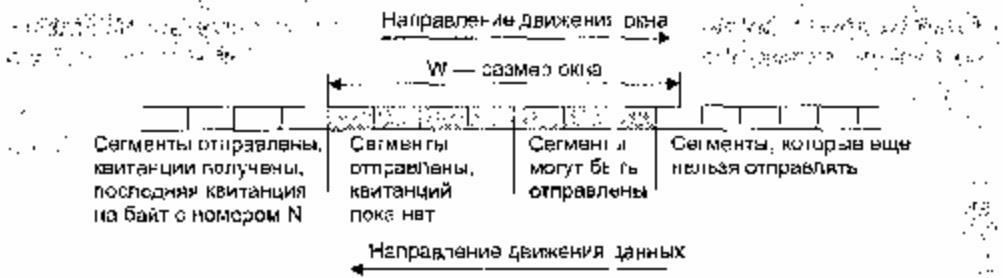
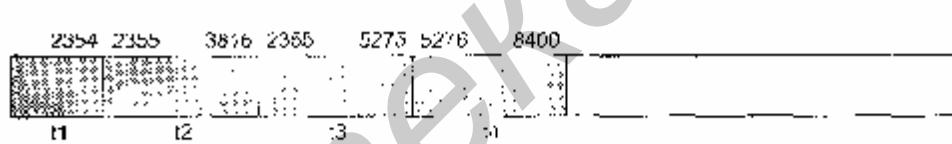


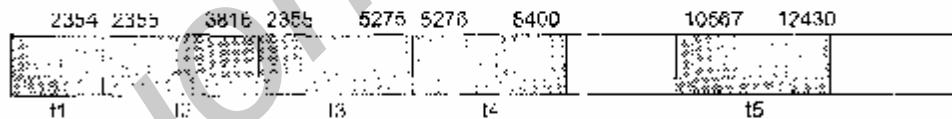
Рис. 13.15. Особенности реализации алгоритма скользящего окна в протоколе TCP

Накопительный принцип подтверждения

Получатель может послать квитанцию на получение сразу нескольких сегментов, если они образуют непрерывный поток байтов. Например, если в буфер, плотно без пропусков заполненный потоком байтов до 2354 включительно (рис. 13.16, а), последовательно поступили сегменты (2355-3816), (3817-5275) и (5276-8400), где цифры означают номера первых и последних байтов каждого сегмента, то получателю достаточно отправить только одну квитанцию на все три сегмента, указав в ней в качестве номера подтверждения значение 8401. Таким образом, процесс подтверждения является накопительным.



Плотное заголнение буфера. В момент t_4 передается квитанция



Неплотное заполнение буфера. В момент t_5 снова передается квитанция 8401

Рис. 13.16. Накопительный принцип квитирования

Еще один пример (рис. 13.16, б). Сегменты могут прийти к получателю не в том порядке, в котором были посланы, то есть в приемном буфере может образоваться «прогалина». Пусть, к примеру, после указанных выше трех сегментов вместо следующего по порядку отсылки сегмента (8401-10566) пришел сегмент (10567-12430). Очевидно, что послать в качестве номера подтверждения значение 12431 нельзя, потому что это бы означало, что получены все байты вплоть до 12430. Поскольку в потоке байтов образовался разрыв, получатель может только еще раз повторить квитанцию 8401, говоря тем самым, что все еще ожидает поступления потока байтов, начиная с 8401. Из этого примера видно, что в отличие от многих других протоколов, протокол TCP подтверждает получение не отдельных блоков данных, а непрерывной последовательности байтов.

Время ожидания квитанции

Когда протокол TCP передает в сеть сегмент, он «на всякий случай» помещает его копию в очередь повторной передачи и запускает таймер. Когда приходит подтверждение на этот сегмент, соответствующая копия удаляется из очереди. Если же подтверждение не приходит до истечения срока, то сегмент посыпается повторно. Может случиться, что повторный сегмент придет тогда, когда исходный сегмент уже окажется на месте, тогда дубликат будет попросту отброшен.

Выбор времени ожидания (тайм-аута) очередной квитанции является важной задачей, результат решения которой влияет на производительность протокола TCP. Тайм-аут не должен быть слишком коротким, чтобы по возможности исключить избыточные повторные передачи, которые снижают полезную пропускную способность системы. Но он не должен быть и слишком большим, чтобы избежать длительных простоев, связанных с ожиданием несуществующей или «заблудившейся» квитанции.

При выборе величины тайм-аута должны учитываться скорость и надежность физических линий связи, их протяженность и многие другие подобные факторы. В протоколе TCP тайм-аут определяется с помощью достаточно сложного адаптивного алгоритма, идея которого состоит в следующем. При каждой передаче засекается время от момента отправки сегмента до прихода квитанции о его приеме (время оборота). Получаемые значения времени оборота усредняются с весовыми коэффициентами, возрастающими от предыдущего замера к последующему. Это делается с тем, чтобы усилить влияние последних замеров. В качестве тайм-аута выбирается среднее время оборота, умноженное на некоторый коэффициент. Практика показывает, что значение этого коэффициента должно превышать 2. В сетях с большим разбросом времени оборота при выборе тайм-аута учитывается и дисперсия этой величины.

Управление окном приема

Размер окна приема зависит от наличия в данный момент места в буфере данных у принимающей стороны, а также от его скорости обработки данных. Поэтому в общем случае окна приема на разных концах соединения имеют разный размер. Например, можно ожидать, что сервер, вероятно обладающий большим буфером, пошлет клиентской станции окно приема большее, чем клиент серверу. В зависимости от состояния сети то одна, то другая сторона могут объявлять новые значения окон приема, динамически уменьшая и увеличивая их.

Варьируя величину окна, можно влиять на загрузку сети. С одной стороны, чем больше окно, тем большую порцию неподтвержденных данных можно послать в сеть. Но если пришло большее количество данных, чем может быть принято программой TCP, данные будут отброшены. Это приведет к излишним пересылкам информации и ненужному увеличению нагрузки на сеть и программу

TCP.

С другой стороны, указание окна малого размера может ограничить передачу данных скоростью, которая определяется временем путешествия по сети каждого посылаемого сегмента. Чтобы избежать применения малых окон, в RFC предлагается получателю данных откладывать реальное изменение окна до тех пор, пока свободное место не составит 20-40 % от максимально возможного объема памяти для этого соединения. Но и отправителю не стоит спешить с посылкой данных, пока окно не станет достаточно большим. Учитывая эти соображения, разработчики протокола TCP предложили схему, согласно которой при установлении соединения заявляется большое окно, но вследствие его размер существенно уменьшается. Существуют и другие, прямо противоположные алгоритмы настройки окна, когда вначале выбирается минимальное окно, а затем, если сеть справляется с предложенной нагрузкой, его размер резко увеличивается.

Управлять размером окна приема может не только сторона, которой предназначаются отправляемые в рамках этого окна данные, но и сторона-отправитель. Если отправляющая сторона фиксирует ненадежную работу линии связи (регулярно запаздывают квитанции, часто требуется повторная передача), то она может по собственной инициативе уменьшить окно. В таких случаях действует правило: в качестве размера окна выбирается минимальное из двух значений — значения, диктуемого приемной стороной, и значения, определяемого «на месте» отправителем.

Признаком перегрузки соединения TCP является возникновение очередей в промежуточных узлах-маршрутизаторах и в конечных узлах-компьютерах. При переполнении приемного буфера конечного узла «перегруженный» протокол TCP, отправляя квитанцию, помещает в нее новый, уменьшенный размер окна. Если он совсем отказывается от приема, то в квитанции указывается окно нулевого размера. Однако даже после этого приложение может послать сообщение на отказавшийся от приема порт. Для этого сообщение должно сопровождаться пометкой «срочно». В такой ситуации порт обязан принять сегмент, даже если для этого придется вытеснить из буфера уже находящиеся там данные. После приема квитанции с нулевым значением окна протокол-отправитель время от времени делает контрольные попытки продолжить обмен данными. Если протокол-приемник уже готов принимать информацию, то в ответ на контрольный запрос он посыпает квитанцию с указанием ненулевого размера окна.

Выводы

- В то время как задачей протокола IP является передача данных между любой парой сетевых интерфейсов в составной сети, задача протоколов TCP и UDP заключается в передаче данных между любой парой прикладных процессов.
- Системные очереди к точкам входа прикладных процессов называют портами. Порты идентифицируются номерами и однозначно определяют приложение в пределах компьютера.
- Приложения, которые передают данные на уровень IP, используя протокол

UDP, получают номера, называемые портами UDP. Аналогично, приложениям, обращающимся к протоколу TCP, выделяются порты TCP.

- Если процессы представляют собой популярные общедоступные службы, такие как FTP, telnet, HTTP, TFTP, DNS и т. п., то за ними централизовано закрепляются стандартные присвоенные (*assigned*) номера.
- Для тех служб, которые еще не стали столь распространенными, чтобы закреплять за ними стандартные номера, номера портов выделяются локальной операционной системой. Такие номера называют динамическими (*dynamic*).
- Процессы, выполняющиеся на двух конечных узлах, устанавливают по протоколу TCP надежную связь через составную сеть, все узлы которой используют для передачи сообщений ненадежный дейтаграммный протокол IP.
- Надежность передачи данных протоколом TCP достигается за счет того, что он основан на установлении логических соединений между взаимодействующими процессами.
- Прикладной процесс в сети однозначно определяется сокетом (*socket*), который представляет собой пару (IP-адрес, номер порта). В свою очередь, пара сокетов взаимодействующих процессов идентифицирует соединение. Каждый процесс одновременно может участвовать в нескольких соединениях.
- В рамках TCP-соединения происходит договорный процесс о следующих параметрах процедуры обмена данными между двумя процессами: максимальном размере сегмента, максимальном объеме данных, которые можно передавать без получения подтверждения (окне приема)", о начальном порядковом номере байта, с которого начинается отсчет потока данных в рамках данного соединения,
- Сторона-приемник передает стороне отправителю размер окна приема, исходя из того, с какой скоростью она сможет обрабатывать присылаемые данные. Однако управлять окном приема может и сторона-отправитель. Если отправляющая сторона фиксирует ненадежную работу линии связи, то она может по собственной инициативе уменьшить окно. В таких случаях действует правило: в качестве действующего размера окна выбирается минимальное из двух значений — значения, диктуемого приемной стороной, и значения, определяемого «на месте» отправителем.

ГЛАВА 14 Протоколы маршрутизации и маршрутизаторы

Протоколы маршрутизации предназначены для автоматического построения таблиц маршрутизации, на основе которых происходит продвижение пакетов сетевого уровня. Протоколы маршрутизации, в отличие от сетевых протоколов, таких как IP и IPX, не являются обязательными, так как таблица маршрутизации может быть построена администратором сети вручную. Однако в крупных сетях со сложной топологией и большим количеством альтернативных маршрутов протоколы маршрутизации выполняют очень важную и полезную работу, автоматизируя построение таблиц маршрутизации, динамически адаптируя текущий набор рабочих маршрутов к состоянию сети и повышая тем самым ее производительность и надежность.

Классификация протоколов маршрутизации

Назначение протоколов маршрутизации

Продвижение пакетов в составных сетях осуществляется на основе таблиц маршрутизации. Содержание такой таблицы зависит от технологии составной сети, так, вид таблиц сетей IP отличается от вида таблиц сетей IPX или X.25. Общим в таблицах маршрутизации является то, что в них содержится информация, достаточная для принятия решения о продвижении любого поступающего в маршрутизатор пакета. Как правило, каждая запись такой таблицы связывает адрес сети назначения с адресом или номером выходного интерфейса, на который нужно передать пакет с этим адресом. Каждый маршрутизатор сети имеет собственную таблицу маршрутизации, определяющую один шаг многошагового процесса перемещения пакета по сети.

Нетрудно заметить, что задача продвижения пакета от сети источника до сети назначения в каждом маршрутизаторе естественно распадается на две задачи:

- обработка пакета с помощью имеющейся таблицы маршрутизации;
- построение таблицы маршрутизации.

Назначение протоколов маршрутизации состоит в автоматическом решении второй задачи. Для этого маршрутизаторы сети обмениваются специальной служебной информацией о топологии составной сети, на основе которой каждый маршрутизатор выбирает маршруты к узлам назначения. Создаваемые таблицы маршрутизации обеспечивают рациональность маршрутов следования пакетов через сеть, при этом критерии выбора маршрутов могут быть различными. Напомним, что обычно выбирается «кратчайший» маршрут, где под расстоянием, проходимым пакетом, понимается либо количество промежуточных маршрутизаторов (хопов), либо комплексный показатель, учитывающий также номинальную пропускную способность каналов между маршрутизаторами, на-

дежность каналов или вносимые ими задержки. Протокол маршрутизации должен создавать в маршрутизаторах согласованные друг с другом таблицы маршрутизации, то есть такие, которые обеспечивают доставку пакета от исходной сети в сеть назначения за конечное число шагов. Можно представить и несогласованную пару таблиц, когда таблица маршрутизатора 1 показывает, что пакет для сети А нужно передать маршрутизатору 2, а таблица маршрутизатора 2 отправляет этот пакет маршрутизатору 1. Современные протоколы маршрутизации обеспечивают согласованность таблиц, однако это их свойство не абсолютно — при изменениях в сети, например при отказе каналов связи или самих маршрутизаторов, существуют периоды нестабильной работы сети, вызванной временной несогласованностью таблиц разных маршрутизаторов. Протоколу маршрутизации обычно нужно некоторое время, чтобы после нескольких итераций обмена служебной информацией все маршрутизаторы сети внесли изменения в свои таблицы и в результате таблицы снова стали согласованными.

Маршрутизация без таблиц

Прежде чем перейти к классификации протоколов маршрутизации, необходимо отметить, что существуют такие способы продвижения пакетов в составных сетях, которые вообще не требуют наличия таблиц маршрутизации в маршрутизаторах, а значит, и протоколов маршрутизации.

Наиболее простым способом передачи пакетов по сети является так называемая лавинная маршрутизация, когда каждый маршрутизатор передает пакет всем своим непосредственным соседям, кроме того, от которого его получил. Понятно, что это не самый рациональный способ, так как пропускная способность сети используется крайне расточительно, но он работоспособен (именно так мосты и коммутаторы локальных сетей поступают с кадрами, имеющими неизвестные адреса).

Другой вариант маршрутизации без таблицы — это маршрутизация, управляемая событиями (Event Dependent Routing), когда пакет к определенной сети назначения посыпается по маршруту, который уже приводил ранее к успеху (для данного адреса назначения). Это достаточно эффективный метод маршрутизации, но он требует наличия обратной связи, чтобы маршрутизатор-отправитель мог фиксировать факт успеха доставки пакета. Подобный метод маршрутизации использовался на этапе становления Интернета, когда перед отправкой пакета данных предварительно всем или нескольким соседям посыпались эхо-запросы по протоколу ICMP¹, а затем на основании времени прихода эхо-ответов выбирался сосед, у которого это время минимально. Данный способ хорошо зарекомендовал себя в сетях, работающих по протоколам с предварительным установлением соединения. Запрос на установление соединения может посыпаться нескольким соседям, а подтверждение установления соединения посыпается тому соседу, который первым пришлет ответ.

Маршрутизация, управляемая событиями, может сочетаться с таблицей маршрутизации. В такой таблице для каждой сети назначения указывается несколько возможных соседей, которым целесообразно направлять запрос на ус-

становление соединения (или тестирующий эхо-запрос). Подобный подход применяется в телефонных сетях, в которых указывается несколько возможных «направлений» передачи запроса на установление соединения, и эти запросы передаются сначала по первому из указанных направлений, пока не будет исчерпана его коммутационная емкость, затем по следующему и т. д.

Еще одним видом маршрутизации, не требующим наличия таблиц маршрутизации, является маршрутизация от источника (Source Routing). В этом случае отправитель помещает в пакет информацию о том, какие промежуточные маршрутизаторы должны участвовать в передаче пакета к сети назначения. На основе этой информации каждый маршрутизатор считывает адрес следующего маршрутизатора и, если он действительно является адресом его непосредственного соседа, передает ему пакет для дальнейшей обработки. Вопрос о том, как отправитель узнает точный маршрут следования пакета через сеть, остается открытым. Маршрут может прокладывать либо администратор вручную, либо узел отправитель автоматически, но в этом случае ему нужно поддерживать тот или иной протокол маршрутизации, который сообщает узлу о топологии и состоянии сети.

Адаптивная маршрутизация

В тех случаях, когда маршрутизация осуществляется на основании таблиц, различают статическую и адаптивную (динамическую) маршрутизацию. В первом случае таблицы составляются и вводятся в память каждого маршрутизатора вручную администратором сети, все записи в таблице имеют статус «статических» (static), что подразумевает бесконечный срок их жизни. При изменении состояния какого-нибудь элемента сети администратору необходимо срочно внести изменения в таблицы маршрутизации тех маршрутизаторов, для которых такое изменение требует смены маршрута (или маршрутов) следования пакетов — иначе сеть будет работать некорректно, и пакеты либо вообще перестанут доходить до сети назначения, либо их маршрут окажется не рациональным. Таким образом, при статической маршрутизации протоколы маршрутизации оказываются невостребованными, так как всю их работу выполняет один или несколько администраторов.

(¹ Internet Control Message Protocol (протокол управляющих сообщений Интернета). Этот протокол работает на сетевом уровне и используется маршрутизатором для оповещения конечного узла об ошибках, возникших при передаче IP-пакета от данного узла, а также для выполнения запросов и ответов о времени, дате, маске и других параметрах работы сети.)

Адаптивная маршрутизация обеспечивает автоматическое обновление таблиц маршрутизации после изменения конфигурации сети. Для адаптации таблиц как раз и нужны протоколы маршрутизации. Эти протоколы работают на основе алгоритмов, позволяющих всем маршрутизаторам собирать информацию о топологии связей в сети, оперативно отрабатывая все изменения конфигурации связей. В таблицах маршрутизации при адаптивной маршрутизации обычно имеется информация об интервале времени, в течение которого данный мар-

шрут будет оставаться действительным. Это время называют временем жизни маршрута (Time To Live, TTL). Если по истечении времени жизни существование маршрута не подтверждается протоколом маршрутизации, то он считается нерабочим, пакеты по нему больше не посылаются.

Протоколы маршрутизации могут быть распределенными и централизованными. При распределенном подходе в сети отсутствуют какие-либо выделенные маршрутизаторы, которые собирали бы и обобщали топологическую информацию: эта работа распределяется между всеми маршрутизаторами сети. Каждый маршрутизатор строит свою собственную таблицу маршрутизации, основываясь на данных, получаемых по протоколу маршрутизации от остальных маршрутизаторов сети.

При централизованном подходе в сети существует один маршрутизатор, который собирает всю информацию о топологии и состоянии сети от других маршрутизаторов. Затем этот выделенный маршрутизатор (который иногда называют сервером маршрутов) может выбрать несколько вариантов поведения. Он может построить таблицы маршрутизации для всех остальных маршрутизаторов сети, а затем распространить их по сети, чтобы каждый маршрутизатор получил собственную таблицу и в дальнейшем самостоятельно принимал решение о продвижении каждого пакета. Центральный маршрутизатор может также сообщить о выбранных маршрутах только конечным узлам (или пограничным маршрутизаторам), чтобы они направляли пакеты в сеть в соответствии с техникой маршрутизации от источника. В сети может быть не один, а несколько выделенных маршрутизаторов, каждый из которых обслуживает определенную группу подчиненных ему маршрутизаторов.

Адаптивные алгоритмы маршрутизации должны отвечать нескольким важным требованиям. Во-первых, они должны обеспечивать если не оптимальность, то хотя бы рациональность маршрута. Во-вторых, алгоритмы должны быть достаточно простыми, чтобы при их реализации не тратилось слишком много сетевых ресурсов, в частности они не должны требовать слишком большого объема вычислений или порождать интенсивный служебный трафик. И наконец, алгоритмы маршрутизации должны обладать свойством сходимости, то есть всегда приводить к однозначному результату за приемлемое время.

Адаптивные протоколы обмена маршрутной информацией, применяемые в настоящее время в вычислительных сетях, в свою очередь, делятся на две группы, каждая из которых связана с одним из следующих типов алгоритмов:

- дистанционно-векторные алгоритмы (Distance Vector Algorithms, DVA);
- алгоритмы состояния связей (Link State Algorithms, LSA).

Дистанционно-векторные алгоритмы

В дистанционно-векторных алгоритмах (Distance Vector Algorithms, DVA) каждый маршрутизатор периодически и широковещательно рассыпает по сети вектор, компонентами которого являются расстояния от данного маршрутизатора до всех известных ему сетей. Пакеты протоколов маршрутизации обычно называют объявлениями (advertisements), так как с их помощью маршрутизатор объ-

являет остальным маршрутизаторам известные ему сведения о сети. Под расстоянием обычно понимается число хопов. Возможна и другая метрика, учитывая не только число промежуточных маршрутизаторов, но и время прохождения пакетов по сети между соседними маршрутизаторами. При получении вектора от соседа маршрутизатор наращивает указанные в векторе расстояния до сетей на величину расстояния до данного соседа, добавляет к нему информацию об известных ему других сетях, о которых он узнал непосредственно (если они подключены к его портам) или из аналогичных объявлений других маршрутизаторов, а затем снова рассыпает новое значение вектора по сети. В конце концов, каждый маршрутизатор узнает информацию обо всех имеющихся в интерсете сетях и о расстоянии до них через соседние маршрутизаторы. Затем он выбирает из нескольких альтернативных маршрутов к каждой сети тот маршрут, который обладает наименьшей метрикой. Маршрутизатор, передавший информацию о данном маршруте, отмечается в таблице маршрутизации как следующий маршрутизатор (next hop). Для того чтобы адаптироваться к изменениям состояния сети, маршрутизаторы продолжают периодически сообщать друг другу информацию о наилучших маршрутах ко всем известным сетям. Если информация о какой-нибудь сети перестала поступать в маршрутизатор в течение определенного времени, то соответствующая запись из таблицы маршрутизации удаляется.

Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях. В больших сетях они засоряют линии связи интенсивным служебным периодическим трафиком, к тому же изменения конфигурации могут отрабатываться по этому алгоритму не всегда корректно, так как маршрутизаторы не имеют точного представления о топологии связей в сети, а располагают только обобщенной информацией — вектором дистанций, к тому же полученной через посредников. Работа маршрутизатора в соответствии с дистанционно-векторным протоколом напоминает работу моста, так как точной топологической картины сети такой маршрутизатор не имеет.

Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол RIP, который распространен в двух версиях — RIP IP, работающий с протоколом IP, и RIP IPX, работающий с протоколом IPX.

Алгоритмы состояния связей

Алгоритмы состояния связей (Link State Algorithms, LSA) обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей сети. Все маршрутизаторы работают на основании одинаковых графов, что делает процесс маршрутизации более устойчивым к изменениям конфигурации. «Широковещательная» рассылка (то есть передача пакета всем непосредственным соседям маршрутизатора) имеет место только при изменениях состояния связей, что происходит в надежных сетях не так часто. Вершинами графа являются как маршрутизаторы, так и объединяемые ими сети. Распространяемая по сети информация состоит из описания связей различных типов:

маршрутизатор-маршрутизатор, маршрутизатор—сеть.

Чтобы понять, в каком состоянии находятся линии связи, подключенные к его портам, маршрутизатор периодически обменивается короткими пакетами HELLO со своими ближайшими соседями. Объявления о состоянии связей не повторяются периодически, как у протоколов DVA, а передаются только в том случае, когда с помощью сообщений HELLO было установлено изменение состояния какой-либо связи. В результате служебный трафик, создаваемый протоколами LSA гораздо менее интенсивный, чем у протоколов DVA.

Протоколами, основанными на алгоритме состояния связей, являются протоколы IS-IS (Intermediate System to Intermediate System) стека OSI, OSPF (Open Shortest Path First) стека TCP/IP и протокол NLSP стека Novell.

Применение нескольких протоколов маршрутизации

В одной и той же сети могут одновременно работать несколько разных протоколов маршрутизации (рис. 14.1). Это означает, что на некоторых (не обязательно всех) маршрутизаторах сети установлено и функционирует несколько протоколов маршрутизации, но при этом, естественно, по сети взаимодействуют только одноименные протоколы. То есть если маршрутизатор 1 поддерживает, например, протоколы RIP и OSPF, маршрутизатор 2 — только RIP, а маршрутизатор 3 — только OSPF, то маршрутизатор 1 будет взаимодействовать с маршрутизатором 2 по протоколу RIP, с маршрутизатором 2 — по OSPF, а маршрутизаторы 2 и 3 вообще непосредственно друг с другом взаимодействовать не смогут.

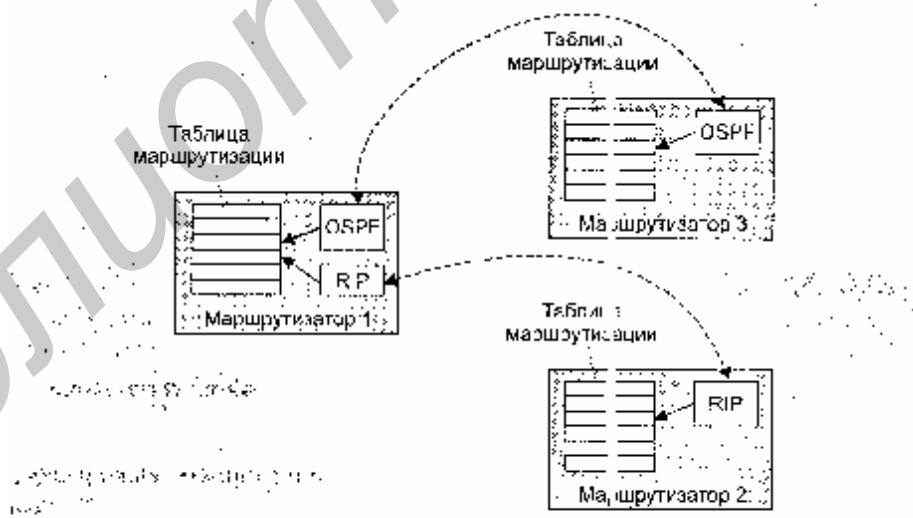


Рис. 14.1. Работа нескольких протоколов маршрутизации в одной сети

В маршрутизаторе, который поддерживает одновременно несколько протоколов, каждая запись в таблице является результатом работы одного из этих протоколов. Если о некоторой сети появляется информация от нескольких протоколов, то для однозначности выбора маршрута (а данные разных протоколов могут вести к различным рациональным маршрутам) устанавливаются приори-

теты протоколов маршрутизации. Обычно предпочтение отдается протоколам LSA как располагающим более полной информацией о сети по сравнению с протоколами DVA. В некоторых ОС в экранных и печатных формах в каждой записи таблицы маршрутизации имеется отметка о протоколе маршрутизации, с помощью которого эта запись получена. Но даже если эта отметка на экран и не выводится, она обязательно имеется во внутреннем представлении таблицы маршрутизации. По умолчанию каждый протокол маршрутизации, работающий на определенном маршрутизаторе, распространяет только ту информацию, которая была получена маршрутизатором по данному протоколу. Таким образом, если о маршруте к некоторой сети маршрутизатор узнал от протокола RIP, то и распространять по сети объявления об этом маршруте он будет с помощью протокола RIP.

Однако возникает вопрос: «Каким образом маршрутизатор, который не поддерживает все протоколы маршрутизации, применяемые в составной сети, узнает обо всех составляющих ее сетях?» Для того чтобы маршрутизатор распространял с помощью одного протокола маршрутизации информацию о маршрутах, полученную с помощью другого протокола маршрутизации, необходимо установить особый внутренний режим работы, называемый часто перераспределением (redistribute). Такой режим обеспечивает использование некоторым протоколом не только «своих» записей из таблицы маршрутизации, но и «чужих», полученных с помощью протокола маршрутизации, указанного при конфигурировании.

Как видно из описания, применение нескольких протоколов маршрутизации в пределах одной составной сети не является таким уж простым делом, от администратора требуется провести определенную работу по конфигурированию каждого маршрутизатора, чтобы создание таблиц выполнялось автоматически. В то же время во многих сетях существует необходимость в нескольких протоколах маршрутизации — причины, как и причины любой гетерогенности сети, могут быть разными, историческими, техническими, вплоть до личных предпочтений администраторов. При этом проблемы взаимодействия маршрутизаторов в такой неоднородной в отношении протоколов маршрутизации сети быстро усложняются с ростом сети. Очевидно, что для крупных составных сетей требуется качественно иное решение.

Такое решение было найдено для самой крупной на сегодня составной сети — сети Интернет.

Внешние и внутренние протоколы Интернета

Большинство протоколов маршрутизации, применяемых в современных сетях с коммутацией пакетов, ведут свое происхождение от Интернета и его предшественницы — сети ARPANET. Для того чтобы дальше продвинуться в их понимании, полезно познакомиться со структурой Интернета.

Интернет состоит не только из сетей, но и из более крупных объединений — автономных систем. Автономная система (Autonomous System) — это совокупность сетей под единым административным управлением, обеспечивающим

общую для всех входящих в автономную систему маршрутизаторов политику маршрутизации. Обычно автономной системой управляет один поставщик услуг Интернета, самостоятельно выбирая, какие протоколы маршрутизации использовать в некоторой автономной системе и каким образом выполнять между ними перераспределение маршрутной информации.

Основная цель деления Интернета на автономные системы — обеспечение многоуровневого подхода к маршрутизации. До введения автономных систем маршрутизация предполагала двухуровневый подход — маршрут на сетевом уровне прокладывался не непосредственно между узлами, а между группами узлов — сетями, а маршрутизацию внутри сети обеспечивали технологии более низких уровней. То есть маршрут определял последовательность прохождения сетей.

С появлением автономных систем появляется третий, верхний уровень маршрутизации — маршрут сначала выбирается на уровне автономных систем, а затем уже на уровне сетей, входящих в эти автономные системы. Подобно сетям все автономные системы централизованно нумеруются. Номер системы состоит из 16 разрядов, и этот номер никак не связан с префиксами IP-адресов сетей, входящих в автономную систему.

В соответствии с этой концепцией Интернет выглядит как набор взаимосвязанных автономных систем, каждая из которых состоит из взаимосвязанных сетей (рис. 14.2).

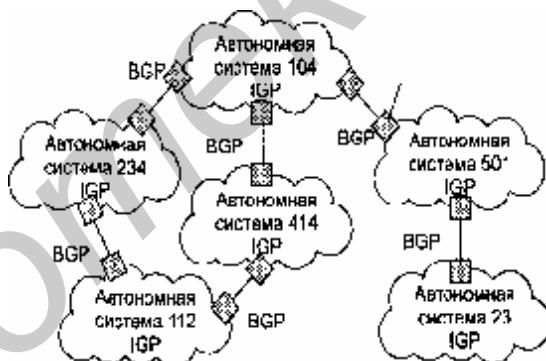


Рис. 14.2. Автономные системы Интернета

Автономные системы соединяются шлюзами, называемыми внешними маршрутизаторами (exterior gateway). Важно, что между внешними маршрутизаторами разрешается использовать только один протокол маршрутизации, причем не произвольный, а тот, который в данное время признается сообществом Интернета в качестве стандартного для внешних маршрутизаторов. Подобный протокол маршрутизации называется внешним протоколом маршрутизации (Exterior Gateway Protocols, EGP). В настоящее время таким стандартным внешним протоколом маршрутизации является протокол Border Gateway Protocol version 4 (BGP-4). Всё остальные протоколы являются внутренними протоколами маршрутизации (Interior Gateway Protocols, IGP).

Внешний протокол маршрутизации отвечает за выбор маршрута как последовательности автономных систем. В качестве адреса следующего маршрутизатора указывается адрес точки входа в соседнюю автономную систему. За маршрут

внутри автономной системы отвечают внутренние протоколы маршрутизации, которые в случае транзитной автономной системы определяют точную последовательность маршрутизаторов от точки входа в автономную систему до точки выхода из нее.

Автономные системы составляют магистраль Интернета. Концепция автономных систем экранирует от администраторов магистрали Интернета проблемы маршрутизации пакетов на более низком уровне — уровне сетей. Для администратора магистрали неважно, какие протоколы маршрутизации применяются внутри автономных систем, для него существует единственный протокол маршрутизации — BGP-4, который он и конфигурирует.

Магистраль Интернета не всегда выглядела так, как показано на рис. 14.2. На ранней стадии развития Интернета магистраль представляла собой одну автономную систему, которая управлялась компанией, выбранной правительством США для этой цели. Все остальные автономные системы подключались к магистрали, образуя древовидную структуру. По мере развития Интернета для повышения его эффективности возникла потребность в прямых связях между автономными системами, так что сегодня Интернет представляет собой объединение равноправных автономных систем с произвольной топологией связей.

Далее мы будем использовать термины «маршрутизатор» и «шлюз» как синонимы, чтобы отдать дань уважения традиционной терминологии Интернета и в то же время не забывать и более современные термины.

Выводы

- Протоколы маршрутизации генерируют для каждого маршрутизатора согласованные друг с другом таблицы маршрутизации, то есть такие, которые позволяют обеспечить доставку пакета по рациональному маршруту от исходной сети в сеть назначения за конечное число шагов. Для этого маршрутизаторы сети обмениваются специальной служебной информацией о топологии составной сети.
- Существуют способы продвижения пакетов в составных сетях, которые не требуют построения таблиц маршрутизации, к ним относятся:
 - лавинная маршрутизация, когда каждый маршрутизатор передает пакет всем своим непосредственным соседям, кроме того, от которого его получил;
 - маршрутизация, управляемая событиями, когда пакет к определенной сети назначения посыпается по маршруту, который уже приводил ранее к успеху (для данного адреса назначения);
 - маршрутизация от источника — в этом случае отправитель помещает в пакет информацию о том, какие промежуточные маршрутизаторы должны участвовать в передаче пакета к сети назначения.
- В тех случаях, когда маршрутизация осуществляется на основании таблиц, различают статическую и адаптивную (динамическую) маршрутизацию:
 - при статической маршрутизации таблицы составляются и вводятся в память каждого маршрутизатора вручную администратором сети;
 - адаптивная (динамическая) маршрутизация обеспечивает автоматическое об-

новление таблиц маршрутизации после изменения конфигурации сети.

□ Протоколы маршрутизации могут быть распределенными и централизованными:

- при распределенном подходе каждый маршрутизатор строит собственную таблицу маршрутизации, основываясь на данных, получаемых по протоколу маршрутизации от остальных маршрутизаторов сети;

- при централизованном подходе в сети существует один маршрутизатор, который собирает всю информацию о топологии и состоянии сети от других маршрутизаторов и строит таблицы маршрутизации для всех остальных маршрутизаторов сети.

□ Адаптивные протоколы маршрутизации делятся на две группы, каждая из которых связана с одним из следующих типов алгоритмов:

- в дистанционно-векторных алгоритмах каждый маршрутизатор периодически и широковещательно рассыпает по сети вектор, компонентами которого являются расстояния от данного маршрутизатора до всех известных ему сетей;

- алгоритмы состояния связей обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей сети.

□ Протоколами, основанными на алгоритме состояния связей, являются протоколы IS-IS стека OSI, OSPF стека TCP/IP и протокол NLSP стека Novell.

□ Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол RIP, который распространен в двух версиях — RIP IP, работающий с протоколом IP, и RIP IPX, работающий с протоколом IPX.

□ Протоколы маршрутизации Интернета делятся на внешние и внутренние. Внешние протоколы (EGP, BGP) переносят маршрутную информацию между автономными системами, а внутренние (IGP) применяются только в пределах определенной автономной системы.

Дистанционно-векторный протокол RIP

Построение таблицы маршрутизации

Протокол RIP (Routing Information Protocol) является внутренним протоколом маршрутизации дистанционно-векторного типа, он представляет собой один из наиболее ранних протоколов обмена маршрутной информацией и до сих пор чрезвычайно распространен в вычислительных сетях ввиду простоты реализации. Кроме версии RIP для сетей TCP/IP существует также версия RIP для сетей IPX/SPX компании Novell.

Для IP имеются две версии протокола RIP: первая и вторая. Протокол RIPv1 не поддерживает масок, то есть он распространяет между маршрутизаторами только информацию о номерах сетей и расстояниях до них, а информацию о масках этих сетей не распространяет, считая, что все адреса принадлежат к стандартным классам А, В или С. Протокол RIPv2 передает информацию о масках сетей, поэтому он в большей степени соответствует требованиям сего-

дняшнего дня. Так как при построении таблиц маршрутизации работа версий 2 и 1 принципиально не отличается, то в дальнейшем для упрощения записей будет описываться работа первой версии.

В качестве расстояния до сети стандарты протокола RIP допускают различные виды метрик: хопы, метрики, учитывающие пропускную способность, вносимые задержки и надежность сетей (то есть соответствующие признакам D, T и R в поле «Качество сервиса» IP-пакета), а также любые комбинации этих метрик. Метрика должна обладать свойством аддитивности, когда метрика составного пути равна сумме метрик составляющих этого пути. В большинстве реализаций RIP используется простейшая метрика — количество хопов, то есть количество промежуточных маршрутизаторов, которые нужно преодолеть пакету до сети назначения.

Рассмотрим процесс построения таблицы маршрутизации с помощью протокола RIP на примере составной сети, представленной на рис. 14.3.

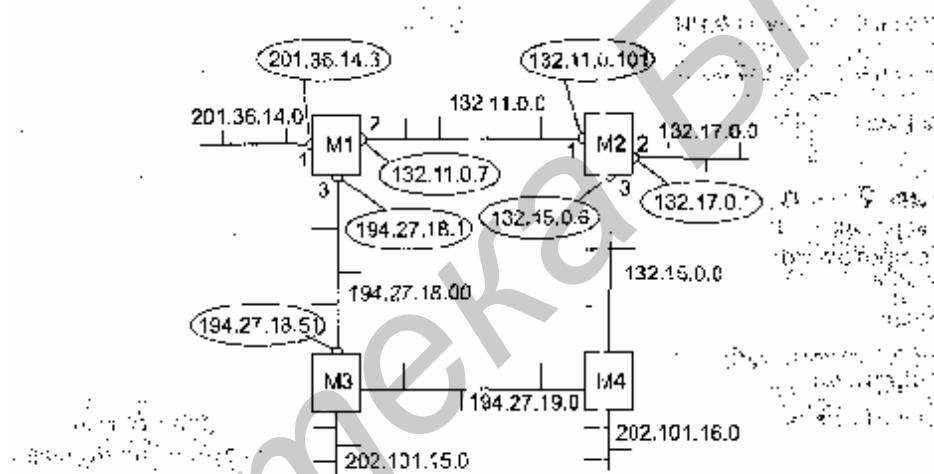


Рис. 14.3. Сеть, объединенная RIP-маршрутизаторами

Этап 1 — создание минимальных таблиц

В показанной на рисунке сети имеется восемь IP-сетей, связанных четырьмя маршрутизаторами с идентификаторами: M1, M2, M3 и M4. Маршрутизаторы, работающие по протоколу RIP, могут иметь идентификаторы, однако для работы протокола они не являются необходимыми. В RIP-сообщениях эти идентификаторы не передаются.

В исходном состоянии в каждом маршрутизаторе программным обеспечением стека TCP/IP автоматически создается минимальная таблица маршрутизации, в которой учитываются только непосредственно подсоединеные сети. На рисунке адреса портов маршрутизаторов в отличие от адресов сетей помещены в овалы.

Примерный вид минимальной таблицы маршрутизации маршрутизатора M1 позволяет оценить табл. 14.1.

Таблица 14.1. Минимальная таблица маршрутизации маршрутизатора M1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1

Минимальные таблицы маршрутизации в других маршрутизаторах будут выглядеть соответственно, например, таблица маршрутизатора M2 будет состоять из трех записей (табл. 14.2).

Таблица 14.2. Минимальная таблица маршрутизации маршрутизатора M2

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
132.11.0.0	132.11.0.101	1	1
132.17.0.0	132.17.0.1	2	1
132.15.0.0	132.15.0.6	3	1

Этап 2 — рассылка минимальных таблиц соседям

После инициализации каждого маршрутизатора он начинает посыпать своим соседям сообщения протокола RIP, в которых содержится его минимальная таблица.

RIP-сообщения передаются в пакетах протокола UDP и включают два параметра для каждой сети: ее IP-адрес и расстояние до нее от передающего сообщение маршрутизатора.

Соседями являются те маршрутизаторы, которым данный маршрутизатор непосредственно может передать IP-пакет по какой-либо своей сети, не пользуясь услугами промежуточных маршрутизаторов. Например, для маршрутизатора M1 соседями являются маршрутизаторы M2 и M3, а для маршрутизатора M4 — маршрутизаторы M2 и M3.

Таким образом, маршрутизатор M1 передает маршрутизаторам M2 и M3 сообщение со следующей информацией:

- сеть 201.36.14.0, расстояние 1;
- сеть 132.11.0.0, расстояние 1;
- сеть 194.27.18.0, расстояние 1.

Этап 3 — получение RIP-сообщений от соседей и обработка полученной информации

После получения аналогичных сообщений от маршрутизаторов M2 и M3 маршрутизатор M1 наращивает каждое полученное поле метрики на единицу и запоминает, через какой порт и от какого маршрутизатора получена новая информация (адрес этого маршрутизатора станет адресом следующего маршрутизатора, когда эта запись будет внесена в таблицу маршрутизации). Затем мар-

шрутизатор начинает сравнивать новую информацию с той, которая хранится в его таблице маршрутизации (табл. 14.3).

Таблица 14.3. Таблица маршрутизации маршрутизатора M1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
194.27.19.0	194.27.18.51	3	2
202.101.15.0	194.27.18.51	3	2
132.11.0.0	132.11.0.101	2	2
194.27.18.0	194.27.18.51	3	2

Записи с четвертой по девятую получены от соседних маршрутизаторов, и они претендуют на помещение в таблицу. Однако только записи с четвертой по седьмую попадают в таблицу, а записи восьмая и девятая — нет. Это происходит потому, что они содержат данные об уже имеющихся в таблице M1 сетях, а расстояние до них менее оптимально, чем в существующих записях.

Протокол RIP замещает запись о какой-либо сети только в том случае, если новая информация имеет лучшую метрику (расстояние в хопах меньше), чем имеющаяся. В результате в таблице маршрутизации о каждой сети остается только одна запись; если же имеется несколько равнозначных в отношении расстояния путей к одной и той же сети, то все равно в таблице остается одна запись, которая пришла в маршрутизатор первой. Для этого правила существует исключение — если информация с худшей метрикой о какой-либо сети пришла от того же маршрутизатора, на основании сообщения которого была создана данная запись, то информация с худшей метрикой замещает информацию с лучшей.

Аналогичные операции с новой информацией выполняют и остальные маршрутизаторы сети.

Этап 4 — рассылка новой, уже не минимальной, таблицы соседям

Каждый маршрутизатор отсылает новое RIP-сообщение всем своим соседям. В этом сообщении он помещает данные обо всех известных ему сетях — как непосредственно подключенных, так и удаленных, о которых маршрутизатор узнал из RIP-сообщений.

Этап 5 — получение RIP-сообщений от соседей и обработка полученной информации

Этап 5 повторяет этап 3 — маршрутизаторы принимают RIP-сообщения, обрабатывают содержащуюся в них информацию и на ее основании корректируют свои таблицы маршрутизации.

Посмотрим, как это делает маршрутизатор M1 (табл. 14.4).

Таблица 14.4. Таблица маршрутизации маршрутизатора M1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
132.15.0.0	194.27.18.54	3	3
194.27.19.0	194.27.18.51	3	2
194.27.19.0	132.11.0.101	2	3
202.101.15.0	194.27.18.51	3	2
202.101.16.0	132.11.0.101	2	3
202.101.16.0	194.27.18.51	3	3

На этом этапе маршрутизатор M1 получил от маршрутизатора M3 информацию о сети 132.15.0.0, которую тот, в свою очередь, на предыдущем цикле работы получил от маршрутизатора M4. Маршрутизатор уже знает о сети 132.15.0.0, причем старая информация имеет лучшую метрику, чем новая, поэтому новая информация об этой сети отбрасывается.

О сети 202.101.16.0 маршрутизатор M1 узнает на этом этапе впервые, причем данные о ней приходят от двух соседей — от M3 и M4. Поскольку метрики в этих сообщениях указаны одинаковые, то в таблицу попадают данные, которые пришли первыми. В нашем примере считается, что маршрутизатор M2 опередил маршрутизатор M3 и первым переслал свое RIP-сообщение маршрутизатору M1.

Если маршрутизаторы периодически повторяют этапы рассылки и обработки RIP-сообщений, то за конечное время в сети установится корректный режим маршрутизации. Под корректным режимом маршрутизации здесь понимается такое состояние таблиц маршрутизации, когда все сети достижимы из любой сети по некоторому рациональному маршруту. Пакеты будут доходить до адресатов и, не «зацикливаться» в петлях, подобных той, которую образуют маршрутизаторы M1-M2-M3-M4 (см. рис. 14.3).

Очевидно, если в сети все маршрутизаторы, их интерфейсы и соединяющие их каналы связи постоянно работоспособны, то объявления по протоколу RIP можно делать достаточно редко, например один раз в день. Однако в сетях по-

стоянно происходят изменения — изменяется как работоспособность маршрутизаторов и каналов, так и сами маршрутизаторы и каналы могут добавляться в существующую сеть или же выводиться из ее состава.

Для адаптации к изменениям в сети протокол RIP использует ряд механизмов.

Адаптация RIP-маршрутизаторов к изменениям состояния сети

К новым маршрутам RIP-маршрутизаторы приспосабливаются просто — они передают новую информацию в очередном сообщении своим соседям, и постепенно эта информация становится известна всем маршрутизаторам сети. А вот к отрицательным изменениям, связанным с потерей какого-либо маршрута, RIP-маршрутизаторы адаптируются сложнее. Это связано с тем, что в формате сообщений протокола RIP нет поля, которое бы указывало на то, что путь к данной сети больше не существует.

Вместо этого используются два механизма уведомления о том, что некоторый маршрут более недействителен:

- истечение времени жизни маршрута;
- указание специального расстояния (бесконечности) до сети, ставшей недоступной.

Для отработки первого механизма каждая запись таблицы маршрутизации (как и записи таблицы продвижения моста/коммутатора), полученная по протоколу RIP, имеет время жизни (TTL). При поступлении очередного RIP-сообщения, которое подтверждает справедливость данной записи, таймер TTL устанавливается в исходное состояние, а затем из него каждую секунду вычитается единица. Если за время тайм-аута не придет новое маршрутное сообщение об этом маршруте, то он помечается как недействительный.

Время тайм-аута связано с периодом рассылки векторов по сети. В RIP IP период рассылки выбран равным 30 с, а в качестве тайм-аута выбрано шестикратное значение периода рассылки, то есть 180 с. Выбор достаточно малого времени периода рассылки объясняется несколькими причинами, которые будут понятны из дальнейшего изложения. Шестикратный запас времени нужен для уверенности в том, что сеть действительно стала недоступной, а не просто произошли потери RIP-сообщений (а это возможно, так как RIP использует транспортный протокол UDP, который не обеспечивает надежной доставки сообщений).

Если какой-либо маршрутизатор отказывает и перестает слать своим соседям сообщения о достижимых через него сетях, то через 180 с все записи, которые породил этот маршрутизатор, у его ближайших соседей станут недействительными. После этого процесс повторится уже для соседей ближайших соседей — они вычеркнут подобные записи уже через 360 с, так как первые 180 с ближайшие соседи еще передавали сообщения об этих записях.

Как видно из объяснения, сведения о недоступных через отказавший маршрутизатор сетях распространяются по сети не очень быстро, время распространения кратно времени жизни записи, а коэффициент кратности равен количеству

хопов между самыми дальними маршрутизаторами сети. В этом заключается одна из причин выбора в качестве периода рассылки небольшой величины в 30 с.

Если отказывает не маршрутизатор, а интерфейс (или сеть), связывающий его с каким-либо соседом, то ситуация сводится к тому что описанной — снова начинает работать механизм тайм-аутов и ставшие недействительными маршруты постепенно вычеркиваются из таблиц всех маршрутизаторов сети.

Тайм-аут работает в тех случаях, когда маршрутизатор не может послать соседям сообщение об отказавшем маршруте, так как либо он сам неработоспособен, либо неработоспособна линия связи, по которой можно было бы передать сообщение.

Когда же сообщение послать можно, RIP-маршрутизаторы не используют специальный признак в сообщении, а указывают бесконечное расстояние до сети, причем в протоколе RIP оно выбрано равным 16 хопам (при другой метрике необходимо указать маршрутизатору ее значение, считающееся бесконечностью). Получив сообщение, в котором некоторая есть сопровождается расстоянием 16 (или 15, что приводит к тому же результату, так как маршрутизатор наращивает полученное значение на 1), маршрутизатор должен проверить, исходит ли эта «плохая» информация о сети от того же маршрутизатора, сообщение которого послужило в свое время основанием для записи о данной сети в таблице маршрутизации. Если это тот же маршрутизатор, то информация считается достоверной и маршрут помечается как недоступный.

Такое небольшое значение «бесконечного» расстояния вызвано тем, что в некоторых случаях отказы связей в сети вызывают длительные периоды некорректной работы RIP-маршрутизаторов, выражаящейся в «зацикливании» пакетов в петлях сети. И чем меньше расстояние, используемое в качестве «бесконечного», тем такие периоды становятся короче.

Рассмотрим случай «зацикливания» пакетов на примере сети, показанной ранее на рис. 14.3.

Пусть маршрутизатор M1 обнаружил, что его связь с непосредственно подключенной сетью 201.36.14.0 потеряна (например, по причине отказа интерфейса 201.36.14.3). M1 отметил в своей таблице маршрутизации, что сеть 201.36.14.0 недоступна. В худшем случае он обнаружил это сразу же после отправки очередных RIP-сообщений, так что до начала нового цикла его объявлений, в котором он должен сообщить соседям, что расстояние до сети 201.36.14.0 стало равным 16, остается почти 30 с.

Каждый маршрутизатор работает на основании своего внутреннего таймера, не синхронизируя работу по рассылке объявлений с другими маршрутизаторами. Поэтому весьма вероятно, маршрутизатор M2 опередил маршрутизатор M1 и передал ему свое сообщение раньше, чем M1 успел передать новость о недостижимости сети 201.36.14.0. А в этом сообщении имеются данные, порожденные следующей записью в таблице маршрутизации M2 (табл. 14.5).

Таблица 14.5. Таблица маршрутизации маршрутизатора M2

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.7	1	2

Эта запись была получена от маршрутизатора M1 и корректна до отказа интерфейса 201.36.14.3, а теперь она устарела, но маршрутизатор M2 об этом не узнал.

Теперь маршрутизатор M1 получил новую информацию о сети 201.36.14.0 — эта сеть достижима через маршрутизатор M2 с метрикой 2. Раньше M1 также получал эту информацию от M2, но игнорировал ее, так как его собственная метрика для 201.36.14.0 была лучше. Теперь M1 должен принять данные о сети 201.36.14.0, полученные от M2, и заменить запись в таблице маршрутизации о недостижимости этой сети (табл. 14.6).

Таблица 14.6. Таблица маршрутизации маршрутизатора M1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.10Г	2	3

В результате в сети образовалась маршрутная петля: пакеты, направляемые узлам сети 201.36.14.0, будут передаваться маршрутизатором M2 маршрутизатору M1, а маршрутизатор M1 будет возвращать их маршрутизатору M2. IP-пакеты будут циркулировать по этой петле до тех пор, пока не истечет время жизни каждого пакета.

Маршрутная петля может существовать в сети достаточно долго. Рассмотрим периоды времени, кратные времени жизни записей в таблицах маршрутизаторов.

- Время 0-180 с. После отказа интерфейса в маршрутизаторах M1 и M2 будут сохраняться некорректные записи, приведенные выше. Маршрутизатор M2 по-прежнему снабжает маршрутизатор M1 своей записью о сети 201.36.14.0 с метрикой 2, так как ее время жизни не истекло. Пакеты «зацикливаются».
- Время 180-360 с. В начале этого периода у маршрутизатора M2 истекает время жизни записи о сети 201.36.14.0 с метрикой 2, так как маршрутизатор M1 в предыдущий период посыпал ему сообщения о сети 201.36.14.0 с худшей метрикой, чем у M2, и они не могли подтверждать эту запись. Теперь маршрутизатор M2 принимает от маршрутизатора M1 запись о сети 201.36.14.0 с метрикой 3 и трансформирует ее в запись с метрикой 4. Маршрутизатор M1 не получает новых сообщений от маршрутизатора M2 о сети 201.36.14.0 с метрикой 2, поэтому время жизни его записи начинает уменьшаться. Пакеты продолжают «зацикливаться».
- Время 360-540 с. Теперь у маршрутизатора M1 истекает время жизни записи о сети 201.36.14.0 с метрикой 3. Маршрутизаторы M1 и M2 опять меняются ролями — M2 снабжает M1 устаревшей информацией о пути к сети 201.36.14.0, уже с метрикой 4, которую M1 преобразует в метрику 5. Пакеты продолжают «зацикливаться».

Если бы в протоколе RIP не было выбрано расстояние 16 в качестве недости-

жимого, то описанный процесс длился бы до бесконечности (вернее, пока не была бы исчерпана разрядная сетка поля расстояния и при очередном наращивании расстояния зафиксировано переполнение).

В результате маршрутизатор M2 на очередном этапе описанного процесса получает от маршрутизатора M1 метрику 15, которая после наращивания, превращаясь в метрику 16, фиксирует недостижимость сети. Период нестабильной работы сети длился 36 минут!

Ограничение в 15 хопов сужает область применения протокола RTP, ограничивая ее сетями, в которых число промежуточных маршрутизаторов не превышает 15. Для более масштабных сетей нужно применять другие протоколы маршрутизации, например OSPF, или разбивать сеть на автономные области.

Приведенный пример хорошо иллюстрирует главную причину нестабильной работы маршрутизаторов, работающих по протоколу RIP. Эта причина коренится в самом принципе работы дистанционно-векторных протоколов — они используют информацию, полученную из «вторых рук». Действительно, маршрутизатор M2 передал маршрутизатору M1 информацию о достижимости сети 201.36.14.0, за достоверность которой он сам не отвечает. Искоренить эту причину полностью нельзя, ведь сам способ построения таблиц маршрутизации связан с передачей чужой информации без указания источника ее происхождения.

Не следует думать, что при любых отказах интерфейсов и маршрутизаторов в сетях возникают маршрутные петли. Если бы маршрутизатор M1 успел передать сообщение о недостижимости сети 201.36.14.0 раньше ложной информации маршрутизатора M2, то маршрутная петля не образовалась бы. Так что маршрутные петли даже без дополнительных методов борьбы с ними, описанными в следующем разделе, возникают в среднем не более чем в половине потенциально возможных случаев.

Методы борьбы с ложными маршрутами в протоколе RIP

Хотя протокол RIP не в состоянии полностью исключить переходные состояния в сети, когда некоторые маршрутизаторы пользуются устаревшей информацией об уже несуществующих маршрутах, имеется несколько методов, которые во многих случаях решают подобные проблемы.

Описанная в предыдущем разделе ситуация с петлей, образующейся между соседними маршрутизаторами, надежно решается методом, получившим название расщепления горизонта (split horizon). Метод заключается в том, что маршрутная информация о некоторой сети, хранящаяся в таблице маршрутизации, никогда не передается тому маршрутизатору, от которого она получена (это следующий маршрутизатор в данном маршруте). Если маршрутизатор M2 в рассмотренном выше примере поддерживает технику расщепления горизонта, то он не передаст маршрутизатору M1 устаревшую информацию о сети 201.36.14.0, так как получил ее именно от маршрутизатора M1.

Практически все сегодняшние маршрутизаторы, работающие по протоколу RIP, используют технику расщепления горизонта.

Однако расщепление горизонта не помогает в тех случаях, когда петли образуются не двумя, а несколькими маршрутизаторами. Рассмотрим более детально ситуацию, которая возникнет в сети, приведенной на рис. 14.3, в случае потери связи маршрутизатора 2 с сетью А. Пусть все маршрутизаторы этой сети поддерживают технику расщепления горизонта. Маршрутизаторы М2 и М3 не будут возвращать маршрутизатору в этой ситуации данные о сети 201.36.14.0 с метрикой 2, так как они получили эту информацию от маршрутизатора М1. Однако они будут передавать маршрутизатору информацию о достижимости сети 201.36.14.0 с метрикой 4 через себя, так как получили эту информацию по сложному маршруту, а не от маршрутизатора М1 непосредственно, например, маршрутизатор М2 получил эту информацию по цепочке М4-М3-М1. Поэтому маршрутизатор М1 снова может быть обманут, пока каждый из маршрутизаторов в цепочке М3-М4-М2 не вычеркнет запись о достижимости сети 1 (а это произойдет через 3x180 с).

Для предотвращения зацикливания пакетов по составным петлям при отказах связей применяются два других приема, называемые триггерными обновлениями (triggered updates) и замораживанием изменений (hold down).

Способ триггерных обновлений состоит в том, что маршрутизатор, получив данные об изменении метрики до какой-либо сети, не ждет истечения периода передачи таблицы маршрутизации, а передает данные об изменившемся маршруте немедленно. Этот прием может во многих случаях предотвратить передачу устаревших сведений об отказавшем маршруте, но он перегружает сеть служебными сообщениями, поэтому триггерные объявления также делаются с некоторой задержкой. Из-за этой задержки снова возможна ситуация, когда регулярное обновление в каком-либо маршрутизаторе чуть опередит по времени приход триггерного обновления от предыдущего в цепочке маршрутизатора и данный маршрутизатор успеет передать по сети устаревшую информацию о несуществующем маршруте.

Второй прием позволяет исключить подобные ситуации. Он связан с введением тайм-аута на принятие новых данных о сети, которая только что стала недоступной. Этот тайм-аут предотвращает принятие устаревших сведений о некотором маршруте от маршрутизаторов, находящихся на некотором расстоянии от отказавшей связи и передающих устаревшие сведения о ее работоспособности. Предполагается, что в течение тайм-аута «замораживания изменений» эти маршрутизаторы вычеркнут данный маршрут из своих таблиц, так как не получат о нем новых записей и не будут распространять устаревшие сведения по сети.

Выводы

- Протокол RIP является наиболее распространенным протоколом маршрутизации сетей TCP/IP. Несмотря на его простоту, определенную использованием дистанционно-векторного алгоритма, RIP успешно работает в небольших сетях с количеством промежуточных маршрутизаторов не более 15.
- Для IP имеются две версии протокола RIP: первая и вторая. Протокол RIPv1

не поддерживает масок. Протокол RIPv2 передает информацию о масках сетей, поэтому он в большей степени соответствует требованиям сегодняшнего дня.

□ RIP-маршрутизаторы при выборе маршрута обычно используют самую простую метрику — количество промежуточных маршрутизаторов между сетями, то есть хопов.

□ В сетях, использующих RIP и имеющих петлевидные маршруты, могут наблюдаться достаточно длительные периоды нестабильной работы, когда пакеты «зацикливаются» в маршрутных петлях и не доходят до адресатов. Для борьбы с этими явлениями в RIP-маршрутизаторах предусмотрено несколько приемов (Split Horizon, Hold Down, Triggered Updates), которые сокращают в некоторых случаях периоды нестабильности.

□ Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях.

Протокол состояния связей OSPF

Протокол OSPF (Open Shortest Path First, открытый протокол «кратчайший путь первым») является достаточно современной реализацией алгоритма состояния связей (он принят в 1991 году) и обладает многими особенностями, ориентированными на применение в больших гетерогенных сетях.

Два этапа построения таблицы маршрутизации

В OSPF процесс построения таблицы маршрутизации разбивается на два крупных этапа. На первом этапе каждый маршрутизатор строит граф связей сети, в котором вершинами графа являются маршрутизаторы и IP-сети, а ребрами — интерфейсы маршрутизаторов. Все маршрутизаторы для этого обмениваются со своими соседями той информацией о графе сети, которой они располагают к данному моменту времени. Этот процесс похож на процесс распространения векторов расстояний до сетей в протоколе RIP, однако сама информация качественно другая — это информация о топологии сети. Подобные сообщения называются *router links advertisement* — объявление о связях маршрутизатора. Кроме того, при передаче топологической информации маршрутизаторы ее не модифицируют, как это делают RIP-маршрутизаторы, а передают в неизменном виде. В результате распространения топологической информации все маршрутизаторы сети располагают идентичными сведениями о графике сети, которые хранятся в топологической базе данных каждого маршрутизатора.

Второй этап состоит в нахождении оптимальных маршрутов с помощью полученного графа. Каждый маршрутизатор считает себя центром сети и ищет оптимальный маршрут до каждой известной ему сети. В каждом найденном таким образом маршруте запоминается только один шаг — до следующего маршрутизатора, в соответствии с принципом одношаговой маршрутизации. Данные об этом шаге и попадают в таблицу маршрутизации. Задача нахождения оптимального пути на графике является достаточно сложной и трудоемкой. В прото-

коле OSPF для ее решения используется итеративный алгоритм Дийкстры. Если несколько маршрутов имеют одинаковую метрику до сети назначения, то в таблице маршрутизации запоминаются первые шаги всех этих маршрутов.

Маршрутные объявления HELLO

После первоначального построения таблицы маршрутизации необходимо отслеживать изменения состояния сети и вносить корректизы в таблицу маршрутизации. Для контроля состояния связей и соседних маршрутизаторов OSPF-маршрутизаторы не используют обмен полной таблицей маршрутизации, как это не очень рационально делают RIP-маршрутизаторы. Вместо этого они передают специальные короткие сообщения HELLO. Если состояние сети не меняется, то OSPF-маршрутизаторы корректируют свою таблицу маршрутизации не занимаются и не посылают соседям объявления о связях. Если же состояние связи изменилось, то ближайшим соседям посылается новое объявление, касающееся только данной связи, что, конечно, экономит пропускную способность сети. Получив новое объявление об изменении состояния связи, маршрутизатор перестраивает граф сети, заново ищет оптимальные маршруты (не обязательно все, а только те, на которых отразилось данное изменение) и корректирует свою таблицу маршрутизации. Одновременно маршрутизатор ретранслирует объявление каждому из своих ближайших соседей (кроме того, от которого он получил это объявление).

При появлении новой связи или нового соседа маршрутизатор узнает об этом из новых сообщений HELLO. В сообщениях HELLO указывается достаточно детальная информация о том маршрутизаторе, который послал это сообщение, а также о его ближайших соседях, чтобы данный маршрутизатор можно было однозначно идентифицировать. Сообщения HELLO отправляются через каждые 10 с, чтобы повысить скорость адаптации маршрутизаторов к изменениям, происходящим в сети. Небольшой объем этих сообщений делает возможным такое частое тестирование состояния соседей и связей с ними.

Так как маршрутизаторы являются одними из вершин графа, то они обязательно

должны иметь идентификаторы.

Метрики и объявления

Протокол OSPF обычно использует метрику, учитывающую пропускную способность сетей. Кроме того, возможно использование двух других метрик, учитывающих требования к качеству обслуживания в IP-пакете, — задержки передачи пакетов и надежности передачи пакетов сетью. Для каждой из метрик протокол OSPF строит отдельную таблицу маршрутизации. Выбор нужной таблицы происходит в зависимости от требований к качеству обслуживания пришедшего пакета (рис. 14.4),

Маршрутизаторы соединены как с локальными сетями, так и непосредственно

между собой глобальными каналами типа «точка-точка» (рис. 14.4).

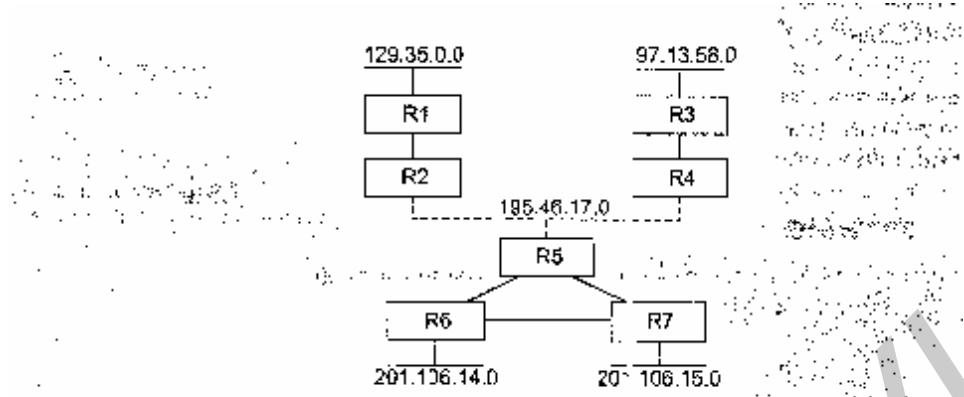


Рис. 14.4. Исходная сеть

Данной сети соответствует граф, приведенный на рис. 14.5.

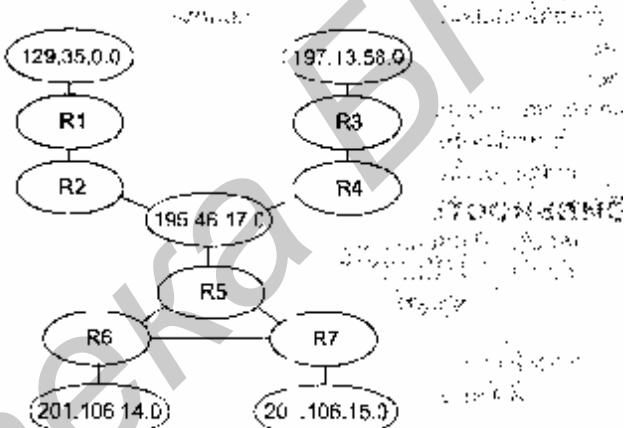


Рис. 14.5. Граф сети, построенный протоколом OSPF

Протокол OSPF в своих объявлениях распространяет информацию о связях двух типов: маршрутизатор—маршрутизатор и маршрутизатор—сеть. Примером связи первого типа служит связь «R3—R4», а второго — связь «R4—195.46.17.0». Если каналам «точка-точка» дать IP-адреса, то они станут дополнительными вершинами графа, как и локальные сети. Вместе с IP-адресом сети передается также информация о маске сети.

После инициализации OSPF-маршрутизаторы знают только о связях с непосредственно подключенными сетями, как и RIP-маршрутизаторы. Они начинают распространять эту информацию своим соседям. Одновременно они посыпают сообщения HELLO по всем своим интерфейсам, так что почти сразу же каждый маршрутизатор узнает идентификаторы своих ближайших соседей, что пополняет его топологическую базу новой информацией, которую он узнал непосредственно. Далее топологическая информация начинает распространяться по сети от соседа к соседу и через некоторое время достигает самых удаленных маршрутизаторов.

Каждая связь характеризуется метрикой. Протокол OSPF поддерживает стандартные для многих протоколов (например, для протокола Spanning Tree) зна-

чения расстояний для метрики, отражающей производительность сетей: Ethernet — 10 единиц, Fast Ethernet — 1 единица, канал T1 — 65 единиц, канал 56 кбит/с — 1785 единиц и т. д.

При выборе оптимального пути на графе с каждым ребром графа связана метрика, которая добавляется к пути, если данное ребро в него входит. Пусть на приведенном примере маршрутизатор R5 связан с R6 и R7 каналами T1, а R6 и R7 связаны между собой каналом 56 кбит/с. Тогда R7 определит оптимальный маршрут до сети 201.106.14.0 как составной, проходящий сначала через маршрутизатор R5, а затем через R6, поскольку у этого маршрута метрика равна $65 + 65 = 130$ единиц. Непосредственный маршрут через R6 не будет оптимальным, так как его метрика равна 1785. При использовании хопов был бы выбран маршрут через R6, который не является оптимальным.

Протокол OSPF разрешает хранить в таблице маршрутизации несколько маршрутов к одной сети, если они обладают равными метриками. Если такие записи образуются в таблице маршрутизации, то маршрутизатор реализует режим баланса загрузки маршрутов (load balancing), отправляя пакеты попеременно по каждому из маршрутов.

Стабильность OSPF

У каждой записи в топологической базе данных имеется срок жизни, как и у маршрутных записей протокола RIP. С каждой записью о связях связан таймер, который используется для контроля времени жизни записи. Если какая-либо запись топологической базы маршрутизатора, полученная от другого маршрутизатора, устаревает, то маршрутизатор может запросить ее новую копию с помощью специального сообщения Link-State Request протокола OSPF, на которое должен поступить ответ Link-State Update от маршрутизатора, непосредственно тестирующего запрошенную связь.

Для инициализации маршрутизаторов и более надежной синхронизации топологических баз маршрутизаторы периодически обмениваются всеми записями базы, но этот период существенно больше, чем у RIP-маршрутизаторов.

Так как информация о некоторой связи изначально генерируется только тем маршрутизатором, который выяснил фактическое состояние этой связи путем тестирования с помощью сообщений HELLO, а остальные маршрутизаторы только ретранслируют эту информацию без преобразования, то недостоверная информация о достижимости сетей, которая может появляться в RIP-маршрутизаторах, в OSPF-маршрутизаторах появиться не может, а устаревшая информация быстро заменяется новой, так как при изменении состояния связи новое сообщение генерируется сразу же.

В OSPF-сетях могут возникать периоды нестабильной работы. Например, при отказе связи, когда информация об этом не дошла до какого-либо маршрутизатора, он продолжает отправлять пакеты сети назначения, считая эту связь работоспособной. Однако эти периоды продолжаются недолго, причем пакеты не «зацикливаются» в маршрутных петлях, а просто отбрасываются при невозможности их передать через неработоспособную связь.

К недостаткам протокола OSPF следует отнести его вычислительную сложность, которая быстро растет с увеличением размерности сети, то есть количества сетей, маршрутизаторов и связей между ними. Для преодоления этого недостатка в протоколе OSPF вводится понятие области сети (area), которую не нужно путать с автономной системой Интернета. Маршрутизаторы, принадлежащие некоторой области, строят граф связей только для этой области, что сокращает размерность сети. Между областями информация о связях не передается, а пограничные для областей маршрутизаторы обмениваются только информацией об адресах сетей, имеющихся в каждой из областей, и расстоянием от пограничного маршрутизатора до каждой сети. При передаче пакетов между областями выбирается один из пограничных маршрутизаторов области, а именно тот, у которого расстояние до нужной сети меньше. Этот стиль напоминает стиль работы протокола RIP, но нестабильность здесь устраняется тем, что петлевидные связи между областями запрещены. При передаче адресов в другую область OSPF-маршрутизаторы агрегируют несколько адресов в один, если обнаруживают у них общий префикс.

OSPF-маршрутизаторы могут принимать адресную информацию от других протоколов маршрутизации, например от протокола RIP, что полезно для работы в гетерогенных сетях. Такая адресная информация обрабатывается так же, как и внешняя информация между разными областями.

Выводы

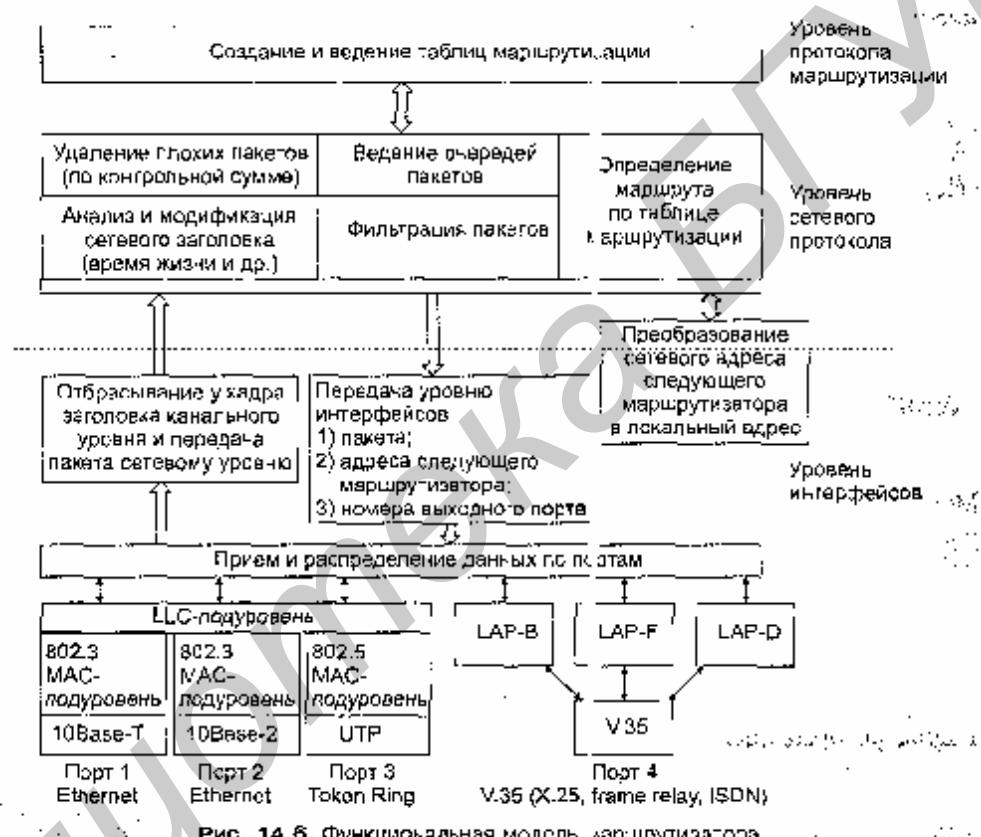
- Протокол OSPF был разработан для эффективной маршрутизации IP-пакетов в больших сетях со сложной топологией, включающей петли. Он основан на алгоритме состояния связей, который обладает высокой устойчивостью к изменениям топологии сети.
- Периоды нестабильной работы в OSPF-сетях продолжаются недолго, причем пакеты не «зацикливаются» в маршрутных петлях, а просто отбрасываются при невозможности их передать через неработоспособную связь.
- При выборе маршрута OSPF-маршрутизаторы используют метрику, учитывающую пропускную способность составных сетей.
- Протокол OSPF является первым протоколом маршрутизации для IP-сетей, который учитывает биты качества обслуживания (пропускная способность, задержка и надежность) в заголовке IP-пакета. Для каждого типа качества обслуживания строится отдельная таблица маршрутизации.
- Протокол OSPF разрешает хранить в таблице маршрутизации несколько маршрутов к одной сети, если они обладают равными метриками, что дает возможность маршрутизатору работать в режиме баланса загрузки маршрутов.
- Протокол OSPF обладает высокой вычислительной сложностью, поэтому чаще всего работает на мощных аппаратных маршрутизаторах.

Маршрутизаторы

Основные функции маршрутизатора

Основная функция маршрутизатора — чтение заголовков пакетов сетевых протоколов, принимаемых и буферизуемых по каждому порту (например, IPX, IP, AppleTalk или DECnet), и принятие решения о дальнейшем маршруте следования пакета по его сетевому адресу, включающему, как правило, номер сети и номер узла.

Функции маршрутизатора могут быть разбиты на три группы в соответствии с уровнями модели OSI (рис. 14.6).



Уровень интерфейсов

На нижнем уровне маршрутизатор, как и любое устройство, подключенное к сети, обеспечивает физический интерфейс со средой передачи, включая согласование уровней электрических сигналов, линейное и логическое кодирование, оснащение определенным типом разъема. В разных моделях маршрутизаторов часто предусматриваются различные наборы физических интерфейсов, представляющих собой комбинацию портов для подсоединения локальных и глобальных сетей. С каждым интерфейсом для подключения локальной сети неразрывно связан определенный протокол канального уровня, например Ethernet, Token Ring, FDDI. Интерфейсы для присоединения к глобальным сетям чаще

всего определяют только некоторый стандарт физического уровня, над которым в маршрутизаторе могут работать различные протоколы канального уровня. Например, глобальный порт может поддерживать интерфейс V.35, над которым могут работать протоколы канального уровня: LAP-B (в сетях X.25), LAP-F (в сетях frame relay), LAP-D (в сетях ISDN). Разница между интерфейсами локальных и глобальных сетей объясняется тем, что технологии локальных сетей работают по собственным стандартам физического уровня, которые не могут, как правило, использоваться в других технологиях, поэтому интерфейс для локальной сети представляет собой сочетание физического и канального уровней и носит название по имени соответствующей технологии, например интерфейс Ethernet.

Интерфейсы маршрутизатора выполняют полный набор функций физического и канального уровней по передаче кадра, включая получение доступа к среде (если это необходимо), формирование битовых сигналов, прием кадра, подсчет его контрольной суммы и передачу поля данных кадра верхнему уровню, в случае если контрольная сумма имеет корректное значение.

ПРИМЕЧАНИЕ

Как и любой конечный узел, каждый порт маршрутизатора имеет собственный аппаратный адрес (в локальных сетях — MAC-адрес), по которому другие узлы направляют ему кадры, требующие маршрутизации.

Перечень физических интерфейсов, которые поддерживает та или иная модель маршрутизатора, является его важнейшей потребительской характеристикой. Маршрутизатор должен поддерживать все протоколы канального и физического уровней, используемые в каждой из сетей, к которым он будет непосредственно присоединен. На рис. 14.6 показана функциональная модель маршрутизатора с четырьмя портами, реализующими следующие физические интерфейсы: 10Base-T и 10Base-2 для двух портов Ethernet, UTP для Token Ring и V.35, над которым могут работать протоколы LAP-B, LAP-D или LAP-F, обеспечивая подключение к сетям X.25, ISDN или frame relay.

Кадры, которые поступают на порты маршрутизатора, после обработки соответствующими протоколами физического и канального уровней освобождаются от заголовков канального уровня. Извлеченные из поля данных кадра пакеты передаются модулю сетевого протокола.

Уровень сетевого протокола

Сетевой протокол, в свою очередь, извлекает из пакета заголовок сетевого уровня и анализирует содержимое его полей. Прежде всего проверяется контрольная сумма, и если пакет пришел поврежденным, то он отбрасывается. Выполняется проверка, не превысило ли время, которое провел пакет в сети (время жизни пакета), допустимой величины. Если превысило — пакет также отбрасывается. На этом этапе вносятся корректировки в содержимое некоторых полей, например наращивается время жизни пакета, пересчитывается контрольная сумма.

На сетевом уровне выполняется одна из важнейших функций маршрутизатора — фильтрация трафика. Маршрутизатор, обладая более высоким интеллектом, нежели мост или коммутатор, позволяет задавать и может отрабатывать значительно более сложные правила фильтрации. Пакет сетевого уровня, находящийся в поле данных кадра, для мостов и коммутаторов представляется неструктурированной двоичной последовательностью. Маршрутизаторы же, программное обеспечение которых содержит модуль сетевого протокола, способны производить разбор и анализ отдельных полей пакета. Они оснащаются развитыми средствами пользовательского интерфейса, которые позволяют администратору без особых усилий задавать сложные правила фильтрации. Они, например, могут запретить прохождение в корпоративную сеть всех пакетов, кроме пакетов, поступающих из подсетей «родного» предприятия. Фильтрация в данном случае производится по сетевым адресам, и все пакеты, адреса которых не входят в разрешенный диапазон, отбрасываются. Маршрутизаторы, как правило, также могут анализировать структуру сообщений транспортного уровня, поэтому фильтры могут не пропускать в сеть сообщения определенных прикладных служб, например службы telnet, анализируя поле типа протокола в транспортном сообщении.

В случае если интенсивность поступления пакетов выше интенсивности их обработки, пакеты могут образовать очередь. Программное обеспечение маршрутизатора может реализовывать различные дисциплины обслуживания очередей пакетов: в порядке поступления по принципу «первый пришел — первым обслужен» (First Input First Output, FIFO), дисциплину случайного раннего обнаружения (Random Early Detection, RED), когда обслуживание идет по правилу FIFO, но при достижении длиной очереди некоторого порогового значения вновь поступающие пакеты отбрасываются, а также различные варианты приоритетного обслуживания.

К сетевому уровню относится основная функция маршрутизатора — определение маршрута пакета. По номеру сети, извлеченному из заголовка пакета, модуль сетевого протокола находит в таблице маршрутизации строку, содержащую сетевой адрес следующего маршрутизатора, и номер порта, на который нужно передать данный пакет, чтобы он двигался в правильном направлении. Если в таблице отсутствуют записи о сети назначения пакета и о маршрутизаторе по умолчанию, то данный пакет отбрасывается.

Перед тем как передать сетевой адрес следующего маршрутизатора на канальный уровень, необходимо преобразовать его в локальный адрес той технологии, которая используется в сети, содержащей следующий маршрутизатор. Для этого сетевой протокол обращается к протоколу разрешения адресов. Протоколы этого типа устанавливают соответствие между сетевыми и локальными адресами либо па основании заранее составленных таблиц, либо путем рассылки широковещательных запросов. Таблица соответствия локальных адресов сетевым адресам строится отдельно для каждого сетевого интерфейса. Протоколы разрешения адресов занимают промежуточное положение между сетевым и канальным уровнями.

С сетевого уровня пакет, локальный адрес следующего маршрутизатора и но-

мер порта маршрутизатора передаются вниз, канальному уровню. На основании указанного номера порта осуществляется коммутация с одним из интерфейсов маршрутизатора, средствами которого выполняется упаковка пакета в кадр соответствующего формата. В поле адреса назначения заголовка кадра помещается локальный адрес следующего маршрутизатора. Готовый кадр отправляется в сеть.

Уровень протоколов маршрутизации

Сетевые протоколы активно используют в своей работе таблицу маршрутизации, но ни ее построением, ни поддержанием ее содержимого не занимаются. Эти функции выполняют протоколы маршрутизации. На основании этих протоколов маршрутизаторы обмениваются информацией о топологии сети, а затем анализируют полученные сведения, определяя наилучшие по тем или иным критериям маршруты. Результаты анализа и составляют содержимое таблиц маршрутизации.

Помимо перечисленных выше функций на маршрутизаторы могут быть возложены и другие обязанности, например операции, связанные с фрагментацией.

Дополнительные функциональные возможности маршрутизаторов

Наряду с функцией маршрутизации многие маршрутизаторы обладают важными дополнительными функциональными возможностями (они перечислены далее), которые значительно расширяют сферу применения этих устройств.

Поддержка одновременно нескольких протоколов маршрутизации. В протоколах маршрутизации обычно предполагается, что маршрутизатор строит свою таблицу на основе работы только этого одного протокола. Деление Интернета на автономные системы также направлено на исключение использования в одной автономной системе нескольких протоколов маршрутизации. Тем не менее иногда в большой корпоративной сети приходится поддерживать одновременно несколько таких протоколов, чаще всего так складывается исторически. При этом таблица маршрутизации может получаться противоречивой — разные протоколы маршрутизации могут выбрать разные следующие маршрутизаторы для какой-либо сети назначения. Большинство маршрутизаторов решает эту проблему за счет придания приоритетов решениям разных протоколов маршрутизации. Высший приоритет отдается статическим маршрутам (администратор всегда прав), следующий приоритет имеют маршруты, выбранные протоколами состояния связей, такими как OSPF или NLSP, а низшим приоритетом обладают маршруты дистанционно-векторных протоколов, как самых несовершенных. Приоритеты сетевых протоколов. Можно установить приоритет одного протокола сетевого уровня над другими. На выбор маршрутов эти приоритеты не оказывают никакого влияния, они влияют только на порядок, в котором много-протокольный маршрутизатор обслуживает пакеты разных сетевых протоколов.

Это свойство бывает полезно в случае недостаточной полосы пропускания кабельной системы и существования трафика, чувствительного к временным задержкам, например трафика SNA или голосового трафика, передаваемого одним из сетевых протоколов.

Поддержка политики маршрутных объявлений. В большинстве протоколов обмена маршрутной информации (RIP, OSPF, NLSP) предполагается, что маршрутизатор объявляет в своих сообщениях обо всех сетях, которые ему известны. Аналогично предполагается, что маршрутизатор при построении своей таблицы учитывает все адреса сетей, которые поступают ему от других маршрутизаторов сети. Однако существуют ситуации, когда администратор хотел бы скрыть существование некоторых сетей в определенной части своей сети от других администраторов, например, по соображениям безопасности. Или же администратор хотел бы запретить некоторые маршруты, которые могли бы существовать в сети. При статическом построении таблиц маршрутизации решение таких проблем не составляет труда. Динамические же протоколы маршрутизации не позволяют стандартным способом реализовывать подобные ограничения. Существует только один широко используемый протокол динамической маршрутизации, в котором описана возможность существования правил (policy), ограничивающих распространение некоторых адресов в объявлениях, — это протокол BGP. Необходимость поддержки таких правил в протоколе BGP понятна, так как это протокол обмена маршрутной информацией между автономными системами, где велика потребность в административном регулировании маршрутов (например, некоторый поставщик услуг Интернета может не захотеть, чтобы через него транзитом проходил трафик другого поставщика услуг). Разработчики маршрутизаторов исправляют этот недостаток стандартов протоколов, вводя в маршрутизаторы поддержку правил передачи и использования маршрутной информации, подобных тем, которые рекомендует BGP.

Защита от широковещательных штормов (broadcast storm). Одна из характерных неисправностей сетевого программного обеспечения — самопроизвольная с высокой интенсивностью генерация широковещательных пакетов. Широковещательным штормом считается ситуация, в которой процент широковещательных пакетов превышает 20 % от общего количества пакетов в сети. Обычный коммутатор или мост слепо передает такие пакеты на все свои порты, как того требует логика его работы, засоряя, таким образом, сеть. Борьба с широковещательным штормом в сети, соединенной коммутаторами, требует от администратора отключения портов, генерирующих широковещательные пакеты. Маршрутизатор не распространяет такие поврежденные пакеты, поскольку в круг его задач не входит копирование широковещательных пакетов во все объединяемые им сети. Поэтому маршрутизатор является прекрасным средством борьбы с широковещательным штормом, правда, если сеть разделена на достаточноное количество подсетей.

Поддержка немаршрутизуемых протоколов, таких как NetBIOS, NetBEUI или DEC LAT, которые не оперируют таким понятием, как сеть. Маршрутизаторы могут обрабатывать пакеты подобных протоколов двумя способами.

□ В первом случае они могут работать с пакетами этих протоколов как мосты,

то есть передавать их па основании изучения MAC-адресов. Маршрутизатор необходимо сконфигурировать особым способом, чтобы по отношению к некоторым немаршрутизуемым протоколам на некоторых портах он выполнял функции моста, а по отношению к маршрутизуемым протоколам — функции маршрутизатора. Такой мост/маршрутизатор иногда называют brouter, то есть bridge (мост) плюс router (маршрутизатор).

□ Другим способом передачи пакетов немаршрутизуемых протоколов является инкапсуляция этих пакетов в пакеты какого-либо сетевого протокола. Некоторые производители маршрутизаторов разработали собственные протоколы, специально предназначенные для инкапсуляции немаршрутизуемых пакетов. Кроме того, существуют стандарты для инкапсуляции некоторых протоколов в другие, в основном в IP. Примером такого стандарта является протокол DLSw, определяющий методы инкапсуляции пакетов SDLC и NetBIOS в IP-пакеты, а также протоколы PPTP и L2TP, инкапсулирующие кадры протокола PPP в IP-пакеты.

Основная вычислительная работа проводится маршрутизатором при составлении таблицы маршрутизации с маршрутами ко всем известным ему сетям. Эта работа состоит в обмене пакетами протоколов маршрутизации, такими как R1P или OSPF, и вычислении оптимального пути к каждой целевой сети по некоторому критерию. Для вычисления оптимального пути на графе, как того требуют протоколы состояния связей, необходимы значительные вычислительные мощности. После того как таблица маршрутизации составлена, функция продвижения пакетов происходит весьма просто — осуществляется просмотр таблицы и поиск совпадения полученного адреса с адресом целевой сети. Если совпадение есть, то пакет передается на соответствующий порт маршрутизатора. Некоторые маршрутизаторы поддерживают только функции продвижения пакетов по готовой таблице маршрутизации. Такие маршрутизаторы считаются усеченными, так как для их полноценной работы требуется наличие полнофункционального маршрутизатора, у которого можно взять готовую таблицу маршрутизации. Этот маршрутизатор часто называется сервером маршрутов. Отказ от самостоятельного выполнения функций построения таблицы маршрутизации резко удешевляет маршрутизатор и повышает его производительность. Примеры такого подхода дают маршрутизаторы NetBuilder компании 3Com, поддерживающие фирменную технологию Boundary Routing, и маршрутизирующие коммутаторы Catalyst 5000 компании Cisco Systems.

Классификация маршрутизаторов по областям применения

По областям применения маршрутизаторы делятся на несколько классов.

Магистральные маршрутизаторы (backbone routers) предназначены для построения центральной сети крупной корпорации или телекоммуникационного оператора.

Магистральные маршрутизаторы оперируют с агрегированными информационными потоками, переносящими данные большого количества пользовательских соединений. Центральная сеть может состоять из множества локальных сетей,

разбросанных по разным зданиям и использующих самые разнообразные сетевые технологии, типы компьютеров и операционных систем. Магистральные маршрутизаторы — это мощные устройства, способные обрабатывать несколько сотен тысяч или даже несколько миллионов пакетов в секунду, имеющие большое количество интерфейсов локальных и глобальных сетей. Поддерживаются не только среднескоростные интерфейсы глобальных сетей, такие как T1/E1, но и высокоскоростные, например ATM или SDH со скоростями 155 Мбит/с или 622 Мбит/с. Чаще всего магистральный маршрутизатор конструктивно выполнен по модульной схеме на основе шасси с множеством слотов — до 12-14. Большое внимание уделяется в магистральных моделях надежности и отказоустойчивости маршрутизатора, которая достигается за счет системы терморегуляции, избыточных источников питания, заменяемых «на ходу» (hot swap) модулей, а также симметричного мультипроцессирования. Примерами магистральных маршрутизаторов могут служить маршрутизаторы Backbone Concentrator Node (BCN) компании Nortel Networks (ранее Bay Networks), Cisco 7500, Cisco 12000.

Маршрутизаторы региональных отделений соединяют региональные отделения между собой и с центральной сетью. Сеть регионального отделения, так же как и центральная сеть, может состоять из нескольких локальных сетей. Такой маршрутизатор обычно представляет собой некоторую упрощенную версию магистрального маршрутизатора. Если он выполнен на основе шасси, то количество слотов его шасси меньше: 4-5. Возможен также конструктив с фиксированным количеством портов. Поддерживаемые интерфейсы локальных и глобальных сетей менее скоростные. Примерами маршрутизаторов региональных отделений могут служить маршрутизаторы BLN, ASN компании Nortel Networks, Cisco 3600, Cisco 2500, NetBuilder II компании 3Com. Это наиболее обширный класс выпускаемых маршрутизаторов, характеристики которых могут приближаться к характеристикам магистральных маршрутизаторов, а могут и опускаться до характеристик маршрутизаторов удаленных офисов.

Маршрутизаторы удаленных офисов соединяют, как правило, единственную локальную сеть удаленного офиса с центральной сетью или сетью регионального отделения по глобальной связи. В максимальном варианте такие маршрутизаторы могут поддерживать и два интерфейса локальных сетей. Как правило, интерфейс локальной сети — это Ethernet 10 Мбит/с, а интерфейс глобальной сети — выделенная линия со скоростью 64 кбит/с, 1,544 или 2 Мбит/с. Маршрутизатор удаленного офиса может поддерживать работу по коммутируемой телефонной линии в качестве резервной связи для выделенного канала. Существует очень большое количество типов маршрутизаторов удаленных офисов. Это объясняется как массовостью потенциальных потребителей, так и специализацией такого типа устройств, проявляющейся в поддержке одного конкретного типа глобальной связи. Например, существуют маршрутизаторы, работающие только с сетью ISDN, существуют модели только для аналоговых выделенных линий и т. п. Типичными представителями этого класса являются маршрутизаторы Nautika компании Nortel Networks, Cisco 1600, Office Connect компании 3Com, семейство Pipeline компании Ascend.

Маршрутизаторы локальных сетей (коммутаторы 3-го уровня) предназначены для разделения крупных локальных сетей на подсети. Основное требование, предъявляемое к ним, — высокая скорость маршрутизации, поскольку в такой конфигурации отсутствуют низкоскоростные порты, такие как модемные порты 33,6 кбит/с или цифровые порты 64 кбит/с. Все порты имеют скорость по крайней мере 10 Мбит/с, а многие работают на скорости 100 Мбит/с. Примерами коммутаторов 3-го уровня служат коммутаторы CoreBuilder 3500 компании 3Com, Accelar 1200 компании Nortel Networks, Waveswitch 9000 компании Plaintiffree, Turboiron Switching Router компании Foudry Networks. В зависимости от области применения маршрутизаторы обладают различными основными и дополнительными техническими характеристиками.

Основные технические характеристики маршрутизаторов

Основные технические характеристики маршрутизатора связаны с тем, как он решает свою главную задачу — маршрутизацию пакетов в составной сети. Именно эти характеристики в первую очередь определяют возможности и сферу применения того или иного маршрутизатора.

Перечень поддерживаемых сетевых протоколов. Магистральный маршрутизатор должен поддерживать большое количество сетевых протоколов и протоколов маршрутизации, чтобы обеспечивать трафик всех существующих на предприятии вычислительных систем (в том числе и устаревших, но все еще успешно эксплуатирующихся, так называемых унаследованных — legacy), а также систем, которые могут появиться на предприятии в ближайшем будущем. Если центральная сеть образует отдельную автономную систему Интернета, то потребуется поддержка и специфических протоколов маршрутизации этой сети, таких как EGP и BGP. Программное обеспечение магистральных маршрутизаторов обычно строится по модульному принципу, поэтому при возникновении потребности можно докупать и добавлять программные модули, реализующие недостающие протоколы.

Перечень поддерживаемых сетевых протоколов обычно включает протоколы IP, CONS и CLNS OSI, IPX, AppleTalk, DECnet, Banyan VINES, Xerox XNS.

Перечень протоколов маршрутизации составляют протоколы IP RIP, IPX RIP, NLSP, OSPF, IS-IS OSI, EGP, BGP, VINES RTF, AppleTalk RTMP.

Перечень поддерживаемых интерфейсов локальных и глобальных сетей. Для локальных сетей — это интерфейсы, реализующие физические и канальные протоколы сетей Ethernet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, 100VG-AnyLAN и ATM.

Для глобальных связей — это интерфейсы физического уровня для связи с аппаратурой передачи данных, а также протоколы канального и сетевого уровней, необходимые для подключения к глобальным сетям с коммутацией каналов и пакетов.

Поддерживаются интерфейсы последовательных линий (serial lines) RS-232, RS-449/422, V.35 (для передачи данных со скоростями до 2-6 Мбит/с), высокоскоростной интерфейс HSSI, обеспечивающий скорость до 52 Мбит/с, а также

интерфейсы с цифровыми каналами T1/E1, T3/E3 и интерфейсами BRI и PRI цифровой сети ISDN. Некоторые маршрутизаторы имеют аппаратуру связи с цифровыми глобальными каналами, что исключает необходимость использования внешних устройств сопряжения с этими каналами.

В набор поддерживаемых глобальных технологий обычно входят технологии X.25, frame relay, ISDN и коммутируемых аналоговых телефонных сетей, сетей ATM, а также поддержка протокола канального уровня PPP.

Общая производительность маршрутизатора. Высокая производительность маршрутизации важна для работы с высокоскоростными локальными сетями, а также для поддержки новых высокоскоростных глобальных технологий, таких как frame relay, T3/E3, SDH и ATM. Общая производительность маршрутизатора зависит от многих факторов, наиболее важными из которых являются: тип используемых процессоров, эффективность программной реализации протоколов, архитектурная организация вычислительных и интерфейсных модулей. Общая производительность маршрутизаторов колеблется от нескольких десятков тысяч пакетов в секунду до нескольких миллионов пакетов в секунду. Наиболее производительные маршрутизаторы имеют мультипроцессорную архитектуру, сочетающую симметричные и асимметричные свойства — несколько мощных центральных процессоров, работая по симметричной схеме, выполняют функции вычисления таблицы маршрутизации, а менее мощные процессоры в интерфейсных модулях занимаются передачей пакетов на подключенные к ним сети и пересылкой пакетов на основании части таблицы маршрутизации, кэшированной в локальной памяти интерфейсного модуля.

Магистральные маршрутизаторы обычно поддерживают максимальный набор протоколов и интерфейсов и обладают высокой общей производительностью в один-два миллиона пакетов в секунду. Маршрутизаторы удаленных офисов поддерживают несколько протоколов локальных сетей и низкоскоростные глобальные протоколы. Общая производительность таких маршрутизаторов обычно составляет от 5 до 20-30 тысяч пакетов в секунду.

Маршрутизаторы региональных отделений занимают промежуточное положение, поэтому их иногда не выделяют в отдельный класс устройств.

Наиболее высокой производительностью обладают коммутаторы 3-го уровня, особенности которых рассмотрены ниже.

Корпоративные модульные концентраторы

Большинство крупных фирм-производителей сетевого оборудования предлагает модульные концентраторы в качестве «коммутационного центра» корпоративной сети. Такие концентраторы отражают тенденцию перехода от полностью распределенных локальных сетей 70-х годов на коаксиальном кабеле к централизованным коммуникационным решениям, активно воздействующим на передачу пакетов между сегментами и сетями. Модульные корпоративные концентраторы представляют собой многофункциональные устройства, которые могут включать несколько десятков модулей различного назначения: повторителей разных технологий, коммутаторов, удаленных мостов, маршрутизаторов

и т. п., которые объединяются в одном устройстве с модулями-агентам и протокола SNMP1 и, следовательно, позволяют централизовано объединять, управлять и обслуживать большое количество устройств и сегментов, что очень удобно в сетях большого размера.

Модульный концентратор масштаба предприятия обычно обладает внутренней шиной или набором шин очень высокой производительности — до нескольких десятков гигабит в секунду, что позволяет реализовать одновременные соединения между модулями с высокой скоростью, гораздо большей, чем скорость внешних интерфейсов модулей. Основная идея разработчиков таких устройств заключается в создании программно настраиваемой конфигурации связей в сети, причем сами связи между устройствами и сегментами могут также поддерживаться различными методами: побитной передачи данных повторителями, передачи кадров коммутаторами и передачи пакетов сетевых протоколов маршрутизаторами.

Пример структуры корпоративного концентратора приведен на рис. 14.7. Он имеет несколько шин для образования независимых разделяемых сегментов Ethernet 10 Мбит/с, Token Ring и FDDI, а также высокоскоростную шину в 10 Гбит/с для передачи кадров и пакетов между модулями коммутации и маршрутизации. Каждый из модулей имеет внешние интерфейсы для подключения конечных узлов и внешних коммуникационных устройств — повторителей, коммутаторов, а также несколько интерфейсов с внутренними шинами концентратора. Концентратор рассчитан на подключение конечных узлов в основном к внешним интерфейсам повторителей (для образования разделяемых сегментов) и коммутаторов (для поддержки микросегментации). Уже готовые сегменты, то есть образованные внешними повторителями и коммутаторами, могут подключаться к внешним интерфейсам коммутаторов и маршрутизаторов корпоративного концентратора. Дальнейшее соединение разделяемых сегментов и коммутируемых узлов и сегментов происходит модулями коммутации и маршрутизации концентратора по внутренним связям с помощью высокоскоростной шины. Конечно, модули могут связываться между собой и через внешние интерфейсы, но такой способ не рационален, так как скорость обмена ограничивается при этом скоростью протокола интерфейса, например 10 Мбит/с или 100 Мбит/с. Внутренняя же шина соединяет модули на гораздо более высокой скорости, примерно $10/N$ Гбит/с, где N — количество портов, одновременно требующих обмена. Внешние связи между модулями превращают корпоративный концентратор просто в стойку с установленными модулями, а внутренний обмен делает эту стойку единым устройством с общей системой программного управления трафиком. Обычно для конфигурирования модулей и связей между ними производители корпоративных концентраторов сопровождают их удобным программным обеспечением с графическим интерфейсом. Отдельный модуль управления выполняет общие для всего концентратора функции: управления по протоколу SNMP, согласование таблиц коммутации и маршрутизации в разных модулях, возможно использование этого модуля как межмодульной коммутационной фабрики вместо общей шины.

1 Читайте о протоколе SNMP в части VI.

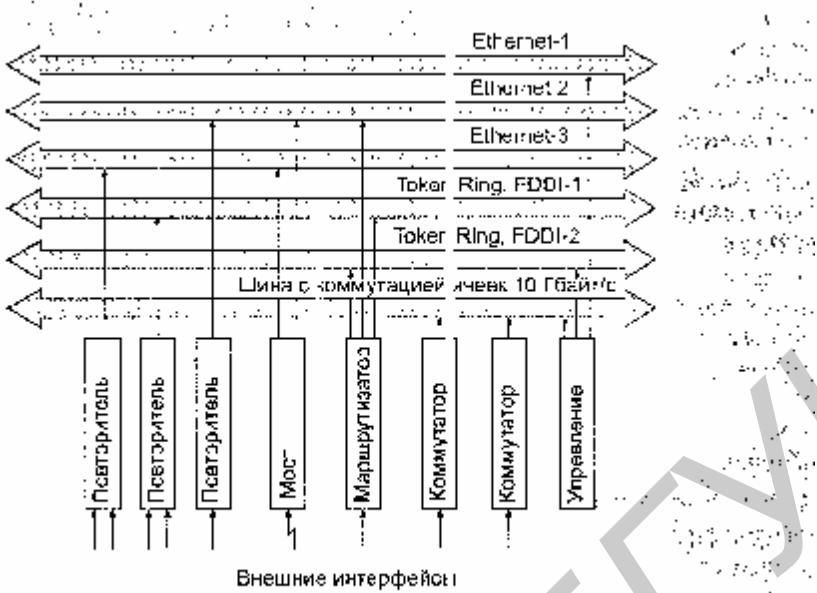


Рис. 14.7. Структура корпоративного модульного концентратора

Примерами корпоративных многофункциональных концентраторов могут служить устройства System 5000 компании Nortel Networks, MMAC-Plus компании Cabletron Systems, CoreBuilder 6012 компании 3Com.

Ввиду того что отказ корпоративного модульного концентратора приводит к очень тяжелым последствиям, в их конструкцию вносятся многочисленные средства обеспечения отказоустойчивости.

Стирание граней между коммутаторами и маршрутизаторами

В классическом понимании терминов коммутатор — это устройство, принимающее решение о продвижении пакетов на основании заголовков протоколов уровня, то есть протоколов типа Ethernet или FDDI, а маршрутизатор - устройство, принимающее аналогичное решение на основании заголовков протоколов 3-го уровня, то есть уровня протоколов IP или IPX. В настоящее время наблюдается отчетливая тенденция по совмещению в одном устройстве функций коммутатора и маршрутизатора.

Соотношение коммутации и маршрутизации в корпоративных сетях

До недавнего времени сложившимся информационным потокам корпоративной сети наилучшим образом соответствовала следующая иерархическая структура. На нижнем уровне (уровне отделов) располагались сегменты сети, построенные на быстро работающих повторителях и коммутаторах. Сегменты включали в себя как рабочие станции, так и серверы. Как уже упоминалось, к большинству случаев было справедливо эмпирическое соотношение 80/20, в соответствии с которым основная часть трафика (80 %) циркулировала внутри сегмента, то есть порождалась запросами пользователей рабочих станций к серверам своего

же сегмента.

На более высоком уровне располагался маршрутизатор, к которому подключалось сравнительно небольшое количество внутренних сетей, построенных на коммутаторах. Через порты маршрутизатора проходил трафик обращений рабочих станций одних сетей к серверам других сетей. Известно, что маршрутизатор затрачивает больше времени на обработку каждого пакета, чем коммутатор, поскольку он выполняет более сложную обработку трафика, включая интеллектуальные алгоритмы фильтрации, выбор маршрута при наличии нескольких возможных путей и т. п. С другой стороны, трафик, проходящий через порты маршрутизатора, был менее интенсивный, чем внутрисегментный, поэтому сравнительно низкая производительность маршрутизатора не делала его узким местом.

Сегодня ситуация в корпоративных сетях быстро меняется. Количество пользователей стремительно растет. Пользователи избавляются от устаревающих текстовых приложений, отдавая предпочтение web-интерфейсу. А завтра эти же пользователи будут работать с аудио, видео, push и другими, абсолютно новыми приложениями, основанными на новых технологиях распространения пакетов, таких как IP Multicast и RSVP. Сегодня не работает и старое правило 80/20 — большое количество информации берется из публичных серверов Интернета, а также из web-серверов других подразделений предприятия, создавая большой межсетевой трафик. Существующие сети не оптимизировались для таких не-предсказуемых потоков трафика, когда каждый может общаться почти с каждым. А с проникновением в корпоративные сети технологии Gigabit Ethernet эта проблема обострится еще больше.

Таким образом, сегодня образовался большой разрыв между производительностью типичного маршрутизатора и типичного коммутатора. В этой ситуации возможны два решения: либо вообще отказаться от маршрутизации, либо увеличить ее производительность.

Отказ от маршрутизации

За последние годы основные усилия были сосредоточены в первом направлении: применять маршрутизацию как можно реже, только там, где от нее никак нельзя отказаться. Например, на границе между локальной и глобальной сетями. Отказ от маршрутизаторов означает переход к так называемой плоской сети, то есть сети, построенной только на коммутаторах, а значит, и отказ от всех интеллектуальных возможностей обработки трафика, присущих маршрутизаторам. Такой подход повышает производительность, но приводит к потере всех преимуществ, которые давали маршрутизаторы, а именно:

- маршрутизаторы более надежно, чем коммутаторы, изолируют части большой составной сети друг от друга, защищая их от ошибочных кадров, порождаемых неисправным программным или аппаратным обеспечением других сетей (например, от широковещательных штормов);
- маршрутизаторы обладают более развитыми возможностями защиты от несанкционированного доступа за счет функций анализа и фильтрации трафика

на более высоких уровнях: сетевом и транспортном;

- сеть, не разделенная маршрутизаторами, имеет ограничения на число узлов (для популярного протокола IP это ограничение составляет 255 узлов для сетей самого доступного класса С).

Из этого следует, что в сети необходимо сохранять функции маршрутизации в привычном смысле этого слова.

Что касается второго направления — повышения производительности маршрутизаторов, — сложилось так, что самые активные действия в этом направлении были предприняты производителями коммутаторов, наделявшими свои продукты некоторыми возможностями маршрутизаторов. Именно в модифицированных коммутаторах были впервые достигнуты скорости маршрутизации в 5-7 миллионов пакетов в секунду, а также опробованы многие важные концепции ускорения функций маршрутизации.

Коммутаторы 3-го уровня с классической маршрутизацией

Термин «коммутатор 3-го уровня» употребляется для обозначения целого спектра коммутаторов различного типа, в которые встроены функции маршрутизации пакетов. Функции коммутации и маршрутизации могут быть совмещены двумя способами;

- классическим, когда выполняется маршрутизация каждого пакета, требующего передачи из сети в сеть, а коммутация выполняется для пакетов, принадлежащих одной сети;
- нестандартным способом ускоренной маршрутизации, когда маршрутизируется несколько первых пакетов устойчивого потока, а все остальные пакеты этого потока коммутируются.

Рассмотрим первый способ.

Классический коммутатор 3-го уровня подобно обычному коммутатору захватывает все кадры своими портами независимо от их MAC-адресов, а затем принимает решение о коммутации или маршрутизации каждого кадра. Если кадр имеет MAC-адрес назначения, отличный от MAC-адреса порта маршрутизатора, то этот кадр коммутируется. Если устройство поддерживает технику VLAN, то перед передачей кадра проверяется принадлежность адресов назначения и источника одной виртуальной сети.

Если же кадр направлен непосредственно MAC-адресу какого-либо порта маршрутизатора, то он маршрутизируется стандартным образом. Коммутатор 3-го уровня может поддерживать динамические протоколы маршрутизации, такие как RIP и OSPF, а может полагаться на статическое задание маршрутов или на получение таблицы маршрутизации от другого маршрутизатора.

Такие комбинированные устройства появились сразу после разработки коммутаторов, поддерживающих виртуальные локальные сети (VLAN), для связи которых требовался маршрутизатор. Размещение маршрутизатора в одном корпусе с коммутатором позволяло получить некоторый выигрыш в производительности, например, за счет исключения одного этапа буферизации пакета, когда он передается из коммутатора в маршрутизатор. Хотя такие устройства с рав-

ным успехом можно называть маршрутизирующими коммутаторами или коммутирующими маршрутизаторами, за ними закрепилось название коммутаторов 3-го уровня.

Примерами таких коммутаторов могут служить хорошо известные коммутаторы LANplex (теперь CoreBuilder) 6000 и 2500 компании 3Com. В этих устройствах совместно используются специализированные большие интегральные микросхемы (ASIC), RISC- и CISC-процессоры. Микросхемы ASIC обеспечивают коммутацию пакетов и их первичный анализ при маршрутизации, RISC-процессоры выполняют основную работу по маршрутизации, а CISC-процессоры реализуют функции управления. За счет такого распараллеливания процесса функционирования подсистем коммутации и маршрутизации достигается достаточно высокий уровень производительности. Так, система CoreBuilder 2500, имеющая один блок коммутации/маршрутизации, способна маршрутизировать 98 тысяч IP-пакетов в секунду (без их потери) на полной скорости каналов связи. Более мощная система CoreBuilder 6000 но данным компании 3Com в конфигурации с 88 портами Fast Ethernet маршрутизирует до трех миллионов пакетов в секунду.

Более быстродействующей реализацией данного подхода являются устройства, в которых функции маршрутизации перенесены из универсального центрального процессора в специализированные заказные микросхемы портов. При этом ускорение процесса маршрутизации происходит не только за счет распараллеливания работы между несколькими процессорами, но и за счет использования специализированных процессоров вместо универсальных процессоров типа Motorola или Intel. Примеры этого подхода — коммутатор CoreBuilder 3500 компании 3Com, маршрутизирующий коммутатор Accelar 1200 компании Nortel Networks.

По данным фирм-производителей коммутаторы 3-го уровня CoreBuilder 3500 и Accelar 1200 способны маршрутизировать соответственно до 4 и 7 миллионов пакетов в секунду. С такой же скоростью они коммутируют поступающие кадры, что говорит о высокой эффективности реализованных в ASIC алгоритмах маршрутизации.

Подход, связанный с переносом процедур маршрутизации из программируемых процессоров, пусть и специализированных, в схемы БИС, работающие по жестким алгоритмам, имеет один принципиальный недостаток — ему недостает гибкости. При необходимости изменения протокола или набора протоколов требуется перепроектировать БИС, что очевидно подразумевает очень большие затраты времени и средств по сравнению с изменением программного обеспечения маршрутизатора. Поэтому быстродействующие маршрутизаторы переносят в БИС только несколько базовых протоколов сетевого уровня, чаще всего IP и IPX, делая такие маршрутизаторы узко специализированными.

Маршрутизация потоков

Еще один тип коммутаторов 3-го уровня — это коммутаторы, которые ускоряют процесс маршрутизации за счет выявления устойчивых потоков в сети и об-

работки по схеме маршрутизации только нескольких первых пакетов потока. Многие фирмы разработали подобные схемы, однако до сих пор они являются нестандартными, хотя работа над стандартизацией этого подхода идет в рамках одной из рабочих групп IETF. Существуют компании, которые считают эти попытки ошибочными, вносящими ненужную путаницу в и так непростую картину работы стека протоколов в сети. Наиболее известной компанией, занявшей такую позицию, является компания Nortel Networks, маршрутизаторы которой Accelar 1200 работают по классической схеме. Тем не менее количество компаний, разработавших протоколы ускоренной маршрутизации, в основном ускоренной IP-маршрутизации, довольно велико.

Поток — это последовательность пакетов, имеющих некоторые общие свойства, по меньшей мере, у них должны совпадать адрес отправителя и адрес получателя, и тогда их можно отправлять по одному и тому же маршруту. Желательно, чтобы пакеты потока имели одно и то же требование к качеству обслуживания.

Ввиду разнообразия предложенных схем опишем только основную идею, лежащую в их основе.

Если бы все коммутаторы/маршрутизаторы, показанные на рис. 14.8, работали по классической схеме, то каждый пакет, отправляемый из рабочей станции, принадлежащей одной IP-сети, серверу, принадлежащему другой IP-сети, проходил бы через блоки маршрутизации всех трех устройств.

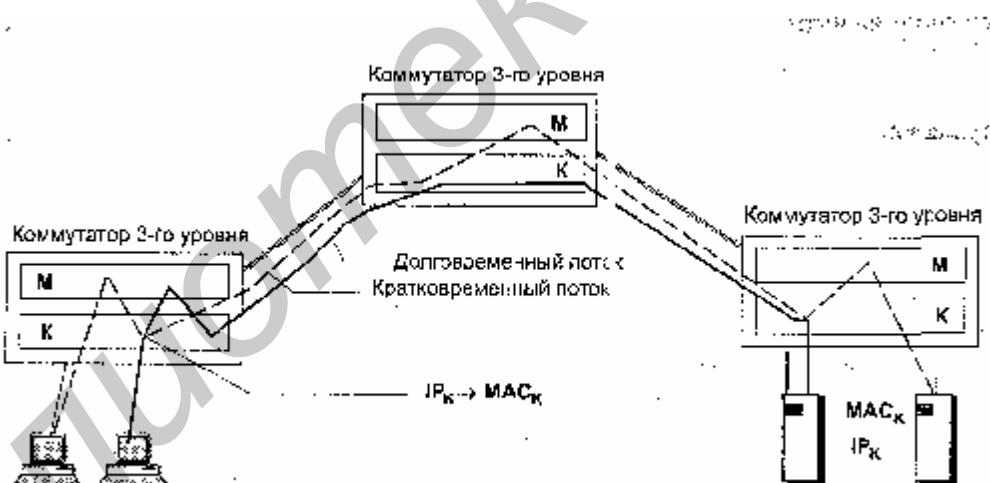


Рис. 14.8. Ускоренная маршрутизация потока пакетов

В схеме ускоренной маршрутизации такую обработку проходит только несколько первых пакетов долговременного потока, то есть классическая схема работает до тех пор, пока долговременный поток не будет выявлен.

После этого первый коммутатор на пути следования потока выполняет нестандартную обработку пакета — он помещает в кадр канального протокола, например Ethernet, не MAC-адрес порта следующего маршрутизатора, а MAC-адрес узла назначения, который на рисунке обозначен как MAC. Как только замена произведена, кадр с таким MAC-адресом перестает поступать на блоки маршрутизации второго и третьего коммутатора/маршрутизатора, а проходит

только через блоки коммутации этих устройств. Процесс передачи пакетов действительно ускоряется, так как они не проходят многократно повторяющиеся этапы маршрутизации. В то же время защитные свойства маршрутизаторы сохраняют, так как первые пакеты проверяются на допустимость передачи в сеть назначения, поэтому сохраняются и фильтрация широковещательного шторма, и защита от несанкционированного доступа, и другие преимущества сети, разделенной на подсети.

Для реализации описанной схемы нужно решить несколько проблем. Первая — на основании каких признаков определяется долговременный поток. Решить эту проблему достаточно легко, поскольку основные подходы к решению очевидны — совпадение адресов и портов соединения, общие признаки качества обслуживания, некоторый порог одинаковых пакетов для фиксации долговременности. Вторая проблема более серьезная. На основании какой информации первый маршрутизатор может узнать MAC-адрес узла назначения? Этот узел находится за пределами непосредственно подключенных к первому маршрутизатору сетей, поэтому использование протокола ARP здесь не поможет. Именно здесь расходятся пути большинства фирменных технологий ускоренной маршрутизации. Многие компании разработали собственные служебные протоколы, с помощью которых маршрутизаторы запрашивают этот MAC-адрес друг у друга, пока последний на пути маршрутизатор не выяснит его с помощью протокола ARP.

Фирменные протоколы используют как распределенный подход, когда вес маршрутизаторы равны в решении проблемы нахождения MAC-адреса, так и централизованный, когда в сети существует выделенный маршрутизатор, который помогает ее решить для всех.

Примерами коммутаторов 3-го уровня, работающими по схеме ускоренной IP-маршрутизации, являются коммутаторы SmartSwitch компании Cabletron, а также коммутатор Catalyst 5000 компании Cisco, выполняющий спой функции в плане распознавания потоков и определения их адресной информации совместно с маршрутизаторами Cisco 7500 по технологии Cisco NetFlow, и ряд других.

Выводы

- Типичный маршрутизатор представляет собой сложный специализированный компьютер, который работает под управлением специализированной операционной системы, оптимизированной для выполнения операций построения таблиц маршрутизации и продвижения пакетов на их основе.
- Маршрутизатор часто строится по мультипроцессорной схеме, причем используется симметричное мультипроцессорование, асимметричное мультипроцессорование и их сочетание. Наиболее рутинные операции обработки пакетов выполняются программно специализированными процессорами или аппаратно большими интегральными схемами (БИС/ASIC). Более высоконивневые действия выполняют программно универсальные процессоры.
- По областям применения маршрутизаторы делятся на магистральные маршрутизаторы, маршрутизаторы региональных подразделений, маршрутизаторы

удаленных офисов и маршрутизаторы локальных сетей — коммутаторы 3-го уровня.

- Основными характеристиками маршрутизаторов являются: общая производительность, измеряемая в пакетах в секунду, набор поддерживаемых сетевых протоколов и протоколов маршрутизации, набор поддерживаемых сетевых интерфейсов глобальных и локальных сетей.
- К числу дополнительных функций маршрутизатора относятся одновременная поддержка сразу нескольких сетевых протоколов и нескольких протоколов маршрутизации, возможность приоритетной обработки трафика, разделение функций построения таблиц маршрутизации и продвижения пакетов между маршрутизаторами разного класса на основе готовых таблиц маршрутизации.
- Основной особенностью коммутаторов 3-го уровня является высокая скорость выполнения операций маршрутизации за счет их перенесения на аппаратный уровень — уровень БИС/ASIC.
- Многие фирмы разработали собственные протоколы ускоренной маршрутизации долговременных потоков в локальных сетях, которые маршрутизируют только несколько первых пакетов потока, а остальные пакеты коммутируют на основе MAC-адресов.
- Корпоративные многофункциональные концентраторы представляют собой устройства, в которых на общей внутреннейшине объединяются модули различного типа — повторители, мосты, коммутаторы и маршрутизаторы. Такое объединение дает возможность программного конфигурирования сети с определением состава подсетей и сегментов вне зависимости от их физического подключения к тому или иному порту устройства.

Трансляция сетевых адресов

Маршрутизация в составной сети осуществляется на основе тех адресов назначения, которые помещены в заголовки пакетов. Как правило, эти адреса остаются неизменными с момента их формирования отправителем до момента поступления на узел получателя. Однако из этого правила есть исключения. В некоторых случаях при маршрутизации пакетов в одной части составной сети используются одни адреса сетевого уровня, а в другой — другие. С такой ситуацией мы сталкиваемся, например, при туннелировании, когда пакет передается через транзитную сеть, построенную по технологии, отличающейся от технологии сетей отправителя и получателя. В широко применяемой сегодня технологии трансляции сетевых адресов (Network Address Translation, NAT) также предполагается продвижение пакета во внешней сети (в Интернете) на основании адресов, отличных от тех, которые используются для маршрутизации пакета во внутренней, корпоративной сети.

Причины подмены адресов

Одной из наиболее популярных причин использования технологии NAT явля-

ется дефицит IP-адресов. Если по каким-либо причинам предприятию, у которого имеется потребность подключения к Интернету, не удается получить у поставщика услуг необходимого количества глобальных IP-адресов, то оно может прибегнуть к технологии NAT. В этом случае для адресации внутренних узлов используются специально зарезервированные для этих целей так называемые частные (private) адреса:

10.0.0.0-10.255.255
172.16.0.0-172.31.255.255
192.168.0.0-192.168.255.255

Приведенные выше диапазоны адресов составляют огромное адресное пространство, достаточное для нумерации узлов сетей практически любых размеров. Эти адреса исключены из множества централизованно распределяемых, а значит, пакеты с такими адресами назначения будут отброшены маршрутизаторами на магистралях Интернета. Для того чтобы узлы с частными адресами могли через Интернет связываться между собой или с узлами, имеющими глобальные адреса, необходимо использовать технологию NAT.

Потребность в трансляции IP-адресов возникает и тогда, когда предприятие из соображений безопасности желает скрыть адреса узлов сессии сети, чтобы не дать возможности злоумышленникам составить представление о структуре и масштабах своей сети, а также о структуре и интенсивности исходящего и входящего трафиков. В этом случае технология NAT также оказывается полезной.

Традиционная технология NAT

Технология трансляции сетевых адресов имеет несколько разновидностей, наиболее популярная из которых — традиционная технология NAT (Traditional NAT) — позволяет узлам из частной сети прозрачным для пользователей образом получать доступ к узлам во внешних сетях. Подчеркнем, что сеансы связи в традиционной технологии NAT являются односторонними, исходящими из частной сети. (Направление сеанса в данном случае определяется положением инициатора: если обмен данными инициируется приложением, работающим на узле внутренней сети, то этот сеанс называется исходящим, несмотря на то, что в рамках данного сеанса в сеть могут поступать данные извне.) Традиционная технология NAT допускает сеанс обратного направления в виде исключения, используя статическое отображение адресов для некоторого выбранного заранее ограниченного набора узлов, для которых устанавливается взаимно однозначное соответствие между внутренними и внешними адресами.

Традиционная технология NAT подразделяется на базовую технологию трансляции сетевых адресов (Basic Network Address Translation, Basic NAT), в которой для отображения используются только IP-адреса, и технологию трансляции адресом и портов (Network Address Port Translation, NAPT), в которой для отображения привлекаются еще и так называемые транспортные идентификаторы, в качестве которых чаще всего используются порты TCP/UDP.

Базовая технология NAT

Этот вариант технологии NAT работает следующим образом (рис. 14.9). Сеть предприятия образует тупиковый домен, узлы которого описываются набором частных адресов. Программное обеспечение NAT, установленное на периферийном устройстве, связывающем сеть предприятия с внешней сетью, динамически отображает набор частных адресов $\{IP^*\}$ на набор глобальных адресов $\{IP\}$, полученных предприятием от поставщика услуг и привязанных к внешнему интерфейсу.

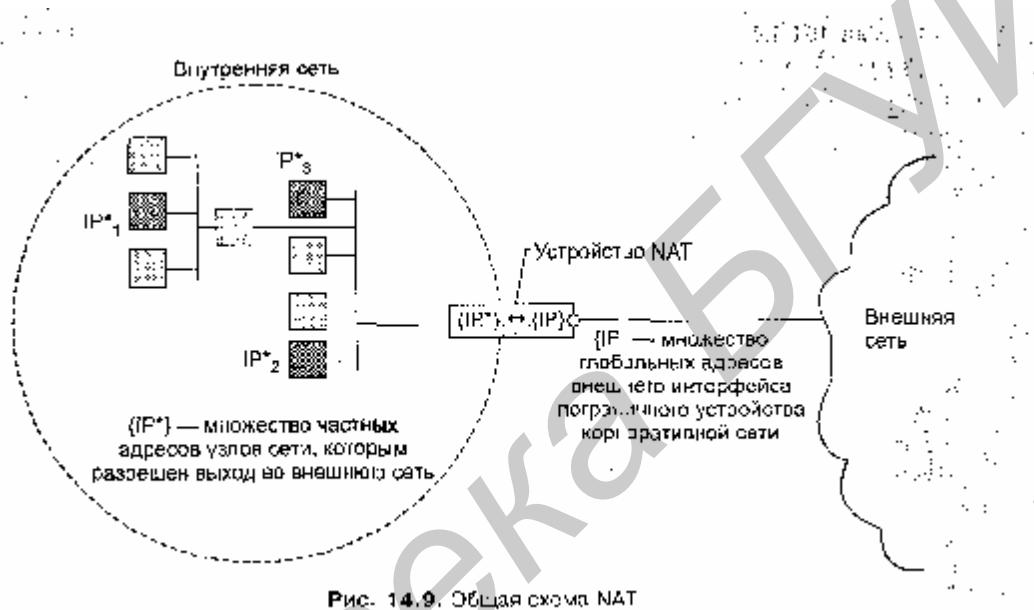


Рис. 14.9. Общая схема NAT

Если количество локальных узлов меньше или равно имеющегося количества глобальных адресов, то для каждого частного адреса гарантировано отображение на глобальный адрес. Понятно, что в этом случае нет проблемы дефицита адресов,

и использование технологии NAT может преследовать только цели безопасности.

В каждый момент времени количество внутренних узлов, которые получают возможность взаимодействовать с внешней сетью, ограничивается количеством адресов в глобальном наборе. Внутренние адреса некоторых узлов могут статически отображаться на определенные глобальные адреса, в этом случае к ним может быть организован доступ извне с помощью фиксированного адреса. Соответствие внутренних адресов внешним задается таблицей, поддерживаемой устройством NAT.

В нескольких тупиковых доменах могут быть совпадающие частные адреса. Например, сети А и В (рис. 14.10) используют для внутренней адресации один и тот же блок адресов 10.0.0.0/8. В то же время внешние адреса обеих сетей (181.230.25.1/24, 181.230.25.2/24 и 181.230.25.3/24 в сети А и 185.127.125.2/24 185.127.125.3/24 185.127.125.4/24 в сети В) - уникальны глобально, так что никакие другие узлы в составной сети или устройства NAT их не используют.

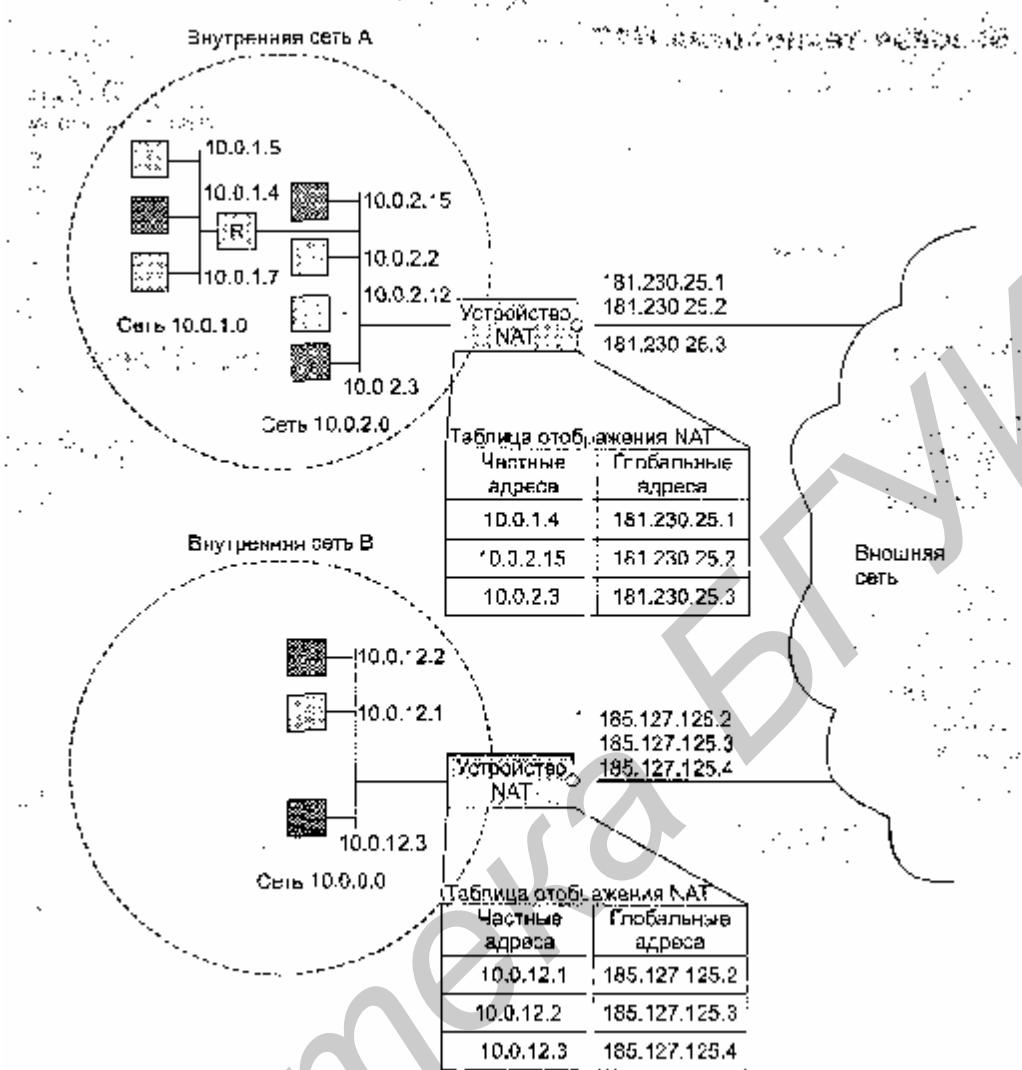


Рис. 14.10. Трансляция адресов для исходящего трафика в базовой технологии NAT

Объявления протоколов маршрутизации о внешних сетях распространяются пограничными устройствами во внутренних сетях и обрабатываются внутренними маршрутизаторами. Обратное утверждение неверно — маршрутизаторы внешних сетей не «видят» внутренние сети, так как объявления о них отфильтровываются при передаче на внешние интерфейсы.

Когда узел 10.0.1.4 сети А хочет послать пакет хосту 10.0.12.2 сети В, то он делает это, используя глобальный адрес 185.127.125.4/24 в качестве адреса назначения. Узел-источник направляет пакет своему маршрутизатору по умолчанию, которому известен маршрут к сети 185.127.125.0/24. Маршрутизатор передает пакет на пограничный маршрутизатор, которому также известен дальний маршрут. Однако перед отправкой пакета протокол NAT, работающий на пограничном маршрутизаторе, используя таблицу отображения внутренних адресов во внешние, транслирует частный адрес источника 10.0.1.4 в соответствующий ему глобальный адрес 181.230.25.1/24. Когда пакет после путешествия по составной сети поступает на внешний интерфейс устройства NAT сети В, глобальный адрес назначения 185.127.125.4/24 преобразуется в частный адрес

10.0.12.2. Пакеты, передаваемые в обратном направлении, проходят аналогичную процедуру трансляции адресов.

Заметим, что в описанной операции не требуется участие узлов отправителя и получателя, то есть она прозрачна для пользователей.

Трансляция адресов и номеров портов

Пусть некоторая организация имеет частную сеть IP и глобальную связь с поставщиком услуг Интернета. Внешнему интерфейсу пограничного маршрутизатора назначен глобальный адрес, а остальным узлам сети организации назначены частные адреса. Технология трансляции адресов и портов (NAPT) позволяет всем узлам внутренней сети одновременно взаимодействовать с внешними сетями, используя единственный зарегистрированный IP-адрес. Может возникнуть законный вопрос, каким образом внешние пакеты, поступающие в ответ на запросы из сети, находят узел-отправитель, ведь в поле адреса источника всех пакетов, отправляющихся во внешнюю сеть, помещается один и тот же адрес — адрес внешнего интерфейса? Для однозначной идентификации узла отправителя привлекается дополнительная информация — номер порта UDP или TCP. Но и это не вносит полной ясности, поскольку из внутренней сети может исходить несколько запросов с совпадающими номерами портов отправителя, а значит, опять возникает вопрос об однозначности отображения единственного глобального адреса на набор внутренних адресов. Решение состоит в том, что при прохождении пакета из внутренней во внешнюю сеть каждой паре {внутренний частный адрес; номер порта TCP/IP отправителя} ставится в соответствие пара {глобальный IP-адрес внешнего интерфейса; назначенный номер порта TCP/IP}. Назначенный номер порта выбирается произвольно, но он должен быть уникальным в пределах всех узлов, получающих выход во внешнюю сеть. Соответствие фиксируется в таблице.

Эта модель удовлетворяет требованиям большинства сетей средних размеров для доступа к внешним сетям при использовании единственного зарегистрированного IP-адреса, полученного от поставщика услуг.

На рис. 14.11 приведен пример, когда тупиковая сеть А использует внутренние адреса из блока 10.0.0.0/8. Внешнему интерфейсу маршрутизатора этой сети поставщиком услуг назначен адрес 181.230.25.1.

Когда хост 10.0.1.4 сети А посыпает макет службы `lei net` хосту 136.56.28.8 во внешней сети, то он, отправляя пакет маршрутизатору, в качестве адреса назначения указывает глобальный адрес 136.56.28.8. Маршрутизатор знает путь к сети 136.56.0.0/16, поэтому он передаст пакет па внешний интерфейс. Но при этом устройство NAPT маршрутизатора транслирует адрес источника 10.0.1.4 и номер порта TCP 1245 (который был присвоен клиенту `telnet` в узле-источнике) в глобально уникальный адрес 181.230.25.1 и уникально назначенный порт TCP, например 3451. Пакеты, идущие в обратном направлении, подвергаются подобной трансляции для адресов назначения. Когда получатель генерирует ответное сообщение, то он в качестве адреса назначения указывает единственный зарегистрированный глобальный адрес сети А, являющийся адресом внешнего

интерфейса устройства NAT, а также назначенный номер порта TCP/UDP, взятый из поля отправителя пришедшего пакета (в данном примере это 3451). При поступлении ответного пакета на устройство NAT сети А именно по номеру порта в таблице трансляции выбирается нужная строка. По ней определяется внутренний IP-адрес соответствующего узла и действительный номер порта. Эта процедура трансляции полностью прозрачна для конечных узлов.

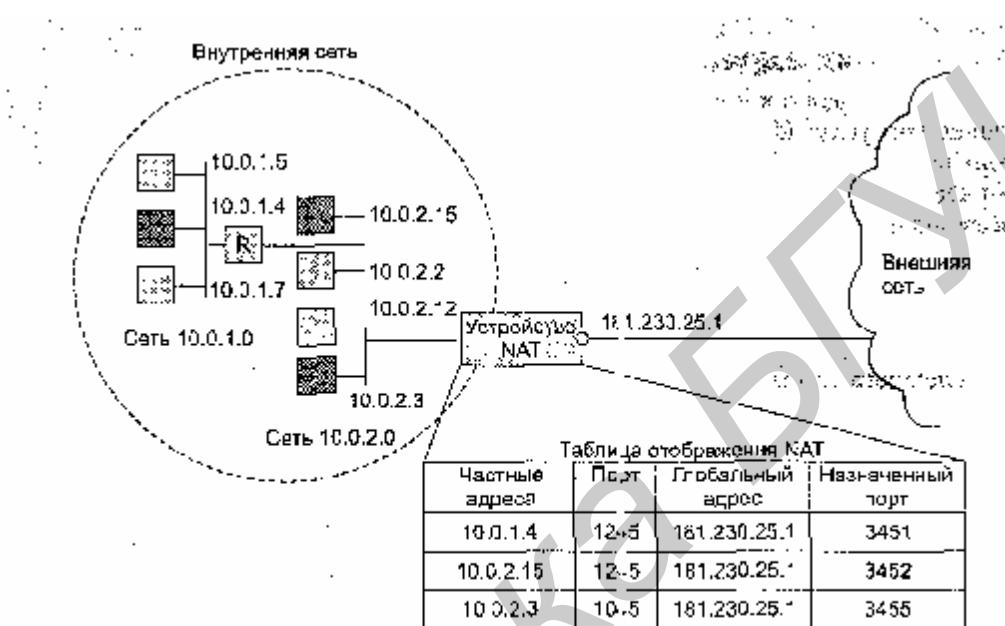


Рис. 14.11. Трансляция адресов и номеров портов для исходящих сеансов TCP/UDP в технологии NAT

В варианте NAPT разрешаются только исходящие из частной сети сеансы TCP/UDP. Однако возникают ситуации, когда нужно обеспечить доступ к некоторому узлу внутренней сети извне. В простейшем случае, когда сервис является зарегистрированным, то есть сервисом, которому присвоен «хорошо известный» номер порта (например, Web или DNS), и, кроме того, этот сервис представлен во внутренней сети в единственном экземпляре, задача решается достаточно просто. Сервис и узел, на котором он работает, однозначно определяются «хорошо известным» зарегистрированным номером порта сервиса.

Двойная трансляция

Помимо традиционной технологии NAT существуют и другие варианты трансляции сетевых адресов, например двойная трансляция, при которой модифицируются оба адреса — и источника, и назначения (в отличие от традиционной технологии NAT, когда модифицируется только один адрес). Двойная трансляция необходима, когда частное и внешние адресные пространства имеют коллизии. Наиболее часто это происходит, когда внутренний домен получает публичные адреса, которые принадлежат другой организации. Эта ситуация может возникнуть из-за того, что сеть организации была изначально изолированной и

адреса назначались произвольно, причем из глобального пространства. Или же такая коллизия может быть следствием смены поставщика услуг, когда организация хочет сохранить старые адреса для узлов внутренней сети. В том и другом случаях общим является то, что адрес внутреннего хоста может совпадать с адресом внешнего хоста, а значит, посланный изнутри пакет не попадет вовне, а будет передан внутреннему хосту.

Двойная трансляция работает следующим образом. Когда хост А хочет инициировать сеанс с хостом Х, то он генерирует запрос DNS для хоста Х. Шлюз DNS перехватывает этот запрос и в ответе, который он возвращает хосту А, заменяет адрес для хоста Х адресом, который может правильно маршрутизоваться в локальном домене (например, Host-XPRIME). Хост А затем инициирует взаимодействие с хостом Host-XPRIME. Когда пакеты пересекают устройство NAT, то адрес источника транслируется (как и в традиционной технологии NAT), а адрес назначения транслируется в Host-X. На обратном пути выполняется аналогичная трансляция.

Выводы

- В широко применяемой сегодня технологии трансляции сетевых адресов (Network Address Translation, NAT) пакеты маршрутизируются во внешней сети (в Интернете) на основании адресов, отличных от тех, которые используются для маршрутизации пакета во внутренней, корпоративной сети.
- Одной из причин применения технологии NAT является дефицит IP-адресов. Она позволяет использовать для адресации внутренних узлов не глобально-уникальные IP-адреса, а специально зарезервированные для этих целей так называемые частные (private) адреса.
- Потребность в технологии NAT возникает и тогда, когда предприятие из соображений безопасности желает скрыть адреса узлов своей сети, чтобы не дать возможности злоумышленникам составить представление о ее структуре.
- Наиболее популярный вариант технологии трансляции сетевых адресов — традиционная технология NAT (Traditional NAT). Она позволяет узлам из частной сети прозрачным для пользователей образом получать доступ к узлам во внешних сетях.
- Традиционная технология NAT подразделяется на базовую технологию NAT (Basic NAT), использующую для отображения только IP-адреса, и технологию NAPT (Network Address Port Translation), в которой для отображения привлекаются еще и так называемые транспортные идентификаторы, которыми чаще всего являются номера портов TCP/UDP.
- Помимо традиционной технологии NAT существуют и другие варианты трансляции сетевых адресов, например двойная трансляция, при которой модифицируются оба адреса — и источника, и назначения (в отличие от традиционной технологии NAT, когда модифицируется только один адрес).

Средства построения составных сетей стека Novell

Общая характеристика протокола IPX

Протокол Internetwork Packet Exchange (IPX) является оригинальным протоколом сетевого уровня стека Novell, разработанным в начале 80-х годов на основе протокола Internetwork Datagram Protocol (IDP) компании Xerox.

Протокол IPX соответствует сетевому уровню модели ISO/OSI (рис. 14.12) и, как и протокол IP, поддерживает только дейтаграммный (без установления соединений) способ обмена сообщениями. Б сети NetWare самая быстрая передача данных при наиболее экономном расходовании памяти реализуется именно протоколом IPX.

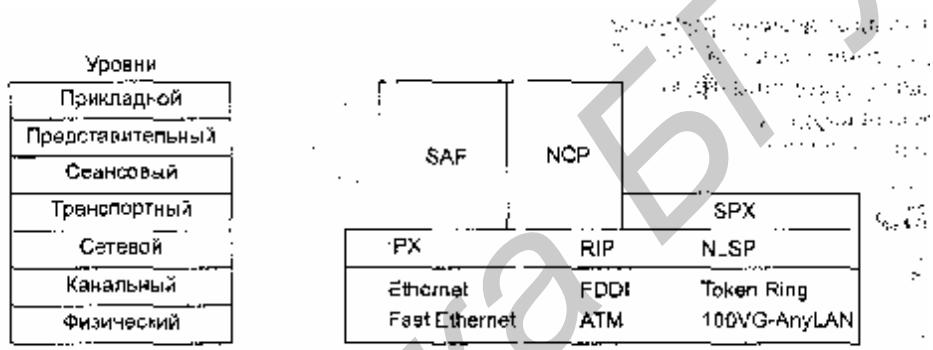


Рис. 14.12. Соответствие протоколов IPX/SPX семиуровневой модели OSI

Надежную передачу пакетов может осуществлять транспортный протокол SPX (Sequenced Packet Exchange), который работает с установлением соединения и восстанавливает пакеты при их потере или повреждении. Как видно из рисунка, использование протокола SPX не является обязательным при передаче сообщений протоколами прикладного уровня.

Прикладной уровень стека IPX/SPX составляют два протокола: NCP и SAP. Протокол NCP (NetWare Core Protocol) поддерживает все основные службы операционной системы Novell NetWare — файловую службу, службу печати и т. д. Протокол SAP (Service Advertising Protocol) выполняет вспомогательную роль. С помощью протокола SAP каждый компьютер, который готов предоставить какую-либо службу для клиентов сети, объявляет об этом широковещательно по сети, указывая в SAP-пакетах тип службы (например, файловая), а также свой сетевой адрес. Наличие протокола SAP позволяет резко уменьшить административные работы по конфигурированию клиентского программного обеспечения, так как всю необходимую информацию для работы клиенты узнают из объявлений SAP (кроме маршрутизаторов по умолчанию, о которых можно узнать с помощью протокола IPX).

В отличие от протокола IP, который изначально разрабатывался для глобальных сетей, протокол IPX создавался для применения в локальных сетях. Именно поэтому он является одним из самых экономичных протоколов в отношении требований к вычислительным ресурсам и хорошо работает в сравнительно не-

больших локальных сетях.

Специфика адресации в протоколе IPX является как достоинством, так и недостатком этого протокола. Протокол IPX работает с сетевыми адресами, включающими три компонента:

- номер сети (4 байта);
- номер узла (6 байт);
- номер сокета (2 байта).

Номер сети в отличие от протокола IP имеет всегда фиксированную длину — 4 байта. В принципе для корпоративных сетей эта длина является избыточной, так как вряд ли у предприятия возникнет потребность разделить свою сеть на 4 миллиарда подсетей. В период доминирования сетей IPX/SPX компания Novell рассматривала возможность создания единого всемирного центра по распределению IPX-адресов, аналогичного центру InterNIC. Однако стремительный рост популярности Интернета лишил это начинание смысла. Хотя протоколы IPX/SPX по-прежнему работают в огромном количестве корпоративных сетей, заменить IP во всемирной сети они уже не смогут. Надо отметить, что специалисты компании Novell приложили немало усилий, чтобы в новой версии 6 протокол IP приобрел некоторые черты,ственные протоколу IPX, и тем самым облегчил переход пользователей IPX на IPv6 (когда это станет практически необходимым). Обычно все три составляющие IPX-адреса, в том числе и номер сети, записываются в шестнадцатеричной форме.

Под номером узла в протоколе IPX понимается аппаратный адрес узла. В локальных сетях это MAC-адрес узла — сетевого адаптера или порта маршрутизатора. Размер адреса узла в 6 байт отражает происхождение этого поля, но в него можно поместить любой аппаратный адрес, если он укладывается по размерам.

Номер сокета (socket) идентифицирует приложение, которое передает свои сообщения по протоколу IPX. Сокет выполняет в стеке IPX/SPX ту же роль, что порт в протоколах TCP/UDP стека TCP/IP. Наличие этого поля в протоколе сетевого уровня, которым является IPX, объясняется тем, что в стеке Novell прикладные протоколы NCP и SAP взаимодействуют с сетевым уровнем непосредственно, минуя транспортный протокол SPX. Поэтому роль мультиплексора-демультиплексора прикладных протоколов приходится выполнять протоколу IPX, для чего в его пакете необходимо передавать номер сокета прикладного протокола. Протоколы NCP и SAP не пользуются услугами SPX для ускорения работы стека, а скорость работы на маломощных персональных компьютерах начала 80-х годов была одной из основных целей компании Novell. Каждый дополнительный уровень в стеке, хотя бы и такой простой, как UDP, замедляет работу стека. За отказ от транспортного уровня компании Novell пришлось добавить средства восстановления утерянных пакетов в протоколе NCP. Тем не менее прикладные программисты, разрабатывающие свои собственные сетевые приложения для стека IPX/SPX, могут пользоваться протоколом SPX, если не захотят встраивать достаточно сложные алгоритмы скользящего окна в свои программы.

Протокол IPX является одним из наиболее легко настраиваемых протоколов се-

тевого уровня. Номер сети задается администратором только на серверах, а номер узла автоматически считывается из сетевого адаптера компьютера. На клиентском компьютере номер сети не задается — клиент узнает эту информацию из серверных объявлений SAP или от локального маршрутизатора.

Адрес маршрутизатора по умолчанию также не нужно задавать вручную на каждом клиентском компьютере. В протоколе IPX есть специальный запрос, который передается на заранее определенный номер сокета. Если в сети клиента есть маршрутизатор или сервер, выполняющий функцию программного маршрутизатора, то клиент при старте системы выдает такой запрос широковещательно, и все маршрутизаторы сообщают ему свои MAC-адреса, используемые в качестве адреса следующего маршрутизатора.

Как видно из описания, административные издержки при конфигурировании сети IPX/SPX сводятся к минимуму. В частности, отпадает необходимость в протоколе типа ARP, что позволяет не тратить время на выполнение ARP-запросов и ARP-ответов для выяснения соответствия между сетевыми адресами узлов и их MAC-адресами.

Формат пакета протокола IPX

Пакет протокола IPX имеет гораздо более простую структуру по сравнению с пакетом IP, что, собственно, и отражает меньшие функциональные возможности протокола IPX.

Ниже перечислены поля IPX-пакета.

- Контрольная сумма (Checksum) — это 2-байтовое поле, являющееся «пережитком прошлого», которое протокол IPX ведет от протокола IP стека Xerox. Так как низкоуровневые протоколы (например, Ethernet) всегда выполняют проверку контрольных сумм, то IPX не использует это поле и всегда устанавливает его в единицы.
- Длина (Length) занимает два байта и задает размер всего пакета, включая IPX-заголовок и поле данных. Самый короткий пакет (30 байт) включает только IPX-заголовок, а рекомендуемый максимально большой (576 байт) — IPX-заголовок плюс 546 байт данных. Максимальный размер пакета в 576 байт соответствует рекомендациям стандартов Интернета для составных сетей. Протокол IPX вычисляет значение этого поля, основываясь на информации, предоставляемой прикладной программой при вызове функции IPX. IPX-пакет может превосходить рекомендуемый максимум в 576 байт, что и происходит в локальных сетях Ethernet, где используются IPX-пакеты в 1500 байт с полем данных в 1470 байт.
- Управление транспортом (Transport Control) имеет длину 8 бит. Это поле определяет время жизни пакета в хопах. IPX-пакет может пересечь до 15 маршрутизаторов. Протокол IPX устанавливает это однобайтовое поле в 0 до начала передачи, а затем увеличивает его на 1 каждый раз, когда пакет проходит через маршрутизатор. Если счетчик превысит 15, то пакет аннулируется.
- Тип пакета (Packet Type) имеет длину 8 бит. Фирма Xerox определила в свое время определенные значения для различных типов пакетов: прикладные про-

граммы, посылающие IPX-пакеты, должны устанавливать это поле в значение, равное 4. Значение 5 соответствует служебным IPX-пакетам, используемым протоколом SPX в качестве служебных сообщений. Значение 17 указывает на то, что в поле данных IPX-пакета находится сообщение протокола NetWare Core Protocol (NCP) — основного протокола файловой службы NetWare.

□ Адрес назначения (Destination Address) — состоит из трех полей: номера сети назначения, номера узла назначения, номера сокета назначения. Эти поля занимают соответственно 4, 6 и 2 байта.

□ Адрес отправителя (Source Address) ~ номер исходной сети, номер исходного узла, номер исходного сокета. Аналогичны адресным полям назначения.

□ Поле данных (Data). Может занимать от 0 до 546 байт. Поле данных нулевой длины может использоваться в служебных пакетах, например, для подтверждения получения предыдущего пакета.

Из анализа формата пакета можно сделать некоторые выводы об ограничениях протокола IPX.

□ Отсутствует возможность динамической фрагментации на сетевом уровне, В IPX-пакете нет полей, с помощью которых маршрутизатор может разбить слишком большой пакет на части. При передаче пакета в сеть с меньшим значением MTU IPX-маршрутизатор отбрасывает пакет. Протокол верхнего уровня, например NCP, должен последовательно уменьшать размер пакета до тех пор, пока не получит на него положительную квитанцию.

□ Большие накладные расходы на служебную информацию. Сравнительно небольшая максимальная длина поля данных IPX-пакета (546 байт при длине заголовка 30 байт) приводит к тому, что как минимум 5 % данных являются служебными.

□ Время жизни пакета ограничено числом 15, что может оказаться недостаточным для большой сети (для сравнения, в IP-сетях пакет может пройти до 255 промежуточных маршрутизаторов).

□ Отсутствует поле качества сервиса, что не позволяет маршрутизаторам автоматически подстраиваться к требованиям приложения к качеству передачи трафика.

Кроме того, некоторые недостатки сетей Novell связаны не с протоколом IPX, а со свойствами других протоколов стека IPX/SPX. Многие недостатки проявляются при работе стека IPX/SPX на медленных глобальных линиях связи, и это закономерно, так как ОС NetWare оптимизировалась для работы в локальной сети.

Например, неэффективная работа по восстановлению потерянных и искаженных пакетов на низкоскоростных глобальных каналах обусловлена тем, что протокол NCP, который выполняет эту работу, использует метод получения квитанций с простоями. В локальных сетях со скоростью 10 Мбит/с такой метод работал вполне эффективно, а на медленных каналах время ожидания квитанции заметно тормозит работу передающего узла.

Маршрутизация протокола IPX

В целом маршрутизация протокола IPX выполняется аналогично маршрутизации протокола IP. Каждый IPX-маршрутизатор поддерживает таблицу маршрутизации, на основании которой принимается решение о продвижении пакета. IPX-маршрутизаторы обеспечивают одношаговую маршрутизацию, при которой каждый маршрутизатор принимает решение только о выборе следующего на пути маршрутизатора. Возможности маршрутизации от источника в протоколе IPX отсутствуют. Рассмотрим типичную таблицу маршрутизации (табл. 14.7) для протокола IPX.

Таблица 14.7. Таблица маршрутизации протокола IPX

Номер	Следующий маршрутизатор	Порт	Задержка	Хопы
A0000010	_	1	0	0
A0000011	_	2	0	0
000013F4	A0000010-008100E30067	1	3	2
00000120	A0000011-C000023300FA	2	2	1
00000033	A0000010-008100E30055	1	10	5

В поле «Номер сети» указывается шестнадцатеричный адрес сети назначения, а в поле «Следующий маршрутизатор» — полный сетевой адрес следующего маршрутизатора, то есть пара «номер сети — MAC-адрес». MAC-адрес из этой записи переносится в поле адреса назначения кадра канального уровня, например Ethernet, который и переносит IPX-пакет следующему маршрутизатору. IPX-пакет при передаче между промежуточными маршрутизаторами изменений не претерпевает.

Если IPX-маршрутизатор обнаруживает, что сеть назначения — это его непосредственно подключенная сеть, то из заголовка IPX-пакета извлекается номер узла назначения, который является MAC-адресом узла назначения. Этот MAC-адрес переносится в адрес назначения кадра канального уровня, например FDDI. Кадр непосредственно отправляется в сеть, и протокол FDDI доставляет его по этому адресу узлу назначения.

IPX-маршрутизаторы обычно используют два типа метрики при выборе маршрута: расстояние в хопах и задержку в некоторых условных единицах — тиках (ticks). Расстояние в хопах имеет обычный смысл — это количество промежуточных маршрутизаторов, которые нужно пересечь IPX-пакету для достижения сети назначения. Задержка также часто используется в маршрутизаторах и мостах/ коммутаторах для более точного сравнения маршрутов. Однако в IPX-маршрутизаторах традиционно задержка измеряется в тиках таймера персонального компьютера, который выдает сигнал прерывания 18,21 раза в секунду. Эта традиция ведется от первых программных IPX-маршрутизаторов, которые работали в составе операционной системы NetWare и пользовались таймером персонального компьютера для измерения интервалов времени. Напомним, что 1P-маршрутизаторы, а также мосты/коммутаторы, поддерживающие протокол Spanning Tree, измеряют задержку, вносимую какой-либо сетью, в 10-наносекундных единицах передачи одного бита информации, так что сеть

Ethernet оценивается задержкой в 10 единиц. Кроме этого, IPX-маршрутизаторы оценивают задержку не одного бита, а стандартного для IPX-пакета в 576 байт.

Поэтому задержка в тиках для сети Ethernet получается равной 0,00839 тика, а для канала 64 кбит/с — 1,31 тика. Задержка в тиках всегда округляется до целого числа тиков в большую сторону, так что сеть Ethernet вносит задержку в один тик, а канал 64 кбит/с — в два тика. При вычислении метрики в тиках для составного маршрута задержки в тиках складываются.

Две метрики в записях таблицы маршрутизации протокола IPX используются в порядке приоритетов. Наибольшим приоритетом обладает метрика, измеренная в задержках, а если эта метрика совпадает для каких-либо маршрутов, то во внимание принимается расстояние в хопах.

Несмотря на традиции измерения задержки в тиках, IPX-маршрутизаторы могут использовать и стандартные задержки сетей, измеренные в 10-наносекундных интервалах.

IPX-маршрутизаторы могут поддерживать как статические маршруты, так и динамические, полученные с помощью протоколов RIP IPX и NLSP.

Протокол RIP IPX очень близок к протоколу RIP IP. Так как в IPX-сетях маски не применяются, то RIP IPX не имеет аналога RIPv2, передающего маски. Интервал между объявлениями у протокола RIP IPX равен 60 с (в отличие от 30 с у RIP IP). В пакетах RIP IPX для каждой сети указываются обе метрики — в хопах и тиках. Для исключения маршрутных петель IPX-маршрутизаторы используют прием расщепления горизонта.

Время жизни динамической записи составляет 180 с. Недостижимость сети идентифицируется значением числа хопов в 15 (0xF), а тиков — в 0xFFFF.

IPX-маршрутизаторы, как и IP-маршрутизаторы, не передают из сети в сеть пакеты, имеющие широковещательный сетевой адрес. Однако для некоторых типов таких пакетов IPX-маршрутизаторы делают исключения. Это пакеты службы SAP, с помощью которой серверы NetWare объявляют о себе по сети. IPX-маршрутизаторы передают SAP-пакеты во все непосредственно подключенные сети, кроме той, от которой этот пакет получен (расщепление горизонта). Если бы IPX-маршрутизаторы не выполняли таких передач, то клиенты NetWare не смогли бы взаимодействовать с серверами в сети, разделенной маршрутизаторами, в привычном стиле, то есть путем просмотра имеющихся серверов с помощью команды SLIST,

IPX-маршрутизаторы всегда используют внутренний номер сети, который относится не к интерфейсам маршрутизатора, а к самому модулю маршрутизации. Внутренний номер сети является некоторым аналогом сети 127.0.0.0 узлов IP-сетей, однако каждый IPX-маршрутизатор должен иметь уникальный внутренний номер сети, причем его уникальность должна распространяться и на внешние номера IPX-сетей в составной сети.

IPX-маршрутизаторы выполняют также функцию согласования форматов кадров Ethernet. В составных IPX-сетях каждая сеть может работать только с одним из 4 возможных типов кадров IPX. Поэтому если в разных сетях используются разные типы кадров Ethernet, то маршрутизатор посыпает в каждую сеть

тот тип кадра, который установлен для этой сети.

Протокол NLSP (NetWare Link Services Protocol) представляет собой реализацию алгоритма состояния связей для IPX-сетей. В основном он работает аналогично протоколу OSPF сетей TCP/IP.

Выводы

- Стек Novell состоит из четырех уровней: канального, который собственно стеком Novell не определяется; сетевого, представленного протоколом дейтаграммного типа IPX; транспортного, на котором работает протокол надежной передачи данных SPX; прикладного, на котором работает протокол NCP, поддерживающий файловую службу и службу печати, а также протоколы SAP и NDS, выполняющие служебные функции по поиску в сети разделяемых ресурсов.
- Особенностью стека Novell является то, что основной прикладной протокол NCP не пользуется транспортным протоколом SPX, а обращается непосредственно к сетевому протоколу IPX. Это значительно ускоряет работу стека, но усложняет прикладной протокол NCP.
- Сетевой IPX-адрес состоит из номера сети, назначаемого администратором, и номера узла, который в локальных сетях совпадает с аппаратным адресом узла, то есть MAC-адресом. Использование аппаратных адресов узлов на сетевом уровне ускоряет работу протокола, так как при этом отпадает необходимость в выполнении протокола типа ARP. Также упрощается конфигурирование компьютеров сети, так как они узнают свой номер сети от локального маршрутизатора, а номер узла извлекается из сетевого адаптера.
- Недостатком IPX-адресации является ограничение в 6 байт, накладываемое на адрес узла на сетевом уровне. Если какая-либо составная сеть использует аппаратные адреса большего размера (это может произойти, например, в сети X.25), то протокол IPX не сможет доставить пакет конечному узлу такой сети.
- IPX-маршрутизаторы используют протоколы динамической маршрутизации RIP IPX, являющийся аналогом RIP IP, и NLSP, который во многом похож на протокол OSPF сетей TCP/IP.

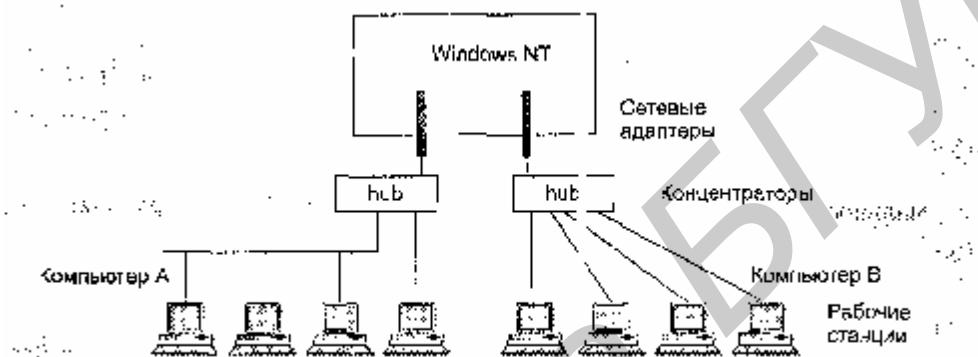
Вопросы и упражнения к части IV

1. В чем состоит отличие задач, решаемых протоколами сетевого уровня в локальных и глобальных сетях?
2. Сравните таблицу моста или коммутатора с таблицей маршрутизатора. Каким образом они формируются? Какую информацию содержат? От чего зависит их объем?
3. Таблица маршрутизации содержит записи о сетях назначения. Должна ли она содержать записи обо всех сетях составной сети или только о некоторых? Если только о некоторых, то о каких именно?

4. Может ли в таблице маршрутизации иметься несколько записей о маршрутизаторах по умолчанию?

5. На следующем рисунке показан Компьютер с двумя сетевыми адаптерами, к которым подсоединены сегменты сети. Компьютер работает под управлением Windows NT. Может ли повлиять на ответ тот факт, что в сегментах используются разные канальные протоколы, например Ethernet и Token Ring? Может ли компьютер А обмениваться данными с компьютером В?

- Да, всегда (А).
- Нет, всегда (В).
- Все зависит от того, как сконфигурирована система Windows NT (С).



6. Сколько уровней имеет стек протоколов TCP/IP? Каковы их функции? Какие особенности этого стека обусловливают его лидирующее положение в мире сетевых технологий?

7. Какие протоколы стека TCP/IP относятся к уровню межсетевого взаимодействия?

8. В чем проявляется ненадежность протокола IP?

9. Могут ли быть обнаружены ошибки на уровне межсетевого взаимодействия? Могут ли они быть исправлены средствами этого уровня?

10. В чем особенности реализации алгоритма скользящего окна в протоколе TCP?

11. В составных сетях используются три вида адресов: символьные, сетевые и локальные. Какие из приведенных ниже адресов могли бы в составной IP-сети являться локальными, а какие нет?

- 6-байтовый MAC-адрес, например 12-B3-3B-51-A2-10 (А);
- адрес X.25, например 25012112654987 (Б);
- 12-байтовый IPX-адрес, например 13.34.B4.OA.C5.10.il.32.54.C5.3B.01 (С);
- адрес VPI/VCI сети ATM (Д).

12. Какие из следующих утверждений верны всегда?

- Каждый порт моста/коммутатора имеет MAC-адрес (А).
- Каждый мост/коммутатор имеет сетевой адрес (Б).
- Каждый порт моста/коммутатора имеет сетевой адрес (С).
- Каждый маршрутизатор имеет сетевой адрес (Д).
- Каждый порт маршрутизатора имеет MAC-адрес (Е),

- Каждый порт маршрутизатора имеет сетевой адрес (F).
13. Какую долю всего множества IP-адресов составляют адреса класса А? Класса В? Класса С?
14. Какие из ниже приведенных адресов не могут быть использованы в качестве IP-адреса конечного узла сети, подключенной к Интернету? Для синтаксически правильных адресов определите их класс: А, В, С, D или Е.
- (A) 127.0.0.1 (E) 10.234.17.25 (I) 193.256.1.16
(B) 201.13.123.245 (F) 154.12.255.255 (J) 194.87.45.0
(C) 226.4.37.105 (G) 13.13.13.13. (K) 195.34.116.255
(D) 103.24.254.0 (H) 204.0.3.1. (L) 1161.23.45.395
15. Пусть IP-адрес некоторого узла подсети равен 198.65.12.67, а значение маски для этой подсети — 255.255.255.240. Определите номер подсети. Какое максимальное число узлов может быть в этой подсети?
16. Пусть поставщик услуг Интернета имеет в своем распоряжении адрес сети класса В. Для адресации узлов своей собственной сети он использует 254 адреса. Определите максимально возможное число абонентов этого поставщика услуг, если размеры требуемых для них сетей соответствуют классу С? Какая маска должна быть установлена на маршрутизаторе поставщика услуг, соединяющем его сеть с сетями абонентов?
17. Какое максимальное количество подсетей теоретически возможно организовать, если в вашем распоряжении имеется сеть класса С? Какое значение должна при этом иметь маска?
18. Почему даже в тех случаях, когда используются маски, в IP-пакете маска не передается?
19. Какие преимущества дает технология CIDR? Что мешает ее широкому внедрению?
20. Имеется ли связь между длиной префикса пула IP-адресов и числом адресов, входящих в этот пул?
21. Почему в записи о маршрутизаторе по умолчанию в качестве адреса сети назначения указывается 0.0.0.0 с маской 0.0.0.0?
22. Отличается ли обработка поля MAC-адреса кадра маршрутизатором и коммутатором?
23. Сравните функции маршрутизаторов, которые поддерживают маршрутизацию от источника, с функциями маршрутизаторов, поддерживающих протоколы адаптивной маршрутизации.
24. Какие метрики расстояния могут быть использованы в алгоритмах сбора маршрутной информации?
Сравните интенсивность широковещательного трафика, порождаемого протоколами RIP и OSPF.
25. Какие элементы сети могут выполнять фрагментацию?
- Только компьютеры (А).
 - Только маршрутизаторы (В).
 - Компьютеры, маршрутизаторы, мосты, коммутаторы (С).
 - Компьютеры и маршрутизаторы (Д).

26. Что произойдет, если при передаче пакета он был фрагментирован и один из фрагментов не дошел до узла назначения после истечения тайм-аута?
- Модуль IP узла- отправителя повторит передачу недошедшего фрагмента (A).
 - Модуль IP узла- отправителя повторит передачу всего пакета, в состав которого входил недошедший фрагмент (B).
 - Модуль IP узла- получателя отбросит все полученные фрагменты пакета, в котором потерялся один фрагмент; модуль IP узла- отправителя не будет предпринимать никаких действий по повторной передаче пакета данного пакета (C).
27. Какие особенности протоколов сетевого уровня стека Novell ограничивают их использование на глобальных линиях?
28. При образовании сетевого адреса в протоколе IPX в качестве номера узла используется MAC-адрес сетевого адаптера этого узла, а в протоколе IP номер узла назначается администратором произвольно. Какой, по вашему мнению, вариант является более эффективным и почему?
29. Каким образом должен быть сконфигурирован маршрутизатор, чтобы он предотвращал «широковещательный шторм»?
30. За счет чего коммутаторы третьего уровня ускоряют процесс маршрутизации?

Часть V

Глобальные сети

ГЛАВА 15 Качество обслуживания в пакетных сетях

Высокая стоимость протяженных каналов передачи данных и сложность простого повышения скорости передачи данных за счет прокладки дополнительных волоконно-оптических жил обуславливает чрезвычайно экономное отношение к пропускной способности канала в глобальных сетях. Для нормальной работы приложений в таких условиях требуется применение методов обеспечения качества обслуживания (Quality of Service, QoS). Другим мощным стимулом внедрения механизмов QoS является стремление к передаче по пакетным сетям всех типов информации, в том числе и мультимедийной — голоса, изображений, видеоданных. Трафик таких приложений отличается чувствительностью к задержкам — одному из параметров, который регулируется с помощью механизмов QoS. Поэтому в большинстве технологий, специально разработанных для глобальных сетей передачи данных — Frame Relay, ATM, — механизмы QoS являются встроенными.

Основные понятия качества обслуживания

Основной движущей силой развития сети являются приложения. В ответ на постоянно растущие требования приложений к пропускной способности сети появляются новые высокоскоростные технологии. Перенос в компьютерные сети новых видов трафика, например IP-телефонии, аудио- и видеовещания, привел к появлению новых требований, связанных с обеспечением низкого уровня задержек пакетов, поддержкой групповой доставки пакетов и т. д.

Простое повышение пропускной способности сети уже больше не является гарантией того, что разнообразные приложения, работающие в сети, получат то обслуживание, которое им требуется. Экстенсивный путь развития, когда каналы и коммуникационное оборудование сети заменяется другим, на порядок более производительным, дает быстрый желаемый результат — приложениям обычно достается необходимая для их качественной работы доля пропускной способности, при этом задержки из-за ожидания пакетов в очередях (неотъемлемая особенность сетей этого типа) становятся неощутимыми. Однако с течением времени количество пользователей сети растет, изменяются сами приложения — как правило, они становятся более ресурсоемкими, и сеть начинает испытывать перегрузки.

Нужны новые механизмы качества обслуживания, учитывающие все многообразие требований, предъявляемых приложениями к сети, и предоставляющие каждому приложению тот уровень качества обслуживания, который ему необходим — то ли в силу объективных потребностей приложения, то ли в соответствии с соглашением о качестве обслуживания, заключенным между клиентом и оператором сети. Такие механизмы и работают в составе службы QoS, являющейся своего рода «распорядителем» пропускной способности каналов и производительности коммуникационных устройств, контролируя и регулируя потребление ресурсов сети трафиком отдельных пользователей и их групп.

Типы QoS

Понятие качества обслуживания в пакетных сетях является сугубо статистическим. В условиях, когда пакеты передаются конечными узлами в сеть в случайные моменты времени, очереди в коммуникационных устройствах тоже представляют собой случайные процессы, что приводит к тому, что мгновенная скорость потока данных и задержки пакетов также носят случайный характер. Поэтому все параметры, которыми измеряется качество обслуживания в пакетных сетях, являются статистическими. Как правило, это средние значения (математическое ожидание) и вариации (дисперсии) скорости информационного потока и задержек пакетов. Нет смысла говорить о задержке отдельного пакета или о скорости потока на очень маленьком промежутке времени, соизмеримом со временем передачи одного пакета. Какие-либо числовые оценки качества обслуживания могут быть на практике измерены путем усреднения соответствующих величин на каком-либо заранее оговоренном промежутке времени.

Типы QoS различаются по степени «строгости», то есть по тому, насколько твердо сервис QoS может гарантировать обеспечение определенных значений характеристик QoS — пропускной способности, задержек, вариаций задержек, уровня потерь пакетов и т. п.

Можно выделить три типа служб QoS,

□ Сервис с максимальными усилиями, который также можно назвать отсутствием QoS, обеспечивает взаимодействие конечных узлов без каких бы то ни было гарантий. Типичные представители таких услуг — классические сети Ethernet или IP, которые не делают никаких различий между пакетами отдельных пользователей и приложений и обрабатывают эти пакеты на основании принципа FIFO (первым пришел — первым обслужен).

□ Сервис с предпочтением (называемый также «мягким» сервисом QoS) — некоторые типы трафика обрабатываются лучше, чем остальные. Подразумеваются более быстрая обработка, в среднем больше пропускной способности и меньше потерь данных. Это статистическое предпочтение, а не численно выраженные гарантии. Точные значения параметров QoS, которые получат приложения в результате работы службы QoS этого типа, неизвестны и зависят от характеристик предлагаемого сети трафика. Например, если высокоприоритетный трафик предлагает сети в данный момент времени низкую интен-

сивность своих пакетов, то низкоприоритетный трафик может в это время получать весьма качественное обслуживание — значительную пропускную способность и низкие задержки. Однако при изменении ситуации, когда высокоприоритетный трафик начинает посыпать в сеть свои пакеты с высокой интенсивностью, низкоприоритетный трафик может вообще некоторое время не обслуживаться.

□ Гарантийный сервис (называемый также «жестким», или «истинным», сервисом QoS) дает статистические численные гарантии различным потокам трафика. Обычно такой вид QoS основан на предварительном резервировании сетевых ресурсов для каждого из потоков, получившим гарантии обслуживания. Трафик, которому выделили ресурсы, гарантированно имеет при прохождении через сеть те параметры пропускной способности или задержек, которые определены для него в числовом виде (конечно, если источники этого трафика не нарушают оговоренных для них условий и генерируют не больше пакетов, чем предполагалось). Службы такого типа способны, например, гарантировать приложению условленную пропускную способность, не уменьшающуюся ни при каких обстоятельствах, какой бы перегруженной сеть не становилась. Следует отметить, что гарантии носят статистический характер, то есть можно гарантировать некоторое числовое значение какого-либо параметра только с некоторой вероятностью, пусть очень высокой, но, естественно, не равной 1. Например, с вероятностью 0,999 можно утверждать, что задержка пакета не превысит 100 мс, следовательно, один пакет из 1000 может задержаться в сети и на большее время. Второй составляющей такого режима работы службы QoS является входной контроль потоков, которым даны гарантии. Действительно, соблюдать данные гарантии можно только в том случае, если интенсивности входящих в сеть потоков не превышают предельных заранее оговоренных значений. В противном случае поток будет потреблять больше ресурсов, чем ему выделялось, а значит, другим потокам достанется меньше, чем было запланировано, и гарантии для них соблюdenы не будут.

Эти три подхода к работе служб QoS не исключают, а дополняют друг друга. В результате их комбинирование позволяет учесть разнообразные требования приложений и различные условия работы сети.

Некоторым приложениям достаточно обслуживания с максимальными усилиями, в то время как другим обязательно жесткое соблюдение гарантий за счет предварительного резервирования.

Трафик приложений, для которых параметры качества обслуживания не абсолютно важны (то есть приложение, а точнее, его пользователь может смириться с некоторым ухудшением реaktivности сети), обрабатывается службой второго типа, которая дифференцирование распределяет оставшиеся после резервирования ресурсы между несколькими приоритетными классами трафика. Если одному из дифференцированных классов требуется на некоторое время большая пропускная способность, то его потребности удовлетворяются за счет трафика дифференцированных классов более низкого приоритета или трафика, обрабатываемого службой обслуживания с максимальными усилиями.

Трафик приложений, для которых обязательным условием является некоторый

гарантированный уровень пропускной способности и/или задержек (например, трафик видеоконференции или трафик, поступающий от измерительной системы реального времени), обслуживается службой QoS третьего типа, предоставляющей гарантированный сервис.

Соглашение об уровне обслуживания

Качество обслуживания может рассматриваться с двух позиций. Во-первых, с точки зрения потребителя транспортных услуг сети, когда характеристики качества обслуживания выступают как некоторые желательные условия, обеспечивающие нормальную работу приложений. В этом случае речь идет о требуемом качестве обслуживания. Во-вторых, с точки зрения поставщика услуг (или администратора корпоративной сети), для которого качество обслуживания — это фактические характеристики работы сети, наблюдаемые в результате ее мониторинга. Реже такие характеристики могут быть получены в результате аналитического исследования или имитационного моделирования сети.

Естественной основой нормального сотрудничества поставщика и потребителя является договор, который в данном случае называется соглашением об уровне обслуживания (Service Level Agreement, SLA). В этом договоре поставщик услуг и его клиент определяют следующие позиции.

□ Параметры качества обслуживания трафика, которые интересуют потребителя и которые согласен поддерживать поставщик (например, средняя пропускная способность, максимальные задержки и вариации задержек, максимальная интенсивность потерь данных, коэффициент готовности сервиса, максимальное время восстановления сервиса после отказа и т. д.).

а Методы измерения параметров качества обслуживания.

□ Определение платы за обслуживание. Система оплаты может быть достаточно сложной, особенно если соглашение предусматривает несколько уровней качества обслуживания, которые оплачиваются по разным тарифам.

□ Санкции за нарушение обязательств поставщика услуг по обеспечению надлежащего качества обслуживания, а также за отклонения параметров трафика пользователя от оговоренных значений. Эти санкции могут выражаться в виде штрафов либо в иной форме, например в форме предоставления сервиса в течение некоторого времени бесплатно или по пониженному тарифу.

□ Как и любой договор, соглашение SLA по взаимному согласию поставщика и клиента может включать большое количество различных дополнительных статей. Например, статью, оговаривающую условия перехода к более качественному обслуживанию или обслуживанию с разным уровнем качества в зависимости от дня недели или времени суток.

□ Соглашение может включать также правила кондиционирования пользовательского трафика, то есть обработки трафика, который выходит за оговоренные пределы, например трафика с большей средней интенсивностью на значительном промежутке времени. Эти правила могут включать формальные способы распознавания различных потоков пользовательского трафика (что нужно в

том случае, когда соглашение оговаривает дифференцированное обслуживание определенных потоков). Также правила кондиционирования могут определять условия отбрасывания или маркировки пакетов-нарушителей (помеченные пакеты будут отбрасываться сетью не всегда, а только в том случае, когда сетевые устройства испытывают перегрузки). В некоторых случаях сетевое оборудование выполняет сглаживание пульсаций трафика для улучшения качества работы приложения (например, работающего с голосом), а также для уменьшения задержек в транзитных узлах сети, так как более равномерно поступающие пакеты уменьшают перегрузку сетевых устройств.

Многие поставщики услуг предлагают своим клиентам типовые контракты SLA. В них не только определен перечень характеристик качества обслуживания, но даже и их конкретные числовые значения, например: «задержка пакетов, усредненная за месяц, не будет превышать 100 мс при передаче между любыми двумя узлами сети». Типовые контракты облегчают жизнь поставщикам услуг, так как для их реализации можно обойтись без средств гарантированного поддержания качества обслуживания. Требуется только поддерживать примерно постоянный уровень запаса пропускной способности и предлагать в SLA те значения параметров QoS, которые демонстрирует работающая сеть.

Соглашение SLA заключается не только между поставщиком услуги публичных сетей и корпоративными клиентами. Достаточно популярным в последнее время стало заключение подобных контрактов между информационным отделом предприятия — поставщиком транспортных услуг, и потребителями — функциональными отделами предприятия.

Требования разных типов приложений

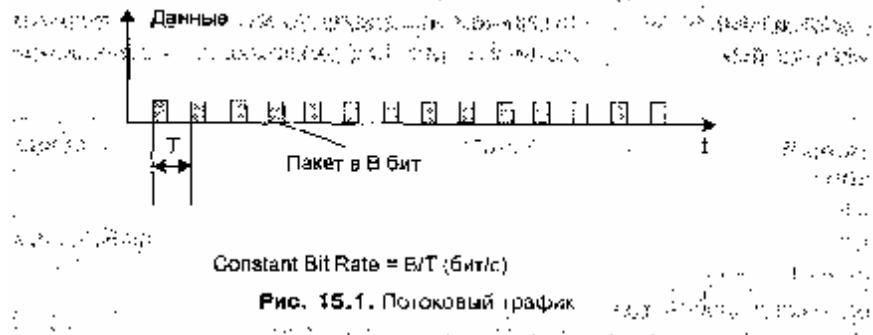
К настоящему времени проделана большая работа по классификации трафика различных приложений. В качестве основных критериев классификации были приняты три характеристики трафика:

- относительная предсказуемость скорости передачи данных;
- чувствительность трафика к задержкам пакетов;
- чувствительность трафика к потерям и искажениям пакетов.

Предсказуемость скорости передачи данных

В отношении предсказуемости скорости трафика приложения делятся на два больших класса: потоковый трафик (stream) и пульсирующий (burst).

Приложения с потоковым трафиком порождают равномерный поток данных, который поступает в сеть с постоянной битовой скоростью (Constant Bit Rate, CBR). При пакетном методе коммутации трафик таких приложений выглядит как последовательность пакетов одинакового размера, равного B бит, следующих друг за другом через один и тот же интервал времени T (рис. 15.1).



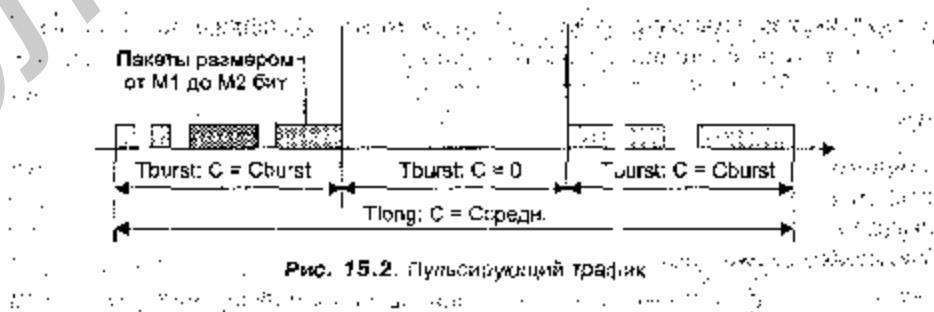
Средняя скорость потокового трафика CBR может быть вычислена путем усреднения на одном периоде:

$$\text{CBR} - \text{B}/\text{T} \text{ (бит/с)}.$$

Здесь В — размер одного пакета в битах, а Т — период повторения пакетов в секундах.

В общем случае средняя скорость потокового трафика меньше номинальной максимальной битовой скорости протокола, с помощью которого передаются данные, так как между пакетами существуют паузы. Как было показано в главе 7, максимальная скорость передачи данных с помощью протокола Ethernet составляет 9,76 Мбит/с (случай кадров максимальной длины), что меньше номинальной скорости в 10 Мбит/с.

Приложения с пульсирующим трафиком отличаются высокой степенью непредсказуемости, когда периоды молчания сменяются пульсацией, в течение которой пакеты «плотно» следуют друг за другом. В результате трафик характеризуется переменной битовой скоростью (Variable Bit Rate, VBR), что иллюстрирует рис. 15.2. Так, при работе приложений файлового сервиса интенсивность трафика, требуемая приложению, может увеличиваться от нуля, когда файлы не передаются, до бесконечности, когда после передачи запроса с координатами файла приложению требуется как можно более быстрая передача данных файла. (Понятно, что реальная скорость передачи ограничена возможностями сети.) Строго говоря, любые приложения генерируют пульсирующий трафик, в том числе и потоковые.



Степень пульсации часто характеризуют коэффициенты пульсации. Коэффициент пульсации равен отношению средней скорости на периоде пульсации к средней скорости, измеренной на длительном интервале времени, включающем большое количество периодов пульсации. На рис. 15.2 показано три периода

измерений, причем пульсации наблюдаются на первом и третьем периодах, а второй период является периодом молчания. Для упрощения расчетов принято, что средние скорости на первом и третьем периодах равны между собой и равны C_{burst} . В результате для приведенного примера коэффициент пульсации оказывается примерно равным 1,5, так как если на каждом периоде пульсации было передано B бит, то средняя скорость на периоде пульсации равна B/T_{burst} , а на всем периоде измерений — $2B/3T_{burst}$, так что отношение скоростей равно $3/2$, или 1,5.

Практически любой трафик, даже трафик потоковых приложений, имеет ненулевой коэффициент пульсации. Просто коэффициент пульсации у потокового и пульсирующего трафиков отличается качественно. У приложений с пульсирующим трафиком он обычно находится в пределах от 2:1 до 100:1, а у потоковых приложений существенно меньше. В локальных сетях коэффициент пульсации обычно выше, чем в глобальных, так как на магистралях глобальных сетей трафик представляет собой сумму трафиков многих источников, так что отдельные пульсации по закону больших чисел складываются.

Чувствительность трафика к задержкам пакетов

Другой критерий классификации приложений по типу трафика — их чувствительность к задержкам пакетов. Далее перечислены основные типы приложений в порядке повышения чувствительности к задержкам пакетов.

- Асинхронные приложения. Практически нет ограничений па время задержки («эластичный» трафик). Пример такого приложения — электронная почта.
- Синхронные приложения. Эти приложения чувствительны к задержкам, но допускают их.
- Интерактивные приложения. Задержки могут быть замечены пользователями, но они не сказываются негативно на функциональности приложений. Пример — текстовый редактор, работающий с удаленным файлом.
- Изохронные приложения. Имеется порог чувствительности к задержкам, при превышении которого резко снижается функциональность приложения. Пример — передача голоса, когда при превышении порога задержек в 100-150 мс резко снижается качество воспроизведенного голоса.
- Сверхчувствительные к задержкам приложения. Задержка доставки данных сводит к нулю функциональность. Пример — приложения, управляющие техническим объектом в реальном времени. При запаздывании управляющего сигнала на объекте может произойти авария.

Вообще говоря, интерактивность приложения всегда повышает его чувствительность к задержкам. Например, широковещательное распространение аудиоинформации может выдерживать значительные задержки передачи пакетов, а интерактивный телефонный разговор их не терпит.

Наряду с приведенной выше классификацией, тонко дифференциющей чувствительность приложений к задержкам и их вариациям, существует и более грубое деление приложений по этому же признаку на два класса — асинхронные и синхронные приложения. К асинхронным относят те приложения, кото-

рые не чувствительны к задержкам передачи данных в очень широком диапазоне, вплоть до нескольких секунд, а все остальные приложения, на функциональность которых задержки влияют существенно, относят к синхронным приложениям. К синхронным приложениям в этом широком смысле относятся такие типы приложений приведенной выше тонкой классификации, как изохронные и сверхчувствительные. Интерактивные приложения могут относиться как к асинхронным (например, текстовый редактор), так и к синхронным (например, видеоконференция).

Чувствительность трафика к потерям и искажениям пакетов

И, наконец, последним критерием классификации приложений является их чувствительность к потерям пакетов. Здесь обычно делят приложения на две группы.

- Чувствительные к потерям приложения. Практически все приложения, передающие алфавитно-цифровые данные (к которым относятся текстовые документы, коды программ, числовые массивы и т. п.), обладают высокой чувствительностью к потере отдельных, даже небольших, фрагментов данных. Такие потери часто ведут к полному обесцениванию остальной, успешно принятой информации. Например, отсутствие хотя бы одного байта в коде программы делает ее совершенно неработоспособной. Все традиционные сетевые приложения (файловый сервис, сервис баз данных, электронная почта и т. д.) относятся к этому типу приложений.
- Устойчивые к потерям приложения. К этому типу относятся многие приложения, передающие трафик, несущий информацию об инерционных физических процессах. Устойчивость к потерям объясняется тем, что небольшое количество отсутствующих данных можно определить на основе принятых. Так, при потере одного пакета, несущего несколько последовательных замеров голоса, отсутствующие замеры при воспроизведении голоса могут быть заменены аппроксимацией на основе соседних значений. К такому типу приложений относится большая часть приложений, работающих с мультимедийным трафиком (аудио- и видеоприложения). Однако устойчивость к потерям имеет свои пределы, поэтому процент потерянных пакетов не может быть большим, (например, не более 1 %). Можно отметить также, что не любой мультимедийный трафик так устойчив к потерям данных, например, компрессированный голос или видеоизображение очень чувствительны к потерям, поэтому относятся к первому типу приложений.

Вообще говоря, между значениями трех характеристик качества обслуживания нет строгой взаимосвязи. То есть приложение с равномерным потоком может быть как асинхронным, так и синхронным, а, например, синхронное приложение может быть как чувствительным, так и нечувствительным к потерям пакетов. Однако практика показывает, что из всего многообразия возможных сочетаний значений этих трех характеристик есть несколько таких, которые охватывают большинство используемых сегодня приложений.

Например, следующее сочетание характеристик приложения «порождаемый трафик — равномерный поток, изохронное, устойчивое к потерям» соответствует таким популярным приложениям, как IP-телефония, поддержка видеоконференций, аудиовещание через Интернет. Существуют и такие сочетания характеристик, для которых трудно привести пример приложения, например: «порождаемый трафик — равномерный поток, асинхронное, чувствительное к потерям».

Устойчивых сочетаний характеристик, порождающих определенный класс приложений, существует не так уж много. Так, при стандартизации технологии ATM были определены 4 класса приложений: A, B, C и D. Кроме того, для всех приложений, которые не попадали ни в один из этих классов, был определен класс X, в котором сочетание характеристик приложения может быть произвольным.

Приведенная классификация приложений лежит в основе типовых требований к параметрам и механизмам качества обслуживания в современных сетях.

Параметры качества обслуживания

Трем критериям классификации приложений (предсказуемость скорости передачи данных, чувствительность к задержкам и чувствительность к потерям и искажениям) соответствуют три группы параметров, используемых при определении и задании требуемого качества обслуживания.

- Параметры пропускной способности. К таким параметрам относятся средняя, максимальная (пиковая) и минимальная скорости передачи данных.
- Параметры задержек. Используется средняя и максимальная величины задержек, а также среднее и максимальное значения вариаций задержек, то есть отклонений межпакетных интервалов в прибывающем трафике по сравнению с исходящим.
- Параметры надежности передачи. Используется процент потерянных пакетов, а также процент искаженных пакетов.

При определении всех этих параметров важно, на каком периоде измеряется данный параметр. Чем меньше этот период, тем более жесткими являются соответствующие требования качества обслуживания и тем труднее для сети их выдержать. Поэтому поставщики услуг IP-сетей, которые испытывают сложности с обеспечением QoS, предпочтут говорить в соглашениях SLA о среднемесячных характеристиках, в то время как поставщики услуг сетей frame relay и ATM, располагающие мощными средствами QoS, способны гарантировать параметры, усредненные на периоде в несколько секунд.

Параметры качества обслуживания могут быть отнесены к предлагаемому трафику, порождаемому приложениями пользователя. Эти же параметры могут характеризовать возможности сети по обслуживанию этого трафика. Пусть, например, у пользователя имеется приложение, которое порождает равномерный поток с постоянной скоростью N. При заключении соглашения SLA с поставщиком услуг пользователь берет на себя обязательства, что предлагаемый тра-

фик приложения не будет превышать максимальную скорость N . Поставщик, в свою очередь, гарантирует со стороны сети, что минимальная величина пропускной способности, предоставляемой этому приложению, будет не меньше JV . Это необходимо для обеспечения приемлемого качества обслуживания трафика данного приложения.

Для приложений с пульсирующим трафиком качество обслуживания лучше всего характеризуется средней скоростью и максимальной скоростью, которая требуется в период пульсации. Обычно при этом оговаривается либо максимальное время пульсации, в течение которого приложение передает данные с максимальной скоростью, либо максимальный объем данных, который можно передать в виде пульсации. Часто используется также вариант с заданием максимальной и минимальной границ скорости. В этом случае приложению гарантируется пропускная способность на уровне минимальной границы, достаточная для его удовлетворительного функционирования, а само приложение обязуется не направлять в сеть трафик со скоростью, превышающей максимальную границу.

Параметры качества обслуживания обычно оговариваются в соглашении об уровне обслуживания SLA между пользователем сети и поставщиком услуг. После заключения соглашения пользователь и поставщик услуг должны соответствующим образом настроить свои программные и аппаратные средства, чтобы они отрабатывали оговоренные параметры. Некоторые технологии позволяют автоматизировать процесс взаимного согласования параметров QoS между оборудованием пользователя и оборудованием поставщика. Например, в технологии ATM при установлении соединения эти параметры согласуются с помощью процедуры, называемой трафик-контрактом.

Выводы

- В зависимости от строгости соблюдения гарантий обеспечения определенных параметров обслуживания — пропускной способности, задержек, вариаций задержек, уровня потерь пакетов и т. п. — различают следующие типы QoS: обслуживание по мере возможности (best effort), обслуживание с предпочтением одного потока (мягкий сервис QoS) и гарантированное обслуживание (жесткий сервис QoS).
- Все множество приложений может быть разделено на классы в зависимости от требований, предъявляемых ими к качеству обслуживания. В основу классификации положены три основных характеристики трафика, порождаемого приложениями: относительная предсказуемость скорости передачи данных, чувствительность трафика к задержкам пакетов, чувствительность трафика к потерям и искажениям пакетов.
- В соглашении об уровне обслуживания (Service Level Agreement, SLA) поставщик услуг и его клиент определяют: параметры качества обслуживания трафика, которые интересуют потребителя и которые согласен поддерживать поставщик, методы измерения этих параметров, финансовые обязательства обеих сторон, правила «кондиционирования» пользовательского трафика и др.

-
- При определении и задании требуемого качества обслуживания в SLA используются: параметры пропускной способности (средняя, максимальная и минимальная скорости передачи данных), параметры задержек (средняя и максимальная величины задержек, а также среднее и максимальное значения вариаций задержек), параметры надежности передачи (процент потерянных пакетов, а также процент искаженных пакетов).

Служба QoS

Модель службы QoS

Сеть — это распределенная среда, состоящая из большого количества устройств для поддержки различных технологий и протоколов. Поэтому достаточно сложно заставить ее соблюдать единые требования по качественному обслуживанию различных видов трафика на всем протяжении составного пути от одного конечного узла до другого, то есть «из конца в конец» (end-to-end). Даже обеспечение простой «связности» сети «из конца в конец» для составной неоднородной сети представляет собой непростую задачу, которая и является основной функцией классических средств стеков протоколов. Задача же продвижения пакетов в такой сети с заданными параметрами качества обслуживания существенно сложнее первой. Особенно если в сети существуют многочисленные потоки данных с трудно совместимыми характеристиками, например пульсирующий файловый трафик и синхронный голосовой.

Для решения поставленных задач в сети необходима служба QoS. Эта служба имеет распределенный характер, так как ее элементы должны присутствовать во всех сетевых устройствах, продвигающих пакеты: коммутаторах, маршрутизаторах, серверах доступа. С другой стороны, работу отдельных сетевых устройств по обеспечению поддержки QoS нужно скоординировать, чтобы качество обслуживания было однородным вдоль всего пути, по которому следуют пакеты потока. Поэтому служба QoS должна включать также элементы централизованного управления, с помощью которых администратор сети может согласованно конфигурировать механизмы QoS в отдельных устройствах сети.

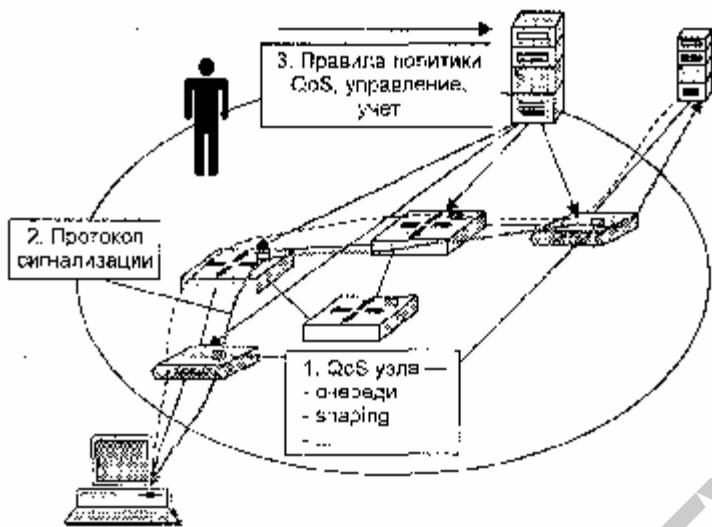


Рис. 15.3. Базовая архитектура службы QoS

Базовая архитектура службы QoS включает элементы трех основных типов, представленных на рис. 15.3:

- средства QoS узла, выполняющие обработку поступающего в узел трафика в соответствии с требованиями качества обслуживания;
- протоколы QoS-сигнализации для координации работы сетевых элементов по поддержке качества обслуживания «из конца в конец»;
- а централизованные функции политики, управления и учета QoS, позволяющие администраторам сети целенаправленно воздействовать на сетевые элементы для разделения ресурсов сети между различными видами трафика с требуемым уровнем QoS.

Средства QoS узла

Средства QoS узла являются основным исполнительным механизмом службы QoS, так как именно они непосредственно влияют на процесс продвижения пакетов между входными и выходными интерфейсами коммутаторов и маршрутизаторов и, следовательно, определяют вклад данного устройства в характеристики качества обслуживания сети.

Средства QoS узла могут включать механизмы двух типов:

- механизмы обслуживания очередей;
- механизмы кондиционирования трафика.

Механизмы обслуживания очередей являются необходимым элементом любого устройства, работающего по принципу коммутации пакетов. Они могут поддерживать различные алгоритмы обработки пакетов, попавших в очередь, от самых простых типа FIFO (первым пришел — первым обслужен) до весьма сложных, поддерживающих обработку нескольких классов потоков, например алгоритмов приоритетного, или взвешенного, обслуживания. По умолчанию в сетевых устройствах действует алгоритм очереди FIFO, но он достаточен толь-

ко для реализации обслуживания с максимальными усилиями, а для поддержки «истинных» сервисов QoS нужны более сложные механизмы.

Механизмы второго типа (кондиционирования трафика) могут реализовываться или не реализовываться в сетевом узле, поддерживающем QoS. Дело в том, что обеспечение нужного качества обслуживания всегда означает создание таких условий, когда скорость продвижения трафика потока согласуется со скоростью поступления этого трафика в узел сети. Очереди возникают в те периоды времени, когда скорость поступления трафика становится больше скорости его продвижения. Механизмы обслуживания очередей рассчитаны на работу как раз в периоды перегрузки и нужны для того, чтобы потоки как можно меньше страдали от существования таких периодов. Задержки от ожидания в очередях должны укладываться в параметры потока. Механизмы кондиционирования трафика решают задачу создания условий качественного обслуживания трафика другим способом — за счет уменьшения скорости поступления потока в данный узел настолько, чтобы она всегда оставалась меньше, чем скорость продвижения этого потока.

Механизм кондиционирования трафика обычно включает выполнение нескольких функций.

- Классификация трафика. Эта функция выделяет из общей последовательности пакетов, поступающих в устройство, пакеты одного потока, имеющего общие требования к качеству обслуживания. Классификация может выполняться на основе различных формальных признаков пакета — адресов источника и назначения, идентификаторов приложений, значения приоритета пакета, значения метки потока.
- Профилирование трафика на основе правил политики (policing). Для каждого входного потока имеется соответствующий ему набор параметров QoS. Этот набор часто называют профилем трафика. Профилирование трафика подразумевает проверку соответствия каждого входного потока параметрам его профиля. В случае нарушения параметров профиля (например, превышения длительности пульсации или средней скорости) происходит отбрасывание или маркировка пакетов этого потока. Отбрасывание некоторых пакетов снижает интенсивность потока и приводит его параметры в соответствие с указанными в профиле. Маркировка пакетов без отбрасывания нужна для того, чтобы пакеты все же были обслужены данным узлом (или последующими по потоку), но с качеством обслуживания, отличным от указанного в профиле (например, с увеличенным значением задержки). Для проверки соответствия входного трафика заданному профилю механизм кондиционирования выполняет измерения параметров потока. Для этого обычно используется один из известных алгоритмов, например алгоритм «дырявого ведра» (leaky bucket).
- Формирование трафика (shaping). Эта функция предназначена для придания прошедшему профилированию трафику нужной временной «формы». В основном с помощью данной функции стремятся сгладить пульсации трафика, чтобы выход пакетов из устройства был более равномерным, чем вход. Сглаживание пульсаций уменьшает очереди в сетевых устройствах, которые будут обрабатывать трафик далее по потоку. Его также целесообразно использовать для вос-

становления временных соотношений трафика приложений, работающих с равномерными потоками, например голосовых приложений.

Механизмы кондиционирования трафика могут поддерживаться каждым узлом сети или реализовываться только в пограничных устройствах. Последний вариант часто используют поставщики услуг, кондиционируя трафик своих клиентов.

Протоколы сигнализации QoS

Протоколы сигнализации QoS нужны механизмам QoS в отдельных узлах для обмена служебной информацией, способствующей координации усилий по обеспечению параметров качества обслуживания «из конца в конец», то есть на всем пути следования потока. Например, с помощью средств сигнализации приложение может зарезервировать себе вдоль всего маршрута следования требуемую среднюю пропускную способность (для сетей IP эту функцию поддерживает протокол RSVP).

Одно из примитивных средств сигнализации — маркировка пакета признаком, несущим информацию о требуемом для пакета качестве обслуживания. Наиболее часто в этих целях используется поле приоритета, которое существует во многих протоколах. Продвигаясь от устройства к устройству, пакет переносит вдоль пути следования свои требования к качеству обслуживания, правда, в достаточно обобщенной форме — так как поле приоритета имеет всего несколько возможных значений, то и качество обслуживания будет предоставляться дифференцирование по нескольким агрегированным потокам сети.

Инициировать работу протокола сигнализации может не только конечный узел, но и промежуточное устройство. Например, пограничный маршрутизатор поставщика услуг способен выполнить классификацию трафика и зарезервировать в сети поставщика данному потоку некоторую пропускную способность. В этом случае координация сетевых устройств будет происходить не на всем пути следования трафика, а только в пределах сети данного поставщика, что, конечно, снижает качество обслуживания трафика.

Централизованные функции политики, управления и учета QoS не являются необходимым элементом архитектуры службы QoS, но очень желательны в крупных сетях. Каждый пользователь и каждое приложение стремятся получить обслуживание с максимально высоким уровнем качества (например, пропускной способности). Следовательно, должны существовать средства, с помощью которых администратор мог бы выполнять функцию арбитра и задавать рациональный уровень качества обслуживания для отдельных пользователей и приложений или для их групп. Функции политики позволяют администратору создавать правила, по которым сетевые устройства могут формально, на основании набора признаков, распознавать отдельные типы трафика и применять к ним определенные функции QoS.

Правила политики могут конфигурироваться и храниться отдельно в каждом сетевом устройстве. Однако это требует от администратора крупной сети значительных усилий и наверняка порождает большое количество ошибок, что

приводит к несогласованной работе сетевых устройств. Ведь одно из них может выделить определенному потоку пропускную способность в 1 Мбит/с, а другое — в 1 кбит/с.

Для крупной сети необходимость централизации средств политики QoS достаточно очевидна. Единые правила политики, справедливые для всех устройств сети, хранятся на сервере политики (или нескольких серверах, реплицирующих базу данных политики, что позволяет обеспечить надежность и производительность системы поддержки политики). Администратор конфигурирует правила политики в одной точке, что снижает затраты его труда и количество ошибок. Затем с помощью специального протокола эти правила распространяются по всем сетевым устройствам, поддерживающим качество обслуживания, а сетевые устройства применяют политику для кондиционирования трафика и управления очередями в соответствии с нужными параметрами.

Службы QoS, в которых работают централизованные системы поддержки политики, называются службами QoS, основанными на политике (policy-based QoS). Правила политики полезны не только для управления QoS, но и для координации сетевых устройств при выполнении других функций, например функций защиты трафика. Поэтому централизованная система политики сети обычно базируется на общей справочной службе сети (Directory Service), традиционно хранящей все учетные данные о пользователях (имя, пароль). В последнее время ее назначение расширено для хранения самых разнообразных данных о сети, в том числе и данных о политике QoS, политике безопасности и т. п.

Описанной модели службы QoS соответствует большинство конкретных протоколов поддержки QoS, таких как RSVP, DiffServ сетей TCP/IP, протоколов служб CBR, VBR и ABR сетей ЛТМ.

Выводы

- Базовая архитектура службы QoS включает:
 - средства QoS узла, выполняющие обработку поступающего в узел трафика в соответствии с требованиями качества обслуживания;
 - протоколы QoS-сигнализации для координации работы сетевых элементов по поддержке качества обслуживания «из конца в конец»;
 - централизованные функции политики, управления и учета QoS, позволяющие администраторам сети целенаправленно воздействовать на сетевые элементы для разделения ресурсов сети между различными видами трафика с требуемым уровнем QoS.
- Средства QoS узла состоят из механизмов обслуживания очередей и механизмов кондиционирования трафика.
- Механизм кондиционирования трафика обычно включает: классификацию, профилирование и формирование трафика. Классификация может выполняться на основе различных формальных признаков потока (адрес назначения, метка потока и др.). Профилирование трафика подразумевает проверку соответствия каждого входного потока определенному для него набору параметров QoS. Пу-

тем формирования трафика сглаживаются его пульсации.

□ Протоколы сигнализации QoS нужны механизмам QoS в отдельных узлах для обмена служебной информацией, способствующей координации усилий по обеспечению параметров качества обслуживания «из конца в конец», то есть на всем пути следования потока.

Алгоритмы управления очередями

Основу средств поддержки QoS в сетевых элементах составляют очереди и алгоритмы обработки этих очередей. Эти механизмы используются в любом сетевом устройстве, которое работает на основе механизма коммутации пакетов, — в маршрутизаторе, в коммутаторе локальной или глобальной сети, в конечном узле (исключение составляют только повторители, которые пакетов не разли чают, а работают на уровне потоков битов).

Очередь нужна для обработки периодов временных перегрузок, когда сетевое устройство не может передавать пакеты на выходной интерфейс в том темпе, в котором они поступают для выполнения такого продвижения. Если причиной перегрузки является процессорный блок сетевого устройства, то для временно го хранения необработанных пакетов используется входная очередь, то есть очередь, связанная с входным интерфейсом. В том же случае, когда причина перегрузки заключается в ограниченной скорости выходного интерфейса (а она всегда ограничена скоростью поддерживаемого протокола), то пакеты времен но хранятся в выходной очереди.

Главным по степени влияния на возникновение очередей фактором является коэффициент нагрузки устройства (*utilization*) — отношение средней интен сивности входного трафика устройства к средней интенсивности продвижения пакетов на выходной интерфейс.

Если коэффициент нагрузки больше единицы, значит, интенсивность входного трафика постоянно выше, чем интенсивность продвижения пакетов на выход ной интерфейс. Поэтому очередь в устройстве существует всегда, и ее длина стремилась бы к бесконечности, если бы не конечный размер буфера, отведен ного под хранение пакетов, стоящих в очереди. Но и в том случае, когда коэффициент нагрузки меньше единицы, очередь тоже может существовать, более того, иметь достаточно значительную среднюю длину.

Это происходит тогда, когда имеется некоторая вариация интервалов поступле ния пакетов в устройство — чем больше эта вариация, тем больше средняя длина очереди. Вариация интервалов поступления пакетов является вторым по важности фактором, влияющим на поведение очередей, после коэффициента нагрузки. При пульсирующем характере многих типов трафика компьютерных сетей, когда коэффициент пульсаций равен 100:1 или более, очереди могут быть значительными. Если же эта вариация отсутствует, то есть пакеты прибы вают строго через определенные промежутки времени, как это происходит у трафика типа равномерного потока, то очередь при коэффициенте загрузки, меньшем 1, не возникает. Влияние пульсаций трафика наявление задержек обслуживания хорошо известно пользователям разделяемых сегментов Ethernet.

Даже при значениях коэффициента нагрузки сегмента 0,5 задержки доступа к сети бывают значительными, что заставляет использовать эти сети с коэффициентом нагрузки сегмента не более 0,3.

Последствием возникновения очередей является ухудшение качества обслуживания трафика. Образуются задержки передачи пакетов, носящие к тому же не-постоянный характер, а это значит, что растут вариации задержек. Кроме того, при длительных пульсациях очереди могут возрастать настолько, что пакеты не помещаются в буферную память сетевых устройств и теряются.

Оценка возможной длины очередей в сетевых устройствах была бы очень полезной, если бы помогала оценить параметры качества обслуживания при известных характеристиках трафика. Однако поведение очередей представляет собой вероятностный процесс, на который влияет много факторов, особенно при сложных алгоритмах обработки очередей, использующих приоритеты или взвешенное обслуживание разных потоков. Хотя для анализа очередей и разработана специальная область прикладной математики — теория массового обслуживания (queuing theory), — она может дать количественные оценки только для очень простых ситуаций, не соответствующих реальным условиям работы сетевых устройств. Поэтому служба QoS использует для поддержки гарантированного уровня QoS достаточно сложную модель, решающую задачу комплексно. Это делается с помощью следующих методов:

- за счет предварительного резервирования полосы пропускания для трафика с известными параметрами (например, значениями средней интенсивности и величины пульсации);
- принудительного профилирования входного трафика, что позволяет поддерживать коэффициент нагрузки устройства на нужном уровне;
- использования сложных алгоритмов управления очередями.

Чаще всего в маршрутизаторах и коммутаторах применяются следующие алгоритмы обработки очередей:

- традиционный алгоритм FIFO;
- приоритетное обслуживание (Priority Queuing), которое также называют «давляющим»;
- взвешенное обслуживание (Weighted Queuing, WQ).

Каждый алгоритм разрабатывался для решения определенных задач и специфическим образом воздействует на качество обслуживания различных типов трафика в сети. Возможно и комбинированное применение этих алгоритмов.

Традиционный алгоритм FIFO

Принцип традиционного алгоритма FIFO состоит в том, что в случае перегрузки пакеты помещаются в очередь, а при прекращении перегрузки передаются на выход в том порядке, в котором поступили, то есть «первым пришел — первым ушел» (First In — First Out, FIFO). Во всех устройствах с коммутацией пакетов — это алгоритм обработки очередей по умолчанию. Достоинством его является простота реализации и отсутствие потребности в конфигурировании.

Однако он имеет и коренной недостаток — невозможность дифференцированной обработки пакетов различных потоков. Все пакеты стоят в общей очереди на равных основаниях — и пакеты чувствительного к задержкам голосового трафика, и пакеты нечувствительного к задержкам, но очень интенсивного трафика резервного копирования, длительные пульсации которого могут надолго задержать голосовой пакет.

Очереди FIFO необходимы для нормальной работы сетевых устройств, но они недостаточны для поддержки дифференцированного качества обслуживания.

Приоритетное обслуживание

Алгоритмы приоритетной обработки очередей (Priority Queuing) очень популярны во многих областях вычислительной техники, в частности в операционных системах, когда одним приложениям нужно отдать предпочтение перед другими при обработке их в мультипрограммной смеси. Применяются эти алгоритмы также и для обеспечения преимущественной обработки одного класса трафика по сравнению с другими.

Механизм приоритетной обработки трафика основан на разделении всего сетевого трафика на небольшое количество классов, а затем назначении каждому классу некоторого числового признака — приоритета. Разделение на классы (классификация) может производиться разнообразными способами. В сущности, выбранный способ классификации не связан непосредственно с работой алгоритма обслуживания на основе приоритетов — главное, чтобы все поступающие в очередь пакеты имели некоторый приоритет. Классификация трафика представляет собой отдельную задачу. Пакеты могут разбиваться на приоритетные классы в соответствии с типом сетевого протокола — например, IP, IPX или DECnet (заметим, что такой способ подходит только для устройств, работающих на втором уровне), на основании адресов назначения и источника, идентификатора приложения и любых других комбинаций признаков, которые содержатся в пакетах. Правила классификации пакетов на приоритетные классы представляют собой часть политики управления сетью.

Точка классификации трафика может размещаться как в самом устройстве, так и вне его. Более масштабируемое решение — размещение функций классификации трафика в одном или нескольких устройствах, расположенных на границе сети (например, в коммутаторах корпоративной сети, к которым подключаются компьютеры пользователей, или во входных маршрутизаторах сети поставщика).

Подобный вариант классификации требует наличия в пакете специального поля, в котором можно запомнить назначенное значение приоритета, чтобы им могли воспользоваться остальные сетевые устройства, обрабатывающие трафик после классифицирующего устройства. Такое поле имеется в заголовке многих протоколов. В тех же случаях, когда специального поля приоритета в заголовке нет, разрабатывается дополнительный протокол, который вводит новый заголовок с таким полем. Так произошло с протоколом Ethernet — для него (и других

протоколов семейства 802) пришлось разработать спецификации IEEE 802.1Q/p, которые вводят дополнительное трехбитное поле приоритета. Приоритеты могут назначаться не только коммутатором или маршрутизатором, но и приложением в узле-отправителе. Необходимо также учитывать, что каждое сетевое устройство может не согласиться с назначенным в другой точке сети приоритетом данному пакету. В этом случае оно перепишет значение приоритета в соответствии с локальной политикой, хранящейся непосредственно в данном устройстве. Поэтому необходимо использовать централизованные способы применения политики в сети, обеспечивающие скоординированную работу устройств.

Независимо от выбранного способа классификации трафика, в сетевом устройстве имеется несколько очередей в соответствии с количеством приоритетных классов. Поступивший пакет в периоды перегрузок помещается в очередь, соответствующую его приоритетному классу. На рис. 15.4 приведен пример использования четырех приоритетных очередей: с высоким, средним, нормальнym и низким приоритетом. Приоритеты очередей имеют абсолютный характер предпочтения при обработке. Так, пока из более приоритетной очереди не будут выбраны все имеющиеся в ней пакеты, устройство не переходит к обработке следующей, менее приоритетной очереди. Поэтому пакеты, имеющие средний приоритет, всегда обрабатываются только тогда, когда очередь пакетов с высоким приоритетом пуста. Соответственно, пакеты с низким приоритетом обрабатываются только тогда, когда пусты все вышестоящие очереди: с высоким, средним и нормальным приоритетами.



Рис. 15.4. Приоритетное управление очередями

Конечный размер буферной памяти сетевого устройства требует, чтобы очереди ожидающих обслуживания пакетов имели некоторую предельную длину. Обычно по умолчанию всем приоритетным очередям отводятся одинаковые буфера, но многие устройства разрешают администратору назначать каждой очереди буфер индивидуального размера. Максимальная длина буфера определяет максимальное количество пакетов, которые могут храниться в очереди данного приоритета, так как если пакет поступает при заполненном буфере, то он просто отбрасывается.

Размер буфера определяется в идеальном случае таким образом, чтобы его хва-

тало с некоторым запасом для хранения очереди среднестатистической длины. Однако установить это значение достаточно сложно, так как оно изменяется в зависимости от нагрузки сети, поэтому требуется постоянное и длительное наблюдение за работой сети. В общем случае, чем выше значимость трафика для предприятия, чем больше его интенсивность и пульсации, тем больший размер буфера требуется этому трафику. В примере, приведенном на рис. 15.4, для трафика высшего и нормального приоритета выбраны большие значения буферов, а для остальных двух классов — меньшие. Мотивы принятого решения для высшего приоритета очевидны, а трафик нормального приоритета имеет, очевидно, высокую интенсивность и значительный коэффициент пульсаций.

Приоритетное обслуживание очередей обеспечивает высокое качество обслуживания для пакетов из самой приоритетной очереди. Если средняя интенсивность их поступления в устройство не превосходит пропускной способности выходного интерфейса (и производительности внутренних продвигающих блоков самого устройства), то пакеты высшего приоритета всегда получают ту пропускную способность, которая им нужна. Уровень задержек высокоприоритетных пакетов также минимален. Однако он не нулевой и зависит в основном от характеристик потока этих пакетов — чем выше пульсации потока и его интенсивность, тем выше уровень задержек. Трафик всех остальных приоритетных классов почти прозрачен для пакетов высшего приоритета. Слово «почти» относится к ситуации, когда высокоприоритетный пакет должен ждать завершения обслуживания низкоприоритетного пакета, так как его приход совпал по времени с началом продвижения низкоприоритетного пакета на выходной интерфейс.

Что же касается остальных классов приоритетов, то качество их обслуживания будет ниже, чем у пакетов самого высокого приоритета, причем уровень снижения заранее предсказать весьма трудно. Это снижение может быть очень существенным, если у высокоприоритетного трафика возникает иногда потребность в передаче данных с большой интенсивностью. Если коэффициент нагрузки выходного интерфейса, определяемый только трафиком высшего приоритетного класса, приближается в какой-то период времени к единице, то трафик остальных классов просто на это время замораживается.

Поэтому приоритетное обслуживание обычно применяется в том случае, когда в сети есть один класс трафика, чувствительный к задержкам, но его интенсивность невелика, а потому обслуживание этого класса не слишком ущемляет обслуживание остального трафика. Например, голосовой трафик чувствителен к задержкам, но его интенсивность обычно не превышает 8-16 кбит/с, так что при назначении ему высшего приоритета остальные классы трафика не будут страдать. Однако в сети могут наблюдаться и другие ситуации, например при существовании видеотрафика его приоритетного обслуживания, но имеющего гораздо более высокую интенсивность. Для таких случаев разработаны алгоритмы обслуживания очередей, дающие низкоприоритетному трафику некоторые гарантии, даже в периоды повышения интенсивности высокоприоритетного трафика.

Взвешенные настраиваемые очереди

Алгоритм взвешенных очередей (Weighted Queuing) разработан для того, чтобы можно было предоставить всем классам трафика определенный минимум пропускной способности или гарантировать некоторые требования к задержкам. Под весом данного класса понимается процент предоставляемой классу трафика пропускной способности от полной пропускной способности выходного интерфейса. Алгоритм, в котором вес классов трафика может назначаться администратором, называется настраиваемой очередью (Custom Queuing).

ПРИМЕЧАНИЕ

В случае когда веса назначаются автоматически, на основании некоторой адаптивной стратегии, реализуется так называемый алгоритм взвешенного справедливого обслуживания (Weighted Fair Queuing, WFQ), рассматриваемый в следующем разделе.

Как при взвешенном, так и при приоритетном обслуживании трафик делится на несколько классов, и для каждого класса ведется отдельная очередь пакетов. Но с каждой очередью связывается не ее приоритет, а процент пропускной способности выходного интерфейса, гарантированный данному классу трафика при перегрузках этого интерфейса. В примере, приведенном на рис. 15.5, устройство поддерживает 5 очередей для пяти классов трафика. Этим очередям при перегрузках выделяется соответственно 10%, 10%, 30%, 20% и 30% пропускной способности выходного интерфейса.

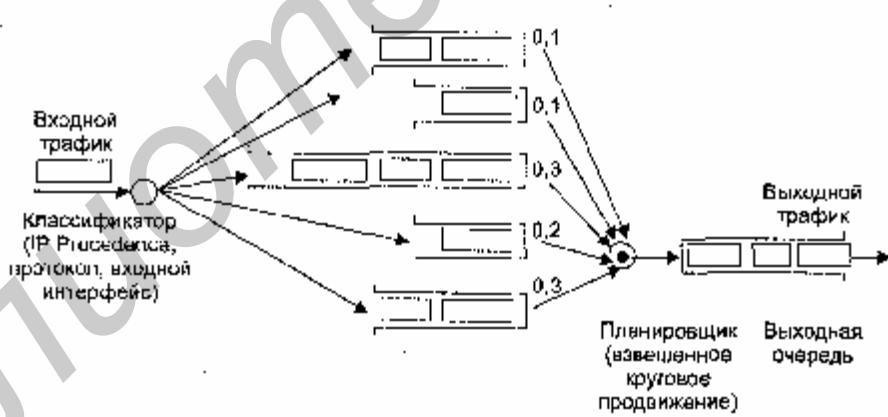


Рис. 15.5. Взвешенные настраиваемые очереди

Достигается поставленная цель тем, что очереди обслуживаются последовательно и циклически, и в каждом цикле обслуживания из каждой очереди выбирается такое число байтов, которое соответствует весу данной очереди. Например, если цикл просмотра очередей в рассматриваемом примере равен одной секунде, а скорость выходного интерфейса равна 100 Мбит/с, то при перегрузках в каждом цикле из первой очереди выбирается 10 Мбит данных, из второй — тоже 10 Мбит, из третьей — 30 Мбит, из четвертой — 20 Мбит, а из пятой — 30 Мбит. В результате каждому классу трафика достается гарантирован-

ный минимум пропускной способности, что во многих случаях является более желательным результатом, чем подавление низкоприоритетных классов высокоприоритетным.

Оценить уровень задержек сложнее, чем уровень пропускной способности. Для того чтобы он был соизмерим со временами передачи пакетов, цикл работы арбитра очереди должен быть, естественно, меньше, чем одна секунда — значение, выбранное для иллюстрации метода и упрощения расчетов. При таком времени цикла задержка может составить одну секунду и больше, так как арбитр возвращается к каждой очереди не чаще, чем раз в секунду, кроме того, в очереди может находиться более одного пакета. Время цикла в 100 мкс более подходит приведенному примеру. С одной стороны, оно обеспечивает обслуживание очереди каждого класса каждые 100 мкс. С другой стороны, этого времени достаточно, чтобы выбрать из каждой очереди в среднем по несколько пакетов (учитывая, что размер пакета в сетях Ethernet колеблется от 64 до 1518 байт).

На уровень задержек и вариации задержек пакетов для некоторого класса трафика при взвешенном обслуживании в значительной степени влияет коэффициент нагрузки трафика данного класса. В этом случае коэффициент подсчитывается как отношение интенсивности входного трафика класса к пропускной способности, выделенной этому классу в соответствии с его весом. Качественное поведение очереди и, соответственно, задержек здесь выглядит примерно так же, как и в случае очереди FIFO — чем меньше коэффициент нагрузки, тем меньше средняя длина очереди и тем меньше задержки.

Точные значения параметров QoS для алгоритма взвешенного обслуживания предсказать трудно. На них оказывают существенное влияние все динамически изменяющиеся параметры нагрузки сетевого устройства — интенсивность пакетов всех классов и вариации промежутков времени между прибытием пакетов. В общем случае, взвешенное обслуживание приводит к большим задержкам и их вариациям, чем приоритетное обслуживание для самого приоритетного класса, даже при значительном превышении выделенной пропускной способности над интенсивностью входного потока данного класса. Но для более низких приоритетных классов это соотношение может оказаться несправедливым, поэтому для создания более благоприятных условий обслуживания всех классов трафика взвешенное обслуживание часто бывает более приемлемым.

Как и для приоритетного обслуживания, администратор может назначать разным классам очередей буферы разных размеров. Уменьшение размеров буферов для очередей приведет к росту числа потерь пакетов при перегрузках, но зато уменьшается время ожидания для тех пакетов, которые не были отброшены и попали в очередь.

Взвешенное справедливое обслуживание

Взвешенное справедливое обслуживание (Weighted Fair Queuing, WFQ) — это комбинированный механизм, сочетающий приоритетное обслуживание очередей с взвешенным. Существует большое количество различных реализаций

WFQ производителями сетевого оборудования. Эти реализации отличаются способом назначения весов и поддержкой различных режимов работы, поэтому в каждом конкретном случае необходимо знакомиться с деталями конкретной реализации WFQ.

Наиболее распространенная схема предусматривает существование одной особой очереди, которая обслуживается по приоритетной схеме, то есть первой и до тех пор, пока все заявки из нее не будут выбраны. Эта очередь предназначена для системных сообщений, сообщений управления сетью и, возможно, пакетов наиболее критичных и требовательных приложений. Во всяком случае, предполагается, что ее трафик имеет невысокую интенсивность, так что значительная часть пропускной способности выходного интерфейса остается другим классам трафика.

Остальные очереди маршрутизатор просматривает последовательно, по алгоритму взвешенного обслуживания (рис. 15.6). Администратор может задать вес каждого класса трафика, то есть количество байтов из каждой очереди, выбираемых перед переходом к следующей.

Возможен и вариант работы по умолчанию, когда всем остальным классам трафика достаются равные доли пропускной способности выходного интерфейса из оставшейся от трафика приоритетного класса доли.

Такая же схема используется многими производителями коммуникационного оборудования в том случае, когда для пользовательского трафика предлагается сочетание приоритетного и взвешенного обслуживания. Существует один класс приоритетного трафика, пакеты которого всегда обслуживаются в первую очередь, так что обслуживание пакетов всех других классов начинается только тогда, когда приоритетная очередь пуста. Остальные очереди являются взвешенными и делят оставшееся время менеджера очередей в заданном процентном отношении.



Рис. 15.6. Взвешенное справедливое обслуживание

Механизмы профилирования и формирования трафика

Существует несколько популярных алгоритмов, которые рекомендуется при-

менять для профилирования и формирования трафика в стандартах на механизмы обеспечения QoS.

Алгоритм «дырявого ведра»

Алгоритм «дырявого ведра» (leaky bucket) разработан для профилирования пульсирующего трафика. Алгоритм позволяет проверить соблюдение трафиком оговоренных значений средней скорости и пульсации.

У алгоритма имеется несколько настраиваемых значений:

- T — период усреднения скорости;
- CIR (Committed Information Rate) — средняя скорость, которую трафик не должен превышать на периоде усреднения (скорость, согласованная с сетью);
- Vc — объем пульсации, соответствующий средней скорости CIR и периоду T :
$$Vc = CIR \times T;$$
- Ve — допустимое превышение объема пульсации.

В алгоритме предполагается, что трафик контролируется каждые T секунд. На каждом из этих интервалов времени трафик должен иметь среднюю скорость не более оговоренной скорости CIR . Скорость контролируется на основе подсчета объема данных, поступивших за период T . Если этот объем меньше или равен Vc , то фактическая скорость трафика была меньше Vc/T , то есть меньше CIR . Превышение объемом пульсации оговоренного значения Vc на величину Ve считается «мягким» нарушением — пакеты-нарушители должны помечаться некоторым признаком, но не отбрасываться. При превышении объемом пульсации величины $Vc + Ve$ кадры отбрасываются. Обычно действия, предпринимаемые в случае превышения пульсации на величину, меньшую $Vc + Ve$, и величину, большую $Vc + Ve$, являются настраиваемыми параметрами алгоритма, так что можно сконфигурировать его так, что кадры будут отбрасываться, даже если превышение находится в пределах уровня $Vc + Ve$, а можно и передавать кадры даже при превышении уровня $Vc + Ve$.

Алгоритм использует счетчик C поступивших от пользователя байтов. Каждые T секунд этот счетчик уменьшается на величину Vc (или же сбрасывается в 0, если значение счетчика меньше, чем Vc), что часто иллюстрируют «ведром», из которого дискретно, каждые T секунд, вытекает объем, равный минимальному из чисел Vc или C (рис. 15.7). Все кадры, данные которых не увеличили значение счетчика свыше порога Vc , пропускаются в сеть со значением признака $DE = 0$. Кадры, данные которых привели к значению счетчика, большему Vc , но меньшему $Vc + Ve$, также передаются в сеть, но с признаком $DE = 1$. И наконец, кадры, которые привели к значению счетчика, большему $Vc+Ve$, отбрасываются коммутатором.

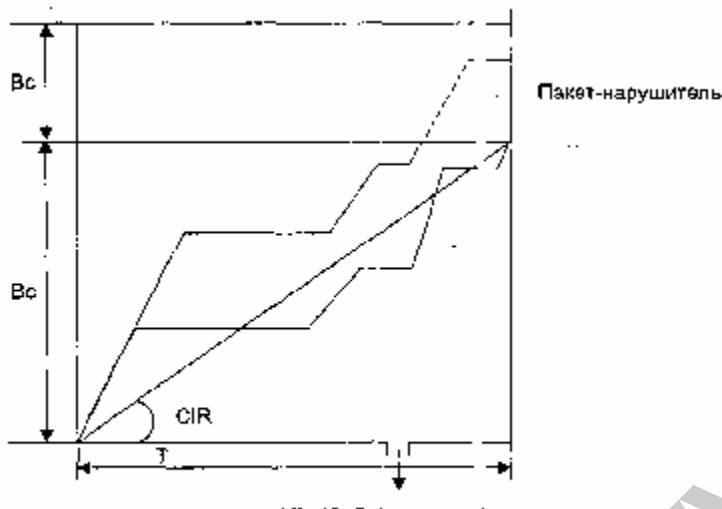


Рис. 15.7. Алгоритм «дырявого ведра»

Здесь описан только один из возможных вариантов алгоритма «дырявого ведра», применяемый для контроля трафика в сетях frame relay. Алгоритм допускает различные модификации, например большее количество порогов объема пульсации, использования вместо объема пульсации значения максимальной скорости и т. п.

Одна из модификаций алгоритма «дырявого ведра» под названием Generic Cell Rate Algorithm (GCRA) применяется в сетях ATM для контроля нескольких параметров: пиковой скорости, средней скорости, вариации интервала прибытия ячеек и объема пульсации.

Алгоритм «ведра токенов»

Алгоритм «ведра токенов» (Token Bucket) применяется как для профилирования, так и для формирования, то есть стягивания, трафика. Его цель — уменьшение неравномерности продвижения пакетов, когда из-за значительной пульсации они сбиваются в плотные группы. Алгоритм иллюстрирует рис. 15.8.

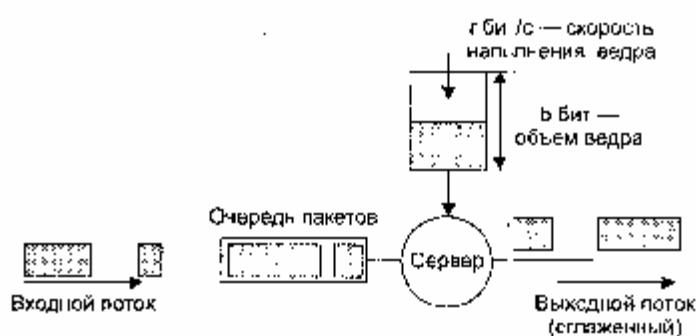


Рис. 15.8. Алгоритм «ведро токенов»

Под токеном в данном случае понимается некий абстрактный объект, носитель «порции» информации, используемый для построения модели обслуживания

трафика. Генератор токенов периодически направляет очередной токен в «ведро» с ограниченным объемом в b байт. Все токены имеют одинаковый объем m байт, а генерация токенов происходит с такой скоростью, что «ведро» заполняется со скоростью g байт в секунду. Скорость g является желательной средней скоростью для формируемого трафика. Пакеты поступают в систему и попадают в очередь объемом K байт.

Из очереди пакет продвигается сервером только в том случае, если к этому моменту «ведро» заполнено токенами до уровня не ниже M байт, где M — объем пакета. Если это условие выполняется и пакет продвигается на выход, то из «недра» удаляются токены общим объемом в M байт (с точностью до m байт). Если же ведро заполнено недостаточно, то пакет из очереди не выбирается, ожидая поступления нужного числа токенов.

Таким образом достигается «улучшение» трафика: даже если в результате пульсации в систему пришла большая пачка пакетов, из очереди пакеты выходят равномерно, в темпе, задаваемом генератором токенов. Поток токенов представляет собой идеальный трафик, к форме которого стараются привести входной трафик.

Планирование качества обслуживания

Для того чтобы потоки трафика обслуживались сетью с максимально возможной эффективностью, необходимо выполнять некоторую предварительную работу по планированию работы сети.

При планировании необходимо ответить на ряд важных вопросов.

- Какие пользовательские потоки и по каким маршрутам будут протекать через сеть?
- Каковы статистические характеристики этих потоков — средняя интенсивность трафика, коэффициент пульсаций?
- Каковы требования потоков к качеству обслуживания — уровню задержек, вероятности потерь?

Прежде всего нужно конкретизировать понятие максимально возможной эффективности. Обычно владелец сети стремится как можно более полно загрузить свои сетевые ресурсы — в первую очередь маршрутизаторы и каналы связи, — чтобы передавать как можно большие объемы данных. Работа пакетной сети может считаться эффективной, когда каждый ресурс ее загружен, но не перегружен. Как было показано выше, величина коэффициента загрузки ресурса оказывает определяющую роль на величину очереди к данному ресурсу, а значит, и на время задержки, возникающей из-за пребывания пакета в очереди.

В результате искусство планирования и управления сетью состоит в достижении двух целей. Во-первых, необходимо стремиться к улучшению качества обслуживания передаваемого трафика, то есть снижению задержек, уменьшению потерь и увеличению интенсивности потоков трафика. Во-вторых, при этом нужно максимально увеличить загрузку всех ресурсов сети для повышения объемов передаваемого трафика. Обе цели достигаются одними и теми же средствами — средствами борьбы с заторами (congestion) в сети. Однако невер-

но было бы считать, что оптимального использования ресурсов сети можно добиться только за счет рациональной настройки параметров ограничения интенсивности входных потоков и алгоритмов управления очередями. Пока без должного внимания остался такой важный параметр, как пути следования трафика через сеть.

Недостатки традиционных подходов

Большинство применяемых методов оптимизации работы сети до недавних пор были направлены на перераспределение ресурсов отдельного маршрутизатора между различными протекающими через него потоками. Именно эту задачу решают методы, объединенные под общим названием Quality of Service (QoS). В то же время такое мощное средство, как выбор путей прохождения трафика через сеть, традиционно использовалось во многих типах сетей очень примитивно. А ведь от путей следования трафика (при его фиксированной интенсивности) в первую очередь зависит загрузка маршрутизаторов и каналов, а значит, и эффективность использования сети. Далее мы рассмотрим недостатки подходов выбора пути, основанных на критерии кратчайшего расстояния до точки назначения, на примере IP-сетей как наиболее популярных в настоящее время. Известно, что все протоколы маршрутизации, в том числе дистанционно-векторные, такие как RIP, и состояния связей, такие как OSPF и IS-IS, выбирают для трафика, направленного в определенную сеть, кратчайший маршрут в соответствии с некоторой метрикой. Выбранный путь может быть более рациональным, например, если в расчет принимается номинальная пропускная способность каналов связи или вносимые ими задержки, или менее рациональным, если учитывается только количество промежуточных маршрутизаторов (хопов) между исходной и конечной сетями. Однако в любом случае выбирается единственный маршрут, даже если существует несколько альтернативных путей.

Классическим примером неэффективности такого подхода является так называемая «рыба» — сеть с топологией, приведенной на рис. 15.9. Несмотря на то что между маршрутизаторами A и E существует два пути — верхний, через маршрутизатор B, и нижний — через маршрутизаторы C и D, — весь трафик от маршрутизатора A к маршрутизатору E в соответствии с принципами маршрутизации, принятыми в IP-сетях, направляется по верхнему пути. Только потому, что нижний путь немного, на один хоп хуже, чем верхний, он игнорируется, хотя он мог бы работать «параллельно» с верхним путём.

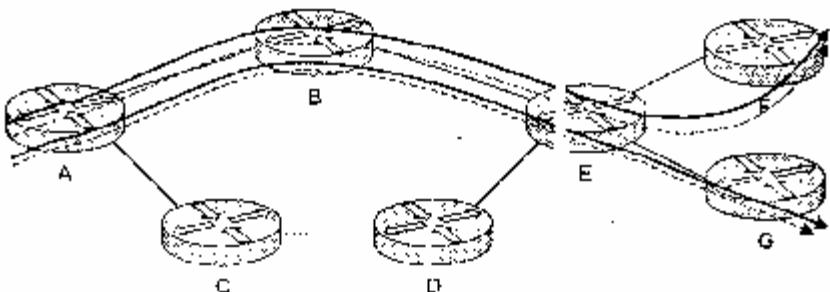


Рис. 15.9. Незэффективность кратчайших путей

Заметим, однако, что при наличии в сети нескольких альтернативных маршрутов равной стоимости (метрики) трафик делится между этими маршрутами и нагрузка на маршрутизаторы и каналы связи распределяется более сбалансировано. Но это слабый инструмент, который не работает в таком часто встречающемся случае, когда стоимость альтернативных маршрутов лишь незначительно хуже, чем у кратчайшего маршрута.

Еще один коренной недостаток традиционных методов маршрутизации трафика в IP-сетях заключается в том, что пути выбираются без учета текущей загрузки ресурсов сети. Даже если кратчайший путь уже перегружен, пакеты все равно посылаются по этому пути. Так, в сети, представленной на рис. 15.9, верхний путь будет продолжать использоваться даже тогда, когда его ресурсов перестанет хватать для обслуживания трафика от А к Е, а нижний путь будет простаивать, хотя, возможно, ресурсов маршрутизаторов В и С хватило бы для качественной передачи этого трафика. Налицо явная ущербность методов распределения ресурсов сети — одни ресурсы работают с перегрузкой, а другие не используются вовсе. Никакие методы QoS узлов эту проблему решить не могут, потому что они часто начинают решать задачу рационального распределения ресурсов маршрутизатора в заведомо «проигрышных» условиях, когда хорошее решение найти просто невозможно — узел уже перегружен, в то время как другие маршрутизаторы почти простаивают. Следовательно, нужны качественно иные механизмы.

Понятие инжиниринга трафика

Одним из мощных, но не используемых ранее в IP-сетях методов влияния на эффективное использование ресурсов сети является технология инжиниринга трафика (Traffic Engineering, TE).

В узком смысле, наиболее соответствующем названию, под ТЕ понимаются методы и механизмы, которые позволяют достичь сбалансированной загрузки всех ресурсов сети путем рационального выбора путей прохождения трафика через сеть.

Исходными данными для выбора путей являются, во-первых, характеристики передающей сети — ее топология, а также производительность составляющих ее маршрутизаторов и каналов связи (рис. 15.10), а во-вторых, сведения о нагрузке сети, то есть о потоках трафика, которые сеть должна передать между

своими пограничными маршрутизаторами (рис. 15.11).

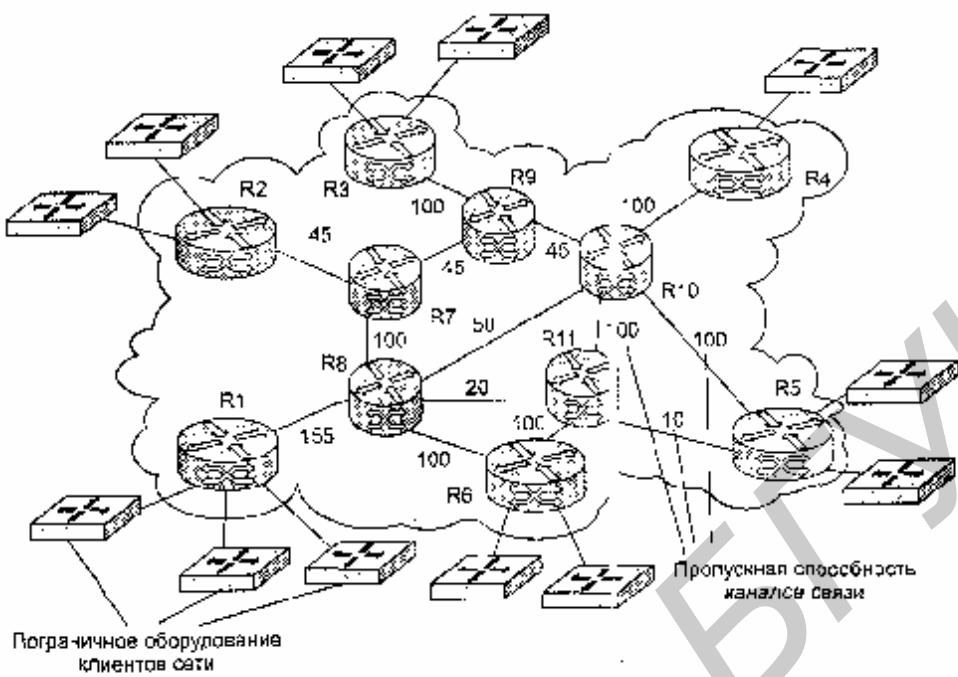


Рис. 15.10. Топология сети и производительность ее ресурсов

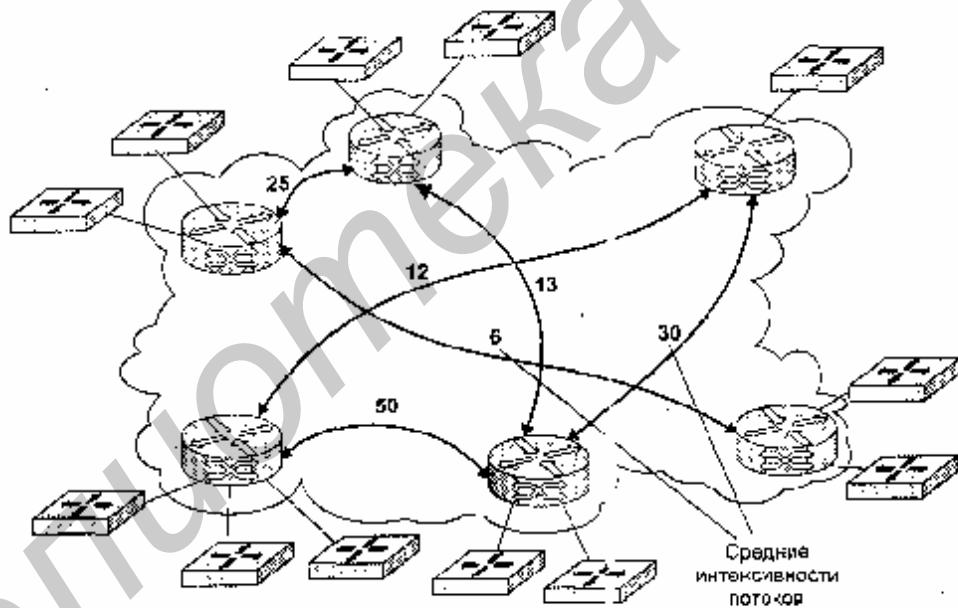


Рис. 15.11. Предложенная нагрузка.

Каждый поток характеризуется точкой входа в сеть, точкой выхода из сети, а также некоторыми параметрами трафика. Для более тонкой оптимизации сети можно привлекать и более детальное описание каждого потока, например величину возможной пульсации трафика или требования к качеству обслуживания — чувствительность к задержкам, вариации задержек и допустимый процент потерь пакетов. Однако поскольку оценить такого рода параметры трафика сложнее, чем среднюю интенсивность, а влияние их на работу сети менее значимо, для нахождения субоптимального распределения путей прохождения потоков через сеть, как правило, учитываются только их средние интенсивности.

Как видно из этого определения, потоки отдельных пользователей сети агрегированы, при этом одним потоком считается трафик пакетов, имеющих общие входную и выходную точки. Агрегированное задание потоков позволяет упростить задачу выбора путей, так как при индивидуальном рассмотрении каждого пользовательского потока, отличающегося, как уже было сказано, адресами конечных узлов (а не промежуточных пограничных маршрутизаторов), а также программных портов UDP/TCP, определяющих тип приложения в конечных узлах, промежуточные маршрутизаторы должны хранить слишком большие объемы информации, поскольку индивидуальных потоков может быть очень много. Необходимо, однако, подчеркнуть, что агрегирование отдельных потоков в один возможно только в том случае, когда все составляющие потоки предъявляют одни и те же требования к качеству обслуживания. В противном случае необходимо рассматривать несколько потоков, возможно агрегированных, поскольку у каждого агрегированного потока имеются единые требования к параметрам QoS.

Задача TE состоит в определении путей прохождения потоков трафика через сеть, то есть для каждого потока требуется найти точную последовательность промежуточных маршрутизаторов и их интерфейсов, находящихся на пути между входной и выходной точками потока. При этом пути должны быть такие, чтобы все ресурсы сети были нагружены как можно более сбалансировано. Это условие можно формализовать разными способами. Например, можно пытаться минимизировать максимальный коэффициент использования ресурса по всем ресурсам сети, пытаясь тем самым минимизировать максимально возможный ущерб для трафика. Именно так формулируется задача TE в документе RFC 2702 «Requirements for Traffic Engineering Over MPLS». В этом документе, содержащем общие рекомендации ГЕТР по решению задач TE, в качестве целевой функции оптимизации путей используется выражение:

$$\min (\max K_i).$$

Здесь K_i — коэффициент использования i -го ресурса.

Другим вариантом постановки задачи TE является поиск такого набора путей, при которых все значения коэффициентов использования ресурсов не превышают некоторый заданный порог K_{\max} . Такой подход проще в реализации, так как связан с перебором меньшего количества вариантов, а потому он чаще используется на практике. На рис. 15.12 показано одно из возможных решений поставленной задачи, гарантирующее, что максимальный коэффициент использования ресурсов не превысит 0,6.

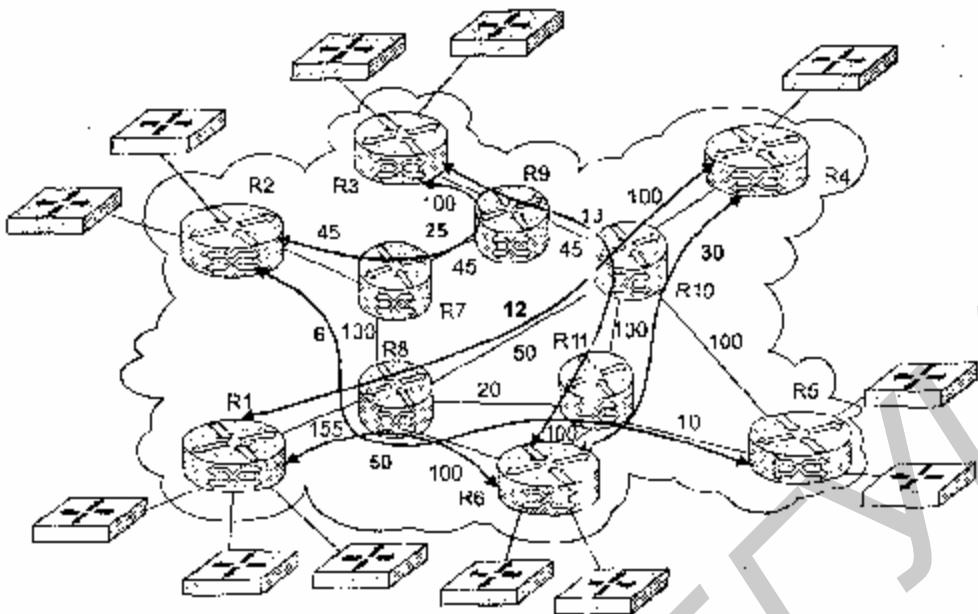


Рис. 15.12. Распределение нагрузки по сети – выбор путей следования трафика

Решение задачи инжиниринга трафика можно находить по-разному. Во-первых, можно искать его заблаговременно, в фоновом режиме. Для этого нужно знать исходные данные: топологию и производительность сети, а также входные и выходные точки потоков трафика и их среднюю интенсивность. После этого задачу рационального распределения путей следования трафика при фиксированных точках входа и выхода, а также заданном уровне максимального значения коэффициента использования ресурса можно передать некоторой программе, которая, например, с помощью направленного перебора вариантов найдет решение. Результатом работы программы будут точные маршруты для каждого потока с указанием всех промежуточных маршрутизаторов.

Во-вторых, можно решать задачу ТЕ в оперативном режиме, поручив ее самим маршрутизаторам. Для этого используются расширения протоколов маршрутизации, работающих на основе алгоритма состояния связей. Сегодня такие расширения стандартизованы для протоколов OSPF и IS-IS. Причина применения протоколов маршрутизации именно этого класса достаточно очевидна — эти протоколы, в отличие от дистанционно-векторных протоколов, к которым относится, например, RIP, дают маршрутизатору полную топологическую информацию о сети. Объявления этих протоколов содержат данные о маршрутизаторах и сетях, а также о физических связях между ними. Каждая связь характеризуется текущим состоянием работоспособности и метрикой, которой является величина, обратная пропускной способности канала. В традиционном варианте граф сети, ребра которого нагружены значениями метрик, используется маршрутизатором для выбора кратчайшего (с минимальной суммарной метрикой) пути до каждой из сетей. При этом из найденного пути в таблице маршрутизации запоминается только следующий хоп (IP-адрес ближайшего маршрутизатора), а остальные промежуточные хопы отбрасываются — так требует распределенный принцип продвижения пакетов, принятый в сетях IP, в соответ-

вии с которым каждый маршрутизатор принимает решение только об одном та-
ге маршрута.

Для решения задачи ТЕ в протоколы OSPF и TS-1S включены новые типы объ-
явлений, которые распространяют по сети информацию о номинальной и
.незарезервированной (свободной для потоков ТЕ) пропускной способности
каждой связи. Таким образом, ребра результирующего графа сети, который
создается в топологической базе каждого маршрутизатора, идентифицируются
этими двумя дополнительными параметрами.

Располагая таким графом, а также параметрами потоков, для которых нужно
найти пути ТЕ, маршрутизатор получает возможность найти рациональное ре-
шение, удовлетворяющее, например, одному из сформулированных выше огра-
ничений на коэффициенты использования ресурсов сети, обеспечив тем самым
сбалансированную загрузку сети. Для упрощения задачи оптимизации выбор
путей для некоторого набора потоков может осуществляться по очереди, при
этом в качестве ограничения выступает суммарная загрузка каждого ресурса
сети. Обычно считается, что внутренней производительности маршрутизатора
достаточно (в среднем) для обслуживания любого трафика, который способны
принять интерфейсы маршрутизатора. Поэтому ограничениями являются толь-
ко максимально допустимые значения коэффициентов загрузки каналов связи,
устанавливаемые индивидуально или же имеющие общее значение. Решение
задачи нахождения маршрута с учетом ограничений получило название Con-
strained-based Routing, а протокол OSPF с соответствующими расширениями —
Constrained SPF или CSPF.

Понятно, что поиск путей ТЕ по очереди снижает качество решения — при од-
новременном рассмотрении всех потоков можно найти более рациональный ва-
риант загрузки ресурсов. В примере, показанном в верхней части рис. 15.13, ог-
раничением является максимально допустимое значение коэффициента исполь-
зования ресурсов, равное 0,65. В варианте 1 решение было найдено при очеред-
ности рассмотрения потоков 1->2->3. Для первого потока был выбран путь А-
В-С, так как в этом случае он, с одной стороны, удовлетворяет ограничению
(все ресурсы вдоль пути — каналы А-В, А-С и соответствующие интерфейсы
маршрутизаторов оказываются загруженными на $50/155 = 0,32$), а с другой сто-
роны, обладает минимальной метрикой $65 + 65 = 130$. Для второго потока также
был выбран путь А-В-С, так как и в этом случае ограничение удовлетворяется
— результирующий коэффициент использования оказывается равным $50 +$
 $40/155 = 0,58$. Третий поток направляется по пути А-Д-Е-С и загружает ресурсы
каналов А-Д, Д-Е и Е-С на 0,3. Решение 1 можно назвать удовлетворительным,
так как коэффициент использования любого ресурса в сети не превышает 0,58.

Однако существует лучшее решение, представленное вариантом 2. Здесь по
верхнему пути А-В-С были направлены потоки 2 и 3, а поток 1 — по нижнему
А-Д-Н-С. Ресурсы верхнего пути оказываются загруженными на 0,45, а нижне-
го — на 0,5, то есть налицо более равномерная загрузка ресурсов, а максималь-
ный коэффициент использования по всем ресурсам сети не превышает 0,5. Этот
вариант может быть получен при одновременном рассмотрении всех трех пото-
ков с учетом ограничения $\min(\max K_j)$ или же при рассмотрении потоков по

очереди в последовательности 2→3→1.

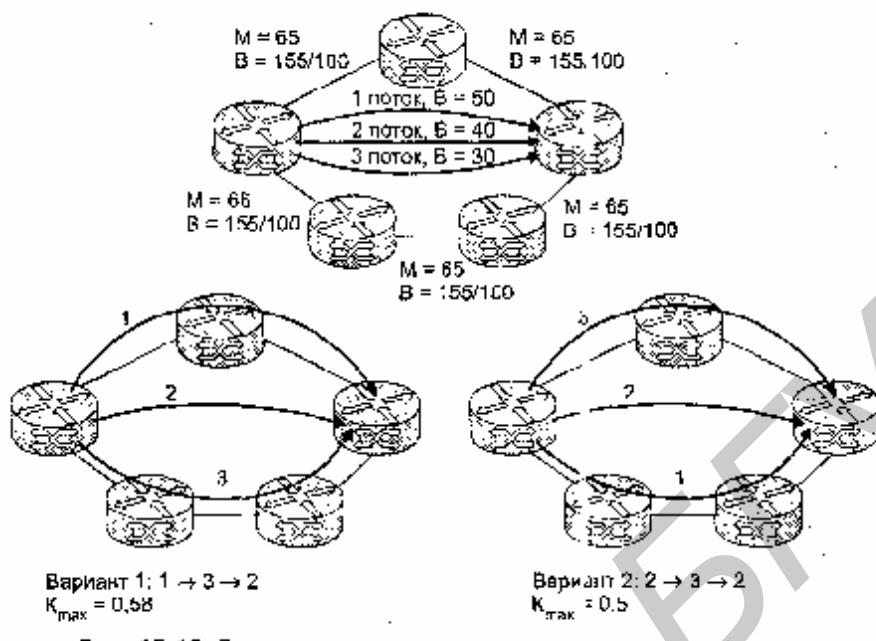


Рис. 15.13. Влияние порядка рассмотрения м: пропусков на результат

Тем не менее в производимом сегодня оборудовании применяется вариант MPLS TE с последовательным рассмотрением потоков. Он проще в реализации, ближе к стандартным для протоколов OSPF и IS-IS процедурам нахождения кратчайшего пути для одной сети назначения (если нет ограничений, то найденное решение для набора кратчайших путей не зависит от последовательности рассмотрения сетей, для которых производился поиск). Кроме того, при изменении ситуации — появлении новых потоков или изменении интенсивности существующих — задача часто не может быть решена иначе, как только для одного потока. Существует еще одна возможность улучшения решения — нахождение оптимального решения для набора потоков внешней по отношению к сети системой в автономном (off-line) режиме. Это может быть достаточно сложная система, которая включает подсистему имитационного моделирования, способную учесть не только средние интенсивности потоков, но и их пульсации и оценить не только загрузку ресурсов, но и результирующие параметры QoS — задержки, потери и т. п. После нахождения оптимального решения его можно модифицировать уже в режиме подключения (on-line) путем поочередного поиска путей.

В технологии TE информация о найденном рациональном пути используется полностью — запоминается не только первый хоп пути, как в основном режиме IP-маршрутизации, а все промежуточные узлы сети вместе с начальным и конечным, то есть имеет место маршрутизация от источника. Поэтому достаточно, чтобы поиском путей занимались только пограничные маршрутизаторы сети, а внутренние — лишь поставляли им информацию о текущем состоянии сети, которая необходима для принятия решений. Такой подход обладает некоторыми преимуществами по сравнению с распределенной моделью поиска

пути, лежащей в основе стандартных протоколов маршрутизации IP. Во-первых, он позволяет использовать «внешние» решения, когда пути находятся какой-либо системой оптимизации сети в автономном режиме, а потом устанавливаются в сети. Во-вторых, каждый из граничных маршрутизаторов сети может работать по собственной версии алгоритма поиска пути, в то время как при распределенном поиске необходимо во всех устройствах использовать идентичный алгоритм, что усложняет построение сети на оборудовании разных производителей. Ну и, в-третьих, такой подход разгружает внутренние LSR-маршрутизаторы от работы по поиску путей.

Однако мало найти решение — надо его реализовать. После того как пути определены, необходим механизм, который бы позволял направлять пакеты, относящиеся к определенному потоку трафика, именно по тем промежуточным маршрутизаторам, которые были выбраны. Эта задача для IP-сетей тоже не тривиальна, так как основной режим маршрутизации предлагает собственный единственный «кратчайший» маршрут, а режим маршрутизации от источника (Source Routing), позволяющий источнику пакета задавать точную последовательность промежуточных узлов вдоль пути, обладает несколькими очень серьезными ограничениями. Во-первых, сегодня он поддерживается в текущей версии 4 протокола ТР далеко не всеми производителями оборудования. Во-вторых, при маршрутизации от источника повышается степень избыточности служебной информации, так как каждый пакет переносит все адреса промежуточных маршрутизаторов, а их может быть не так уж и мало. В-третьих, этот режим чрезвычайно уязвим с точки зрения безопасности; именно поэтому администраторы при конфигурировании маршрутизаторов, как правило, его отключают.

Следовательно, для реализации в сети путей, найденных методом ТЕ, нужен специальный служебный протокол, который обычно называют протоколом сигнализации (терминология пришла из телефонных сетей, где с помощью таких протоколов устанавливаются соединения между абонентами). Сегодня в качестве такого протокола чаще всего в IP-сетях используется протокол резервирования ресурсов RSVP (Resource reSerVation Protocol), в сообщениях которого указан точный устанавливаемый путь — в виде последовательности IP-адресов интерфейсов промежуточных маршрутизаторов. Маршрутизаторы запоминают устанавливаемый в сети путь в виде специальных таблиц — таблиц коммутации. А для того чтобы пакеты продвигались не с помощью таблиц маршрутизации, а с помощью таблиц коммутации, используется специальная технология продвижения — MPLS (Multiprotocol Label Switching). Рассмотрение данной технологии выходит за рамки данной книги, можно только отметить, что у нее есть много общего с технологиями, опирающимися на технику виртуальных путей — такими как X.25, ATM и frame relay (они подробно рассматриваются в главе 17). Существенным для инженеринга трафика является тот факт, что все эти технологии позволяют передавать пакеты, принадлежащие определенному потоку, по заранее выбранному и установленному в сети пути. При этом специфика техники ТЕ заключается в том, что пути выбираются с целью соблюдения баланса загрузки ресурсов сети.

Соблюдение верхнего предела загрузки после решения задачи ТЕ гарантирует, что средняя длина и вариация очереди к каждому ресурсу будет находиться в определенных пределах, так что задержки пакетов и потери из-за нехватки места в буферной памяти также с определенной вероятностью не будут превышать некоторых граничных значений. Правда, точные значения этих параметров по одному только коэффициенту использования ресурса определить практически невозможно, хотя этот коэффициент и оказывает на них наиболее существенное влияние, есть и другие факторы, которые, хоть и в меньшей степени, но также влияют на результат — вариация интервалов поступления пакетов в коммуникационные устройства, вариация времен принятия решения о продвижении и т. п. Обычно точные значения задержек и процента потерь данных определяются в сети путем натурных измерений, и именно такие измеренные значения илагаются клиентам сети в соглашении об уровне обслуживания (в SLA). В условиях, когда через сеть проходит большое количество потоков данных многочисленных (сотен или даже тысяч) клиентов, добавка трафика еще одного или нескольких клиентов не изменит кардинально загрузку ресурсов, поэтому параметры качества обслуживания останутся приблизительно в тех же границах, что и прежде. Очевидно, что так будет только в том случае, когда пути следования трафика новых клиентов также выбираются методами ТЕ, так что ни один ресурс после добавления новых потоков не оказывается перегруженным.

Инжиниринг трафика различных классов

Нетрудно заметить, что задача инжиниринга трафика в предыдущем пункте решалась в достаточно упрощенной постановке — все потоки трафика считались равнозначными, потому что предъявляли одинаковые требования к качеству обслуживания. Пользователей сети удовлетворяло, что все потоки обслуживаются с заданной средней скоростью и в условиях, когда загрузка каждого ресурса не превышает определенного значения, например 0,6. Более реальной является ситуация, когда у каждого пользователя сети имеется несколько классов трафика, и эти классы отличаются требованиями к качеству обслуживания. Типичным примером является разделение всего трафика, по крайней мере, на два класса — чувствительного и не чувствительного к задержкам. В первый класс попадает, например, трафик IP-телефонии и других мультимедийных интерактивных приложений, а также трафик приложений управления технологическими объектами. Остальной трафик составляет второй класс. Если оператор сети не будет различать эти два класса в задачах ТЕ, то у него останется два варианта действий. В первом случае он может предлагать клиентам сети гарантированные пределы задержек и вариаций задержек пакетов, но держать ресурсы своей сети недогруженными, так чтобы коэффициент использования каждого ресурса не превышал 0,2-0,3. Тогда для пакетов обоих классов будут созданы условия для качественной в отношении задержек передачи, но такой подход трудно назвать рациональным. Если же оператор сети хочет хорошо загрузить

свои ресурсы, например, до уровня 0,6-0,7 (оставляя часть пропускной способности для служебного трафика), то он может предлагать соглашение SLA только для нечувствительного к задержкам трафика, гарантируя только среднюю скорость продвижения.

Остается еще одна возможность — различать классы трафика и решать задачу ТЕ с учетом их существования. Для этого расширения протоколов маршрутизации должны учитывать загрузку каждого ресурса отдельно для каждого класса трафика, так чтобы загрузка для чувствительного к задержкам трафика не превысила 0,2-0,3, а загрузка для остального трафика находилась в пределах 0,6-0,7. Если при этом чувствительный к задержкам трафик будет обслуживаться в единственной приоритетной очереди, а остальной трафик — по схеме кругового обслуживания, то для каждого класса уровень обслуживания в отношении задержек будет таким, какой и требуется (рис. 15.14).

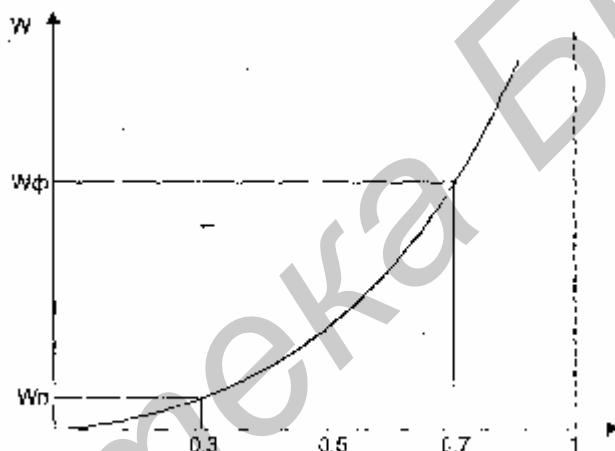


Рис. 15.14. Качество обслуживания при приоритизации чувствительного к задержкам трафика

Для достижения такого результата с каждым ресурсом должно быть связано два счетчика загрузки — один для приоритетного трафика и второй для остального (фонового) трафика. Когда решается задача ТЕ для нахождения пути приоритетного трафика по сети, то при проверке возможности прохождения пути через конкретный ресурс средняя интенсивность нового потока должна сравниваться со свободной интенсивностью ресурса для приоритетного трафика. Если снобд-пой интенсивности достаточно и новый поток будет проходить через данный интерфейс, то значение интенсивности нового потока необходимо вычесть как из счетчика загрузки приоритетного трафика, так и из счетчика загрузки фонового трафика, так как приоритетный трафик всегда будет обслуживаться перед фоновым и создаст дополнительную загрузку и для фонового трафика. Если же задача ТЕ решается для фонового трафика, то его интенсивность сравнивается со свободной интенсивностью счетчика фонового трафика и в случае положительного решения значение этой интенсивности вычитается только из счетчика фонового трафика, так как для приоритетного трафика фоновый трафик прозрачен.

Протоколы маршрутизации с расширениями в новых условиях должны распространять по сети информацию о двух параметрах свободной пропускной способности — для каждого класса трафика отдельно. Если же задача обобщается для случая передачи через сеть трафика нескольких классов, то, соответственно, с каждым ресурсом должно быть связано столько счетчиков, сколько классов трафика существует в сети, а протоколы маршрутизации должны распространять вектор свободных интенсивностей соответствующей размерности.

Выводы

- Для сбалансированной загрузки сети, обеспечивающей соблюдение параметров качества обслуживания для различных классов трафика с максимально возможным при этом коэффициенте загрузки ресурсов сети, применяются методы инжиниринга трафика (Traffic Engineering, TE).
- Методы TE опираются на расширенные протоколы маршрутизации, которые включают в информацию о состоянии связей данные о текущей свободной пропускной способности. На основе этих данных задача TE может быть решена автоматически маршрутизаторами или коммутаторами сети.

ГЛАВА 16 Глобальные сети с коммутацией каналов

Глобальные сети с коммутацией каналов создавались более ста лет назад для решения очень важной задачи — предоставления общедоступных телефонных услуг населению городов, сел и стран. Эта задача была с честью решена, и сегодня с помощью всемирной телефонной сети можно за несколько секунд связаться с нужным абонентом на другом континенте (несколько сложнее бывает установить соединения с соседним районом, но эти проблемы в нашей стране постепенно решаются). Телефонные сети работают на основе техники коммутации каналов, при этом каждая пара разговаривающих абонентов предварительно устанавливает через телефонную сеть соединение, образуя в ней дуплексный канал. Техника коммутации каналов оказалась для этого типа сетей наиболее подходящей, сочетающей хорошее качество передачи голоса с экономичностью реализации телефонных коммутаторов.

Как уже не раз подчеркивалось выше, для передачи пульсирующего компьютерного трафика техника коммутации каналов оказалась не вполне адекватной, в результате была разработана альтернативная техника коммутации пакетов. Однако сети с коммутацией каналов по-прежнему в разных вариантах часто используются для передачи данных. Чаще всего в качестве сетей доступа к сетям с коммутацией пакетов. Этот вариант применения связан с распространностью телефонной сети — ее абонентские окончания есть сегодня в большинстве городских квартир. Пакетные же сети, в частности Интернет, пока не могут похвастаться такой степенью распространения, специализированные для передачи пакетов абонентские окончания, образуемые модемами xDSL или кабельными модемами, сегодня редкость даже в наиболее развитых странах. Другим вариантом использования телефонной сети для передачи данных является применение ее коммутируемого или полупостоянного канала в качестве промежуточного звена магистрали пакетной сети. Здесь применение коммутируемых каналов весьма нежелательно, так как на коммутацию тратится дополнительное время, ведущее к нестабильной работе пакетной сети — например, заторам трафика и потерям пакетов. В любом случае сеть с коммутацией каналов предоставляет для пакетной сети услуги физического уровня, являясь после коммутации только физическим каналом «точка-точка».

В данной главе кратко рассмотрены принципы работы телефонных сетей двух типов: аналоговых и цифровых с интегральными услугами (ISDN). Необходимо отметить, что изучение этих сетей полезно для специалистов, занимающихся передачей данных, по некоторым причинам. Кроме необходимости знать особенности подключения к такой сети для использования ее в качестве физического канала» полезно знать принципы работы протоколов сигнализации телефонных сетей, с помощью которых каналы устанавливаются, так как в пакетных сетях аналогичные протоколы применяются для установления виртуальных каналов. Кроме того, процессы конвергенции привели к появлению новой услуги пакетных сетей — передачи телефонных разговоров (услуги IP-телефонии и пакетной телефонии). Протоколы пакетной телефонии для уста-

новления соединения между абонентами также используют протоколы, очень близкие к протоколам телефонных сетей.

Аналоговые телефонные сети

Устройство

Первые телефонные сети были аналоговыми, так как в них абонентское устройство — телефонный аппарат — преобразовывало звуковые колебания, являющиеся аналоговыми сигналами, в колебания электрического тока (также аналоговые сигналы). Коммутаторы телефонной сети тоже передавали пользовательскую информацию в аналоговой форме, возможно только перенося эти сигналы в другую область частотного спектра с помощью методов частотного уплотнения (FDM), описанного в главе 6. Сегодня в таких сетях все чаще между телефонными коммутаторами (автоматическими телефонными станциями, АТС) применяется передача голоса в цифровой форме путем мультиплексирования пользовательских каналов по времени, то есть с помощью методов TDM. Однако абонентские окончания остаются аналоговыми, что позволяет пользоваться теми же сравнительно простыми и недорогими аналоговыми телефонными аппаратами, что и раньше. В результате для абонента такие сети остаются аналоговыми, так как на его подключение и процедуры работы внутренние цифровые участки сети никакого влияния не оказывают. В англоязычной литературе такие сети иногда называют POTS (Plain Old Telephone Service) — что-то вроде «старая добная телефонная служба», хотя, конечно, название PSTN (Public Switched Telephone Network) — «публичная коммутируемая телефонная сеть», или в русском варианте — телефонная сеть общего пользования (ТфОП), является более официальным (оно, естественно, не включает корпоративные телефонные сети, которые не предоставляют публичные услуги, хотя технически устроены примерно так же).

Типичная структура аналоговой телефонной сети представлена на рис. 16.1. Сеть образована некоторым количеством АТС, которые соединены между собой физическими каналами. Топология связей между АТС в общем случае носит произвольный характер, хотя часто используется многоуровневая иерархия, когда несколько АТС нижнего уровня подключаются к одной АТС более высокого уровня и т. п. Предполагается, что эта сеть оказывает публичные услуги, поэтому на рисунке она обозначена как ТфОП.

К АТС нижнего уровня с помощью абонентских окончаний, которые представляют собой медные пары, подключаются телефонные аппараты абонентов. Обычно длина абонентского окончания не должна превышать одного-двух километров — отсюда возникает пресловутая проблема «последней мили» (примерно 1,6 км), которую нужно преодолеть при передаче данных с максимально возможной скоростью.

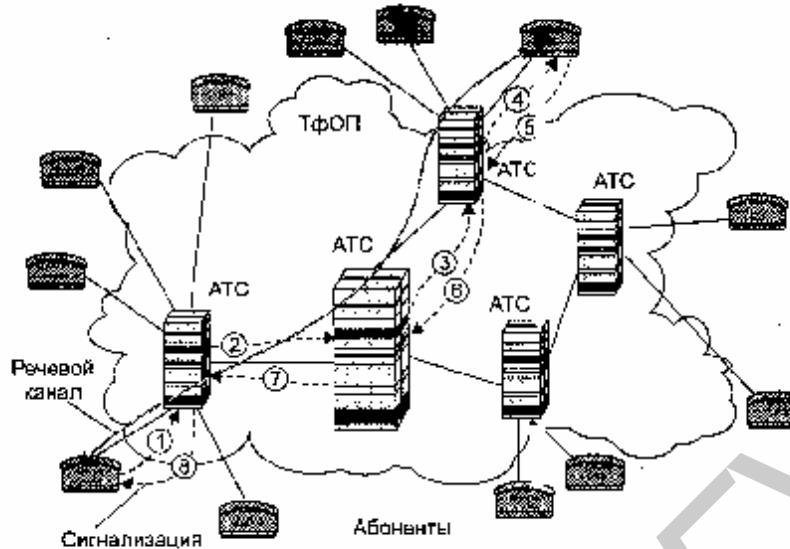


Рис. 16.1. Структура аналоговой телефонной сети

Отличительной особенностью телефонной сети является обязательная процедура предварительного установления соединения между абонентскими устройствами, в случае успеха которой в сети устанавливается канал между этими абонентами, через который они могут вести разговор. Эта процедура носит названия протокола сигнализации (signaling protocol).

Существует большое количество различных протоколов сигнализации, разработанных за долгие годы существования телефонных сетей. Однако все они сводятся к одной общей схеме. Сначала телефон вызывающего абонента передает в сеть сообщение, в котором указывается факт вызова и то, каким образом пересылается номер вызываемого абонента (этап 1 на рис. 16.1).

Так как аналоговый телефон — достаточно примитивное устройство, то сообщение вызова обычно представляет собой последовательность замыканий и размыканий электрической цепи, образуемой проводами абонентского окончания (часто называемого шлейфом). В ответ на первое замыкание телефонный коммутатор подает на абонентскую цепь некоторое напряжение, которое воспроизводится в виде постоянного зуммера динамика телефонной трубки. Человек активно участвует в процедуре вызова, набирая в ответ на зуммер цифры вызываемого номера. Существует несколько вариантов передачи номера. При импульсном способе, принятом в нашей стране, каждая цифра передается соответствующим числом последовательных импульсов размыкания-замыкания частотой 10 или 20 Гц (рис. 16.2).

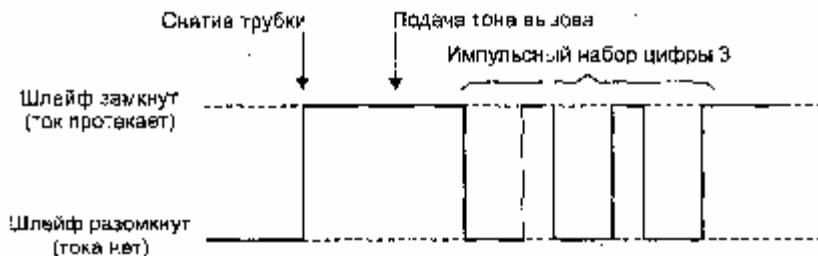


Рис. 16.2. Взаимодействие телефонного аппарата с АТС при вызове

При тоновом наборе для кодирования цифр используется комбинация двух частот, одна из низкочастотной группы (697, 770, 852 и 941 Гц) и одна из высокочастотной группы (1209, 1336 и 1477 Гц). Тоновый набор выполняется с частотой 10 Гц сигналами длительностью в 50 мс с паузами также в 50 мс. Так как одна цифра номера при импульсном наборе передается несколькими импульсами, а при тоновом наборе — одним сигналом, то скорость тонового набора в несколько раз выше, чем импульсного.

После приема такого условного «сообщения» от телефонного аппарата первая АТС маршрутизирует его дальше (этапы 2, 3, 4). Для этого у него должна быть каким-то образом задана таблица маршрутизации. Она не обязательно хранится в оперативной памяти в форме двоичных чисел, как таблица маршрутизации современного коммутатора или маршрутизатора пакетных сетей, ее вид зависит от конструкции телефонного коммутатора. Коммутаторы подразделяются на электромеханические и программируемые электронные. Электромеханические коммутаторы (например, координатные или декадно-шаговые) управляются по проводным цепям и приводятся в действие электродвигателями или шаговымиискателями. В электромеханических системах логика маршрутизации встроена в аппаратуру. В программируемых коммутаторах логика коммутации задается хранимой в памяти таблицей маршрутизации, а сама коммутация выполняется электронным способом.

Выбрав нужный выходной порт, первая АТС передает на него сообщение вызова (во многих протоколах оно носит название Call или Setup), которое в результате попадает на следующую АТС, повторяющую процедуру его обработки и передающую его следующей АТС. Так как АТС представляют собой гораздо более сложные, чем телефонный аппарат, устройства, они обычно используют более компактную форму представления данных вызова, например в виде частотных сигналов или даже двоичных нулей или единиц. Поэтому протокол сигнализации между АТС в общем случае отличается от протокола сигнализации между телефоном и АТС. В сообщения сигнализации, передаваемые между АТС, может вставляться дополнительная информация, необходимая для работы сети или делающая услугу более удобной, например номер вызывающего абонента.

Каждая АТС передает сообщение вызова далее только в том случае, если в этом направлении у нее осталась свободной коммутационная емкость, то есть она может проложить через данный порт еще одно пользовательское соединение.

Например, если между станциями используется уплотненный канал FDM с двенадцатью пользовательскими подканалами и через него проходит 11 активных соединений, то еще одно соединение установить можно и сообщение через этот порт будет передано далее. Однако при двенадцати активных соединениях АТС уже не сможет обработать сообщение вызова в прежнем стиле и вернет абоненту сигнал отказа от обслуживания, который индицируется в трубке короткими гудками. Отказ принять запрос на соединение из-за того, что в каком-то направлении исчерпана коммутационная емкость, — характерная особенность сетей с коммутацией каналов.

Если свободная емкость в нужном направлении имеется, то АТС резервирует его для вновь образуемого соединения, запоминая этот факт.

В том случае, когда сообщение вызова успешно дошло до АТС, к которой непосредственно подключен телефонный аппарат вызываемого абонента, то на абонентском окончании вновь, как и в самом начале, применяется сигнализация путем замыкания и размыкания шлейфа. На аппарат вызываемого абонента по шлейфу подается сигнал звонка, в результате телефон начинает звонить. В обратном направлении также посыпается соответствующее сообщение, в результате которого на аппарат вызывающего абонента приходят сигналы длинных гудков.

Если вызываемый абонент снимает трубку (этап 5), то вызов на абонентском конце считается принятым, и оконечная АТС передает в обратном направлении сообщение протокола сигнализации Connect — «Соединение». Сообщение «Соединение» передается в обратном направлении через промежуточные АТС, участвовавшие в прямом продвижении запроса, при этом происходит окончательная коммутация составного канала в пределах данной АТС (этапы 6, 7, 8). Далее по установленному через телефонную сеть каналу передается в дуплексном режиме голосовая информация. Если сеть является полностью аналоговой, то голос на всем пути следования передается в аналоговой форме, при этом каждая промежуточная АТС выделяет ему канал шириной 3,1 кГц (канал тональной частоты), но перенесенный для мультиплексирования по частоте (FDM) в другой диапазон частот, например в диапазон от 60 до 108 кГц, если между АТС работает канал первого уровня уплотнения. Если же на пути составного канала встречаются цифровые АТС, то на их входе выполняется преобразование аналогового сигнала в цифровой, коммутация осуществляется методами ТОМ, а при передаче на аналоговую АТС информация претерпевает обратное преобразование в аналоговую форму.

Передача данных

В аналоговых телефонных сетях составной канал между абонентами имеет полосу пропускания 3,1 кГц, которая обусловлена методом мультиплексирования FDM и не может быть расширена. Для передачи данных по такому каналу (который на самом деле может быть и немного шире, потому что для надежного разделения каналов их ширина с некоторым запасом выбрана в 4 кГц, оставшиеся 900 Гц АТС может использовать для служебных целей) и были созданы

такие популярные устройства, как модемы (модуляторы-демодуляторы). Для телефонной сети модемы являются терминальными устройствами, которые, как и телефоны, выполняют стандартную процедуру вызова абонента путем замыкания-размыкания шлейфа. Если у вызываемого абонента к телефонной сети также присоединен модем, то он отвечает на вызов стандартным для телефонной сети способом, в результате чего устанавливается составной канал тональной частоты. Затем модемы его используют для установления соединения уже на логическом уровне (характерный свист и шум в динамике модема), после чего начинают обмениваться данными, представленными в виде модулированных синусоид. Наивысшим достижением современных модемов на канале тональной частоты является достижение скорости в 33,6 кбит/с в том случае, когда на пути следования информации необходимо делать хотя бы одно аналого-цифровое преобразование, и 56 кбит/с в случае, когда информация подвергается только цифро-аналоговому преобразованию. Такая асимметрия связана с тем, что аналого-цифровое преобразование вносит существенно более значительные искажения в передаваемые дискретные данные, чем цифро-аналоговое. Очевидно, что такие скорости нельзя назвать приемлемыми для большинства современных приложений, которые широко используют графику и другие мультимедийные формы представления данных.

Поэтому чаще всего каналы тональной частоты используются для индивидуального удаленного доступа к сети или же в качестве резервных линий связи небольших офисов с центральным отделением предприятия. Доступ по телефонной сети имеет англоязычное название «dial-up access». Тем не менее при недостатке средств коммутируемые аналоговые линии обеспечивают связь локальных сетей между собой. Это выгодный режим соединения, если количество передаваемых данных невелико и данные не требуют частого обновления. В этом случае две сети могут соединяться по аналоговой телефонной сети, например, раз в сутки, передавать в течение нескольких минут данные, а затем разрывать соединение. При повременной оплате телефонного соединения такой режим оказывается эффективным. Обычно к нему прибегают для передачи сводок работы предприятия за день, точнее тех частей сводок, которые имеют небольшие объемы (чаще всего — это числовые показатели, без графики).

Скорость передачи данных ограничивает не только недостаточная ширина полосы канала тональной частоты, а также и то обстоятельство, что электромеханические коммутаторы создают значительные помехи в коммутируемых каналах. Кроме того, дополнительные помехи создает сам способ коммутации уплотненных каналов на основе FDM. Это объясняется тем, что коммутировать уплотненные в общий канал исходные сигналы отдельных абонентов невозможно. Перед операцией коммутации всегда нужно провести полное демультиплексирование сигналов абонентских каналов, то есть превратить сигнал высокочастотной несущей, который находится, например, в диапазоне от 60 до 108 кГц для уплотненного канала первого уровня, в голосовой сигнал со спектром от 300 до 3400 Гц. Только затем такие каналы можно коммутировать с помощью шаговых искателей или электронных ключей. После коммутации абонентские каналы снова уплотняются в высокочастотный канал, но каждый входной

канал теперь уже накладывается на несущую другой порядковой частоты, что и соответствует операции коммутации.

Операция демультиплексирования высокочастотной несущей, а затем повторное наложение сигналов на высокочастотные несущие создает значительные помехи (треск и свист в телефонной трубке), которые существенно снижают качество коммутируемых каналов по сравнению с выделенными аналоговыми. Понятно, что наличие электромеханических элементов только усугубляет картину, а старые АТС с шаговымиискателями еще эксплуатируются.

Для повышения качества каналов телефонных сетей до уровня, пригодного для высокоскоростной передачи данных, необходим переход от аналоговых телефонных сетей к цифровым, в которых на всех участках сети, начиная от абонентского окончания, голос передается в дискретной форме. Наиболее развитыми (но не единственными) сетями такого типа являются сети с интегральными услугами, ISDN, в которых не только осуществлен переход к полностью цифровой форме передачи голоса, но и значительно расширен набор предоставляемых абонентам сети услуг.

Выводы

- Сети с коммутацией каналов используются в корпоративных сетях в основном для удаленного доступа многочисленных домашних пользователей и гораздо реже — для соединения локальных сетей.
- Отличительными особенностями всех сетей с коммутацией каналов являются: работа в режиме установления соединений, возможность блокировки вызова конечным абонентом или промежуточным коммутатором, необходимость использования на обоих концах сети устройств, поддерживающих одну и ту же скорость передачи данных, так как этот вид сетей не выполняет промежуточную буферизацию данных.
- Сети с коммутацией каналов делятся на аналоговые и цифровые. Аналоговые сети могут использовать аналоговую (FDM) и цифровую (TDM) коммутацию, но в них всегда абонент подключен по аналоговому двухпроводному окончанию. В цифровых сетях мультиплексирование и коммутация всегда выполняются по способу коммутации ТОМ, а абоненты всегда подключаются по цифровому абонентскому окончанию (DSL).
- Аналоговые сети обеспечивают вызов посредством импульсного или тонового набора номера с частотой 110 Гц, причем тоновый набор примерно в 5 раз быстрее импульсного.
- Аналоговые сети используют электромеханические коммутаторы, создающие большие помехи, и электронные программируемые коммутаторы. При работе электронного коммутатора в режиме частотного уплотнения (FDM) создаются дополнительные помехи при демультиплексировании и мультиплексировании абонентских каналов.

Цифровые сети с интегральными услугами

Цели и история создания технологии ISDN

ISDN (Integrated Services Digital Network — цифровые сети с интегральными услугами) относятся к сетям, в которых основным режимом коммутации является' режим коммутации каналов, а данные обрабатываются в цифровой форме. Идеи перехода с телефонных сетей общего пользования на полностью цифровую обработку данных, при которой конечный абонент передает данные непосредственно в цифровой форме, высказывались давно. Сначала предполагалось, что абоненты этой сети будут передавать только голосовые сообщения. Такие сети получили название 1DN (Integrated Digital Network — интегрированная цифровая сеть). Термин «интегрированная» относился к интеграции цифровой обработки информации сетью с цифровой передачей голоса абонентом. Идея такой сети была высказана еще в 1959 году. Затем было решено, что она должна предоставлять своим абонентам не только возможность поговорить между собой, но и воспользоваться другими услугами — в первую очередь передачей компьютерных данных. Кроме того, сеть должна была поддерживать для абонентов разнообразные услуги прикладного уровня — факсимильную связь, телетекс (передачу данных между двумя терминалами), видеотекс (получение хранящихся в сети данных на свой терминал), голосовую почту и ряд других. Предпосылки для создания такого рода сетей сложились к середине 70-х годов. К этому времени уже широко применялись цифровые каналы T1 для передачи данных в цифровой форме между АТС, а первый мощный цифровой коммутатор телефонных каналов 4ESS был выпущен компанией Western Electric в 1976 году.

В результате работ, проводимых по стандартизации интегральных сетей в CCITT, в 1980 году появился стандарт G.705, в котором излагались общие идеи такой сети. Конкретные спецификации сети ISDN появились в 1984 году в виде серии рекомендаций I. Этот набор спецификаций был неполным и не подходил для построения законченной сети. К тому же в некоторых случаях он допускал неоднозначность толкования или был противоречивым. В результате, хотя оборудование ISDN и начало появляться примерно с середины 80-х годов, оно часто было несовместимым, особенно если производилось в разных странах. В 1988 году рекомендации серии I были пересмотрены и приобрели гораздо более детальный и законченный вид, хотя некоторые неоднозначности сохранились. В 1992 и 1993 годах стандарты ISDN были еще раз пересмотрены и дополнены. Процесс стандартизации этой технологии продолжается.

Внедрение сетей ISDN началось достаточно давно — с конца 80-х годов, однако техническая сложность пользовательского интерфейса, отсутствие единых стандартов на многие жизненно важные функции, а также необходимость крупных капиталовложений для переоборудования телефонных АТС и каналов связи привели к тому, что «инкубационный период» затянулся на многие годы, и сейчас, когда прошло уже более десяти лет, распространенность сетей ISDN

оставляет желать лучшего. Кроме того, в разных странах судьба ISDN складывалась по-разному. Раньше других в национальном масштабе эти сети зарабатали в таких странах, как Германия и Франция. Однако даже в этих странах доля абонентов ISDN составляет немногим более 5 % от общего числа абонентов телефонной сети. В США процесс внедрения сетей ISDN намного отстал от Европы, поэтому сетевая индустрия только недавно заметила наличие такого рода сетей. Если судить о тех или иных типах глобальных сетей по коммуникационному оборудованию для корпоративных сетей, то может сложиться ложное впечатление, что технология ISDN появилась где-то в 1994-1995 годах, так как именно в эти годы начали появляться маршрутизаторы с поддержкой интерфейса ISDN. Это

обстоятельство просто отражает тот факт, что именно в эти годы сеть ISDN стала достаточно распространенной в США — стране, компании которой являются лидерами в производстве сетевого оборудования для корпоративных сетей.

Архитектура сети ISDN предусматривает несколько видов служб (рис. 16.3):

- некоммутируемые средства (выделенные цифровые каналы);
- коммутируемая телефонная сеть общего пользования;
- сеть передачи данных с коммутацией каналов;
- сеть передачи данных с коммутацией пакетов;
- сеть передачи данных с трансляцией кадров (режим frame relay);
- средства контроля и управления работой сети.

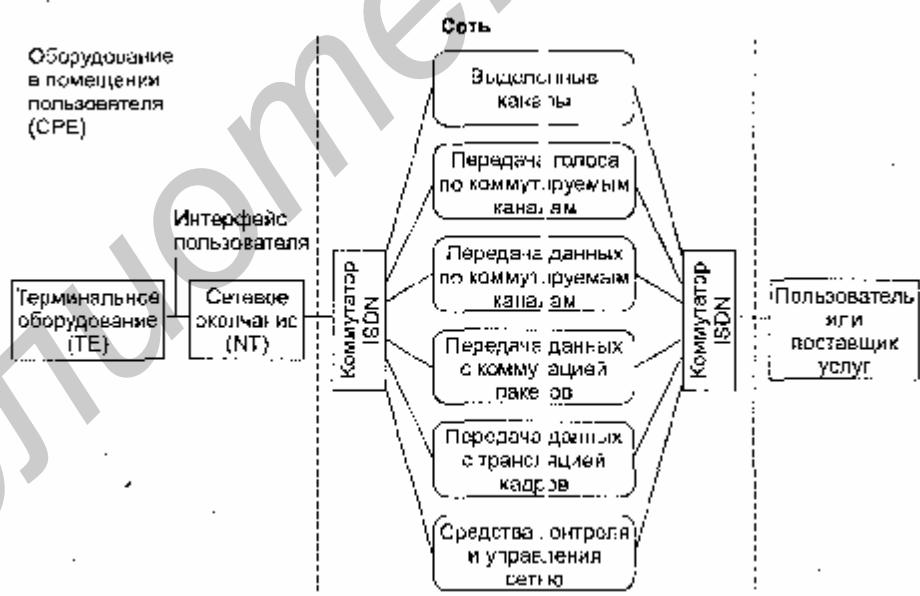


Рис. 16.3. Службы ISDN

Как видно из приведенного списка, транспортные службы сетей ISDN действительно покрывают очень широкий спектр услуг, включая популярные услуги frame relay. Кроме того, большое внимание уделено средствам контроля сети, которые позволяют маршрутизировать вызовы для установления соединения с абонентом сети, а также осуществлять мониторинг и управление сетью. Управ-

ляемость сети обеспечивается интеллектуальностью коммутаторов и конечных узлов сети, поддерживающих стек протокола», в том числе и специальных протоколов управления.

Стандарты ISDN описывают также ряд услуг прикладного уровня: факсимильную связь на скорости 64 кбит/с, телексную связь на скорости 9600 бит/с, видеотекс на скорости 9600 бит/с и некоторые другие

На практике не все сети ISDN поддерживают все стандартные службы. Служба frame relay хотя и была разработана в рамках сети ISDN, однако реализуется, как правило, с помощью отдельной сети коммутаторов кадров, не пересекающейся с сетью коммутаторов ISDN.

Базовой скоростью сети ISDN является скорость канала DS-0, то есть 64 кбит/с. Эта скорость ориентируется на самый простой метод кодирования голоса — ИКМ, хотя дифференциальное кодирование и позволяет передавать голос с тем же качеством на скорости 32 или 16 кбит/с.

Одной из оригинальных идей, положенных в основу ISDN, является совместное использование принципов коммутации каналов и коммутации пакетов. Однако сеть с коммутацией пакетов, работающая в составе ISDN, выполняет служебные функции — с помощью этой сети передаются сообщения протокола сигнализации. А вот основная информация, то есть сам голос, по-прежнему передается с помощью сети с коммутацией каналов. В таком разделении функций есть вполне понятная логика — сообщения о вызове абонентов образуют пульсирующий трафик, поэтому его более эффективно передавать по сети с коммутацией пакетов.

Пользовательские интерфейсы ISDN

Одним из базовых принципов ISDN является предоставление пользователю стандартного интерфейса, с помощью которого пользователь может запрашивать у сети разнообразные услуги. Этот интерфейс образуется между двумя типами оборудования, устанавливаемого в помещении пользователя (Customer Premises Equipment, CPE): терминальным оборудованием пользователя ТЕ (компьютер с соответствующим адаптером, маршрутизатор, телефонный аппарат) и сетевым окончанием NT, которое представляет собой устройство, завершающее канал связи с ближайшим коммутатором ISDN.

Пользовательский интерфейс основан на каналах трех типов:

- В — со скоростью передачи данных 64 кбит/с;
- D — со скоростью передачи данных 16 или 64 кбит/с;
- Н — со скоростью передачи данных 384 кбит/с (НО), 1536 кбит/с (НИ) или 1920 кбит/с (Н12).

Каналы типа В обеспечивают передачу пользовательских данных (оцифрованного голоса, компьютерных данных или емиси голоса и данных) и с более низкими скоростями, чем 64 кбит/с. Разделение данных выполняется с помощью техники ТОМ. Разделением канала В на подканалы в этом случае должно заниматься пользовательское оборудование, сеть ISDN всегда коммутирует целые каналы типа В. Каналы типа В могут соединять пользователей с помощью тех-

ники коммутации каналов друг с другом, а также образовывать так называемые полупостоянные (semipermanent) соединения, которые эквиваленты соединениям службы выделенных каналов. Канал типа В может также подключать пользователя к коммутатору сети X.25.

Канал типа D является каналом доступа к служебной сети с коммутацией пакетов, передающей сигнальную информацию. Передача адресной информации, на основе которой осуществляется коммутация каналов типа В в коммутаторах сети, является основной функцией канала D. Другой его функцией является поддержание услуг низкоскоростной сети с коммутацией пакетов для пользовательских данных. Обычно эта услуга выполняется сетью в то время, когда каналы типа D свободны от выполнения основной функции.

Каналы типа Н предоставляют пользователям возможности высокоскоростной передачи данных. На них могут работать службы высокоскоростной передачи факсов, видеинформации, качественного воспроизведения звука.

Пользовательский интерфейс ISDN представляет собой набор каналов определенного типа и с определенными скоростями.

Сеть ISDN поддерживает два типа пользовательского интерфейса — начальный (Basic Rate Interface, BRI) и основной (Primary Rate Interface, PRI).

Начальный интерфейс BRI предоставляет пользователю два канала по 64 кбит/с для передачи данных (каналы типа В) и один канал с пропускной способностью 16 кбит/с для передачи управляющей информации (канал типа D). Всего каналы работают в полнодуплексном режиме. В результате суммарная скорость интерфейса BRI для пользовательских данных составляет 144 кбит/с по каждому направлению, а с учетом служебной информации — 192 кбит/с. Различные каналы пользовательского интерфейса разделяют один и тот же физический двухпроводный кабель по технологии TDM, то есть являются логическими, а не физическими каналами. Данные по интерфейсу BRI передаются кадрами, состоящими из 48 бит. Каждый кадр содержит по два байта каждого из В каналов, а также 4 бита канала D. Передача кадра длится 250 мс, что обеспечивает скорость данных 64 кбит/с для каналов В и 16 кбит/с для канала D. Кроме битов данных кадр содержит служебные биты для синхронизации кадров, а также обеспечения нулевой постоянной составляющей электрического сигнала.

Интерфейс BRI может поддерживать не только схему 2B+D, но и B+D и просто D (когда пользователь направляет в сеть только пакетизированные данные).

Начальный интерфейс стандартизован в рекомендации 1.430.

Основной интерфейс PRI предназначен для пользователей с повышенными требованиями к пропускной способности сети. Интерфейс PRI поддерживает либо схему 30B+D, либо схему 23B+D. В обеих схемах канал D обеспечивает скорость 64 кбит/с. Первый вариант предназначен для Европы, второй — для Северной Америки и Японии. Ввиду большой популярности скорости цифровых каналов 2,048 Мбит/с в Европе и скорости 1,544 Мбит/с в остальных регионах привести стандарт на интерфейс PRI к общему варианту не удалось.

Возможны варианты интерфейса PRI с меньшим количеством каналов типа В, например 20B+D. Каналы типа В могут объединяться в один логический высокоскоростной канал с общей скоростью до 1920 кбит/с. При установке у пользо-

зователя нескольких интерфейсов PRI все они могут иметь один канал типа D, при этом количество каналов B в том интерфейсе, который не имеет канала D, может увеличиваться до 24 или 31.

Основной интерфейс может быть основан на каналах типа H. При этом общая пропускная способность интерфейса все равно не должна превышать 2,048 или 1,544 Мбит/с. Для каналов HO возможны интерфейсы 3HO+D для американского варианта и 5HO+D для европейского. Для каналов HI возможен интерфейс, состоящий только из одного канала H11 (1,536 Мбит/с) для американского варианта или одного канала 1112 (1,920 Мбит/с) и одного канала D для европейского варианта.

Кадры интерфейса PRI имеют структуру кадров DS-1 для каналов T1 или E1. Основной интерфейс PRI стандартизован в рекомендации 1.431.

Подключение пользовательского оборудования к сети ISDN

Подключение пользовательского оборудования к сети ISDN осуществляется в соответствии со схемой подключения, разработанной CCITT (рис. 16.4). Оборудование делится па функциональные группы, и в зависимости от группы различается несколько справочных точек (reference points) соединения разных групп оборудования между собой.

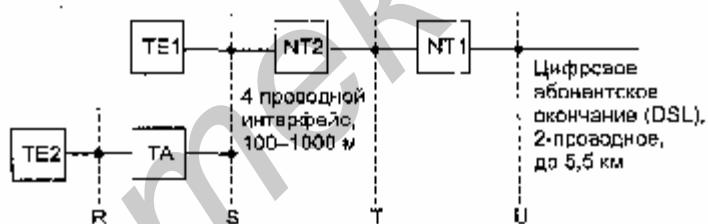


Рис. 16.4. Подключение пользовательского оборудования ISDN

Устройства функциональной группы NT1 (Network Termination 1) образуют цифровое абонентское окончание (Digital Suscriber Line, DSL) на кабеле, соединяющем пользовательское оборудование с сетью ISDN. Фактически NT1 представляет собой устройство типа CSU, которое работает на физическом уровне и образует дуплексный канал с соответствующим устройством CSU, установленным на территории оператора сети ISDN. Справочная точка U соответствует точке подключения устройства NT1 к сети. Устройство NT1 может принадлежать оператору сети (хотя всегда устанавливается в помещении пользователя), а может принадлежать и пользователю. В Европе принято считать устройство NT1 частью оборудования сети, поэтому пользовательское оборудование (например, маршрутизатор с интерфейсом ISDN) выпускается безстроенного устройства NT1. В Северной Америке принято считать устройство NT1 принадлежностью пользовательского оборудования, поэтому пользовательское оборудование часто выпускается со встроенным устройством NT1.

Если пользователь подключен через интерфейс BRI, то цифровое абонентское

окончание выполнено по 2-проводной схеме (как и обычное окончание аналоговой телефонной сети). Для организации дуплексного режима используется технология одновременной выдачи передатчиками потенциального кода 2B1Q с эхо-подавлением и вычитанием своего сигнала из суммарного. Максимальная длина абонентского окончания в этом случае составляет 5,5 км.

При использовании интерфейса PRI цифровое абонентское окончание выполняется по схеме канала T1 или E1, то есть является 4-проводным с максимальной длиной около 1800 м.

Устройства функциональной группы NT2 {Network Termination 2} представляют собой устройства канального или сетевого уровня, которые выполняют функции концентрации пользовательских интерфейсов и их мультиплексирование. Например, к этому типу оборудования относятся: офисная АТС (PBX), коммутирующая несколько интерфейсов BRI, маршрутизатор, работающий в режиме коммутации пакетов (например, по каналу D), простой мультиплексор TDM, который мультиплексирует несколько низкоскоростных каналов в один канал типа В. Точка подключения оборудования типа NT2 к устройству NT1 называется справочной точкой типа Т. Наличие этого типа оборудования не является обязательным в отличие от NT1.

Устройства функциональной группы TE1 (Terminal Equipment 1) относятся к устройствам, которые поддерживают интерфейс пользователя BRI или PRI. Справочная точка S соответствует точке подключения отдельного терминалного оборудования, поддерживающего один из интерфейсов пользователя ISDN. Таким оборудованием может быть цифровой телефон или факс-аппарат. Так как оборудование типа NT2 может отсутствовать, то справочные точки S и Т объединяются и обозначаются как S/T.

Устройства функциональной группы TE2 (Terminal Equipment 2) представляют собой устройства, которые не поддерживают интерфейс BRI или PRI. Таким устройством может быть компьютер, маршрутизатор с последовательными интерфейсами, не относящимися к ISDN, например RS-232C, X.21 или V.35. Для подключения такого устройства к сети ISDN необходимо использовать терминальный адаптер (Terminal Adaptor, TA). Для компьютеров терминальные адAPTERы выпускаются в формате сетевых адаптеров — как встраиваемая карта.

Физически интерфейс в точке S/T представляет собой 4-проводную линию. Так как кабель между устройствами TE1 или TA и сетевым окончанием NT1 или NT2 обычно имеет небольшую длину, то разработчики стандартов ISDN решили не усложнять оборудование, поскольку организация дуплексного режима на 4-проводной линии намного легче, чем на 2-проводной. Для интерфейса BRI в качестве метода кодирования выбран биполярный метод AMI, причем логическая единица кодируется нулевым потенциалом, а логический ноль — чередованием потенциалов противоположной полярности. Для интерфейса PRI используются другие коды, те же, что и для интерфейсов T1 и E1, то есть соответственно B8ZS и HDB3.

Физическая длина интерфейса PRI колеблется от 100 до 1000 м в зависимости от схемы подключения устройств (рис. 16.5).

Дело в том, что при небольшом количестве терминалов (TE1 или TE2+TA) разрешается не использовать местную офисную АТС, а подключать до 8 устройств к одному устройству типа NT1 (или NT2 без коммутационных возможностей) по схеме монтажного ИЛИ (подключение напоминает подключение станций к коаксиальному кабелю Ethernet, но только в 4-проводном варианте). При подключении одного устройства TE (через терминальные резисторы R, согласующие параметры линии) к сетевому окончанию NT (см. рис. 16.5, а) длина кабеля может достигать 1000 м. При подключении нескольких устройств к пассивному кабелю (см. рис. 16.5, б) максимальная длина кабеля сокращается до 100-200 м. Правда, если эти устройства сосредоточены на дальнем конце кабеля (расстояние между ними не превышает 25-50 м), то длина кабеля может быть увеличена до 500 м (рис. 16.5, в). И наконец, существуют специальные многопортовые устройства NT1, которые обеспечивают звездообразное подключение до 8 устройств, при этом длина кабеля увеличивается до 1000 м (рис. 16.5, г).

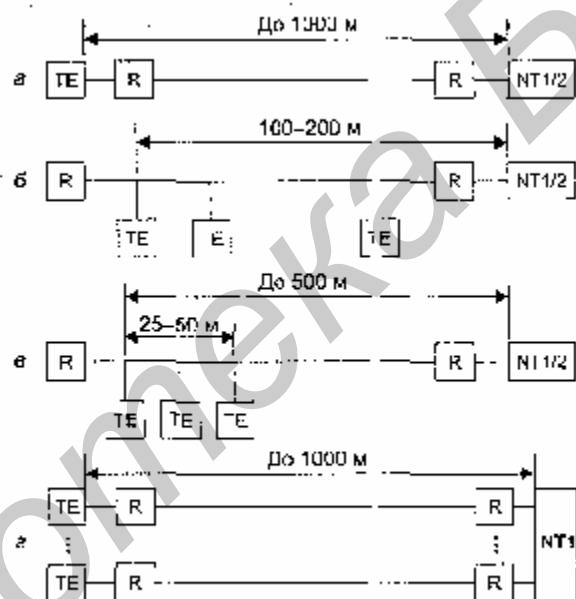


Рис. 16.5. Многоточечное подключение терминалов к сетевому окончанию

Адресация в сетях ISDN

Технология ISDN разрабатывалась как основа всемирной телекоммуникационной сети, позволяющей связывать как телефонных абонентов, так и абонентов других глобальных сетей — компьютерных, телексных. Поэтому при разработке схемы адресации узлов ISDN необходимо было, во-первых, сделать эту схему достаточно емкой для всемирной адресации, а во-вторых, совместимой со схемами адресации других сетей, чтобы абоненты этих сетей, в случае соединения своих сетей через сеть ISDN, могли бы пользоваться привычными форматами адресов. Разработчики стека TCP/IP пошли по пути введения собственной системы адресации, независимой от систем адресации объединяемых сетей, а разработчики технологии ISDN пошли по другому пути — они решили до-

биться использования в адресе ISDN адресов объединяемых сетей.

Основное назначение ISDN — передача телефонного трафика. Поэтому за основу адреса ISDN был взят формат международного телефонного плана номеров, описанный в стандарте ITU-T E.163. Однако этот формат был расширен для поддержки большего числа абонентов и для использования в нем адресов других сетей, например X.25. Стандарт адресации в сетях ISDN получил номер E.1G4.

Формат E.163 предусматривает до 12 десятичных цифр в номере, а формат адреса ISDN в стандарте E.164 расширен до 55 десятичных цифр. В сетях ISDN различают номер абонента и адрес абонента. Номер абонента соответствует точке T подключения всего пользовательского оборудования к сети. Например, вся офисная АТС может идентифицироваться одним номером ISDN. Номер ISDN состоит из 15 десятичных цифр и делится, как и телефонный номер по стандарту E.163, на поле «Код страны» (от 1 до 3 цифр), поле «Код города» и поле «Номер абонента». Адрес ISDN включает номер плюс до 40 цифр подадреса. Подадрес используется для нумерации терминальных устройств за пользовательским интерфейсом, то есть подключенных к точке S. Например, если на предприятии имеется офисная АТС, то ей можно присвоить один номер, например 7-095-640-20-00, а для вызова абонента, имеющего подадрес 134, внешний абонент должен набрать номер 7-095-640-20-00-134.

При вызове абонентов из сети, не относящейся к ISDN, их адрес может непосредственно заменять адрес ISDN. Например, адрес абонента сети X.25, в которой используется система адресации по стандарту X.121, может быть помещен целиком в поле адреса ISDN, но для указания, что это адрес стандарта X.121, ему должно предшествовать поле префикса, в которое помещается код стандарта адресации, в данном случае стандарта X.121. Коммутаторы сети ISDN могут обработать этот адрес корректно и установить связь с нужным абонентом сети X.25 через сеть ISDN, либо коммутируя канал типа B с коммутатором X.25, либо передавая данные по каналу типа D в режиме коммутации пакетов. Префикс описывается стандартом ISO 7498.

Стандарт ISO 7498 определяет достаточно сложный формат адреса, причем основой схемы адресации являются первые два поля. Поле AFI (Authority and Format Identifier) определяет значения всех остальных полей адреса и формат этих полей. Значением поля AFI является один из 6 типов поддоменов глобального домена адресации:

- четыре типа доменов соответствуют четырем типам публичных телекоммуникационных сетей — сетей с коммутацией пакетов, телексных сетей, публичных телефонных сетей и сетей ISDN;
- пятый тип домена — это географический домен, который назначается каждой стране (в одной стране может быть несколько географических доменов);
- шестой тип домена — это домен организационного типа, в который входят международные организации, например ООН или ATM Forum.

За полем AFI идет поле IDI (Initial Domafi Identifier — начальный идентификатор домена), а за ним располагается дополнительное поле DSP (Domain Specific Part), которое может нести дополнительные цифры номера абонента, если раз-

рядности поля INI не хватает.

Определены перечисленные ниже значения AFI.

□ Международные сети с коммутацией пакетов со структурой адресов в стандарте X.121 — 36, если адрес задается только десятичными цифрами, и 37, если адрес состоит из произвольных двоичных значений. При этом поле INI имеет формат в 14 десятичных цифр, а поле DSP может содержать еще 24 цифры.

□ Международные сети ISDN со структурой адресов в стандарте E.164 — 44, если адрес задается только десятичными цифрами, и -15, если адрес состоит из произвольных двоичных значений. При этом поле IDI имеет формат в 15 десятичных цифр, а поле DSP может содержать еще 40 цифр.

□ Международные телефонные сети PSTN со структурой адресов в стандарте E.163 — 42, если адрес задается только десятичными цифрами, и 43, если адрес состоит из произвольных двоичных значений. При этом поле IDI имеет формат в 12 десятичных цифр, а поле DSP может содержать еще 26 цифр.

□ Международные географические домены со структурой адресов в стандарте ISO DCC (Digital Country Codes) — 38, если адрес задается только десятичными цифрами, и 39, если адрес состоит из произвольных двоичных значений. При этом поле INI имеет формат в три десятичных цифры (код страны), а поле DSP может содержать еще 35 цифр.

□ Домен международных организаций. Для него однобайтовое поле IDI содержит код международной организации, от которой зависит формат поля DSP.

Для первых четырех доменов адрес абонента помещается непосредственно в поле TDI. Для пятого и шестого типов доменов ID1 содержит только код страны или код организации, которая контролирует структуру и нумерацию части DSP.

Еще одним способом вызова абонентов из других сетей является указание в адресе ISDN двух адресон: адреса ISDN пограничного устройства, например, соединяющего сеть ISDN с сетью X.25, и адреса узла в сети X.25. Адреса должны разделяться специальным разделителем. Два адреса используются за два этапа — сначала сеть ISDN устанавливает соединение типа коммутируемого канала с пограничным устройством, присоединенным к сети ISDN, а затем передает ему вторую часть адреса, чтобы это устройство осуществило соединение с требуемым абонентом.

Стек протоколов и структура сети ISDN

В сети ISDN существует два стека протоколов: стек каналов типа D и стек каналов типа B (рис. 16.6).

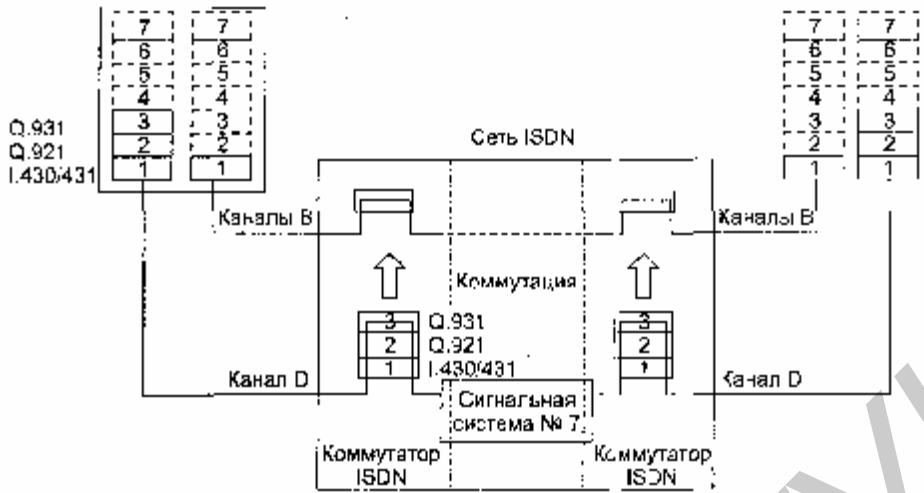


Рис. 16.6. Структура сети ISDN

Каналы типа D образуют достаточно традиционную сеть с коммутацией пакетов. Прообразом этой сети послужила технология сетей X.25. Для сети каналов D определены три уровня протоколов: физический протокол определяется стандартом 1.430/431, канальный протокол LAP-D определяется стандартом Q.921, а на сетевом уровне может использоваться протокол Q..931, с помощью которого выполняется маршрутизация вызова абонента службы с коммутацией каналов, или же протокол X.25 — в этом случае в кадры протокола LAP-D вкладываются пакеты X.25 и коммутаторы ISDN выполняют роль коммутаторов X.25.

Сеть каналов типа D внутри сети ISDN служит транспортной сетью с коммутацией пакетов для так называемой системы сигнализации номер 7 (Signal System Number 7, SS7). Система SS7 была разработана для целей внутреннего мониторинга и управления коммутаторами телефонной сети общего назначения. Эта система применяется и в сети ISDN. Служба SS7 относится к прикладному уровню модели OSI. Конечному пользователю ее услуги недоступны, так как сообщениями SS7 коммутаторы сети обмениваются только между собой. В России применяется несколько модифицированный вариант этой системы сигнализации под названием общеканальная сигнализация № 7 (ОКС 7). Термин «общеканальная» означает, что сообщения сигнализации передаются по выделенному служебному каналу, общему для всех пользовательских каналов, а по последним передается только информация пользователей, и никакая другая.

Каналы типа B образуют сеть с коммутацией цифровых каналов. В терминах модели OSI на каналах типа B в коммутаторах сети ISDN определен только протокол физического уровня — протокол 1.430/431. Коммутация каналов типа B происходит по указаниям, полученным по каналу D. Когда пакеты протокола Q.931 маршрутизируются коммутатором, то при этом происходит одновременная коммутация очередной части составного канала от исходного абонента к конечному.

Протокол LAP-D принадлежит семейству HDLC, к которому относится и описанный в главе 7 протокол LLC2. Протокол LAP-D обладает всеми родовыми

чертами этого семейства, но имеет и некоторые особенности. Адрес кадра LAP-D состоит из двух байтов — один байт определяет код, службы, которой пересылаютсяложенные в кадр пакеты, а второй используется для адресации одного из терминалов, если у пользователя к сетевому окончанию NT1 подключено несколько терминалов. Терминальное устройство может поддерживать разные службы — службу установления соединения по протоколу Q.931, службу коммутации пакетов X.25, службу мониторинга сети и т. п. Протокол LAP-D обеспечивает два режима работы: с установлением соединения (единственный режим работы протокола LLC2) и без установления соединения. Последний режим используется, например, для управления и мониторинга сети.

Протокол Q.931 переносит в своих пакетах адрес ISDN вызываемого абонента, на основании которого и происходит настройка коммутаторов на поддержку составного канала типа В. Процедуру установления соединения по протоколу Q.931 иллюстрирует рис. 16.7.

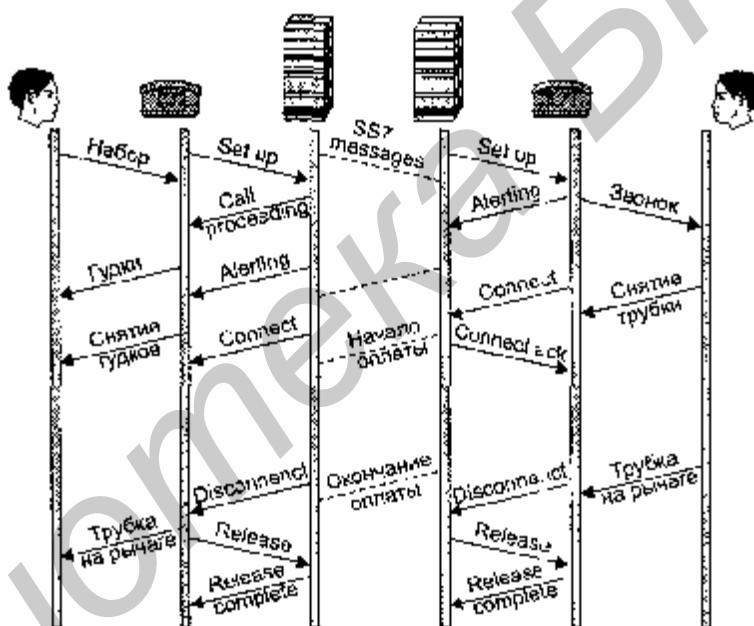


Рис. 16.7. Базовая процедура выполнения вызова в телевизионной сети по протоколу Q.931

Использование служб ISDN для передачи данных

Несмотря на значительные отличия от аналоговых телефонных сетей, сети ISDN сегодня используются и основном так же, как аналоговые телефонные сети, то есть как сети с коммутацией каналов, но только более скоростные: интерфейс BRI дает возможность установить дуплексный режим обмена со скоростью 128 кбит/с (логическое объединение двух каналов типа Б), а интерфейс PRI — 2,048 Мбит/с. Кроме того, качество цифровых каналов гораздо выше, чем аналоговых. Это значит, что процент искаженных кадров будет гораздо ниже, а полезная скорость обмена данными существенно выше.

Обычно интерфейс BRI используется в коммуникационном оборудовании для

Обычно интерфейс BRI используется в коммуникационном оборудовании для

подключения отдельных компьютеров или небольших локальных сетей, а интерфейс PR1 — в маршрутизаторах, рассчитанных на сети средних размеров. Что же касается объединения компьютерных сетей для поддержки службы с коммутацией пакетов, то здесь возможности сетей ISDN не слишком велики.

На каналах типа B режим коммутации пакетов поддерживается путем постоянного или коммутируемого соединения с коммутатором сети X.25. То есть каналы типа B в сетях ISDN являются только транзитными для доступа к «настоящей» сети X.25. Собственно, это сводится к первому случаю использования сети ISDN — только как сети с коммутацией каналов. Развитие технологии трансляции кадров на каналах типа B — технологии frame relay — привело к тому, что сети frame relay стали самостоятельным видом сетей со своей инфраструктурой каналов и коммутаторов. Поэтому эта технология рассматривается в главе 17, посвященной сетям с коммутацией пакетов.

Остается служба коммутации пакетов, доступная по каналу D. Так как после передачи адресной информации канал D остается свободным, по нему можно реализовать передачу компьютерных пакетов X.25, поскольку протокол LAP-D позволяет это делать. Чаще всего сеть ISDN используется не как замена сети X.25, а как разветвленная сеть доступа к ней, как к менее географически распространенной и узкоспециализированной (рис. 16.8). Такая услуга обычно называется «доступ к сети X.25 через канал типа D». Скорость доступа к сети X.25 по каналу типа D обычно не превышает 9600 бит/с.

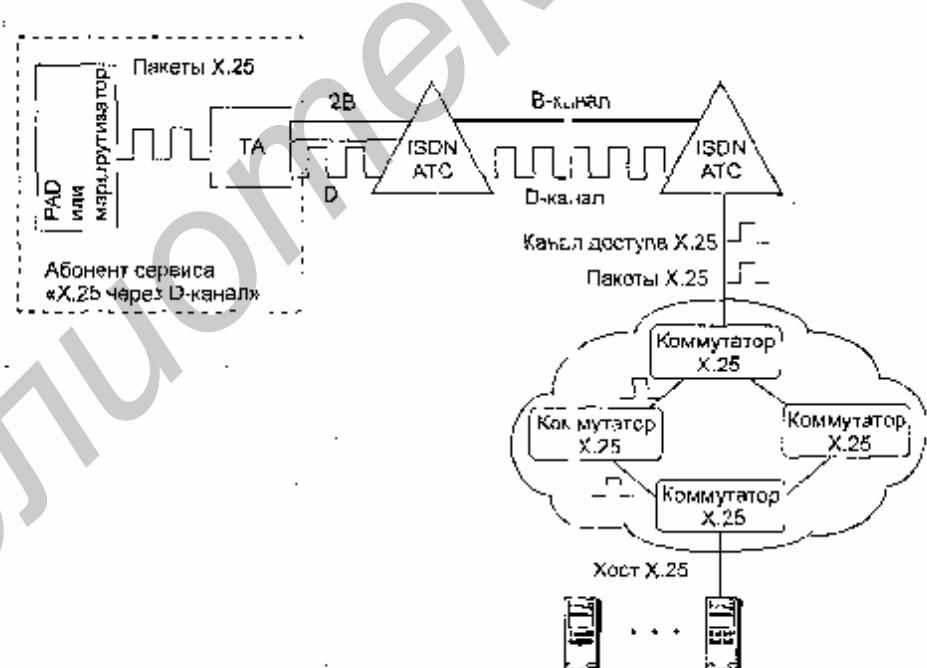


Рис. 16.8. Доступ к сети X.25 через канал D сети ISDN

Сети ISDN не рассматриваются разработчиками сетей передачи данных как хорошее средство для создания магистрали. Основная причина — отсутствие скоростной службы коммутации пакетов и невысокие скорости каналов, предоставляемых конечным пользователям. Для целей же подключения мобильных и

домашних пользователей, небольших филиалов и образования резервных каналов связи сети ISDN сейчас используются очень широко, естественно там, где они существуют. Производители коммуникационного оборудования выпускают широкий спектр продуктов для подключения локальных сетей к ISDN — терминальных адаптеров, удаленных мостов и офисных маршрутизаторов невысокой стоимости.

Выводы

- Цифровые сети с интегральными услугами (ISDN) разработаны для объединения в одной сети различных транспортных и прикладных служб. ISDN предоставляет своим абонентам услуги выделенных каналов, коммутируемых каналов, а также коммутации пакетов и кадров (frame relay).
- Интерфейс UNI предоставляется пользователям ISDN в двух видах — BRI и PRI. Интерфейс BRI предназначен для массового пользователя и построен по схеме 2B+D. Интерфейс PRI имеет две разновидности — североамериканскую 23B+D и европейскую 30B+D.
- Каналы типа D образуют сеть с коммутацией пакетов, выполняющую двоякую роль в сети ISDN: во-первых, передачу запроса на установление коммутируемого канала типа В с другим абонентом сети, во-вторых, обмен пакетами X.25 с абонентами сети ISDN или внешней сети X.25, соединенной с сетью ISDN.
- Цифровое абонентское окончание DSL сети ISDN для интерфейса BRI представляет собой 2-проводной кабель с максимальной длиной 5,5 км.
- Построение глобальных связей на основе сетей ISDN в корпоративной сети ограничено в основном организацией удаленного доступа и объединением небольших локальных сетей на основании службы коммутации каналов. Служба коммутации пакетов по каналу типа D реализуется редко — это связано с его невысокой скоростью, которая обычно составляет не более 9600 бит/с. Поэтому сети ISDN используются так же, как и аналоговые телефонные сети, только более скоростные и надежные.

ГЛАВА 17 Глобальные сети с коммутацией пакетов

Многие общие принципы и механизмы работы телекоммуникационных сетей, которые были достаточно подробно рассмотрены в предыдущих главах книги, применимы и к глобальным сетям. Это в первую очередь принцип коммутации пакетов, методы управления потоком и надежной доставки пакетов, построение в виде вторичной сети над инфраструктурой каналов первичной сети PDH/SDH/DWDM, использование при построении составных сетей вообще и сетей ТР в частности. Также были описаны структура сети оператора связи, являющейся очень распространенным типом глобальной сети, и набор оказываемых услуг. Поэтому читатель уже имеет представление о том, как работает глобальная вычислительная сеть и как она устроена.

В то же время глобальные сети имеют достаточно много отличий от локальных сетей. Эти отличия касаются как принципом работы (например, принципы маршрутизации почти во всех типах глобальных сетей, кроме глобальных сетей TCP/IP, основаны на предварительном образовании виртуального канала), так и предъявляемыми к сети требованиями. Имеется также специфика в организации составных глобальных сетей на основе протокола IP, как при их работе непосредственно через выделенные каналы («чистые» сети IP), так и при работе поверх сетей Frame Relay и ATM (оверлейные или наложенные сети IP).

Архитектура и терминология

В идеале сегодняшняя глобальная сеть с коммутацией пакетом должна передавать трафик абонентов и абонентского оборудования любых типов — к этому ведет процесс конвергенции сетей всех типов. То есть глобальная сеть должна предоставлять комплексные услуги: передачу пакетов локальных сетей, передачу пакетов мини-компьютеров и мэйнфреймов, обмен факсами, передачу трафика офисных АТС, выход в городские, междугородные и международные телефонные сети, обмен видеоизображениями для организации видеоконференций, передачу трафика кассовых аппаратов, банкоматов и т. д., и т. п. Основные типы потенциальных потребителей услуг глобальной сети с коммутацией пакетов иллюстрирует рис. 17.1.

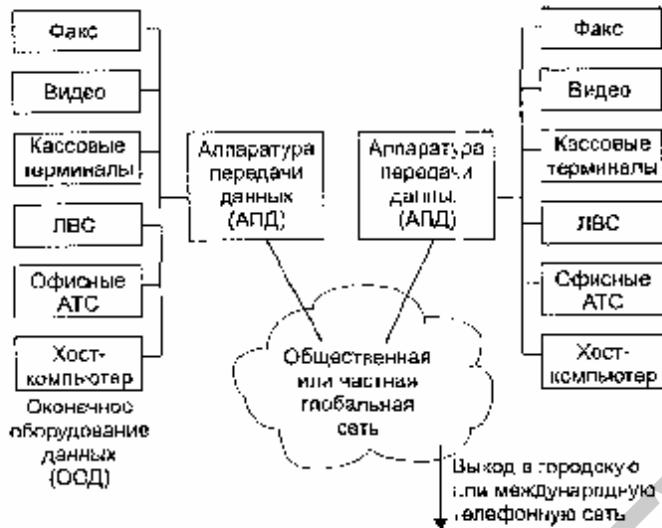


Рис. 17.1. Абоненты глобальной сети

Нужно подчеркнуть, что когда идет речь о передаче трафика офисных АТС или о взаимодействии с городскими, междугородными и международными телефонными сетями, то имеется в виду, что этот трафик передается по рассматриваемой глобальной сети в виде пакетов, то есть речь идет о пакетной телефонии. Большинство территориальных сетей в настоящее время обеспечивают только передачу компьютерных данных (и частности — трафик Интернета), но количество сетей, которые могут передавать остальные типы данных, постоянно растет. Очевидно, что для одновременной передачи пульсирующего и потокового трафиков, порождаемых различными приложениями, глобальная сеть должна поддерживать развитые методы обеспечения качества обслуживания (QoS) и инжиниринга трафика (TE), рассмотренные выше,

ПРИМЕЧАНИЕ

Отметим, что термин «передача данных» в территориальных сетях используется в узком смысле и означает передачу только компьютерных данных, а передачу речи и изображения обычно к передаче данных не относят.

Архитектура глобальной сети в наиболее общем виде представлена на рис. 17.2. Здесь используются следующие обозначения; S (switch) — коммутатор, K — компьютер, R (router) — маршрутизатор, MUX (multiplexor) — мультиплексор, UNI (User-to-Network Interface) — интерфейс пользователь-сеть и NNI (Network-to-Network Interface) — интерфейс сеть-сеть. Кроме того, офисная АТС обозначена аббревиатурой РВХ, а маленькими черными квадратиками — устройства DCE, о которых будет рассказано ниже.

Сеть строится на основе некоммутируемых (выделенных) каналов связи, которые соединяют коммутаторы глобальной сети между собой. На рис. 17.2 не показана лежащая под сетью с коммутацией пакетов первичная сеть PDH или SDH, в инфраструктуре которой собственно и формируются каналы связи между определенными географическими пунктами. В редких случаях такая первичная сеть отсутствует, и каналы связи являются просто кабельными линиями связи без каких-либо кросс-коннекторов, обеспечивающих гибкость прокладки

каналов.

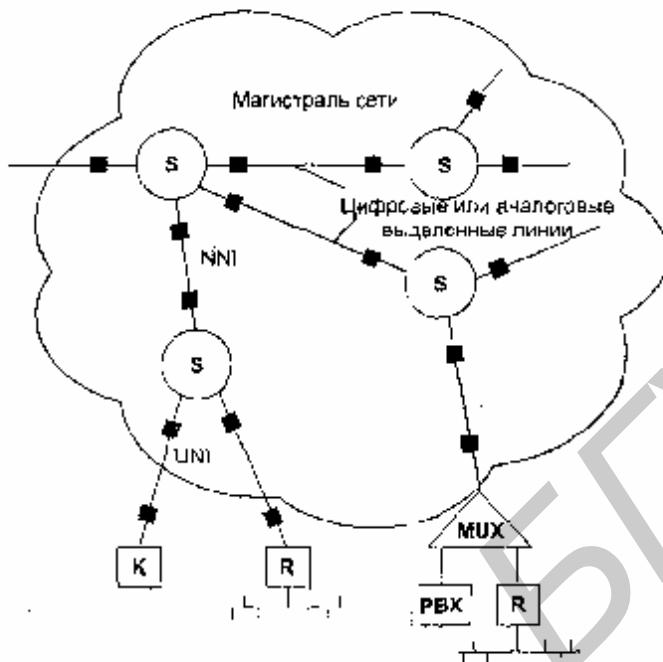


Рис. 17.2. Архитектура глобальной сети с коммутацией пакетов

Коммутаторы в русскоязычной литературе называют также центрами коммутации пакетов (ЦКП), они являются коммутаторами пакетов, которые в разных технологиях глобальных сетей могут иметь и другие названия — кадры, ячейки (cell). Как и в технологиях локальных сетей, принципиальной разницы между этими единицами данных нет, однако в некоторых технологиях есть традиционные названия, которые к тому же часто отражают специфику обработки пакетов. Например, кадр технологии frame relay редко называют пакетом, поскольку он не инкапсулируется в кадр или пакет более низкого уровня и обрабатывается протоколом канального уровня. В случае обработки трафика протоколов сетевого уровня термин «коммутатор» применяется реже, его заменяет термин «маршрутизатор».

Коммутаторы устанавливаются в тех географических пунктах, в которых требуется ответвление или слияние потоков данных конечных абонентов или магистральных каналов, переносящих данные многих абонентов. Естественно, выбор мест расположения коммутаторов определяется многими соображениями, в которые включается также возможность обслуживания коммутаторов квалифицированным персоналом, наличие выделенных каналов связи в данном пункте, надежность сети, определяемая избыточными связями между коммутаторами.

Абоненты сети подключаются к коммутаторам в общем случае также с помощью выделенных каналов связи. Эти каналы связи имеют более низкую пропускную способность, чем магистральные каналы, объединяющие коммутаторы, иначе сеть бы не справилась с потоками данных споих многочисленных поль-

зователей. Для подключения конечных пользователей допускается использование коммутируемых каналов, то есть каналов телефонных сетей, хотя в таком случае качество транспортных услуг обычно ухудшается. Принципиально замена выделенного канала коммутируемым ничего не меняет, но вносятся дополнительные задержки, отказы и разрывы канала по вине сети с коммутацией каналов, которая в таком случае становится промежуточным звеном между пользователем и сетью с коммутацией пакетов. Кроме того, в аналоговых телефонных сетях канал обычно имеет низкое качество из-за высокого уровня шумов. Применение коммутируемых каналов на магистральных связях коммутатор-коммутатор также возможно, но по тем же причинам весьма нежелательно. Каналы связи глобальных сетей организуются первичными сетями технологий FDM, РОН или SDH. Аппаратура мультиплексирования и коммутации этих сетей соединяется кабельными или беспроводными линиями связи. В глобальных сетях используются те же основные типы кабелей, что и в локальных: на витых медных парах, медный коаксиальный и волоконно-оптические (одномодовые). В каждой стране действуют свои национальные стандарты на кабели для телекоммуникационных сетей.

В глобальной сети наличие большого количества абонентов с невысоким средним уровнем трафика весьма желательно — именно в этом случае начинают в наибольшей степени проявляться выгоды метода коммутации пакетов. Если же абонентов мало и каждый из них создаст трафик большой интенсивности (по сравнению с возможностями каналов и коммутаторов сети), то равномерное распределение во времени пульсаций трафика становится маловероятным и для качественного обслуживания абонентов необходимо использовать сеть с низким коэффициентом нагрузки.

Конечные узлы глобальной сети более разнообразны, чем конечные узлы локальной сети. На рис. 17.2 показаны основные типы конечных узлов глобальной сети: отдельные компьютеры K, локальные сети, маршрутизаторы R и мультиплексоры MUX, которые используются для одновременной передачи по компьютерной сети данных и голоса (или изображения). Такие мультиплексоры также называют шлюзами, так как они транслируют весь стек протоколов, например стек протоколов традиционной телефонии в стек протоколов IP-сети с протоколами IP-телефонии на прикладном уровне.

Все эти устройства вырабатывают данные для передачи в глобальной сети, поэтому являются для нее оконечным оборудованием данных (ООД) или в англоязычной терминологии — устройствами Data Terminal Equipment (DTE). Например, устройством DTE является маршрутизатор, соединяющий локальную сеть с глобальной.

При передаче данных через глобальную сеть мосты и маршрутизаторы работают в соответствии с той же логикой, что и при соединении локальных сетей. Мосты, которые в этом случае называются удаленными мостами (*remote bridges*), строят таблицу MAC-адресов на основании проходящего через них трафика и по данным этой таблицы принимают решение — передавать кадры в удаленную сеть или нет. В отличие от своих локальных собратьев, удаленные мосты выпускаются и сегодня, привлекая сетевых интеграторов тем, что их не

нужно конфигурировать, а в удаленных офисах, где нет квалифицированного обслуживающего персонала, это свойство оказывается очень полезным. Маршрутизаторы принимают решение на основании номера сети пакета какого-либо протокола сетевого уровня (например, TCP или IPX) и, если пакет нужно переправить следующему маршрутизатору по глобальной сети, например frame relay, упаковывают его в кадр этой сети, снабжают соответствующим аппаратным адресом следующего маршрутизатора и отправляют в глобальную сеть. Мультиплексоры «голос-данные» предназначены для совмещения в рамках одной территориальной сети компьютерного и голосового трафиков. Так как рассматриваемая глобальная сеть передает данные в виде пакетов, то мультиплексоры «голос-данные», работающие на сети данного типа, упаковывают голосовую информацию в кадры или пакеты территориальной сети и передают их ближайшему коммутатору точно так же, как и любой конечный узел глобальной сети, то есть мост или маршрутизатор. Если глобальная сеть поддерживает приоритезацию трафика, то кадрам голосового трафика мультиплексор присваивает наивысший приоритет, чтобы коммутаторы обрабатывали и продвигали их в первую очередь. Приемный узел на другом конце глобальной сети также должен быть мультиплексором «голос-данные», который должен понять, что за тип данных находится в пакете — замеры голоса или пакеты компьютерных данных, — и отсортировать эти данные по своим выходам. Голосовые данные направляются в офисной АТС, а компьютерные данные поступают через маршрутизатор в локальную сеть. Часто модуль мультиплексора «голос-данные» встраивается в маршрутизатор. Так как конечные узлы глобальной сети должны передавать данные по каналу связи определенного стандарта, то каждое устройство типа DTE требуется оснастить устройством, носящим название аппаратура передачи данных (АПД), или Data Circuit Terminating Equipment (DCE). Такая аппаратура обеспечивает необходимый протокол физического уровня, данного канала. В зависимости от типа канала для связи с каналами глобальных сетей используются устройства DCE трех основных типов: модемы для работы по выделенным и коммутируемым аналоговым каналам, устройства DSU/CSU для работы по цифровым выделенным каналам сетей технологии TDM (PDH и SDH) и терминальные адаптеры (TA) для работы по цифровым каналам сетей ISDN. Устройства DTE и DCE обобщенно называются оборудованием, размещаемым на территории абонента глобальной сети (Customer Premises Equipment, CPE).

Если предприятие не строит свою территориальную сеть, а пользуется услугами общественной, то внутренняя структура этой сети его не интересует. Для абонента общественной сети главное — это предоставляемые сетью услуги и четкое определение интерфейса взаимодействия с сетью, чтобы его окончное оборудование и программное обеспечение корректно сопрягались с соответствующим оборудованием и программным обеспечением общественной сети.

Поэтому в глобальной сети обычно строго описан и стандартизован интерфейс «пользователь-сеть» (User-to-Network Interface, UNI). Это необходимо для того, чтобы пользователи могли без проблем подключаться к сети с помощью коммуникационного оборудования любого производителя, который соблюдает

стандарт UNI данной технологии (например, X.25).

Протоколы взаимодействия коммутаторов внутри глобальной сети, называемые интерфейсом «сеть-сеть» (Network-to-Network Interface, AW/), стандартизуются не всегда. Считается, что организация, создающая глобальную сеть, должна иметь свободу действий, чтобы самостоятельно решать, как должны взаимодействовать внутренние узлы сети между собой. В связи с этим внутренний интерфейс, в случае его стандартизации, носит название «сеть-сеть», а не «коммутатор-коммутатор», подчеркивая тот факт, что он должен использоваться в основном при взаимодействии двух территориальных сетей различных операторов. Тем не менее если стандарт NNI принимается, то и соответствие с ним обычно организуется взаимодействие всех коммутаторов сети, а не только пограничных.

Выводы

- Глобальные компьютерные сети (WAN) используются для объединения абонентов разных типов: отдельных компьютеров разных классов — от мэйнфреймов до персональных компьютеров, локальных компьютерных сетей, удаленных терминалов.
- Ввиду большой стоимости инфраструктуры глобальной сети существует острая потребность передачи по одной сети всех типов трафика, которые возникают на предприятии, а не только компьютерного: голосового трафика внутренней телефонной сети, работающей на офисных АТС (PBX), трафика факс-аппаратов, видеокамер, кассовых аппаратов, банкоматов и другого производственного оборудования.
- Для поддержки мультимедийных видов трафика создаются специальные технологии: ISDN, B-ISDN. Кроме того, технологии глобальных сетей, которые разрабатывались для передачи исключительно компьютерного трафика, в последнее время адаптируются для передачи голоса и изображения. Для этого пакеты, переносящие замеры голоса или данные изображения, приоритезируются, а в тех технологиях, которые это допускают, для их переноса создается соединение с заранее резервируемой пропускной способностью. Имеются специальные устройства доступа — мультиплексоры «голос-данные» или «видеоданные», которые упаковывают мультимедийную информацию в пакеты и отправляют ее по сети, а на приемном конце распаковывают и преобразуют в исходную форму — голос или видеоизображение.
- Технологии глобальных сетей определяют два типа интерфейса: «пользователь-сеть» (UNI) и «сеть-сеть» (NNI). Интерфейс UNI всегда глубоко детализирован, чтобы обеспечивать подключение к сети оборудования доступа от разных производителей. Интерфейс NNI может быть детализирован не так подробно, так как взаимодействие крупных сетей может обеспечиваться на индивидуальной основе.

Арендуемые элементы глобальной сети

Достаточно часто для организации или предприятия по разным причинам — финансовым, организационным или иным — нецелесообразно строить глобальную сеть целиком на собственных элементах, то есть собственных кабельных линиях связи, канaloобразующей аппаратуре PDH/SDH и собственных коммутаторах или маршрутизаторах. В то же время очень часто операторы связи предоставляют такие услуги, как аренда элементов транспортной инфраструктуры, например аренда выделенных каналов. Поэтому при построении глобальной сети можно дополнить недостающие компоненты услугами и оборудованием, арендаемыми у владельцев первичной или телефонной сети.

В зависимости от того, какие компоненты приходится брать в аренду, принято различать глобальные сети, построенные с использованием:

- арендного кабеля;
- выделенных каналов;
- транспортных услуг сети с коммутацией каналов;
- транспортных услуг сети с коммутацией пакетов.

В первом случае арендуются только сам кабель (или часть жил кабеля), соединяющий некоторые географические пункты. Вся канaloобразующая аппаратура первичной сети, например мультиплексоры SDH, приобретаются и устанавливаются самостоятельно.

Во втором случае у некоторого оператора связи приобретается в аренду цифровой или аналоговый канал, также соединяющий два географических пункта. Однако в этом случае первичная сеть не является собственностью организации, создающей глобальную сеть с коммутацией пакетов. Вместо этого оператор связи в своей первичной сети конфигурирует полупостоянный канал, который существует только в период действия аренды. Арендаемый канал используется для соединения собственных коммутаторов или маршрутизаторов.

Если же по каким-то причинам услуги первичной сети оказываются недоступными или нецелесообразными, то можно использовать каналы телефонной сети, аналоговой сети или ISDN. Такие каналы могут быть как коммутируемыми, требующими динамического установления перед каждым использованием, так и полупостоянными, конфигурация которых не меняется длительное время. В последнем случае для связи локальных сетей и другого оборудования DTE на некоторой территории используется имеющаяся там сеть с коммутацией пакетов. А если так, то общая архитектура сети по-прежнему соответствует показанной на рис. 17.2, только некоторая часть этой сети теперь будет относиться к другой организации.

Выводы

- Глобальные компьютерные сети работают на основе технологии коммутации пакетов, кадров и ячеек. Чаще всего глобальная компьютерная сеть принадлежит телекоммуникационной компании, которая предоставляет службы своей сети в аренду. При отсутствии такой сети в нужном регионе предприятия само-

стоятельно создают глобальные сети, арендую выделенные или коммутируемые каналы у телекоммуникационных или телефонных компаний.

- На арендованных каналах можно строить сеть с промежуточной коммутацией на основе какой-либо технологии глобальной сети (X.25, frame relay, ATM) или же соединять арендованными каналами непосредственно маршрутизаторы или мосты локальных сетей. Выбор способа использования арендованных каналов зависит от количества и топологии связей между локальными сетями.
- Глобальные сети делятся на магистральные сети и сети доступа.

Подключение к постоянным и коммутируемым каналам

Вопросы подключения устройств DTE к каналам связи с помощью устройств DCE являются очень важными при построении любой глобальной сети. Эти вопросы возникают и при соединении с помощью каналов связи коммутаторов сети, и при подключении к ней через каналы оборудования абонентов. В данном разделе рассмотрены типы устройств DCE, используемых для подключения оборудования DTE к каналам различного типа: аналоговым и цифровым, выделенным и коммутируемым.

Выделенные аналоговые каналы

Постоянные аналоговые каналы предоставляются пользователю оператором телефонной сети с 4-проводным или 2-проводным окончанием. На каналах с 4-проводным окончанием организация полнодуплексной связи, естественно, выполняется проще.

Выделенные линии могут быть разделены на две группы по другому признаку — наличию промежуточной аппаратуры коммутации и усиления или ее отсутствию.

Первую группу составляют так называемые нагруженные линии, проходящие через оборудование частотного уплотнения (FDM-коммутаторы и мультиплексоры), расположенное, например, на АТС. Телефонные компании обычно предоставляют в аренду два типа выделенных каналов: канал тональной частоты с полосой пропускания 3,1 кГц и широкополосный канал с полосой 48 кГц, который представляет собой базовую группу из 12 каналов тональной частоты. Широкополосный канал имеет границы полосы пропускания от 60 до 108 кГц. Так как широкополосный канал используется для связи АТС между собой, то получение его в аренду более проблематично, чем канала тональной частоты.

Выделенные нагруженные каналы также классифицируются на категории в зависимости от их качества. От категории качества зависит и арендная месячная плата за канал.

Вторая группа выделенных линий — это пенагруженные физические проводные линии. Они могут кроссироваться, но при этом не проходят через аппаратуру частотного уплотнения. Часто такими линиями связываются близко стоящие здания. Разветвленные сети каналов, представляющих собой ненагружен-

ные линии, используются, например, муниципальными службами (энергонадзора, водопровода, пожарной охраны и др.) для передачи технологической информации. При небольшой длине неагруженной выделенной линии она обладает достаточно широкой полосой пропускания, иногда до 1 МГц, что позволяет передавать импульсные немодулированные сигналы. На первый взгляд может показаться, что ненагруженные линии не имеют отношения к глобальным сетям, так как их можно использовать при протяженности максимум несколько километров, иначе затухание становится слишком большим для передачи данных. Однако в последнее время именно этот вид выделенных каналов привлекает пристальное внимание разработчиков средств удаленного доступа. Дело в том, что телефонные абонентские окончания — отрезки витой пары от АТС до жилого или производственного здания — представляют собой именно такой вид каналов. Широкая (хотя и заранее точно неизвестная) полоса пропускания этих каналов позволяет достичь на коротком отрезке линии высокой скорости — до нескольких мегабит в секунду. В связи с этим до ближайшей АТС данные от удаленного компьютера или сети можно передавать гораздо быстрее, чем по каналам тональной частоты, которые начинаются в данной АТС.

Модемы для выделенных каналов

Для передачи данных по выделенным нагруженным аналоговым линиям используются модемы, работающие па основе методов аналоговой модуляции сигнала, рассмотренных в главе 2. Протоколы и стандарты модемов определены в рекомендациях CCITT серии V и делятся на три группы:

- стандарты, определяющие скорость передачи данных и метод кодирования; и стандарты исправления ошибок; а стандарты сжатия данных.

Эти стандарты определяют работу модемов как для выделенных, так и для коммутируемых линий. Модемы можно также классифицировать в зависимости от того, какой режимы работы они поддерживают, а также к какому окончанию (4-проводному или 2-проводному) они подключены.

В отношении режима работы модемы делятся на три группы:

- модемы, поддерживающие только асинхронный режим работы;
- модемы, поддерживающие асинхронный и синхронный режимы работы;
- модемы, поддерживающие только синхронный режим работы.

Модемы, работающие только в асинхронном режиме, обычно обеспечивают низкую скорость передачи данных — до 1200 бит/с. Так, модемы, работающие по стандарту V.23, могут обеспечивать скорость 1200 бит/с на 4-проводной выделенной линии в дуплексном асинхронном режиме, а по стандарту V.21 — на скорости 300 бит/с по 2-проводной выделенной линии также в дуплексном асинхронном режиме. Дуплексный режим на 2-проводном окончании обеспечивается частотным разделением канала. Асинхронные модемы представляют наиболее дешевый вид модемов, так как им не требуются высокоточные схемы синхронизации сигналов на кварцевых генераторах. Кроме того, асинхронный режим работы неприхотлив к качеству линии.

Модемы, работающие только в синхронном режиме, могут подключаться только к 4-проводному окончанию. Синхронные модемы используют для выделения сигнала высокоточные схемы синхронизации и поэтому обычно значительно

дороже асинхронных модемов. Кроме того, синхронный режим работы предъявляет высокие требования к качеству линии.

Для выделенного канала тональной частоты с 4-проводным окончанием разработано достаточно много стандартов серии V. Все они поддерживают дуплексный режим:

- V.26 — скорость передачи 2400 бит/с;
- V.27 — скорость передачи 4800 бит/с;
- V.29 — скорость передачи 9600 бит/с;
- V.32 ter — скорость передачи 19 200 бит/с.

Для выделенного широкополосного канала 60-108 кГц существуют три стандарта;

- V.35 — скорость передачи 48 кбит/с;
- V.36 — скорость передачи 48-72 кбит/с;
- V.37 — скорость передачи 96-168 кбит/с.

Коррекция ошибок в синхронном режиме работы обычно реализуется по протоколу HDLC, но допустимы и устаревшие протоколы SDLC и BSC компании IBM. Модемы стандартов V.35, V.36 и V.37 используют для связи с DTE интерфейс V.35.

Модемы, работающие в асинхронном и синхронном режимах, являются наиболее универсальными устройствами. Чаще всего они могут работать как по выделенным, так и по коммутируемым каналам, обеспечивая дуплексный режим работы. На выделенных каналах они поддерживают в основном 2-проводное окончание и гораздо реже — 1-проводное.

Для асинхронно-синхронных модемов разработан ряд стандартов серии V;

- V.22 — скорость передачи до 1200 бит/с;
- V.22 bis — скорость передачи до 2400 бит/с;
- V.26 ter — скорость передачи до 2400 бит/с;
- V.32 — скорость передачи до 9600 бит/с;
- V.32 bis — скорость передачи 14 400 бит/с; Q V.34 — скорость передачи до 28,8 кбит/с;
- V.34+ — скорость передачи до 33,6 кбит/с.

Стандарт V.34, принятый летом 1994 года, знаменует новый подход к передаче данных по каналу тональной частоты. Этот стандарт разрабатывался CCITT довольно долго — с 1990 года. Большой вклад в его разработку внесла компания Motorola, которая является одним из признанных лидеров этой отрасли. Стандарт V.34 разрабатывался для передачи информации по каналам практически любого качества. Особенностью стандарта являются процедуры динамической адаптации к изменениям характеристик канала во время обмена информацией. Адаптация осуществляется в ходе сеанса связи. — без прекращения и без разрыва установленного соединения.

Основное отличие V.34 от предшествующих стандартов заключается в том, что в нем определено 10 процедур, по которым модем после тестирования линии выбирает свои основные параметры: несущую и полосу пропускания (выбор проводится из 11 комбинаций), фильтры передатчика, оптимальный уровень передачи и др. Первоначальное соединение модемов проводится по стандарту

V.2I на минимальной скорости 300 бит/с, что позволяет работать на самых плохих линиях. Для кодирования данных используются избыточные коды квадратурной амплитудной модуляции QAM. Применение адаптивных процедур сразу позволило поднять скорость передачи данных более чем в два раза по сравнению с предыдущим стандартом — V.32 bis.

Принципы адаптивной настройки к параметрам линии были развиты в стандарте V.34+, который является усовершенствованным вариантом стандарта V.34. Стандарт V.34+ позволил несколько повысить скорость передачи данных за счет усовершенствования метода кодирования. Один передаваемый кодовый символ несет в новом стандарте в среднем не 8,4 бита, как в протоколе V.34, а 9,8. При максимальной скорости передачи кодовых символов в 3429 бод (это ограничение преодолеть нельзя, так как оно определяется полосой пропускания канала тональной частоты) усовершенствованный метод кодирования дает скорость передачи данных в 33,6 кбит/с ($3429 \times 9,8 = 33\,604$). Правда, специалисты отмечают, что даже в Америке только 30 % телефонных линий в состоянии обеспечить такой низкий уровень помех, чтобы модемы V.34+ смогли работать на максимальной скорости. Тем не менее модемы стандарта V.34+ имеют преимущества по сравнению с модемами V.34 даже на зашумленных линиях — они лучше «держат» связь, чем модемы V.34.

Протоколы V.34 и V.34+ позволяют работать на 2-проводной выделенной линии в дуплексном режиме. Дуплексный режим передачи в стандартах V.32, V.34, V.34+ обеспечивается не путем частотного разделения канала, а одновременной передачей данных в обоих направлениях. Принимаемый сигнал определяется вычитанием с помощью сигнальных процессоров (DSP) передаваемого сигнала из общего сигнала в канале. Для этой операции используются также процедуры эхо-подавления, так как передаваемый сигнал, отражаясь от ближнего и дальнего концов канала, вносит искажения в общий сигнал (метод передачи данных, описанный в проекте стандарта 802.3ab, определяющего работу технологии Gigabit Ethernet на питой паре категории 5, взял многое из стандартов V.32-V.34+).

На высокой скорости модемы V.32-V.34+ фактически всегда используют в канале связи синхронный режим. При этом они могут работать с DTE как по асинхронному интерфейсу, так и по синхронному. В первом случае модем преобразует асинхронные данные в синхронные.

Модемы различаются не только поддерживаемыми протоколами, но и определенной ориентацией па область применения. Различают профессиональные модемы, предназначенные для работы в модемных пулах корпоративных сетей, и модемы для применения в небольших офисах или на дому.

Профессиональные модемы отличаются высокой надежностью, способностью устойчиво работать в непрерывном режиме и поддержкой средств удаленного централизованного управления. Обычно система управления модемными стойками поставляется отдельно и оправдывает себя в условиях большого предприятия. Стандарт V.34 выделяет в общей полосе пропускания линии отдельную полосу для управления модемом по тому же каналу, но которому передаются и пользовательские данные.

Типовая структура соединения двух компьютеров или локальных сетей через маршрутизатор с помощью выделенной аналоговой линии приведена на рис. 17.3. В случае 2-проводного окончания (рис. 17.3, а) для обеспечения дуплексного режима модем использует трансформаторную развязку. Телефонная сеть благодаря своей схеме развязки обеспечивает разъединение потоков данных, циркулирующих в разных направлениях. При наличии 4-проводного окончания (рис. 17.3, б') схема модема упрощается.

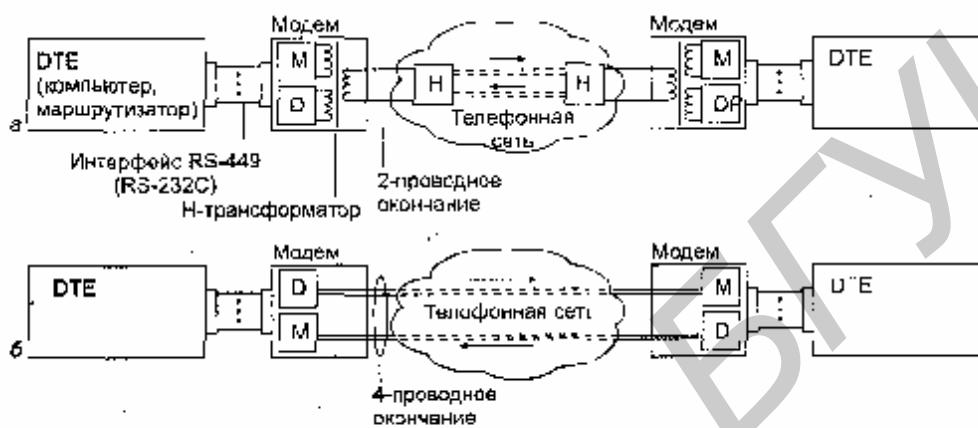


Рис. 17.3. Соединение локальных сетей или компьютеров по выделенному аналоговому каналу

Модемы для коммутируемых аналоговых линий

Для передачи данных по аналоговым коммутируемым телефонным каналам используются модемы, которые:

- поддерживают процедуру автозвона абонента;
- работают по 2-проводному окончанию, так как в телефонных сетях для коммутируемых каналов предусмотрено именно это окончание.

Чаще всего сегодня для коммутируемых каналов используются те же модели модемов, что и для выделенных, так как последние стандарты определяют два режима работы — по выделенным каналам и по коммутируемым. Естественно, такие комбинированные модели дороже моделей, поддерживающих только один режим работы — по коммутируемым каналам.

Для передачи данных по коммутируемым каналам комитет ССПТ разработал ряд основных стандартов, определяющих скорость и метод кодирования сигналов.

Стандарты первой группы являются основными и состоят из следующих спецификаций:

- V.21 — дуплексная асинхронная передача данных на скорости 300 бит/с;
- V.22 — дуплексная асинхронная/синхронная передача данных на скорости 1,2 кбит/с;
- V.22 bis — дуплексная асинхронная/синхронная передача данных на скоростях 1,2 и 2,4 кбит/с;

- V.26 ter — дуплексная асинхронная/синхронная передача данных на скоростях 1,2 и 2,4 кбит/с;
- V.32 — дуплексная асинхронная/синхронная передача данных на скоростях 4,8 и 9,6 кбит/с;
- V.32 bis — дуплексная асинхронная/синхронная передача на скорости до 1^,4 кбит/с;
- V.34 — дуплексная передача на скорости до 28,8 кбит/с;
- V.34+ — дуплексная передача на скорости до 33,6 кбит/с;
- V.90 — дуплексная передача на скоростях до 33,6/56 кбит/с.

На практике сегодня в основном применяют модемы, которые поддерживают стандарт V.90 и могут адаптироваться к качеству линии.

Для реализации функций автозвызова современные модемы предлагают несколько способов. При работе с модемом по асинхронному интерфейсу обычно используется система команд, предложенная компанией Mayes для своей модели Smart modem в начале 80-х годов. Каждая команда состоит из набора обычных символов, передаваемых модему в старт-стопном режиме. Например, для указания необходимости набора номера в импульсном режиме модему посыпается команда ATDP. Это можно сделать даже вручную, если модем подключен к обычному алфавитно-цифровому терминалу через интерфейс RS-232C.

Для синхронных интерфейсов между модемом и DTE используются два стандарта автонабора номера: V.25 и V.25bis. Стандарт V.25 требует, чтобы, помимо основного интерфейса для передачи данных, модем соединялся с DTE отдельным интерфейсом V.25/RS-366 на специальном 25-контактном разъеме. В стандарте V.25 bis для передачи команд автозвона предусмотрен тот же разъем, что и в основном интерфейсе, то есть RS-232C. Интерфейсы V.25 и V.25 bis могут работать не только в синхронном режиме с DTE, но и в асинхронном, но в основном характерны для синхронных интерфейсов, так как в асинхронном режиме для автозвона чаще используются Hayes-команды.

Для модемов, работающих с DTE по асинхронному интерфейсу, комитет CCITT разработал протокол коррекции ошибок V.42. Для принятия в модемах, работающих по асинхронному интерфейсу, коррекция ошибок обычно выполнялась по протоколам фирмы Microcom, еще одного лидера в области модемных технологий. Эта компания реализовала в своих модемах несколько разных процедур коррекции ошибок, назвав их протоколами MNP (Microcom Networking Protocol) классов 2-4.

В стандарте V.42 основным является другой протокол — протокол LAP-M (Link Access Protocol for Modems). Однако стандарт V.42 поддерживает и процедуры MNP 2-4, поэтому модемы, соответствующие рекомендации V.42, позволяют устанавливать связь без ошибок с любым модемом, поддерживающим этот стандарт, а также с любым MNP-совместимым модемом. Протокол LAP-M принадлежит семейству HDT.C и в основном работает так же, как и другие протоколы этого семейства — с установлением соединения, кадрированием данных, нумерацией кадров и восстановлением кадров с поддержкой метода скользящего окна. Основное отличие от других протоколов этого семейства — наличие кадров XID и BREAK. С помощью кадров XID (cXchange Identification) мо-

демы при установлении соединения могут договориться о некоторых параметрах протокола, например о максимальном размере поля данных кадра, о величине тайм-аута при ожидании квитанции, о размере окна и т. п. Эта процедура напоминает переговорные процедуры протокола PPP. Команда BREAK (BRK) служит для уведомления модема-напарника о том, что поток данных временно приостанавливается. При асинхронном интерфейсе с DTE такая ситуация может возникнуть. Команда BREAK посыпается в ненумерованном кадре, она не влияет на нумерацию потока кадров сеанса связи. После возобновления поступления данных modem возобновляет и отпранку кадров, как если бы паузы в работе не было.

Почти все современные модемы при работе по асинхронному интерфейсу поддерживают стандарты сжатия данных CCITT V.42bis и MNP-5 (обычно с коэффициентом 1:4, некоторые модели — до 1:8). Сжатие данных увеличивает пропускную способность линии связи. Передающий modem автоматически сжимает данные, а принимающий их восстанавливает. Модем, поддерживающий протокол сжатия, всегда пытается установить связь со сжатием данных, но если второй modem этот протокол не поддерживает, то и первый modem переходит на обычную связь без сжатия.

При работе модемов по синхронному интерфейсу наиболее популярным является протокол сжатия SDC (Synchronous Data Compression) компании Motorola. Новый modemный стандарт V.90 описывает технологию, направленную на обеспечение недорогого и быстрого способа доступа пользователей к сетям поставщиков услуг. Этот стандарт обеспечивает асимметричный обмен данными: со скоростью 56 кбит/с из сети и со скоростью 33,6 кбит/с в сеть. Стандарт совместим со стандартом V.34+.

Основная идея технологии асимметричных модемов состоит в следующем. В современных телефонных сетях часто единственным аналоговым звеном в соединении с сервером удаленного доступа является телефонная пара, связывающая modem компьютера с коммутатором телефонной станции. Этот канал оптимизирован для передачи речевых сигналов: максимальная скорость передачи данных определяется из условия предельно допустимого соотношения между шумами физической линии передачи и погрешностью дискретизации звукового сигнала при его оцифровывании. Эта величина задается стандартом V.34+ и равна 33,6 кбит/с.

Однако все вышеприведенные соображения справедливы только для одного направления передачи данных — от аналогового модема к телефонной станции. Именно на этом участке выполняется аналого-цифровое преобразование, которое вносит погрешность квантования. Эта погрешность добавляется к другим помехам линии и ограничивает скорость передачи значением 33,6 кбит/с. Обратное же цифро-аналоговое преобразование не вносит дополнительного шума, что делает возможным увеличение скорости передачи от телефонной станции к модему пользователя до 56 кбит/с.

Достоинством новой технологии является то, что для ее внедрения не требуется вносить какие-либо изменения в оборудование телефонной станции — нужно лишь изменить программу в цифровых модемах, установленных в стойках у

поставщика услуг, а также загрузить в пользовательский модем новую программу либо заменить микросхему памяти в зависимости от модели и производителя.

Технологии асимметричных модемов рассчитаны на то, что сервер удаленного доступа поставщика услуг корпоративной или публичной сети с коммутацией пакетов подключен к какой-либо АТС телефонной сети по цифровому интерфейсу, например BRI ISDN, или же по выделенному каналу T1/E1. Так что цифровой поток данных, идущий от сервера, постоянно пересыпается сетью в цифровой форме и только на абонентском окончании преобразуется в аналоговую форму. Если же сервер удаленного доступа подключен к телефонной сети по обычному аналоговому окончанию, то даже наличие модема V.90 у сервера не спасет положение — данные будут подвергаться аналого-цифровому преобразованию и их максимальная скорость не сможет превысить 33,6 кбит/с. При подключении же модемов V.90 к телефонной сети с обеих сторон обычным способом, то есть через аналоговые окончания, они работают как модемы V.34+. Та же картина будет наблюдаться в случае, если в телефонной сети на пути трафика встретится аналоговый коммутатор.

В последнем принятом на момент написания второй редакции книги стандарте модемов V.92 учитывается возможность принятия модемом второго вызова во время соединения. В таких случаях современные станции передают на телефонный аппарат специальные двойные тоновые сигналы, так что абонент может распознать эту ситуацию и, нажав на аппарате клавишу Falsh, переключиться на второе соединение, переведя первое соединение в режим удержания. Модемы предыдущих стандартов в таких случаях просто разрывают соединение, что не всегда удобно для абонента — может быть в этот момент он заканчивает «выкачивать» из Интернета большое письмо, и вся его работа «идет на смарку». Модемы V.92 отрабатывают такую ситуацию более интеллектуально, автоматически переключаясь на новое соединение, и, не разрывая предыдущего, переводят его в режим удержания, уведомляя об этом пользователя, который может вернуться к первому соединению.

Устройства DSU/CSU для подключения к выделенному каналу

Связь компьютера или маршрутизатора с цифровой выделенной линией PDH/SDH осуществляется с помощью пары устройств, обычно выполненных в одном корпусе или же совмещенных с маршрутизатором. Этими устройствами являются; устройство обслуживания данных (УОД) и устройство обслуживания канала (УОК). В англоязычной литературе эти устройства называются соответственно Data Service Unit (DSU) и Channel Service Unit (CSU). DSU преобразует сигналы, поступающие от DTE (обычно по интерфейсу RS-232C, RS-449 или V.35). DSU выполняет всю синхронизацию, формирует кадры капалон E1-E4 или же STM-N, усиливает сигнал и осуществляет выравнивание загрузки канала. Устройство CSU выполняет более узкие функции, в основном оно занимается созданием оптимальных условий передачи в линии. Эти устройства, как и модуляторы-демодуляторы, часто обозначаются одним словом DSU/CSU (рис.

17.4).

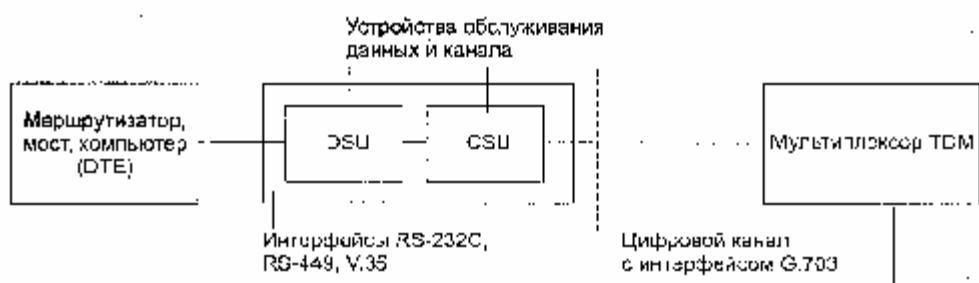


Рис. 17.4. Использование устройства DSU/CSU для подключения к цифровой выделенной линии

Нередко под устройствами DSU/CSU понимают более сложные устройства, которые кроме согласования интерфейсов выполняют функции мультиплексора T1/E1. В состав такого устройства может входить модуль мультиплексирования низкоскоростных потоков голоса и данных в канал 64 кбит/с или в несколько таких каналов (голос при этом обычно сжимается до скорости 8-16 кбит/с).

Устройства для подключения к коммутируемому каналу ISDN

Стандарт ISDN называет устройства для подключения к коммутируемому каналу ISDN терминальными адаптерами (Terminal Adapter, TA). Эти устройства преобразуют сигналы интерфейсов DTE-DCE, например интерфейса V.35, в сигналы интерфейса точки S/T стандарта ISDN.

Интерфейсы DTE-DCE

Для подключения устройств DCE к аппаратуре, вырабатывающей данные для глобальной сети, то есть описанным выше устройствам DTE (модемам, DSU/CSU и TA), существует несколько стандартных интерфейсов, которые представляют собой стандарты физического уровня. К этим стандартам относятся стандарты серии V CCITT, а также стандарты EIA серии RS (Recomended Standards). Две линии стандартов во многом дублируют одни и те же спецификации, но с некоторыми вариациями. Эти интерфейсы позволяют передавать данные со скоростями от 300 бит/с до нескольких мегабит в секунду на небольшие расстояния (15-20 м), достаточные для удобного размещения, например, маршрутизатора и модема.

Интерфейс RS-232C/V.24 является наиболее популярным низкоскоростным интерфейсом. Первоначально он был разработан для передачи данных между компьютером и модемом со скоростью не выше 9600 бит/с на расстояние до 15 м. Позднее практические реализации этого интерфейса стали работать и на более высоких скоростях — до 115 200 бит/с. Интерфейс поддерживает как асинхронный, так и синхронный режимы работы. Особую популярность этот интерфейс получил после его реализации в персональных компьютерах (его под-

держивают СОМ-порты), где он работает, как правило, только в асинхронном режиме и позволяет подключить к компьютеру не только коммуникационное устройство (такое, как модем), но и многие другие периферийные устройства — мышь, графопостроитель и т. д.

Интерфейс использует 25-контактный разъем или в упрощенном варианте — 9-контактный разъем (рис. 17.5).

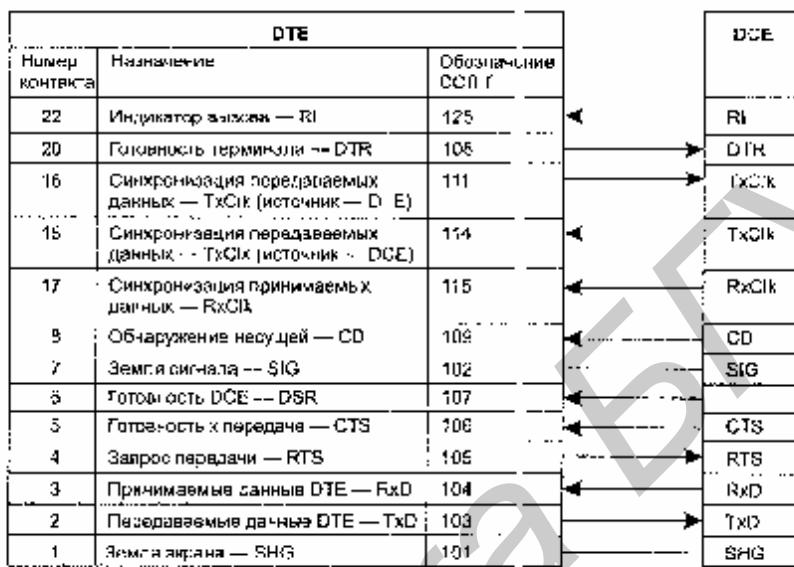


Рис. 17.5. Сигналы интерфейса RS-232C/V.24

Для обозначения сигнальных цепей используется нумерация CCITT, которая получила название «серия 100». Существуют также двухбуквенные обозначения EIA, которые на рисунке не показаны.

В интерфейсе реализован биполярный потенциальный код (+V, -V) на линиях между DTE и DCE. Обычно используется довольно высокий уровень сигнала: 12 или 15 В, чтобы более надежно распознавать сигнал на фоне шума.

При асинхронной передаче данных синхронизирующая информация содержится и самих кодах данных, поэтому сигнал синхронизации TxClk и RxClk отсутствуют. При синхронной передаче данных модем (DCE) передает на компьютер (DTE) сигналы синхронизации, без которых компьютер не может правильно интерпретировать потенциальный код, поступающий от модема по линии RxD. В случае когда используется код с несколькими состояниями (например, QAM), то один тактовый сигнал соответствует нескольким битам информации.

Нуль-модемный интерфейс характерен для прямой связи компьютеров на небольшом расстоянии с помощью интерфейса RS-232C/V.24. В этом случае необходимо применить специальный нуль-модемный кабель, так как каждый компьютер будет ожидать приема данных по линии RxD, что в случае применения модема будет корректно, но в случае прямого соединения компьютеров — нет. Кроме того, нуль-модемы и кабель должен имитировать процесс соединения и разрыва через модемы, и котором используется несколько линий (RI, СВ и т. д.). Поэтому для нормальной работы двух непосредственно соединенных компьютеров нуль-модемный кабель должен выполнять следующие со-

единения:

- RI+DSR-1 - DTR-2;
- DTR-1 - RI-2+DSR-2;
- CD-1 - CTS-2+RTS-2;
- CTS-1+RTS-1 - CD-2;
- RxD-1 - TxD-2;
- TxD-1 - RxD-2;
- SIG-1 - SIG-1;
- SHG-1 - SHG-2.

Знак «+» обозначает соединение соответствующих контактов на одной стороне кабеля.

Иногда при изготовлении нуль-модемного кабеля ограничиваются только перекрестным соединением линий приемника RxD и передатчика TxD, что для некоторого программного обеспечения бывает достаточно, но к общему случае может привести к некорректной работе программ, рассчитанных на реальные модемы,

Интерфейс RS-449/V.10/V.11 поддерживает более высокую скорость обмена данными и большую удаленность DCE от DTE. Этот интерфейс имеет две отдельные спецификации электрических сигналов. Спецификация RS-423/V.10 (аналогичные параметры имеет спецификация X.26) поддерживает скорость обмена до 100 000 бит/с на расстоянии до 10 м и скорость до 10 000 бит/с на расстоянии до 100 м. Спецификация RS-422/V.11(X.27) поддерживает скорость до 10 Мбит/с на расстоянии до 10 м и скорость до 1 Мбит/с на расстоянии до 100 м. Как и RS-232C, интерфейс RS-449 поддерживает асинхронный и синхронный режимы обмена между DTE и DCE. Для соединения используется 37-контактный разъем.

Интерфейс V.35 был разработан для подключения синхронных модемов. Он обеспечивает только синхронный режим обмена между DTE и DCE на скорости до 168 кбит/с. Для синхронизации обмена используются специальные тактирующие линии. Максимальное расстояние между DTE и DCE не превышает 15 м, как и в интерфейсе RS-232C.

Интерфейс X.21 разработан для синхронного обмена данными между DTE и OCE в сетях с коммутацией пакетов X.25. Это достаточно сложный интерфейс, который поддерживает процедуры установления соединения в сетях с коммутацией пакетов и каналов. Интерфейс был рассчитан на цифровые устройства OCE. Для поддержки синхронных модемов была разработана версия интерфейса X.21 bis, которая имеет несколько вариантов спецификации электрических сигналов: RS-232C, V.10, V.11 и V.35.

Интерфейс «токовая петля 20 мА» используется для увеличения расстояния между DTE и DCE. Сигналом является не потенциал, а ток величиной 20 мА, протекающий в замкнутом контуре передатчика и приемника. Дуплексный обмен реализован на двух токовых петлях. Интерфейс работает только в асинхронном режиме. Расстояние между DTE и OCE может составлять несколько километров, а скорость передачи — до 20 кбит/с. /

Интерфейс HSSI (High-Speed Serial Interface) разработан для подключения к

устройствам ОСЕ, работающим на высокоскоростные каналы, такие как каналы Т3 (45 Мбит/с), SQNET OC-1 (52 Мбит/с). Интерфейс работает и синхронном режиме и поддерживает передачу данных в диапазоне скоростей от 300 кбит/с до 52 Мбит/с.

Выводы

- Модемы для работы по коммутируемым аналоговым телефонным каналам должны поддерживать функцию автозвона удаленного абонента. При асинхронном интерфейсе модем использует для этого команды Hayes-совместимых модемов, а при синхронном интерфейсе — стандарт V.25 или V.25 bis.
- Основные стандарты модемов для коммутируемых каналов тональной частоты — это стандарты V.34+, V.90, V.42 и V.42 bis. Стандарт V.34 + является общим стандартом для работы по выделенным и коммутируемым каналам при 2-проводном окончании. Стандарт V.42 определяет протокол коррекции ошибок ЕЛР-М из семейства HDLC, а стандарт V.42 bis — метод сжатия данных при асинхронном интерфейсе. В синхронном интерфейсе для коррекции ошибок используется протокол HDLC, а для сжатия — фирменный протокол SDC компании Motorola.
- Стандарт V.90 полезен в том случае, когда между модемом пользователя и сервером удаленного доступа поставщика услуг все АТС обеспечивают цифровые методы коммутации, а сервер подключен по цифровому абонентскому окончанию. В этом случае благодаря отсутствию аналогово-цифрового преобразования на направлении от сервера к пользователю скорость передачи данных повышается до 56 кбит/с.
- Все устройства, используемые для подключения абонентов к глобальной сети, делятся на два класса: DTE, собственно вырабатывающие данные, и ОСЕ, служащие для передачи данных в соответствии с требованиями интерфейса глобального канала и завершающие канал.
- Интерфейсы DCE-DTE стандартизованы. Они могут работать как и синхронном, так и в асинхронном режимах. Наиболее популярными являются интерфейсы RS-232, RS-422, RS-530, V.35, HSSI.

Техника виртуальных каналов и дейтаграммные сети

Принцип работы

Практически все технологии пакетных сетей, специально разработанных для применения в глобальном масштабе, приводят данные на основе техники виртуальных каналов. В главе 2 этот механизм был рассмотрен в общих чертах, теперь познакомимся с его деталями.

Существует два типа виртуальных каналов — коммутируемый виртуальный канал (Switched Virtual Circuit, SVC) и постоянный виртуальный канал (Permanent Virtual Circuit, PVC). При создании коммутируемого виртуального канала

коммутаторы сети настраиваются на передачу пакетов автоматически, по запросу конечного (иногда и промежуточного) узла сети, а создание постоянного виртуального канала происходит заранее, причем коммутаторы настраиваются вручную администратором сети, возможно, с привлечением централизованной системы управления сетью и некоторого служебного протокола (пока чаще всего — фирменного). Фактически при создании коммутируемого виртуального канала используется протокол сигнализации, подобный протоколам сигнализации телефонных сетей, рассмотренным в главе 15. Даже названия протоколов сигнализации пакетных сетей указывают на то, что в их основе лежит определенный протокол телефонной сети — протокол Q.933 сети frame relay и протокол Q.2931 ведут свое происхождение от протокола Q.931 сети ISDN.

Рассмотрим сначала процесс создания коммутируемого виртуального канала, то есть SVC.

Создание коммутируемого виртуального канала требует наличия в коммутаторах таблиц маршрутизации, аналогичных тем, которые используются дейтаграммными сетями, например сетями ТР. Пример такой таблицы приведен на рис. 17.6. Рисунок иллюстрирует процесс прокладки виртуального канала между узлами N1,A1 и N2,A2 через сеть, представленной здесь двумя коммутаторами R1 и R2. В таблице маршрутизации указывается адрес сети назначения (в примере для определенности используются 3-байтовые адреса подсетей и 2-байтовые адреса конечных узлов, на практике в глобальных сетях на основе виртуальных каналов часто применяются более длинные адреса стандарта ЕЛ 64).

Установление коммутируемого виртуального канала выполняется некоторым служебным протоколом, выполняющим роль протокола сигнализации. Установление виртуального канала начинается с того, что узел-инициатор N1,A1 генерирует специальный пакет — запрос на установление логического соединения с узлом N2,A2. В нашем обобщенном примере этот запрос назван Call Setup (такое же название он носит и в некоторых реальных протоколах сигнализации, например в Q.933 и Q.2931). Это запрос содержит многоразрядный адрес узла назначения (в примере это 132456.8112, где старшая часть — номер подсети, а младшая — номер узла), а также начальное значение идентификатора виртуального канала (Virtual Channel Identifier, VC1) — 102.

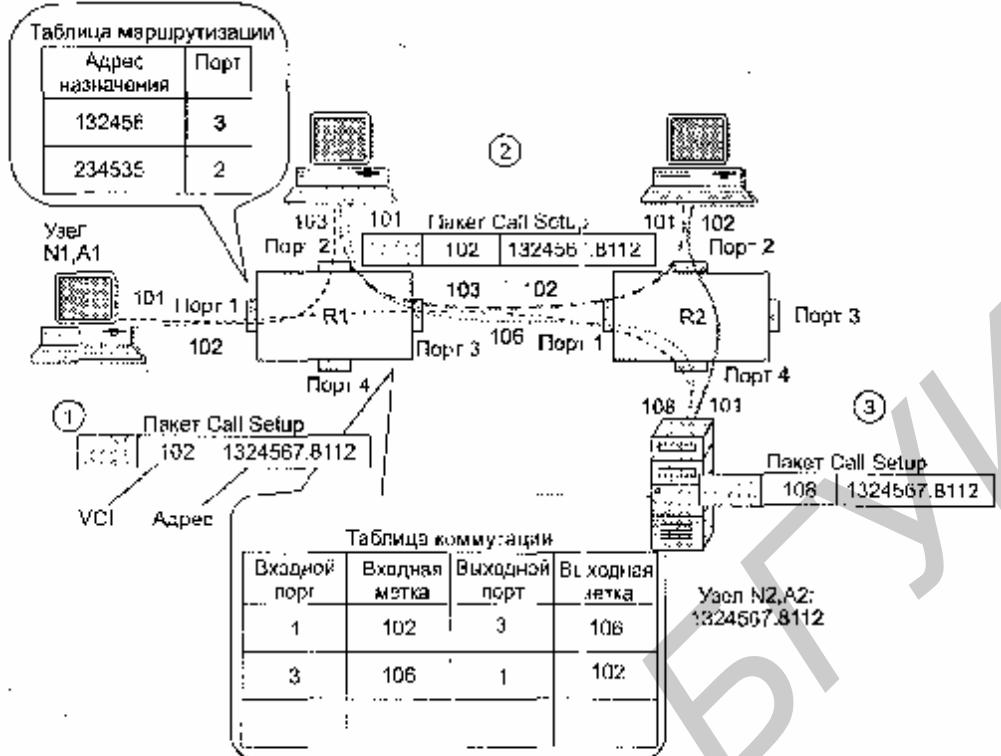


Рис. 17.8. Установление виртуального канала

Узел-инициатор должен выбрать коммутатор сети, которому целесообразно передать запрос на установление канала. Такой выбор может происходить на основе таблицы маршрутизации узла-отправителя, но если узел соединен с сетью единственным портом, как в приведенном примере, то таблица маршрутизации узлу, естественно, не требуется.

Присвоенный виртуальному каналу номер 102 имеет локальное значение для порта компьютера, через который устанавливается соединение. Так как через порт уже проходит виртуальный канал с номером 101, то программное обеспечение протокола сигнализации, работающее на конечном узле, просто выбирало первый свободный (не используемый в данный момент на данном порту) номер из разрешенного диапазона. Такой подход гарантирует уникальную идентификацию виртуальных каналов в пределах каждого порта.

Попав в буфер порта 1 коммутатора R1, пакет Call Setup обрабатывается в соответствии со своим адресом назначения и значениями таблицы маршрутизации. Запись с адресом 132456 говорит, что пакет нужно передать па порт 3. Заметим, что в приведенной таблице маршрутизации нет информации об адресе следующего коммутатора — в отличие от таблиц сетей IP. Это связано с тем, что в глобальных сетях коммутаторы всегда связаны физическими каналами «точка-точка», не поддерживающими множественное подключение, типичное для технологий локальных сетей, поэтому номер выходного порта однозначно определяет следующий коммутатор.

После определения выходного порта коммутатор R1 генерирует новое значение номера виртуального канала, а именно 106. Это связано с тем, что на участке сети «порт 3 коммутатора R1 — порт 1 коммутатора R2» номер 102 уже ис-

пользуется — для другого виртуального канала, проходящего через эти порты. Первым свободным номером оказался 106, в пределах указанного локального участка сети он однозначно идентифицирует устанавливаемый виртуальный канал. Именно это обстоятельство имелось в виду, когда ранее отмечалось, что идентификаторы виртуальных каналов имеют локальный характер. После изменения значения идентификатора виртуального канала пакет Call Setup передается через выходной порт 3 в коммутатор R2.

Одновременно с продвижением пакета коммутатор создает таблицу коммутации. Эта таблица потребуется впоследствии, когда виртуальный канал будет реализован и по нему начнут передаваться пользовательские данные, причем уже без адресов узлов назначения.

Каждая запись таблицы коммутации состоит из четырех основных полей; номер входного порта, входная метка (идентификатор виртуального канала в поступающих на входной порт пакетах), номер выходного порта и выходная метка (идентификатор виртуального канала в передаваемых через выходной порт пакетах). Запись в таблице коммутации «1-102-3-106» означает, что все пакеты, которые поступят на порт 1 с идентификатором виртуального канала 102, будут продвигаться на порт 3, а в поле идентификатора виртуального канала появится новое значение — 106.

Виртуальные каналы могут быть односторонними и двунаправленными, в рассматриваемом примере создается двунаправленный канал, поэтому коммутатор создает еще одну запись в таблице коммутации — для продвижения пакетов в обратном направлении, от узла N2,A2 к узлу N1,A1. Эта запись является зеркальной по отношению к первой записи, так что пакет, идущий в обратном направлении, получит при выходе из порта 1 первоначальное значение метки, а именно 102. В результате узел N1,A1 правильно распознает принадлежность пришедшего пакета виртуальному каналу, несмотря на постоянные смены номеров в процессе путешествия пакета по сети.

Процедуру установления виртуального канала продолжает коммутатор R2: по адресу назначения, указанному в запросе, и по своей таблице маршрутизации (на рисунке она не показана) он определяет выходной порт и передает на него запрос, обновляя при этом поле идентификатора виртуального канала, а также формирует записи таблицы коммутации. В результате запрос Call Setup приходит в конечный узел со значением номера виртуального канала 108. Конечный узел, получив запрос, может его принять или отвергнуть (по своим внутренним причинам), сообщив о своем решении служебным пакетом Connect, который проходит по сети в обратном направлении, подтверждая коммутаторам и узлу-инициатору факт установления виртуального канала.

После получения подтверждения Connect конечные узлы могут начать пользоваться проложенным виртуальным каналом, посыпая по нему пользовательские данные. Отправляемые узлом N1,A1 ячейки продвигаются на основе значения идентификатора виртуального канала, который обычно имеет небольшую длину, например, в технологии X.25 он занимает всего полтора байта. Применение дей-таграммной техники потребовало бы резервирования 8-20 байт для адреса узла назначения (в зависимости от применяемой и сети разрядности адресов

конечных узлов), что снизило бы коэффициент использования ячейки до неприемлемого уровня.

Режим постоянных виртуальных каналов, то есть PVC, не позволяет узлам сети динамически прокладывать виртуальные каналы. Вместо этого администратор сети заранее создает таблицы коммутации вручную. Обычно для облегчения работы он использует ту или иную систему управления сетью, которой передаст данные о том, через какие узлы должен проходить виртуальный канал. Эта система затем взаимодействует с коммутаторами сети, автоматически выбирая нужные значения меток и создавая записи таблиц коммутации. Использование в системах управления фирменных протоколов конфигурирования порождает проблему — обычно автоматизировать прокладку PVC можно только в пределах части сети, работающей на оборудовании одного производителя, а «сшивать» части PVC приходится вручную. Очевидно, что при создании PVC таблицы маршрутизации становятся ненужными, так как путь выбирается администратором. Для того чтобы созданный постоянный виртуальный канал мог быть использован, администратор должен ввести в конечные узлы, для которых канал создавался, его номер — для каждого конца канала свой.

Но существуя, техника коммутируемых виртуальных каналов использует два различных режима работы сети. При прокладке канала запрос на установление соединения передается по сети с использованием стандартного режима маршрутизации с глобальными (для всей сети) адресами назначения и информацией о полной топологии сети. То есть протоколы прокладки виртуальных каналов работают на сетевом уровне модели OSI.

После установления соединения сеть начинает работать па основе локальных адресов и локальных таблиц продвижения, что позволяет отнести такой режим к канальному уровню модели OSI, а коммуникационные устройства — к классу коммутаторов (стандартное название для устройств этого уровня).

От общего к частному

Нетрудно заметить, что виртуальный канал является частным случаем логического соединения между узлами сети. И для лучшего понимания техники виртуальных каналов полезно рассмотреть общие свойства протоколов, опирающихся на механизм логических соединений (а таких протоколов в сетях используется немало, примерами могут служить TCP, Microsoft SMB, Novell SPX).

Сеть фиксирует начало и конец логического соединения. Процедура обработки определяется не для отдельного пакета, а для всего множества пакетов, передаваемых в рамках каждого соединения. Процесс обслуживания вновь поступившего пакета напрямую зависит от предыстории, например, при потере нескольких предыдущих пакетов может быть снижена скорость отправки следующих.

Для того чтобы реализовать дифференцированное обслуживание пакетов, принадлежащих разным соединениям, сеть должна, во-первых, идентифицировать каждое соединение, а во-вторых, запоминать (отслеживать) значения параметров процедуры обработки, характеризующие данное соединение. В качестве признаком, по которым обрабатывающие узлы могут распознавать принадлеж-

ность пакета к тому или иному соединению, часто выступает набор параметров, идентифицирующих пару — отправителя и получателя (адреса узлов, номера портов приложений и т. п.).

Знание «предыстории» обмена позволяет более рационально обрабатывать каждый вновь поступающий пакет. Параметры соединения могут быть использованы для разных целей. Так, например, динамически изменяемый параметр соединения — текущий размер окна (количество пакетов, которое в данный момент отправителю разрешено передать в сеть без получения от получателя подтверждения) делает возможным управление производительностью обмена. А нумерация пакетов и отслеживание номеров уже отправленных и принятых пакетов служат для повышения надежности передачи, позволяя в рамках соединения упорядочивать, отбрасывать дубли и повторять передачу потерянных пакетов. В случае когда соединения устанавливаются с целью защиты данных, его параметры могут задавать режим защиты (аутентификация, обеспечение целостности или шифрование), используемую функцию и ключи шифрования.

Параметры процедуры обработки могут быть как постоянными в течение всего соединения (например, максимальный размер пакета), так и переменными, динамически отражающими текущее состояние соединения (например, упомянутые выше номера пакетов или текущий размер окна). Когда отправитель и получатель фиксируют начало нового сеанса, они прежде всего «договариваются» о начальных значениях параметров процедуры обмена и только после этого начинают передачу собственно данных.

В число параметров соединений многих протоколов с установлением соединения не входят параметры, регламентирующие маршрут продвижения пакетов по сети. Так, например, в задачу протокола TCP вообще не входит выбор маршрута — этим занимается нижележащий дейтаграммный протокол IP, работающий на всех промежуточных узлах сети. С учетом состояния сети — работоспособности каналов и коммутаторов, значений длины очередей пакетов в соседних коммутаторах и т. п. — протокол IP может направлять пакеты, даже имеющие одни и те же адреса отправления и назначения, по разным маршрутам. Таким образом, все пакеты в рамках соединения TCP перемешаются по независимым друг от друга маршрутам.

Сравнение

В отличие от дейтаграммных протоколов протоколы с поддержкой виртуальных каналов требуют предварительного установления соединения, что вносит дополнительную задержку перед передачей данных. Эта задержка особенно сказывается при передаче небольшого объема данных, когда время установления виртуального канала может быть соизмеримым со временем передачи данных. Кроме того, дейтаграммный метод быстрее адаптируется к изменениям в сети. При отказе коммутатора или канала на пути виртуального канала соединение разрывается, и виртуальный канал нужно прокладывать заново по маршруту, который, естественно, обходит отказавшие участки сети.

Однако сравнивая эти два принципиально различных подхода, следует учесть,

что время, затраченное на установление виртуального канала, компенсируется последующей быстрой передачек всего потока пакетов. Маршрутизация пакетов в сети с поддержкой виртуальных каналов ускоряется за счет двух фактором. Первый состоит в том, что решение о продвижении пакета принимается быстрее из-за меньшего размера таблицы коммутации. Вторым фактором является уменьшение доли служебной информации в пакетах. Адреса конечных узлов в глобальных сетях обычно имеют достаточно большую длину — 14-15 десятичных цифр, которые занимают до 8 байт (в технологии ATM — 20 байт) в служебном поле пакета. Номер же виртуального канала обычно занимает 10-12 бит, так что накладные расходы па адресную часть существенно сокращаются, а значит, полезная скорость передачи данных возрастает.

Наиболее эффективным по критерию производительности является режим PVC. Значительную часть работы по маршрутизации пакетов администратор сети уже выполнил, вручную проложив виртуальные каналы, коммутаторам остается только быстро продвигать пакеты на основе готовых таблиц коммутации портов. Постоянный виртуальный канал подобен выделенному каналу в том смысле, что не требуется устанавливать или разрывать соединение, обмен пакетами по PVC может происходить в любой момент времени. Отличие PVC от выделенной линии состоит в том, что пользователь не имеет тех гарантий относительно действительной пропускной способности канала. Зато использование PVC обычно намного дешевле, чем аренда выделенной линии, так как пользователь делит пропускную способность сети с другими пользователями.

Техника виртуальных каналов имеет свои достоинства и недостатки по сравнению с техникой IP- или IPX-маршрутизации. Маршрутизация каждого пакета без предварительного установления соединения (ни IP, ни IPX не работают с установлением соединения) эффективна для кратковременных потоков данных. Кроме того, возможно распараллеливание трафика для повышения производительности сети при наличии параллельных путей в сети. Быстрее отрабатывается отказ маршрутизатора или канала связи, так как последующие пакеты просто идут по новому пути (здесь, правда, нужно учесть время установления новой конфигурации в таблицах маршрутизации). При использовании виртуальных каналов очень эффективно передаются через сеть долговременные потоки, но для кратковременных этот режим не очень подходит, так как на установление соединения обычно уходит много времени — даже коммутаторы технологии ATM, работающие на очень высоких скоростях, тратят на установление соединения по 5-10 мс каждый.

Постоянные виртуальные каналы выгодно использовать для передачи агрегированных потоков трафика, состоящих из большого количества индивидуальных потоков абонентов сети. В этом случае виртуальный канал прокладывается не между конечными абонентами, а между участком магистрали сети, на котором данный агрегированный поток существует, например, от одного граничного маршрутизатора сети оператора связи до другого. В силу закона больших чисел агрегированные потоки обладают высокой степенью устойчивости, так что для них нет смысла динамически создавать коммутируемые виртуальные каналы — лучше эффективно использовать постоянные, которые оказываются

при хорошем планировании (методами инжиниринга трафика) достаточно загруженными.

Выводы

- К технологиям глобальных сетей с коммутацией пакетов относятся сети X.25, frame relay, SMDS, ATM и IP. Все эти сети, кроме сетей IP, используют маршрутизацию пакетов, основанную на виртуальных каналах между конечными узлами сети.
- Сети IP занимают особое положение среди технологий глобальных сетей, так как они играют роль технологии объединения сетей любых типов, в том числе и сетей всех остальных глобальных технологий. Таким образом, сети IP относятся к более высокоуровневым технологиям, чем технологии собственно глобальных сетей.
- Техника виртуальных каналов заключается в разделении операций маршрутизации и коммутации пакетов. Первый пакет таких сетей содержит адрес вызываемого абонента и прокладывает виртуальный путь в сети, настраивая промежуточные коммутаторы. Остальные пакеты проходят по виртуальному каналу в режиме коммутации на основании номера виртуального канала, который является локальным адресом для каждого порта каждого коммутатора.
- Техника виртуальных каналов имеет преимущества и недостатки по сравнению с техникой маршрутизации каждого пакета, характерной для сетей IP или IPX. Преимуществами являются: ускоренная коммутация пакетов по номеру виртуального канала, а также сокращение адресной части пакета, а значит, и избыточности заголовка. К недостаткам следует отнести невозможность распараллеливания потока данных между двумя абонентами по параллельным путям, а также неэффективность установления виртуального пути для кратковременных потоков данных.

Сети X-25

Назначение и структура сетей X.25

Сети X.25 являются на сегодняшний день «старейшиной» применяемых пакетных сетей, хотя популярность их быстро падает. Долгое время сети X.25 были единственными доступными сетями с коммутацией пакетов коммерческого типа, в которых давались гарантии коэффициента готовности сети. Интернет также имеет долгую историю существования, но как коммерческая сеть он начал эксплуатироваться совсем недавно, поэтому для корпоративных пользователей выбора не было. Кроме того, сети X.25 хорошо работают на ненадежных линиях благодаря протоколам с установлением соединения и коррекцией ошибок на двух уровнях — канальном и сетевом.

Стандарт X.25 «Интерфейс между оконечным оборудованием данных и аппаратурой передачи данных для терминалов, работающих в пакетном режиме в се-

тях передачи данных общего пользования» был разработан комитетом CCITT в 1974 году и пересматривался несколько раз. Стандарт наилучшим образом подходит для передачи трафика низкой интенсивности, характерного для терминалов, и в меньшей степени соответствует более высоким требованиям трафика локальных сетей. Как видно из названия, стандарт не описывает внутреннее устройство сети X.25, а только определяет пользовательский интерфейс с сетью. Взаимодействие двух сетей X.25 определяет стандарт X.75.

Технология сетей X.25 имеет несколько существенных признаков, отличающих ее от других технологий.

- Наличие в структуре сети специального устройства — PAD (Packet Assembler Disassembler), предназначенного для сборки нескольких низкоскоростных старт-стопных потоков байтов от алфавитно-цифровых терминалов в пакеты, передаваемые по сети и направляемые компьютерам для обработки. Эти устройства имеют также русскоязычное название сборщик-разборщик пакетов, СРП.

- Наличие трехуровневого стека протоколов с использованием на канальном и сетевом уровнях протоколов с установлением соединения, управляющих потоками данных и исправляющих ошибки.

- Ориентация на однородные стеки транспортных протоколов во всех узлах сети — сетевой уровень рассчитан на работу только с одним протоколом канального уровня и не может подобно протоколу IP объединять разнородные сети.

Сеть X.25 состоит из коммутаторов (Switches, S), называемых также центрами коммутации пакетов (ЦКП), расположенных в различных географических точках и соединенных высокоскоростными выделенными каналами (рис. 17.7). Выделенные каналы могут быть как цифровыми, так и аналоговыми.

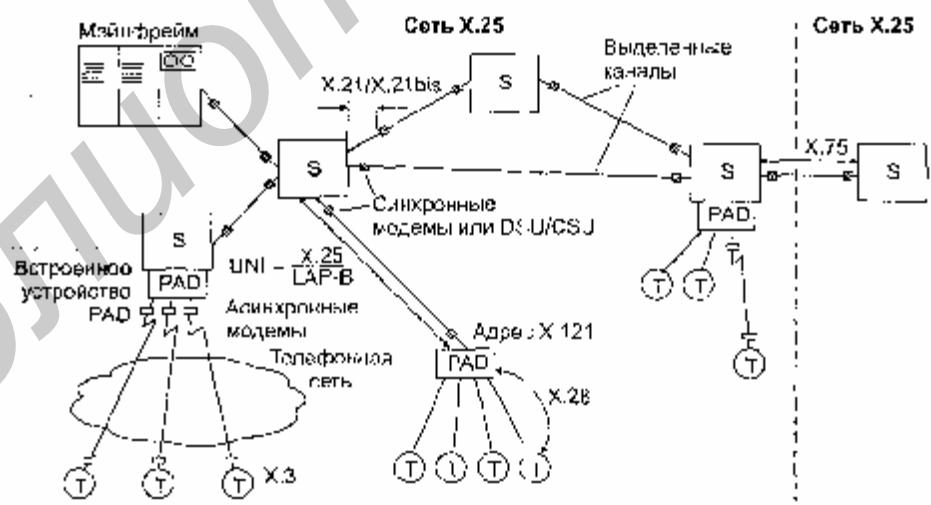


Рис. 17.7. Структура сети X.25

Асинхронные старт-стопные терминалы подключаются к сети через устройства PAD. Они могут быть встроенными или удаленными. Встроенное устройство PAD обычно расположено в стойке коммутатора. Терминалы получают доступ к встроенному устройству PAD по телефонной сети с помощью модемов с

асинхронным интерфейсом. Встроенное устройство PAD также подключается к телефонной сети с помощью нескольких модемов с асинхронным интерфейсом. Удаленное устройство PAD представляет собой небольшое автономное устройство, подключенное к коммутатору через выделенный канал связи X.25. К удаленному устройству PAD терминалы подключаются по асинхронному интерфейсу, обычно для этой цели используется интерфейс RS-232C. Одно устройство PAD обычно обеспечивает доступ для 8, 16 или 24 асинхронных терминалов.

К основным функциям PAD, определенным стандартом X.3, относятся:

- сборка символов, полученных от асинхронных терминалов, в пакеты;
- разборка полей данных в пакетах и вывод данных на асинхронные терминалы;
- управление процедурами установления соединения и разъединения по сети X.25 с нужным компьютером;
- передача символов по требованию асинхронного терминала, включая старт-стопные сигналы и биты проверки на четность;
- продвижение пакетов при наличии соответствующих условий, таких как заполнение пакета, истечение времени ожидания и др.

Терминалы не имеют конечных адресов сети X.25. Адрес присваивается порту PAD, который подключен к коммутатору пакетов X.25 с помощью выделенного канала.

Несмотря на то что задача подключения «неинтеллектуальных» терминалов к удаленным компьютерам возникает сейчас достаточно редко, функции PAD все еще остаются востребованными. Устройства PAD часто используются для подключения к сетям X.25 кассовых терминалов и банкоматов, имеющих асинхронный интерфейс RS-232.

Стандарт X.28 определяет параметры терминала, а также протокол взаимодействия терминала с устройством PAD. При работе на терминале пользователь сначала проводит некоторый текстовый диалог с устройством PAD, используя стандартный набор символьных команд. PAD может работать с терминалом в двух режимах: управляющем и передачи данных. В управляющем режиме пользователь с помощью команд может указать адрес компьютера, с которым нужно установить соединение по сети X.25, а также установить некоторые параметры работы PAD, например выбрать специальный символ для обозначения команды немедленной отправки пакета, установить режим эхо-ответов символов, набираемых на клавиатуре, от устройства PAD (при этом дисплей не будет отображать символы, набираемые на клавиатуре до тех пор, пока они не вернутся от PAD — это обычный локальный режим работы терминала с компьютером). При наборе комбинации клавиш Ctrl+P PAD переходит в режим передачи данных и воспринимает все последующие символы как данные, которые нужно передать в пакете X.25 узлу назначения.

В сущности, протоколы X.3 и X.28 определяют протокол эмуляции терминала, подобный протоколу telnet стека TCP/IP. Пользователь с помощью устройства PAD устанавливает соединение с нужным компьютером, а затем уже может вести диалог с операционной системой этого компьютера (в режиме передачи данных устройством PAD), запуская нужные программы и просматривая ре-

зультаты их работы на своем экране, как и при локальном подключении терминала к компьютеру.

Компьютеры и локальные сети обычно подключаются к сети X.25 непосредственно через адаптер X.25 или маршрутизатор, поддерживающий на своих интерфейсах протоколы X.25. Для управления устройствами PAD в сети существует протокол X.29, с помощью которого узел сети может управлять и конфигурировать PAD удаленно, по сети. При необходимости передачи данных компьютеры, подключенные к сети X.25 непосредственно, услугами PAD не пользуются, а самостоятельно устанавливают виртуальные каналы в сети и передают по ним данные в пакетах X.25.

Адресация в сетях X.25

Если сеть X.25 не связана с внешним миром, то она может использовать адрес любой длины (в пределах формата поля адреса) и давать адресам произвольные значения. Максимальная длина поля адреса в пакете X.25 составляет 16 байт.

Рекомендация X.121 CCITT определяет международную систему нумерации адресов для сетей передачи данных общего пользования. Если сеть X.25 хочет обмениваться данными с другими сетями X.25, то в ней нужно придерживаться адресации стандарта X.121.

Адреса X.121 (называемые также International Data Numbers, IDN) имеют разную длину, которая может доходить до 14 десятичных знаков. Первые четыре цифры IDN называют кодом идентификации сети (Data Network Identification Code, DNIC). Код DNIC поделен на две части; первая часть (3 цифры) определяет страну, в которой находится сеть, а вторая — номер сети X.25 в данной стране. Таким образом, внутри каждой страны можно организовать только 10 сетей X.25. Если же требуется перенумеровать больше, чем 10 сетей для одной страны, проблема решается тем, что одной странедается несколько кодов. Например, Россия имела до 1995 года один код — 250, а в 1995 году ей был выделен еще один код — 251. Остальные цифры называются номером национального терминала (National Terminal Number, NTN). Эти цифры позволяют идентифицировать определенное устройство DTE в сети X.25.

Международные сети X.25 могут также использовать международный стандарт нумерации абонентов ISO 7498, описанный в предыдущей главе.

По стандарту ISO 7498 для нумерации сетей X.25 к адресу в формате X.121 добавляется только один байт префикса, несущий код 36 (использование в адресе только кодов десятичных цифр) или 37 (использование произвольных двоичных комбинаций). Этот код позволяет универсальным коммутаторам, например коммутаторам сети ISDN, поддерживающим также и коммутацию пакетов X.25, автоматически распознавать тип адреса и правильно заполнять маршрутизацию запроса на установление соединения.

Стек протоколов сети X.25

Стандарты сетей X.25 описывают три уровня протоколов (рис. 17.8).

- На физическом уровне определены синхронные интерфейсы X.21 и X.21 his к оборудованию передачи данных — либо DSU/CSU, если выделенный канал является цифровым, либо к синхронному модему, если канал аналоговый.
- На канальном уровне используется подмножество протокола HDLC, обеспечивающее возможность автоматической передачи в случае возникновения ошибок в линии. Предусмотрен выбор из двух процедур доступа к каналу: LAP или LAP-B.
- На сетевом уровне определен протокол X.25/3 обмена пакетами между оконечным оборудованием и сетью передачи данных.

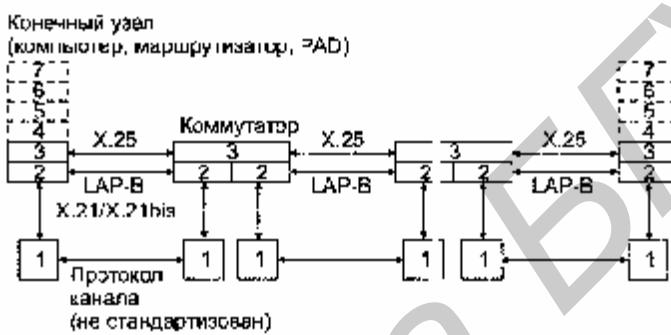


Рис. 17.8. Стэк протоколов сети X.25

Транспортный уровень может быть реализован в конечных узлах, но он стандартом не определяется.

Протокол физического уровня канала связи не оговорен, и это дает возможность использовать каналы разных стандартов.

На канальном уровне обычно используется протокол LAP-B. Этот протокол обеспечивает сбалансированный режим работы, то есть оба узла, участвующих в соединении, равноправны. По протоколу LAP-B устанавливается соединение между пользовательским оборудованием DTE (компьютером, IP- или IPX-маршрутизатором) и коммутатором сети. Хотя стандарт это и не оговаривает, но по протоколу LAP-B возможно также установление соединения па канальном уровне внутри сети между непосредственно связанными коммутаторами. Протокол LAP-B почти во всех отношениях идентичен протоколу LLC2, описанному в главе 7, кроме адресации. Кадр LAP-B содержит одно однобайтовое адресное поле (а не два — DSAP и SSAP), в котором указывается не адрес службы верхнего уровня, а направление передачи кадра — 0x01 для направления команд от DTE к OCE (в сеть) или ответов от OCE к OTE (из сети) и 0x03 для направления ответов от OTE к DCE или команд от BCE к OTE. Поддерживается как нормальный режим (с максимальным окном в 8 кадров и однобайтовым полем управления), так и расширенный режим (с максимальным окном в 128 кадров и двухбайтовым полем управления).

Сетевой уровень X.25/3 (в стандарте он назван не сетевым, а пакетным уровнем) реализуется с использованием 14 различных типов пакетов, по назначению аналогичных типам кадров протокола LAP-B. Так как надежную передачу

данных обеспечивает протокол LAP-B, протокол X.25/3 выполняет функции маршрутизации пакетов, установления и разрыва виртуального канала между конечными абонентами сети и управления потоком пакетов.

После установления соединения на канальном уровне конечный узел должен установить виртуальное соединение с другим конечным узлом сети. Для этого он в кадрах LAP-B посыпает пакет Call Request протокола X.25. Формат пакета Call Request показан на рис. 17.9. Этот пакет является пакетом сигнализации для сети X.25, которая отличается тем, что режим сигнализации в ней не выделен в отдельный протокол, а представляет собой один из режимов работы общего протокола сетевого уровня X.25/3.



Рис. 17.9. Формат пакета Call Request

Поля, расположенные в первых трех байтах заголовка пакета, используются во всех типах кадров протокола X.25. Признаки Q. и D и Modulo находятся в старшей части первого байта заголовка. Признак Q предназначен для распознавания на сетевом уровне типа информации в поле данных пакета. При получении пакета информация, расположенная в поле данных, а также значение бита Q передаются верхним уровням пользовательского стека протоколов (непосредственно транспортному уровню этого стека). Значение Q=1 означает управляющую пользовательскую информацию, а Q = 0 — данные. Признак D означает подтверждение приема пакета, узлом назначения. Обычный механизм подтверждения принятия пакетов с помощью квитанций имеет для протокола X.25 только локальный смысл — прием пакета подтверждает ближайший коммутатор сети, через который конечный узел запросил и установил виртуальное соединение. Если же узел-источник запросил подтверждение приема конечным узлом, то это подтверждение индицируется установкой бита D (delivery confirmation) в пакетах, идущих от узла назначения.

Признак Modulo идентифицирует модуль, по которому ведется нумерация пакетов (8 или 128). Значение 10 означает модуль 128, а 01 — модуль 8.

Поле LGN (Logical Group Number — номер логической группы) содержит значение номера логической группы виртуального канала. Каналы образуют логические группы по функциональному признаку, например:

- постоянный виртуальный канал;
- коммутируемый виртуальный канал только для входящих сообщений (симплексный канал);
- коммутируемый виртуальный канал только для исходящих сообщений (симплексный канал);
- коммутируемый дуплексный виртуальный канал.

Максимальное количество логических групп — 12, хотя в конкретной сети допустимо и меньшее количество.

Поле LCN (Logical Channel Number — номер логического канала) содержит номер виртуального канала, назначаемый узлом-источником (для коммутируемых виртуальных каналов) или администратором сети (для постоянных виртуальных каналов). Максимальное количество виртуальных каналов, проходящих через один порт, равно 256.

Поле Туре (тип) идентифицирует тип пакета. Например, для пакета Call Request отведено значение типа, равное 0x0B. Младший бит этого поля определяет, является ли пакет управляющим (бит равен 1) или пакетом данных (бит равен 0). Значение 0x0B содержит 1 в младшем бите, поэтому это управляющий пакет, а остальные биты в этом случае определяют подтип пакета. В пакете данных остальные биты поля Туре используются для переноса номеров квитанций N(S) и N(R).

Следующие два поля определяют длину адресов назначения и источника (DA и SA) в пакете. Запрос на установление виртуального канала содержит оба адреса. Первый адрес нужен для маршрутизации пакета Call Request, а второй — для принятия решения узлом назначения о возможности установления виртуального соединения с данным узлом-источником. Если узел назначения решает принять запрос, то он должен отправить пакет Call Accepted — «Запрос принят», в котором также указать оба адреса, поменяв их, естественно, местами. Адреса могут иметь произвольный формат или же соответствовать требованиям стандарта X.121 или ISO 7498.

Сами адреса назначения и источника занимают отведенное им количество байтов в следующих двух полях.

Поля FL (Facilities Length — длина поля услуг) и Facilities (услуги) нужны для согласования дополнительных услуг, которые оказывает сеть абоненту. Например, услуга «Идентификатор пользователя сети» позволяет задать идентификатор пользователя (отличный от его сетевого адреса), на основании которого могут оплачиваться счета за пользование сетью. Пользователь с помощью услуги «Согласование параметров управления потоком» может попросить сеть использовать нестандартные значения параметров протокола — размера окна, максимального размера поля данных пакета и т. п. Протокол X.25 допускает следующие максимальные значения длины поля данных: 16, 32, 64, 128, 256, 512 и

1024 байт. Предпочтительной является длина 128 байт.

Пакет Call Request принимается коммутатором сети и маршрутизируется па основании таблицы маршрутизации, прокладывая при этом виртуальный канал. Начальное значение номера виртуального канала задает пользователь в этом пакете в поле LCN (аналог поля VCI, упоминавшегося при объяснении принципа установления виртуальных каналов). Протокол маршрутизации для сетей X.25 не определен.

Для сокращения размера адресных таблиц в коммутаторах в сетях X.25 реализуется принцип агрегирования адресов. Все терминалы, имеющие общий префикс в адресе, подключаются при этом к общему входному коммутатору подсети, соответствующей значению префикса. Например, если путь ко всем терминалам, имеющим адреса с префиксом 250 720, пролегает через общий коммутатор K1, то в таблице маршрутизации коммутаторов, через которые проходит путь к коммутатору K1, помещается единственная запись — 250 720, которая соответствует как конечному узлу 250 720 11, так и конечному узлу 250 720 26. Маски в коммутаторах не используются, а младшие разряды адреса, которые не нужны при маршрутизации, просто опускаются.

После установления виртуального канала конечные узлы обмениваются пакетами другого формата — формата пакетов данных (пакет Data). Этот формат похож на описанный формат пакета Call Request — первые три байта в нем имеют те же поля, а адресные поля и поля услуг отсутствуют. Пакет данных не имеет поля, которое бы определяло тип переносимых в пакете данных, то есть поля, аналогичного полю Protocol и IP-пакете. Для устранения этого недостатка первый байт в поле данных всегда интерпретируется как признак типа данных. Коммутаторы (ЦКП) сетей X.25 представляют собой гораздо более простые и дешевые устройства по сравнению с маршрутизаторами сетей TCP/IP. Это объясняется тем, что они не поддерживают процедур обмена маршрутной информацией и нахождения оптимальных маршрутов, а также не выполняют преобразований форматов кадров канальных протоколов. По принципу работы они ближе к коммутаторам локальных сетей, чем к маршрутизаторам. Однако работа, которую выполняют коммутаторы X.25 над пришедшими кадрами, включает больше этапов, чем при продвижении кадров коммутаторами локальных сетей. Коммутатор X.25 должен принять кадр LAP-B и ответить па него другим кадром LAP-B, в котором подтвердить получение кадра с конкретным номером. При утере или искажении кадра коммутатор должен организовать повторную передачу кадра. Если же с кадром LAP-B все в порядке, то коммутатор должен извлечь пакет X.25, на основании номера виртуального канала определить выходной порт, а затем сформировать новый кадр LAP-B для дальнейшего продвижения пакета. Коммутаторы локальных сетей такой работой не занимаются и просто передают кадр в том виде, в котором он пришел, на выходной порт.

В результате производительность коммутаторов X.25 оказывается обычно невысокой — несколько тысяч пакетов в секунду. Для низкоскоростных каналов доступа, которыми много лет пользовались абоненты этой сети, такой производительности коммутаторов (1200-9600 бит/с) хватало для работы сети.

Гарантий пропускной способности сеть X.25 не даст. Максимум, что может

сделать сеть, — это приоритезировать трафик отдельных виртуальных каналов. Приоритет канала указывается в запросе на установление соединения в поле услуг.

Протоколы сетей X.25 были специально разработаны для низкоскоростных линий с высоким уровнем помех. Именно такие линии составляют пока большую часть телекоммуникационной структуры нашей страны, поэтому сети X.25 будут по-прежнему еще долго являться наиболее рациональным выбором для многих регионов.

Выводы

- Сети X.25 относятся к одной из наиболее старых и отработанных технологий глобальных сетей. Трехуровневый стек протоколов сетей X.25 хорошо работает на ненадежных зашумленных каналах связи, исправляя ошибки и управляя потоком данных на канальном и пакетном уровнях.
- Сети X.25 поддерживают групповое подключение к сети простых алфавитно-цифровых терминалов за счет включения в сеть специальных устройств PAD, каждое из которых представляет собой особый вид терминального сервера.
- На надежных волоконно-оптических каналах технология X.25 становится избыточной и неэффективной, так как значительная часть работы ее протоколов ведется «вхолостую».

Сети frame relay

Назначение и общая характеристика

Сети frame relay — сравнительно новые сети, которые гораздо лучше подходят для передачи пульсирующего трафика локальных сетей по сравнению с сетями X.25, правда, это преимущество проявляется только тогда, когда каналы связи приближаются по качеству к каналам локальных сетей, а для глобальных каналов такое качество обычно достижимо только при использовании волоконно-оптических кабелей.

Преимущество сетей frame relay заключается в их низкой протокольной избыточности и дейтаграммном режиме работы, что обеспечивает высокую пропускную способность и небольшие задержки кадров. Надежную передачу кадров технология frame relay не обеспечивает. Сети frame relay специально разрабатывались как общественные сети для соединения частных локальных сетей. Они обеспечивают скорость передачи данных до 2 Мбит/с.

Особенностью технологии frame relay является гарантированная поддержка основных показателей качества транспортного обслуживания локальных сетей — средней скорости передачи данных по виртуальному каналу при допустимых пульсациях трафика.

Технология frame relay была стандартизована как служба в сетях ISDN. В рекомендациях 1.122, вышедших в свет в 1988 году, эта служба входила в число до-

полнительных служб пакетного режима, но затем уже при пересмотре рекомендаций в 1992-93 годах она была названа службой frame relay и вошла в число служб режима передачи кадров наряду со службой frame switching. Служба frame switching работает в режиме гарантированной доставки кадров с регулированием потока. На практике поставщики телекоммуникационных услуг предлагаю только службу frame relay.

Технология frame relay сразу привлекла большое внимание ведущих телекоммуникационных компаний и организаций по стандартизации. В ее становлении и стандартизации помимо CCITT (ITU-T) активное участие принимают Frame Relay Forum и комитет T1S1 института ANSI.

Некоммерческую организацию Frame Relay Forum образовали в 1990 году компании Cisco Systems, StrataCom (сегодня — подразделение Cisco Systems), Northern Telecom и Digital Equipment Corporation для развития и конкретизации стандартов CCITT и ANSI. Спецификации Frame Relay Forum носят название FRF и имеют порядковые номера. Спецификации FRF часто стандартизуют те аспекты технологии frame relay, которые еще не нашли свое отражение в стандартах ITU-T и ANSI. Например, спецификация FRF.11 определяет режим передачи голоса по сетям frame relay.

Консорциум Frame Relay Forum разработал спецификацию, отвечающую требованиям базового протокола frame relay, разработанного T1S1 и CCITT. Однако консорциум расширил базовый протокол, включив дополнительные возможности по управлению сетью со стороны пользователя, что очень важно при использовании сетей frame relay в сложных составных корпоративных сетях. Эти дополнения к frame relay называют обобщенно Local Management Interface (LM1) — локальный интерфейс управления.

Стандарты ITU-T обычно отличаются высоким уровнем сложности и наличием многих возможностей, которые достаточно трудно воплотить на практике. Спецификации Frame Relay Forum упрощают некоторые аспекты стандартов ITU-T или отбрасывают некоторые возможности. Так, технология frame switching не нашла своего отражения в спецификациях FRF, а процедуры создания коммутируемых виртуальных каналов появились в спецификациях FRF позже, чем в стандартах ITU-T, и оказались более простыми.

Стандарты frame relay, как ITU-T/ANSI, так и Frame Relay Forum, определяют два типа виртуальных каналов — постоянные (PVC) и коммутируемые (SVC). Это соответствует потребностям пользователей, так как для соединений, по которым трафик передается почти всегда, больше подходят постоянные каналы, а для соединений, которые нужны только на несколько часов в месяц, больше подходят коммутируемые каналы.

Однако производители оборудования frame relay и поставщики услуг сетей frame relay начали с поддержки только постоянных виртуальных каналов. Это, естественно, является большим упрощением технологии. Тем не менее в последние годы оборудование, поддерживающее коммутируемые виртуальные каналы, появилось, и появились поставщики, предлагающие такую услугу.

Стек протоколов frame relay

Технология frame relay использует для передачи данных технику виртуальных соединений, аналогичную той, которая применяется в сетях X.25, однако стек протоколов frame relay передает кадры (при установленном виртуальном соединении) по протоколам только физического и канального уровней, в то время как в сетях X.25 и после установления соединения пользовательские данные передаются протоколом 3-го уровня.

Кроме того, протокол канального уровня LAP-F в сетях frame relay имеет два режима работы — основной (core) и управляющий (control). В основном режиме, который фактически практикуется в сегодняшних сетях frame relay, кадры передаются без преобразования и контроля, как и в коммутаторах локальных сетей. За счет этого сети frame relay обладают весьма высокой производительностью, так как кадры в коммутаторах не подвергаются преобразованию, а сеть не передает квитанции подтверждения между коммутаторами на каждый пользовательский кадр, как это происходит в сети X.25. Пульсации трафика передаются сетью frame relay достаточно быстро и без больших задержек.

При таком подходе уменьшаются накладные расходы при передаче пакетов локальных сетей, так как они вкладываются сразу в кадры канального уровня, а не в пакеты сетевого уровня, как это происходит в сетях X.25.

Структура стека (рис. 17.10) хорошо отражает происхождение технологии frame relay в недрах технологии ISDN, так как сети frame relay заимствуют многое из стека протоколов ISDN, особенно из процедурах установления коммутируемого виртуального канала.

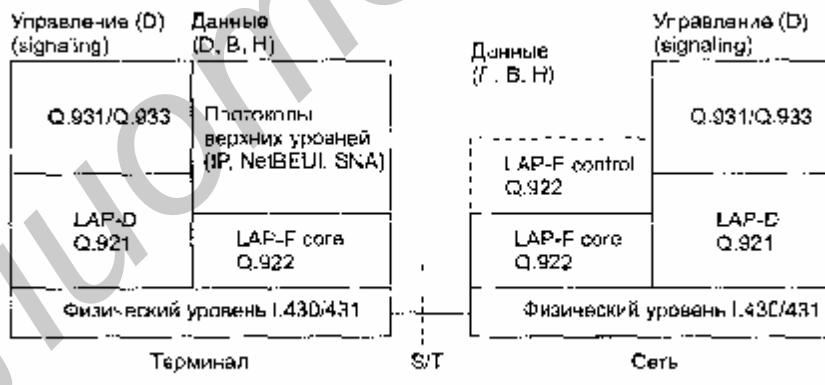


Рис. 17.10. Стек протоколов frame relay

Основу технологии составляет протокол LAP-F core, который является весьма упрощенной версией протокола LAP-D. Протокол LAP-F (стандарт Q.922 ITU-T) работает на любых каналах сети ISDN, а также на каналах типа T1/E1. Терминальное оборудование посылает в сеть кадры LAP-F в любой момент времени, считая, что виртуальный канал в сети коммутаторов уже проложен. При использовании PVC оборудованию frame relay нужно поддерживать только протокол LAP-F core.

Протокол LAP-F control является необязательной надстройкой над LAP-F core,

которая выполняет функции контроля доставки кадров и управления потоком. С помощью протокола LAP-F control сетью реализуется служба frame switching. Для установки коммутируемых виртуальных каналов стандарт ITU-T предлагает канал D пользовательского интерфейса. На нем по-прежнему работает знакомый протокол LAP-D, который используется для надежной передачи кадров в сетях ISDN. Поверх этого протокола работает протокол Q.931 или протокол Q.933 (который является упрощением и модификацией протокола Q.931 ISDN), устанавливающий виртуальное соединение на основе адресов конечных абонентов (в стандарте E.1G4 или ISO 7498), а также номера виртуального соединения, который в технологии frame relay носит название Data Link Connection Identifier - DLCI.

После того как коммутируемый виртуальный канал в сети frame relay установлен посредством протоколов LAP-D и Q.931/933, кадры могут транслироваться по протоколу LAP-F, коммутирующему их с помощью таблиц коммутации портов, в которых используются локальные значения DLCI. Протокол LAP-F core выполняет не все функции канального уровня по сравнению с протоколом LAP-D, поэтому ITU-T изображает его на ноль-уроне ниже, чем протокол LAP-D, оставляя место для функций надежной передачи пакетов протоколу LAP-F control.

Из-за того, что технология frame relay заканчивается на канальном уровне, она хорошо согласуется с идеей инкапсуляции пакетов единого сетевого протокола, например IP, в кадры канального уровня любых сетей, составляющих интересы. Процедуры взаимодействия протоколов сетевого уровня с технологией frame relay стандартизованы, например, принятая спецификация RFC 1490, определяющая методы инкапсуляции в трафик frame relay трафика сетевых протоколов и протоколов канального уровня локальных сетей и SNA.

Другой особенностью технологии frame relay является отказ от коррекции обнаруженных и кадрах искажений. Протокол frame relay подразумевает, что конечные узлы будут обнаруживать и корректировать ошибки за счет работы протоколов транспортного или более высоких уровней. Это требует некоторой степени интеллектуальности от конечного оборудования, что по большей части справедливо для современных локальных сетей. В этом отношении технология frame relay близка к технологиям локальных сетей, таким как Ethernet, Token Ring и FDDI, которые тоже только отбрасывают искаженные кадры, но сами не занимаются их повторной передачей.

Структура кадра протокола LAP-F приведена на рис. 17.11.



Рис. 17.11. Формат кадра LAP-F

За основу взят формат кадра HDLC, но поле адреса существенно изменило свой формат, а поле управления вообще отсутствует.

Поле номера виртуального соединения (Data Link Connection Identifier, DLCI) состоит из 10 бит, что позволяет использовать до 1024 виртуальных соединений. Поле DLCI может занимать и большее число разрядов — этим управляют признаки EA0 и EA1 (Extended Address — расширенный адрес). Если бит в этом признаком установлен в ноль, то признак называется EA0 и означает, что в следующем байте имеется продолжение поля адреса, а если бит признака равен 1, то поле называется EA1 и индицирует окончание поля адреса.

Десятиразрядный формат DLCI является основным, но при использовании трех байтов для адресации поле DLCT имеет длину 16 бит, а при использовании четырех байтов — 23 бита.

Стандарты frame relay (ANSI, ITU-T) распределяют адреса DLCI между пользователями и сетью следующим образом:

- 0 — используется для виртуального канала локального управления (LMI);
- 1-15 — зарезервированы для дальнейшего применения;
- 16-991 — используются абонентами для нумерации PVC и SVC;
- 992-1007 — используются сетевой транспортной службой для внутрисетевых соединений;
- 1008-1022 — зарезервированы для дальнейшего применения;
- 1023 — используются для управления канальным уровнем.

Таким образом, в любом интерфейсе frame relay для оконечных устройств пользователя отводится 97G адресов DLCI.

Поле данных может иметь размер до 4096 байт.

Поле C/R имеет обычный для протокола семейства HDLC смысл — это признак «команда—ответ».

Поля DE, FECN и BECN используются протоколом для управления трафиком и поддержания заданного качества обслуживания виртуального канала.

ПРИМЕЧАНИЕ

Способность технологии frame relay гарантировать некоторые параметры качества обслуживания (QoS) является ключевой. Именно поэтому данная технология получила широкое распространение.

Поддержка качества обслуживания

Для каждого виртуального соединения определяется несколько параметров, связанных со скоростью передачи данных и влияющих на качество обслуживания.

- CIR (Committed-Information Rate) — согласованная информационная скорость, с которой сеть будет передавать данные пользователя.
- B_c (Committed Burst Size) — согласованный объем пульсации, то есть максимальное количество байтов, которое сеть будет передавать от этого пользователя за интервал времени T.

□ B_e (Excess Burst: Size) — дополнительный объем пульсации, то есть максимальное количество байтов, которое сеть будет пытаться передать сверх установленного значения B_c за интервал времени T .

Параметры QoS, связанные с задержками и вариациями задержек, стандартами frame relay не оговариваются, так как изначально технология разрабатывалась только для передачи трафика, не чувствительного к задержкам.

Если приведенные выше величины определены, то время T определяется следующей формулой:

$$T = B_c/CIR.$$

Можно задать значения CIR и T , тогда производной величиной станет величина всплеска трафика B_e .

Соотношение между параметрами CIR, B_c , B_e и T иллюстрирует рис. 17.12 (R — скорость канала доступа; f_1-f_5 — кадры).

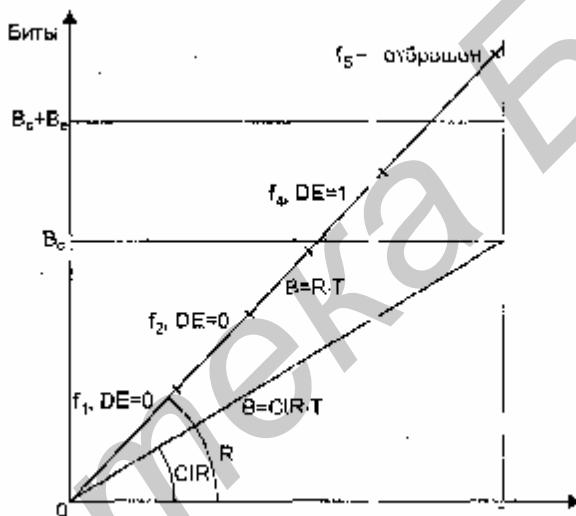


Рис. 17.12. Реакция сети на гово ячные пульзателя

Основным параметром, по которому абонент и сеть заключают соглашение при установлении виртуального соединения, является согласованная скорость передачи данных. Для постоянных виртуальных каналов это соглашение является частью контракта на пользование услугами сети. При установлении коммутируемого виртуального каната соглашение о качестве обслуживания заключается автоматически с помощью протокола Q.931/933 — требуемые параметры CIR, B_c и B_e передаются в пакете запроса на установление соединения.

Так как скорость передачи данных измеряется на каком-то интервале времени, то интервал 7"и является таким контрольным интервалом, на котором проверяются условия соглашения. В общем случае пользователь не должен за этот интервал передать в сеть данные со средней скоростью, превосходящей CIR. Если же он нарушает соглашение, то сеть не только не гарантирует доставку кадра, но помечает этот кадр признаком DE (Discard Eligibility), равным 1, то есть как кадр, подлежащий удалению. Однако кадры, отмеченные таким признаком, удаляются из сети только в том случае, если коммутаторы сети испытывают

перегрузки. Если же перегрузок нет, то кадры с признаком DE = 1 доставляются адресату.

Такое щадящее поведение сети соответствует случаю, когда общее количество данных, переданных пользователем в сеть за период T, не превышает объема $B_c + B_e$. Если же этот порог превышен, то кадр не помечается признаком DE, а немедленно удаляется из сети.

Рисунок 17.12 иллюстрирует случай, когда за интервал времени T в сеть по виртуальному каналу поступило 5 кадров. Средняя скорость поступления информации и сеть составила на этом интервале R бит/с, и она оказалась выше CIR. Кадры f_1 , f_2 и f_3 доставили в сеть данные, суммарный объем которых не превысил порог B_c , поэтому эти кадры ушли дальше транзитом с признаком DE=0. Данные кадра f_4 прибавленные к данным кадров f_1 , f_2 и f_3 уже превысили порог B_c , но еще не превысили порога $B_c + B_e$, поэтому кадр d также ушел дальше, но уже с признаком DE = 1. Данные кадра f_5 , прибавленные к данным предыдущих кадров, превысили порог $B_c + B_e$ поэтому этот кадр был удален из сети. Для контроля соглашения о параметрах качества обслуживания все коммутаторы сети frame relay выполняют так называемый алгоритм «дырявого ведра» (Leaky Bucket), описанный в главе 5.

Пользователь может договориться о включении не всех параметров качества обслуживания на данном виртуальном канале, а только некоторых.

Например, можно использовать только параметры CIR и Bt.. Этот вариант дает более качественное обслуживание, так как кадры никогда не отбрасываются коммутатором сразу. Коммутатор только помечает кадры, которые превышают порог B_c за время T, признаком DE = 1. Если сеть не сталкивается с перегрузками, то кадры такого канала всегда доходят до конечного узла, даже если пользователь постоянно нарушает договор с сетью.

Популярен еще один вид заказа на качество обслуживания, при котором оговаривается только порог B_c , а скорость CIR полагается равной нулю. Все кадры такого канала сразу же отмечаются признаком DE=1, но отправляются в сеть, а при превышении порога B_c они отбрасываются. Контрольный интервал времени T в этом случае вычисляется как B_e/R , где R — скорость доступа канала.

На рис. 17.13 приведен пример сети frame relay с пятью удаленными региональными отделениями корпорации. Обычно доступ к сети осуществляется каналами с большей чем CIR пропускной способностью. Но при этом пользователь платит не за пропускную способность канала, а за заказанные величины CIR, B_c и B_e . Так, при использовании и качестве канала доступа канала T1 и заказа службы со скоростью CIR, равной 128 кбит/с, пользователь будет платить только за скорость 128 кбит/с, а скорость капала T1 в 1,544 Мбит/с окажет влияние на верхнюю границу возможной пульсации $B_c + B_e$.

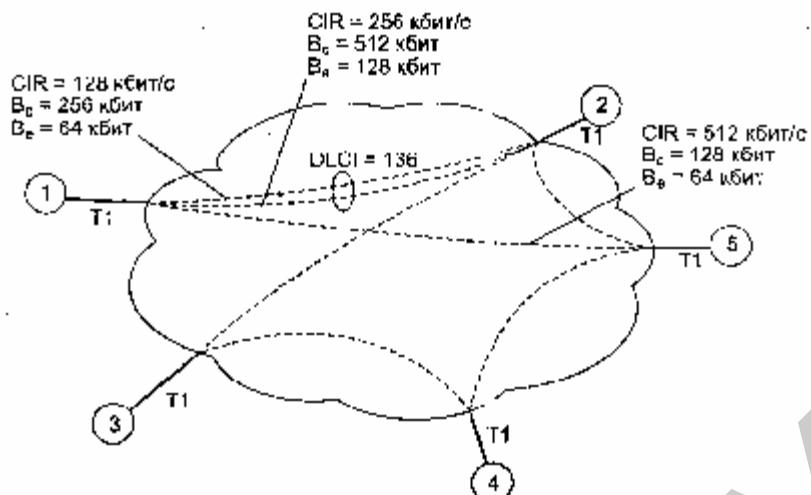


Рис. 17.13. Пример использования сети frame relay

Параметры качества обслуживания могут быть различными для разных направлений виртуального канала. Так, на рисунке абонент 1 соединен с абонентом 2 виртуальным каналом с $DLCI = 136$. При направлении от абонента 1 к абоненту 2 канал имеет среднюю скорость 128 кбит/с с пульсациями $B_c = 256$ кбит (интервал T составил 1 с) и $B_e = 64$ кбит. А при передаче кадров в обратном направлении средняя скорость уже может достигать значения 256 кбит/с с пульсациями $B_c = 512$ кбит и $B_e = 128$ кбит.

Механизм заказа средней пропускной способности и максимальной пульсации является основным механизмом управления потоками кадров в сетях frame relay. Соглашения должны заключаться таким образом, чтобы сумма средних скоростей виртуальных каналов не превосходила возможностей портов коммутаторов. При заказе постоянных каналов за это отвечает администратор, а при установлении коммутируемых виртуальных каналов — программное обеспечение коммутаторов. При правильно взятых на себя обязательствах сеть борется с перегрузками путем удаления кадров с признаком $DE = 1$ и кадров, превысивших порог $B_c + B_e$.

В технологии frame relay определен еще и дополнительный (необязательный) механизм управления кадрами. Это механизм оповещения конечных пользователей о том, что в коммутаторах сети возникли перегрузки (переполнение необработанными кадрами). Бит FECN (Forward Explicit Congestion Bit) кадра извещает об этом принимающую сторону. На основании значения этого бита принимающая сторона должна с помощью протоколов более высоких уровней (TCP/IP, SPX и т. п.) известить передающую сторону о том, что та должна снизить интенсивность отправки пакетов в сеть.

Бит BECN (Backward Explicit Congestion Bit) извещает о переполнении в сети передающую сторону и является рекомендацией немедленно снизить темп передачи. Бит BECN обычно отрабатывается на уровне устройств доступа к сети frame relay — маршрутизаторов, мультиплексоров и устройств CSU/DSU. Протокол frame relay не требует от устройств, получивших кадры с установленными битами FECN и BECN, немедленного прекращения передачи кадров в дан-

ном направлении, как того требуют кадры RNR сетей X.25. Эти биты должны служить указанием для протоколов более высоких уровней (TCP, SPX, NCP и т. п.) о снижении темпа передачи пакетов. Так как регулирование потока и принимающей, и передающей сторонами инициируется в разных протоколах по-разному, то разработчики протоколов frame relay учли оба направления снабжения предупреждающей информацией о переполнении сети.

В общем случае биты FECN и BECN могут игнорироваться. Но обычно устройства доступа к сети frame relay (Frame Relay Access Device, FRAD) отрабатывают по крайней мере признак BECN.

При создании коммутируемого виртуального канала параметры качества обслуживания передаются в сеть по протоколу Q.931. Этот протокол устанавливает виртуальное соединение с помощью нескольких служебных пакетов.

Абонент сети frame relay, который хочет установить коммутируемое виртуальное соединение с другим абонентом, должен передать в сеть по каналу D сообщение SETUP, которое имеет несколько параметров, в том числе:

- DLCT;
- адрес назначения (в формате E.164, X.121 или ISO 7498);
- максимальный размер кадра в данном виртуальном соединении;
- запрашиваемое значение CIR для двух направлений;
- запрашиваемое значение B_c для двух направлений;
- запрашиваемое значение B_e для двух направлений.

Коммутатор, с которым соединен пользователь, сразу же передает пользователю пакет CALL PROCEEDING — вызов обрабатывается. Затем он анализирует параметры, указанные в пакете, и если коммутатор может их удовлетворить (располагая, естественно, информацией о том, какие виртуальные каналы на каждом порту он уже поддерживает), то пересыпает сообщение SETUP следующему коммутатору. Следующий коммутатор выбирается по таблице маршрутизации. Протокол автоматического составления таблиц маршрутизации для технологии frame relay не определен, поэтому может использоваться фирменный протокол производителя оборудования или же выполняться ручное составление таблицы. Если все коммутаторы на пути к конечному узлу согласны принять запрос, то пакет SETUP передается в конечном счете вызываемому абоненту. Вызываемый абонент немедленно передаст в сеть пакет CALL PROCEEDING и начинает обрабатывать запрос. Если запрос принимается, то вызываемый абонент передает в сеть новый пакет — CONNECT, который проходит в обратном порядке по виртуальному пути. Все коммутаторы должны отметить, что данный виртуальный канал принят вызываемым абонентом. При поступлении сообщения CONNECT вызывающему абоненту он должен передать в сеть пакет CONNECT ACKNOWLEDGE.

Сеть также должна передать вызываемому абоненту пакет CONNECT ACKNOWLEDGE, и на этом соединение считается установленным. Но виртуальному каналу могут передаваться данные.

Использование сетей frame relay

Услуги frame relay обычно предоставляются теми же операторами, которые эксплуатируют сети X.25. Большая часть производителей выпускает сейчас коммутаторы, которые могут работать как по протоколам X.25, так и по протоколам frame relay.

Технология frame relay начинает занимать в территориальных сетях с коммутацией пакетов ту же нишу, которую заняла в локальных сетях технология Ethernet. Их роднит то, что они предоставляют только быстрые базовые транспортные услуги, доставляя кадры в узел назначения без гарантий, дейтаграммным способом. Однако если кадры теряются, то сеть frame relay, как и сеть Ethernet, не предпринимает никаких усилий для их восстановления. Отсюда следует простой вывод — полезная пропускная способность прикладных протоколов при работе через сеть frame relay будет зависеть от качества каналов и методов восстановления пакетов на уровнях стека, расположенного над протоколом frame relay. Если каналы качественные, то кадры будут теряться и искажаться редко, так что скорость восстановления пакетов протоколом TCP или NCP окажется вполне приемлемой. Если же кадры будут искажаться и теряться часто, то полезная пропускная способность в сети frame relay может упасть в десятки раз, как это происходит в сетях Ethernet при плохом состоянии кабельной системы.

Поэтому сети frame relay следует применять только при наличии на магистральных каналах волоконно-оптических кабелей высокого качества. Каналы доступа могут быть на витой паре, как это разрешает интерфейс G.703 или абонентское окончание ISDN. Используемая на каналах доступа аппаратура передачи данных должна обеспечить приемлемый уровень искажения данных — не ниже 10~.

На величины задержек сеть frame relay гарантий не дает, и это основная причина, которая сдерживает применение этих сетей для передачи голоса. Передача видеоизображения тормозится и другим отличием сетей frame relay от ATM — низкой скоростью доступа в 2 Мбит/с, что для передачи видео часто недостаточно.

Тем не менее многие производители оборудования для сетей frame relay поддерживают передачу голоса. Поддержка устройствами доступа заключается в присвоении кадрам, переносящим замеры голоса, приоритетов. Магистральные коммутаторы frame relay должны обслуживать такие кадры в первую очередь. Кроме того, желательно, чтобы сеть frame relay, передающая кадры с замерами голоса, была недогруженной. При этом в коммутаторах не возникают очереди кадров, и средние задержки в очередях близки к нулевым.

Необходимо также соблюдение еще одного условия для качественной передачи голоса — передавать замеры голоса необходимо в кадрах небольших размеров, иначе на качество будут влиять задержки упаковки замеров в кадр, так называемые задержки пакетизации, которые более подробно рассматриваются ниже в разделе «Технология ATM».

Для стандартизации механизмов качественной передачи голоса через сеть frame relay выпущена спецификация FRF.11. Однако в ней решены не все проблемы

передачи голоса, поэтому работа в этом направлении продолжается.

Ввиду преобладания в коммерческих сетях frame relay услуг постоянных коммутируемых каналов и гарантированной пропускной способности, эти сети предоставляют услуги, очень похожие на услуги дробных выделенных линий T1/E1, но только за существенно меньшую плату.

При использовании PVC сеть frame relay хорошо подходит для объединения локальных сетей с помощью мостов, так как в этом случае от моста не нужна поддержка механизма установления виртуального канала, что требует некоторого программного «интеллекта». Мост может отправлять кадры протокола Ethernet или FDDI непосредственно в кадрах LAP-F или же выполнять протокол PPP поверх протокола LAP-F. Стандарт Интернета RFC 1490 определяет формат заголовка SNAP для случая передачи через сеть frame relay непосредственно кадров канального уровня.

Чаще доступ к сетям frame relay реализуют не удаленные мосты, а маршрутизаторы, которые в случае поддержки на последовательных портах протокола frame relay как основного называют устройствами доступа FRAD (хотя и мост, и любое устройство, которое поддерживает протоколы UNI, frame relay, относятся к классу FRAD).

Так как сети frame relay передают кадры с небольшими задержками, с их помощью часто передают трафик сетей SNA, особенно в том случае, когда они используют такие чувствительные к задержкам протоколы, как SDLC (фирменный протокол канального уровня компании IBM).

Виртуальные каналы в качестве основы построения корпоративной сети имеют один недостаток — при большом количестве точек доступа и сметанном характере связей необходимо большое количество виртуальных каналов, каждый из которых оплачивается отдельно. В сетях с маршрутизацией отдельных пакетов, таких как TCP/IP, абонент платит только за количество точек доступа, а не за количество связей между ними.

Выводы

- Сети frame relay работают на основе весьма упрощенной, по сравнению с сетями X.25, технологией, которая передает кадры только по протоколу канального уровня — протоколу LAP-F. Кадры при передаче через коммутатор не подвергаются преобразованиям, из-за чего технология и получила свое название.
- Важной особенностью технологии frame relay является концепция резервирования пропускной способности при прокладке в сети виртуального канала. Сети frame relay создавались специально для передачи пульсирующего компьютерного трафика, поэтому при резервировании пропускной способности указывается средняя скорость трафика C1R и согласованный объем пульсаций B_c .
- Сеть frame relay гарантирует поддержку заказанных параметров качества обслуживания за счет предварительного расчета возможностей каждого коммутатора, а также отбрасывания кадров, которые нарушают соглашение о трафике, то есть посылаются в сеть слишком интенсивно.

- Большинство первых сетей frame relay поддерживали только службу постоянных виртуальных каналов, а служба коммутируемых виртуальных каналов стала применяться на практике только недавно.

Технология ATM

Гетерогенность — неотъемлемое качество любой крупной вычислительной сети, и на согласование разнородных компонентов системные интеграторы и администраторы тратят большую часть своего времени. Поэтому любое средство, сулящее перспективу снижения неоднородности сети, привлекает пристальный интерес сетевых специалистов. Технология асинхронного режима передачи (Asynchronous Transfer Mode, ATM) разработана как единый универсальный транспорт для нового поколения сетей с интеграцией услуг, которые называются широкополосными сетями ISDN (Broadband-ISDN, B-ISDN).

По планам разработчиков единообразие, обеспечиваемое ATM, будет состоять в том, что одна транспортная технология сможет обеспечить несколько перечисленных ниже возможностей.

- Передачу в рамках одной транспортной системы компьютерного и мультимедийного (голос, видео) трафика, чувствительного к задержкам, причем для каждого вида трафика качество обслуживания будет соответствовать его потребностям.
- Иерархию скоростей передачи данных, от десятков мегабит до нескольких гигабит в секунду с гарантированной пропускной способностью для ответственных приложений.
- Общие транспортные протоколы для локальных и глобальных сетей.
- Сохранение имеющейся инфраструктуры физических каналов или физических протоколов: T1/E1, T3/E3, SDH STM-n, FDDI.
- Взаимодействие с унаследованными протоколами локальных и глобальных сетей: IP, SNA, Ethernet, ISDN.

Главная идея технологии асинхронного режима передачи была высказана достаточно давно — этот термин ввела лаборатория Bell Labs еще в 1968 году. Основной разрабатываемой технологией тогда была технология ТОМ с синхронными методами коммутации, основанными на порядковом номере байта в объединенном кадре. Главный недостаток технологии ТОМ, которую также называют технологией синхронной передачи STM (Synchronous Transfer Mode), заключается в невозможности перераспределять пропускную способность объединенного канала между подканалами. В те периоды времени, когда по подканалу не передаются пользовательские данные, объединенный канал все равно передаст байты этого подканала, заполненные пульями.

Попытки загрузить периоды простоя подканалов приводят к необходимости введения заголовка для данных каждого подканала. В промежуточной технологии STDM (Statistical TDM), которая позволяет заполнять периоды простоя передачей пульсаций трафика других подканалов, действительно вводятся заголовки, содержащие номер подканала. Данные при этом оформляются в пакеты, похожие по структуре на пакеты компьютерных сетей. Наличие адреса у каж-

дого пакета позволяет передавать его асинхронно, так как местоположение его относительно данных других подканалов уже не является его адресом. Асинхронные пакеты одного подканала вставляются в свободные тайм-слоты другого подканала, но не смешиваются с данными этого подканала, так как имеют собственный адрес.

Технология ATM совмещает в себе подходы двух технологий — коммутации пакетов и коммутации каналов. От первой она взяла па вооружение передачу данных в виде адресуемых пакетов, а от второй — технику виртуальных каналов и использование пакетов небольшого фиксированного размера, в результате чего задержки в сети становятся более предсказуемыми. С помощью техники виртуальных каналов, предварительного заказа параметров качества обслуживания канала и приоритетного обслуживания виртуальных каналов с разным качеством обслуживания удается добиться передачи в одной сети разных типов трафика без дискриминации. Хотя сети ISDN также разрабатывались для передачи различных видов трафика в рамках одной сети, голосовой трафик явно был для разработчиков более приоритетным. Технология ATM с самого начала разрабатывалась как технология, способная обслуживать все виды трафика в соответствии с их требованиями.

Службы верхних уровней сети B-ISDN должны быть примерно такими же, что и у сети ISDN — это передача факсов, распространение телевизионного изображения, голосовая почта, электронная почта, различные интерактивные службы, например телефония и проведение видеоконференций. Высокие скорости технологии ATM создают гораздо больше возможностей для служб верхнего уровня, которые не могли быть реализованы сетями ISDN — например, для передачи цветного телевизионного изображения необходима полоса пропускания в районе 30 Мбит/с. Технология ISDN такую скорость поддержать не может, а для ATM она не составляет больших проблем.

Разработку стандартов ATM осуществляют группа организаций под названием ATM Forum под эгидой специального комитета IEEE, а также комитеты ITU-T и ANSI. ATM — это очень сложная технология, требующая стандартизации в самых различных аспектах, поэтому, хотя основное ядро стандартов было принято в 1993 году, работа по стандартизации активно продолжается. Оптимизм внушает тот факт, что в ATM Forum принимают участие практически все заинтересованные стороны — производители телекоммуникационного оборудования, производители оборудования локальных сетей, операторы телекоммуникационных сетей и сетевые интеграторы. До широкого распространения технологии ATM по оценкам специалистов должно пройти еще 5-10 лет. Такой прогноз связан не только с отсутствием полного набора принятых стандартов, но и с невозможностью быстрой замены уже установленного дорогостоящего оборудования, которое хотя и не так хорошо, как хотелось бы, но все же справляется со своими обязанностями. Кроме того, многое еще нужно сделать в области стандартизации взаимодействия ATM с существующими сетями, как компьютерными, так и телефонными.

Основные принципы технологии ATM

Сеть ATM имеет классическую структуру крупной территориальной сети — конечные станции соединяются индивидуальными каналами с коммутаторами нижнего уровня, которые, в свою очередь, соединяются с коммутаторами более высоких уровней. Коммутаторы ATM пользуются 20-байтными адресами конечных узлов для маршрутизации трафика на основе техники виртуальных каналов. Для частных сетей ATM определен протокол маршрутизации PNNI (Private NNI), с помощью которого коммутаторы могут строить таблицы маршрутизации автоматически. В публичных сетях ATM таблицы маршрутизации могут строиться администраторами вручную, как и в сетях X.25, или поддерживаться протоколом PNNI.

Коммутация пакетов происходит на основе идентификатора виртуального канала (Virtual Channel Identifier, VCI), который назначается соединению при его установлении и уничтожается при разрыве соединения. Адрес конечного узла ATM, на основе которого прокладывается виртуальный канал, имеет иерархическую структуру, подобную номеру в телефонной сети, и использует префиксы, соответствующие кодам стран, городов, сетям поставщиков услуг и т. п., что упрощает маршрутизацию запросов на установление соединения, как и в случае агрегированных IP-адресов в соответствии с техникой CIDR. В публичных сетях ATM обычно используются адреса в стандарте E.1G4, что делает простым изаимодействие этих сетей с телефонными сетями.

Виртуальные соединения могут быть постоянными (Permanent Virtual Circuit, PVC) и коммутируемыми (Switched Virtual Circuit, SVC). Для ускорения коммутации в больших сетях используется понятие виртуального пути — Virtual Path, который объединяет виртуальные каналы, имеющие в сети ATM общий маршрут между исходным и конечным узлами или общую часть маршрута между некоторыми двумя коммутаторами сети. Идентификатор виртуального пути (Virtual Path Identifier, VPI) является старшей частью локального адреса и представляет собой общий префикс для некоторого количества различных виртуальных каналов. Таким образом, идея агрегирования адресов в технологии ATM применена на двух уровнях — на уровне адресов конечных узлов (работает на стадии установления виртуального канала) и на уровне номеров виртуальных каналов (работает при передаче данных по имеющемуся виртуальному каналу).

Соединения конечной станции ATM с коммутатором нижнего уровня определяются стандартом UNI (User Network Interface). Спецификация UNI определяет структуру пакета, адресацию станций, обмен управляющей информацией, уровни протокола ATM, способы установления виртуального канала и способы управления трафиком.

Стандарт ATM не вводит свои спецификации на реализацию физического уровня. Здесь он основывается на технологии SDH/SONET, принимая ее иерархию скоростей. В соответствии с этим начальная скорость доступа пользователя сети — это скорость STM-1 155 Мбит/с. Магистральное оборудование ATM работает и на более высоких скоростях STM-4 622 Мбит/с и STM-16 2,5 Гбит/с. На скорости 155 Мбит/с можно использовать не только волоконно-оптический

кабель, но и неэкранированную витую пару категории 5. На скорости 622 Мбит/с допустим только волоконно-оптический кабель, причем для глобальных сетей это одномодовый кабель, а для локальных — как одномодовый, так и многомодовый (в зависимости от скорости и расстояния). Работа на сверхвысоких скоростях существенно удорожает оборудование ATM из-за сложности реализации операций разбиения пакетов на ячейки и сборки ячеек в пакеты в интерфейсных заказных БИС коммутаторов.

Имеются и другие физические интерфейсы к сетям ATM, отличные от SDH/SONET. К ним относятся интерфейсы T1/E1 и T3/E3, распространенные в глобальных сетях, и интерфейсы локальных сетей — интерфейс с кодировкой 4B/5B со скоростью 100 Мбит/с (FDDI) и интерфейс со скоростью 25 Мбит/с, предложенный компанией IBM и утвержденный ATM Forum. Кроме того, для скорости 155,52 Мбит/с определен так называемый физический уровень «cell-based[^]», то есть уровень, основанный на ячейках, а не на кадрах SDH/SONET. Этот вариант физического уровня не использует кадры SDH/SONET, а отправляет по каналу связи непосредственно ячейки формата ATM, что сокращает накладные расходы на служебные данные, но несколько усложняет задачу синхронизации приемника с передатчиком на уровне ячеек.

Все перечисленные выше характеристики технологии ATM не свидетельствуют о том, что это некая «особенная» технология, а скорее представляют ее как типичную технологию глобальных сетей, основанную на технике виртуальных каналов. Особенности же технологии ATM лежат в области качественного обслуживания разнородного трафика и объясняются стремлением решить задачу совмещения в одних и тех же каналах связи и в одном и том же коммуникационном оборудовании компьютерного и мультимедийного трафика таким образом, чтобы каждый тип трафика получил требуемый уровень обслуживания и не рассматривался как «второстепенный».

Трафик вычислительных сетей имеет ярко выраженный асинхронный и пульсирующий характер. Компьютер посылает пакеты в сеть в случайные моменты времени, по мере возникновения в этом необходимости. При этом интенсивность посылки пакетов в сеть и их размер могут изменяться в широких пределах — например, коэффициент пульсаций трафика (отношения максимальной мгновенной интенсивности трафика к его средней интенсивности) протоколов без установления соединений может доходить до 200, а протоколов с установлением соединений — до 20. Чувствительность компьютерного трафика к потерям данных высокая, так как без утраченных данных обойтись нельзя и их необходимо восстановить путем повторной передачи.

Мультимедийный трафик, передающий, например, голос или изображение, характеризуется низким коэффициентом пульсаций, высокой чувствительностью к задержкам передачи данных (отражающихся на качестве воспроизведенного непрерывного сигнала) и низкой чувствительностью к потерям данных (из-за инерционности физических процессов потерю отдельных замеров голоса или кадров изображения можно компенсировать сглаживанием на основе предыдущих и последующих значений).

Сложность совмещения компьютерного и мультимедийного трафиков с диа-

метрально противоположными характеристиками хорошо видна на рис. 17.14.



Рис. 17.14. Два типа трафика: а — компьютерный; б — мультимедийный

На возможности совмещения этих двух видов трафика большое влияние оказывает размер компьютерных пакетов. Если размер пакета может меняться в широком диапазоне (например, от 29 до 4500 байт, как в технологии FDDI), то даже при придании голосовым пакетам высшего приоритета обслуживания в коммутаторах время ожидания компьютерного пакета может оказаться недопустимо высоким. Например, пакет в 4500 байт будет передаваться в выходной порт на скорости 2 Мбит/с (максимальная скорость работы порта коммутатора frame relay) 18 мс. При совмещении трафика за это время необходимо через этот же порт передать 144 замера голоса. Прерывать передачу пакета в сетях нежелательно, так как при распределенном характере сети накладные расходы на оповещение соседнего коммутатора о прерывании пакета, а потом — о возобновлении передачи пакета с прерванного места оказываются слишком большими.

Подход, реализованный в технологии ATM, состоит в передаче любого вида трафика — компьютерного, телефонного или видео — пакетами фиксированной и очень маленькой длины в 53 байта. Пакеты ATM называют ячейками (cell). Поле данных ячейки занимает 48 байт, а заголовок — 5 байт.

Чтобы пакеты содержали адрес узла назначения и в то же время процент служебной информации не превышал размер поля данных пакета, в технологии ATM применен стандартный для глобальных вычислительных сетей прием — передача ячеек в соответствии с техникой виртуальных каналов с длиной номера виртуального канала размером в 24 бит, что вполне достаточно для обслуживания большого количества виртуальных соединений каждым портом коммутатора глобальной (может быть, всемирной) сети ATM.

Размер ячейки ATM является результатом компромисса между телефонистами и компьютерщиками — первые настаивали на размере поля данных в 32 байта, а вторые — в 64 байта.

Чем меньше пакет, тем легче имитировать услуги каналов с постоянной битовой скоростью, которая характерна для телефонных сетей. Ясно, что при отказе от жестко синхронизированных временных слотов для каждого канала идеальной синхронности добиться будет невозможно, однако чем меньше размер пакета, тем легче этого достичь.

Для пакета, состоящего из 53 байт, при скорости в 155 Мбит/с время передачи кадра на выходной порт составляет менее 3 мкс. Так что эта задержка не очень

существенна для трафика, пакеты которого должны передаваться каждые 125 мкс.

Однако на выбор размера ячейки большее влияние оказала не величина ожидания передачи ячейки, а задержка пакетизации. Задержка пакетизации — это время, в течение которого первый замер голоса ждет момента окончательного формирования пакета и отправки его по сети. При размере поля данных в 48 байт одна ячейка ATM обычно переносит 48 замеров голоса, которые делаются с интервалом в 125 мкс. Поэтому первый замер должен ждать примерно 6 мс, прежде чем ячейка будет отправлена по сети. Именно по этой причине телефонисты боролись за уменьшения размера ячейки, так как 6 мс — это задержка, близкая к пределу, за которым начинаются нарушения качества передачи голоса. При выборе размера ячейки в 32 байта задержка пакетизации составила бы 4 мс, что гарантировало бы более качественную передачу голоса. А стремление компьютерных специалистов увеличить поле данных до 64 байт вполне попятно — при этом повышается полезная скорость передачи данных. Избыточность служебных данных при использовании 48-байтного поля данных составляет 10 %, а при использовании 32-байтного поля данных она сразу повышается до 16 %.

Выбор для передачи данных любого типа небольшой ячейки фиксированного размера еще не решает задачу совмещения разнородного трафика в одной сети, а только создаст предпосылки для ее решения. Для полного решения этой задачи технология ATM привлекает и развивает идеи заказа пропускной способности и качества обслуживания, реализованные в технологии frame relay. Но если сеть frame relay изначально была предназначена для передачи только пульсирующего компьютерного трафика (в связи с этим для сетей frame relay так трудно дается стандартизация передачи голоса), то разработчики технологии ATM проанализировали всевозможные образцы трафика, создаваемые различными приложениями, и выделили 4 основных класса трафика, для которых разработали различные механизмы резервирования и поддержания требуемого качества обслуживания.

Класс трафика (называемый также классом услуг — service class) качественно характеризует требуемые услуги по передаче данных через сеть ATM. Если приложение указывает сети, что требуется, например, передача голосового трафика, то из этого становится ясно, что особенно важными для пользователя будут такие показатели качества обслуживания, как задержки и вариации задержек ячеек, существенно влияющие на качество переданной информации — голоса или изображения, а потеря отдельной ячейки с несколькими замерами не так уж важна, так как, например, воспроизводящее голос устройство может аппроксимировать недостающие замеры и качество пострадает не слишком. Требования к синхронности передаваемых данных очень важны для многих приложений — не только голоса, но и видеоизображения, и наличие этих требований стало первым критерием для деления трафика на классы.

Другим важным параметром трафика, существенно влияющим на способ его передачи через сеть, является величина пульсаций. Разработчики технологии ATM решили выделить два различных типа трафика в отношении этого пара-

метра — трафик с постоянной битовой скоростью (Constant Bit Rate, CBR) и трафик с переменной битовой скоростью (Variable Bit Rate, VBR).

К разным классам были отнесены трафики, порождаемые приложениями, использующими для обмена сообщениями протоколы с установлением соединения и без установления соединения. В первом случае данные передаются самим приложением достаточно надежно, как это обычно делают протоколы с установлением соединения, поэтому от сети ATM высокой надежности передачи не требуется. А во втором случае приложение работает без установления соединения и восстановлением потерянных и искаженных данных не занимается, что предъявляет повышенные требования к надежности передачи ячеек сетью ATM.

В результате было определено пять классов трафика, отличающихся следующими качественными характеристиками:

- наличием или отсутствием пульсации трафика, то есть трафики CBR или VBR;
- требованием к синхронизации данных между передающей и принимающей сторонами;
- типом протокола, передающего свои данные через сеть ATM, — с установлением соединения или без установления соединения (только для случая передачи компьютерных данных).

Основные характеристики классов трафика ATM приведены в табл. 17.1.

Таблица 17.1. Классы трафика ATM

Класс трафика	Характеристика
A	Постоянная битовая скорость — Constant Bit Rate, CBR
	Требуются временные соотношения между передаваемыми и принимаемыми данными
	С установлением соединения
	Примеры: голосовой трафик, трафик телевизионного изображения
B	Переменная битовая скорость — Variable Bit Rate, VBR
	Требуются временные соотношения между передаваемыми и принимаемыми данными
	С установлением соединения
C	Переменная битовая скорость — Variable Bit Rate, VBR
	Не требуются временные соотношения между передаваемыми и принимаемыми данными
	С установлением соединения
	Примеры: трафик компьютерных сетей, в которых конечные узлы работают по протоколам с установлением соединения

	нений: frame relay, X.25, LLC2, TCP
D	Переменная битовая скорость — Variable Bit Rate, VBR
	Не требуются временные соотношения между передаваемыми и принимаемыми данными
	Без установления соединения
	Примеры: трафик компьютерных сетей, в которых конечные узлы работают по протоколам без установления соединений (IP, Ethernet, DNS, SNMP)
X	Тип трафика и его параметры определяются пользователем

Библиотека БГУИР

Очевидно, что только качественных характеристик, задаваемых классом трафика, для описания требуемых услуг недостаточно. В технологии ATM для каждого класса трафика определен набор количественных параметров, которые приложение должно задать. Например, для трафика класса А необходимо указать постоянную скорость, с которой приложение будет посыпать данные в сеть, а для трафика класса В — максимально возможную скорость, среднюю скорость и максимально возможную пульсацию. Для голосового трафика можно не только указать на важность синхронизации между передатчиком и приемником, но и количественно задать верхние границы задержки и вариации задержки ячеек.

В технологии ATM поддерживается следующий набор основных количественных параметров:

- Peak Cell Rate (PCR) — максимальная скорость передачи данных;
- Sustained Cell Rate (SCR) — средняя скорость передачи данных;
- Minimum Cell Rate (MCR) — минимальная скорость передачи данных;
- Maximum Burst Size (MBS) — максимальный размер пульсации;
- Cell Loss Ratio (CLR) — доля потерянных ячеек;
- Cell Transfer Delay (CTD) — задержка передачи ячеек;
- Cell Delay Variation (CDV) — вариация задержки ячеек.

Параметры скорости измеряются в ячейках в секунду, максимальный размер пульсации — в ячейках, а временные параметры — в секундах. Максимальный размер пульсации определяет количество ячеек, которое приложение может передать с максимальной скоростью PCR, если задана средняя скорость. Доля потерянных ячеек является отношением потерянных ячеек к общему количеству отправленных ячеек по данному виртуальному соединению. Так как виртуальные соединения являются дуплексными, то для каждого направления соединения могут быть заданы разные значения параметров.

В технологии ATM принят не совсем традиционный подход к трактовке термина «качество обслуживания» (QoS). Обычно качество обслуживания трафика характеризуется параметрами пропускной способности (здесь это RCR, SCR, MCR, MBS), параметрами задержек пакетов (CTD и CDV), а также параметрами надежности передачи пакетов (CLR). В ATM характеристики пропускной способности называют параметрами трафика и не включают их в число параметров качества обслуживания QoS, хотя, по существу, они таковыми являются. Параметрами QoS в ATM являются только параметры CTD, CDV и CLR. Сеть старается обеспечить такой уровень услуг, чтобы поддерживались требуемые значения и для параметров трафика, и для задержек ячеек, и для доли потерянных ячеек.

Соглашение между приложением и сетью ATM называется трафик-контрактом. Основным его отличием от соглашений, применяемых в сетях frame relay, является выбор одного из нескольких определенных классов трафика, для которого наряду с параметрами пропускной способности трафика могут указываться параметры задержек ячеек, а также параметр надежности доставки ячеек. В сети frame relay класс трафика один, и он характеризуется только параметрами про-

пускной способности.

Необходимо подчеркнуть, что задание только параметров трафика (вместе с параметрами QoS) часто не полностью характеризует требуемую услугу, поэтому задание класса трафика полезно для уточнения нужного характера обслуживания данного соединения сетью.

В некоторых случаях специфика приложения такова, что ее трафик не может быть отнесен к одному из четырех стандартных классов. Для этого случая введен еще один класс X, который не имеет никаких дополнительных описаний, а полностью определяется теми количественными параметрами трафика и QoS, которые оговариваются в трафик-контракте.

Если для приложения не критично поддержание параметров пропускной способности и QoS, то оно может отказаться от задания этих параметров, указав признак «Best Effort» в запросе на установление соединения. Такой тип трафика получил название трафика с неопределенной битовой скоростью — Unspecified Bit Rate, UBR.

После заключения трафик-контракта, который относится к определенному и виртуальному соединению, в сети ATM работает несколько протоколов и служб, обеспечивающих нужное качество обслуживания. Для трафика UBR сеть выделяет ресурсы «по возможности», то есть те, которые в данный момент свободны от использования виртуальными соединениями, заказавшими определенные параметры качества обслуживания.

Технология ATM изначально разрабатывалась для поддержки как постоянных, так и коммутируемых виртуальных каналов (в отличие от технологии frame relay, долгое время не поддерживающей коммутируемые виртуальные каналы). Автоматическое заключение трафик-контракта при установлении коммутируемого виртуального соединения представляет собой весьма непростую задачу, так как коммутаторам ATM необходимо определить, смогут ли они в дальнейшем обеспечить передачу трафика данного виртуального канала наряду с трафиком других виртуальных каналов таким образом, чтобы выполнялись требования качества обслуживания каждого канала.

Стек протоколов ATM

Стек протоколов ATM показан на рис. 17.15, а распределение протоколов по конечным узлам и коммутаторам ATM — на рис. 17.16.

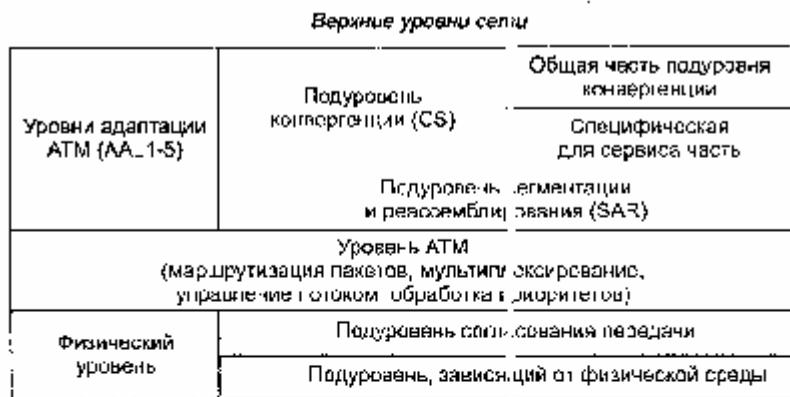


Рис. 17.15. Структура стека протоколов ATM

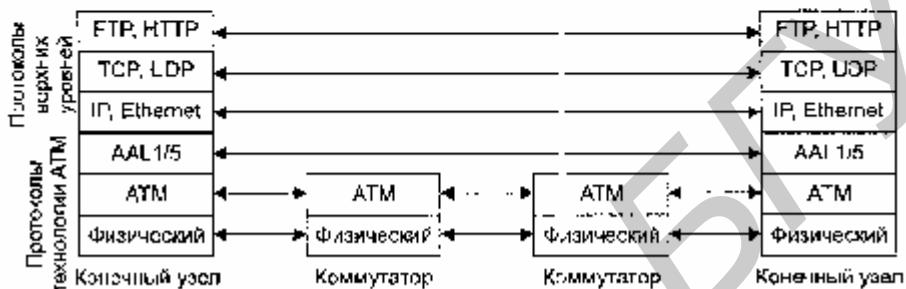


Рис. 17.16. Распределение протоколов по узлам и коммутаторам сети ATM

Стек протоколов ATM соответствует нижним уровням семиуровневой модели ISO/OSI и включает уровень адаптации ATM, собственно уровень ATM и физический уровень. Прямого соответствия между уровнями протоколов технологии ATM и уровнями модели OSI нет.

Уровень адаптации AAL

Уровень адаптации (ATM Adaptation Layer, AAL) представляет собой набор протоколов AAL1-AAL5, которые преобразуют сообщения протоколов верхних уровней сети ATM в ячейки ATM нужного формата. Функции этих уровней достаточно условно соответствуют функциям транспортного уровня модели OSI, например функциям протоколов TCP или UDP. Протоколы AAL при передаче пользовательского трафика работают только в конечных узлах сети, как и транспортные протоколы большинства технологий.

Каждый протокол уровня AAL обрабатывает пользовательский трафик определенного класса. На начальных этапах стандартизации каждому классу трафика соответствовал свой протокол AAL, который принимал в конечном узле пакеты от протокола верхнего уровня и заказывал с помощью соответствующего протокола нужные параметры трафика и качества обслуживания для данного виртуального канала. При развитии стандартов ATM такое однозначное соответствие между классами трафика и протоколами уровня AAL исчезло, и сегодня разрешается использовать для одного и того же класса трафика различные протоколы уровня AAL.

Уровень адаптации состоит из нескольких подуровней. Нижний подуровень

AAL называется подуровнем сегментации и реассемблирования (Segmentation And Reassembly, SAR). Эта часть не зависит от типа протокола AAL (и, соответственно, от класса передаваемого трафика) и занимается разбиением (сегментацией) сообщения, принимаемого AAL от протокола верхнего уровня, на ячейки ATM, снажением их соответствующим заголовком и передачей уровню ATM для отправки в сеть.

Верхний подуровень AAL называется подуровнем конвергенции (Convergence Sublayer, CS). Этот подуровень зависит от класса передаваемого трафика. Протокол подуровня конвергенции решает такие задачи, как обеспечение временной синхронизации между передающим и принимающим узлами (для трафика, требующего такой синхронизации), контролем и возможным восстановлением битовых ошибок в пользовательской информации, контролем целостности передаваемого пакета компьютерною протокола (X.25, frame relay).

Протоколы AAL для выполнения своей работы используют служебную информацию, размещаемую в заголовках уровня AAL. После приема ячеек, пришедших по виртуальному каналу, подуровень SAR протокола AAL собирает посланное по сети исходное сообщение (в общем случае разбитое на несколько ячеек ATM) с помощью заголовков AAL, которые для коммутаторов ATM являются прозрачными, так как помещаются в 48-битном поле данных ячейки, как и полагается протоколу более высокого уровня. После сборки исходного сообщения протокол AAL проверяет служебные поля заголовка и концевика кадра AAL и на их основании принимает решение о корректности полученной информации.

Ни один из протоколов AAL при передаче пользовательских данных конечных узлов не занимается восстановлением потерянных или искаженных данных. Максимум, что делает протокол AAL, — уведомляет конечный узел о таком событии. Так сделано для ускорения работы коммутаторов сети ATM в расчете на то, что случаи потерь или искажений данных будут редкими. Восстановление потерянных данных (или игнорирование этого события) отводится протоколам верхних уровней, не входящим в стек протоколов технологии ATM.

Протокол AAL1 обычно обслуживает трафик класса А с постоянной битовой скоростью (Constant Bit Rate, CBR), который характерен, например, для цифрового видео и цифровой речи и чувствителен к временным задержкам. Этот трафик передается в сетях ATM таким образом, чтобы эмулировать обычные выделенные цифровые линии. Заголовок AAL1 занимает в поле данных ячейки ATM один или два байта, оставляя для передачи пользовательских данных соответственно 47 или 46 байт. В заголовке один байт отводится для нумерации ячеек, чтобы приемная сторона могла судить о том, все или не все посланные ячейки дошли до нее. При отправке голосового трафика временная отметка каждого замера известна, так как они следуют друг за другом с интервалом в 125 мкс, поэтому при потере ячейки можно скорректировать временную привязку байтов следующей ячейки, сдвинув ее на 125×46 мкс. Потеря нескольких байтов замеров голоса не так страшна, так как на приемной стороне воспроизведение оборудования сглаживает сигнал. В задачи протокола AAL1 входит сглаживание неравномерности поступления ячеек данных в узел назначения.

Протокол AAL2 был разработан для передачи трафика класса В, но при развитии стандартов его исключили из стека протоколов ATM, и сегодня трафик класса В передается с помощью протокола AAL1, AAL3/4 или AAL5.

Протокол AAL3/4 обрабатывает пульсирующий трафик — обычно характерный для трафика локальных сетей — с переменной битовой скоростью (Variable Bit Rate, VBR). Этот трафик обрабатывается так, чтобы не допустить потерь ячеек, но ячейки могут задерживаться коммутатором. Протокол AAL3/4 выполняет сложную процедуру контроля ошибок при передаче ячеек, нумеруя каждую составляющую часть исходного сообщения и снабжая каждую ячейку контрольной суммой. Правда, при искажениях или потерях ячеек уровень не занимается их восстановлением, а просто отбрасывает все сообщение — то есть все оставшиеся ячейки, так как для компьютерного трафика или компрессированного голоса потеря части данных является фатальной ошибкой. Протокол AAL3/4 представляет собой результат слияния протоколов AAL3 и AAL4, которые обеспечивали поддержку трафика компьютерных сетей соответственно с установлением соединения и без установления соединения. Однако ввиду большой близости используемых форматов служебных заголовков и логики работы протоколы AAL3 и AAL4 были впоследствии объединены.

Протокол AAL5 является упрощенным вариантом протокола AAL4 и работает быстрее, так как вычисляет контрольную сумму не для каждой ячейки сообщения, а для всего исходного сообщения в целом и помещает ее в последнюю ячейку сообщения. Первоначально протокол AAL5 разрабатывался для передачи кадров сетей frame relay, но теперь он чаще всего применяется для передачи любого компьютерного трафика. Протокол AAL5 может поддерживать различные параметры качества обслуживания, кроме тех, которые связаны сシンхронизацией передающей и принимающей сторон. Поэтому он обычно используется для поддержки всех классов трафика, относящегося к передаче компьютерных данных, то есть классов С и D. Некоторые производители оборудования с помощью протокола AAL5 обслуживают трафик CBR, оставляя задачу синхронизации трафика протоколам верхнего уровня.

Протокол AAL5 работает не только в конечных узлах, но и в коммутаторах сети ATM. Однако там он выполняет служебные функции, не связанные с передачей пользовательских данных. В коммутаторах ATM протокол AAL5 поддерживает служебные протоколы более высоких уровней, занимающиеся установлением коммутируемых виртуальных соединений.

Существует определенный интерфейс между приложением, которому требуется передать трафик через сеть ATM, и уровнем адаптации AAL. С помощью этого интерфейса приложение (протокол компьютерной сети, модуль оцифровывания голоса) заказывает требуемую услугу, определяя тип трафика, его параметры, а также параметры QoS. Технология ATM допускает два варианта определения параметров QoS: первый — непосредственное задание их каждым приложением, второй — назначение их по умолчанию в зависимости от типа трафика. Последний способ упрощает задачу разработчика приложения, так как в этом случае выбор максимальных значений задержки доставки ячеек и вариации задержек перекладывается на плечи администратора сети.

Самостоятельно обеспечить требуемые параметры трафика и QoS протоколы AAL не могут. Для выполнения соглашений трафик-контракта требуется согласованная работа коммутаторов сети вдоль всего виртуального соединения. Эта работа выполняется протоколом ATM, обеспечивающим передачу ячеек различных виртуальных соединений с заданным уровнем качества обслуживания.

Протокол ATM

Протокол ATM занимает в стеке протоколов ATM примерно то же место, что протокол IP в стеке TCP/IP или протокол LAP-F в стеке протоколов технологии frame relay. Протокол ATM занимается передачей ячеек через коммутаторы при установленном и настроенном виртуальном соединении, то есть на основании готовых таблиц коммутации портов. Протокол ATM выполняет коммутацию по номеру виртуального соединения, который в технологии ATM разбит па две части — идентификатор виртуального пути (Virtual Path Identifier, VPI) и идентификатор виртуального канала (Virtual Channel Identifier, VCI). Кроме этой основной задачи протокол ATM выполняет ряд функций по контролю за соблюдением трафик-контракта со стороны пользователя сети, маркировке ячеек-нарушителей, отбрасыванию ячеек-нарушителей при перегрузке сети, а также управлению потоком ячеек для повышения производительности сети (естественно, при соблюдении условий трафик-контракта для всех виртуальных соединений).

Формат ячеек протокола ATM представлен на рис. 17.17.

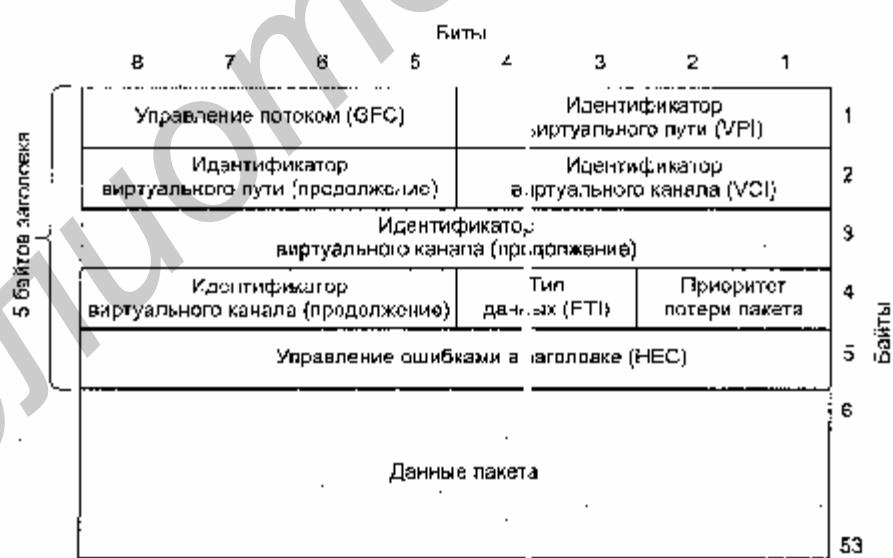


Рис. 17.17. Формат ячейки ATM

Поле Управление потоком (Generic Flow Control, GFC) используется только при взаимодействии конечного узла и первого коммутатора сети. В настоящее время его точные функции не определены.

Поля Идентификатор виртуального пути (Virtual Path Identifier, VPI) и Иденти-

фикатор виртуального канала (Virtual Channel Identifier, VCI) занимают соответственно 1 и 2 байта. Эти поля задают номер виртуального соединения, разделенный на старшую (VPI) и младшую (VCI) части.

Поле Идентификатор типа данных (Payload Type Identifier, PTI) состоит из 3 бит и задает тип данных, переносимых ячейкой, — пользовательские или управляющие (например, управляющие установлением виртуального соединения). Кроме того, один бит этого поля используется для указания перегрузки в сети — он называется Explicit Congestion Forward Identifier, EFCI — и играет ту же роль, что бит FECN в технологии frame relay, то есть передает информацию о перегрузке по направлению потока данных.

Поле Приоритет потери кадра (Cell Loss Priority, CLP) играет в данной технологии ту же роль, что и поле DE в технологии frame relay — и нем коммутаторы ATM отмечают ячейки, которые нарушают соглашения о параметрах качества обслуживания, чтобы удалить их при перегрузках сети. Таким образом, ячейки с полем CLP = 0 являются для сети высокоприоритетными, а ячейки с полем CLP = 1 — низкоприоритетными.

Поле Управление ошибками в заголовке (Header Error Control, HEC) содержит контрольную сумму, вычисленную для заголовка ячейки. Контрольная сумма вычисляется с помощью техники корректирующих кодов Хэмминга, поэтому она позволяет не только обнаруживать ошибки, но и исправлять все одиночные ошибки, а также некоторые двойные. Поле HEC обеспечивает не только обнаружение и исправление ошибок в заголовке, но и нахождение границы начала кадра в потоке байтов кадров SDH, которые являются предпочтительным физическим уровнем технологии ATM, или же в потоке битов физического уровня, основанного на ячейках. Указателей, позволяющих в поле данных кадра STS-n (STM-n) технологии SONET/SDH обнаруживать границы ячеек ATM (подобных тем указателям, которые используются для определения, например, границ виртуальных контейнеров подканалов T1/E1), не существует. Поэтому коммутатор ATM вычисляет контрольную сумму для последовательности из 5 байт, находящихся в поле данных кадра STM-n, и, если вычисленная контрольная сумма говорит о корректности заголовка ячейки ATM, первый байт становится границей ячейки. Если же это не так, то происходит сдвиг на один байт и операция продолжается. Таким образом, технология ATM выделяет асинхронный поток ячеек ATM в синхронных кадрах SDH или потоке битов физического уровня, основанного на ячейках.

Рассмотрим методы коммутации ячеек ATM на основе пары чисел VPI/VCI. Коммутаторы ATM могут работать в двух режимах — коммутации виртуального пути и коммутации виртуального канала. В первом режиме коммутатор выполняет продвижение ячейки только на основании значения поля VPI, а значение ноля VCI он игнорирует. Обычно так работают магистральные коммутаторы территориальных сетей. Они доставляют ячейки из одной сети пользователя в другую на основании только старшей части номера виртуального канала, что соответствует идеи агрегирования адресов. В результате один виртуальный путь соответствует целому набору виртуальных каналов, коммутируемых как единое целое.

После доставки ячейки в локальную сеть ATM ее коммутаторы начинают коммутировать ячейки с учетом как VPI, так и VCI, но при этом им хватает для коммутации только младшей части номера виртуального соединения, так что фактически они работают с VCI, оставляя VPI без изменения. Последний режим называется режимом коммутации виртуального канала.

Для создания коммутируемого виртуального канала в технологии ATM используются протоколы, не показанные на рис. 17.17. Подход здесь аналогичен подходу в сети ISDN — для установления соединения разработан отдельный протокол Q.2931, который весьма условно можно отнести к сетевому уровню. Этот протокол во многом похож на протоколы Q..931 и Q.933 (далее номером), но в него внесены, естественно, изменения, связанные с наличием нескольких классов трафика и дополнительных параметров качества обслуживания. Протокол Q.2931 опирается на достаточно сложный протокол канального уровня SSCOP, который обеспечивает надежную передачу пакетов Q.2931 в своих кадрах. В свою очередь, протокол SSCOP работает поверх протокола AAL5, который необходим для разбиения кадров SSCOP на ячейки ATM и сборки этих ячеек в кадры при доставке кадра SSCOP в коммутатор назначения.

ПРИМЕЧАНИЕ

Протокол Q.2931 появился в стеке протоколов технологии ATM после принятия версии интерфейса UNI 3.1, а до этого в версии UNI 3.0 вместо него использовался протокол Q.93B. Из-за несовместимости протоколов Q.2931 и Q.U3B версии пользовательского интерфейса UNI 3.0 и UNI 3.1 также несовместимы. Версия UNI 4.0 обратно совместима с UNI 3.1, так как основана на тех же служебных протоколах, что и версия UNI 3.1.

Виртуальные соединения, образованные с помощью протокола Q.2931, бывают симплексными (однонаправленными) и Дуплексными.

Протокол Q.2931 позволяет также устанавливать виртуальные соединения типа «один к одному» (point-to-point) и «один ко многим» (point-to-multipoint). Первый случай поддерживается во всех технологиях, основанных на виртуальных каналах, а второй характерен для технологии ATM и является аналогом мультивещания, но с одним ведущим вещающим узлом. При установлении соединения «один ко многим» ведущим считается узел, который является инициатором этого соединения. Сначала этот узел устанавливает виртуальное соединение всего с одним узлом, а затем путем специального вызова добавляет к соединению по одному новому члену. Ведущий узел становится вершиной дерева соединения, а остальные узлы — листьями этого дерева. Сообщения, которые посыпает ведущий узел, принимают все листья соединения, но сообщения, которые посыпает какой-либо лист (если соединение дуплексное), принимает только ведущий узел.

Пакеты протокола Q.2931, предназначенные для установления коммутируемого виртуального канала, имеют те же названия и то же назначение, что и пакеты протокола Q.933, рассмотренные выше при изучении технологии frame relay, но структура их полей, естественно, другая.

Адресом конечного узла в коммутаторах ATM является 20-байтный адрес. Этот адрес может иметь различный формат, описываемый стандартом ISO 7498. При работе в публичных сетях используется адрес стандарта E.164, при этом 1 байт составляет AFI, 8 байт занимает IDI — основная часть адреса E.164 (15 цифр телефонного номера), а остальные 11 байт части DSP (Domain Specific Part) распределяются следующим образом.

- 4 байта занимает поле старшей части DSP — High-Order Domain Specific Part (HO-DSP), имеющее гибкий формат и, в сущности, представляющее собой номер сети ATM, который может делиться на части для агрегированной маршрутизации по протоколу PNNI, подобной той, которая используется в технике CIDR для сетей IP.
- 6 байт занимает поле идентификатора конечной системы — End System Identifier (ESI), которое имеет смысл MAC-адреса узла ATM, причем формат его также соответствует формату MAC-адресов IEEE.
- 1 байт составляет поле селектора, которое не используется при установлении виртуального канала, а имеет для узла локальное назначение.

При работе в частных сетях ATM обычно применяется формат адреса, соответствующий домену международных организаций, причем в качестве международной организации выступает ATM Forum. В этом случае поле IDI занимает два байта, которые содержат код ATM Forum, данный ISO, а структура остальной части DSP соответствует описанной выше за исключением того, что и поле HO-DSP занимает не 4, а 10 байт.

Адрес ESI присваивается конечному узлу на предприятии-изготовителе в соответствии с правилами IEEE, то есть три первых байта содержат код предприятия, а остальные три байта — порядковый номер, за уникальность которого отвечает данное предприятие.

Конечный узел при подключении к коммутатору ATM выполняет так называемую процедуру регистрации. При этом конечный узел сообщает коммутатору свой ESI-адрес, а коммутатор сообщает конечному узлу старшую часть адреса, то есть номер сети, в которой работает узел.

Кроме адресной части пакет CALL SETUP протокола Q.2931, с помощью которого конечный узел запрашивает установление виртуального соединения, включает также части, описывающие параметры трафика и требования QoS. При поступлении такого пакета коммутатор должен проанализировать эти параметры и решить, достаточно ли у него свободных ресурсов производительности для обслуживания нового виртуального соединения. Если да, то новое виртуальное соединение принимается и коммутатор передает пакет CALL SETUP дальше в соответствии с адресом назначения и таблицей маршрутизации, а если нет, то запрос отвергается.

Категории услуг протокола ATM и управление трафиком

Для поддержания требуемого качества обслуживания различных виртуальных соединений и рационального использования ресурсов в сети на уровне прото-

кола ATM реализовано несколько служб, предоставляющих услуги различных категорий (service categories) по обслуживанию пользовательского трафика. Эти службы являются внутренними службами сети ATM, они предназначены для поддержания пользовательского трафика различных классов совместно с протоколами AAL. Но в отличие от протоколов AAL, которые работают в конечных узлах сети, данные службы распределены по всем коммутаторам сети. Услуги этих служб разбиты на категории, которые в общем соответствуют классам трафика, поступающим на вход уровня AAL конечного узла. Услуги уровня ATM заказываются конечным узлом через интерфейс UNI с помощью протокола Q.2931 при установлении виртуального соединения. Как и при обращении к уровню AAL, при заказе услуги необходимо указать категорию услуги, а также параметры трафика и параметры QoS. Эти параметры берутся из аналогичных параметров уровня AAL или же определяются по умолчанию в зависимости от категории услуги.

Всего на уровне протокола ATM определено пять категорий услуг, которые поддерживаются одноименными службами;

- CBR — услуги для трафика с постоянной битовой скоростью;
- rtVBR — услуги для трафика с переменной битовой скоростью, требующего соблюдения средней скорости передачи данных и синхронизации источника и приемника;
- nrtVBR — услуги для трафика с переменной битовой скоростью, требующего соблюдения средней скорости передачи данных и не требующего синхронизации источника и приемника;
- ABR — услуги для трафика с переменной битовой скоростью, требующего соблюдения некоторой минимальной скорости передачи данных и не требующего синхронизации источника и приемника;
- UBR — услуги для трафика, не предъявляющего требований к скорости передачи данных и синхронизации источника и приемника.

Названия большинства категорий услуг совпадают с названием типов пользовательского трафика, для обслуживания которого они разработаны, по необходимо понимать, что сами службы уровня ATM и их услуги — это внутренние механизмы сети ATM, которые экранируются от приложения уровнем AAL.

Услуги категории CBR предназначены для поддержания трафика синхронных приложений — голосового, эмуляции цифровых выделенных каналов и т. п. Когда приложение устанавливает соединение категории CBR, оно заказывает пиковую скорость трафика ячеек PCR, являющуюся максимальной скоростью, которую может поддерживать соединение без риска потерять ячейку, а также параметры QoS: величины максимальной задержки ячеек CTD, вариации задержки ячеек CDV и максимальной доли потерянных ячеек CLR.

Затем данные передаются по этому соединению с запрошенной скоростью — не с большей и, в большинстве случаев, не меньшей, хотя уменьшение скорости приложением возможно, например, при передаче компрессированного голоса с помощью услуги категории CBR. Любые ячейки, передаваемые станцией с большей скоростью, контролируются первым коммутатором сети и помечаются признаком CLP = 1. При перегрузках сети они могут просто отбрасываться се-

тью. Ячейки, которые запаздывают и не укладываются в интервал, оговоренный параметром идишн задержки CDV, также считаются мало значащими для приложения и отмечаются признаком низкого приоритета CLP = 1.

Для соединений CBR нет ограничений па некоторую дискретность заказа скорости PCR, как, например, в каналах T1/E1, где скорость должна быть кратна 64 кбит/с.

По сравнению со службой CBR, службы VBR требуют более сложной процедуры заказа соединения между сетью и приложением. В дополнение к пиковой скорости PCR приложение VBR заказывает еще и два других параметра: длительно поддерживаемую скорость — SCR, которая представляет собой среднюю скорость передачи данных, разрешенную приложению, а также максимальный размер пульсации — MBS. Максимальный размер пульсации измеряется в количестве ячеек: ATM. Пользователь может превышать скорость вплоть до величины PCR, но только па короткие периоды времени, в течение которых передается объем данных, не превышающий MBS. Этот период времени называется терпимостью к пульсации (Burst Tolerance, BT). Сеть вычисляет этот период как производный от трех заданных значений PCR, SCR и MBS.

Если скорость PCR наблюдается в течение периода времени, большего чем BT, то ячейки помечаются как нарушители — устанавливается признак CLP=1.

Для услуг категории rtVBR задаются и контролируются тс же параметры QoS, что и для услуг категории CBR, а услуги категории nrtVBR ограничиваются поддержанием параметров трафика. Сеть также поддерживает для обеих категорий услуг VBR определенный максимальный уровень доли потерянных ячеек CLR, который либо задается явно при установлении соединения, либо назначается по умолчанию в зависимости от класса трафика.

Для контроля параметров трафика и QoS в технология ATM применяется так называемый обобщенный алгоритм контроля скорости ячеек — Generic Cell Rate Algorithm, который может проверять соблюдение пользователем и сетью таких параметров, как PCR, CDV, SCR, BT, СТО и CDV. Он работает по модифицированному алгоритму «дырявого недра», применяемому в технологии frame relay.

Для многих приложений, которые могут быть чрезвычайно «взрывными» в отношении интенсивности трафика, невозможно точно предсказать параметры трафика, оговариваемые при установлении соединения. Например, обработка транзакций или трафик двух взаимодействующих локальных сетей непредсказуемы по своей природе — изменения интенсивности трафика слишком велики, чтобы заключить с сетью какое-либо разумное соглашение.

В отличие от CBR и обеих служб VBR, служба UBR не поддерживает ни параметры трафика, ни параметры качества обслуживания. Служба UBR предлагает только доставку «по возможности» без каких-либо гарантий. Разработанная специально для превышения полосы пропускания, служба UBR представляет собой частичное решение для тех непредсказуемых «взрывных» приложений, которые не готовы согласиться с фиксацией параметров трафика.

Главными недостатками услуг UBR являются отсутствие механизмов управления потоком данных и неспособность принимать во внимание другие типы

трафика. Несмотря на перегрузку сети, соединения UBR будут продолжать передачу данных. Коммутаторы сети могут буферизовать некоторые ячейки поступающего трафика, но в некоторый момент буфера переполняются, и ячейки теряются. А так как для соединений UBR не оговаривается никаких параметров трафика и QoS, то их ячейки отбрасываются в первую очередь.

Служба ABR подобно службе UBR предоставляет возможность превышения полосы пропускания, но благодаря технике управления трафиком при перегрузке сети она лает некоторые гарантии сохранности ячеек. ABR — это первый тип служб уровня ATM, который действительно обеспечивает надежный транспорт для пульсирующего трафика за счет того, что может находить неиспользуемые интервалы в общем трафике сети и заполнять их своими ячейками, если другим категориям служб ути интервалы не нужны.

Как и в службах CBR и VBR, при установлении соединения категории ABR оговаривается значение пиковой скорости PCR. Однако соглашение о пределах изменения задержки передачи ячеек или о параметрах пульсации не заключается. Вместо этого сеть и конечный узел заключают соглашение о требуемой минимальной скорости передачи MCR. Это гарантирует приложению, работающему в конечном узле, небольшую пропускную способность, обычно минимально необходимую для того, чтобы приложение работало. Конечный узел соглашается не передавать данные со скоростью, выше пиковой, то есть PCR, а сеть соглашается всегда обеспечивать минимальную скорость передачи ячеек MCR.

Если при установлении соединения ABR не задаются значения максимальной и минимальной скоростей, то по умолчанию считается, что PCR совпадает со скоростью линии доступа станции к сети, а MCR считается равной нулю.

Трафик соединения категории ABR получает гарантированное качество услуг в отношении доли потерянных ячеек и пропускной способности. Что касается задержек передачи ячеек, то хотя сеть и старается свести их к минимуму, но гарантий по этому параметру не дает. Следовательно, служба ABR не предназначена для приложений реального времени, а предназначена для приложений, в которых поток данных не очень чувствителен к задержкам в передаче,

При передаче трафика CBR, VBR и UBR явное управление перегрузками в сети отсутствует. Вместо этого используется механизм отбрасывания ячеек-нарушителей, а узлы, пользующиеся услугами CBR и VBR, стараются не нарушать условия контракта под угрозой потери ячеек, поэтому они обычно не действуют дополнительную пропускную способность, даже если она в данный момент доступна в сети.

Служба ABR позволяет воспользоваться резервами пропускной способности сети, так как сообщает конечному узлу о наличии в данный момент избыточной пропускной способности с помощью механизма обратной связи. Этот же механизм может помочь службе ABR снизить скорость передачи данных конечным узлом в сеть (вплоть до минимального значения MCR), если сеть испытывает перегрузку.

Узел, пользующийся услугами ABR, должен периодически посыпать в сеть парами с ячейками данных специальные служебные ячейки управления ресурсами

— Resource Management, RM. Ячейки RM, которые узел отправляет вдоль потока данных, называются прямыми ячейками RM — Forward Recource Management (FRM), а ячейки, которые идут в обратном направлении тен то к потоку данных направлении, насыпаются обратными ячейками RM — Backward Reconrce Management (BRM).

Существует несколько петель обратной связи. Самая простая петля обратной связи — между конечными станциями. При ее наличии коммутатор сети извещает конечную станцию о перегрузке с помощью специального флага в поле прямого управления перегрузками (флаг EFCI) ячейки данных, переносимой протоколом ATM. Затем конечная станция посылает через сеть сообщение, содержащееся в специальной ячейке управления BRM исходной станции, говоря ей о необходимости снизить скорость посылки ячеек в сеть.

В этом способе конечная станция несет основную ответственность за управление потоком, а коммутаторы играют пассивную роль в петле обратной связи, только уведомляя станцию-отправитель о перегрузке.

Такой простой способ имеет несколько очевидных недостатков. Конечная станция не узнает из сообщения BRM, на какую величину нужно уменьшить скорость передачи данных в сеть. Поэтому она просто понизит скорость до минимальной величины MCR, хотя, возможно, это и не обязательно. Кроме того, при большой протяженности сети коммутаторы должны продолжать буферизовать данные все время, пока уведомление о перегрузке будет путешествовать по сети, а для глобальных сетей это время может быть достаточно большим, и буфера могут переполниться, так что требуемый эффект достигнут не будет,

Разработаны и более сложные схемы управления потоком, в которых коммутаторы играют более активную роль, а узел-отправитель узнает более точно о возможной в данный момент скорости отправки данных в сеть.

В первой схеме узел-источник посылает в ячейке FRM явное значение скорости передачи данных в сеть, которую он хотел бы поддерживать в данное время. Каждый коммутатор, через который проходит по виртуальному пути это сообщение, может уменьшить запрашиваемую скорость до некоторой величины, которую он способен поддерживать в соответствии с имеющимися у него свободными ресурсами (или оставить запрашиваемую скорость без изменения). Узел назначения, получив ячейку FRM, превращает ее в ячейку 15RM и отправляет в обратном направлении, причем он тоже может уменьшить запрашиваемую скорость. Получив ответ в ячейке BRM, узел-источник точно знает, какая скорость отправки ячеек в сеть для него в данный момент доступна.

Во второй схеме каждый коммутатор сети может работать как узел-источник и узел назначения. Как узел-источник он может сам генерировать ячейки FRM и отправлять их по имеющимся виртуальным каналам. Как узел назначения он может отправлять на основе получаемых ячеек FRM ячейки BRM в обратном направлении. Такая схема является более быстродействующей и полезной в протяженных территориальных сетях.

Как видно из описания, служба ABR предназначена не только для прямого поддержания требований к обслуживанию конкретного виртуального соединения, но и для более рационального распределения ресурсов сети между ее абонентами.

тами, что в конечном итоге также приводит к повышению качества обслуживания всех абонентов сети.

Коммутаторы сети ATM используют различные механизмы для поддержания требуемого качества услуг. Кроме описанных в стандартах 1TU-T и ATM Forum механизмов заключения соглашения на основе параметров трафика и параметров QoS, а затем отбрасывания ячеек, не удовлетворяющих условиям соглашения, практически все производители оборудования ATM реализуют в своих коммутаторах несколько очередей ячеек, обслуживаемых с различными приоритетами.

Стратегия приоритетного обслуживания трафика основана на категориях услуг каждого виртуального соединения. До принятия спецификации ABR в большинстве коммутаторов ATM была реализована простая одноуровневая схема обслуживания, которая давала трафику CBR первый приоритет, трафику VBR второй, а трафику UBR — третий. При такой схеме комбинация CBR и VBR может потенциально заморозить трафик, обслуживаемый другим классом служб. Такая схема не будет правильно работать с трафиком ABR, так как не обеспечит его требования к минимальной скорости передачи ячеек. Для обеспечения этого требования должна быть выделена некоторая гарантированная полоса пропускания.

Чтобы поддерживать службу ABR, коммутаторы ATM должны реализовать двухуровневую схему обслуживания, которая бы удовлетворяла требованиям CBR, VBR и ABR. По этой схеме коммутатор предоставляет некоторую часть своей пропускной способности каждому классу служб. Трафик CBR получает часть пропускной способности, необходимую для поддержания пиковой скорости PCR, трафик VBR получает часть пропускной способности, необходимую для поддержания средней скорости SCR, а трафик ABR получает часть пропускной способности, достаточную для обеспечения требования минимальной скорости ячеек MCR. Это гарантирует, что каждое соединение может работать без потерь ячеек и не будет доставлять ячейки ABR за счет трафика CBR или VBR. На втором уровне этого алгоритма трафик CBR и VBR может забрать всю оставшуюся пропускную способность сети, если это необходимо, так как соединения ABR уже получили свою минимальную пропускную способность, которая им гарантировалась.

Отдельной задачей, которую нужно решать для поддержки корректной работы описанных служб и, соответственно, обеспечения заданного уровня QoS для всех классов трафика, является оптимизация работы сети ATM методами инженеринга трафика (TE). Использование в сетях ATM (как и в сетях frame relay) техники виртуальных каналов создает хорошие предпосылки для решения задачи TE. Однако каких-либо автоматизированных процедур для динамического выбора маршрутов следопатия виртуальных путей в целях обеспечения сбалансированной загрузки ресурсов сети пока не существует. Вся работа по оптимизации маршрутов должна выполняться предварительно, с помощью каких-либо внешних программных систем моделирования или оптимизации сети, а затем вручную реализовываться за счет установления постоянных виртуальных каналов PVC в соответствии с выбранными маршрутами. Тем не менее са-

ма возможность оптимизации загрузки сети путем рационального выбора путей следования виртуальных сетей является очень привлекательной, в особенности для операторов связи, которым необходимо соблюсти баланс между объемами передаваемого трафика и качеством предоставляемых услуг.

Выводы

- Технология ATM является дальнейшим развитием идей предварительного резервирования пропускной способности виртуального канала, реализованных в технологии frame relay.
- Технология ATM поддерживает основные типы трафика, существующие у абонентов разного типа: трафик с постоянной битовой скоростью CBR, характерный для телефонных сетей и сетей передачи изображения, трафик с переменной битовой скоростью VBR, характерный для компьютерных сетей, а также для передачи компрессированного голоса и изображения.
- Для каждого типа трафика пользователь может заказать у сети значения нескольких параметров качества обслуживания — максимальной битовой скорости PCR, средней битовой скорости SCR, максимальной пульсации MBS, а также контроля временных соотношений между передатчиком и приемником, важных для трафика, чувствительного к задержкам.
- Технология ATM сама не определяет новые стандарты для физического уровня, а пользуется существующими. Основным стандартом для ATM является физический уровень каналов технологий SONET/SDH и PDH.

«Чистые» сети IP

Общая структура глобальной сети IP

Революционный рост популярности Интернета привел к тому, что сегодня практически каждая глобальная сеть должна быть способна передавать трафик протокола IP. А это означает, что сегодня все глобальные сети являются составными сетями IP, а отличия между ними заключаются и лежащих под уровнем IP технологиях.

Сети IP изначально были задуманы как экономичные дейтаграммные сети, предоставляющие споим пользователям только услуги типа best effort. В результате классическая составная сеть IP не в состоянии гарантировать своим пользователям качество обслуживания их трафика, если только сеть не работает с очень низким уровнем загрузки, что в большинстве случаев неприемлемо. Процесс же по встраиванию в сети IP механизмов поддержки качества обслуживания пока еще далек от завершения. Хотя практически все маршрутизаторы ведущих производителей поддерживают сегодня такие базовые механизмы QoS, как кондиционирование трафика, приоритетное и взвешенное обслуживание очередей, созданию стройной законченной системы поддержки QoS в рамках протокола IP мешает его дейтаграммная природа. Действительно, неопределен-

ность путей следования пакетов при существовании альтернативных маршрутов равной стоимости (в отношении выбранной метрики) не позволяет выполнить точную оценку загруженности каждого ресурса сети, а значит, не позволяет применить в сетях IP методы инжиниринга трафика.

По так как трафик IP сегодня является непременным атрибутом любой сети передачи данных (и некоторых телефонных сетей тоже) и не поддерживать его просто невозможно, то для предоставления качества [пнях] услуг большинство крупных глобальных сетей, особенно сетей операторов спязи, строится по четырехуровневой схеме (рис. 17.18).

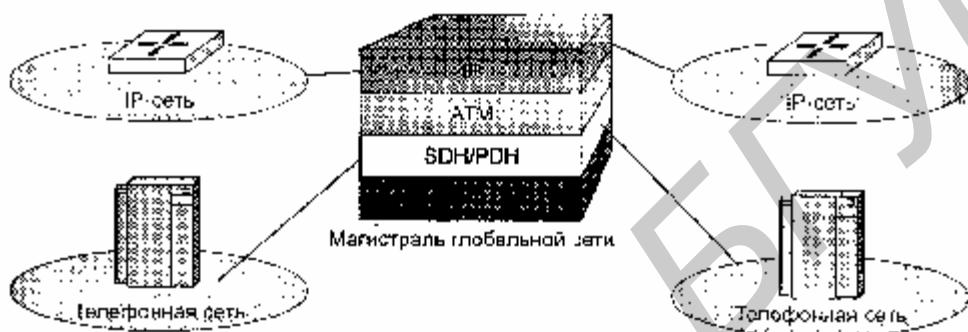


Рис. 17.18. Четырехуровневая структура современной глобальной сети

Два нижних уровня не относятся к собственно пакетным сетям — это уровни первичной сети, с помощью которой оператор сети может достаточно быстро организовать постоянный цифровой канал между точками подключения оборудования вышестоящей наложенной сети — пакетной или телефонной. На рисунке первичная сеть представлена двумя уровнями. На нижнем уровне работает наиболее скоростная на сегодняшний день технология DWDM, образующая спектральные скорости 10 Гбит/с и выше. На следующем уровне работает технология SDH (с сетью доступа PDH), с помощью которой пропускная способность спектральных каналов делится на более мелкие TDM-подканалы, связывающие интерфейсы коммутаторов пакетной сети (или телефонных коммутаторов). На рисунке показан наиболее масштабируемый на сегодня вариант построения первичной сети, включающий слои DWDM и SDH. Такое построение сегодня характерно только для наиболее крупных территориальных сетей, покрывающих страны и континенты. Во многих случаях слой DWDM отсутствует, технология SDH тоже применяется не всегда — вместо нее может работать менее скоростная и отказоустойчивая технология, но более экономичная технология PDH. В самом простом случае первичная сеть для образования постоянных каналов вообще отсутствует, и коммутаторы или маршрутизаторы пакетной сети соединяются непосредственно кабельными или беспроводными линиями связи. Последнее решение хоть и требует меньших начальных затрат, но страдает от недостатка гибкости — чтобы подключить новое устройство, необходимо физически прокладывать новую линию связи, в то время как наличие разветвленной первичной сети дает возможность установить новый канал в сети путем перепрограммирования матрицы коммутации мультиплексоров и

кросс-коннекторов.

Третий уровень образован сетью ATM, основным назначение которой является создание инфраструктуры постоянных виртуальных каналов, соединяющих интерфейсы маршрутизаторов IP, работающих на четвертом, верхнем уровне глобальной сети. Для каждого типа трафика образуется отдельный виртуальный канал, обеспечивающий требуемые для трафика параметры QoS — среднюю скорость, пульсацию, уровень задержек, уровень потерь.

Уровень IP, освобожденный в представленной модели от проблем обеспечения параметров QoS, выполняет свои классические функции — образует составную сеть и обеспечивает услуги конечным пользователям, передающим по глобальной сети слой IP-трафик транзитом или взаимодействующим по IP с Интернетом.

Несмотря на сложность многослойной структуры, подобные сети получили большое распространение и для крупных операторов комплексных услуг являются на сегодня фактическим эталоном глобальной сети, с помощью которой можно оказывать комплексные услуги — как IP, так и ATM, классической телефонии и услуги; предоставления цифровых каналов в аренду.

Тем не менее остаются востребованными и «чистые» сети IP, называемые так из-за того, что под уровнем IP нет другой сети с коммутацией пакетов, такой как ATM (которую может заменять frame relay в сетях небольшого масштаба). Структура «чистой» сети IP представлена на рис. 17.19.



Рис. 17.19. Структура «чистой» сети IP

В такой сети цифровые каналы по-прежнему образуются инфраструктурой двух нижних уровней, а этими каналами непосредственно пользуются интерфейсы маршрутизаторов IP, без какого-либо промежуточного слоя. Некоторые элементы поддержки QoS возможны и в такой сети, если на маршрутизаторах активизированы механизмы кондиционирования трафика, контролирующие входящие потоки таким образом, чтобы их интенсивность не превысила заранее оговоренные пределы, а также механизмы управления очередями. Однако маршруты трафика в такой сети выбираются классическими протоколами маршрутизации, а, значит, вес пакеты следуют к сетям назначения по кратчайшим маршрутам, то есть не эффективно, поэтому высокого качества обслуживания достичь здесь не удастся.

Тем не менее, в тех случаях, когда сеть передает трафик, не чувствительный к задержкам; «чистая» сеть IP может оказаться вполне рациональным решением, к тому же более экономичным и простым в эксплуатации, чем сложная четырехуровневая модель, требующая для своего обслуживания еще и специалистов

по технологии АТМ.

Однако для того, чтобы маршрутизаторы ТР могли использовать цифровые каналы, на этих каналах должен работать какой-либо протокол канального уровня, так как сам канал является устройством физического уровня. Далее рассматриваются наиболее популярные на сегодня протоколы канального уровня для каналов «точка-точка», поверх которых может работать протокол IP.

Протоколы канального уровня для выделенных линий

Если выделенный канал соединяет сети через маршрутизаторы, то протокол сетевого уровня определен, а в качестве протокола канального уровня маршрутизатор может использовать любой протокол, в том числе и протокол канального уровня локальной сети, например Ethernet. Мост должен передавать кадры канального протокола из одной локальной сети в другую, при этом ему тоже можно непосредственно задействовать протокол локальной сети (Ethernet, Token Ring, FDDI) поверх физического уровня канала.

До недавнего времени протокол Ethernet практически не использовался на выделенных каналах глобальных пакетных сетей, однако сегодня ситуация меняется. С появлением стандартов Gigabit Ethernet и особенно 10 Gigabit Ethernet (Ethernet со скоростью передачи 10 Гбит/с) эта технология, первоначально предназначенная исключительно для локальных сетей, будет находить все большее применение и в глобальных.

Однако существует ряд протоколов канального уровня, специально разработанных для связей «точка-точка» в глобальных сетях. В эти протоколы встроен ряд процедур, полезных при работе в глобальных сетях. В этот перечень входят: процедура управления потоком данных, процедура взаимной аутентификации удаленных устройств (часто необходимая для защиты сети от «подставного» маршрутизатора или моста, отводящего корпоративный трафик не по назначению), а также процедуры согласования параметров обмена данными на канальном и системном уровнях. При удаленном взаимодействии, когда два устройства расположены в разных городах, полезно автоматически согласовывать перед началом обмена пользовательскими данными такие параметры, как, например, максимальный размер поля данных (MTU) или IP-адрес партнера (как для безопасности, так и для автоматического конфигурирования стека TCP/IP на удаленных одиночных компьютерах).

Сегодня для работы протокола IP могут использоваться такие протоколы канального уровня, как SLIP, HDLC, PPP.

Протокол SLIP

Протокол SLIP (Serial Line IP) был первым стандартом де-факто, позволяющим устройствам, соединенным последовательной линией связи, работать по протоколам TCP/IP. Он был создан в начале 80-х годов и в 1981 году встроен Риком Адамсом (Rick Adams) в операционную систему 4.2 Berkley UNIX. Позднее

SLIP был поддержан в других версиях UNIX и реализован и программном обеспечении для ПК.

Правда, ввиду его функциональной простоты, SLIP использовался и используется в основном на коммутируемых линиях связи, которые не характерны для ответственных и скоростных сетевых соединений. Тем не менее коммутируемый канал отличается от искоммутируемого только более низким качеством и необходимостью выполнять процедуру вызова абонента, поэтому SLIP вполне применим и на выделенных каналах.

Протокол SLIP выполняет единственную функцию — он позволяет в потоке битов, которые поступают на выделенному (или коммутируемому) каналу, распознать начало и конец IP-пакета. Помимо протокола IP, другие протоколы сетевого уровня SLIP не поддерживает.

Чтобы распознать границы IP-пакетов, протокол SLIP предусматривает использование специального символа END, значение которого в шестнадцатеричном представлении равно С0. Применение специального символа может породить конфликт: если байт пересылаемых данных тождественен символу END, то он будет ошибочно определен как признак конца пакета. Чтобы предотвратить такую ситуацию, байт данных со значением, равным значению символа END, заменяется составной двухбайтовой последовательностью, состоящей из специального символа ESC (DB) и кода DC. Если же байт данных имеет тот же код, что и символ SLIP ESC, то он заменяется двухбайтовой последовательностью, состоящей из собственно символа SLIP ESC и кода DD. После последнего байта пакета передается символ END.

Механизм формирования составных последовательностей показан на рис. 17.20. Здесь приведены стандартный IP-пакет (один байт которого тождественен символу END, а другой — символу SLIP ESC) и соответствующий ему SLIP-пакет, который больше на 4 байта.

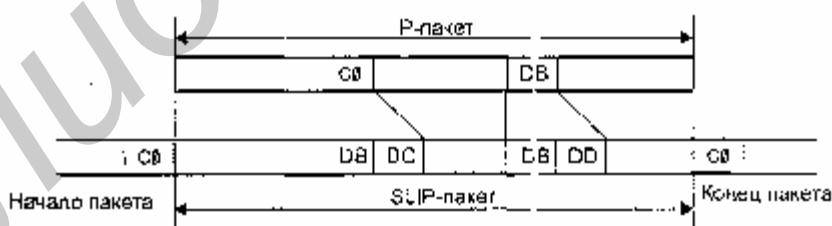


Рис. 17.20. Инакапсуляция IP-пакетов в SLIP-пакеты

Хотя в спецификации протокола SLIP не определена максимальная длина передаваемого пакета, реальный размер 1P пакета не должен превышать 1006 байт. Данное ограничение связано с первой реализацией протокола SLIP в соответствующем драйвере для Berkley UNIX, и его соблюдение необходимо для совместимости разных реализаций SLIP (большинство современных реализаций позволяют администратору самому установить размер пакета, а по умолчанию используют размер 1500 байт).

Для установления связи по протоколу SUP компьютеры должны иметь инфор-

мацию об IP-адресах друг друга. Однако возможна ситуация, когда, скажем, при осуществлении соединения между хостом и маршрутизатором последнему понадобится передать хосту информацию о его IP-адресе. В протоколе SLIP нет механизмов, дающих возможность обмениваться адресной информацией. Это ограничение не позволяет использовать SLIP для некоторых видов сетевых служб.

Другой недостаток SLIP - отсутствие индикации типа протокола, пакет которого инкапсулируется в SLIP-пакет. Поэтому через последовательную линию по протоколу SLIP можно передавать трафик лишь одного сетевого протокола — IP.

При работе с реальными телефонными линиями, загруженными и поэтому искажающими пакеты при пересылке, требуются процедуры обнаружения и коррекции ошибок. В протоколе SLIP такие процедуры не предусмотрены. Эти функции обеспечивают вышеперечисленные протоколы: протокол IP проводит тестирование целостности пакета по заголовку IP, а один из двух транспортных протоколов (UDP или TCP) проверяет целостность всех данных по контрольным суммам.

Низкая пропускная способность последовательных линий связи вынуждает сокращать время передачи пакетов, снижая объем содержащейся в них служебной информации. Эта задача решается с помощью протокола Compressed SLIP (CSLIP), поддерживающего сжатие заголовков пакетов. Появление CSLIP объясняется тем фактом, что при использовании программ типа Telnet, Rlogin и других для пересылки одного байта данных требуется передать 20-байтовый заголовок IP-пакета и 20-байтовый заголовок TCP-пакета (итого 40 байт). Спецификация CSLIP обеспечивает сжатие 40-байтового заголовка до 3-5 байт. На сегодняшний момент большинство реализаций протокола SLIP поддерживают спецификацию CSLIP.

Таким образом, протокол SLIP выполняет работу по выделению из последовательности передаваемых по последовательному каналу битов границ IP-пакета. Протокол не имеет механизмов передачи адресной информации, идентификации типа протокола сетевого уровня, определения и коррекции ошибок.

Протоколы семейства HDLC

Долгое время основным протоколом выделенных линий для глобальных пакетных сетей был протокол HDLC (High-level Data Link Control), имеющий статус стандарта ISO. Протокол HDLC на самом деле представляет собой семейство протоколов, в которое входят известные протоколы: LAP-B, образующий канальный уровень сетей X.25, LAP-D — канальный уровень сетей ISDN, LAP-M — канальный уровень асинхронно-синхронных модемов, LAP-E — канальный уровень сетей frame relay.

Основные принципы работы протокола ITDLC (режим логического соединения, контроль искаженных и потерянных кадров по методу скользящего окна, управление потоком кадров с помощью команд RNR и RR, а также различные

типы кадров этого протокола) уже были рассмотрены и главе 7 при изучении еще одного представителя семейства HDLC — протокола LLC2.

Однако сегодня протокол HDLC на выделенных каналах вытеснил протокол «точка-точка» (Point-to-Point Protocol, PPP).

Дело в том, что одна из основных функций протокола HDLC — это восстановление искаженных и утерянных кадров. Действительно, применение протокола HDLC обеспечивает снижение вероятности искажения бита (BER) с КГ'1, что характерно для территориальных аналоговых каналов, до 10~!).

Однако сегодня популярны цифровые каналы, которые и без внешних процедур восстановления кадров обладают высоким качеством (величина BER составляет КГ8-10). Для работы по такому каналу восстановительные функции протокола HDLC не нужны. При передаче по аналоговым выделенным каналам современные модемы сами применяют протоколы семейства HDLC (синхронные модемы — HDLC, а асинхронно-синхронные с асинхронным интерфейсом — LAP-M, который также принадлежит семейству HDLC). Поэтому использование HDLC на уровне маршрутизатора или моста становится неоправданным.

Протокол PPP

Протокол PPP (Point-to-Point Protocol) разработан группой IETF (Internet Engineering Task Force) как часть стека TCP/IP для передачи кадров информации по последовательным глобальным каналам связи взамен устаревшего протокола SLIP (Serial Line TP). Протокол PPP стал фактическим стандартом для глобальных линий связи при соединении удаленных клиентов с серверами и для образования соединений между маршрутизаторами в корпоративной сети. При разработке протокола PPP за основу был взят формат кадров HDLC и дополнен собственными полями. Поля протокола PPP вложены в поле данных кадра HDLC. Позже были разработаны стандарты, использующие вложение кадра PPP в кадры frame relay и других протоколов глобальных сетей.

Основное отличие PPP от других протоколов канального уровня состоит в том, что он добивается согласованной работы различных устройств с помощью переговорной процедуры, во время которой передаются различные параметры, такие как качество линии, протокол аутентификации и инкапсулируемые протоколы сетевого уровня. Переговорная процедура происходит во время установления соединения.

Протокол PPP основан на четырех принципах: переговорное принятие параметров соединения, многопротокольная поддержка, расширяемость протокола, независимость от глобальных служб.

Переговорное принятие параметров соединения. В корпоративной сети конечные системы часто отличаются размерами буферов для временного хранения пакетов, ограничениями по размер пакета, списком поддерживаемых протоколов сетевого уровня. Физическая линия, связывающая конечные устройства, может варьироваться от низкоскоростной аналоговой линии до высокоскоростной цифровой линии с различными уровнями качества обслуживания.

Чтобы справиться со всеми возможными ситуациями, в протоколе PPP имеется набор стандартных установок, действующих по умолчанию и учитывающих вес стандартные конфигурации. При установлении соединения два взаимодействующих устройства для нахождения взаимопонимания пытаются сначала использовать эти установки. Каждый конечный узел описывает свои возможности и требования. Затем на основании этой информации принимаются параметры соединения, устраивающие обе стороны, в которые входят форматы инкапсуляции данных, размеры пакетов, качество линии и процедура аутентификации.

Протокол, в соответствии с которым принимаются параметры соединения, называется протоколом управления связью (Link Control Protocol, LCP). Протокол, который позволяет конечным узлам договориться о том, какие сетевые протоколы будут передаваться в установленном соединении, называется протоколом управления сетевым уровнем (Network Control Protocol, NCP). Внутри одного PPP-соединения могут передаваться потоки данных различных сетевых протоколов.

Одним из важных параметров PPP-соединения является режим аутентификации. Для целей аутентификации PPP предлагает по умолчанию протокол PAP (Password Authentication Protocol), передающий пароль по линии связи в открытом виде, или протокол CHAP (Challenge Handshake Authentication Protocol), не передающий пароль по линии связи и поэтому обеспечивающий большую безопасность сети. Пользователям также разрешается добавлять и новые алгоритмы аутентификации. Дисциплина выбора алгоритмов компрессии заголовка и данных аналогична.

Многопротокольная поддержка — способность протокола PPP поддерживать несколько протоколов сетевого уровня — обусловила распространение PPP как стандарта де-факто. В отличие от протокола SLIP, который может переносить только IP-пакеты, или LAP-B, который может переносить только пакеты X.25, PPP работает со многими протоколами сетевого уровня, включая IP, Novell IPX, AppleTalk, DECnet, XNS, Banyan VINES и OSI, а также протоколами канального уровня локальной сети. Каждый протокол сетевого уровня конфигурируется отдельно с помощью соответствующего протокола NCP. Под конфигурированием понимается, во-первых, констатации того факта, что данный протокол будет использоваться в текущем сеансе PPP, а во-вторых, переговорное утверждение некоторых параметров протокола. Больше всего параметров устанавливается для протокола IP — IP-адрес узла, IP-адрес серверов DNS, наличие компрессии заголовка IP-пакета и т. д. Протоколы конфигурирования параметров соответствующего протокола верхнего уровня называются по имени этого протокола с добавлением аббревиатуры CP (Control Protocol), например протокол TPCP, IPXCP и т. п.

Расширяемость протокола. Под расширяемостью понимается как возможность включения новых протоколов в стек PPP, так и возможность использования собственных протоколов пользователей вместо рекомендуемых в PPP по умолчанию. Это позволяет наилучшим образом настроить PPP для каждой конкретной ситуации.

Независимость от глобальных служб. Начальная версия PPP работала только с

кадрами HDLC. Теперь в стек PPP добавлены спецификации, позволяющие использовать PPP в любой технологии глобальных сетей, например ISDN, frame relay, X.25, Sonct и HDLC.

Переговорная процедура протоколов LCP и NCP может и не завершиться соглашением о каком-нибудь параметре. Если, например, один узел предлагает в качестве MTU значение 1000 бант, а другой отвергает это предложение и в свою очередь предлагает значение 1500 бант, которое отвергается первым узлом, то по истечении тайм-аута переговорная процедура может закончиться безрезультатно.

Возникает вопрос, каким образом два устройства, ведущих переговоры по протоколу PPP, узнают о тех параметрах, которые они предлагают своему партнеру? Обычно у реализации протокола PPP есть некоторый набор параметров по умолчанию, которые и используются в переговорах. Тем не менее каждое устройство (и программа, реализующая протокол PPP в операционной системе компьютера) позволяет администратору изменить параметры по умолчанию, а также задать параметры, которые не входят в стандартный набор. Например, IP-адрес для удаленного узла отсутствует в параметрах по умолчанию, но администратор может задать его для сервера удаленного доступа, после чего сервер будет предлагать его удаленному узлу.

Хотя протокол PPP и работает с кадром HDLC, но в нем отсутствуют процедуры контроля кадров и управления потоком протокола НИМС. Поэтому в PPP используется только один тип кадра HDLC — ненумерованный информационный. В поле управления такого кадра всегда содержится величина 03. Для исправления очень редких ошибок, возникающих в канале, необходимы протоколы верхних уровней - TCP, SPX, NctBUEI, NCP и т. п.

Одной из возможностей протокола PPP является возможность использования нескольких физических линий для образования одного логического канала, так называемый транкинг каналов. Эту возможность реализует дополнительный протокол, который носит название MLPPP (Muhi Link PPP). Многие производители поддерживают такое свойство в своих маршрутизаторах и серверах удаленного доступа фирменным способом. Использование стандартного способа всегда лучше, так как он гарантирует совместимость оборудования разных производителей.

Общий логический канал может состоять из каналов разной физической природы. Например, один канал может быть образован в телефонной сети, а другой являться виртуальным коммутируемым каналом сети frame relay.

Использование выделенных линий для построения пакетной сети

Для связи двух локальных сетей по арендуемому или собственному выделенному каналу обычно используются мосты или маршрутизаторы. Эти устройства нужны для того, чтобы по выделенному каналу пересыпались не все кадры, циркулирующие в каждой локальной сети, а только те, которые предназначены для другой локальной сети.

Схема установки моста или маршрутизатора в этом случае однотипна (рис. 17.21). Сначала необходимо решить проблему физического сопряжения выходного порта моста или маршрутизатора с аппаратурой передачи данных, то есть ВСЕ, подключаемой непосредственно к абонентскому окончанию линии. Если канал аналоговый, то это интерфейс с модемом, а если цифровой — то с устройством DSU/CSU. Интерфейс определяется требованиями ПСЕ — это может быть RS-232C для низкоскоростных линий или же RS-449 или V.35 для высокоскоростных каналов типа Т1/E1. Для канала Т3/E3 потребуется наличие интерфейса HSS1. Если же маршрутизатор работает на каналы SI3H, то он обычно имеет встроенный порт с интерфейсом SDH определенной скорости STM-N. Для такого случая, когда пакеты IP передаются (с помощью канального протокола PPP или HDLC) непосредственно по каналам ЗПН, технология получила название *Packet Over SONET/SDH, POS*.

Некоторые устройства имеют программно настраиваемые последовательные интерфейсы, которые могут работать и как RS-449/V.11, и как RS-449/V.10, и как V.35.



Рис. 17.21. Соединение сетей с помощью выделенного канала

Встроенные порты E1 -E4 или STM-N могут как поддерживать, так и не поддерживать внутреннюю структуру кадров этих технологий. В том случае, когда порт различает подкадры, из которых состоит кадр — например, отдельные тайм-слоты кадра E1 или отдельные виртуальные контейнеры VC-12 (2 Мбит/с), входящие в кадр STM-1, и может использовать их как отдельные физические подканалы, то говорят, что это порт с разделением каналов (channelized). Каждому такому каналу присваивается отдельный IP-адрес. В противном случае порт целиком рассматривается как один физический канал с одним IP-адресом.

В качестве примера на рис. 17.21 выбрано соединение через цифровой канал E1, поэтому мост или маршрутизатор использует для подключения к каналу устройство DSU/CSU с внутренним интерфейсом RS-4-19 и внешним интерфейсом G.703. Часто крупные маршрутизаторы имеют модули со встроенным интерфейсом G.703, тогда необходимость в устройстве DSU/CSU отпадает. Если же выделенный канал был аналоговым, то в качестве DCE был бы необходим модем, поддерживающий режим работы по выделенной линии, причем

кроме других критериев (скорость, контроль ошибок, компрессия) необходимо учитывать возможность модема работать по предоставленному абонентскому окончанию: 4-проводному или 2-проводному.

После решения проблем физического уровня удаленные мосты готовы к работе. После включения каждый мост начинает передавать все кадры из всей локальной сети в выделенный канал и одновременно (так как практически все выделенные каналы дуплексные) принимать кадры из выделенного канала. На основании проходящего трафика каждый мост строит адресную таблицу и начинает передавать в выделенный канал кадры только тем станциям, которые действительно находятся в другой сети, а также широковещательные кадры и кадры с неизвестными MAC-адресами. Современные удаленные мосты при пересылке кадров локальных сетей упаковывают их в кадры протокола PPP, Переговорная процедура, которую ведут мосты при установлении PPP-соединения, сводится в основном к выбору параметров канального уровня с помощью протокола T>PC, а также к взаимной аутентификации (если такая процедура задана в параметрах протокола PPP обоих мостов).

Маршрутизатор после подключения к выделенной линии и локальной сети необходимо конфигурировать. На рис. 17.21 IP-маршрутизаторы связаны по выделенному каналу. Конфигурирование маршрутизаторов в этом случае подобно конфигурированию в локальных сетях. Каждая локальная сеть получает свой IP-адрес с соответствующей маской. Выделенный канал также является отдельной IP-сетью, поэтому можно ему также дать некоторый IP-адрес из диапазона адресов, которым распоряжается администратор корпоративной сети (в данном случае выделенному каналу присвоен адрес сети, состоящей из двух узлов, что определяется маской 255.255.255.252). Можно выделенному каналу и не присваивать IP-адрес — такой интерфейс маршрутизатора называется неиумерпированным (unnumbered). Маршрутизатор будет нормально работать в обоих случаях. Как и из локальной сети, маршрутизаторам не нужно вручную задавать аппаратные адреса своих непосредственных соседей, так как отсылая пакеты протокола маршрутизации (RIP или OSPF) по выделенному каналу, маршрутизаторы будут их получать без проблем. Протокол ЛИР на выделенном канале не используется, так как аппаратные адреса на выделенном канале не имеют практического смысла (в кадре PPP есть два адреса — кадр от DCE или от DTE, но маршрутизатор всегда будет получать кадр от DCE).

Как и в локальных сетях, важной характеристикой удаленных мостов или маршрутизаторов является скорость фильтрации и скорость маршрутизации пакетов, которые часто ограничиваются не внутренними возможностями устройства, а скоростью передачи данных по линии. Для устойчивой работы сети скорость маршрутизации устройства должна быть выше, чем средняя скорость межсетевого трафика. При объединении сетей с помощью выделенного канала рекомендуется сначала выяснить характер межсетевого трафика — его среднее значение и пульсацию. Для хорошей передачи пульсаций пропускная способность канала должна быть большей или равной величине пульсации трафика. Но такой подход приводит к очень нерациональной загрузке канала, так как при коэффициенте пульсаций 50:1 в среднем будет использоваться только 1/50 про-

пускной способности канала. Поэтому чаще при выборе канала ориентируются на среднее значение межсетевого трафика. Правда, при этом пульсация будет создавать очередь кадров во внутреннем буфере моста или маршрутизатора, так как канал не может передавать данные с такой высокой скоростью, но очередь обязательно рассосется за конечное время, если среднее значение интенсивности межсетевого трафика меньше средней пропускной способности канала.

Для преодоления ограничений на скорость линии, а также для уменьшения части локального трафика, передаваемого по глобальной линии, в удаленных мостах и маршрутизаторах, работающих па глобальные каналы, используются специальные приемы, не применяемые в локальных устройствах. Эти приемы не входят в стандарты протоколов, но они реализованы практически во всех устройствах, обслуживающих низкоскоростные каналы, особенно каналы со скоростями в диапазоне от 9600 бит/с до 6-1 кбит/с.

К таким приемам относятся технологии сжатия пакетов, спуфинга и сегментации пакетов.

Сжатие пакетов (компрессия). Некоторые производители, используя собственные алгоритмы, обеспечивают коэффициент сжатия до 8:1. Стандартные алгоритмы сжатия, применяемые в модемах, устройствах DSU/CSU, самих мостах и маршрутизаторах, обеспечивают коэффициент сжатия до 4:1. После сжатия данных для передачи требуется существенно меньшая скорость канала.

Спуфинг (spoofing). Эта технология позволяет значительно повысить пропускную способность линий, объединяющих локальные сети, работающие по протоколам с большим количеством широковещательных расылок. Во многих стеках протоколом для локальных сетей широковещательные рассылки обеспечивают решение задач поиска ресурсов сети. «Спуфинг» означает надувательство, мистификацию. Главной идеей технологии спуфинга является имитация передачи пакета по глобальной сети. Спуфинг используется не только на выделенных каналах, но и на коммутируемых, а также всегда, когда пропускная способность глобальной сети оказывается на границе некоторого минимального уровня.

Рассмотрим технику спуфинга на примере передачи между удаленными сетями пакетов SAP (Service Advertising Protocol — протокол объявления служб) серверами ОС NetWare. Эти пакеты каждый сервер генерирует каждую минуту, чтобы все клиенты сети могли составить правильное представление об имеющихся в сети разделяемых ресурсах — файловых службах, службах печати и т. п. SAP-пакеты распространяются в IPX-пакетах с широкопрещательным сетевым адресом (ограниченное широковещание). Маршрутизаторы не должны передавать такие пакеты из сети в сеть, но для SAP-пакетов сделано исключение — маршрутизатор, поддерживающий IPX, распространяет его на все порты, кроме того, на который этот пакет поступил (техника, подобная технике «расщепления горизонта»). Это делается для того, чтобы клиенты работали в одинаковых условиях независимо от сети, в которой они находятся. Удаленные мосты передают SAP-пакеты «по долгу службы», так как они имеют широковещательные MAC-адреса.

Таким образом, по выделенной линии может проходить достаточно большое

количество SAP-пакетов, которое зависит от количества серверов в каждой из локальных сетей, а также количества служб, о которых объявляет каждый сервер. Если эти пакеты посылаются каким-либо сервером, но не доходят до клиентов, то клиенты не могут воспользоваться службами этого сервера.

Если маршрутизаторы или мосты, объединяющие сети, поддерживают технику спуфинга, то они передают по выделенному каналу не каждый SAP-пакет, а, например, только каждый пятый. Интенсивность служебного трафика в выделенном канале при этом уменьшается. Но для того чтобы клиенты не теряли из списка ресурсов удаленной сети серверы, маршрутизатор или мост имитирует приход этих пакетов по выделенному каналу, посылая SAP-пакеты от своего имени каждую минуту, как это и положено по протоколу. При этом маршрутизатор или мост посылает несколько раз копию реального SAP-пакета, получаемого раз в 5 минут по выделенному каналу. Такую процедуру маршрутизатор или мост осуществляет для каждого сервера удаленной сети, генерирующего SAP-пакеты.

Существует несколько различных реализаций техники спуфинга: посылка оригинальных пакетов в глобальный канал происходит по времени или по количеству принятых пакетом, при изменениях в содержимом пакетов. Последний способ достаточно логичен, так как сервер обычно каждый раз повторяет содержимое своего объявления — изменения в составе служб происходят редко. Поэтому, как в алгоритмах маршрутизации типа «изменение связей» достаточно передавать только измененные пакеты, так и для подтверждения нормальной работы достаточно периодически пересыпать даже неизмененный пакет (в качестве сообщения HELLO).

Существует достаточно много протоколов, которые пользуются широковещательными рассылками, и пограничный маршрутизатор или мост должен их все учитывать. Только ОС UNIX весьма редко работает по этому способу, так как ее основной коммуникационный стек TCP/IP проектировался для низкоскоростных глобальных линий связи. Л такие ОС, как NetWare, Windows, OS/2, разрабатывались в основном в расчете на локальные сети, поэтому пропускную способность каналов связи не экономили.

В ОС NetWare существуют три основных типа широковещательных межсетевых сообщений — кроме сообщений SAP, необходимо еще передавать сообщения протокола маршрутизации RIP, который программные маршрутизаторы, работающие на серверах NetWare, поддерживают по умолчанию, а также специальные сообщения watchdogs (называемые также keep alive), которыми обмениваются сервер и клиент, установившие логическое соединение. Сообщения watchdogs используются в том случае, когда временно в рамках данного логического сеанса пользовательские данные не передаются. Чтобы поддержать соединение, клиент каждые 5 минут посылает такие сообщения серверу, говоря, что он «жив». Если сервер не получает таких сообщений в течение 15 минут, то сеанс с данным клиентом прекращается. В интерфейсе NetBIOS (а его используют в качестве программного интерфейса приложения во многих ОС) порождается служебный трафик разрешения имен — запросы Name-Query посылаются (также широковещательным способом) каждые 20 минут, если зарегистриро-

ванное ранее имя не проявило себя в течение этого периода времени. Для реализации анализа технология спуфинга требует пакетов сетевого уровня и выше. Поэтому для мостов реализация спуфинга — не такое обычное дело, как для маршрутизаторов. Мосты, поддерживающие спуфинг, не строят таблицы маршрутизации и не продвигают пакеты на основе сетевых адресов, но разбор заголовков и содержимого пакетов верхних уровней делают. Такие интеллектуальные удаленные мосты выпускает, например, компания Gandalf, хотя недорогие маршрутизаторы постепенно вытесняют мосты и в этой области. Сегментация пакетов — позволяет разделять большие передаваемые пакеты и передавать их сразу через две телефонные линии. Хотя это и не делает телефонные каналы более эффективными, но все же увеличивает скорость обмена данными почти вдвое.

Выводы

- Глобальные сети IP можно разделить на два класса: «чистые» сети IP и сети «IP поверх ATM/FR».
- В «чистых» сетях IP под слоем протокола IP нет еще одного слоя, работающего по технологии коммутации пакетов. Обеспечение качества обслуживания в таких сетях затрудняется размытостью путей трафика и нерациональностью загрузки маршрутизаторов из-за выбора маршрута с наименьшей метрикой.

Сети IP поверх сетей ATM/PR

При построении сети IP поверх сети ATM/FR между слоем каналов и слоем IP работает сеть ATM или frame relay. Так как скорости, на которых работает сеть frame relay, как правило, не превышают 2 Мбит/с, а уровень задержек и их вариаций входит в число параметров QoS, поддерживаемых данной технологией, то чаще всего в качестве промежуточного слоя магистрали применяется технология ATM.

Взаимодействие слоя IP со слоем ATM иллюстрирует рис. 17.22.

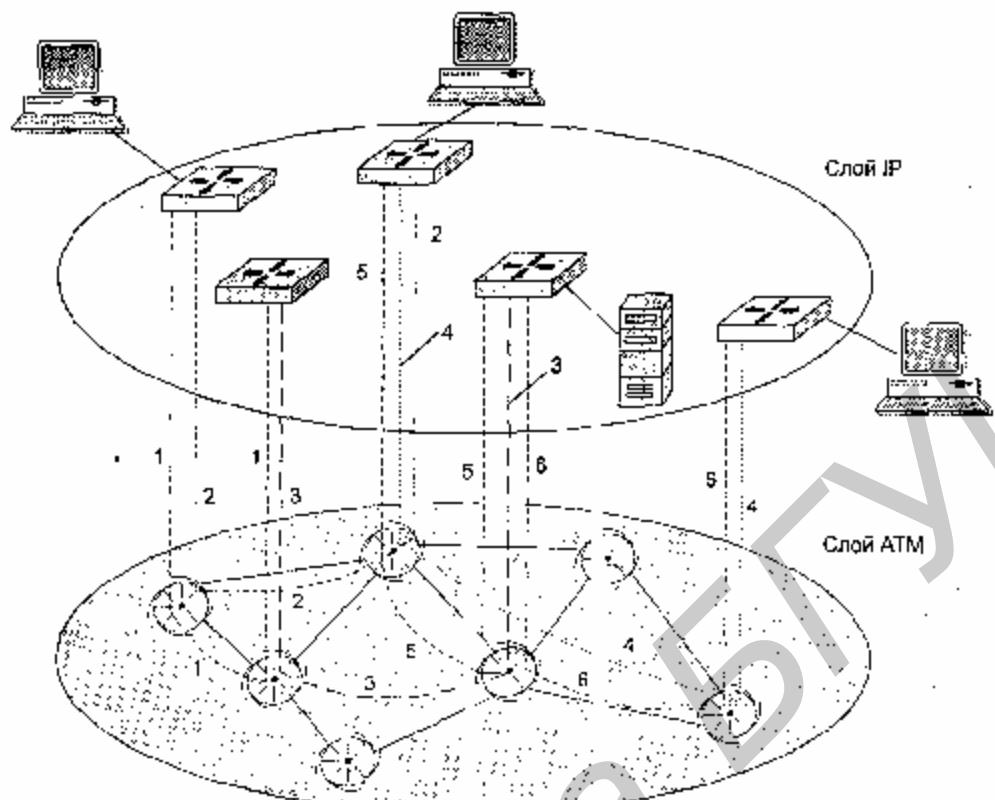


Рис. 17.22. Взаимодействие слоев IP и ATM

В сети ATM проложено шесть постоянных виртуальных каналов, соединяющих порты маршрутизаторов. Каждый порт маршрутизатора должен поддерживать технологию ATM в качестве конечного узла. После того как виртуальные каналы установлены, маршрутизаторы могут пользоваться ими как физическими, посылая данные порту соседнего (по отношению к виртуальному каналу) маршрутизатора.

Виртуальные каналы образуют в сети ATM сеть каналов со своей собственной топологией. Топология каналов, соответствующая сети, представленной на рис. 17.22, показана на рис. 17.23. Сеть ATM прозрачна для маршрутизаторов IP, они ничего не знают о физических связях между портами коммутаторов ATM. Сеть IP является наложенной (overlay) по отношению к сети ATM.

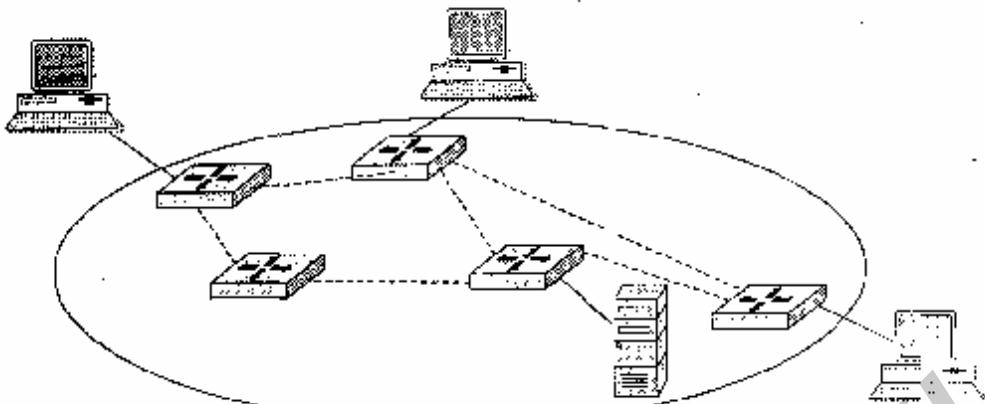


Рис. 17.23. Топология связей между маршрутизаторами.

Для того чтобы протокол IP мог корректно работать, ему необходимо знать соответствие между IP-адресами соседей и адресами виртуальных каналов ЛТМ, с помощью которых достичим соответствующий IP-адрес, то есть знать таблицу ARP-протокола. Такая таблица в данной ситуации создается не автоматически путем широковещательных запросов протокола ARP, а вручную. Администратор сети IP должен выполнить конфигурирование каждого интерфейса маршрутизатора, задав таблицу соответствия для всех номеров виртуальных каналов, исходящих и находящихся в этот интерфейс. При этом физический интерфейс представляется в виде набора логических интерфейсов (или подинтерфейсов — subinterface), имеющих IP-адреса.

Например, на маршрутизаторах компании Cisco Systems конфигурирование логического интерфейса, соответствующего виртуальному каналу с адресом VPI/VC1, равным 0/36, выглядит следующим образом:

```
pvc 0/36
protocol tp 10.2.1.1
```

После выполнения этой команды маршрутизатор будет знать, что в случае необходимости пересылки пакета по адресу NcxtHop, равному 10.2.1.1, ему нужно будет разбить пакет на последовательность ячеек ATM (с помощью функции SAR интерфейса ATM) и отправить их все по постоянному виртуальному каналу с адресом 0/36.

Если многослойная сеть ATM/IP должна передавать трафик различных классов с соблюдением параметров QoS для каждого класса, то соседние маршрутизаторы должны быть связаны несколькими виртуальными каналами, по одному для каждого класса. У маршрутизатора должна быть задана политика классификации пакетов, позволяющая отнести передаваемый пакет к определенному классу. Пакеты определенного класса направляются на определенный виртуальный канал, который обеспечивает трафику требуемые параметры QoS. Для того чтобы эти параметры соблюдались, необходимо предварительно провести инжиниринг трафика для сети ATM, выбрав маршруты следования каждого виртуального канала так, чтобы соблюдались средние скорости потоков, которые будут протекать по виртуальным каналам, а коэффициент загрузки каждого

интерфейса ATM коммутаторов не превышал определенной пороговой величины, гарантирующей каждому классу трафика приемлемый уровень задержек. Наложенная сеть IP может также использовать режим коммутируемых виртуальных каналов (SVC) для передачи трафика ТР. Этот режим подходит для неустойчивых потоков, которые существуют в течение небольших периодов времени. Создавать для таких потоков инфраструктуру постоянных каналов невыгодно, так как большую часть времени они будут простаивать. Для того чтобы маршрутизаторы могли использовать режим SVC, необходимо задать отображение IP-адресов, но не на номера виртуальных каналов, а на ATM-адреса конечных точек сети ATM, то есть ATM-адреса интерфейсов маршрутизатора. Один из вариантов задания такого отображения для маршрутизаторов Cisco имеет следующий вид:

Map-list a

1p 10.1.0.3 atm-nsap 33.3333.33.333333.3333.3333.3333.3333.3333.3333.3333.3333.33

После задания такого отображения маршрутизатор для отправки пакета по адресу NcxtHop, равному 10.1.0.3, предварительно устанавливает с помощью протокола Q.2931 коммутируемый виртуальный канал SVC с адресом 33.3333.33.333333.3333.3333.3333.3333.3333.3333.3333.3333.33, а затем, автоматически получим от этого протокола адрес VPI/VCI, отправляет по нему ячейки, на которые разбит исходный пакет. Интерфейс соседнего маршрутизатора, получив все ячейки, объединяет их в исходный пакет и передает наверх протоколу ТР.

Если по коммутируемому виртуальному каналу нужно передавать трафик с некоторыми требуемыми параметрами QoS, то эти параметры передаются протоколу Q.2931, который выбирает маршрут для канала с их учетом.

Структура «IP поверх ATM» на сегодня является основной у всех операторов связи, которые предоставляют услуги с заключением соглашения об уровне обслуживания (Service Level Agreement, SLA).

В качестве альтернативы сети «IP поверх ATM» сегодня рассматривается вариант 1P/MPLS. Технология многопротокольной коммутации с помощью меток (Multiprotocol Label Switching. MPLS) является сегодня одной из самых перспективных транспортных технологий. Рамки данной книги не позволяют дать ее подробное описание, поэтому скажем только, что эта технология объединяет возможности продвижения пакетов по каналам коммутации меток, аналогичным по своим функциям виртуальным каналам, с топологическими возможностями выбора маршрутов протокол он маршрутизации IP — таких как OSPF или IS-IS. При этом трафик перемещается по сети по устойчивым маршрутам, так что создаются предпосылки для решения задачи инжиниринга трафика и обеспечения для различных классов дифференцированного качества обслуживания. Можно со значительной долей приближения считать, что под слоем протокола IP в сети IP/MPLS работает некоторая канальная технология, основанная на технике виртуальных путей, но это не самостоятельная (и очень сложная) технология, подобная ATM, а тесно интегрированная с IP технология, пользующаяся возможностями IP по классификации трафика и выяснению топологии сети совместно с текущим состоянием загруженности ресурсов сети.

Еще одним преимуществом сети IP/MPLS по сравнению с вариантом «IP поверх ATM» является наличие устройств в сети только одного типа, так как функции MPLS встраиваются в IP-маршрутизаторы, так что физически два слоя коммуникационных устройств сливаются в один.

Иногда считают, что сеть IP/MPLS представляет собой «чистую» сеть IP. Это не совсем так, поскольку под протоколом IP работает в данном случае еще один протокол — протокол коммутации меток. Тем не менее такая точка зрения имеет право на существование, так как физически сеть действительно выглядит как «чистая» сеть IP, состоящая только из одних IP-маршрутизаторов.

Выводы

- В оверлейных (наложенных) сетях «IP поверх ATM/PR» маршрутизаторы объединяются не физическими, а виртуальными каналами ATM/FR. Это дает возможность обеспечить рациональную загрузку сети за счет инжиниринга трафика, а также воспользоваться стройной системой служб ATM, обеспечивающих каждому классу трафика свой уровень QoS.
- Перспективными являются сети IP/MPLS, которые приближаются по простоте своей организации к «чистым» сетям IP, а по рациональности загрузки и возможностям поддержки QoS — к сетям «IP поверх ATM/FR».

ГЛАВА 18 Организация доступа

Если связи между локальными сетями всегда строятся соединением локальных сетей с помощью транспортных функций некоторой глобальной пакетной сети, то для организации удаленного доступа могут использоваться различные схемы и продукты. Продукты удаленного доступа могут существенно отличаться реализованными в них функциями, а значит, и возможностями решения конкретной практической задачи.

Схемы доступа

Удаленный доступ — очень широкое понятие, которое включает в себя различные типы и варианты взаимодействия компьютеров, сетей и приложений. Если рассматривать все многочисленные схемы взаимодействия, которые обычно относят к удаленному доступу, то всем им при взаимодействии присуще использование глобальных каналов, или глобальных сетей. Кроме того, для удаленного доступа, как правило, характерна несимметричность взаимодействия, когда, с одной стороны, имеется центральная крупная сеть или центральный компьютер, а с другой — отдельный удаленный терминал, компьютер или небольшая сеть, которые хотят получить доступ к информационным ресурсам центральной сети. Количество удаленных от центральной сети узлов и сетей, требующих этот доступ, постоянно растет, поэтому современные средства удаленного доступа рассчитаны на поддержку большого количества удаленных клиентов.

Типы взаимодействующих систем

На рис. 18.1 приведены основные схемы удаленного доступа, отличающиеся типом взаимодействующих систем:

- терминал-компьютер — (1);
- компьютер-компьютер — (2);
- компьютер-сеть — (3);
- сеть-сеть — (4).

Первые три вида удаленного доступа часто объединяют понятием индивидуального доступа, а схемы доступа сеть-сеть иногда делят на два класса — НОВО и SOHO. Класс ROBO (Regional Office/Branch Office) соответствует случаю подключения к центральной сети сетей средних размеров — сетей региональных подразделений на предприятия, а классу SOHO (Small Office/Home Office) соответствует случай удаленного доступа сетей небольших офисов и домашних сетей.

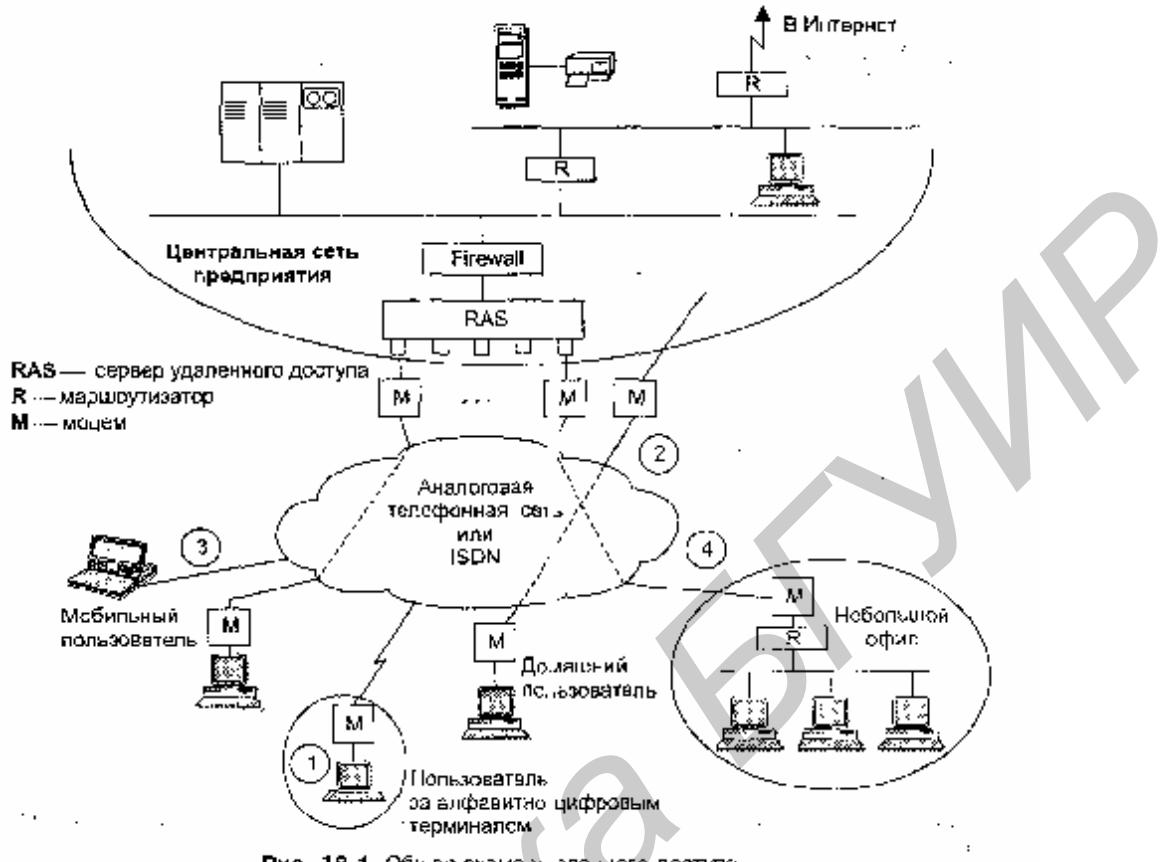


Рис. 18.1. Общая схема удаленного доступа.

Типы поддерживаемых служб

Схемы удаленного доступа могут отличаться также и типом служб, которые поддерживаются для удаленного клиента. Наиболее часто используется удаленный доступ к файлам, базам данных, принтерам в том же стиле, к которому пользователь привык при работе в локальной сети. Такой режим называется режимом удаленного узла (remote node). Иногда при удаленном доступе реализуется обмен с центральной сетью сообщениями электронной почты, с помощью которых можно в автоматическом режиме получить запрашиваемые корпоративные данные, например из базы данных.

Особое место среди всех видов удаленного доступа к компьютеру занимает способ, при котором пользователь получает возможность удаленно работать с компьютером таким же способом, как если бы он управлял им с помощью локально подключенного терминала. В этом режиме он может запускать на выполнение программы на удаленном компьютере и видеть результаты их выполнения. При этом принято подразделять такой способ доступа на терминальный доступ и удаленное управление,. Если у удаленного пользователя в распоряжении имеется только неинтеллектуальный алфавитно-цифровой терминал (вариант 1 на рис. 18.1) или же он запускает на своем персональном компьютере программу эмуляции такого терминала (например, программу Hyper Terminal

из утилит Windows), то этот режим работы называют терминальным доступом. Для владельца алфавитно-цифрового терминала, например УГ-ЮО, подобный вид удаленного доступа является единственным возможным. Доступ к мейнфрейму IBM, работающему под управлением операционной системы MVS, с помощью доступа через удаленное или встроенное устройство PAD, которое затем работает с мейнфреймом через сеть X.25. также является примером терминального доступа. Отличительной особенностью терминального доступа является то, что операционные системы на компьютере, к которому получают доступ пользователи, рассчитаны на многотерминальный режим работы, поэтому главное здесь — отличная от стандартного варианта схема подключения терминала, ориентированная на глобальные сети.

При удаленном управлении пользователь запускает на своем компьютере программу, эмулирующую на экране сеанс работы с операционной системой (DOS, Windows, OS/2), которая не поддерживает многотерминальный режим работы. Программа эмуляции экрана через глобальные каналы взаимодействует с дополнительным программным обеспечением, работающим под управлением соответствующей операционной системы на удаленном компьютере. Пользователь, как и при терминальном доступе, также получает полное управление над удаленным компьютером, при этом он видит на экране графический интерфейс привычной ему операционной системы, в качестве которой чаще всего выступает Windows. Результат получается практически тот же, но за счет нестандартного дополнительного программного обеспечения на удаленном компьютере.

Типы используемых глобальных служб

Схема организации удаленного доступа во многом определяется теми глобальными транспортными службами, которые доступны в точках нахождения многочисленных клиентов удаленного доступа. Кроме степени распространенности необходимо учитывать и стоимость глобальной службы. С учетом этих двух обстоятельств наиболее часто для организации удаленного доступа используются услуги телефонных сетей общего пользования - аналоговых п, если это возможно, ISDN. Только эти сети пока могут обеспечить дешевый доступ практически из любого географического пункта. Правда, услуги ISDN доступны только в городах и то фрагментарно. Обычно для подключения офиса по ISDN нужно предварительно заключить договор на такую услугу с оператором связи, после чего в течение нескольких недель к нам подводится абонентское окончание TSDN с соответствующей аппаратурой передачи данных NT1.

Услуги выделенных каналов экономически оправданы только при подключении небольшого числа крупных подразделений предприятия, а для отдельных пользователей ее применение не всегда экономически оправдано. Правда в последнее время ситуация в этой области начинает улучшаться, так, технологии xDSL, организующие выделенный канал до ближайшей АТС, обладают доступной стоимостью даже для индивидуального пользователя. Выделенный же

канал к узлу поставщика услуг, который удален от помещения пользователя на значительное расстояние, превышающее возможности телефонного абонентского окончания, то есть свыше 3-6 км, остаются доступными только корпоративным пользователям.

Выделенный канал соединяет пользователя с сетью с коммутацией пакетов. Это может быть конечная сеть, к ресурсам которой пользователю нужно получить доступ, например сеть его предприятия или Интернет. Это также может быть промежуточная сеть, а уже к промежуточной сети и некоторой точке присоединяется сеть, к ресурсам которой пользователь получает доступ. Чаще всего в качестве такой промежуточной сети выступает сегодня Интернет, но это может быть и частная сеть (например, frame relay, ATM или X.25) какого-либо оператора сотовой связи.

Экономические аспекты удаленного доступа должны учитывать способ его оплаты и интенсивность использования, которое обычно оценивается количеством часов загруженности глобальных каналов в месяц. Необходимо иметь в виду, что практически все транспортные службы удаленного доступа, связанные с коммутируемыми каналами, оплачиваются повременно, а в транспортных службах постоянных каналов схема оплаты помесячная, не зависящая от загрузки канала. Например, доступ через аналоговую телефонную сеть или ISDN оплачивается повременно, а доступ через выделенный канал 64 кбит/с или постоянный канал 64 кбит/с в сети frame relay оплачивается фиксированной месячной суммой.

Поэтому обычно сначала определяется, какое количество часов в месяц будет тот или иной удаленный пользователь работать с центральной локальной сетью удаленно. Затем на основании тарифов оплаты телекоммуникационных услуг находится тот вид услуги, который более экономичен для данного количества часов месячной работы. Обычно при количестве часов до 20-40 более выгодными являются аналоговые телефонные сети и ISDN. Для пользователей, которым требуется большее чем 40 количество часов доступа в месяц, например 60-80, может оказаться более выгодным выделенный канал frame relay. Необходимо отметить, что коммутируемые услуги в сетях, основанных на технике виртуальных каналов, обычно оплачиваются также по временной схеме. Так оплачиваются услуги коммутируемых виртуальных каналов ATM и только появляющаяся аналогичная служба сетей frame relay.

Выводы

- Удаленный доступ характеризуется использованием глобальных транспортных служб, несимметричностью взаимодействия и большим количеством удаленных пользователей.
- При удаленном доступе в основном используются аналоговые телефонные сети и ISDN — ввиду их распространенности и невысокого уровня оплаты при соединениях небольшой длительности.
- Удаленные пользователи подключаются к специальному устройству цен-

тральной сети — серверу удаленного доступа (RAS), который работает в режиме маршрутизатора или шлюза, в зависимости от протоколов, применяемых удаленным пользователем.

Доступ к компьютер-сеть

В связи с широким распространением на предприятиях локальных сетей наиболее часто встречающийся вид удаленного доступа — это доступ не к отдельному компьютеру, а к сети в целом. Для этой цели в центральной сети предприятия устанавливается специальная система — сервер удаленного доступа (Remote Access Server, RAS), который выполняет большой спектр функций по обслуживанию многочисленных удаленных клиентов. Задачи сервера удаленного доступа, который часто называют также коммуникационным сервером, зависят от схемы удаленного доступа.

Очевидно, что для экономии модемов можно не ставить на каждый компьютер центральной сети отдельный модем, а организовать общий пул модемов и сделать его разделяемым ресурсом как для звонков из локальной сети, так и для звонков извне. Действительно, если каждому пользователю выделить персональный модем (и персональную линию связи), то, как правило, большую часть времени он будет простоять, поэтому гораздо эффективнее ограничиться тем числом модемов (и линий), которое реально необходимо.

Разделяемый для пользователей локальный пул модемов создается с помощью так называемого коммуникационного сервера (Communication Server). Коммуникационный сервер — это обычный компьютер или специализированное устройство, предоставляющее пользователям локальной сети прозрачный доступ к последовательным портам ввода-вывода, к которым подключены разделяемые модемы. Пользователь, подключившийся по локальной сети к коммуникационному серверу, может работать с одним из подключенных к нему модемов точно так же, как если бы этот модем был подключен непосредственно к компьютеру пользователя. Таким образом, коммуникационный сервер обслуживает пользователей локальной сети, делая локальные модемы разделяемыми ресурсами. Говорят, что коммуникационный сервер поддерживает режим dial-out, который позволяет пользователям локальной сети устанавливать по своей инициативе связь через телефонную сеть с каким-либо удаленным компьютером.

Сервер удаленного доступа обслуживает не локальных, а удаленных пользователей, предоставляя им доступ к ресурсам локальной сети - файлам, принтерам и т. п. — извне. Сервер удаленного доступа поддерживает режим dial-in, который позволяет пользователю, работающему на удаленном компьютере, устанавливать связь с локальной сетью по его инициативе. Именно это является основной задачей систем удаленного доступа. С этой точки зрения удаленный доступ можно определить как эффективный способ разделения ресурсов централизованных серверов между удаленными клиентами.

Часто коммуникационный сервер и сервер удаленного доступа являются одним и тем же продуктом, выполненным либо в качестве дополнительного про-

граммного обеспечения в среде какой-либо популярной ОС, либо в качестве отдельного устройства. Название «сервер удаленного доступа» обычно закрепляется именно за таким комбинированным продуктом. Примерами программных серверов удаленного доступа являются сервер Microsoft RAS, работающий в составе ОС Windows, и сервер NetWare Connect, работающий в среде ОС NetWare.

Однако если режим dial-in поддерживают все серверы удаленного доступа по определению, то режим dial-out является факультативным и реализуется не всегда.

Режимы dial-in и dial-out отличаются только тем, кто является инициатором установления соединения — удаленный пользователь или пользователь локальной сети.

В зависимости от потребностей пользователей и возможностей программно-аппаратного обеспечения удаленный доступ может осуществляться в соответствии с различными схемами; удаленный узел, удаленное управление и взаимодействие с помощью электронной почты.

Удаленный узел

Одним из вариантов удаленного доступа типа компьютер-сеть является режим удаленного узла (remote node). Программное обеспечение удаленного узла на клиентской машине позволяет последовательному порту и модему (или терминальному адаптеру ISDN) стать медленным узлом удаленной локальной сети, взаимодействующим обычным способом с сетевыми операционными системами при разделении их ресурсов. В локальной сети должен быть установлен сервер удаленного доступа, поддерживающий режим удаленного узла. Это означает, что сервер должен поддерживать один из протоколов канального уровня, используемых в глобальном канале. Протокол канального уровня необходим для связи удаленного компьютера с центральной локальной сетью. Так как чаще всего этот канал является коммутируемым каналом телефонной сети или ISDN, то сервер удаленного доступа должен поддерживать протоколы PPP и SLIP, используемые на этих каналах. В сети X.25 или frame relay сервер удаленного доступа должен поддерживать протокол этих сетей, то есть протоколы LAP-B и X.25/3 для первого случая и LAP-F для второго (если сеть frame relay поддерживает только постоянные виртуальные каналы). При получении по глобальному каналу кадров соответствующего протокола, сервер, работающий в режиме удаленного узла, извлекает из кадра, например PPP, пакеты тех общих протоколов сетевого уровня, по которым работают удаленный компьютер и компьютеры локальной сети. Такими протоколами могут быть протоколы IP, IPX или немаршрутизуемый протокол NetBEUI. Далее вступают в работу протоколы верхних уровней, и пользователь получает такой же доступ, как если бы его компьютер находился непосредственно в локальной сети, но с небольшим исключением — скорость обмена его компьютера с остальными компьютерами удаленной сети зависит от пропускной способности глобального канала связи.

Клиенты, работающие в режиме удаленного узла, могут логически войти в сеть таким же образом, как если бы они были локальными пользователями, отображать сетевые диски и даже загружать программы через удаленную связь. Но удаленная загрузка больших программ неразумна, так как самый скоростной модем 33,6 кбит/с работает со скоростью, составляющей только 3 % от скорости сегмента Ethernet, и программа, которая в локальной сети загружается за 30 с, будет загружаться по удаленной связи в течение 15-20 минут. Поэтому в режиме удаленного узла локальные копии программ, как правило, эффективнее.

Другая проблема связана со способом работы сетевых операционных систем. Серверы часто рассылают широковещательные сообщения всем узлам сети для проверки подключенных и работающих клиентов. Такие широковещательные рассылки могут заблокировать удаленный доступ, если они не фильтруются перед отправкой по удаленным связям. Поэтому перед приобретением любого продукта необходимо проверить по его описаниям, может ли он работать в режиме удаленного доступа.

Компьютер, использующий режим удаленного узла, наиболее эффективно работает с системами клиент-сервер, так как в этом случае трафик по глобальному каналу обычно передается не очень интенсивный — запрос клиента обрабатывается на сервере, а по глобальному каналу передается только ответ. В режиме кли-снт-сервер работают многие корпоративные СУБД (например, Oracle, Informix, Microsoft SQL Server), а также приложения, ориентированные на эту архитектуру. Многие административные утилиты современных операционных систем поддерживают такой режим, например User Manager for Domains в ОС Windows NT.

Серверы, работающие в режиме удаленного узла, выполняют свои функции различным образом.

Первый вариант — это реализация в сервере удаленного узла функционального эквивалента маршрутизатора с WAN-портами для асинхронных модемов, TSDN-линий или асинхронного доступа к PAD X.25. Этот вариант универсален, так как обеспечивает доступ как отдельных компьютеров, так и локальных сетей. Однако данный вариант при подключении отдельного компьютера избыточен, поскольку требует выделения ему специального номера сети (как и каждому подключившемуся к сети пользователю).

Второй вариант основан на работе сервера удаленного узла в режиме шлюза. Если удаленные клиенты и локальная сеть работают па протоколе IP, то всем удаленным компьютерам присваивается один и тот же номер сети IP, совпадающий с номером локальной сети, к которой они получают доступ. В этом случае сервер выполняет функции посредника по протоколу ARP (говорят, что он поддерживает режим proxy ARP), отвечая компьютерам локальной сети своим MAC-адресом на запросы о IP-адресах, принадлежащих удаленным подключившимся узлам. Для протокола NetBIOS работа сервера в режиме шлюза — это единственно возможный режим работы, так как этот протокол не может маршрутизоваться.

В сервере удаленного узла могут быть реализованы оба варианта работы, которые выбираются в зависимости от типа клиента (компьютер или сеть), а также

протокола.

Операционные системы Mac OS, OS/2, Windows 95 и Windows NT Workstation включают в стандартную поставку клиентскую часть программного обеспечения удаленного узла. В настоящее время имеется явная тенденция использования клиентами удаленного узла протокола PPP. В результате достигается совместимость клиентских и серверных частей систем различных производителей, работающих в режиме удаленного узла.

Удаленное управление и терминальный доступ

Другим распространенным вариантом удаленного доступа являются две разновидности практически одного и того же режима — удаленное управление (remote control) и терминальный доступ (terminal access). При этом способе удаленный компьютер становится, в сущности, виртуальным терминалом компьютера-хоста, который может быть, а может и не быть подключен к сети. Этот вариант позволяет запустить любое приложение па компьютере-хосте, а также получить доступ к любым данным этого хоста. Если компьютер-хост подключен к сети, то его удаленные пользователи также становятся полноправными членами сети, действуя как пользователи компьютера-хоста.

Выше уже было сказано, что отличия удаленного управления от терминального доступа только в том, что при удаленном управлении пользователь связывается с операционной системой, не рассчитанной на поддержку многотерминального режима (MS-DOS, Windows 3.1, Windows 95/98, Windows NT, OS/2 Warp), а терминальный доступ осуществляется к операционным системам, для которых многотерминальный режим является основным (UNIX, 1ГШ VM, 1MB OS-400, VAX VMS).

Удаленное управление или терминальный доступ нужны тогда, когда удаленный пользователь работает с приложениями, не оптимизированными для работы в сети, например с традиционными СУБД персональных компьютеров типа dBase, Paradox или Access. Иначе, когда такое приложение находится па одном компьютере, а файлы баз данных — на другом, в сети создается чрезмерно интенсивный трафик.

Централизованная схема удаленного управления требует установки в локальной сети предприятия специального программного продукта — сервера удаленного управления, например сервера Win Frame компании Citrix. На клиентских удаленных компьютерах также нужно установить дополнительное программное обеспечение — клиента удаленного управления.

Протоколы, используемые программами удаленного управления для передачи информации об обновлении экрана, нажатиях клавиш и перемещениях мыши, являются нестандартными — поэтому нужно устанавливать серверную и клиентские части удаленного управления от одного производителя. Например, пользователи программного клиента удаленного доступа Norton pcLpuwhere не смогут дозвониться до хоста, работающего под управлением программ Reach Out, Lap-Link for Windows, Carbon Copy, Remotely Possible или Close-Up.

При терминальном доступе также желательно установить в центральной сети специальный продукт — терминальный сервер. Можно обойтись и без него, но тогда на каждый компьютер, к которому нужно подключиться в режиме удаленного терминала, нужно ставить модем и выделять ему отдельный телефонный номер. Терминальный сервер принимает запросы на связь с определенным компьютером и передает по локальной сети коды нажатия клавиш и символы, подлежащие отображению на экране пользовательского терминала. Для взаимодействия по локальной сети с многотерминальными ОС терминальный сервер использует стандартные протоколы эмуляции терминала, например telnet для UNIX, DEC LAT для VAX VMS.

Почта

Почта является еще одним видом удаленного доступа на прикладном уровне. Почтовые шлюзы, доступные по коммутируемым телефонным линиям, и клиентское почтовое обеспечение удаленного доступа могут быть достаточными для удовлетворения потребностей многих обычных пользователей. Такие почтовые шлюзы позволяют удаленным пользователям или даже удаленным офисам звонить в почтовую систему центрального отделения, обмениваться входящими и исходящими сообщениями и файлами, а затем отключаться.

Продукты, предназначенные для этих целей, варьируются от клиентских программ для одного пользователя, таких как ee:mail Mobile фирмы Lotus, до полномасштабных шлюзон, которые организуют почтовый обмен между удаленными серверами и корпоративной локальной сетью (например, Exchange компании Microsoft).

Почтовые шлюзы могут быть полезны в случае, когда количество данных, которыми обмениваются удаленные пользователи с центральным офисом, не очень большое. Из-за того, что среднее время сеанса пользователь-шлюз сравнительно невелико, шлюз центральной сети ни должен поддерживать большое количество телефонных линий. Обычно почтовое соединение легко устанавливается, а стоимость программного обеспечения шлюза незначительна.

Шлюзы работают в автоматическом режиме без вмешательства человека. Если в удаленном офисе работает один или два сотрудника и им не нужен доступ к корпоративным данным о реальном масштабе времени, то почтовый шлюз может быть хорошим решением. Некоторые приложения автоматически принимают запросы в виде писем электронной почты, а затем посыпают в таком же виде ответы. Так, например, работают многие СУБД.

Не только почта, но и другие приложения, написанные для локальной вычислительной сети, могут иметь специфические программные модули, предназначенные для удаленных соединений. Такие программы устанавливают соединения между собой с помощью нестандартных протоколов и часто увеличивают эффективность соединения за счет специальных приемов, например путем передачи только обновлений между удаленным компьютером и хостом. Примером продуктов этого класса являются программные системы коллективной ра-

боты.

Выводы

- Наиболее универсальным режимом удаленного доступа является режим удаленного узла, при котором компьютер пользователя является узлом локальной сети предприятия со всеми его возможностями, но только подключенным к сети через низкоскоростной канал по протоколу PPP.
- Связь с центральной локальной сетью по инициативе удаленного пользователя называется режимом dial-in (основной режим), а по инициативе пользователя центральной сети — dial-out.
- Режимы терминального доступа и удаленного управления позволяют удаленному пользователю подключиться к компьютеру нейтральной сети в режиме, имитирующем работу локального терминала. Этот режим очень экономно расходует полосу пропускания глобального канала и рекомендуется для тех случаев, когда необходим линзкоскоростной канал — 4800 или 9600 бит/с.
- Для удаленного доступа может использоваться режим электронной почты, который автоматически поддерживается многими приложениями, в том числе СУБД, для получения запросов и отправки ответов.

Удаленный доступ через промежуточную сеть

Общая схема двухступенчатого доступа

Раньше удаленный международный или междугородный доступ отдельных пользователей всегда реализовывался по схеме, основанной на международной или междугородной телефонной связи. Публичные территориальные сети с коммутацией пакетов (в основном — сети X.25) не были так распространены, чтобы, находясь в любом городе, посланный в командировку сотрудник мог получить доступ к этой сети, а через нее — к маршрутизатору или серверу удаленного доступа своего предприятия.

Поэтому удаленные пользователи непосредственно звонили по телефонной сети на сервер удаленного доступа своего предприятия, не считаясь, с затратами на международные или междугородные переговоры.

Однако сегодня очень часто служба международной сети с коммутацией пакетов имеется во многих городах, и чаще всего это служба Интернета. По мере развития услуг сетей frame relay возможно, что и эта технология получит такое же массовое распространение. Поэтому стала возможной двухступенчатая связь удаленного пользователя со своей корпоративной сетью — сначала выполняется доступ по городской телефонной сети к местному поставщику услуг Интернета, а затем через Интернет пользователь соединяется со своей корпоративной сетью.

Такой вариант может значительно удешевить доступ по сравнению с непосред-

ственным подключением через междугородные АТС.

Обычно экономия происходит за счет перехода от междугородных (или международных) звонков к местным. Если поставщик услуг сети с коммутацией пакетов поддерживает доступ по коммутируемым телефонным сетям, то непосредственный доступ к серверу, установленному в центральной сети, находящейся в другом городе, заменяется звонком на сервер удаленного доступа местного поставщика услуг (рис. 18.2).

Центральная сеть предприятия, используя выделенный канал, обычно непосредственно подключается к той же сети с коммутацией пакетов, что и удаленные пользователи и других городах.

Стандартизация клиентов удаленного доступа на основе протоколов PPP и SLIP упрощает проблемы обслуживания разнородных пользователей одним поставщиком услуг при использовании Интернета в качестве промежуточной сети. Для сетей X.25 протоколы взаимодействия сети офиса с сетью поставщика услуг также вполне определены, хотя иногда наблюдаются случаи различной настройки одного и того же протокола в оборудовании и программном обеспечении клиента и поставщика услуг.

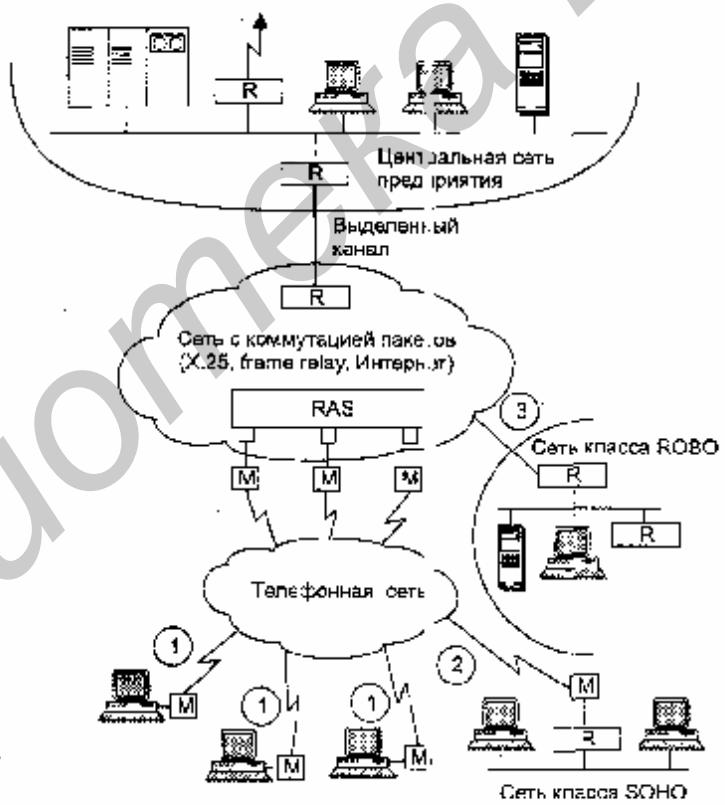


Рис. 18.2. Подключение удаленных пользователей через промежуточную публичную сеть с коммутацией пакетов

Выгода от Интернета в качестве промежуточного транспорта оказывается особенно ощутимой, так как расценки-поставщиков услуг Интернета намного ниже, чем расценки поставщиков услуг сетей X.25. Это обстоятельство является не последней причиной бурного распространения технологии inlranct, использу-

зующей транспортные и информационные службы Интернета для внутрикорпоративных нужд.

Ввиду большой популярности Интернета в качестве инструмента доступа к корпоративной сети для этой двухступенчатой схемы разработано много протоколов и средств, которые создают виртуальный туннель между пользователем и точкой входа в корпоративную сеть — маршрутизатором или сервером удаленного доступа. Этот туннель решает две задачи. Во-первых, передачу через сеть IP, которой является Интернет, чужеродного для нас трафика — протоколов IPX, NetBEUI, непосредственно Ethernet и т. п. Во-вторых, туннель создает защищенный канал, данные в котором шифруются.

Промежуточная телефонная сеть делает доступ через Интернет к корпоративной сети весьма медленным. В последнее время появилось несколько решений, позволяющих пользователю получить достаточно быстрый доступ к Интернету через существующие инфраструктуры абонентских окончаний телефонных сетей и сетей кабельного телевидения.

Технологии xDSL

Сегодня многие поставщики услуг и операторы связи начали активно внедрять различные варианты цифровых абонентских линий (DSL). Наибольшее внимание специалистов привлекла технология асимметричной цифровой абонентской линии (Asymmetric Digital Subscriber Line, ADSL), но помимо нее пользователям предложены также службы симметричной цифровой абонентской линии (SDSL), цифровой абонентской линии с переменной скоростью (Rate Adaptive DSL, RADSL) и сверхбыстрой цифровой абонентской линии (Very high-speed DSL, VDSL).

Цифровые абонентские окончания появились достаточно давно — впервые их ввели первичные сети каналов T1/E1. Цифровое абонентское окончание High-speed DSL (HDSL) работает по 4-проводной линии со скоростью до 1,5-14 или 2,048 Мбит/с. Цифровое абонентское окончание сети ISDN работает по 2-проводному окончанию со скоростью 128 кбит/с.

Однако сегодня пользователям хотелось бы получать доступ к Интернету (и через Интернет к своим корпоративным сетям) с помощью стандартного 2-проводного телефонного окончания, установив при этом на своем домашнем компьютере какое-нибудь устройство типа модема. Перечисленные выше технологии позволяют это сделать с помощью специальных модемов.

Эти технологии рассчитаны на высокоскоростную передачу данных на коротком отрезке витой пары, соединяющей абонента с ближайшей телефонной АТС, то есть на решение проблемы «последней мили», отделяющей потребителя от поставщика услуг. В то время как обычные модемы (V.34, V.90) рассчитаны на работу с полосой пропускания в 3100 Гц через сеть с произвольным количеством коммутаторов, модемы xDSL могут получить в свое распоряжение полосу порядка 1 МГц — эта величина зависит от длины кабеля до АТС и сечения используемых проводов. Отличие условий работы модемов xDSL от обычных модемов показаны на рис. 18.3 на примере ADSL-модемов.

ADSL-модемы, подключаемые к обоим концам короткой линии между абонентом и АТС, образуют три канала: быстрый канал передачи данных из сети в компьютер, менее быстрый дуплексный канал передачи данных из компьютера в сеть и простой канал телефонной связи, по которому передаются обычные телефонные разговоры. Передача данных в канале «сеть-абонент» происходит со скоростью от 1,5 до 6 Мбит/с, в канале «абонент-сеть» — со скоростью от 16 кбит/с до 1 Мбит/с.

В обоих случаях конкретная величина скорости передачи зависит от длины и качества линии. Асимметричный характер скорости передачи данных вводится специально, так как удаленный пользователь Интернета или корпоративной сети обычно загружает данные из сети в свой компьютер, а в обратном направлении идут либо квитанции, либо поток данных существенно меньшей скорости. Для получения асимметрии скорости полоса пропускания абонентского окончания делится между каналами также асимметрично.

На дальнем конце абонентского окончания должен располагаться так называемый мультиплексор доступа ADSL — DSLAM. Этот мультиплексор выделяет подканалы из общего канала и отправляет голосовой подканал в 3100 Гц на АТС, а высокоскоростные каналы данных направляет на маршрутизатор, который должен находиться рядом с ADSLAM.

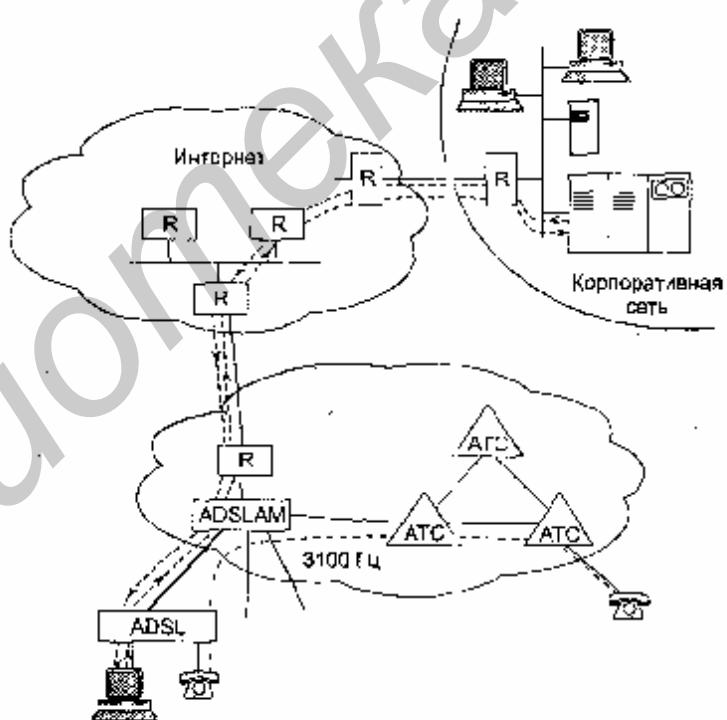


Рис. 18.3. Отличия условий работы ADSL-модемов от обычных модемов

Одно из главных преимуществ технологии ADSL по сравнению с аналоговыми модемами и протоколами ISDN и HDSL — то, что поддержка голоса никак не отражается на параллельной передаче данных по двум быстрым каналам. Причина подобного эффекта состоит в том, что технология ADSL основана на принципах разделения частот, благодаря чему голосовой канал надежно отде-

ляется от двух других каналов передачи данных. Такой метод передачи гарантирует надежную работу канала ТфОП даже при нарушении питания ADSL-модема. Никакие конкурирующие системы передачи данных не обеспечивают работу обычного телефонного канала столь же надежно. Хотя технологии ISDN и HDSL поддерживают режим обычной телефонной связи, для ее установления они требуют организации специального канала с пропускной способностью 64 кбит/с.

Маршрутизатор, расположенный в здании АТС, должен соединяться выделенным высокоскоростным каналом с другим маршрутизатором Интернета (или другой сети с коммутацией пакетов). Если центральная сеть предприятия подключена к Интернету через выделенный высокоскоростной канал, то все удаленные пользователи, у которых установлены модемы ADSL, получают высокоскоростной доступ к сети своего предприятия на тех же телефонных каналах, которые всегда соединяли их с городской АТС.

Широкое распространение технологий xDSL должно сопровождаться некоторой перестройкой работы поставщиков услуг Интернета и поставщиков услуг телефонных сетей, так как их оборудование должно теперь работать совместно. Возможен вариант, когда альтернативный оператор связи берет оптом в аренду большое количество абонентских окончаний у традиционного местного оператора.

Стандарт на ADSL-модемы уже принят. Эта технология поддерживает несколько вариантов кодирования информации ПМТ, САР и 2B1Q. Достижения технологий xDSL во многом определяются достижениями техники кодирования, которая за счет применения сигнальных процессоров DSP смогла повысить скорость передачи данных при одновременном увеличении расстояния между модемом и DSLAM.

Скорости каналов ADSL существенно зависят от качества физической линии, сечения ее проводников и расстояния между модемом и DSLAM. Чем больше это расстояние, тем ниже скорости. Обычно модем позволяет изменять скорость передачи данных, поэтому при установке его на определенное абонентское окончание можно подобрать оптимальный режим работы, обеспечивающий при удовлетворительном качестве передачи максимально возможную скорость.

Технология SDSL позволяет на одной паре абонентского окончания организовать два симметричных канала передачи данных. Канал тональной частоты в этом случае не предусматривается. Обычно скорости каналов в восходящем и нисходящем направлениях составляют по 2 Мбит/с, но, как и у технологии ADSL, эта скорость зависит от качества линии и расстояния до DSLAM.

Кроме абонентских окончаний телефонных сетей в последнее время для скоростного доступа к Интернету стали применять абонентские окончания кабельного телевидения. Для этих целей уже разработан специальный вид модемов — кабельные модемы. В кабельных модемах используется имеющийся коаксиальный 75-омный телевизионный кабель для передачи данных из сети в компьютер со скоростью до 30 Мбит/с, а из компьютера и сеть — со скоростью до 10 Мбит/с. При этом качество передаваемых сигналов очень высокое.

Высокоскоростные абонентские окончания создают для поставщиков услуг Интернета дополнительную проблему — им необходимо иметь очень скоростные каналы доступа к остальной части Интернета, так как 10 абонентов с трафиком по 8 Мбит/с создают общий трафик в 80 Мбит/с, который качественно можно передать только с помощью технологий SONET/SDH или ATM. Ведущие поставщики услуг Интернета, например UUCP, такие каналы уже имеют.

Выводы

- Для экономичного удаленного доступа в последнее время часто используется двухступенчатая схема доступа, в которой на первом этапе удаленный пользователь подключается через местную телефонную линию к местному поставщику услуг Интернета, а через Интернет выполняется второй этап подключения — к центральной сети, расположенной в другом городе или другой стране.
- Для скоростного доступа к Интернету через инфраструктуру абонентских окончаний телефонных аналоговых сетей или сетей кабельного телевидения разработаны новые технологии цифрового абонентского окончания — технологии xDSL, из которых наибольший интерес для индивидуальных клиентов представляет технология асимметричного доступа ADSL, а для корпоративных — SDSL.

Вопросы и упражнения к части V

1. Какие виды нарушений передачи пакетов в большей степени ухудшают качество голоса: переменные задержки или потери отдельных пакетов?
2. Что происходит с коэффициентом пульсации трафика при объединении большого числа потоков?
3. Маршрутизатор обслуживает трафик двух классов: приоритетного, имеющего среднюю интенсивность 500 кбит/с, и фонового, имеющего среднюю интенсивность 1000 кбит/с. Производительность маршрутизатора равна 2 Мбит/с. Каков коэффициент загрузки маршрутизатора для каждого класса трафика?
4. Какую функцию алгоритма «token bucket» не поддерживает алгоритм «leaky bucket»?
5. Что может произойти с трафиком best effort, если приоритетный трафик на входном интерфейсе не будет ограничен по средней интенсивности?
6. Какие параметры сети выбираются с помощью методов инжиниринга трафика?
7. Чем отличаются модемы от устройств DSU/CSU?
8. Предприятие решило создать собственную глобальную сеть. Какой тип глобальных связей будет наиболее эффективен, если предприятию необходимо соединить локальную сеть в штаб-квартире с тремя локальными сетями региональных подразделений, расположенных в разных городах? Средняя интенсивность трафика между сетями подразделений и центральной сетью оценивается

диапазоном значений от 500 кбит/с до 1 Мбит/с.

9. Вы убедились, что модем устойчиво работает на выделенном 2-проводном канале как в асинхронном, так и в синхронном режимах. Какой режим вы предпочтете?

10. К устройству какого уровня в терминах модели OSI можно отнести современный модем?

11. Можно ли использовать обычное абонентское окончание телефонной аналоговой сети, имеющееся в офисе, для подключения к каналу E1?

12. Каким видом услуг цифровых сетей можно воспользоваться, если необходимо соединить две локальные сети, находящиеся в разных городах, причем интенсивность межсетевого трафика составляет от 100 до 180 кбит/с?

13. Сколько каналов E1 можно передать в одном канале STM-1?

14. Может ли сеть X.25 работать без устройств PAD?

15. Какие устройства необходимо применить для подключения мэйнфрейма, имеющего только интерфейсы RS-232C, к локальной сети Ethernet, если известно, что сетевые адAPTERы Ethernet для этого мэйнфрейма не выпускаются?

16. Каким образом пользователь может подключиться к встроенному устройству PAD через телефонную сеть, если он работает за терминалом, который не поддерживает процедуры вызова абонента через телефонную сеть автоматически?

17. Какую услугу ISDN целесообразно использовать, если к этой сети подключены с помощью терминальных адAPTERов два персональных компьютера и им нужно постоянно обмениваться данными со скоростью 2400 бит/с с пульсациями до 9600 бит/с, причем величины задержек пакетов не являются критичными?

18. Какую услугу ISDN целесообразно использовать, если к этой сети подключены с помощью маршрутизаторов две локальные сети, причем межсетевой трафик имеет интенсивность от 100 до 512 кбит/с в течение длительного периода времени?

19. Сравните количество кадров, которое порождает обмен двумя сообщениями TCP (посылка данных и получение квитанции) между двумя конечными хостами, соединенными одним промежуточным коммутатором для случаев, когда этот коммутатор является коммутатором X.25 и когда этот коммутатор является коммутатором frame relay?

20. В каком случае процент дошедших кадров через сеть frame relay до конечного узла будет выше: когда услуга заказана на основании параметров CIR, Вс и Вe или когда услуга заказана на основании только параметров CIR и Вc (подразумевается, что значения параметров CIR и Вc в обоих случаях совпадают), при условии, что сеть frame relay недогружена, а узел-источник отправляет данные со скоростью, часто значительно превышающей CIR?

21. Если у вашего предприятия появилась необходимость соединить многочисленные сети филиалов с центральной сетью и между собой, но в распоряжении имеются только выделенные аналоговые каналы с установленными синхронными модемами 19,2 кбит/с, то какую технологию из следующих вы выберете; X.25, frame relay или ATM? Обоснуйте факторы, которые повлияют на ваше

решение.

22. Какую категорию услуг целесообразно выбрать для передачи голоса через сеть ATM?

23. Сколько виртуальных каналов должно быть установлено в каждом направлении между каждой парой коммутаторов ATM, если необходимо передавать три класса трафика с различным уровнем качества обслуживания?

24. Для какой из категорий услуг сеть ATM явно управляет потоком данных? Почему для других категорий услуг управление потоком данных не используется?

25. Вы хотите вручную настроить постоянный виртуальный канал в двух корпоративных сетях ATM, соединенных публичной сетью ATM. Вы не хотите, чтобы ваши номера VCI зависели от номеров виртуальных каналов, используемых администратором в публичной сети ATM. Какой вид коммутации вы заказаете у поставщика услуг публичной сети ATM?

26. Вы купили модем V.90 и связываетесь по телефонной сети со своим знакомым, который также использует модем V.90. Вы уверены, что все АТС на пути между вами и вашим знакомым работают в цифровом режиме. На какой скорости вы получите соединение со своим знакомым?

27. В каких случаях выгоднее использовать для удаленного доступа: сеть ISDN с интерфейсом B+D, выделенный цифровой канал 64 кбит/с, постоянный виртуальный канал frame relay с CTR = 64 кбит/с?

28. Вы соединили две локальные сети удаленным мостом, работающим через постоянный виртуальный канал в сети frame relay. Сеанс протокола NetBEUI между компьютерами разных сетей часто разрывается, в то же время в том случае, когда компьютеры принадлежат одной локальной сети, их взаимодействие протекает без проблем. В чем может быть причина такой ситуации?

29. В чем принципиальное отличие «чистой» сети ТР от сети IP, работающей поверх ATM?

30. Какие функции выполняет слой ATM в многослойной сети IP?

31. Что необходимо изменить, если модем xDSL работает на данном абонентском окончании с недопустимо высоким процентом ошибок?

Часть VI

Средства анализа и управления сетями

ГЛАВА 19 Функции и архитектура систем управления сетями

Любая сложная вычислительная сеть требует дополнительных специальных средств управления помимо тех, которые имеются в стандартных сетевых операционных системах. Это связано с большим количеством разнообразного коммуникационного оборудования, работа которого критична для выполнения сетью своих основных функций. Распределенный характер крупной корпоративной сети делает невозможным поддержание ее работы без централизованной системы управления, которая в автоматическом режиме собирает информацию о состоянии каждого концентратора, коммутатора, мультиплексора и маршрутизатора и предоставляет эту информацию оператору сети. Обычно система управления работает в автоматизированном режиме, выполняя наиболее простые действия по управлению сетью автоматически, а сложные решения предоставляя принимать человеку на основе подготовленной системой информации. Система управления должна быть интегрированной. Это означает, что функции управления разнородными устройствами должны служить общей цели обслуживания конечных пользователей сети с заданным качеством.

Сами системы управления представляют собой сложные программно-аппаратные комплексы, поэтому существует граница целесообразности применения системы управления — она зависит от сложности сети, разнообразия применяемого коммуникационного оборудования и степени его распределенности по территории. В небольшой сети можно применять отдельные программы управления наиболее сложными устройствами, например коммутатором, поддерживающим технику VLAN. Обычно каждое устройство, которое требует достаточно сложного конфигурирования, производитель сопровождает автономной программой конфигурирования и управления. Однако при росте сети может возникнуть проблема объединения разрозненных программ управления устройствами в единую систему управления, и для решения этой проблемы придется, возможно, отказаться от этих программ и заменить их интегрированной системой управления.

Системы управления корпоративными сетями существуют не очень давно. Одной из первых систем такого назначения, получившей широкое распространение, был программный продукт SunNet Manager, выпущенный в 1989 году компанией SunSoft. Система SunNet Manager была ориентирована на управление коммуникационным оборудованием и контроль трафика сети. Именно эти

функции имеют чаще всего в виду, когда говорят о системе управления сетью. Кроме систем управления сетями существуют и системы управления другими элементами корпоративной сети: системы управления ОС, базами данных (СУБД), корпоративными приложениями. Применяются также системы управления телекоммуникационными сетями: телефонными, а также первичными сетями технологий PDH и SDH.

Функциональные группы задач управления

Независимо от объекта управления, желательно, чтобы система управления выполняла ряд функций, которые определены международными стандартами, обобщающими опыт применения систем управления в различных областях. Существуют рекомендации ITU-T X.700 и близкий к ним стандарт ISO 7498-4, которые делят задачи системы управления на пять функциональных групп:

- управление конфигурацией сети и именованием;
- обработка ошибок;
- анализ производительности и надежности;
- управление безопасностью;
- учет работы сети.

Рассмотрим задачи этих функциональных областей управления применительно к системам управления сетями.

Управление конфигурацией сети и именованием

Задачи управления конфигурацией сети и именованием (Configuration Management) заключаются в конфигурировании параметров как элементов сети (Network Element, NE), так и сети в целом. Для элементов сети, таких как маршрутизаторы, мультиплексоры и т. п., с помощью этой группы задач определяются сетевые адреса, идентификаторы (имена), географическое положение и пр.

Для сети в целом управление конфигурацией обычно начинается с построения карты сети, то есть отображении реальных связей между элементами сети и изменении связей между элементами сети — образование новых физических или логических каналов, изменение таблиц коммутации и маршрутизации.

Управление конфигурацией (как и другие задачи системы управления) могут выполняться в автоматическом, ручном или полуавтоматическом режимах. Например, карта сети может составляться автоматически, на основании зондирования реальной сети пакетами-исследователями, а может быть построена оператором системы управления вручную. Чаще всего применяются полуавтоматические методы, когда автоматически полученную карту оператор подправляет вручную. Методы автоматического построения топологической карты, как правило, являются фирменными разработками.

Более сложной задачей является настройка коммутаторов и маршрутизаторов на поддержку маршрутов и виртуальных путей между пользователями сети. Согласованная ручная настройка таблиц маршрутизации при полном или час-

тичном отказе от использования протокола маршрутизации (а в некоторых глобальных сетях, например X.25, такого протокола просто не существует) представляет собой сложную задачу. Многие системы управления сетью общего назначения ее не выполняют, но существуют специализированные системы конкретных производителей, например система NetSys компании Cisco Systems, которые решают се для маршрутизаторов этой же компании.

Обработка ошибок

Группа задач обработки ошибок (Fault Management) включает выявление, определение и устранение последствий сбоев и отказов в работе сети. На этом уровне выполняется не только регистрация сообщений об ошибках, но и их фильтрация, маршрутизация и анализ на основе некоторой корреляционной модели. Фильтрация позволяет выделить из весьма интенсивного потока сообщений об ошибках, который обычно наблюдается в большой сети, только важные сообщения, маршрутизация обеспечивает их доставку нужному элементу системы управления, а корреляционный анализ позволяет найти причину, породившую поток взаимосвязанных сообщений (например, обрыв кабеля может быть причиной большого количества сообщений о недоступности сетей и серверов).

Устранение ошибок может быть как автоматическим, так и полуавтоматическим. В первом случае система непосредственно управляет оборудованием или программными комплексами и обходит отказавший элемент за счет резервных каналов и т. п. В полуавтоматическом режиме основные решения и действия по устранению неисправности выполняют люди, а система управления только помогает в организации этого процесса — оформляет квитанции на выполнение работ и отслеживает их поэтапное выполнение (подобно системам групповой работы).

В этой группе задач иногда выделяют подгруппу задач управления проблемами, подразумевая под проблемой сложную ситуацию, требующую для разрешения обязательного привлечения специалистов по обслуживанию сети.

Анализ производительности и надежности

Задачи анализа производительности и надежности (Performance Management) связаны с оценкой на основе накопленной статистической информации таких параметров, как время реакции системы, пропускная способность реального или виртуального канала связи между двумя конечными абонентами сети, интенсивность трафика в отдельных сегментах и каналах сети, вероятность иска~~жения~~ данных при их передаче через сеть, а также коэффициент готовности сети или ее определенной транспортной службы. Функции анализа производительности и надежности сети нужны как для оперативного управления сетью, так и для планирования развития сети.

Результаты анализа производительности и надежности позволяют контролировать соглашение об уровне обслуживания (Service Level Agreement, SLA), за-

ключаемое между пользователем сети и ее администраторами (или компанией, продающей услуги). Обычно в SLA оговариваются такие параметры надежности, как коэффициент готовности службы в течение года и месяца, максимальное время устранения отказа, а также параметры производительности, например средняя и максимальная пропускная способности при соединении двух точек подключения пользовательского оборудования, время реакции сети (если информационная служба, для которой определяется время реакции, поддерживается внутри сети), максимальная задержка пакетов при передаче через сеть (если сеть используется только как транзитный транспорт). Без средств анализа производительности и надежности поставщик услуг публичной сети или отдел информационных технологий предприятия не сможет ни проконтролировать, ни тем более обеспечить нужный уровень обслуживания для конечных пользователей сети.

Управление безопасностью

Задачи управления безопасностью (Security Management) подразумевают контроль доступа к ресурсам сети (данным и оборудованию) и сохранение целостности данных при их хранении и передаче через сеть. Базовыми элементами управления безопасностью являются процедуры аутентификации пользователей, назначение и проверка прав доступа к ресурсам сети, распределение и поддержка ключей шифрования, управления полномочиями и т. п. Часто функции этой группы не включаются в системы управления сетями, а реализуются либо в виде специальных продуктов (например, систем аутентификации и авторизации Kerberos, различных защитных экранов, систем шифрования данных), либо входят в состав операционных систем и системных приложений.

Учет работы сети

К задачам учета работы сети (Accounting Management) относится регистрация времени использования различных ресурсов сети — устройств, каналов и транспортных служб. Подобные задачи имеют дело с такими понятиями, как время использования службы и плата за ресурсы — billing. Ввиду специфического характера оплаты услуг у различных поставщиков и различными формами соглашения об уровне услуг эта группа функций обычно не включается в коммерческие системы и платформы управления типа HP OpenView, а реализуется в заказных системах, разрабатываемых для конкретного заказчика.

Системы управления системами

Модель управления OSI не делает различий между управляемыми объектами — каналами, сегментами локальных сетей, мостами, коммутаторами и маршрутизаторами, модемами и мультиплексорами, аппаратным и программным обеспечением компьютеров, СУБД. Все эти объекты управления входят в общее понятие

тие «система», и управляемая система взаимодействует с управляющей системой по открытым протоколам OSI.

Однако на практике деление систем управления по типам управляемых объектов широко распространено. Ставшими классическими системы управления сетями, такие как SunNet Manager, HP OpenView или Cabletron Spectrum, управляют только коммуникационными объектами корпоративных сетей, то есть концентраторами и коммутаторами локальных сетей, а также маршрутизаторами и удаленными мостами, как устройствами доступа к глобальным сетям. Оборудованием территориальных сетей обычно управляют системы производителей телекоммуникационного оборудования, такие как RADView компании RAD Data Communications, MainStreetXpress 46020 компании Newbridge и т. п. Рассмотрим, как преломляются общие функциональные задачи системы управления, определенные в стандартах X.700/ISO 7498-4, в задачи такого конкретного класса систем управления, как системы управления компьютерами и их системным и прикладным программным обеспечением. Их называют системами управления системой (System Management System, SMS).

Обычно система управления системой выполняет следующие функции.

- Учет используемых аппаратных и программных средств (Configuration Management). Система автоматически собирает информацию об установленных в сети компьютерах и создает записи в специальной базе данных об аппаратных и программных ресурсах. После этого администратор может быстро выяснить, какими ресурсами он располагает и где тот или иной ресурс находится, например, узнать о том, на каких компьютерах нужно обновить драйверы принтеров, какие компьютеры обладают достаточным объемом памяти, дискового пространства и т. п.
- Распределение и установка программного обеспечения (Configuration Management). После завершения обследования администратор может создать пакеты рассылки нового программного обеспечения, которое нужно установить на всех компьютерах сети или на какой-либо группе компьютеров. В большой сети, где проявляются преимущества системы управления, такой способ установки может существенно снизить трудоемкость этой процедуры. Система может также позволять централизованно устанавливать и администрировать приложения, которые запускаются с файловых серверов, а также дать возможность конечным пользователям запускать такие приложения с любой рабочей станции сети.
- Удаленный анализ производительности и возникающих проблем (Fault Management and Performance Management). Эта группа функций позволяет удаленно измерять наиболее важные параметры компьютера, операционной системы, СУБД и т. д. (например, коэффициент использования процессора, интенсивность страничных прерываний, коэффициент использования физической памяти, интенсивность выполнения транзакций). Для разрешения проблем эта группа функций может давать администратору возможность брать на себя удаленное управление компьютером в режиме эмуляции графического интерфейса популярных операционных систем. База данных системы управления обычно хранит детальную информацию о конфигурации всех компьютеров в сети для

того, чтобы можно было выполнять удаленный анализ возникающих проблем. Примерами систем управления системами являются Microsoft System Management Server (SMS), CA Unicenter, HP Operationscenter и многие другие.

Как видно из описания функций системы управления системами, они повторяют функции системы управления сетью, но только для других объектов. Действительно, функция учета используемых аппаратных и программных средств соответствует функции построения карты сети, функция распределения и установки программного обеспечения — функции управления конфигурацией коммутаторов и маршрутизаторов, а функция анализа производительности и возникающих проблем — функции производительности.

Эта близость функций систем управления сетями и систем управлением системами позволила разработчикам стандартов OSI не делать различия между ними и разрабатывать общие стандарты управления.

На практике уже несколько лет также заметна отчетливая тенденция интеграции систем управления сетями и системами в единые интегрированные продукты управления корпоративными сетями, например CA Unicenter TNG или TME-10 IBM/Tivoli. Наблюдается также интеграция систем управления телекоммуникационными сетями с системами управления корпоративными сетями.

Выводы

- Функции систем управления стандартизованы в рекомендации ITU-T X.700 и документе ISO 7498-4.
- Всего определено пять групп функций: управление конфигурацией и именованием (Configuration Management), обработка ошибок (Fault Management), анализ производительности и надежности (Performance Management), управление безопасностью (Security Management), учет работы сети (Accounting Management).
- Стандарты ITU X.700 и ISO 7498-4 определяют функции управления для телекоммуникационных систем широкого класса, при конкретизации системы (телефонная сеть, сеть передачи данных, операционная система) функции управления также конкретизируются.

Многоуровневое представление задач управления

Кроме описанного выше разделения задач управления на несколько функциональных групп, полезно разделять задачи управления на уровни в соответствии с иерархической организацией корпоративной сети. Корпоративная сеть строится иерархически, отражая иерархический характер самого предприятия и его задач. Нижний уровень сети составляют элементы сети — отдельные компьютеры, коммуникационные устройства, каналы передачи данных. На следующем уровне иерархии эти элементы образуют сети разного масштаба — сеть рабочей группы, сеть отдела, сеть отделения и, наконец, сеть предприятия в целом.

Для построения интегрированной системы управления разнородными элемен-

тами сети естественно применить многоуровневый иерархический подход. Это в принципе стандартный подход для построения большой системы любого типа и назначения — от государства до автомобильного завода. Применительно к системам управления сетями наиболее проработанным и эффективным для создания многоуровневой иерархической системы является стандарт Telecommunication Management Network (TMN), разработанный совместными усилиями ITU-T, ISO,

ANSI и ETSI. Хотя этот, стандарт и предназначался изначально для телекоммуникационных сетей, но ориентация на использование общих принципов делает его полезным для построения любой крупной интегрированной системы управления сетями. Стандарты TMN состоят из большого количества рекомендаций ITU-T (и стандартов других организаций), но основные принципы модели TMN описаны в рекомендации М.3Ю.

На каждом уровне иерархии модели TMN решаются задачи одних и тех же пяти функциональных групп, рассмотренных выше (то есть управления конфигурацией, производительностью, ошибками, безопасностью и учетом), однако на каждом уровне эти задачи имеют свою специфику. Чем выше уровень управления, тем более общий и агрегированный характер приобретает собираемая о сети информация, а сугубо технический характер собираемых данных начинает по мере повышения уровня меняться на производственный, финансовый и коммерческий.

Модель TMN упрощенно можно представить в виде двухмерной диаграммы (рис. 19.1).

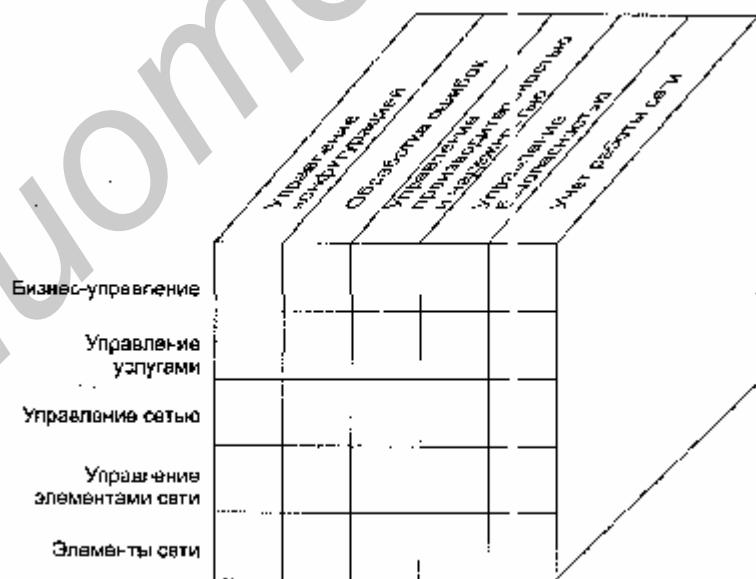


Рис. 19.1. Многоуровневое представление задач управления сетью

Элементы сети

Нижний уровень — уровень элементов сети (Network Element layer, NE) — со-

стоит из отдельных устройств сети: каналов, усилителей, оконечной аппаратуры, мультиплексоров, коммутаторов и т. п. Элементы могут содержать встроенные средства для поддержки управления — датчики, интерфейсы управления, а могут и представлять вещь в себе, требующую для связи с системой управления разработки специального оборудования — устройств связи с объектом, УСО. Современные технологии обычно имеют встроенные функции управления, которые позволяют выполнять хотя бы минимальные операции по контролю за состоянием устройства и за передаваемым устройством трафиком. Подобные функции встроены в технологии FDDI, ISDN, frame relay, SDH. В этом случае устройство всегда можно охватить системой управления, даже если оно не имеет специального блока управления, так как протокол технологии обязывает устройство поддерживать некоторые функции управления. Устройства, которые работают по протоколам, не имеющим встроенных функций контроля и управления, снабжаются отдельным блоком управления, который поддерживает один из двух наиболее распространенных протоколов управления — SNMP или СМIP. Эти протоколы относятся к прикладному уровню модели OSI.

Управление элементами сети

Следующий уровень — уровень управления элементами сети (network element management layer) — представляет собой элементарные системы управления. Элементарные системы управления автономно управляют отдельными элементами сети — контролируют канал связи SDH, управляют коммутатором или мультиплексором. Уровень управления элементами изолирует верхние слои системы управления от деталей и особенностей управления конкретным оборудованием. Этот уровень ответственен за моделирование поведения оборудования и функциональных ресурсов нижележащей сети. Атрибуты этих моделей позволяют управлять различными аспектами поведения управляемых ресурсов. Обычно элементарные системы управления разрабатываются и поставляются производителями оборудования. Примерами таких систем могут служить системы управления CiscoView от Cisco Systems, Optivily от Bay Networks, RAD-View от RAD Data Communications и т. д.

Управление сетью

Выше лежит уровень управления сетью (Network management layer). Этот уровень координирует работу элементарных систем управления, позволяя контролировать конфигурацию составных каналов, согласовывать работу транспортных подсетей разных технологий и т. п. С помощью этого уровня сеть начинает работать как единое целое, передавая данные между своими абонентами.

Управление услугами

Следующий уровень — уровень управления услугами (Service management

layer) — занимается контролем и управлением за транспортными и информационными услугами, которые предоставляются конечным пользователям сети. В задачу этого уровня входит подготовка сети к предоставлению определенной услуги, ее активизация, обработка вызовов клиентов. Формирование услуги (service provisioning) заключается в фиксации в базе данных значений параметров услуги, например, требуемой средней пропускной способности, максимальных величин задержек пакетов, коэффициента готовности и т. гг. В функции этого уровня входит также выдача уровню управления сетью задания на конфигурирование виртуального или физического канала связи для поддержания услуги. После формирования услуги данный уровень занимается контролем за качеством ее реализации, то есть за соблюдением сетью всех принятых на себя обязательств в отношении производительности и надежности транспортных услуг. Результаты контроля качества обслуживания нужны, в частности, для подсчета оплаты за пользование услугами клиентами сети. Например, в сети frame relay уровень управления услугами следит за заказанными пользователем значениями средней скорости CIR и согласованной пульсации Be, фиксируя нарушения со стороны пользователя и сети.

Бизнес-управление

Уровень бизнес-управления (Business management layer) занимается вопросами долговременного планирования сети с учетом финансовых аспектов деятельности организации, владеющей сетью. На этом уровне помесячно и поквартально подсчитываются доходы от эксплуатации сети и ее отдельных составляющих, учитываются расходы на эксплуатацию и модернизацию сети, принимаются решения о развитии сети с учетом финансовых возможностей. Уровень бизнес-управления обеспечивает для пользователей и поставщиков услуг возможность предоставления дополнительных услуг. Этот уровень является частным случаем уровня автоматизированной системы управления предприятием (АСУП), в то время как все нижележащие уровни соответствуют уровням автоматизированной системы управления технологическими процессами (АСУТП), для такого специфического типа предприятия, как телекоммуникационная или корпоративная сеть. Но если телекоммуникационная сеть чаще всего является основой телекоммуникационной компании, то корпоративную сеть и обслуживающий ее персонал трудно назвать предприятием. Тем не менее на некоторых западных фирмах корпоративная сеть выделена в автономное производственное подразделение со своим бюджетом и со своими финансовыми договорами на обслуживание, которое данное подразделение заключает с основными производственными подразделениями предприятия.

Выводы

- Система управления большой сетью должна иметь многоуровневую иерархическую структуру, которая обеспечит масштабируемость решения.

- Набор стандартов Telecommunication Management Network (TMN) описывает функции уровней иерархической системы управления телекоммуникационными системами.
- Каждый из уровней модели TMN реализует пять групп функций системы управления в соответствии с рекомендацией X.700, но делает это в различных масштабах и с различной степенью детализации задач.

Архитектуры систем управления сетями

Выделение в системах управления типовых групп функций и разбиение этих функций на уровни еще не дает ответа на вопрос, каким же образом устроены системы управления, из каких элементов они состоят и какие архитектуры связей этих элементов используются на практике.

Схема менеджер-агент

В основе любой системы управления сетью лежит элементарная схема взаимодействия агента с менеджером. На основе этой схемы могут быть построены системы практически любой сложности с большим количеством агентов и менеджеров разного типа.

Схема «менеджер-агент» представлена на рис. 19.2.



Рис. 19.2. Взаимодействие агента менеджера и управляемого ресурса

Агент является посредником между управляемым ресурсом и основной управляющей программой-менеджером. Чтобы один и тот же менеджер мог управлять различными реальными ресурсами, создается некоторая модель управляемого ресурса, которая отражает только те характеристики ресурса, которые нужны для его контроля. Например, модель маршрутизатора обычно включает такие характеристики, как количество портов, их тип, таблицу маршрутизации, количество кадров и пакетов протоколов канального, сетевого и транспортного уровней, прошедших через эти порты.

Менеджер получает от агента только те данные, которые описываются моделью ресурса. Агент же является некоторым экраном, освобождающим менеджера от ненужной информации о деталях реализации ресурса. Агент поставляет менед-

жеру обработанную и представленную в нормализованном виде информацию. На основе этой информации менеджер принимает решения по управлению, а также выполняет дальнейшее обобщение данных о состоянии управляемого ресурса, например строит зависимость загрузки порта от времени.

Для получения требуемых данных от объекта, а также для выдачи на него управляющих воздействий агент взаимодействует с реальным ресурсом некоторым нестандартным способом. Когда агенты встраиваются в коммуникационное оборудование, то разработчик оборудования предусматривает точки и способы взаимодействия внутренних узлов устройства с агентом. При разработке агента для операционной системы разработчик агента пользуется теми интерфейсами, которые существуют в этой ОС, например интерфейсами ядра, драйверов и приложений. Агент может снабжаться специальными датчиками для получения информации, например датчиками релейных контактов или датчиками температуры.

Менеджер и агент должны располагать одной и той же моделью управляемого ресурса, иначе они не смогут понять друг друга. Однако в использовании этой модели агентом и менеджером имеется существенное различие. Агент наполняет модель управляемого ресурса текущими значениями характеристик данного ресурса, и в связи с этим модель агента называют базой данных управляющей информации (Management Information Base, MIB). Менеджер использует модель, чтобы знать о том, чем характеризуется ресурс, какие характеристики он может запросить у агента и какими параметрами можно управлять.

Менеджер взаимодействует с агентами по стандартному протоколу. Этот протокол должен позволять менеджеру запрашивать значения параметров, хранящихся в базе MIB, а также передавать агенту управляющую информацию, на основе которой тот должен управлять устройством. Различают управление *in-band*, то есть по тому же каналу, по которому передаются пользовательские данные, и управление *out-of-band*, то есть вне канала, по которому передаются пользовательские данные. Например, если менеджер взаимодействует с агентом, встроенным в маршрутизатор, по протоколу SNMP, передаваемому по той же локальной сети, что и пользовательские данные, то это будет управление *in-band*. Если же менеджер контролирует коммутатор первичной сети, работающий по технологии частотного уплотнения FDM, с помощью отдельной сети X.25, к которой подключен агент, то это будет управление *out-of-band*. Управление по тому же каналу, по которому работает сеть, более экономично, так как не требует создания отдельной инфраструктуры передачи управляющих данных. Однако способ *out-of-band* более надежен, так как он предоставляет возможность управлять оборудованием сети и тогда, когда какие-то элементы сети выходят из строя и через основные каналы оборудование оказывается недоступным.

Стандарт многоуровневой системы управления TMN имеет в своем названии слово Network, подчеркивающее, что в общем случае для управления телекоммуникационной сетью создается отдельная управляющая сеть, которая обеспечивает режим *out-of-band*.

Обычно менеджер работает с несколькими агентами, обрабатывая получаемые

от них данные и выдавая на них управляющие воздействия. Агенты могут встраиваться в управляемое оборудование, а могут и работать на отдельном компьютере, связанном с управляемым оборудованием по какому-либо интерфейсу. Менеджер обычно работает на отдельном компьютере, который выполняет также роль консоли управления для оператора или администратора системы.

Модель менеджер-агент лежит в основе таких популярных стандартов управления, как стандарты Интернета на основе протокола SNMP и стандарты управления ISO/OSI на основе протокола CMIP.

Агенты могут отличаться различным уровнем интеллекта — они могут обладать как самым минимальным интеллектом, необходимым для подсчета проходящих через оборудование кадров и пакетов, так и весьма высоким, достаточным для выполнения самостоятельных действий по выполнению последовательности управляющих команд в аварийных ситуациях, построению временных зависимостей, фильтрации аварийных сообщений и т. п.

Структуры распределенных систем управления

В крупной корпоративной сети полностью централизованная система управления, построенная на базе единственного менеджера, вряд ли будет работать хорошо по нескольким причинам. Во-первых, такой вариант не обеспечивает необходимой масштабируемости по производительности, так как единственный менеджер вынужден будет обрабатывать весь поток сообщений от всех агентов, что при нескольких тысячах управляемых объектов потребует очень высокопроизводительной платформы для работы менеджера и перегрузит служебной управляющей информацией каналы передачи данных в той сети, где будет расположен менеджер. Во-вторых, такое решение не обеспечит необходимого уровня надежности, так как при отказе единственного менеджера окажется парализованным управление сетью. В-третьих, в большой распределенной сети целесообразно располагать в каждом географическом пункте отдельным оператором или администратором, управляющим своей частью сети, а это удобнее реализовать с помощью отдельных менеджеров для каждого оператора.

Схема «менеджер-агент» позволяет строить достаточно сложные в структурном отношении распределенные системы управления.

Обычно распределенная система управления включает большое количество связок менеджер-агент, дополненных рабочими станциями операторов сети, с помощью которых они получают доступ к менеджерам (рис. 19.3).



Рис. 19.3. Распределенная система управления на основе нескольких менеджеров и рабочих станций

Каждый агент собирает данные и управляет определенным элементом сети. Менеджеры, иногда также называемые серверами системы управления, собирают данные от своих агентов, обобщают их и хранят в базе данных. Операторы, работающие за рабочими станциями, могут соединиться с любым из менеджеров и с помощью графического интерфейса просмотреть данные об управляемой сетью, а также выдать менеджеру некоторые директивы по управлению сетью или ее элементами.

Наличие нескольких менеджеров позволяет распределить между ними нагрузку по обработке данных управления, обеспечивая масштабируемость системы. Как правило, связи между агентами и менеджерами носят более упорядоченный характер, чем тот, который показан на рис. 19.3. Чаще всего используются два подхода к их соединению — одноранговый (рис. 19.4) и иерархический (рис. 19.5).

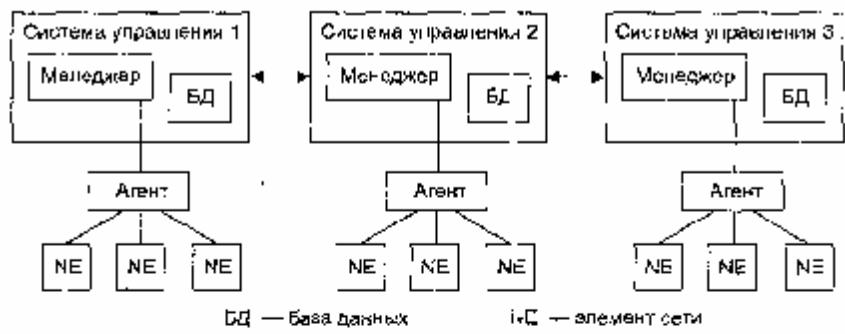


Рис. 19.4. Одноранговые связи между менеджерами

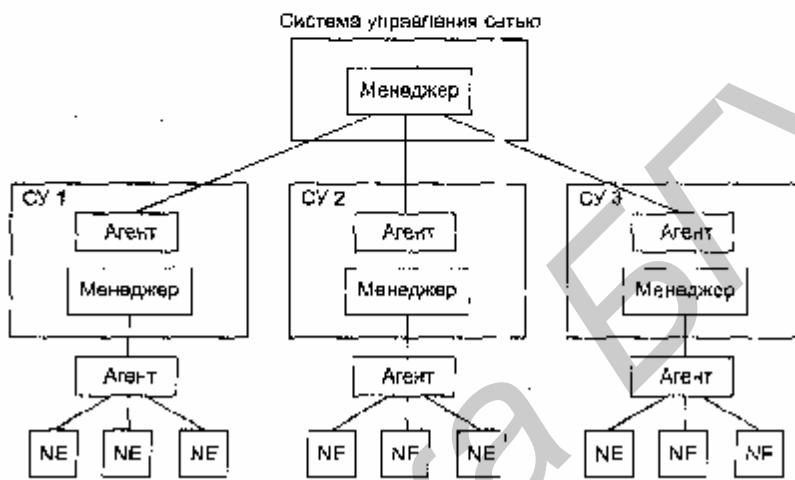


Рис. 19.5. Иерархические связи между менеджерами

В случае одноранговых связей каждый менеджер управляет своей частью сети на основе информации, получаемой от нижележащих агентов. Центральный менеджер отсутствует. Координация работы менеджеров достигается за счет обмена информацией между базами данных каждого менеджера.

Одноранговое построение системы управления сегодня считается неэффективным и устаревшим. Обычно оно вызвано тем обстоятельством, что элементарные системы управления построены как монолитные системы, которые первоначально не были ориентированы на модульность (например, многие системы управления, разработанные производителями оборудования, не поддерживают стандартные интерфейсы для взаимодействия с другими системами управления). Затем эти менеджеры нижнего уровня стали объединяться для создания интегрированной системы управления сетью, но связи между ними оказалось возможным создавать только на уровне обмена между базами данных, что достаточно медленно. Кроме того, в базах данных таких менеджеров накапливается слишком детальная информация об управляемых элементах сети (так как первоначально эти менеджеры разрабатывались как менеджеры нижнего уровня), вследствие чего собранная информация оказывается малопригодной для координации работы всей сети в целом. Такой подход к построению системы управления называется подходом «снизу вверх».

Гораздо более гибким является иерархическое построение связей между менеджерами. Каждый менеджер нижнего уровня выполняет также функции

агента для менеджера верхнего уровня. Такой агент работает уже с гораздо более укрупненной моделью (МТВ) своей части сети, в которой собирается именно та информация, которая нужна менеджеру верхнего уровня для управления сетью в целом. Обычно для разработки моделей сети на разных уровнях проектирование начинают с верхнего уровня, на котором определяется состав информации, требуемой от менеджеров-агентов более низкого уровня, поэтому такой подход назван подходом «сверху вниз». Он сокращает объемы информации, циркулирующей между уровнями системы управления, и приводит к гораздо более эффективной системе управления.

Модель TMN в наибольшей степени соответствует иерархической архитектуре связей между менеджерами, хотя известны реализации принципов TMN и в одноуровневых архитектурах.

Платформенный подход

При построении систем управления крупными локальными и корпоративными сетями обычно используется платформенный подход, когда индивидуальные программы управления разрабатываются не «с нуля», а задействуют службы и примитивы, предоставляемые специально разработанным для этих целей программным продуктом — платформой. Примерами платформ для систем управления являются такие известные продукты, как HP OpenView, SunNet Manager и Sun Soltice, Cabletron Spectrum, IMB/Tivoli TMN10.

Эти платформы создают общую операционную среду для приложений системы управления точно так же, как универсальные операционные системы, такие как UNIX или Windows NT, создают операционную среду для приложений любого типа, таких как MS Word, Oracle и т. п. Платформа обычно включает поддержку протоколов взаимодействия менеджера с агентами — SNMP и реже CMIP, набор базовых средств для построения менеджеров и агентов, а также средства графического интерфейса для создания консоли управления. В набор базовых средств обычно входят средства построения карты сети, фильтрации сообщений от агентов, ведения базы данных. Набор интерфейсных функций платформы образует интерфейс прикладного программирования (API) системы управления. Пользуясь этим API, сторонние производители создают законченные системы управления, которые могут управлять специфическим оборудованием в соответствии с пятью основными группами функций.

Обычно платформа управления поставляется с каким-либо универсальным менеджером, который может выполнять некоторые базовые функции управления без программирования. Чаще всего к этим функциям относятся функции построения карты сети (группа Configuration Management), а также функции отображения состояния управляемых устройств и функции фильтрации сообщений об ошибках (группа Fault Management). Например, одна из наиболее популярных платформ HP OpenView поставляется с менеджером Network Node Manager, который реализует перечисленные функции.

Чем больше функций выполняет платформа, тем лучше. В том числе и таких,

которые нужны для разработки любых аспектов работы приложений, прямо не связанных со спецификой управления. В конце концов, приложения системы управления — это, прежде всего, приложения, а потом уже приложения системы управления. Поэтому полезны любые средства, предоставляемые платформой, которые ускоряют разработку приложений вообще и распределенных приложений в частности.

Компании, производящие коммуникационное оборудование, разрабатывают дополнительные менеджеры для популярных платформ, которые выполняют функции управления оборудованием данного производителя более полно. Примерами таких менеджеров могут служить менеджеры системы Optivity компании Bay Networks и менеджеры системы Trancsend компании 3Com, которые могут работать в среде платформ HP OpenView и SunNet Manager.

Выводы

- В основе всех систем управления сетями лежит схема «менеджер—агент». Эта схема использует абстрактную модель управляемого ресурса, называемую базой управляющей информации — Management Information Base, МТВ.
- Агент взаимодействует с управляемым ресурсом по нестандартному интерфейсу, а с менеджером — по стандартному протоколу через сеть.
- В больших системах управления используется несколько менеджеров, которые взаимодействуют друг с другом по одной из двух схем — одноранговой или иерархической.
- Иерархическая схема взаимодействия менеджеров соответствует стандартам TMN и является более перспективной.
- При построении систем управления активно используется платформенный подход. Платформа системы управления выполняет для менеджеров роль операционной системы для обычных приложений, так как обеспечивает разработчика менеджеров набором полезных системных вызовов общего для любой системы управления назначения.

ГЛАВА 20 Стандарты систем управления

Кроме стандартов, определяющих в наиболее общем виде функции системы управления, существуют более конкретные стандарты, которые определяют функции, внутреннюю структуру и протоколы взаимодействия менеджеров и агентов.

При формализации схемы «менеджер-агент» могут быть стандартизованы следующие аспекты ее функционирования:

- протокол взаимодействия агента и менеджера;
- интерфейс «агент-управляемый ресурс»;
- интерфейс «агент-модель управляемого ресурса»;
- интерфейс «менеджер-модель управляемого ресурса»;
- справочная система о наличии и местоположении агентов и менеджеров, упрощающая построение распределенной системы управления;
- язык описания моделей управляемых ресурсов, то есть язык описания МТБ;
- схема наследования классов моделей объектов (дерево наследования), которая позволяет строить модели новых объектов на основе моделей более общих объектов, например, модели маршрутизаторов на основе модели обобщенного коммуникационного устройства;
- схема иерархических отношений моделей управляемых объектов (дерево включения), которая позволяет отразить взаимоотношения между отдельными элементами реальной системы, например, принадлежность модулей коммутации определенному коммутатору или отдельных коммутаторов и концентраторов определенной подсети.

Существующие стандарты на системы управления отличаются тем, что в них может быть стандартизованы не все перечисленные выше аспекты схемы «менеджер-агент».

В стандартах систем управления как минимум стандартизуется некоторый способ формального описания моделей управляемых объектов, а также определяется протокол взаимодействия между менеджером и агентом.

Сегодня на практике применяются два семейства стандартов управления сетями — стандарты Интернета, построенные на основе протокола SNMP (Simple Network Management Protocol), и международные стандарты ISO/ITU-T, использующие в качестве протокола взаимодействия агентов и менеджеров протокол CMIP (Common Management Information Protocol).

Стандарты систем управления, основанных на протоколе SNMP, формализуют минимум аспектов системы управления, а стандарты ISO/ITU-T — максимум аспектов, как и большинство стандартов, разработанных ITU-T. Традиционно в локальных и корпоративных сетях применяются в основном системы управления на основе SNMP, а стандарты ISO/ITU-T и протокол CMTP находят применение в телекоммуникационных сетях.

Стандарты систем управления на основе протокола SNMP

Концепции SNMP-управления

В системах управления, построенных на основе протокола SNMP, стандартизуются следующие элементы:

- протокол взаимодействия агента и менеджера;
- язык описания моделей MIB и сообщений SNMP — язык абстрактной синтаксической нотации ASN.1 (стандарт ISO 8824:1987, рекомендации ITU-T X.208);
- несколько конкретных моделей MIB (MIB-I, MIB-II, RMON, RMON 2), имена объектов которых регистрируются в дереве стандартов ISO.

Все остальное отдается «на откуп» разработчику системы управления.

Протокол SNMP и тесно связанная с ним концепция SNMP МТВ были разработаны для управления маршрутизаторами Интернета как временное решение. Но, как это часто бывает со всем временным, простота и эффективность решения обеспечили успех этого протокола, и сегодня он используется для управления практически любыми видами оборудования и программного обеспечения вычислительных сетей. И хотя в области управления телекоммуникационными сетями наблюдается устойчивая тенденция применения стандартов ITU-T, в которые входит протокол CMIP, и здесь имеется достаточно много примеров успешного использования SNMP-управления. Агенты SNMP встраиваются в аналоговые модемы, модемы ADSL, коммутаторы ATM и т. д.

SNMP — это протокол прикладного уровня, разработанный для стека TCP/IP, хотя имеются его реализации и для других стеков, например IPX/SPX. Протокол SNMP используется для получения от сетевых устройств информации об их статусе, производительности и других характеристиках, которые хранятся в базе данных управляющей информации MIB (Management Information Base). Простота SNMP во многом определяется простотой MIB SNMP, особенно их первых версий MIB-I и MIB-II. Кроме того, сам протокол SNMP также весьма несложен.

Существуют стандарты, определяющие структуру MIB, в том числе набор типов объектов этой базы, их имена и допустимые операции над ними (например, «читать»).

Древовидная структура MIB содержит обязательные (стандартные) поддеревья, а также в ней могут находиться частные (private) поддеревья, позволяющие изготовителю интеллектуальных устройств управлять какими-либо специфическими функциями устройства на основе специфических объектов MIB.

Агент в протоколе SNMP — это обрабатывающий элемент, который обеспечивает менеджерам, размещенным на управляющих станциях сети, доступ к значениям переменных MIB и тем самым дает им возможность реализовывать функции по управлению и наблюдению за устройством.

Основные операции по управлению вынесены в менеджер, а агент SNMP выполняет чаще всего пассивную роль, передавая в менеджер по его запросу зна-

чения накопленных статистических переменных. При этом устройство работает с минимальными издержками на поддержание управляющего протокола. Оно использует почти всю свою вычислительную мощность для выполнения своих основных функций маршрутизатора, моста или концентратора, а агент занимается сбором статистики и значений переменных состояния устройства и передачей их менеджеру системы управления.

Примитивы протокола SNMP

SNMP — это протокол типа «запрос-ответ», то есть на каждый запрос, поступивший от менеджера, агент должен передать ответ. Особенностью протокола является его чрезвычайная простота — он включает в себя всего несколько команд.

- Команда Get-request используется менеджером для получения от агента значения какого-либо объекта по его имени.
- Команда GetNext-request используется менеджером для извлечения значения следующего объекта (без указания его имени) при последовательном просмотре таблицы объектов.
- С помощью команды Get-response агент SNMP передает менеджеру ответ на команды Get-request или GetNext-request.
- Команда Set позволяет менеджеру изменять значения какого-либо объекта. С помощью команды Set и происходит собственно управление устройством. Агент должен понимать смысл значений объекта, который используется для управления устройством, и на основании этих значений выполнять реальное управляющее действие — отключить порт, приписать порт определенной VLAN и т. п. Команда Set пригодна также для установки условия, при выполнении которого агент SNMP должен дослать менеджеру соответствующее сообщение. Может быть определена реакция на такие события, как инициализация агента, рестарт агента, обрыв связи, восстановление связи, неверная аутентификация и потеря ближайшего маршрутизатора. Если происходит любое из этих событий, то агент инициализирует прерывание.
- Команда Trap используется агентом для сообщения менеджеру о возникновении особой ситуации.
- Версия SNMP v.2 добавляет к этому набору команду GetBulk, которая позволяет менеджеру получить несколько значений переменных за один запрос.

Структура SNMP MIB

На сегодня существует несколько стандартов на базы данных управляющей информации для протокола SNMP. Основными являются стандарты MIB-I и MIB-II, а также версия базы данных для удаленного управления RMON MIB. Кроме этого существуют стандарты для специальных устройств MIB конкретного типа (например, MIB для концентраторов или MIB для модемов), а также частные МШ конкретных фирм-производителей оборудования.

Первоначальная спецификация MIB-1 определяла только операции чтения значений переменных. Операции изменения или установки значений объекта являются частью спецификаций МТВ-П.

Версия MIB-I (RFC 1156) определяет 114 объектов, которые подразделяются на 8 групп.

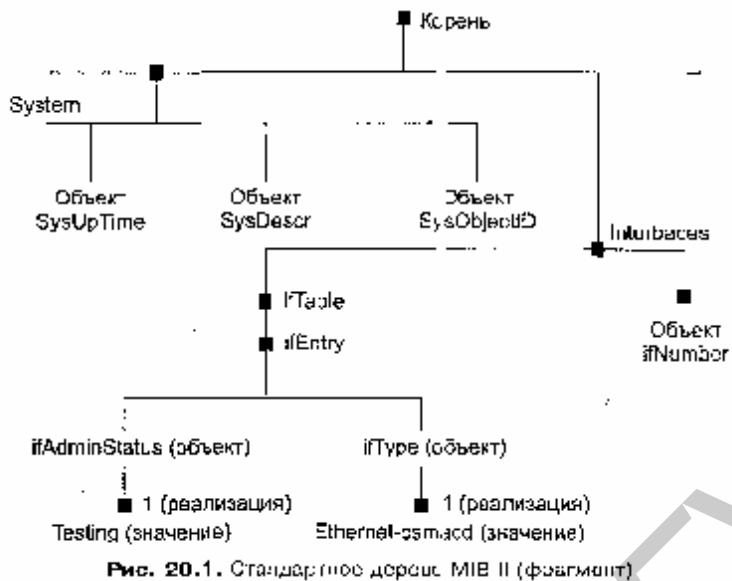
- System — общие данные об устройстве (например, идентификатор поставщика, время последней инициализации системы).
- Interfaces — параметры сетевых интерфейсов устройства (например, их количество, типы, скорости обмена, максимальный размер пакета).
- Address Translation Table — описание соответствия между сетевыми и физическими адресами (например, по протоколу ARP).
- Internet Protocol — данные, относящиеся к протоколу IP (адреса IP-шлюзов, хостов, статистика о IP-пакетах).
- ICMP — данные, относящиеся к протоколу обмена управляющими сообщениями ICMP.
- TCP — данные, относящиеся к протоколу TCP (например, о TCP-соединениях).
- UDP — данные, относящиеся к протоколу UDP (число переданных, принятых и ошибочных UDP-дейтаграмм).
- EGP — данные, относящиеся к протоколу обмена маршрутной информацией Exterior Gateway Protocol, используемому в Интернете (число принятых с ошибками и без ошибок сообщений).

Из этого перечня групп переменных видно, что стандарт MIB-I разрабатывался с жесткой ориентацией на управление маршрутизаторами, поддерживающими протоколы стека TCP/IP.

В версии MIB-II (RFC 1213), принятой в 1992 году, был существенно (до 185) расширен набор стандартных объектов, а число групп увеличилось до 10.

На рис. 20.1 приведен пример древовидной структуры базы объектов MIB-II. На нем показаны две из 10 возможных групп объектов — System (имена объектов начинаются с префикса Sys) и Interfaces (префикс if). Объект SysUpTime содержит значение продолжительности времени работы системы с момента последней перезагрузки, объект SysObjectID — идентификатор устройства (например, маршрутизатора).

Объект if Number определяет количество сетевых интерфейсов устройства, а объект If Entry является вершиной поддерева, описывающего один из конкретных интерфейсов устройства. Входящие в это поддерево объекты if Type и if AdminStatus определяют соответственно тип и состояние одного из интерфейсов, в данном случае интерфейса Ethernet.



В число объектов, описывающих каждый конкретный интерфейс устройства, включены следующие.

- if Type — тип протокола, который поддерживает интерфейс. Этот объект принимает значения всех стандартных протоколов канального уровня, например rfc877-x25, ethernet-csmacd, iso88023-csmacd, iso88024-tokenBus, iso88025-token-Ring и т. д.
 - ifMtu — максимальный размер пакета сетевого уровня, который можно послать через этот интерфейс.
 - if Speed — пропускная способность интерфейса в битах в секунду (100 для Fast Ethernet).
 - if PhysAddress — физический адрес порта, для Fast Ethernet им будет MAC-адрес.
 - ifAdminStatus — желаемый статус порта: О up — готов передавать пакеты; О down — не готов передавать пакеты; О testing — находится в тестовом режиме.
 - ifOperStatus — фактический текущий статус порта, имеет те же значения, что HifAdminStatus.
 - if InOctets — общее количество байтов, принятое данным портом, включая служебные, с момента последней инициализации SNMP-агента.
 - if InUcastPkts — количество пакетов с индивидуальным адресом интерфейса, доставленных протоколу верхнего уровня.
 - if InNUcastPkts — количество пакетов с широковещательным или мультиплексорным адресом интерфейса, доставленных протоколу верхнего уровня.
 - ifInDiscards — количество пакетов, которые были приняты интерфейсом, оказались корректными, но не были доставлены протоколу верхнего уровня, скорее всего из-за переполнения буфера пакетов или же по иной причине.
 - ifInErrors — количество пришедших пакетов, которые не были переданы протоколу верхнего уровня из-за обнаружения в них ошибок.
- Кроме объектов, описывающих статистику по входным пакетам, имеются аналогичные объекты, но относящиеся к выходным пакетам,

Как видно из описания объектов MIB-II, эта база данных не дает детальной статистики по характерным ошибкам кадров Ethernet, кроме этого, она не отражает изменение характеристик во времени, что часто интересует сетевого администратора.

Эти ограничения были впоследствии сняты новым стандартом на MTB — RMON MIB, который специально ориентирован на сбор детальной статистики по протоколу Ethernet, к тому же с поддержкой такой важной функции, как построение агентом зависимостей статистических характеристик от времени.

Форматы и имена объектов SNMP MIB

Для именования переменных базы MTB и однозначного определения их форматов используется дополнительная спецификация, называемая SMI (Structure of Management Information). Например, спецификация SMI включает в качестве стандартного имя IpAddress и определяет его формат как строку из 4 байт. Другой пример — имя Counter, для которого определен формат в виде целого числа в диапазоне от 0 до $2^{32} - 1$.

При описании переменных MIB и форматов протокола SNMP спецификация SMI опирается на формальный язык ASN.1, принятый ISO в качестве нотации для описания терминов коммуникационных протоколов (правда, многие коммуникационные протоколы, например IP, PPP или Ethernet, обходятся без этой нотации). Нотация ASN.1 служит для установления однозначного соответствия между терминами, взятыми из стандартов, предназначенных для людей, и теми данными, которые передаются в коммуникационных протоколах аппаратурой. Достигаемая однозначность очень важна для гетерогенной среды, характерной для корпоративных сетей. Так, вместо того чтобы указать, что некоторая переменная протокола представляет собой целое число, разработчик протокола, использующий нотацию ASN.1, должен точно определить формат и допустимый диапазон переменной. В результате документация на MIB, написанная с помощью нотации ASN.1, может точно и механически транслироваться в форму кодов, характерных для сообщений протоколов.

Нотация ASN.1 похожа на другие метаязыки, например нормальную Бэкусову форму, используемую при описании языков программирования, в частности Алгола. Нотация ASN.1 поддерживает базовый набор различных типов данных, таких как целое число, строка и т. п., а также позволяет конструировать из этих базовых типов составные данные — массивы, перечисления, структуры.

Существуют правила трансляции структур данных, описанных на ASN.1, в структуры данных языков программирования, например C++. Соответственно, имеются трансляторы, выполняющие эту работу. Примеры описаний данных с помощью ASN.1 приведены ниже для протокольных блоков данных SNMP.

Нотация ASN.1 широко используется при описании многих стандартов OS1, в частности моделей управляемых объектов и структуры сообщений протокола

СМIP.

Имена переменных MIB могут быть записаны как в символьном, так и в числовом форматах. Символьный формат используется для представления переменных в текстовых документах и на экране дисплея, а числовые имена ~ в сообщениях протокола SNMP. Например, символьному имени SysDescr соответствует числовое имя 1, а точнее 1.3.6.1.2.1.1.1.

Составное числовое имя объекта SNMP M1B соответствует полному имени этого объекта в дереве регистрации объектов стандартизации ISO. Разработчики протокола SNMP не стали использовать традиционный для стандартов Интернета способ фиксации численных параметров протокола в специальном документе RFC, называемом «Assigned Numbers» (там описываются, например, численные значения, которые может принимать поле

Protocol пакета IP и т. п.). Вместо этого они зарегистрировали объекты баз MIB SNMP во всемирном дереве регистрации стандартов ISO, показанном па рис. 20.2.

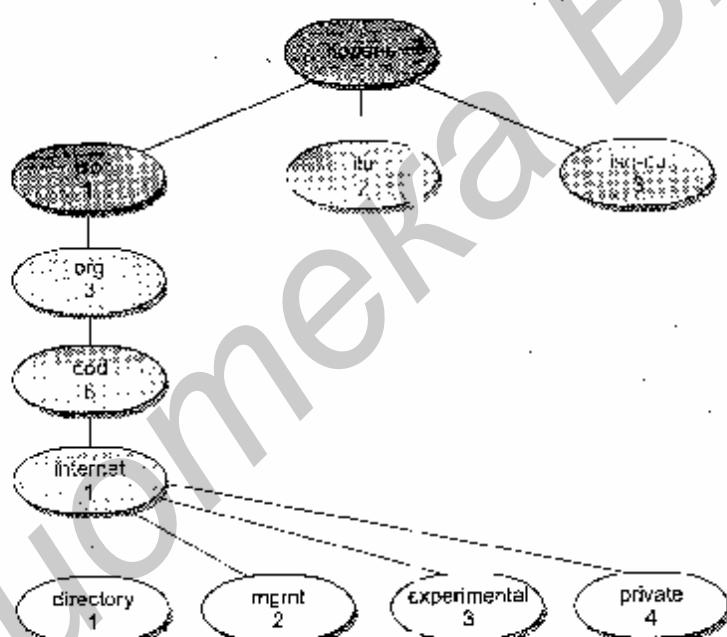


Рис. 20.2. Пространство имен объектов ISO

Как и в любых сложных системах, пространство имен объектов ISO имеет древовидную иерархическую структуру, причем на рисунке показана только его верхняя часть. От корня этого дерева отходят три ветви, соответствующие стандартам, контролируемым ISO, ITU и совместно ISO-ITU. В свою очередь, организация ISO создала ветвь для стандартов, создаваемых национальными и международными организациями (ветвь org). Стандарты Интернета создавались под эгидой Министерства обороны США (Departament of Defence, DoD), поэтому стандарты МШ попали в поддерево dod-internet, а далее, естественно, в группу стандартов управления сетью — ветвь mgmt. Объекты любых стандартов, создаваемых под эгидой ISO, однозначно идентифицируются составными

символьными именами, начинающимися от корня этого дерева. В сообщениях протоколов используются не символьные имена, а однозначно соответствующие им составные числовые имена. Каждая ветвь дерева имен объектов нумеруется в дереве целыми числами слева направо, начиная с единицы, и эти числа и заменяют символьные имена. Поэтому полное символьное имя объекта МШ имеет вид 1so.org. dod. internet .mgmt .mib, а полное числовое имя — 1.3.6.1.2.1. Группа объектов private (4) зарезервирована за стандартами, создаваемыми частными компаниями, например Cisco, Hewlett-Packard и т. п. Это же дерево регистрации используется для именования классов объектов СМIP и TMN. Соответственно, каждая группа объектов MIB-I и МШ-11 также имеет кроме кратких символьных имен, приведенных выше, полные символьные имена и соответствующие им числовые имена. Например, краткое символьное имя группы System имеет полную форму iso.org.dod. internet.mgmt.mib.system, а ее соответствующее числовое имя — 1.3.6.1.2.1. Часть дерева имен ISO, включающая группы объектов MIB, показана на рис. 20.3.

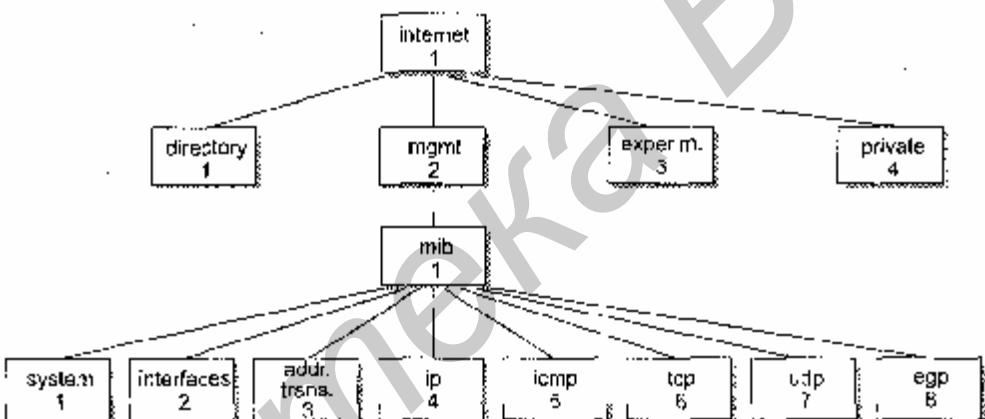


Рис. 20.3. Часть дерева имен ISO, включающая группы объектов MIB-I

Формат сообщений SNMP

Протокол SNMP обслуживает передачу данных между агентами и станцией, управляющей сетью. SNMP использует дейтаграммный транспортный протокол UDP, не обеспечивающий надежной доставки сообщений. Протокол, организующий надежную передачу дейтаграмм на основе соединений TCP, весьма загружает управляемые устройства, которые на момент разработки протокола SNMP были не очень мощные, поэтому от услуг протокола TCP решили отказаться.

SNMP часто рассматривают только как решение для управления сетями TCP/IP. Хотя SNMP чаще всего и работает над UDP (и над TCP), он может работать и над транспортными сетевыми протоколами стека OSI — ТР0, ТР4, CNLS, а также над протоколами MAC-уровня. Растет поддержка протокола SNMP и в других транспортных средах. Например, фирма Novell начала поддерживать

протокол SNMP с версии NetWare 3.11, а некоторые производители оборудования (например, Bay Networks) реализуют в своих устройствах передачу сообщений SNMP с помощью как IP, так и IPX.

Сообщения SNMP, в отличие от сообщений многих других коммуникационных протоколов, не имеют заголовков с фиксированными нолями. В соответствии с нотацией ASN.1 сообщение SNMP состоит из произвольного количества нолей, и каждое поле предваряется описателем его типа и размера.

Любое сообщение SNMP состоит из трех основных частей: версии протокола (version), идентификатора общности (community), используемого для группирования устройств, управляемых определенным менеджером, и области данных, в которой собственно и содержатся описанные выше команды протокола, имена объектов и их значения. Область данных делится на блоки данных протокола (Protocol Data Unit, PDU).

Общий формат сообщения SNMP в нотации ASN.1 выглядит следующим образом:

```
SNMP-Message ::=  
SEQUENCE {  
    version INTEGER {  
        version-1 (0)  
    }.  
    community  
    OCTET STRING.  
    SNMP-PDUs  
    ANY  
}
```

Область данных может содержать пять различных типов блоков PDU, соответствующих пяти командам протокола SNMP:

```
SNMP-PDUs ::=  
CHOICE {  
    get-request  
    GetRequest-PDU,  
    get-next-request  
    GetNextRequest-PDU.  
    get-response  
    GetResponse-PDU,  
    set-request  
    SetRequest-PDU.  
    trap  
    Trap-FDD.  
}
```

И, наконец, для каждого типа блока PDU имеется определение его формата. Например, формат блока GetRequest-PDU описан следующим образом:

```
GetRequest-PDU ::=  
IMPLICIT SEQUENCE {  
    request-ID  
    RequestID,  
    error-status  
    Error Status.  
    error-index
```

```

ErrorIndex,
variable-bindings
VarBindList
}

```

Далее стандарт SNMP определяет соответственно формат переменных блока GetRequest-PDU. Переменная RequestID — это 4-байтовое целое число (используется для установления соответствия ответов запросам), ErrorStatus и Error-Index — это однобайтовые целые, которые в запросе должны быть установлены в 0. VarBindList — это список числовых имен объектов, значениями которых интересуется менеджер. В нотации ASN.1 этот список состоит из пар «имя-значение». При запросе значение переменной должно быть установлено в null. Вот пример сообщения протокола SNMP, которое представляет собой запрос о значении объекта SysDescr (числовое имя 1.3.6.1.2.1.1.1).

30	2B	02	C	00				
SEQUENCE	len = 4*	INTEGER	len = 1	vers = 0				
04	06	70	75	82	6C	69	03	
string	len = 6	P	u	b	I	I	c	
AD	0C	02	04	05	AE	58	02	
getreq	len = 28	INTEGER	len = 4	-----	requested	-----	-----	
02	01	00	02	01	30	07	00	
INTEGER	len = 1	status	INTEGER	len = 1	error	index		
SU	CE	30	30	06	08	00		
SEQUENCE	len = 14	SEQUENCE	len = 12	object	len = 8			
2B	06	01	02	01	01	C1	00	
1.3	6	1	2	1	1	1	0	
05	CD							
null	len = 0							

Как видно из описания, сообщение начинается с кода 30 (все коды шестнадцатеричные), который соответствует ключевому слову SEQUENCE (последовательность). Длина последовательности указывается в следующем байте (41 байт). Далее следует целое число длиной один байт — это версия протокола SNMP (в данном случае 0, то есть SNMP v.1, а 1 означала бы SNMP v.2). Поле community имеет тип string (строка символов) длиной о 6 байт со значением public. Остальную часть сообщения составляет блок данных GetRequest-PDU. То, что это операция Get-request, говорит код АО (это значение определено в протоколе SNMP, а не в нотации ASN.1), а общая длина блока данных — 28 байт. В соответствии со структурой блока Getrequest-PDU далее идет идентификатор запроса (он определен как целое 4-байтовое число). Затем в блоке следуют два однобайтовых целых числа статуса и индекса ошибки, которые в запросе установлены в 0. И наконец, завершает сообщение список объектов, состоящий из одной пары — имени 1.3.6.1.2.1.1.0 и значения null.

Спецификация RMON MIB

Новейшим добавлением к функциональным возможностям SNMP является спецификация RMON, которая обеспечивает удаленное взаимодействие с базой MIB. До появления RMON протокол SNMP не мог использоваться удаленным образом, он допускал только локальное управление устройствами. База RMON MIB обладает улучшенным набором свойств для удаленного управления, так как содержит агрегированную информацию об устройстве, не требующую передачи по сети больших объемов информации. Объекты RMON MIB включают дополнительные счетчики ошибок в пакетах, более гибкие средства анализа трендов и статистики, более мощные средства фильтрации для захвата и анализа отдельных пакетов, а также более сложные условия установления сигналов предупреждения. Интеллектуальность агентов RMON MIB выше, чем агентов MIB-I или MIB-II, что позволяет им выполнять значительную часть работы по обработке информации об устройстве, которую раньше выполняли менеджеры. Эти агенты могут располагаться внутри различных коммуникационных устройств или выполняться в виде отдельных программных модулей, работающих на универсальных персональных компьютерах и ноутбуках.

Объекту RMON присвоен номер 16 в наборе объектов MIB, а сам объект RMON объединяет 10 групп объектов (десятую группу составляют специальные объекты протокола Token Ring).

- Statistics (1) — текущие накопленные статистические данные о характеристиках пакетов, количестве коллизий и т. п.
- History (2) — статистические данные, сохраненные через определенные промежутки времени для последующего анализа тенденций их изменений.
- Alarms (3) — пороговые значения статистических показателей, при превышении которых агент RMON посыпает сообщение менеджеру.

- Hosts (4) — данные о хостах сети, в том числе и их MAC-адресах.
- Host TopN (5) — таблица наиболее загруженных хостов сети.
- Traffic Matrix (6) — статистика об интенсивности трафика между каждой парой хостов сети, упорядоченная в виде матрицы,
- Filter (7) — условия фильтрации пакетов.
- Packet Capture (8) — условия захвата пакетов.
- Event (9) — условия регистрации и генерации событий.

Данные группы пронумерованы в указанном порядке, поэтому, например, группа Hosts имеет числовое имя 1.3.6.1.2.1.1C.4.

Всего стандарт RMON MIB определяет около 200 объектов в 10 группах, зафиксированных в двух документах -- RFC 1271 для сетей Ethernet и RFC 1513 для сетей Token Ring.

Отличительной чертой стандарта RMON MIB является его независимость от протокола сетевого уровня (и отличие от стандартом MIB-I и MIB-II, ориентированных на протоколы TCP/IP). Поэтому он удобен для гетерогенных сред, использующих различные протоколы сетевого уровня.

Рассмотрим более подробно группу Statistics, которая определяет, какую ин-

формацию о кадрах (называемых в стандарте пакетами) Ethernet может предоставить агент RMON. Группа History основана на объектах группы Statistics , так как се объекты просто позволяют строить временные ряды для объектов группы Statistics.

В группу Statistics входят наряду с некоторыми другими следующие объекты.

- etherStatsDropEvents — общее число событий, при которых пакеты были проигнорированы агентом из-за недостатка его ресурсов. Сами пакеты при этом не обязательно были потеряны интерфейсом.
- etherStatsOctets — общее число байтов (включая ошибочные пакеты), принятых из сети (исключая преамбулу и включая байты контрольной суммы).
- etherStatsPkts — общее число полученных пакетов (включая ошибочные).
- etherStatsBroadcastPkts — общее число хороших пакетов, которые были посланы по широковещательному адресу.
- etherStatsMulticastPkts — общее число хороших пакетов, полученных по мультиплексенному адресу.
- etherStatsCRCAlignErrors — общее число полученных пакетов, которые имели длину (исключая преамбулу) в диапазоне между 64 и 1518 байт, не содержали целое число байтов (alignment error) или имели неверную контрольную сумму (FCS error).
- etherStatsUndersizePkts — общее число пакетов, которые имели длину меньше, чем 64 байт, но были правильно сформированы.
- etherStatsOversizePkts — общее число полученных пакетов, которые имели длину больше, чем 1518 байт, по были тем не менее правильно сформированы.
- etherStatsFragments — общее число полученных пакетов, которые не состояли из целого числа байтов или имели неверную контрольную сумму и имели к тому же длину, меньшую 6-1 байт.
- etherStatsJabbers — общее число полученных пакетов, которые не состояли из целого числа байтов или имели неверную контрольную сумму и имели к тому же длину, большую 1518 байт.
- etherStatsCollisions - наилучшая оценка числа коллизий на данном сегменте Ethernet.
- etherStatsPkts640ctets — общее количество полученных пакетов (включая плохие) размером 64 байт.
- etherStatsPkts65to127Octets — общее количество полученных пакетов (включая плохие) размером от 65 до 127 байт.
- etherStatsPkts128to255Octets — общее количество полученных пакетов (включая плохие) размером от 128 до 255 байт.
- etherStatsPkts256to511Octets — общее количество полученных пакетов (включая плохие) размером от 256 до 511 байт.
- etherStatsPkts512to1023Octets — общее количество полученных пакетов (включая плохие) размером от 512 до 1023 байт.
- etherStatsPkts1024to1518Octets — общее количество полученных пакетов (включая плохие) размером от 1024 до 1518 байт.

Как видно из описания объектов, с помощью агента RMON, встроенного в повторитель или другое коммуникационное устройство, можно провести очень детальный анализ работы сегмента Ethernet или Fast Ethernet. Сначала можно получить данные о встречающихся в сегменте тинах ошибок в кадрах, а затем целесообразно собрать с помощью группы History зависимости, интенсивности этих ошибок от времени (в том числе и привязав их ко времени). После анализа временных зависимостей часто уже можно сделать некоторые предварительные выводы об источнике ошибочных кадров и на этом основании сформулировать более тонкие условия захвата кадров со специфическими признаками (задав условия в группе Filter), соответствующими выдвинутой версии. После этого можно провести еще более детальный анализ путем изучения захваченных кадров, извлекая их из объектов группы Packet Capture.

Позже был принят стандарт RMOK' 2, который распространяет идеи интеллектуальной базы КМОН М1В на протоколы верхних уровней, выполняя часть работы анализаторов протоколов.

Недостатки протокола SNMP

Протокол SNMP служит основой многих систем управления, хотя имеет несколько принципиальных недостатков, которые перечислены ниже.

□ Отсутствие средств взаимной аутентификации агентов и менеджеров. Единственным средством, которое можно было бы отнести к средствам аутентификации, является использование в сообщениях так называемой «строки сообщества» (community string). Эта строка передается по сети в открытой форме в сообщении SNMP и служит основной и для деления агентов и менеджеров на «сообщества», так что агент взаимодействует только с теми менеджерами, которые указывают в поле community string ту же символьную строку, что и строка, хранящаяся в памяти агента. Это, безусловно, не способ аутентификации, а способ структурирования агентов и менеджеров. Версия SNMP v.2 должна была ликвидировать этот недостаток, но в результате разногласий между разработчиками стандарта новые средства аутентификации хотя и появились в этой версии, но как необязательные.

а Работа через ненадежный протокол UDP (а именно так работает подавляющее большинство реализаций агентов SNMP) приводит к потерям аварийных сообщений (сообщений trap) от агентов к менеджерам, что может привести к некачественному управлению. Исправление ситуации путем перехода на надежный транспортный протокол с установлением соединения чревато потерей связи с огромным количеством встроенных агентов SNMP, имеющихся и установленном в сетях оборудования. (Протокол СМIP изначально работает поверх надежного транспорта стека OSI и этим недостатком не страдает.)

Разработчики платформ управления стараются преодолеть эти недостатки. Например, в системе HP OV Telecom DM TMN, являющейся платформой для разработки многоуровневых систем управления в соответствии со стандартами TMN и ISO, работает новая реализация SNMP, организующая надежный обмен сообщениями между агентами и менеджерами за счет самостоятельной органи-

зации повторных передач сообщений SNMP при их потерях.

Выводы

- Существует два популярных семейства стандартов систем управления: стандарты Интернета, описывающие системы управления на основе протокола SNMP, и международные стандарты управления открытых систем (OSI), разработанные ISO и 1TU-T, опирающиеся на протокол управления СМIP. Семейство стандартов Интернета специфицирует минимум аспектов и элементов системы управления, а семейство стандартов TSO/1TU-T — максимум.
- Системы управления SNMP основаны на следующих концепциях, ориентированных на минимальную загрузку управляемых устройств:
 - агент выполняет самые простые функции и работает в основном по инициативе менеджера;
 - система управления состоит из одного менеджера, который периодически опрашивает всех агентов;
 - протокол взаимодействия между агентом и менеджером SNMP опирается на простой ненадежный транспортный протокол UDP (для разгрузки управляемого устройства) и использует два основных типа команд — get для получения данных от агента и set для передачи управляющих воздействий агенту;
 - агент может послать данные менеджеру по своей инициативе с помощью команды trap, но число ситуаций, в которых он применяет эту команду, очень невелико.
- Базы управляющей информации МТБ в стандартах Интернета состоят из дерева атрибутов, называемых объектами и группами объектов.
- Первые базы MIB в стандартах Интернета были ориентированы на управление маршрутизаторами: MIB-I — только контроль, MIB-II — контроль и управление. Более поздняя разработка RMON МШ была направлена на создание интеллектуальных агентов, контролирующих нижний уровень, — интерфейсы Ethernet и Token Ring. Имена объектов стандартных баз MIB для Интернета зарегистрированы в дереве регистрации имен стандартов ISO.

Стандарты управления OSI

Модель сетевого управления OSI (OSI Management Framework) определена в документе ISO/IEC 7498-4: Basic Reference Model, Part 4, Management Framework. Она является развитием общей семиуровневой модели взаимодействия открытых систем для случая, когда одна система управляет другой.

Документ 1SO/IEC 7498-4 состоит из следующих основных разделов.

- Термины и общие концепции.
- Модель управления системами.
- Информационная модель.
- Функциональные области управления системами.
- Структура стандартов управления системами.

Функциональные области управления системами уже были рассмотрены в начале этой части, как имеющие общее значение для любых систем управления. Стандарты ISO в области управления используют терминологию, которая частично совпадает с терминологией систем управления SNMP, а частично от нее отличается.

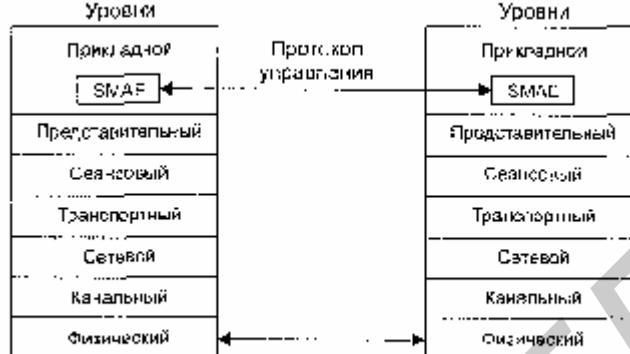


Рис. 20.4. Концепция SMAE

Как показано на рис. 20.4, обмен управляющей информацией с использованием протокола управления (Management Protocol) происходит между субъектами приложений управления системами (Systems Management Application Entities, SMAE). Субъекты SMAE расположены на прикладном уровне семиуровневой модели OSI и являются элементами службы управления. Под субъектом в модели OSI понимается активный в данный момент элемент протокола какого-либо уровня, участвующий во взаимодействии. Примерами SMAE являются агенты и менеджеры систем управления.

Агенты и менеджеры

Определения функций агентов и менеджеров в стандартах OSI достаточно хорошо согласуются с определениями систем SNMP, за некоторыми исключениями в терминологии. Сообщения, которые агент посыпает менеджеру по своей инициативе, называются уведомлениями (notifications).

Например, если некоторый элемент сети X отказал, то менеджеру необходимо обновить свою базу данных конфигурации сети. Элемент X, который является для системы управления управляемым объектом, (managed object), может послать уведомление агенту. Элемент X может находиться в той же управляемой системе, что и агент, или в другой системе. В свою очередь, агент посыпает уведомление менеджеру о том, что элемент X отказал. В соответствии с этим уведомлением менеджер обновляет базу данных конфигурации.

ПРИМЕЧАНИЕ

В стандартах Интернета под объектом понимается отдельный атрибут базы MIB, являющейся моделью управляемого ресурса, а в стандартах ISO объект обозначает всю модель управляемого ресурса.

Менеджер не только собирает и сопоставляет данные, получаемые от агентов,

на «споне» этих данных он может также выполнять административные функции, управляя операциями удаленных агентов.

В стандартах OSI границы между менеджерами и агентами не очень четкие. Субъект SMAE, выполняющий в одном взаимодействии роль менеджера, может в другом взаимодействии выполнять роль агента, и наоборот.

Стандарты OS1 не определяют способов взаимодействия агента с управляемыми объектами. Стандарты OSI также не говорят о том, как агент взаимодействует с управляемыми объектами, находящимися за пределами управляемой системы, то есть объектами, с которыми нужно взаимодействовать через сеть. В таких случаях может потребоваться, например, чтобы один агент запросил данные о некотором объекте от другого агента. Порядок такого рода взаимодействий также не определяется стандартами OSI.

Чтобы менеджер и агент смогли взаимодействовать, каждый должен иметь определенные знания о другом. Эти знания модель OSI называет контекстом, приложения (Application Context, AC). AC описывает элементы прикладного уровня стека OSI, которые используются агентами и менеджерами.

ПРИМЕЧАНИЕ

Необходимо отметить, что стандарты управления OS1 в значительной степени ориентированы па стек протоколов OSI (именно стек, а не модель OST), так же как системы управления SNMP ориентированы на работу со стеком TCP/IP.

Прикладной уровень стека OSI включает несколько вспомогательных служб общего назначения, которые применяются прикладными протоколами и пользовательскими приложениями (в том числе и приложениями управления) для автоматизации наиболее часто выполняемых действий. Это не законченные протоколы прикладного уровня, подобные протоколам ftp, telnet или NCP, с помощью которых пользователь сети может выполнить какое-то полезное действие, а вспомогательные системные функции, помогающие разработчику прикладного протокола или приложения написать его программу компактно и эффективно. На прикладном уровне стека OSI существуют следующие вспомогательные службы.

- ACSE (Association Control Service Element). Отвечает за установление соединений между приложениями различных систем. Соединение (сессия, сеанс) на прикладном уровне OSI носит название ассоциации. Ассоциации бывают индивидуальными и групповыми (shared).
- RTSE (Reliable Transfer Service Element). Обеспечивает восстановление диалога, вызванного разрывом нижележащих коммуникационных служб, в рамках ассоциации.
- ROSE (Remote Operations Service Element). Организует выполнение программных функций на удаленных машинах (аналог службы вызова удаленных процедур RPC).

Протокол CMIP, используемый в стандартах OSI для взаимодействия между менеджерами и агентами, а также программные реализации менеджеров и агентов широко применяют услуги данных вспомогательных служб, в особенности

службы ROSE для вызова удаленных процедур.

Управление системами, управление уровнем и операции уровня

Основная модель управления OSI включает: управление системами, управление N-уровнем и операции N-уровня. Это разбиение на три области сделано для того, чтобы учесть все возможные ситуации, возникающие при управлении.

Управление системами имеет дело с управляемыми объектами на всех семи уровнях OSI, включая прикладной уровень. Оно основано на надежной передаче с установлением соединения управляющей информации между конечными системами. Необходимо подчеркнуть, что модель управления OS1 не разрешает использования служб без установления соединения.

Управление N-уровнем ограничено управляемыми объектами какого-то определенного уровня семиуровневой модели. Протокол управления использует при этом коммуникационные протоколы нижележащих уровней. Управление K-уровнем полезно, когда нет возможности задействовать все семь уровней OS1. В этом случае допускается ограничиться протоколом управления N-уровня, который строго предназначен для данного уровня. Примерами уровневого протокола управления являются протоколы управления для локальных сетей, разработанные институтом IEEE (SMT технологии EDD1), которые ограничены уровнями 1 и 2.

Наконец, операции N-уровня сводятся к мониторингу и управлению на основе управляющей информации, содержащейся в коммуникационных протоколах только данного уровня. Например, данные мониторинга сети, содержащиеся в кадрах STM-n технологии SDH, относятся к операциям N-уровня, а именно физического уровня.

Стандарты на управление N-уровнем и операции N-уровня не входят в набор стандартов управления OSI. Стандарты OSI рассматривают только управление системами с помощью полного семиуровневого стека.

Основная модель управления системами подразумевает выполнение управляющих операций и передачу уведомлений между одноранговыми системами, что означает необязательность жесткого распределения ролей на управляющие и управляемые системы. Эта модель облегчает реализацию распределенных аспектов управления. С другой стороны, допускается реализация одноранговых систем как управляющих и управляемых.

Информационная модель управления

Управляемый объект — это представление OSI о ресурсе в целях управления. Ресурс может быть описан как управляемый объект. Конкретный управляемый объект — это экземпляр (instance) некоторого класса управляемых объектов. Модель управления OSI широко использует объектно-ориентированный подход. Класс управляемых объектов — это набор свойств, которые могут быть обязательными или условными. Путем описания одного класса управляемых

объектов, например коммутаторов, можно создать другой класс управляемых объектов, например коммутаторов, поддерживающих технику VLAN, унаследовав все свойства класса коммутаторов, но добавив новые атрибуты.

Для управления ресурсами менеджер и агент должны быть осведомлены о деталях этих ресурсов. Детализация представления управляемых объектов, которые требуются для выполнения функций управления, хранится в хранилище, известном как Management Information Base (MIB). Базы MIB ОСТ хранят не только описания классов управляемых объектов, но и характеристики сети и ее элементов. Базы MIB содержат характеристики каждой части управляемого оборудования и ресурсов. MIB также включает описание действий, которые могут выполняться на основе собранных данных или же вызываемые внешними командами. Базы MIB позволяют внешним системам опрашивать, изменять, создавать и удалять управляемые объекты (реальные ресурсы сети при этом, естественно, продолжают работать). Протокол СМIP и локальные интерфейсы управления обеспечивают доступ к этим возможностям.

MIB — это концептуальная модель, и она не имеет никакой связи со способом физического или логического хранения данных в ресурсе. Стандарты не описывают аспекты собственно хранения данных.

Протоколы OSI определяют синтаксис информации, хранящейся в MIB, и семантику обмена данными.

Управляющие знания и деревья знаний

Крупная система управления обычно состоит из большого количества агентов и менеджеров. Для автоматического взаимодействия между менеджерами и агентами необходимо каким-то образом задать данные, содержащие характеристики агентов и менеджеров. Менеджеру нужно знать о том, какие агенты работают в системе управления, их имена и сетевые адреса, поддерживаемые ими классы управляемых объектов и т. п. Агенту также требуется аналогичная информация о менеджерах, так как ему нужно отправлять по своей инициативе уведомления и отвечать на запросы менеджеров.

Такие данные называются в модели OSI разделяемыми управляющими знаниями (shared management knowledge) между менеджером и агентом. (В системах SNMP организация этих данных не стандартизована, и в каждой конкретной системе управления эти данные хранятся в индивидуальной форме.)

Разделяемые управляющие знания должны быть известны до установления ассоциации между агентом и менеджером. Обычно они хранятся в каком-либо файле или распределенной базе данных и запрашиваются каждый раз, когда устанавливается ассоциация. Во время установления ассоциации происходит обмен разделяемыми управляющими знаниями.

В OSI стандартизуются различные аспекты организации управляющих знаний и доступа к ним. Следование объектно-ориентированному подходу обусловило использование для хранения этих знаний специальных системных объектов.

Стандарт ISO 101G4-16.2 определяет модель объектов управляющих знаний и

классы таких объектов. Кроме того, определены функции работы с управляющими знаниями.

Имеются три типа управляющих знаний и, соответственно, три типа объектов, которые описывают эти знания.

- Знания репертуара (Repertoire Knowledge) описывают возможности управляемой системы, включающие перечень поддерживаемых классов управляемых объектов, поддерживаемые функции управления и именования. Знания репертуара помогают менеджеру идентифицировать возможности управляемых систем без доступа к ним.
- Знания определений (Definition Knowledge) включают формальные описания классов управляемых объектов, категории тестов, классов взаимосвязей и определения управляющей информации, понимаемой управляемой системой.
- Знания экземпляров (Instance Knowledge) обеспечивают информацию о конкретных экземплярах управляемых объектов, имеющихся в управляемой системе.

Использование древовидных баз данных для хранения управляющих знаний

В системе управления знания о поддерживаемых классах объектов и о порожденных экземплярах объектов должны храниться в какой-либо форме, удобной для предоставления модулям системы управления доступа к этой информации. Архитектура управления OSI предусматривает несколько схем балл данных об управляемых объектах и их классах. Эти схемы обычно называют деревьями из-за иерархической организации информации. Существуют следующие деревья.

- Дерево наследования (Inheritance Tree), называемое также дереном регистрации. Описывает отношения между базовыми и производными классами. Подчиненный класс наследует все характеристики суперкласса и дополняет их специфическими расширениями (дополнительными атрибутами, поведениями и действиями). Классы объектов OSI регистрируются в том же дерене, что и объекты MIB Интернета. Дерево наследования может быть глобальным, то есть начинаться с корня, представляющего весь мир, или локальным, имеющим корень, соответствующий верхнему уровню объектов данной организации или сети. Все управляемые объекты OSI должны быть зарегистрированы в глобальном дереве ISO (в котором зарегистрированы объекты MIB-I, MIB-II, RMON MIB стандарта SNMP). Объекты, представляющие международные стандарты, регистрируются в международной ветви дерева, а частные модели, разработанные производителями систем управления, регистрируются в ветвях дерева, начинающихся с ветви private.
- Дерево включений (Containment Tree). Описывает отношения включения управляемых объектов реальной системы.

ПРИМЕЧАНИЕ

Между деревом наследования и деревом включений нет прямой связи. Напри-

мер, в дереве включений объект «корпоративный концентратор» может включать объекты «интерфейс Ethernet» и «модуль удаленного доступа», которые представляют модели реальных модулей, установленных в слоты корня рационального концентратора. В то же время в дереве наследования класс объемом «интерфейсы Ethernet» подчинен классу объектов «интерфейсы», а класс объектов «модуль удаленного доступа» подчинен классу «коммуникационное оборудование третьего уровня», на основании которого он порожден.

□ Дерево имен (naming tree) определяет способ именования объектов в системе управления. Объекты ОСТ могут иметь имена нескольких типов: относительное отличительное имя (Relative Distinguished Name, RDN), отличительное имя (Distinguished Name, DN), иногда называемое полным отличительным именем (Full Distinguished Name, FDN), и локальное отличительное имя (Local Distinguished Name, LDN). Эти имена связаны с деревом включений, так как определяют имена объектов относительно включающих их объектов. Относительное имя, RDN, соответствует короткому имени, которое однозначно определяет объект среди множества других объектов, подчиненных тому же родительскому объекту. Например, имя interface_a является RDN-именем, уникально характеризующим объект среди объектов, подчиненных объекту node_a. Полное отличительное имя FDN представляет собой последовательность RDM-имен, начинающуюся в вершине глобального дерева имен, то есть дерева, описывающего некоторую глобальную сеть. Наконец, локальное отличительное имя - это последовательность RDM-имен, но начинающаяся не в глобальном корне, а в корне дерева имен локальной системы управления, отвечающей за часть глобального дерева имен данной сети.

Дерево имен обычно совмещается с деревом включений.

Пример дерева включений показан на рис. 20.5. Экземпляр управляемого объекта класса corp-conc (корпоративный концентратор) имеет имя B1, а также атрибут max-slots, описывающий максимальное количество слотов данного класса концентратором, равный в данном случае 14. В этот объект включено ряд других объектов: объекты класса repeater, switch и RA5, которые, в свою очередь, включают объекты типа interface, описывающие порты модулей концентратора.

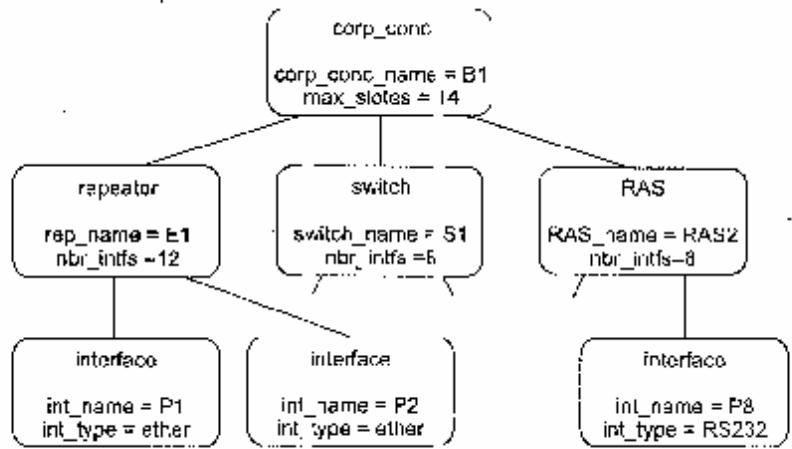


Рис. 20.5. Пример дерева включений.

Имя класса объекта позволяет обратиться к описанию класса и узнать полный список атрибутов этого класса или ссылку на родительский класс, у которого наследуются все или некоторые атрибуты. Имя экземпляра объекта даст информацию о принадлежности конкретного модуля или интерфейса определенному коммуникационному устройству, например имя B1.E1.P2 определяет второй порт модуля повторителя E1, входящего в состав корпоративного концентратора B1.

Правила определения управляемых объектов

Классы управляемых объектов OSI должны определяться в соответствии со стандартом CDMO (Guidelines for the Definition of Managed Objects — правила определения управляемых объектов), являющимся стандартом ISO 10165-4.

В GDMO определяется несколько шаблонов (templates) — пустых форм, которые заполняются для описания определенного класса управляемых объектов. В шаблоне класса перечисляются комплекты свойств (PACKAGES), которые составляют класс. Шаблон комплекта свойств PACKAGE перечисляет атрибуты, группы атрибутов, действия, поведения и уведомления, то есть свойства, сгруппированные для удобства описания класса объектов. Отношения наследования между классами описываются с помощью шаблона связывания имен.

Атрибуты и группы атрибутов определяют параметры объекта, которые можно читать и узнавать из них о состоянии объекта. Свойства 'действия' описывают возможные управляющие воздействия, которые допускается применять к данному объекту — например, мультиплексировать несколько входных потоков в один выходной. Свойство поведение описывает реакцию объекта на примененное к нему действие. Уведомления составляют набор сообщений, которые генерирует объект по своей инициативе.

Заполненные шаблоны GDMO определяют представление класса и его свойств. Заполнение шаблонов выполняется в соответствии с нотацией ASN.1. В отличие от стандартов SNMP, использующих только подмножество типов данных ASN.1, и GDMO и CMIP применяется полная версия ASN.1.

На основании правил GDMO определено несколько международных стандартов на классы управляемых объектов. Документы Definition of Management Information (DMI, ISO/IEC 10165-2:1991) и Generic Management Information (GM1, ISO/IEC CD 10165-5:1992) являются первыми определениями MIB на основе окончательной версии GDMO. Эти определения могут рассматриваться как ISO-эквивалент для Интернета MIB-II, так как они создают основу для построения более специфических баз MIB. Например, DMI определяет класс объектов, называемый Тор, который является верхним суперклассом, — он содержит атрибуты, наследуемые всеми другими классами управляемых объектов. Определены также классы объектов System и Network, занимающие верхние позиции в дереве наследования, так что любой агент должен понимать их атрибуты.

В 1992 году была завершена работа над более специфическими классами объектов — объектами сетевого и транспортного уровней (ISO/ИКС 10737-1 и ISO/IEC 10733).

Сегодня многие организации работают над созданием классов объектов на основе GDMO. Это и международные организации по стандартизации — ISO, ITU-T, ANSI, ETSI, X/Open, и организации, разрабатывающие платформы и инструментальные средства для систем управления, такие как SunSoft, Hewlett-Packard, Vertel, ISR Global. Для телекоммуникационных сетей в рамках архитектуры TMN разработан стандарт M.3100, который описывает ряд специфических для телекоммуникационных сетей классов объектов.

Описания классов управляемых объектов OSI регистрируются как в частных ветвях дерева ISO — ветвях компаний Sun, Hewlett-Packard, IBM и т. п., так и в публичных ветвях, контролируемых ISO или другими международными организациями стандартизации.

В отсутствие одной регистрирующей организации, такой как IETF Internet, использование классов объектов OSI представляет собой непростую задачу.

Протокол CMIP и услуги CMIS

Доступ к управляющей информации, хранящейся в управляемых объектах, обеспечивается с помощью элемента системы управления, называемого службой CMSIE (Common Management Information Service Element). Служба CMSIE построена в архитектуре распределенного приложения, где часть функций выполняет менеджер, а часть — агент. Взаимодействие между менеджером и агентом осуществляется по протоколу CMIP. Услуги, предоставляемые службой CMSIE, называются услугами CMIS (Common Management Information Services).

Протокол CMIP и услуги CMIS определены в стандартах X.710 и X.711 TTU-T. Услуги CMIS разделяются на две группы — услуги, инициируемые менеджером (запросы), и услуги, инициируемые агентом (уведомления).

Услуги, инициируемые менеджером, включают следующие операции:

- M-CREATE инструктирует агента о необходимости создать новый экземпляр

объекта определенного класса или новый атрибут внутри экземпляра объекта;

- M-DELETE инструктирует агента о необходимости удаления некоторого экземпляра объекта определенного класса или атрибута внутри экземпляра объекта;

- M-GET инструктирует агента о возвращении значения некоторого атрибута определенного экземпляра объекта;

- M-SET инструктирует агента об изменении значения некоторого атрибута определенного экземпляра объекта;

- M-ACTION инструктирует агента о необходимости выполнения определенного действия над одним или несколькими экземплярами объектов.

Агент инициирует только одну операцию M-EVENT_REPORT — отправка уведомления менеджеру.

Для реализации своих услуг служба CMISE должна использовать службы прикладного уровня стека OSI — ACSE, ROSE.

Отличие услуг CMIS от аналогичных услуг SNMP состоит в большей гибкости. Если запросы GET и SET протокола SNMP применимы только к одному атрибуту одного объекта, то запросы M-GET, M-SET, M-ACTION и M-DELETE могут применяться к более чем одному объекту. Для этого стандарты CMIP/CMIS вводят такие понятия, как обзор (scoping), фильтрация (filtering) и синхронизация (synchronization).

Обзор

Запрос CMISE может использовать обзор, чтобы опросить одновременно несколько объектов. Вводятся четыре уровня обзора:

- базовый объект, определенный своим отличительным именем FDN;

- объекты, расположенные на n-м уровне подчинения относительно базового (сам базовый объект находится на уровне 0) в дереве включения;

- базовый объект и все объекты, расположенные на подчиненных ему уровнях до n-го (включительно) в дереве включения;

- поддерево — базовый объект и все ему подчиненные в дереве включения.

Фильтрация

Фильтрация заключается в применении булевого выражения к запросу менеджера. Запрос применяется только к тем объектам и их атрибутам, для которых данное булево выражение верно. Булевые выражения могут включать операторы отношения ($=$, \geq , \leq , $>$, $<$) или определенные атрибуты. Возможно построение сложных фильтров на основе объединения нескольких фильтров в один составной.

Синхронизация

При выполнении запросов к нескольким объектам используется одна из двух

схем синхронизации: атомарная или «по возможности». При атомарной схеме запрос выполняется только в том случае, когда все объекты, попадающие в область действия обзора или фильтра, могут успешно выполнить данный запрос. Синхронизация «но возможности» подразумевает передачу запроса всем объектам, к которым запрос относится. Операция завершается при выполнении запроса любым количеством объектов.

Протокол CMIP представляет собой набор операций, прямо соответствующих услугам CMIS. Таким образом, в протоколе CMIP определены операции M-GET, M-SET, M-CREATE и т. д. Для каждой операции определен формат блока данных, переносимых по сети от менеджера агенту, и наоборот.

Формат протокольных блоков данных CMIP описывается нотацией ASN.1 и имеет гораздо более сложную структуру, чем блоки SNMP. Например, блок данных операции M-GET имеет поля для задания имен атрибутов, значения которых запрашивает менеджер, а также поля задания параметров обзора и фильтрации, определяющих множество опрашиваемых экземпляров объектов. Имеются также поля для задания параметров прав доступа к объекту.

Сравнение протоколов SNMP и CMIP

- Применение протокола SNMP позволяет строить как простые, так и сложные системы управления, а применение протокола CMIP определяет некоторый, достаточно высокий начальный уровень сложности системы управления, так как для его работы необходимо реализовать ряд вспомогательных служб, объектов и баз данных объектов.
- Агенты CMIP выполняют, как правило, более сложные функции, чем агенты SNMP. Из-за этого операции, которые менеджеру можно выполнить над агентом SNMP, носят атомарный характер, что приводит к многочисленным обменам между менеджером и агентом.
- Уведомления (traps) агента SNMP посылаются менеджеру без ожидания подтверждения, что может привести к тому, что важные сетевые проблемы останутся незамеченными, так как соответствующее уведомление окажется потерянным, в то время как уведомления агента CMIP всегда передаются с помощью надежного транспортного протокола и в случае потери будут переданы повторно.
- Решение части проблем SNMP может быть достигнуто за счет применения более интеллектуальных баз MIB (к которым относится RMON MIB), но для многих устройств и ситуаций таких МШ нет (или нет стандарта, или нет соответствующей базы MIB в управляемом оборудовании).
- Протокол CMIP рассчитан па интеллектуальных агентов, которые могут выполнить простой команде от менеджера выполнить сложную последовательность действий.
- Протокол CMIP существенно лучше масштабируется, так как воздействовать сразу на несколько объектов, а ответы от агентов проходят через фильтры, которые ограничивают передачу управляющей информации только определен-

ным агентам и менеджерам.

Выводы

- Стандарты ISO/ITU-T для представления управляемых устройств используют объектно-ориентированный подход. Определено несколько суперклассов обобщенных управляемых объектов, на основании которых путем наследования свойств должны создаваться более специфические классы объектов.
- Для описания управляемых объектов OSI разработаны и правила GDMO, основанные на формах определенной структуры, заполняемых с помощью языка ASN.1.
- Для представления знаний об управляемых объектах, агентах и менеджерах системы управления OSI используется три древовидные базы данных: дерево наследования, которое описывает отношения наследования между классами объектов, дерево включения, которое описывает отношения соподчинения между конкретными элементами системы управления, и дерево имен, которое определяет иерархические имена объектов в системе.
- Протокол CMIP, который является протоколом взаимодействия между агентами и менеджерами системы управления OS1, позволяет с помощью одной команды воздействовать сразу на группу агентов, указав параметры обзора и фильтрации.

ГЛАВА 21 Мониторинг и анализ локальных сетей

Постоянный контроль за работой локальной сети, составляющей основу любой компьютерной сети, необходим для поддержания ее в работоспособном состоянии. Контроль — это необходимый первый этап, который должен выполняться при управлении сетью. Ввиду важности этой функции ее часто отделяют от других функций систем управления и реализуют специальными средствами. Такое разделение функций контроля и собственно управления полезно для небольших и средних сетей, для которых установка интегрированной системы управления экономически нецелесообразна. Использование автономных средств контроля помогает администратору сети выявить проблемные участки и устройства сети, а их отключение или реконфигурацию он может выполнять в этом случае вручную.

Процесс контроля работы сети обычно делят на два этапа — мониторинг и анализ.

На этапе мониторинга выполняется более простая процедура — процедура сбора первичных данных о работе сети: статистики о количестве циркулирующих в сети кадров и пакетов различных протоколов, состоянии портов концентраторов, коммутаторов и маршрутизаторов и т. п.

Далее выполняется этап анализа, под которым понимается более сложный и интеллектуальный процесс осмысливания собранной на этапе мониторинга информации, сопоставления ее с данными, полученными ранее, и выработки предложений о возможных причинах замедленной или ненадежной работы сети.

Задачи мониторинга решаются программными и аппаратными измерителями, тестерами, сетевыми анализаторами,строенными средствами мониторинга коммуникационных устройств, а также агентами систем управления. Задача анализа требует более активного участия человека и использования таких сложных средств, как экспертные системы, аккумулирующие практический опыт многих сетевых специалистов.

Средства мониторинга и анализа

Классификация

Все многообразие средств, применяемых для анализа и диагностики вычислительных сетей, можно разделить на несколько крупных классов.

- Агенты систем управления, поддерживающие функции одной из стандартных баз MIB и поставляющие информацию по протоколу SNMP или CM1P. Для получения данных от агентов обычно требуется наличие системы управления, собирающей данные от агентов в автоматическом режиме.
- Встроенные системы диагностики и управления (Embedded systems). Эти системы выполняются в виде программно-аппаратных модулей, устанавливаемых в коммуникационное оборудование, а также в виде программных модулей,

встроенных в операционные системы. Они выполняют функции диагностики и управления только одним устройством, и в этом их основное- отличие от централизованных систем управления. Примером средств этого класса может служить модуль управления многосегментным повторителем Ethernet, реализующий функции автосегментации портов при обнаружении неисправностей, приписывания портов внутренним сегментам повторителя и некоторые другие. Как правило, встроенные модули управления «по совместительству» выполняют роль SNMP-агентов, поставляющих данные о состоянии устройства для систем управления.

□ Анализаторы протоколов (Protocol analyzers). Представляют собой программные или аппаратно-программные системы, которые в отличие от систем управления ограничиваются лишь функциями мониторинга и анализа трафика в сетях. Хороший анализатор протоколов может захватывать и декодировать пакеты большого количества протоколов, применяемых в сетях, — обычно несколько десятков. Анализаторы протоколов позволяют установить некоторые логические условия для захвата отдельных пакетов и выполняют полное декодирование захваченных пакетов, то есть показывают в удобной для специалиста форме вложенность пакетов протоколов разных уровней друг в друга с расшифровкой содержания отдельных полей каждого пакета.

□ Экспертные системы. Этот вид систем аккумулирует знания технических специалистов о выявлении причин аномальной работы сетей и возможных способах приведения сети в работоспособное состояние. Экспертные системы часто реализуются в виде отдельных подсистем различных средств мониторинга и анализа сетей: систем управления сетями, анализаторов протоколов, сетевых анализаторов. Простейшим вариантом экспертной системы является контекстно-зависимая система помощи. Более сложные экспертные системы представляют собой так называемые базы знаний, обладающие элементами искусственного интеллекта. Примерами таких систем являются экспертные системы, встроенные в систему управления Spectrum компании Cabletron и анализатора протоколов Sniffer компании Network General. Работа экспертных систем состоит в анализе большого числа событий для выдачи пользователю краткого диагноза о причине неисправности сети.

□ Оборудование для диагностики и сертификации кабельных систем. Условно это оборудование можно поделить на четыре основные группы: сетевые мониторы, приборы для сертификации кабельных систем, кабельные сканеры и тестеры.

□ Сетевые мониторы (называемые еще сетевыми анализаторами) предназначены для тестирования кабелей различных категорий. Сетевые мониторы собирают также данные о статистических показателях трафика — средней интенсивности общего трафика сети, средней интенсивности потока пакетов с определенным типом ошибки и т. п. Эти устройства являются наиболее интеллектуальными устройствами из всех четырех групп устройств данного класса, так как работают не только на физическом, но и на канальном, а иногда и на сетевом уровнях.

□ Устройства для сертификации кабельных систем выполняют сертификацию в

соответствии с требованиями одного из международных стандартов на кабельные системы.

- Кабельные сканеры используются для диагностики медных кабельных систем.
- Тестеры предназначены для проверки кабелей па отсутствие физического разрыва.
- Многофункциональные портативные устройства анализа и диагностики. В связи с развитием технологий больших интегральных схем появилась возможность производства портативных приборов, которые совмещали бы функции нескольких устройств; кабельных сканеров, сетевых мониторов и анализаторов протоколов.

Анализаторы протоколов

Анализатор протоколов представляет собой либо специализированное устройство, либо персональный компьютер, обычно переносной, класса Notebook, оснащенный специальной сетевой картой и соответствующим программным обеспечением. Применяемые сетевая карта и программное обеспечение должны соответствовать технологии сети (Ethernet, Token Ring, FDDI, Fast Ethernet). Анализатор подключается к сети точно так же, как и обычный узел. Отличие состоит в том, что анализатор может принимать все пакеты данных, передаваемые по сети, в то время как обычная станция — только адресованные ей. Для этого сетевой адаптер анализатора протоколов переводится в режим «беспорядочного» захвата (promiscuous mode).

Программное обеспечение анализатора состоит из ядра, поддерживающего работу сетевого адаптера и программного обеспечения, декодирующего протокол канального уровня, с которым работает сетевой адаптер, а также наиболее распространенные протоколы верхних уровней, например IP, TCP, ftp, telnet, HTTP, IPX, NCP, NetBEUI, DECnet и т. п. В состав некоторых анализаторов может входить также экспертная система, которая позволяет выдавать пользователю рекомендации о том, какие эксперименты следует проводить в данной ситуации, что могут означать те или иные результаты измерений, как устранить некоторые виды неисправности сети.

Анализаторы протоколов имеют некоторые общие свойства.

- Возможность (помимо захвата пакетов) измерения среднестатистических показателей трафика в сегменте локальной сети, в котором установлен сетевой адаптер анализатора. Обычно измеряется коэффициент использования сегмента, матрицы перекрестного трафика узлов, количество хороших и плохих кадров, прошедших через сегмент.
- Возможность работы с несколькими агентами, поставляющими захваченные пакеты из разных сегментов локальной сети. Эти агенты чаще всего взаимодействуют с анализатором протоколов по собственному протоколу прикладного уровня, отличному от SNMP или СМПР.
- Наличие развитого графического интерфейса, позволяющего представить ре-

зультаты декодирования пакетов с разной степенью детализации.

□ **Фильтрация** захватываемых и отображаемых пакетов. Условия фильтрации задаются в зависимости от значения адресов назначения и источника, типа протокола или значения определенных полей пакета. Пакет либо игнорируется, либо записывается в буфер захвата. Использование фильтров значительно ускоряет и упрощает анализ, так как исключает захват или просмотр ненужных в данный момент пакетов.

□ **Использование триггеров.** Триггеры — это задаваемые администратором некоторые условия начала и прекращения процесса захвата данных из сети. Такими условиями могут быть: время суток, продолжительность процесса захвата, появление определенных значений в кадрах данных. Триггеры могут использоваться совместно с фильтрами, позволяя более детально и тонко проводить анализ, а также продуктивнее расходовать ограниченный объем буфера захвата.

□ **Многоканальность.** Некоторые анализаторы протоколов позволяют проводить одновременную запись пакетов от нескольких сетевых адаптеров, что удобно для сопоставления процессов, происходящих в разных сегментах сети. Возможности анализа проблем сети на физическом уровне у анализаторов протоколов минимальные, поскольку всю информацию они получают от стандартных сетевых адаптеров. Поэтому они передают и обобщают информацию физического уровня, которую сообщает им сетевой адаптер, а она во многом зависит от типа сетевого адаптера. Некоторые сетевые адаптеры сообщают более детальные данные об ошибках кадров и интенсивности коллизий в сегменте, а некоторые вообще не передают такую информацию верхним уровням протоколов, на которых работает анализатор протоколов.

С распространением серверов Windows NT все более популярным становится анализатор Network Monitor фирмы Microsoft. Он является частью сервера управления системой SMS, а также входит в стандартную поставку Windows NT Server, начиная с версии 4.0 (версия с усеченными функциями). Network Monitor в версии SMS является многоканальным анализатором протоколов, поскольку может получать данные от нескольких агентов Network Monitor Agent, работающих в среде Windows NT Server, однако в каждый момент времени анализатор может работать только с одним агентом, так что сопоставить данные разных каналов с его помощью не удастся. Network Monitor поддерживает фильтры захвачу (достаточно простые) и дисплейные фильтры (более сложные), отображающие нужные кадры после захвата. Экспертной системой Network Monitor не располагает.

Сетевые анализаторы

Сетевые анализаторы представляют собой эталонные измерительные приборы для диагностики и сертификации кабелей и кабельных систем. Они могут с высокой точностью измерить все электрические параметры кабельных систем, а также работают на более высоких уровнях стека протоколов. Сетевые анализа-

торы генерируют синусоидальные сигналы в широком диапазоне частот, что позволяет измерять на приемной паре амплитудно-частотную характеристику и перекрестные наводки, затухание и суммарное затухание. Сетевой анализатор представляет собой лабораторный прибор больших размеров, достаточно сложный в обращении.

Многие производители дополняют сетевые анализаторы функциями статистического анализа трафика — коэффициента использования сегмента, уровня широковещательного трафика, процента ошибочных кадров, а также функциями анализатора протоколов, которые обеспечивают захват пакетов разных протоколов в соответствии с условиями фильтров и декодирование пакетов.

Кабельные сканеры и тестеры

Основное назначение кабельных сканеров — измерение электрических и механических параметров кабелей: длины кабеля, параметра NEXT, затухания, импеданса, схемы разводки пар проводников, уровня электрических шумов в кабеле. Точность измерений, произведенных этими устройствами, ниже, чем у сетевых анализаторов, но вполне достаточна для оценки соответствия кабеля стандарту.

Для определения местоположения неисправности кабельной системы (обрыва, короткого замыкания, неправильно установленного разъема и т. д.) используется метод «отраженного импульса» (Time Domain Reflectometry, TDR). Суть этого метода состоит в том, что сканер излучает в кабель короткий электрический импульс и измеряет время задержки до прихода отраженного сигнала. По полярности отраженного импульса определяется характер повреждения кабеля (короткое замыкание или обрыв). В правильно установленном и подключенном кабеле отраженный импульс почти отсутствует.

Точность измерения расстояния зависит от того, насколько точно известна скорость распространения электромагнитных волн в кабеле. В различных кабелях она будет разной. Скорость распространения электромагнитных волн в кабеле (Nominal Velocity of Propagation, NVP) обычно задается в процентах от скорости света в вакууме. Современные сканеры содержат в себе электронную таблицу данных о NVP для всех основных типов кабелей, что дает возможность пользователю устанавливать эти параметры самостоятельно после предварительной калибровки.

Кабельные сканеры — это портативные приборы, которые обслуживающий персонал может постоянно носить с собой.

Кабельные тестеры — наиболее простые и дешевые приборы для диагностики кабеля. Они позволяют определить непрерывность кабеля, однако в отличие от кабельных сканеров не дают ответа на вопрос о том, в каком месте произошел сбой.

Многофункциональные портативные приборы мониторинга

В последнее время начали выпускаться многофункциональные портативные приборы, которые объединяют в себе возможности кабельных сканеров, анализаторов протоколов и даже некоторые функции систем управления, сохраняя в то же время такое важное свойство, как портативность. Многофункциональные приборы мониторинга (МТМ) имеют специализированный физический интерфейс, позволяющий выявлять проблемы и тестировать кабели на физическом уровне, который дополняется микропроцессором с программным обеспечением для выполнения высокогоуровневых функций.

Рассмотрим типичный набор функций и свойств такого прибора, который оказывается очень полезным для диагностики причин разнообразных неполадок в сети, происходящих на всех уровнях стека протоколов, от физического до прикладного.

Интерфейс пользователя

Прибор обычно предоставляет пользователю удобный и интуитивно понятный интерфейс, основанный на системе меню. Графический интерфейс пользователя реализован на многострочном жидкокристаллическом дисплее и индикаторах состояния на светодиодах, извещающих пользователя о наиболее общих проблемах наблюдаемых сетей. Имеется обширный файл подсказок оператору с уровневым доступом в соответствии с контекстом. Информация о состоянии сети представляется таким образом, что пользователи любой квалификации могут ее быстро понять.

Функции проверки аппаратуры и кабелей

Многофункциональные приборы сочетают наиболее часто используемые на практике функции кабельных сканеров с рядом новых возможностей тестирования.

Сканирование кабеля

Функция сканирования позволяет измерять длину кабеля, расстояние до самого серьезного дефекта и распределение импеданса по длине кабеля. При проверке неэкранированной витой пары могут быть выявлены следующие ошибки: расщепленная пара, обрывы, короткое замыкание и другие виды нарушения соединения.

Для сетей Ethernet, на коаксиальном кабеле эти проверки могут быть осуществлены на работающей сети.

Определение распределения кабельных жил

Функция определения распределения кабельных жил осуществляет проверку правильности подсоединения жил, наличие промежуточных разрывов и перемычек на витых парах. На дисплей выводится перечень связанных между собой контактных групп.

Определение карты кабелей

Функция определения карты кабелей используется для составления карты основных кабелей и кабелей, ответвляющихся от центрального помещения.

Автоматическая проверка кабеля

В зависимости от конфигурации можно определить длину, импеданс, схему подключения жил, затухание и параметр NEXT на частоте до 100 МГц.

Автоматическая проверка выполняется для:

- коаксиальных кабелей;
- экранированной витой пары с импедансом 150 Ом;
- неэкранированной витой пары с сопротивлением 100 Ом.

Целостность цепи при проверке постоянным током

Функция выяснения целостности цепи при проверке постоянным током обеспечивает проверку коаксиальных кабелей на предмет правильности используемых терминаторов и их установки.

Номинальная скорость распространения

Функция определения номинальной скорости распространения вычисляет номинальную скорость распространения (Nominal Velocity of Propagation, NVP) по кабелю известной длины и дополнительно сохраняет полученные результаты в файле для определяемого пользователем типа кабеля (User Defined cable type) или стандартного кабеля.

Комплексная автоматическая проверка пары «сетевой адаптер-концентратор»

Комплексная автоматическая проверка пары «сетевой адаптер концентратор» позволяет последовательно подключить прибор между конечным узлом сети и концентратором. Тест даст возможность автоматически определить местонахождение источника неисправности — кабель, концентратор, сетевой адаптер или программное обеспечение станции.

Автоматическая проверка сетевых адаптеров

Автоматическая проверка сетевых адаптеров предназначена для выяснения правильности функционирования вновь установленных или «подозрительных» сетевых адаптеров. Для сетей Ethernet по итогам проверки сообщаются: MAC-адрес, уровень напряжения сигналов (а также присутствие и полярность импульсов Link Test для 10BASE-T). Если сигнал не обнаружен на сетевом адаптере, то тест автоматически сканирует соединительный разъем и кабель для их /диагностики.

Функции сбора статистики

Функции сбора статистики позволяют в реальном масштабе времени проследить за изменением наиболее важных параметров, характеризующих «здоровье» сегментов сети. Статистика обычно собирается с разной степенью детализации по разным группам.

Сетевая статистика

В группе сетевой статистики собраны наиболее важные статистические показатели — коэффициент использования сегмента (utilization), уровень коллизий, и уровень ошибок и уровень широковещательного трафика. Превышение этими показателями определенных порогов в первую очередь говорит о проблемах в том сегменте сети, к которому подключен многофункциональный прибор.

Статистика ошибочных кадров

Статистика ошибочных кадров позволяет отслеживать все типы ошибочных кадров для определенной технологии. Например, для технологии Ethernet характерны следующие типы ошибочных кадров.

- Укороченные кадры (Short frames). Это кадры, имеющие длину, меньше допустимой, то есть меньше 6-1 байт. Иногда этот тип кадров дифференцируют на два класса — просто короткие кадры (short), у которых имеется корректная контрольная сумма, и «коротышки» (runts), не имеющие корректной контрольной суммы. Наиболее вероятными причинами появления укороченных кадров являются неисправные сетевые адаптеры и их драйверы.
 - Удлиненные кадры (Jabbers). Это кадры, имеющие длину, превышающую допустимое значение в 1518 байт с хорошей или плохой контрольной суммой. Удлиненные кадры являются следствием затянувшейся передачи, которая появляется из-за неисправностей сетевых адаптеров.
 - Кадры нормальных размеров, но с плохой контрольной суммой (Bad PCS) и кадры с ошибками выравнивания по границе байта. Кадры с неверной контрольной суммой являются следствием множества причин: плохих адаптеров, помех на кабелях, плохих контактов, некорректно работающих портов повторителей, мостов, коммутаторов и маршрутизаторов. Ошибка выравнивания всегда сопровождается ошибкой по контрольной сумме, поэтому некоторые средства анализа трафика не делают между ними различий. Ошибка выравнивания может быть следствием прекращения передачи кадра при распознавании коллизии передающим адаптером.
 - Кадры-призраки (ghosts) являются результатом электромагнитных наводок на кабеле. Они воспринимаются сетевыми адаптерами как кадры, не имеющие нормального признака начала кадра — 10101011. Кадры-призраки имеют длину более 72 байт, в противном случае они классифицируются как удаленные коллизии. Количество обнаруженных кадров-призраков в значительной степени зависит от точки подключения сетевого анализатора. Причинами их возникновения являются петли заземления и другие проблемы с кабельной системой.
- Знание процентного распределения общего количества ошибочных кадров по их типам может многое подсказать администратору о возможных причинах неполадок в сети. Даже небольшой процент ошибочных кадров может привести к

значительному снижению полезной пропускной способности сети, если протоколы, восстанавливающие искаженные кадры, работают с большими тайм-аутами ожидания квитанций. Считается, что в нормально работающей сети процент ошибочных кадров не должен превышать 0,01 %, то есть не более одного ошибочного кадра из 10000.

Статистика по коллизиям

Группа характеристик, относящаяся к статистике по коллизиям, дает информацию о количестве и видах коллизий, отмеченных на сегменте сети, позволяя определить сам факт коллизии и найти источник проблемы. Анализаторы протоколов обычно не могут дать дифференцированной картины распределения общего числа коллизий по их отдельным типам, в то же время знание преобладающего типа коллизий может помочь понять причину плохой работы сети.

Ниже приведены основные типы коллизий сети Ethernet.

- Локальная коллизия (Local Collision). Является результатом одновременной передачи двух или более узлов, принадлежащих к тому сегменту, в котором производятся измерения. Если многофункциональный прибор не генерирует кадры, то в сети на витой паре или волоконно-оптическом кабеле локальные коллизии не фиксируются. Слишком высокий уровень локальных коллизий является следствием проблем с кабельной системой.
- Удаленная коллизия (Remote Collision). Эти коллизии происходят на другой стороне повторителя (по отношению к тому сегменту, в котором установлен измерительный прибор). В сетях, построенных на многопортовых повторителях (10Base-T, 10Base-FL/FB, 100Base-TX/FX/T4, Gigabit Ethernet), все измеряемые коллизии являются удаленными (кроме тех случаев, когда анализатор сам генерирует кадры и может быть виновником коллизии). Не все анализаторы протоколов и средства мониторинга одинаковым образом фиксируют удаленные коллизии. Это происходит из-за того, что некоторые измерительные средства и системы не фиксируют коллизии, происходящие при передаче преамбулы.
- Поздняя коллизия (Late Collision). Это коллизия, которая происходит после передачи первых 64 байт кадра (по протоколу Ethernet коллизия должна обнаруживаться при передаче первых 64 байт кадра). Результатом поздней коллизии будет кадр, который имеет длину более 64 байт и содержит неверное значение контрольной суммы. Чаще всего это указывает на то, что сетевой адаптер, являющийся источником конфликта, оказывается не в состоянии правильно прослушивать линию и поэтому не может вовремя остановить передачу. Другой причиной поздней коллизии является слишком большая длина кабельной системы или слишком большое количество промежуточных повторителей, приводящее к превышению максимального значения времени двойного оборота сигнала.

Средняя интенсивность коллизий в нормально работающей сети должна быть меньше 5 %. Большие всплески (более 20 %) могут быть индикатором кабельных проблем.

Распределение используемых сетевых протоколов

Статистическая группа параметров по распределению используемых сетевых протоколов относится к протоколам сетевого уровня. На дисплее отображается список основных протоколов в убывающем порядке относительно процентного соотношения кадров, содержащих пакеты данного протокола к общему числу кадров в сети.

Основные отправители

Функция определения основных отправителей (Top Senders) позволяет отслеживать наиболее активные передающие узлы локальной сети. Прибор можно настроить на фильтрацию по единственному адресу и выявить список основных отправителей кадров для данной станции. Данные отражаются на дисплее в виде диаграммы вместе с перечнем основных отправителей кадров.

Основные получатели

Функция определения основных получателей (Top Receivers) позволяет следить за наиболее активными узлами-получателями сети. Информация отображается так же, как и для функции основных отправителей кадров.

Основные генераторы широковещательного трафика

Функция определения основных генераторов широковещательного трафика (Top Broadcasters) выявляет станции сети, которые больше остальных генерируют кадры с широковещательными и групповыми адресами.

Генерирование трафика

Прибор генерирования трафика (Traffic Generation) может генерировать трафик для проверки работы сети при повышенной нагрузке. Трафик может генерироваться параллельно с активизированными функциями сетевой статистики, статистики ошибочных кадров и статистики по коллизиям.

Пользователь может задать параметры генерируемого трафика, такие как интенсивность и размер кадров. Для тестирования мостов и маршрутизаторов прибор может автоматически создавать заголовки IP- и IPX-пакетов, и все что требуется от оператора — это внести адреса источника и назначения.

Непосредственно в ходе испытаний пользователь может увеличивать размер и частоту следования кадров с помощью клавиш управления курсором. Это особенно ценно при поиске источника проблем производительности сети и условии возникновения отказов.

Функции анализа протоколов

Обычно портативные многофункциональные приборы поддерживают декодирование и анализ только основных протоколов локальных сетей, таких как протоколы стеков TCP/IP, Novell NetWare, NetBIOS и Banyan VINES.

В некоторых многофункциональных приборах отсутствует возможность декодирования захваченных пакетов, как в анализаторах протоколов, а вместо этого собирается статистика о наиболее важных пакетах, свидетельствующих о наличии проблем в сетях. Например, при анализе протоколов стека TCP/IP собира-

ется статистика по пакетам протокола ICMP с помощью которого маршрутизаторы сообщают конечным узлам о возникновении разного рода ошибок. Для ручной проверки достижимости узлов сети в приборы включается поддержка утилиты IP Ping, а также аналогичных по назначению утилит NetWare Ping и NetBIOS Ping.

Выводы

-
- Мониторинг и анализ сети представляют собой важные этапы контроля за работой сети. Для выполнения этих этапов разработай ряд средств, применяемых автономно и тех случаях, когда применение интегрированной системы управления экономически не оправдано.
 - В состав автономных средств мониторинга и анализа сети входят встроенные средства диагностики, анализаторы протоколов, экспертные системы, сетевые анализаторы, кабельные сканеры и тестеры, многофункциональные приборы.
 - Анализаторы протоколов чаще всего представляют собой специальное программное обеспечение для персональных компьютеров и ноутбуков, которое переходит сетевой адаптер компьютера в режим «беспорядочного» захвата всех кадров. Анализатор протоколов выполняет декодирование захваченных кадров для вложенных пакетов протоколов всех уровней, включая прикладной.
 - Сетевые анализаторы представляют собой прецизионные приборы для сертификации кабельных систем по международным стандартам. Кроме того, эти устройства могут выполнять некоторые функции анализаторов протоколов.
 - Кабельные сканеры являются портативными приборами, которые могут измерить электрические параметры кабелей, а также обнаружить место повреждения кабеля. Кабельные тестеры представляют собой наиболее простые портативные приборы, способные обнаружить неисправность кабеля.
 - Многофункциональные портативные приборы сочетают в себе функции кабельных сканеров и анализаторов протоколов. Они снабжены многострочными дисплеями, контекстной системой помощи, встроенным микропроцессором с программным обеспечением и позволяют выполнять комплексную проверку сегментов сети на всех уровнях, от физического (что не умеют делать анализаторы протоколов), до прикладного. Отличаются от анализаторов протоколов поддержкой только базового набора протоколов локальных сетей.

Мониторинг локальных сетей на основе коммутаторов

Наблюдение за трафиком

Так как перегрузки процессоров портов и других обрабатывающих элементов коммутатора могут приводить к потерям кадров, то функция наблюдения за распределением трафика в сети, построенной на основе коммутаторов, очень важна.

Однако если сам коммутатор не снабжен встроенным агентом SNMP для каждого своего порта, то задача слежения за трафиком, 'традиционно решаемая в сетях с разделяемыми средами путем установки в сеть внешнего анализатора протоколов, очень усложняется.

Обычно в традиционных сетях анализатор протоколов или многофункциональный прибор подключался к свободному порту концентратора, что позволяло ему наблюдать за всем трафиком, передаваемым между любыми узлами сети.

Если же анализатор протокола подключить к свободному порту коммутатора, то он не зафиксирует почти ничего, так как кадры ему передавать никто не будет, а чужие кадры в его порт также направляться не будут. Единственный вид трафика, который зафиксирован анализатором, — это трафик широковещательных пакетов, передаваемых всем узлам сети, а также трафик кадров с неизвестными коммутатору адресами назначения. В случае, когда сеть разделена на виртуальные сети, анализатор протоколов зафиксирован только широковещательный трафик своей виртуальной сети.

Чтобы анализаторами протоколов можно было по-прежнему пользоваться и в коммутируемых сетях, производители коммутаторов снабжают сезон устройства функцией зеркального отображения трафика любого порта на специальный порт. К специальному порту подключается анализатор протоколов, а затем па коммутатор подается команда через его модуль SNMP-управления для отображения трафика какого-либо порта на специальный порт.

Наличие функции организации зеркал для портов частично снимает проблему, но оставляет некоторые вопросы. Например, как просматривать одновременно трафик двух портов или трафик порта, работающего в полнодуплексном режиме.

Более надежным способом слежения за трафиком, проходящим через порты коммутатора, является замена анализатора протокола па агента RMON МГВ для каждого порта коммутатора.

Агент RMON выполняет все функции хорошего анализатора протоколов Ethernet и Token Ring, собирая детальную информацию об интенсивности трафика, различных типах плохих кадров, о потерянных кадрах, причем самостоятельно строя временные ряды для каждого фиксируемого параметра. Кроме того, агент RMON может самостоятельно строить матрицы перекрестного трафика между узлами сети, которые очень нужны для анализа эффективности применения коммутатора.

Так как агент RMON, реализующий все 9 групп объектов Ethernet, стоит весьма дорого, производители для снижения стоимости коммутатора часто реализуют только первые несколько групп объектов RMON MIB. Другим приемом снижения стоимости коммутатора является использование одного агента RMON для нескольких портов. Такой агент по очереди подключается к нужному порту, позволяя снять с него требуемые статистические данные.

Управление виртуальными сетями

Виртуальные локальные септ (VLAN) порождают проблемы для традиционных систем управления па платформе SNMP как при их создании, так и при наблюдении за их работой.

Как правило, для создания виртуальных сетей требуется специальное программное обеспечение компании-производителя, которое работает на платформе системы управления, например HP Open View. Сами платформы систем управления этот процесс поддержать не могут в основном из-за долгого отсутствия стандарта на виртуальные сети. Можно надеяться, что появление стандарта 802.1Q изменит ситуацию в этой области.

Наблюдение за работой виртуальных сетей также создает проблемы для традиционных систем управления. При создании карты сети, включающей виртуальные сети, необходимо отображать как физическую, так и логическую структуру сети, соответствующую связям отдельных узлов виртуальной сети. При этом по желанию администратора система управления должна уметь отображать соответствие логических и физических связей в сети, то есть на одном физическом канале должны отображаться все или отдельные пути виртуальных сетей.

К сожалению, многие системы управления либо вообще не отображают виртуальные сети, либо делают это очень неудобным для пользователя способом, что вынуждает для решения задачи обращаться к менеджерам компаний-производителей.

Выводы

- В локальных сетях на основе коммутаторов мониторинг трафика затруднен логической структуризацией сети, поэтому обычная техника подключения анализатора протоколов к любому физическому сегменту здесь не срабатывает, так как в коммутируемой сети анализатор захватывает только широковещательный трафик и трафик кадров с неизвестными коммутатору адресами.
- Для мониторинга трафика у коммутаторов обычно имеется зеркальный порт для подключения анализатора протоколов. На зеркальный порт можно отобразить трафик любого другого порта коммутатора.
- Более полную картину распределения трафика дает установка агентов RMON в каждом сегменте коммутируемой сети.

Вопросы и упражнения к части VI

1. К какой из пяти стандартных функциональных групп системы управления относится функция концентратора Ethernet по обнулению поля данных в кадрах, поступающих на порты, к которым не подключен узел назначения?
2. К какому уровню модели TMN относится большинство выпускаемых сегодня систем управления?

3. Как объяснить, что наличие в одном сегменте сети NetWare сравнительно небольшого числа (3 %) ошибочных кадров Ethernet резко снижает пропускную способность сети. Рассчитайте коэффициент снижения полезной пропускной способности сети, если при передаче файлов используется метод квитирования с простоями, причем тайм-аут ожидания квитанции составляет 0,5 с, сервер тратит на подготовку очередного кадра данных 20 мкс после получения квитанции от клиентской станции, а клиентская станция отсылает квитанции через 30 мкс после получения очередного кадра данных от сервера. Служебная информация протоколов верхних уровней занимает в кадре Ethernet 58 бант, причем данные передаются в кадрах Ethernet с полем данных максимального размера и 1500 байт, а квитанции помещаются в заголовке протокола прикладного уровня.
4. Какая функция в системах управления системами соответствует функции построения карты сети в системах управления сетями?
5. Какое свойство агента, поддерживающего RMON MIB, послужило поводом назвать данную базу МТВ базой управляющих данных для удаленного мониторинга?
6. Какие действия предпринимает агент SNMP, если его сообщение о сбое управляемого устройства, посланное с помощью команды trap, теряется?
7. Можно ли построить систему управления, работающую без платформы управления?
8. Относится ли средство, называемое community string, к средствам аутентификации?
9. Какую базу данных использует протокол СМПР для воздействия сразу на группу агентов?
10. У вас есть подозрение, что часть коллизий в вашей сети вызвана электромагнитными наводками. Сможет ли анализатор протоколов прояснить ситуацию?