

А.А. Атцук, А.Б. Гольдштейн, Б.С. Гольдштейн



СПРАВОЧНИК
ПО ТЕЛЕКОММУНИКАЦИОННЫМ ПРОТОКОЛАМ

Протокол Медасо/Н.248

А.А. Атцик, А.Б. Гольдштейн, Б.С. Гольдштейн

Серия
«Телекоммуникационные протоколы ЕСЭ РФ»

Протокол МегаСО/Н.248

Справочник

Санкт-Петербург

«БХВ-Петербург»

2014

УДК 621.395
А92
ББК 32.88

А.А. Атцик, А.Б. Гольдштейн, Б.С. Гольдштейн

Протокол Megaco/H.248 : Справочник. – СПб.: БХВ-Петербург, 2014. – 816 с. : ил.

ISBN 978-5-9775-2780-4

«Машины должны работать. Люди должны думать» – гласит девиз компании IBM, достигшей значительных успехов в создании первых, в подборе вторых, а также в протоколах взаимодействия между первыми и вторыми. Для сетей NGN существует такое же разделение обязанностей между медиа-шлюзами MG, которые работают, и контроллерами медиа-шлюзов MGC, которые думают. А для взаимодействия между ними разработан рассматриваемый в этой книге протокол управления шлюзами Megaco/H.248, обеспечивающий расширенную поддержку мультимедийной связи и конференций и использующий удобный синтаксис для обработки сообщений, альтернативные транспортные технологии, разнообразные методы кодирования сообщений, эффективные способы расширения своих функциональных возможностей. Подробно рассматривается наиболее распространенная сегодня версия 2 протокола H.248, однако значительное внимание уделяется и перспективной версии 3 этого протокола.

Справочник

ISBN 978-5-9775-2780-4

© А.А. Атцик, А.Б. Гольдштейн, Б.С. Гольдштейн, 2009, 2014

Содержание

Предисловие	7
Глава 1. Принцип декомпозиции шлюзов	10
1.1. Архитектура распределенного шлюза	10
1.2. Классификация медиа-шлюзов	14
1.3. H.248 в современных сетях	15
Глава 2. Основные понятия	17
2.1. Модель организации связи	17
2.2. Окончания (Terminations)	17
2.3. Контекст (Context)	20
2.4. Использование групповых символов	22
2.5. Мультиплексирование потоков	24
Глава 3. Дескрипторы протокола H.248	27
3.1. Понятие дескриптора	27
3.2. Дескриптор модема (Modem Descriptor)	29
3.3. Дескриптор мультиплексора (Multiplex Descriptor)	32
3.4. Медиа-дескриптор (Media Descriptor)	33
3.5. Дескриптор состояния порта (TerminationState Descriptor)	34
3.6. Дескриптор потока (Stream Descriptor)	34
3.7. Дескриптор местного управления (LocalControl Descriptor)	35
3.8. Местный и удаленный дескрипторы (Local & Remote Descriptors)	37
3.9. Дескриптор Событий (Events Descriptor)	41
3.10. Дескриптор буфера Событий (EventsBuffer Descriptor)	44
3.11. Дескриптор Сигналов (Signals Descriptor)	44
3.12. Дескриптор проверки (Audit Descriptor)	47
3.13. Дескриптор смены режима обслуживания (Service Change Descriptor)	48
3.14. Дескриптор плана нумерации (DigitMap Descriptor)	53
3.15. Дескриптор Статистики (Statistics Descriptor)	59
3.16. Дескриптор Пакета (Package Descriptor)	60
3.17. Дескриптор обнаруженных Событий (ObservedEvents Descriptor)	60
3.18. Дескриптор топологии (Topology Descriptor)	60
3.19. Дескриптор ошибки (Error Descriptor)	61

Глава 4. Команды и сообщения	63
4.1. Команды	63
4.2. Управление атрибутами контекста и их проверка	70
4.3. Транзакции	71
4.4. Сообщения	75
Глава 5. Пакеты (Packages)	76
5.1. Концепция Пакетов в H.248	76
5.2. Пакеты базовой версии протокола	79
5.3. Пакеты, описанные в спецификации H.248.2	107
5.4. Пакеты, описанные в спецификации H.248.3	141
5.5. Пакеты, описанные в спецификации H.248.6	154
5.6. Пакеты, описанные в спецификации H.248.7	156
5.7. Пакеты, описанные в спецификации H.248.9	161
5.8. Пакеты, описанные в спецификации H.248.10	199
5.9. Пакеты, описанные в спецификации H.248.11	200
5.10. Пакеты, описанные в спецификации H.248.12	209
5.11. Пакеты, описанные в спецификации H.248.13	223
5.12. Пакеты, описанные в спецификации H.248.14	225
5.13. Пакеты, описанные в спецификации H.248.16	226
5.14. Пакеты, описанные в спецификации H.248.17	236
5.15. Пакеты, описанные в спецификации H.248.18	248
5.16. Пакеты, описанные в спецификации H.248.19	250
5.17. Пакеты, описанные в спецификации H.248.21	270
5.18. Пакеты, описанные в спецификации H.248.22	272
5.19. Пакеты, описанные в спецификации H.248.23	274
5.20. Пакеты, описанные в спецификации H.248.24	281
5.21. Пакеты, описанные в спецификации H.248.25	285
5.22. Пакеты, описанные в спецификации H.248.26	297
5.23. Пакеты, описанные в спецификации H.248.27	311
5.24. Пакеты, описанные в спецификации H.248.28	319
5.25. Пакеты, описанные в спецификации H.248.29	326
5.26. Пакеты, описанные в спецификации H.248.30	350
5.27. Пакеты, описанные в спецификации H.248.31	355
5.28. Пакеты, описанные в спецификации H.248.32	360

5.29. Пакеты, описанные в спецификации H.248.33	365
5.30. Пакеты, описанные в спецификации H.248.34	368
5.31. Пакеты, описанные в спецификации H.248.35	386
5.32. Коды ошибок.....	388
5.33. Обобщенная таблица Пакетов.....	398
Глава 6. Взаимодействие MG и MGC	412
6.1. Интерфейс управления MG-MGC	412
6.2. Защита соединений	417
Глава 7. Кодирование сообщений протокола H.248	421
7.1. Кодеры/декодеры сообщений H.248	421
7.2. Двоичное кодирование	422
7.3. Текстовое кодирование.....	477
7.4. Кодирование дескрипторов Local и Remote с мультиплексированием H.221 и H.223.....	518
Глава 8. Транспортные технологии	524
8.1. Транспортировка команд	525
8.2. Транспортировка по IP	527
8.3. Протокол управления шлюзом. Транспортировка посредством ATM.....	535
Глава 9. Сценарии соединений	539
9.1. Соединение между двумя резидентными шлюзами	539
9.2. Соединение между пользователем, подключенным к резидентному шлюзу (Residential Gateway), и транкинговым шлюзом (Trunking Gateway)	557
9.3. Соединение между Trunking gateway и Residential Gateway.....	566
9.4. Соединение между двумя транкинговыми шлюзами	575
9.5. Соединение между RGW и трактом ОКС7, подключенным к TGW	582
9.6. Соединение между трактом ОКС7, подключенным к TGW, и RGW	591
9.7. Соединение между ISDN-трактом, подключенным к TGW, и трактом ОКС7, подключенным к TGW	600
9.8. Тест на целостность соединения со стороны TGW.....	605
9.9. Соединение между RGW и терминалом, использующим протокол H.323.....	609
9.10. Соединение между пользователем, подключенным к RGW, и пользователем SIP	616
9.11. Использование команды Service Change	624
9.12. Использование команды Audit	642

Глава 10. Профили	649
10.1. Механизм профилей	649
10.2. Шаблон профиля	650
10.3. Профиль для Trunking GateWay (транкингового шлюза)	664
10.4. Профиль. Border GateWay (граничный шлюз)	687
10.5. Профиль. Access and Residential GateWays (шлюзы доступа и резидентные шлюзы)	710
10.6. Профиль. Multimedia Resource Function Processors	744
Глава 11. IP-телефоны MEGACO	777
11.1. Архитектура MEGACO IP-Phone	777
11.2. Типы окончаний	781
11.3. Профиль H.248 IPPhone Protocol Profile	783
Глава 12. Реализация, эксплуатация и тестирование	785
12.1. Варианты реализации H.248 в MGC	785
12.2. Шлюзы доступа iMAK	787
12.3. Эксплуатационное управление сетью доступа	791
Заключение	799
Список сокращений	802
Список литературы	810

Предисловие

«Машины должны работать, люди должны думать» – гласит девиз компании IBM, достигшей грандиозных успехов как в создании первых, так и в подборе вторых, а также в разработке протоколов взаимодействия между первыми и вторыми. Для сетей связи следующего поколения *NGN (Next Generation Network)* такое же разделение обязанностей существует между медиа-шлюзами, которые работают, и контроллерами медиа-шлюзов *MGC (Media Gateway Controller)*, которые думают. Именно в этом заключается суть принципа декомпозиции шлюза, чему посвящена первая глава книги. А для взаимодействия медиа-шлюзов *MG (Media Gateway)* и контроллеров медиа-шлюзов *MGC* разработаны два наиболее популярных сегодня протокола управления шлюзами: *MGCP (Media Gateway Control Protocol)* и *Megaco/H.248 (Media Gateway Control/H.248)*.

Первые пилотные версии *MGCP* появились в комитете инженерных задач Интернет *IETF (Internet Engineering Task Force)* еще в 1999 году, сразу после образования рабочей группы управления медиа-шлюзами *MEGACO (Media Gateway Control group)*. Протокол *MGCP (RFC 2705)* быстро стал популярным, и даже после первой публикации *RFC 2885* с описанием более перспективного протокола *Megaco/H.248* его продолжают широко использовать многие Операторы, руководствующиеся доводом «зачем что-то заменять, если оно хорошо работает». К тому же, впоследствии обнаружилось, что в протоколе *Megaco/H.248* есть некоторые ошибки, и был подготовлен отдельный проект *RFC 2886*, перечисляющий опечатки и предлагающий соответствующие исправления. Затем был доработан исходный *RFC*, и его опубликовали как *RFC 3015*, куда были внесены необходимые исправления ошибок в *RFC 2885*. Таким образом, теперь официальной версией 1 *Megaco* является *RFC 3015*. У этого нового протокола, созданного совместными усилиями *IETF* и Исследовательской комиссией 16 сектора стандартизации телекоммуникаций Международного союза электросвязи *ITU-T (International Telecommunications Union Telecommunication Standardization Sector)*, имеется ряд существенных преимуществ перед *MGCP*: расширенная поддержка мультимедиа и конференций, более удобный

синтаксис для обработки сообщений, альтернативные транспортные технологии, разнообразные варианты кодирования сообщений, формализованные и эффективные методы расширения функциональных возможностей протокола и др. Поэтому переход от MGCP к Megaco/H.248 – это только вопрос времени, и авторы, отдавая должное процессу эволюционного развития телекоммуникационных сетей, сознательно свели к минимуму упоминания о MGCP, и, по сути, никак не описывают в справочнике этот протокол, за исключением некоторых сравнений его с Megaco/H.248.

Что же касается протокола Megaco/H.248, то ему посвящена практически вся книга, и в силу частого упоминания вместо полного его названия мы будем оставлять лишь часть, полученную от ITU-T. Тому есть причина: IETF стояла у истоков как MGCP, так и Megaco, разработала требования к этим протоколам, а затем и сами протоколы, вела активное сотрудничество с ITU-T в области создания расширений и усовершенствования протокола Megaco, но в марте 2006 года последний председатель рабочей группы MEGACO официально объявил о ее закрытии, отдав, таким образом, ITU-T всю работу по дальнейшей стандартизации этого протокола. На момент завершения справочника последней версией протокола H.248 является 3-я версия, однако наиболее распространенной остается версия 2. В основном, материал справочника базируется именно на второй версии, хотя для удобства пользователей в него включена также существенная информация и о 3-й версии. В качестве одного из преимуществ протокола называлась хорошая расширяемость, которая привела к тому, что количество дополнений к протоколу постоянно увеличивается, поэтому авторам пришлось искусственно ограничить себя рамками рекомендации H.248.35 и не пытаться объять необъятное. Разумеется, по мере становления технологии, разработки, которые найдут свое отражение в новых номерах H.248.x, будут учтены в будущих изданиях настоящего справочника.

Первая глава познакомит читателя с принципом декомпозиции шлюзов, осветит историю возникновения H.248 и покажет его место в современных сетях. Во второй главе рассматриваются основные понятия и абстрактные принципы работы протокола H.248. Третья и четвертая главы являются основной информационной частью книги, в которой раскрывается принцип построения сообщений протокола H.248. Самая емкая глава 5 содержит описание информационного наполнения структуры сообщений, описанной в предыдущей главе. Шестая глава посвящена отдельным процедурам управления взаимодействием MG и MGC. Седьмая глава описывает два способа кодирования сообщений H.248 и содержит общие сведения о языках текстового и двоичного кодирования, необходимые для того, чтобы разобраться в приведенных там же спецификациях сообщений протокола H.248, представленных тем и другим способом. Восьмая глава описывает специфические процедуры протокола H.248, необходимые при работе

с разными транспортными технологиями. Девятая глава содержит ряд примеров обмена сообщениями при обслуживании базового вызова, а также при регистрации и при других сервисных процедурах; в реальных сетях обмен сообщениями может изменяться, если при этом он не противоречит логике обмена сообщениями H.248. В десятой главе описывается механизм профилей, служащих для определения конфигурации медиа-шлюза в том или ином сетевом окружении. Одиннадцатая глава описывает устройство и функционирование IP-телефона MEGACO, а также требования к нему. Двенадцатая глава посвящена практическим реализациям протокола H.248, методам и средствам его тестирования. В Заключении оцениваются перспективы развития H.248.

Как и многие другие новые технологии, протокол H.248 еще не имеет устоявшейся русскоязычной терминологии, некоторые определения в книге могут оказаться не совсем привычными, в ряде случаев термины приводятся сразу на русском и английском языках. Особое внимание следует обратить на названия таких элементов протокола, как *Пакеты (Packages)*, *Статистики (Statistics)*, *События (Events)*, *Свойства (Properties)* и *Сигналы (Signals)*, которые в тексте приводятся с прописной буквы. Это сделано для того, чтобы отличать их от общепринятых значений этих слов, например, от обычных пакетов (packet). Изложенные в книге материалы базируются на международных стандартах и документах, подготовленных рабочими группами ITU-T, IETF и ETSI (European Telecommunications Standards Institute), технических публикациях отечественных и зарубежных авторов, собственных разработках, исследовательских проектах и работах по тестированию и сертификации оборудования NGN, выполненных под руководством и/или при непосредственном участии авторов.

Мы постарались сделать наш справочник максимально полным, тем не менее, описания некоторых процедур, не актуальных для отечественных сетей связи, были опущены или сокращены, чтобы не перегружать справочник информацией, мало востребованной специалистами.

Авторы пользуются случаем выразить благодарность за помощь в подготовке этой книги специалистам Научно-технических центров ПРОТЕЙ, АРГУС, СЕВЕНТЕСТ и всей Группе компании ЭКРАН, сотрудникам сертификационных центров ЛОНИИС и СПбГУТ им. проф. М.А. Бонч-Бруевича, коллегам из отдела системных исследований НИИТС, студентам и аспирантам кафедры SKиРИ СПбГУТ, в первую очередь, Евгению Зорину, Ольге Ващиловой и их сокурсникам, а также всем, кто помогал, кто снимал вместе с нами трейсы на испытаниях оборудования NGN основных мировых производителей, кто просто морально поддерживал в этой нелегкой работе со столь многообразным и бурно развивающимся протоколом.

Глава 1. Принцип декомпозиции шлюзов

1.1. Архитектура распределенного шлюза

Впервые принцип декомпозиции шлюзов был предложен в проекте TIPHON института ETSI в качестве некоторой сетевой архитектуры, предусматривающей разбиение множества функций шлюза IP-телефонии на структурные элементы. Тогда проект решал ряд задач интеграции различных сетей и совершенствования технологии IP-телефонии таким образом, чтобы она могла обеспечить приемлемое качество услуг при взаимодействии с телефонными сетями общего пользования (ТФОП). Проект *TIPHON (Telecommunications and Internet Protocol Harmonization Over Networks)* был организован в апреле 1997 года, и в этой работе с самого начала участвовало несколько десятков ведущих телекоммуникационных компаний. Первоначально в этом проекте распределенный шлюз входил в состав сети, построенной на базе рекомендации H.323, и взаимодействовал с устройством, называемым *привратником* или *гейткипером GK (GateKeeper)*. В дальнейшем принцип декомпозиции шлюзов получил широкое распространение уже за пределами архитектуры H.323.

Принцип декомпозиции шлюза делит традиционный для сетей H.323 шлюз IP-телефонии на следующие функциональные блоки:

- медиа – (или транспортный) шлюз – *Media Gateway (MG)*, который выполняет функции преобразования речевой информации, поступающей со стороны ТФОП, в вид, пригодный для передачи по сетям с маршрутизацией IP-пакетов – кодирование и упаковку речевой информации в пакеты

RTP/UDP/IP (Real Time Protocol/User Datagram Protocol/Protocol IP), а также обратное преобразование;

- устройство управления шлюзом – *Media Gateway Controller (MGC)*, управляющее работой шлюза и входящее в качестве основной составляющей в гибкий коммутатор Softswitch;
- шлюз сигнализации – *Signaling Gateway (SG)*, который обеспечивает доставку сигнальной информации, поступающей со стороны ТФОП, к устройству управления шлюзом MGC с помощью протокола SIGTRAN и перенос сигнальной информации в обратном направлении.

Архитектуру сети, основанную на принципе декомпозиции шлюза, иллюстрирует рис. 1.1.

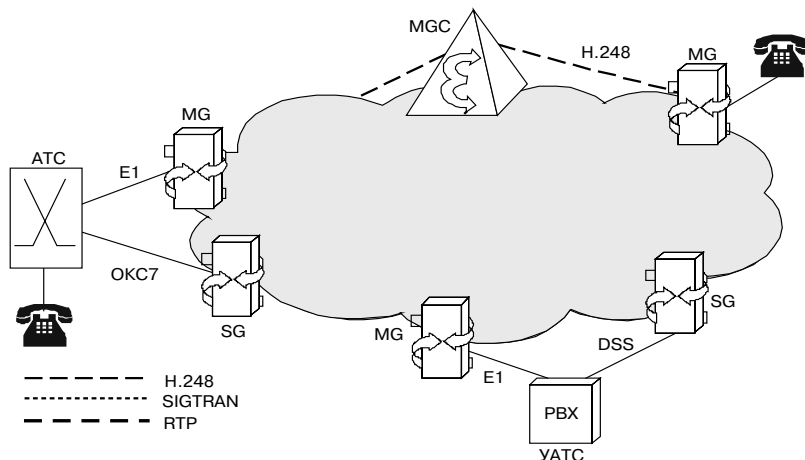


Рис. 1.1. Архитектура сети, базирующаяся на принципе декомпозиции шлюза

Таким образом, весь интеллект функционально распределенного шлюза размещается в устройстве управления MGC. Шлюз сигнализации SG выполняет также функции *STP (Signal Transfer Point)* – транзитного пункта системы сигнализации по общему каналу ОКС7 и/или функции конвертера других систем сигнализации. В случае подключения распределенного шлюза к ТФОП с применением сигнализации по выделенным сигнальным каналам (ВСК), сигнальная информация вместе с пользовательской информацией сначала поступает в медиа-шлюз MG,

а затем передается оттуда в устройство управления без посредничества шлюза сигнализации. Функции преобразования речевой информации выполняются медиа-шлюзами MG. Одно устройство управления MGC обслуживает одновременно несколько шлюзов. В сети может присутствовать несколько устройств управления. Предполагается, что эти устройства синхронизованы между собой и согласованно управляют шлюзами, участвующими в соединении. Рабочая группа MEGACO (MEdia GAteway COntrol) не определяет протокол синхронизации устройств управления, однако в ряде работ, посвященных исследованию возможностей протоколов управления медиа-шлюзами, для этой цели предусматривается использовать протоколы SIP, H.323 или ISUP/IP (см., например, рис. 1.2).

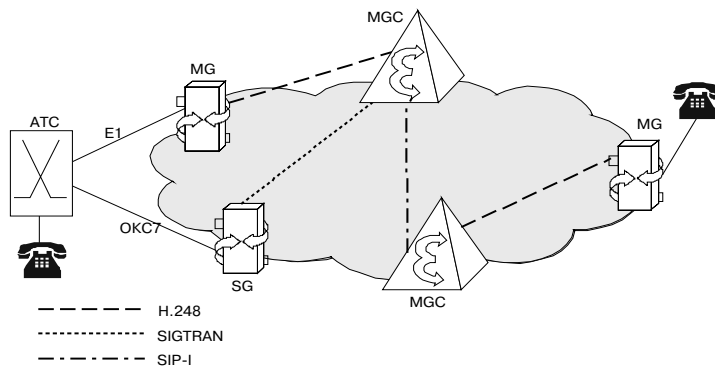


Рис. 1.2. Синхронизация работы устройств управления

Главное, что обеспечивает декомпозиция шлюза, – это полное логическое и физическое разделение функций управления обслуживанием вызова и функций поддержания соединения (коммутации), размещенных, соответственно, в MGC и в медиа-шлюзе. Следовательно, важнейшее значение в данной архитектуре приобретает организация эффективного взаимодействия этих компонентов, для чего и были разработаны так называемые протоколы управления медиа-шлюзами.

Первым протоколом, относящимся к этому семейству и получившим широкое распространение, стал протокол управления шлюзами – *MGCP (Media Gateway Control Protocol)*, разработанный рабочей группой IETF. Как показано на рис. 1.3, у истоков создания MGCP лежали такие алгоритмы, как DSM-CC и DIAMETER, развитие которых привело к появлению протокола *IPDC (IP Device Control)*, разработанного компанией Level3 и ориентированного в большей степени на управ-

ление IP-устройствами, чем на обслуживание речевых вызовов. Одновременно компания Telcordia разработала схожий с ним протокол *SGCP (Simple Gateway Control Protocol)*, лучше приспособленный к решению задач управления обслуживанием телефонного вызова. Позднее SGCP получил поддержку и компании Cisco. После принятия решения об объединении этих двух родственных протоколов и была создана рабочая группа в рамках IETF, разработавшая впоследствии протокол MGCP.

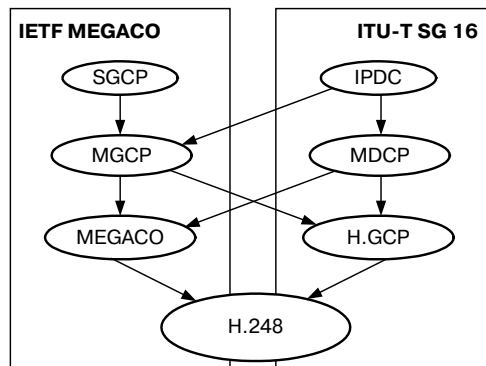


Рис. 1.3. Дерево эволюции протоколов управления шлюзами

Протокол MGCP имел ряд ограничений, в частности, он был очень привязан к протоколу IP и к управлению обслуживанием телефонных вызовов, но время IP-телефонии в ее первоначальном упрощенном виде уже подходило к концу. Объединенные усилия рабочей группы MEGACO комитета IETF и 16-й Исследовательской комиссии ITU-T привели к радикальной переработке MGCP, в результате которой и был создан протокол H.248. Этот новый протокол поддерживает мультимедийные потоки данных и лучше подходит для взаимодействия со шлюзами разных типов (IP-TDM, ATM-TDM, TDM-TDM и др.). Не следует полагать, что указанные протоколы могут служить только объединению различных сетей на базе IP-телефонии; их модели достаточно универсальны, и пользователь может воспользоваться услугами IP-телефонии на базе H.248, используя резидентный шлюз, который обеспечит ему доступ, как к речевым услугам, так и к услугам передачи данных. Отметим, что протоколы управления шлюзами являются внутренними протоколами, поддерживающими обмен информацией между функциональными блоками распределенного шлюза. Протоколы используют принцип *master/slave (ведущий/ведомый)*, причем устройство управления шлюзами является ведущим, а шлюз – ведомым устройством.

вом, выполняющим команды, которые поступают от устройства управления. Такое решение обеспечивает полную детерминированность обмена сообщениями протокола, масштабируемость сети и простоту эксплуатационного управления ею через устройство управления шлюзами. К тому же, шлюзы не должны иметь интеллекта, требуют меньшей производительности процессоров и, как следствие, оказываются менее дорогими. Кроме того, обеспечивается возможность быстро добавлять новые протоколы сигнализации и новые дополнительные услуги, так как нужные для этого изменения затрагивают только устройство управления шлюзами, а не сами шлюзы.

1.2. Классификация медиа-шлюзов

Разрабатывая протоколы и модели управления шлюзами, IETF попутно создал классификацию медиа-шлюзов:

- *Trunking gateway (TG)* – шлюз между ТфОП и сетью с маршрутизацией IP-пакетов, ориентированный на подключение к телефонной сети большим количеством цифровых каналов (от десятков до нескольких тысяч). Работает в паре с сигнальным шлюзом SG, транслирующим сигнальную информацию OKC7;
- *Voice over ATM gateway (VoATM GW)* – шлюз между ТфОП и ATM-сетью, который также подключается к телефонной сети большим количеством цифровых каналов и представляет собой аналог Trunking Gateway;
- *Residential gateway (RG)* – шлюз, подключающий к IP-сети аналоговые устройства: кабельные модемы, линии xDSL и широкополосные устройства беспроводного доступа. Обычно рассчитан на небольшое (1-2) количество портов;
- *Access gateway (AG)* – шлюз доступа для подключения к сети IP-коммуникаций небольшой учрежденческой АТС через аналоговый или цифровой интерфейс, или для прямого подключения аналоговых абонентских линий, как это имеет место в *мультисервисных концентраторах доступа (iMAK)*, рассматриваемых в главе 12;
- *Network Access Server (NAS)* – сервер доступа к IP-сети для передачи данных;
- *Business gateway (BG)* – шлюз с цифровым интерфейсом для подключения к сети с маршрутизацией IP-пакетов учрежденческой АТС с использованием, например, системы сигнализации DSS1.

1.3. H.248 в современных сетях

Происходящий в настоящее время процесс конвергенции сетей связи привел к растущей популярности решений на базе распределенного шлюза, поскольку конвергентная сеть представляет собой технологическую мозаику, фрагменты которой как раз и соединяются между собой при помощи шлюзов, управляемых по протоколу H.248. Одной из важных областей применения протокола H.248 являются сети IP-телефонии, построенные согласно последним версиям рекомендации H.323 и с учетом предложения группы TIPHON. Распределенный шлюз в такой сети воспринимается другими элементами сети H.323 как единое устройство. Протоколы MGCP и H.248 также могут быть использованы для построения сети IP-коммуникаций с использованием принципов и сетевой концепции, отличных от разработанных в проекте TIPHON. Для этого достаточно, чтобы контроллер медиа-шлюзов был совмещен с модулем, именуемым *Call Agent* и содержащим логику обслуживания вызова и предоставления базовых услуг телефонной связи. Такой подход может быть удобен ряду Операторов, специализирующихся на транзите трафика. Широкое внедрение разных вариантов сетевой архитектуры на базе распределенного шлюза началось после того, как на смену моделям, прямо следующим спецификациям международных стандартизирующих организаций, пришел новый тип оборудования, получивший название *Softswitch*. Именно *Softswitch* породил варианты распределенной сетевой архитектуры, эффективные для работы в реальных сетях Операторов. Централизованные управляющие узлы такой сетевой архитектуры, помимо описанных в стандартах функций управления шлюзами и преобразования номеров, дополнялись функциями реализации логики обслуживания вызова, преобразования протоколов сигнализации, начисления платы, технической эксплуатации, и многими другими. При этом отличительной чертой всех *Softswitch*-решений является взаимодействие сетей ТфОП и VoIP через распределенный шлюз, функции MGC которого располагаются как раз в центральном управляющем устройстве *Softswitch*.

Из приведенной выше классификации медиа-шлюзов можно видеть, что практически все узлы сети NGN могут использовать протокол H.248 для установления и разрушения соединений. Таким образом у Операторов имеется выбор, какой протокол IP-телефонии или какую их комбинацию использовать для построения своей мультисервисной сети SIP, H.248, SIGTRAN или H.323, и, соответственно, какими справочниками этой серии пользоваться.

В данном случае для облегчения этого выбора можно отметить некоторые характеристики протокола H.248:

- архитектура master/slave, которая жестко определяет порядок обмена сообщениями, и позволяет избежать неопределенности в некоторых ситуациях, характерных для протокола SIP, например, при встречных посылках сообщений INVITE;
- применение как текстового, так и двоичного кодирования;
- формирование акустических сигналов медиа-шлюзом по команде MGC.

В процессе стандартизации сетей связи 3G организацией *3GPP (3rd Generation Partnership Project)* была создана архитектура сети управления мультимедийными услугами *IMS (IP Multimedia Subsystem)*, которая сегодня считается наиболее перспективной архитектурой построения сетей NGN как для стационарных систем, так и для систем мобильной связи, а также конвергентных решений проводной и беспроводной связи *FMC (Fixed-Mobile Convergence)*.

Подсистема IMS определена как набор стандартных функциональных модулей, объединенных открытыми интерфейсами. Для некоторых интерфейсов специфицировано использование протокола H.248. При развитии концепции IMS в организованном ETSI проекте *TISPAN (Telecoms & Internet converged Services & Protocols for Advanced Networks)* была разработана полноценная NGN-архитектура, в которой была усилена поддержка проводного доступа и увеличилось число модулей, взаимодействующих по протоколу H.248. Недавно появившийся тип оборудования – пограничные контроллеры сессий *SBC (Session Border Controller)*, выполняющие ряд задач на границах IP-сетей. SBC – это еще одна точка контроля и управления трафиком, расположенная на границе пакетных сетей. При этом функции SBC можно разделить на функции обработки сигнализации и функции обработки пользовательского трафика. Такое разделение позволяет строить распределенные SBC, состоящие из центрального узла *SBC-SIG*, обрабатывающего сигнализацию, и периферийных устройств *SBC-MEDIA*, обрабатывающих пользовательский трафик. Управление транспортными потоками, проходящими через SBC-MEDIA, может выполняться центральным SBC или узлом Softswitch с помощью протокола H.248. Таким образом, видно, что протокол H.248 используется в широком спектре решений для современных телекоммуникационных сетей.

Глава 2. Основные понятия

2.1. Модель организации связи

Для описания алгоритма установления соединения протокол H.248 использует специальную модель процесса обслуживания вызова, согласно которой протокол оперирует двумя логическими объектами внутри медиа-шлюза – *окончанием (termination)* и *контекстом (context)*, – управляемыми контроллером шлюза MGC.

2.2. Окончания (Terminations)

Окончания (в русскоязычной литературе часто встречается также термин *порты*) являются логическими объектами медиа-шлюза, выполняющими функции источников и приемников медиа-информации. Можно выделить два вида окончаний в зависимости от того, какой интерфейс они представляют, – физический или виртуальный.

Физические окончания, существующие постоянно с момента конфигурации шлюза, это аналоговые телефонные интерфейсы оборудования, поддерживающие каждый одно телефонное соединение, или интерфейсы цифровых трактов.

Виртуальные окончания, иногда называемые эфемерными, существующие только в течение одной разговорной сессии, являются интерфейсами со стороны IP-сети (например, RTP-окончания), через которые ведутся передача и прием IP-пакетов. Виртуальные окончания создаются шлюзом при получении от контроллера команды Add и ликвидируются при получении команды Subtract,

тогда как физические окончания при получении команды Add или Subtract, соответственно, выводятся из так называемого нулевого контекста или возвращаются обратно в нулевой контекст. Все команды H.248 описаны в главе 4.

Окончание имеет уникальный идентификатор *TerminationID*, который назначается шлюзом при конфигурации порта. Идентификатором физического окончания может служить номер тракта E1 и номер временного канала внутри тракта для Trunking Gateway или номер платы аналоговых интерфейсов и номер абонентского комплекта на этой плате для Access Gateway. Например, идентификатор физического окончания для Trunking Gateway может иметь вид e1/2/20, где e1 – фиксированный префикс, 2 – номер тракта E1, а 20 – номер временного интервала в нем. Другой пример идентификатора TerminationID для Access Gateway – aln/3/12, где aln – фиксированный префикс, 3 – номер платы аналоговых интерфейсов, 12 – номер абонентского комплекта на этой плате. Идентификатор виртуального окончания может быть любым, но обычно он представляет собой фиксированный префикс и какое-либо число, например eph2345, где eph – фиксированный префикс, а 2345 – сгенерированный номер виртуального окончания. Иногда команды могут относиться ко всему шлюзу, и тогда используется общий идентификатор окончания ROOT.

Окончания обладают рядом *Свойств (Properties)*, каждое из которых имеет уникальный идентификатор *PropertyID*. Например, окончания могут обладать способностью генерировать акустические и вызывные сигналы, речевые подсказки, а также выявлять сигналы DTMF. Свойства окончаний группируются в *дескрипторы*, которые, в свою очередь, переносятся в командах протокола. При создании окончаний или при пребывании окончания в нулевом контексте свойства всех дескрипторов, за исключением *TerminationState* и *LocalControl* (см. ниже), должны быть пустыми, то есть не иметь значений. Конфигурация какой-либо характеристики в медиа-шлюзе аннулирует любое значение Свойства окончания, заданное по умолчанию в протоколе. Свойства окончаний делятся на общие *Свойства*, называемые также *Свойствами* состояния окончания (Termination State), и на *Свойства*, относящиеся к медиа-потокам, которые включают в себя локальные *Свойства* (передающей стороны) и *Свойства* принимаемых и передаваемых потоков.

Окончания могут генерировать и передавать *Сигналы (Signals)*, при этом говорят, что Сигнал *применен на* окончании. Следует различать реальные физические сигналы, передаваемые оборудованием медиа-шлюза, и Сигналы, являющиеся информационными элементами протокола H.248. То есть модуль H.248 применяет Сигнал на окончании, и с этого окончания должен быть передан физический

сигнал с характеристиками, описанными в Сигнале. Следует также сказать о *тональных сигналах*, представляющие собой совокупность одно-частотных и многочастотных сигналов, передачу которых можно инициировать, указав соответствующий *идентификатор тонального сигнала*.

Окончания могут быть запрограммированы на обнаружение *Событий (Events)*, при возникновении которых MG должен отправить извещение контроллеру или выполнить определенные действия. Понятие *событие* также дуалистично, с одной стороны – это некоторая ситуация, возникновение которой, по указанным характеристикам и условиям может обнаружить медиа-шлюз, а с другой – это информационный элемент протокола H.248, который как раз и описывает необходимые характеристики и условия, а также предоставляет *событию* идентификатор и наименование. Если возникновение некоторого События отслеживается на окончании, то говорится что Событие *установлено на окончании*. Учитывая, что управляющая информация может быть передана медиа-шлюзу предварительно, то чтобы подчеркнуть отслеживание События в медиа-шлюзе, иногда уточняют: *установлено и активизировано*. Когда на окончании были обнаружены указанные характеристики и условия, то медиа-шлюз *создает Событие* (как параметр H.248, который будет передан к MGC) и отправляет в MGC соответствующее извещение. На Окончании могут собираться *Статистики (Statistics)* и потом передаваться контроллеру по его запросу или при удалении окончания из контекста.

Статистики – это также информационный элемент протокола H.248, и он содержит спецификацию статистических данных, которые должен собирать медиа-шлюз и передавать на MGC.

Все перечисленные элементы: Свойства, Сигналы, События и Статистики имеют соответствующие идентификаторы, которые определяются для них в Пакетах, описанных в 5 главе справочника.

Протокол имеет средства для создания новых окончаний и для модификации значений параметров существующих окончаний. Модификация, помимо изменения таких параметров, как применяемые кодеки, транспортные адреса, используемые планы нумерации, предполагает также возможность редактировать список обнаруживаемых Событий и указание Сигналов, применяемых на окончании. Значения параметров могут быть изменены путем включения в команду соответствующих *дескрипторов*. Если дескриптор в команде опущен, то окончание сохраняет Свойства, присвоенные ему либо по умолчанию, либо предыдущим дескриптором. Если же в дескрипторе, принятом MG, некоторое Свойство

опущено, то оно принимает значение, присвоенное ему по умолчанию. Поскольку в данном разделе пришлось разъяснять много терминов, связанных с настройкой окончаний медиа-шлюза, то здесь же приведем пояснения, касающиеся одного из важнейших параметров окончания, а именно – *плана нумерации*.

В англоязычной литературе используются два отдельных термина, переводящихся на русский язык как *план нумерации*. Так *Dialing Plan* – касается всей сети и равнозначен русскому термину «Система нумерации», а *Digit Map* – это правила, которыми руководствуется отдельно взятый узел, при получении адресной информации. *Digit Map* содержит набор шаблонов, которым могут соответствовать набираемые пользователями адреса (в т.ч. и телефонные номера). Для каждого шаблона определяется количество позиций, допустимые значения для каждой из позиций, а также межсимвольные выдержки времени. По мере приема символов адресной информации от пользователя, неподходящие шаблоны отсеиваются до тех пор, пока, в идеальном случае, не произойдет однозначного соответствия набранного номера одному из шаблонов. Тогда говорят, что произошло *завершение работы* плана нумерации по однозначному соответствию. Работа плана нумерации может завершиться неоднозначно, по выдержке времени или по отсутствию шаблонов, которым может соответствовать введенная информация.

2.3. Контекст (Context)

Контекст – это отображение связи между несколькими окончаниями, то есть абстрактное представление соединения двух или более портов одного шлюза. Контекст описывает топологию и параметры смешивания или коммутации потоков этих окончаний.

В любой момент времени окончание может относиться только к одному контексту, который имеет свой уникальный идентификатор. Существует особый вид контекста – нулевой. Все окончания, входящие в нулевой контекст, не связаны ни между собой, ни с другими портами. Например, абстрактным представлением свободного (не занятого) канала в модели соединений является окончание в нулевом контексте. В общем случае для присоединения окончания к контексту служит команда *Add*. При этом, если контроллер не специфицирует существующий контекст, к которому должен быть подсоединен порт, то шлюз создает новый контекст. Удаление окончания из контекста производится командой *Subtract*. Перемещение окончания из одного контекста в другой производится командой *Move*.

Максимальное количество окончаний, включенных в контекст, ограничивается возможностями шлюза. Если он поддерживает только соединения «точка-точка», то их будет всего два.

Атрибутами контекста являются:

- идентификатор контекста ContextID;
- топология контекста (от какого окончания к какому передается информация и откуда она принимается). Топология контекста описывает потоки информации внутри контекста, т.е. внутри шлюза, в то время как подобный параметр окончания – режим его работы – описывает внешние потоки шлюза, входящие и исходящие;
- приоритет используется для того, чтобы указать шлюзу на первоочередное обслуживание контекста. Приоритетная система предусматривает 16 уровней приоритета, низший – нулевой, высший – 15-й;
- индикатор «экстренного вызова» позволяет получить высший приоритет в обслуживании.

Протокол имеет средства для управления параметрами контекста. Удаление контекста происходит автоматически после исключения из него последнего окончания. Примеры контекстов приведены на рис. 2.1.

На рис. 2.1 изображены (сверху вниз) следующие варианты соединений:

- многоточечное соединение двух физических окончаний, мультиплексируемых в одно виртуальное;
- физическое окончание, находящееся в нулевом контексте, и виртуальное окончание, включенное в контекст, но еще не соединенное ни с одним другим окончанием;
- соединение «точка-точка» одного физического окончания с одним виртуальным, что наиболее характерно для VoIP-шлюзов.

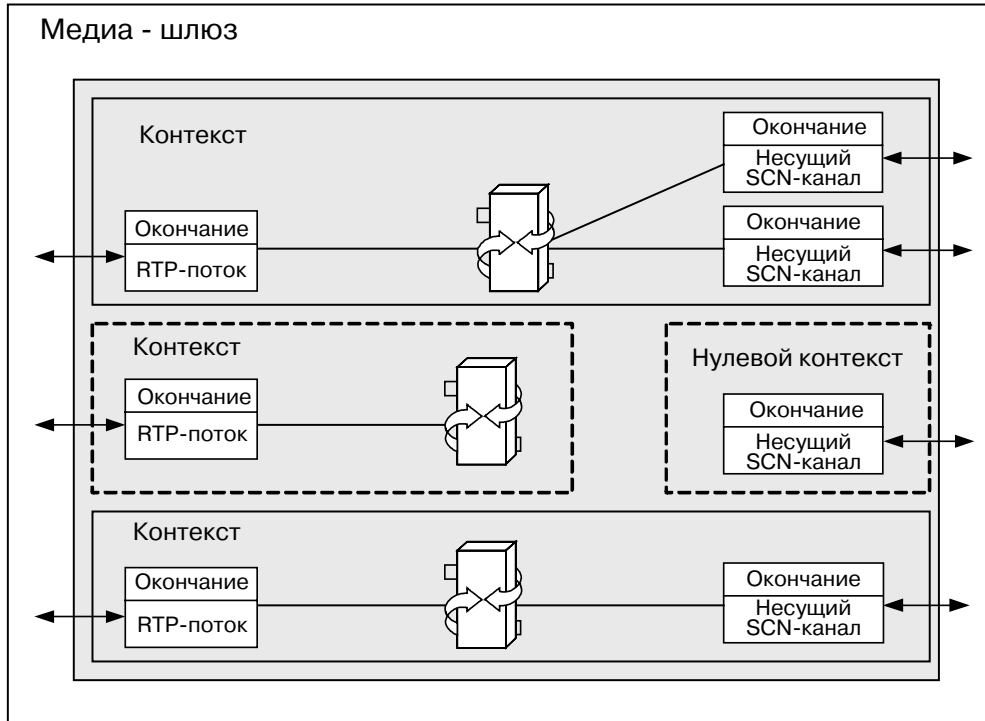


Рис. 2.1. Модель процесса обслуживания вызова, используемая в H.248

2.4. Использование групповых символов

В H.248 используется механизм групповых символов wildcard: ALL и CHOOSE. Первый символ позволяет адресоваться сразу к нескольким элементам одновременно, а второй позволяет предоставить шлюзу право выбора любого подходящего элемента. Эти символы могут использоваться как в отношении контекстов, так и в отношении окончаний.

2.4.1. Явно указанный ContextID и групповой символ в TerminationID

Когда ContextID указан явно, а TerminationID определен символом ALL, эффект будет аналогичен повторению команды для каждого отдельного окончания, и для каждой из команд создается отдельный ответ. Поэтому общий объем передаваемого трафика может оказаться значительным. Использование ALL не распространяется на корневое окончание ROOT. Если ни одно окончание в указанном контексте не будет соответствовать групповому символу, то будет передано извещение об ошибке 431.

2.4.2. Групповой символ в ContextID и явно указанный TerminationID

В случае если ContextID определен групповым символом (например, ALL), а окончание определено явно, эффект будет аналогичен передаче команды на ненулевой контекст, содержащий указанное окончание. Таким образом, производится поиск окончания по контекстам, а действие выполняется в найденном контексте. Если окончание не обнаружено ни в одном из ненулевых контекстов, передается извещение об ошибке 431, если же ненулевых контекстов нет, то передается сообщение об ошибке 411.

2.4.3. Групповые символы в ContextID и в TerminationID

Когда и окончание, и контекст определены групповыми символами, эффект будет аналогичен повторению команды для всех окончаний, соответствующих групповому символу и находящихся в ненулевых контекстах. Если ни одно окончание ни в одном контексте не соответствуют групповым символам, то передается сообщение об ошибке 431.

2.4.4. Обобщенные ответы

Если индивидуальные ответы для каждого окончания не требуются, может быть запрошен групповой ответ UNION, который будет содержать групповой символ для идентификатора окончания и последовательность параметров со списком значений, которые они принимают на окончаниях, определенных групповым символом.

Например, если Termination Ta имеет значения свойств $p1=a$, $p2=b$, а Termination Tb – значения свойств $p2=c$, $p3=d$, обобщенный ответ UNION будет состоять из TerminationId, содержащего групповой символ, и последовательности свойств $p1=a$, $p2=b,c$ и $p3=d$.

Обобщенные ответы могут быть полезны при командах проверки (Audit). Если групповой символ UNION используется вместе с групповым символом в ContextID, то обобщенный ответ будет содержать ContextID=ALL, TerminationID=ALL и набор значений свойств. Если при выполнении запроса с групповыми символами, требующего обобщенного ответа, произойдет ошибка, связанная с одним из элементов, то будет передан обобщенный ответ для успешных команд, а затем – дополнительные ответы для окончаний, с которыми команду выполнить не удалось.

2.5. Мультиплексирование потоков

Медиа-шлюзы могут обрабатывать мультиплексированные медиа-потоки. Например, рекомендация H.221 описывает такую структуру потока, когда несколько информационных потоков мультиплексируются в несколько цифровых каналов 64 кб/с.

Согласно модели обслуживания вызова H.248, подобный случай обрабатывается следующим образом. Для каждого несущего канала, который переносит часть мультиплексированных потоков, задается физическое или виртуальное окончание.

Окончания, являющиеся источниками/приемниками информации цифровых каналов, соединены в контексте с «мультиплексирующим окончанием». Мультиплексирующее окончание – это виртуальное окончание, представляющее пакетную сессию. Дескриптор мультиплексора для этого окончания описывает используемое мультиплексирование и указывает порядок, в котором содержимое цифровых пакетов помещается в пакеты или ячейки. Индивидуальные информационные потоки, переносимые в рамках сессии, описываются дескриптором потока на мультиплексирующем окончании. На приведенных ниже рисунках показано применение мультиплексирующего окончания и дескриптора мультиплексора.

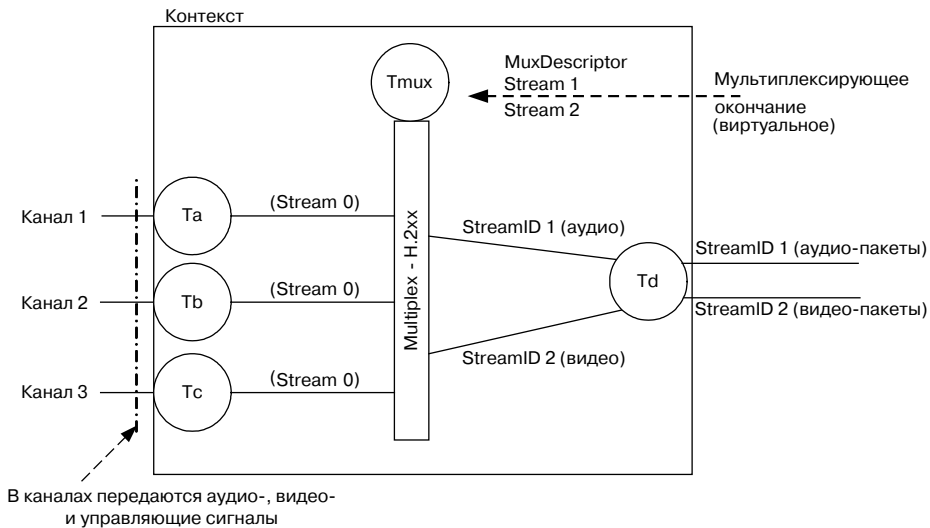


Рис. 2.2. Сценарий мультиплексирования – каналный интерфейс – пакетный интерфейс

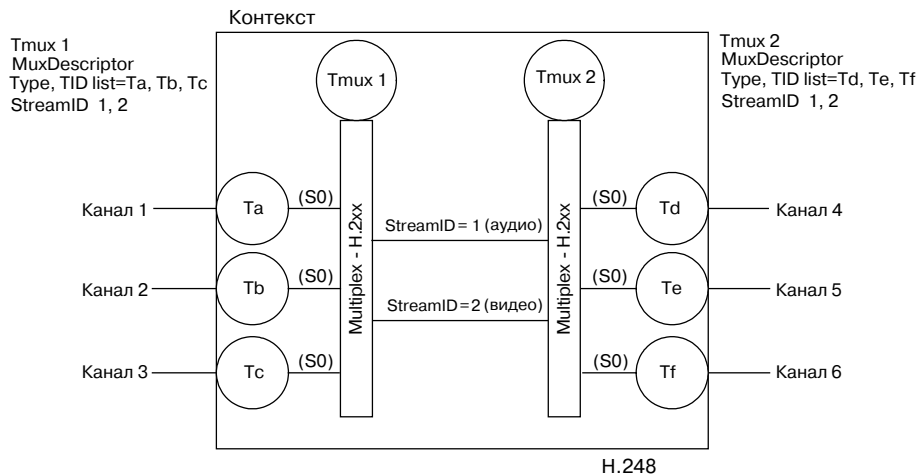


Рис. 2.3. Сценарий мультиплексирования – каналный интерфейс – каналный интерфейс

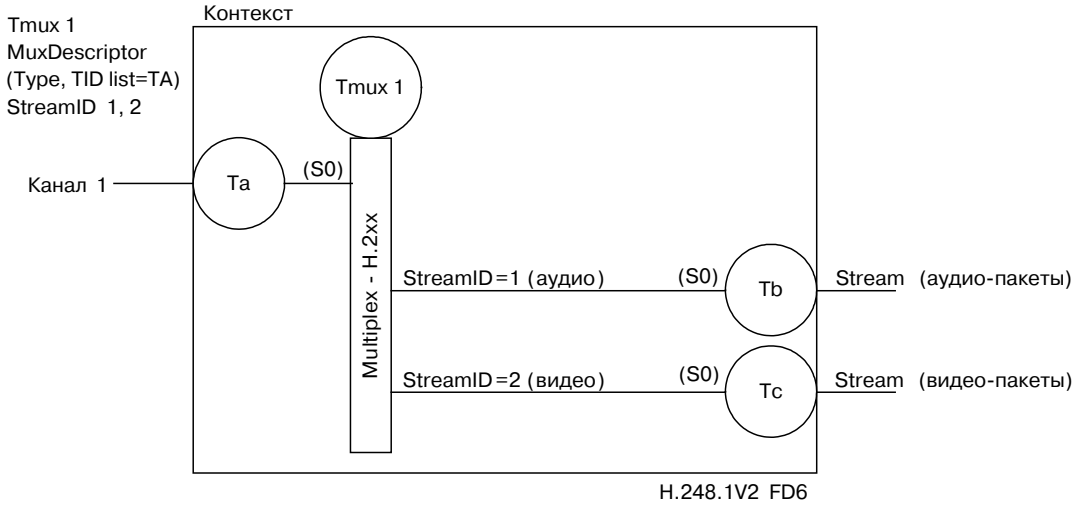


Рис. 2.4. Сценарий мультиплексирования – точка – группа точек

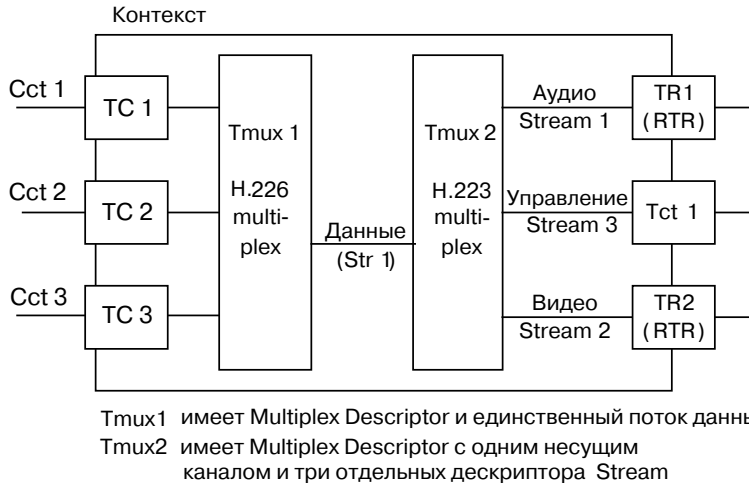


Рис. 2.5. Сценарий мультиплексирования – каскадное мультиплексирование

Глава 3. Дескрипторы протокола H.248

3.1. Понятие дескриптора

Основными элементами алфавита H.248 являются его команды. Они позволяют управлять контекстами, окончаниями и их свойствами. Команды могут содержать в себе более мелкие элементы, называемые дескрипторами, а также объединяться в более крупные – *действия (Actions)*, *транзакции (Transactions)* и *сообщения (Messages)*. На рис. 3.1 приведена типичная структура вложения друг в друга основных элементов и понятий протокола H.248. Она не является формальной, так как не учитывает разделения на запрос и ответ, присутствия необязательных элементов и прочее.

Рассматривать структуру протокольных данных H.248 мы начнем с меньшего элемента – дескриптора.

Дескриптор состоит из названия и списка элементов и описывает все возможные характеристики окончаний и контекстов. С точки зрения окончаний дескрипторы – это сгруппированные по смысловому значению наборы характеристик, а с точки зрения H.248 – это параметры команд.

Многие команды содержат общие дескрипторы, – один и тот же дескриптор может передаваться в разных командах, т.е. обычно нет жесткой привязки дескриптора к какой-либо команде H.248.

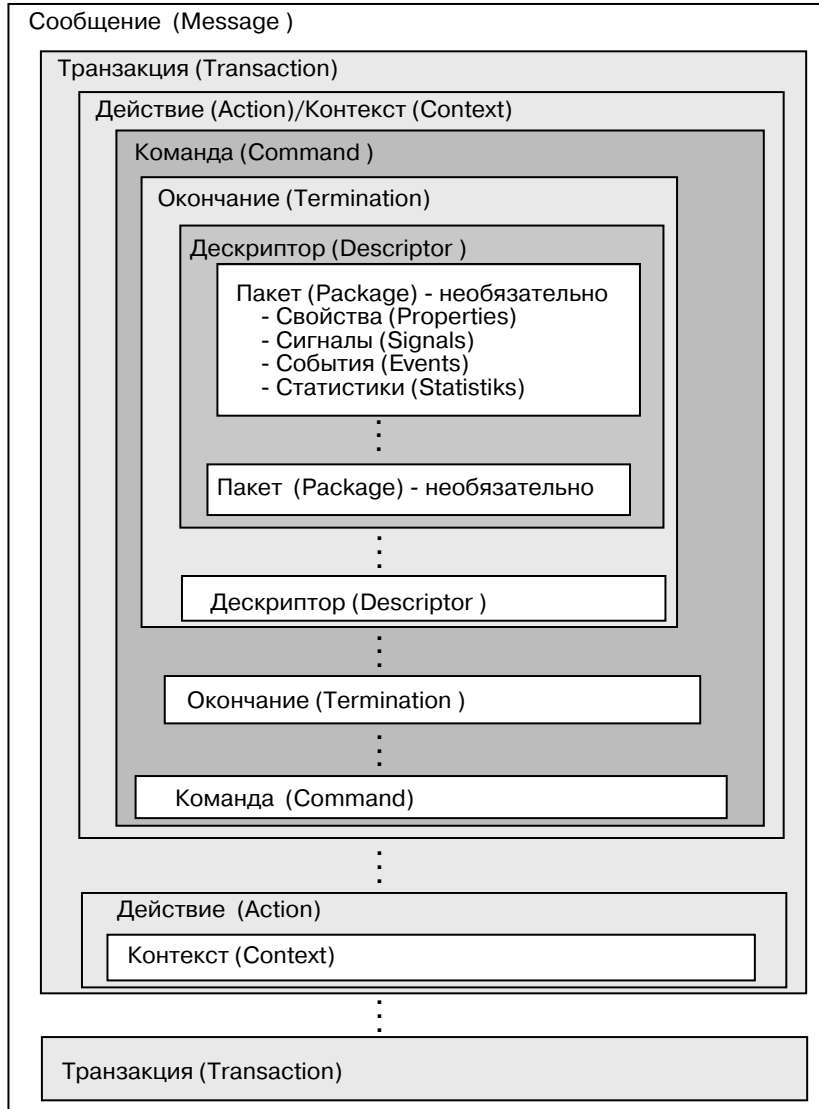


Рис. 3.1. Типичная структура вложения основных элементов и понятий Н.248

Общий вид дескриптора можно представить следующим образом:

Название дескриптора = <Идентификатор ID> {параметр = значение, параметр = значение...}.

Параметры могут быть строго *определенными*, не строго *определенными* и *неопределенными* (*fully specified, overspecified, underspecified*).

Строго определенные (*fully specified*) параметры имеют только одно указанное отправителем команды и точно выраженное значение, которое получатель команды должен использовать.

Не строго определенные (*overspecified*) параметры имеют список потенциально возможных значений, составленный отправителем команды. Получатель команды может выбрать из предложенного списка любое значение и известить об этом отправителя. Не строго определенные параметры могут быть также заданы в виде интервалов значений или нижних или верхних предельных значений.

Неопределенные (*underspecified*) параметры имеют значение CHOOSE, позволяющее получателю выбрать любое поддерживаемое им значение параметра.

Если какой-то дескриптор (кроме Audit descriptor) не специфицирован в команде (например, отсутствует), то используются предыдущие значения дескриптора для этого окончания, если таковые имеются. Каждая команда определяет идентификатор окончания (TerminationID), к которому она относится. Этот идентификатор может быть групповым («wildcarded»). В этом случае эффект будет таким же, как если бы команда повторялась для каждого окончания в отдельности с использованием всякий раз соответствующего идентификатора окончания. При рассмотрении нескольких конкретных дескрипторов в следующих параграфах помимо общего описания будут приводиться в табличном виде параметры, в формате ASN.1 входящие в тот или иной дескриптор. Однако, многие параметры являются составными, и чтобы не усложнять структуру таблицы, их описание будет приводиться под ней. Подобный разбор сделан только для того, чтобы помочь читателю освоить чтение параметров в двоичном и текстовом кодировании, и носит иллюстративный характер.

3.2. Дескриптор модема (Modem Descriptor)

Описание дескрипторов мы начнем с малозначимого дескриптора модема, использование которого, начиная с момента появления документа H.248.1