Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«Самарский государственный аэрокосмический университет
имени академика С.П. Королева
(национальный исследовательский университет)»

С.В. Суханов

Компьютерные сети. Конспект лекций

Электронное учебное пособие

САМАРА
2011

Автор: СУХАНОВ Сергей Васильевич

Учебное пособие содержит конспект лекций по курсу Компьютерные сети. Содержание соответствует вводному курсу по компьютерным сетям. Рассмотрены терминология и базовые концепции телекоммуникационных технологий. Модель OSI. Вопросы стандартизации компьютерных сетей. Характеристики линий связи, типы и устройство кабелей, принципы передачи цифровых данных на физическом уровне. Технологии локальных сетей на основе разделяемой среды и коммутируемые локальные сети. Технологии беспроводных сетей. Адресация, технологии и протоколы сетей на основе стека TCP/IP. Алгоритмы и протоколы маршрутизации.

Учебное пособие предназначено для бакалавров направлений 010400.62 «Прикладная математика и информатика» и 010900.62 «Прикладные математика и физика»

Тема лекции: **Введение в компьютерные сети**

План лекции:

1. Сводка терминов
2. Задачи физической передачи данных по линии связи
3. Топология физических связей компьютерных сетей
4. Адресация узлов сети
5. Обобщенная задача коммутации
6. Коммутация каналов и коммутация пакетов
7. Физическая и логическая структуризация сетей

Сводка терминов

компьютерная сеть — это набор компьютеров, связанных коммуникационной системой и снабженных соответствующим программным обеспечением, которое предоставляет пользователям сети доступ к ресурсам этого набора компьютеров; сеть могут образовывать компьютеры разных типов — небольшие микропроцессоры, рабочие станции, мини-компьютеры, персональные компьютеры или суперкомпьютеры; передачу сообщений между любой парой компьютеров сети обеспечивает коммуникационная система, которая может включать кабели, повторители, коммутаторы, маршрутизаторы и другие устройства; *компьютерная сеть* позволяет пользователю работать со своим компьютером, как с автономным, и добавляет к этому возможность доступа к информационным и аппаратным ресурсам других компьютеров сети.

Сетевая технология – это согласованный набор программных и аппаратных средств (драйверов, сетевых адаптеров, кабелей и разъемов) и механизмов передачи данных по линиям связи, достаточный для построения вычислительной сети.

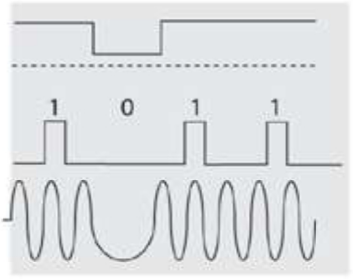
Локальные сети (Local Area Network, LAN) – объединения компьютеров, на небольшой территории, принадлежат одной организации, построены на основе единой сетевой технологии.

Глобальные сети (Worldwide Area Network, WAN) – объединения компьютеров, на больших территориях, построены на основе разных сетевых технологий (Интернет).

Городские сети, сети мегаполисов (Metropolitan Area Network, MAN) – объединения компьютеров, на городских территориях , построены на основе разных сетевых технологий (Сеть образования и науки, Дом.ру.)

Задачи физической передачи данных по линии связи

- Кодирование



Потенциальное

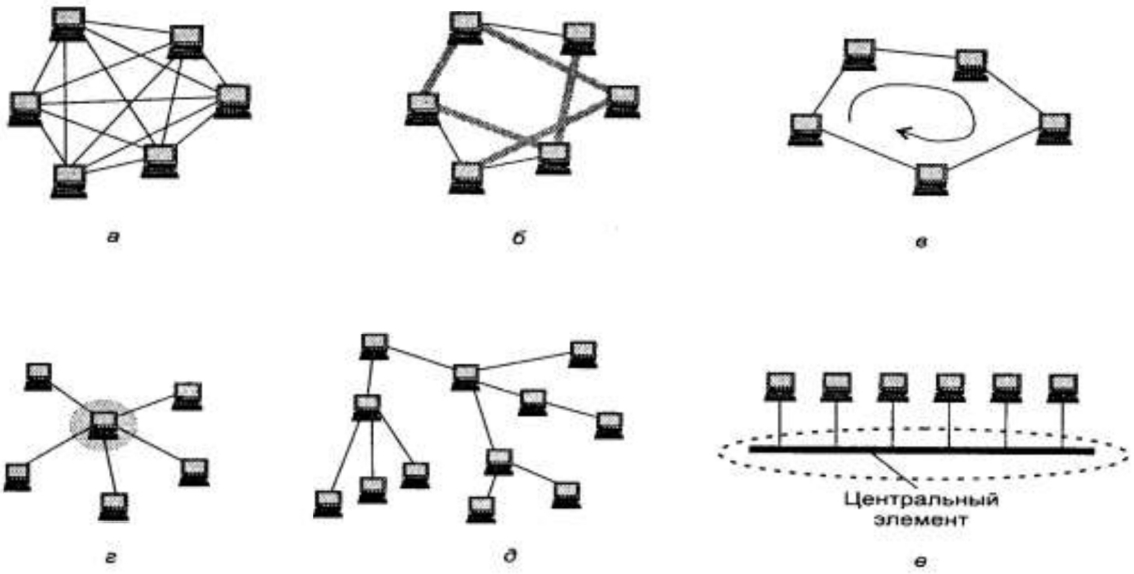
Импульсное

Частотная модуляция

- Компрессия
- Преобразование информации из параллельной в последовательную форму (экономия линий связи)
- Обеспечение надежности передачи - контрольные суммы, квитирование

Элементы, реализующие физическую передачу:

Сетевые адаптеры, сетевые интерфейсы коммутаторов, маршрутизаторов и т.д. Аппаратура передачи данных (модемы)



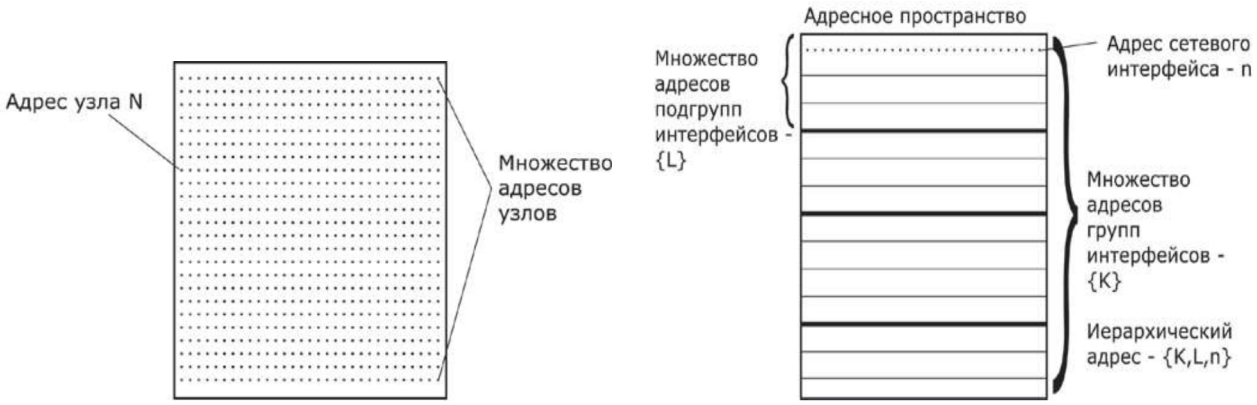
Топологии физических связей компьютерных сетей а) полносвязная; б) ячеистая (mesh); в) кольцо; г) звезда; д) дерево; е) общая шина.

Адресация узлов сети

Адреса могут быть **числовыми** (129.26.255.254) и ***символьными*** (site.domain.ru).

Адресное пространство может иметь **плоскую** (MAC-адрес) или **иерархическую** организацию.

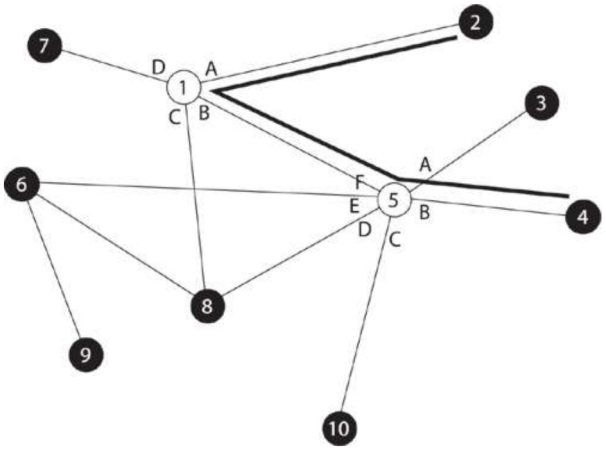
Для преобразования адресов из одного вида в другой используются специальные протоколы, которые называют ***протоколами разрешения адресов (address resolution),*** например протокол ARP.



Обобщенная задача коммутации

В самом общем виде задача *коммутации* — задача соединения конечных узлов через сеть транзитных узлов — может быть представлена в виде нескольких взаимосвязанных частных задач:

- Определение *информационных потоков*, для которых требуется прокладывать пути.
- Определение маршрутов для *потоков*.
- Сообщение о найденных маршрутах узлам сети.
- Продвижение – распознавание *потоков* и локальная коммутация на каждом транзитном узле.
- Мультиплексирование и демультиплексирование *потоков*.
- Разделение среды передачи



- Определить потоки – это значит задать для них набор отличительных признаков, на основании которых коммутаторы смогут направлять потоки по предназначенным для них маршрутам.
- Определить маршрут — однозначно задать последовательность транзитных узлов и их интерфейсов, через которые надо передавать данные, чтобы доставить их адресату. В качестве критериев выбора могут выступать, например: номинальная пропускная способность; загруженность каналов связи; задержки, вносимые каналами; количество промежуточных транзитных узлов; надежность каналов и транзитных узлов.

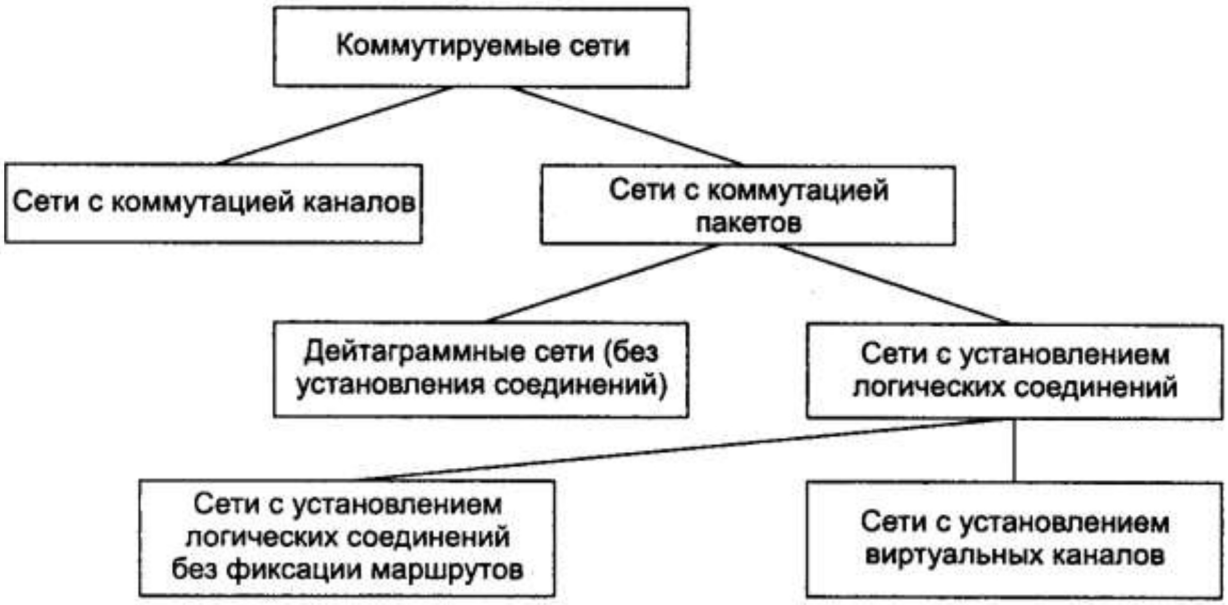
- Оповестить сеть о найденных маршрутах — это значит вручную или автоматически настроить каждый коммутатор таким образом, чтобы он "знал", в каком направлении следует передавать каждый поток.
- Продвижение потоков — распознавание потоков и коммутация на каждом транзитном узле.
- Демультиплексирование (demultiplexing) — разделение суммарного агрегированного потока, поступающего на один интерфейс, на несколько составляющих потоков. Как правило, операцию коммутации сопровождает также обратная операция:
- Мультиплексирование (multiplexing) — образование из нескольких отдельных потоков общего агрегированного потока, который можно передавать по одному физическому каналу связи.

Коммутация каналов и коммутация пакетов

- коммутация каналов (circuit switching) (коммутационная сеть образует между конечными узлами непрерывный составной физический канал из последовательно соединенных коммутаторами промежуточных канальных участков);
- коммутация пакетов (packet switching) (пульсирующий трафик, буферизация);
- коммутация сообщений.

Сравнение сетей с коммутацией каналов и пакетов

Коммутация каналов	**Коммутация пакетов**
Необходимо предварительно устанавливать соединение	Отсутствует этап установления соединения (дейтаграммный способ)
Адрес требуется только на этапе установления соединения	Адрес и другая служебная информация передается с каждым пакетом
Сеть может отказать абоненту в установлении соединения	Сеть всегда готова принять данные от абонента
Гарантированная пропускная способность (полоса пропускания) для взаимодействующих абонентов	Пропускная способность сети для абонентов неизвестна, задержки передачи носят случайный характер
Трафик реального времени передается без задержек	Ресурсы сети используются эффективно при передаче пульсирующего трафика
Высокая надежность передачи	Возможные потери данных из-за переполнения буферов
Нерациональное использование пропускной способности каналов, снижающее общую эффективность сети	Автоматическое динамическое распределение пропускной способности физических каналов в соответствии с фактической интенсивностью трафика абонентов



Методы продвижения пакетов в сетях с коммутацией пакетов



Передача без установления соединения (*а*) и с установлением соединения (*б*)

Структуризация локальных сетей

При построении больших сетей *однородная структура* связей превращается из преимущества в недостаток. Важнейшими *ограничениями* здесь являются:

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

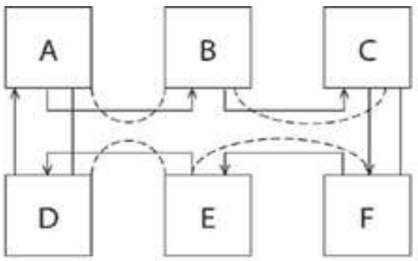
Для снятия этих *ограничений* используются методы ***структуризации*** сети и специальное структурообразующее ***коммуникационное*** оборудование — ***повторители***, ***концентраторы***, **мосты, коммутаторы, маршрутизаторы**.

Различают:

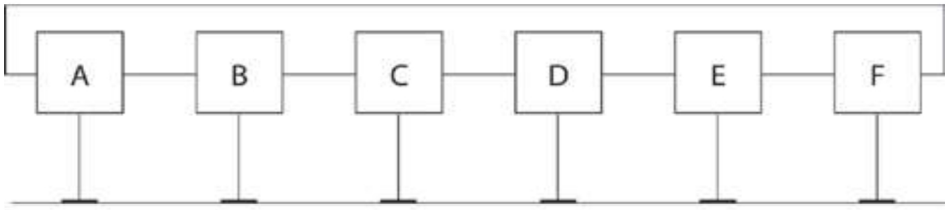
1. Топологию физических связей (***физическую структуру*** сети). В этом случае конфигурация физических связей определяется электрическими соединениями компьютеров, то есть ребрам графа соответствуют отрезки кабеля, связывающие пары узлов.

2. Топологию логических связей (***логическую структуру*** сети). Здесь в качестве логических связей выступают маршруты передачи данных между узлами сети, которые образуются путем соответствующей настройки *коммуникационного оборудования*.

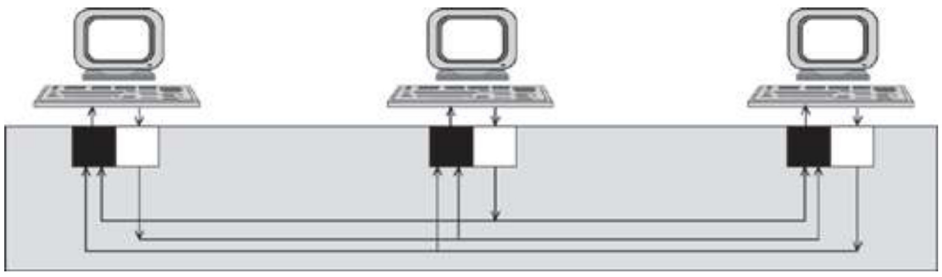
1) логическая и физическая структуры сети совпадают;
2) логическая структура не совпадает с физической;
3) концентратор (Hub) Ethernet дает физическую топологию «звезда». Логическая топология осталась: «общая шина».



Физическое кольцо
и логическое кольцо

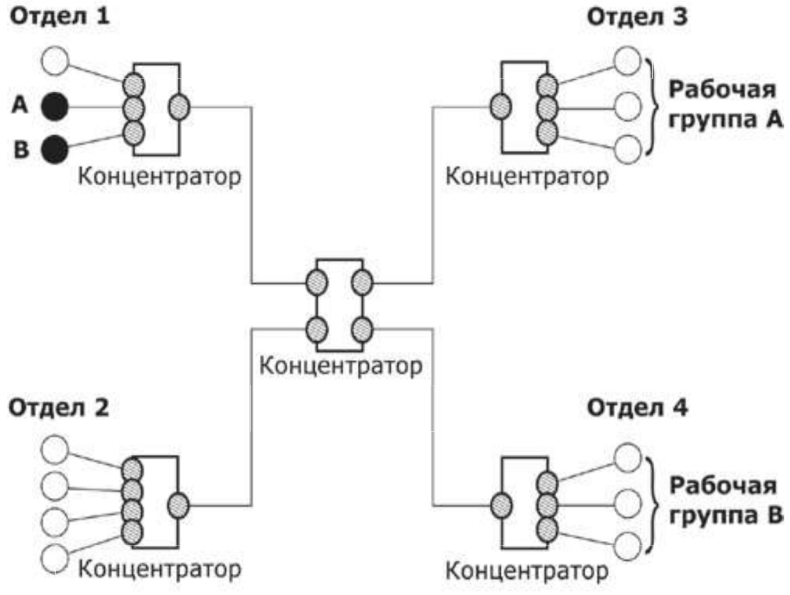


Физическая общая шина, логическое кольцо

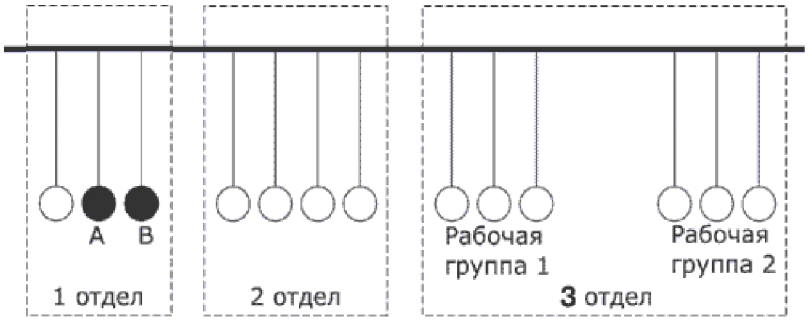


Физическая структуризация сети

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, что позволяет преодолеть ограничения на длину линий связи за счет улучшения качества передаваемого сигнала — восстановления его мощности и амплитуды, улучшения фронтов и т. п.



■ В результате физической структуризации логическая структура не изменилась



Повторитель с несколькими портами, соединяющий несколько физических сегментов, называют **концентратором** или **хабом** (hub). Он **не изменяет** логическую структуру сети.

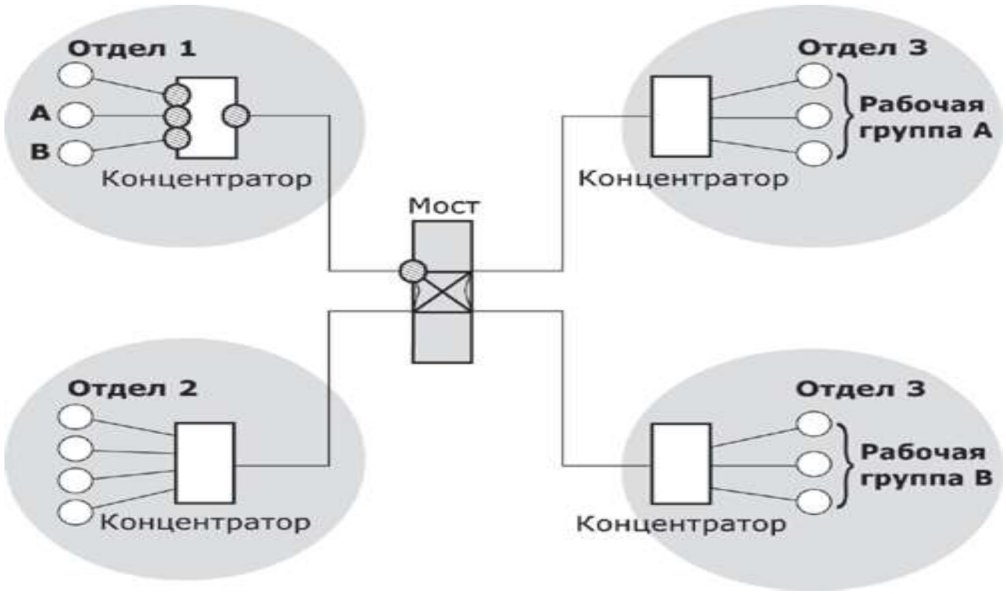
Логическая структуризация сети

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.

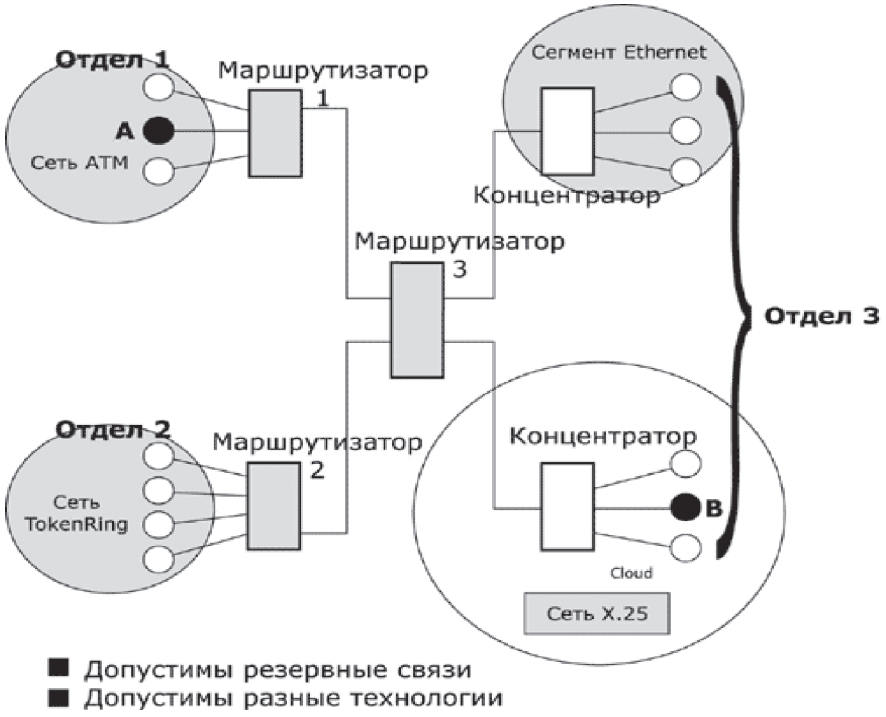
Для логической структуризации сети используются коммуникационные устройства:

мосты;

коммутаторы;
маршрутизаторы;
шлюзы.



Логическая структуризация сети с помощью **моста**



Логическая структуризация сети с помощью **маршрутизаторов**

Тема лекции: **Модель взаимодействия открытых**

План лекции:

1. История и значение модели ISO/OSI
2. Ключевые термины модели OSI
3. Схематическое представление модели OSI
4. Инкапсуляция / декапсуляция
5. Соответствие английских и русских терминов
6. Соответствие функций сетевых устройств уровням модели OSI
7. Описания каждого из семи уровней модели OSI
8. Сетезависимые и сетенезависимые уровни модели OSI

История и значение модели ISO/OSI

С 1977 по 1984 гг ряд международных организаций по стандартизации (**Международная организация по стандартизации** - International Organization for Standardization, ISO, **Международный союз электросвязи** - *International Telecommunications Union, ITU* и др.) разработали стандартную **Модель взаимодействия открытых систем** (**Open System Interconnection, OSI**).

Назначение модели **OSI** состоит *в обобщенном представлении средств сетевого взаимодействия*. В настоящее время эта модель играет роль *универсального языка* сетевых специалистов, поэтому ее называют *справочной моделью*.

Ключевые термины модели OSI

Формализованные правила, определяющие последовательность и формат *сообщений*, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются ***протоколом***.

Интерфейс — определяет последовательность и формат *сообщений*, которыми обмениваются сетевые компоненты, лежащие на соседних уровнях в одном узле. *Интерфейс* определяет набор услуг, предоставляемый данным уровнем соседнему уровню.

Средства каждого уровня должны отрабатывать, во-первых, 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.

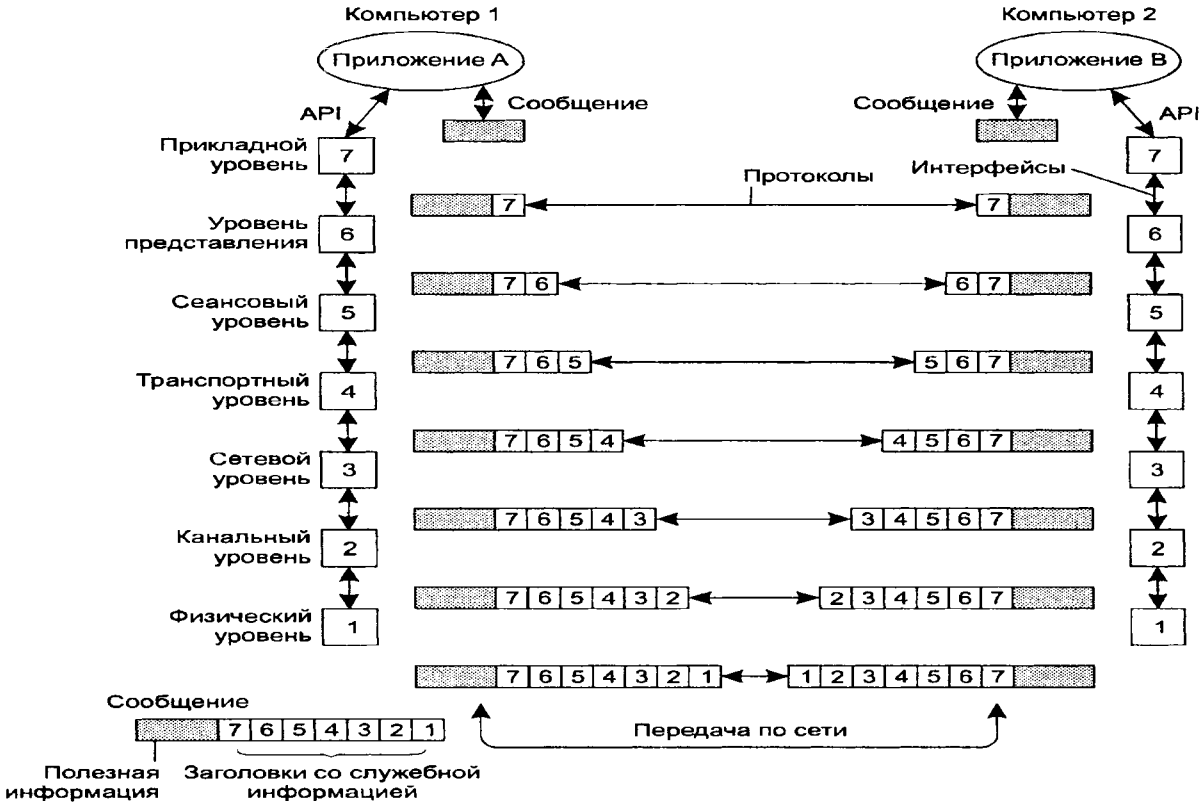
Аппаратно протоколы реализуются не только компьютерами, но и другими сетевыми устройствами — ***концентраторами***, ***мостами***, ***коммутаторами***, ***маршрутизаторами*** и т. д.

Модель взаимодействия открытых систем (Open System Interconnection, OSI) определяет:

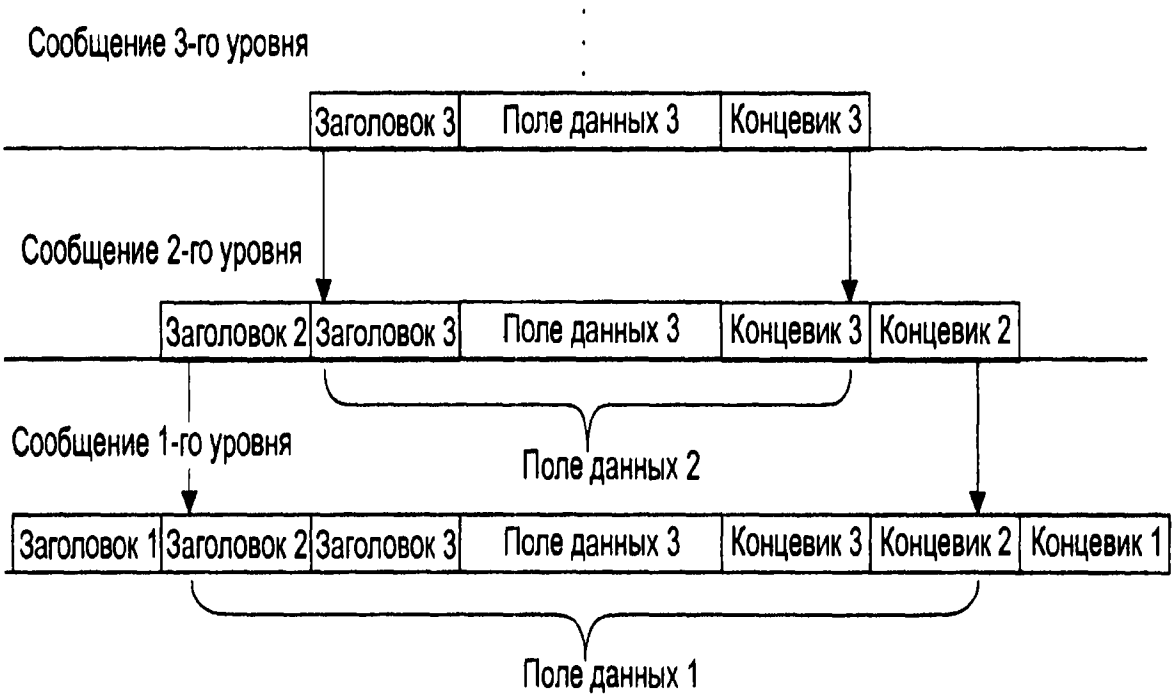
- уровни взаимодействия систем в сетях с коммутацией пакетов,
- стандартные названия уровней,
- функции, которые должен выполнять каждый уровень.

Модель **OSI *не содержит*** описаний реализаций конкретного набора протоколов.

В стандартах ISO для обозначения единиц данных, с которыми имеют дело протоколы разных уровней, используется общее название **протокольный блок данных (Protocol Data Unit, PDU)**.



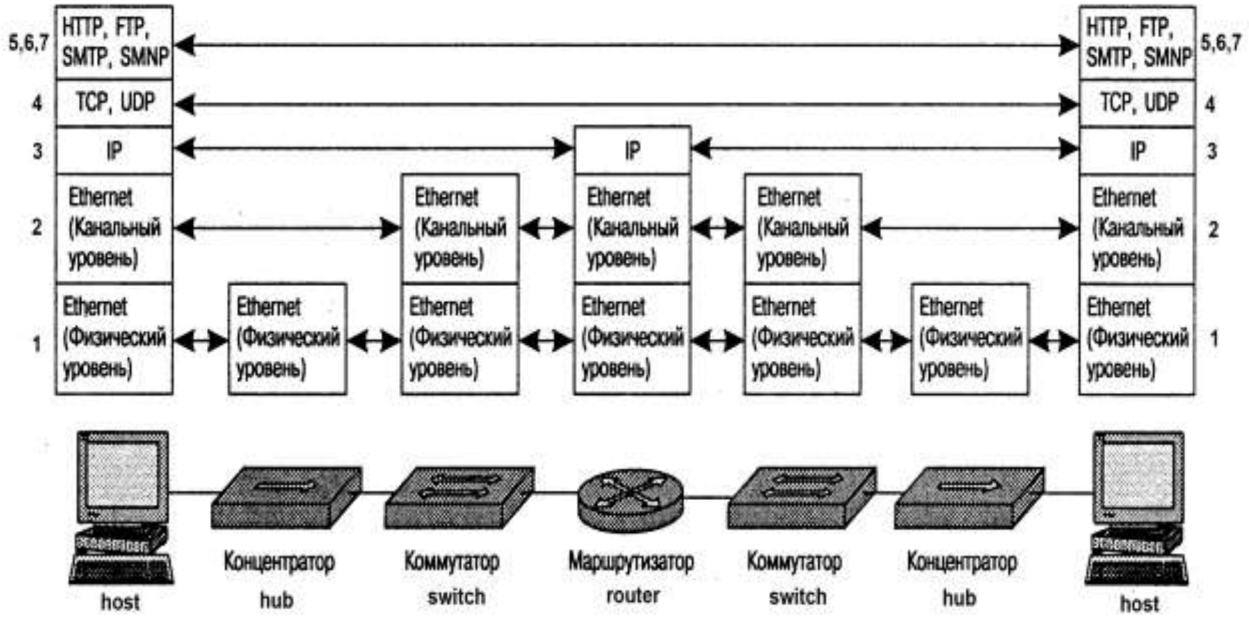
Модель взаимодействия открытых систем ISO/OSI



Инкапсуляция / декапсуляция (вложенность)

Таблица соответствия английских и русских терминов модели ISO/OSI

№ уровня	Уровень	Layer	(Протокольная) Единица данных уровня	Protocol data unit (PDU)
7	Прикладной (Приложений)	Application	«данные»	“data”
6	Представления (Презентационный)	Presentation		
5	Сеансовый (Сессионный)	Session		
4	Транспортный	Transport	Сегмент / дейтаграмма	segment/ datagram
3	Сетевой	Network	пакет	packet
2	Канальный	Data Link	кадр	frame
1	Физический	Physical	«бит»	“bit”



Соответствие функций различных устройств в сети уровням модели OSI

Физический уровень модели OSI

Физический уровень (Physical layer) имеет дело с передачей битов по физическим каналам связи (витая пара, оптоволоконный кабель). Здесь стандартизуются:

- Характеристики физических сред передачи данных (полоса пропускания, помехозащищенность, волновое сопротивление и др.).
- Характеристики электрических сигналов, передающих дискретную информацию (крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов).
- Типы разъемов и назначение каждого контакта.

Физический уровень реализуется аппаратно и выполняет следующие основные функции:

- передача битов по физическим каналам;
- формирование электрических сигналов;
- кодирование информации;
- синхронизация;
- модуляция.

Канальный уровень модели OSI

Функции **канального уровня** (**Data Link Layer**) (реализуются программно-аппаратно компьютерами, мостами, коммутаторами и маршрутизаторами):

- в **локальных сетях** с типовой топологией - надежная доставка кадра между ***любыми*** узлами сети (Ethernet), включает:
    - проверку доступности разделяемой среды;
    - выделение *кадров* из потока данных, поступающих по сети; формирование *кадров* при отправке данных;

- подсчет и проверку *контрольной суммы*.

- в **глобальных сетях** с произвольной топологией - надежная доставка кадра между двумя ***соседними*** узлами в сети (примеры протоколов «точка-точка» - PPP, HDLC)

Сетевой уровень модели OSI

Сетевой уровень (network layer) – уровень межсетевого взаимодействия (internetworking), PDU - **пакет (packet).**

Функции сетевого уровня реализуются:

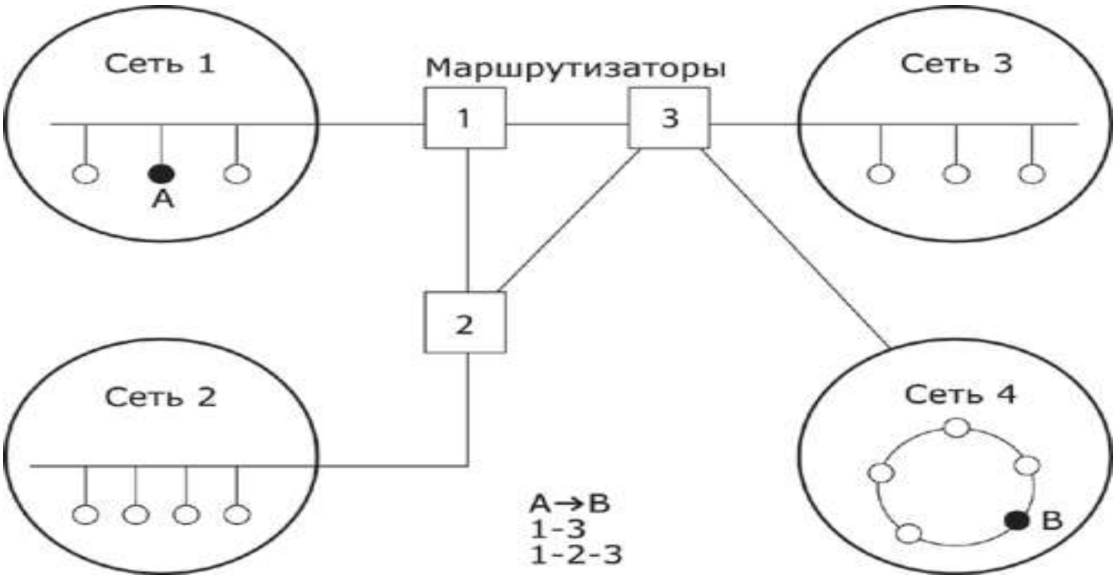
- специальными устройствами – **маршрутизаторами** (routers);
- набором **протоколов**:
  - протоколами **маршрутизации** (**routing** protocols - RIP, OSPF),
  - **сетевыми** протоколами (**routed** protocols - IP),
  - протоколами **разрешения адресов** (**address resolution** protocols - ARP).

Задачи сетевого уровня — доставка пакета:

- между любыми двумя узлами сети с произвольной топологией;
- между любыми двумя сетями в составной сети.

Сеть — совокупность компьютеров, использующих для обмена данными единую сетевую технологию, сетевой адрес узлов которой содержит единый номер сети.

Сети соединяются между собой маршрутизаторами, устройствами, которые собирают информацию о топологии межсетевых соединений (в таблицы маршрутизации) и пересылают (продвигают – forwarding) пакеты 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).

Маршрут — последовательность прохождения пакетом маршрутизаторов в составной сети.

Транспортный уровень модели OSI

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* (0-4). 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,
– способностью к *обнаружению и исправлению ошибок* передачи, таких как искажение, потеря и дублирование *пакетов*.

Транспортный уровень обеспечивает доставку информации с требуемым качеством между любыми узлами сети. Для этого выполняется:

- разбивка сообщения *сеансового уровня* на *пакеты* , их *нумерация*;
- *буферизация* принимаемых *пакетов*;
- *упорядочивание* прибывающих *пакетов*;
- *адресация* прикладных *процессов*;
- *управление потоком*.

Как правило, все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети — компонентами их сетевых операционных систем (протоколы TCP и UDP *стека* TCP/IP и протокол SPX *стека* Novell). Протоколы четырех нижних уровней обобщенно называют сетевым транспортом или транспортной подсистемой.

Сеансовый уровень модели OSI

Сеансовый уровень (***Session layer***) обеспечивает управление *диалогом*: фиксирует, какая из сторон является активной в настоящий момент, предоставляет средства *синхронизации*. На практике функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

Функции сеансового уровня :

- установление способа обмена сообщениями (*дуплексный или полудуплексный*);
- *синхронизация* обмена сообщениями;
- организация "*контрольных точек*" *диалога*.

Уровень представления модели OSI

Уровень представления (***Presentation layer***) обеспечивает представление передаваемой по сети информации, не меняя при этом ее содержания.

На этом уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных служб (например, протокол Secure Socket Layer (SSL) обеспечивает

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

Таким образом ***уровень представления*** согласовывает представление (синтаксис) данных при взаимодействии двух прикладных процессов, выполняя следующие функции:

- преобразование данных из внешнего формата во внутренний;
- шифрование и расшифровка данных.

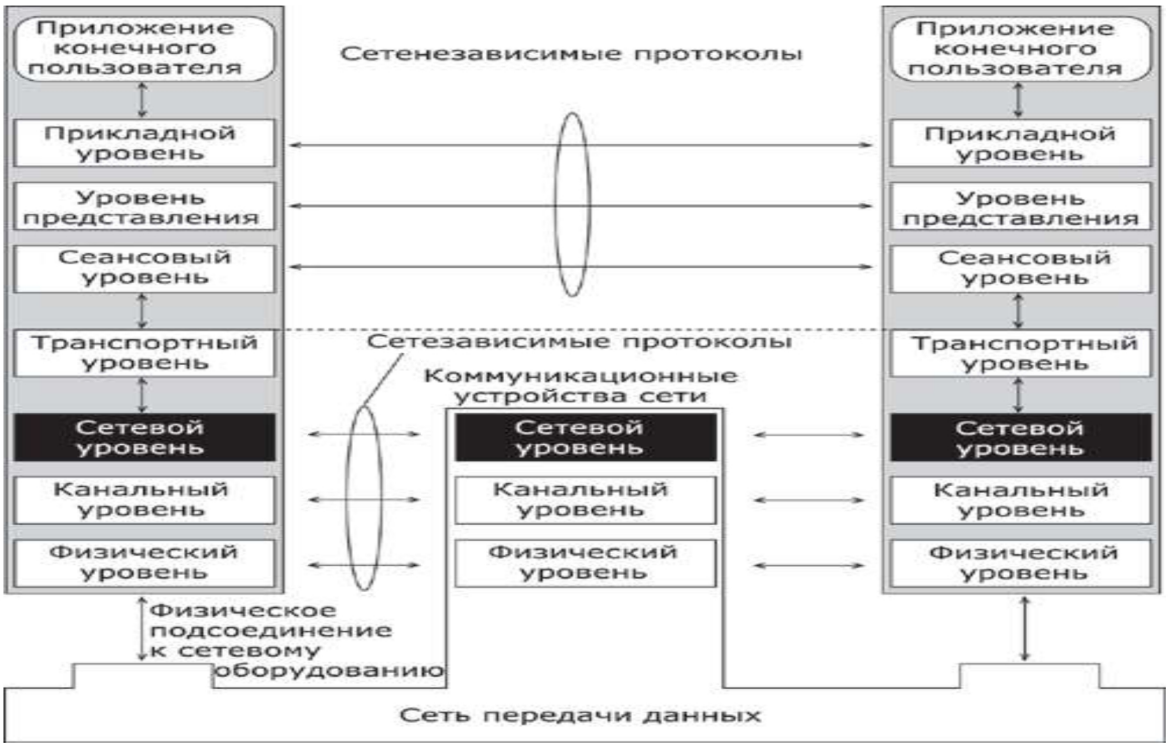
Прикладной уровень модели OSI

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 совместную работу, например с помощью протокола электронной почты. Единица данных, которой оперирует прикладной уровень, обычно называется ***сообщением*** (***message***).

Таким образом, ***прикладной уровень*** — набор всех сетевых сервисов, которые предоставляет система конечному пользователю, например:

- идентификация, проверка прав доступа;
- принт- и файл-сервис, почта, удаленный доступ...

Примеры: NFS, FTP и TFTP, входящие в *стек* TCP/IP; SMB в Microsoft Windows NT; NCP в операционной системе Novell NetWare.



Сетезависимые и сетенезависимые уровни модели OSI

Тема лекции: **Стандартизация сетей**

План лекции:

1. Термины и мотивация
2. Источники стандартов
3. Организации, разрабатывающие стандарты в области компьютерных сетей
4. Управление и стандарты Internet
5. Стандартные стеки коммуникационных протоколов
6. Соответствие популярных стеков протоколов модели OSI
7. Стек TCP/IP

Открытые системы и спецификации

В широком смысле **открытой системой** может быть названа любая система, построенная в соответствии с открытыми спецификациями.

Под термином **"спецификация"** понимают формализованное описание аппаратных или программных компонентов, способов их функционирования, взаимодействия с другими компонентами, условий эксплуатации, ограничений и особых характеристик. Не всякая спецификация является стандартом.

Под **открытыми спецификациями** понимаются опубликованные, общедоступные спецификации, соответствующие стандартам и принятые в результате достижения согласия после всестороннего обсуждения всеми заинтересованными сторонами.

Что дает стандартизация сетей?

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 одних компонентов сети другими, что позволяет сети развиваться с минимальными затратами;
- возможность легкого сопряжения одной сети с другой;
- простота освоения и обслуживания сети.

Источники стандартов:

- **стандарты отдельных фирм** (например, стек протоколов *DECnet* компании Digital Equipment или графический интерфейс OPEN LOOK для Unix-систем компании 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, 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), многочисленные *стандарты* Международного союза электросвязи (ITU), в том числе *стандарты* на сети с коммутацией пакетов X.25, сети frame relay, ISDN, модемы и многие другие.

Организации, разрабатывающие стандарты в области компьютерных 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 стала модель взаимодействия открытых систем OSI. В соответствии с **моделью OSI** этой организацией был разработан стандартный стек коммуникационных протоколов OSI.

Международный союз электросвязи (*International Telecommunications Union, ITU*) - специализированный орган Организации Объединенных Наций.

Главную роль в *стандартизации* вычислительных сетей играет его Международный консультативный комитет по телефонии и телеграфии (МККТТ) (Consultative Committee on International Telegraphy and Telephony, CCITT).

После реорганизации ITU CCITT (1993) 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, передачи данных, аудио- и видеосигналов.

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.

Американский национальный институт стандартов (*American National Standards Institute, ANSI*). Эта организация представляет США в Международной

организации по *стандартизации* ISO. Комитеты ANSI занимаются разработкой *стандартов* в различных областях вычислительной техники.

Так, комитет ANSI X3T9.5 совместно с компанией IBM осуществляет *стандартизацию* локальных сетей крупных ЭВМ (архитектура сетей SNA). Известный *стандарт* FDDI также является результатом деятельности этого комитета ANSI. В области микрокомпьютеров ANSI разрабатывает *стандарты* на языки программирования, интерфейс SCSI. ANSI разработал рекомендации по переносимости для языков C, FORTRAN, COBOL.

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**.

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.

Ассоциация производителей компьютеров и оргтехники (*Computer and Business Equipment Manufacturers Association, CBEMA*) — организация американских производителей аппаратного обеспечения; аналогична европейской ассоциации ECMA.

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.

Управление и стандарты Internet



Internet Society (ISOC) — профессиональное сообщество, которое занимается общими вопросами эволюции и роста Internet как глобальной коммуникационной инфраструктуры.

Под управлением ISOC работает **Internet Architecture Board (IAB)** — организация, в ведении которой находится технический контроль и координация работ для Internet. IAB координирует направление исследований и новых разработок для стека *TCP/IP* и является конечной инстанцией при определении новых *стандартов* Internet.

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

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

Список утвержденных официальных *стандартов* Internet публикуется в виде документа RFC и доступен в Internet, например по адресу http://www.internic.net/

Стандартные стеки коммуникационных протоколов

Важнейшим направлением *стандартизации* в области вычислительных сетей является *стандартизация* коммуникационных протоколов.

В сетях используется несколько стеков (наборов) коммуникационных протоколов.

Наиболее популярны следующие стеки:

- *TCP/IP (доминирует в последние годы)*;
- *IPX/SPX (доминировал до середины 90-х годов)*;
- *NetBIOS/SMB*;
- *DECnet*;
- *SNA*;
- *OSI* .

Все стеки, перечисленные выше, кроме *SNA*, на нижних уровнях — физическом и канальном, — используют одни и те же хорошо *стандартизованные* протоколы Ethernet, Token Ring, FDDI. Зато на верхних уровнях все стеки работают по своим 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*. Если модель OSI является концептуальной схемой взаимодействия *открытых систем*, то 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. 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* отличаются сложностью и неоднозначностью *спецификаций*, требуют больших затрат вычислительной мощности центрального процессора, что делает их наиболее подходящими для мощных машин, а не для сетей персональных компьютеров.

Стек ***OSI*** — независимый от производителей международный *стандарт*. Его поддерживает правительство США в своей программе GOSIP, в соответствии с которой все компьютерные сети, устанавливаемые в правительственных учреждениях США после 1990 года, должны или непосредственно поддерживать стек *OSI*, или обеспечивать средства для перехода на этот стек в будущем.

Стек IPX/SPX

Является оригинальным стеком протоколов фирмы Novell, разработанным для сетевой операционной системы NetWare в начале 80-х гг. Протоколы сетевого

и сеансового уровней Internetwork Packet Exchange (IPX и Sequenced Packet Exchange, SPX), которые дали название стеку, являются прямой адаптацией протоколов XNS фирмы Xerox, распространенных в гораздо меньшей степени, чем стек *IPX/SPX*.

Популярность стека *IPX/SPX* связана с операционной системой Novell NetWare, которая долгое время сохраняла мировое лидерство по числу установленных систем, хотя в последнее время ее популярность снизилась, и по темпам роста она заметно отстает от ОС Microsoft Windows.

Сейчас стек *IPX/SPX* реализован не только в NetWare, но и в нескольких других популярных сетевых ОС, например SCO UNIX, Sun Solaris, Microsoft Windows.

Стек NetBIOS/SMB

Применяется в продуктах компаний IBM и Microsoft. На физическом и канальном уровнях используются 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. Затем он был заменен протоколом расширенного пользовательского интерфейса NetBEUI — NetBIOS Extended User Interface. Протокол NetBEUI разрабатывался для сетей, насчитывающих не более 200 рабочих станций.

Протокол NetBEUI выполняет функций, которые можно отнести к сетевому, транспортному и сеансовому уровням модели OSI, однако он не обеспечивает возможность маршрутизации пакетов. Это ограничивает применение протокола NetBEUI локальными сетями, не разделенными на подсети, и делает невозможным его использование в составных сетях.

Соответствие популярных стеков протоколов модели OSI

<table>
<tr><th>Модель OSI</th><th colspan="3">IBM/Microsoft</th><th>TCP/IP</th><th>Novell</th><th>Стек OSI</th></tr>
<tr><td>Прикладной</td><td rowspan="3">SMB</td><td rowspan="3">Telnet,
FTP,
SNMP,
SMTP,
WWW</td><td rowspan="3">NCP,
SAP</td><td>X.400,
X.500,
FTAM</td></tr>
<tr><td>Представления</td><td>Протокол уровня представления OSI</td></tr>
<tr><td>Сеансовый</td><td>Сеансовый протокол OSI</td></tr>
<tr><td>Транспортный</td><td>NetBIOS</td><td>TCP</td><td>SPX</td><td>Транспортный протокол OSI</td></tr>
<tr><td>Сетевой</td><td colspan="2">IP,
RIP,
OSPF</td><td>IPX,
RIP,
NLSP</td><td>ES-ES,
IS-IS</td></tr>
<tr><td>Канальный</td><td colspan="4">802.3 (Ethernet), 802.5 (Token Ring), FDDI, Fast Ethernet, SLIP, 100VG-AnyLAN, X.25, ATM, LAP-B, LAP-D, PPP</td></tr>
<tr><td>Физический</td><td colspan="4">Коаксиал, экранированная и неэкранированная витая пара, оптоволокно, радиоволны</td></tr>
</table>

Стек TCP/IP

Стек *TCP/IP* был разработан по инициативе Министерства обороны США для связи экспериментальной сети ARPAnet с другими сетями как набор общих протоколов для разнородной вычислительной среды.

Специалисты из университета Беркли, реализовавшие протоколы стека *TCP/IP* в своей версии ОС UNIX способствовали его широкому распространению.

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, которые в терминологии модели OSI относятся к сетевому и транспортному уровням, соответственно. IP обеспечивает продвижение пакета по составной сети, а TCP гарантирует надежность его доставки.

Существует большое количество локальных, корпоративных и территориальных сетей, непосредственно не являющихся частями Internet, в которых также используются протоколы *TCP/IP*. Чтобы отличать эти сети от Internet, их называют сетями *TCP/IP* или просто ***IP-сетями***.

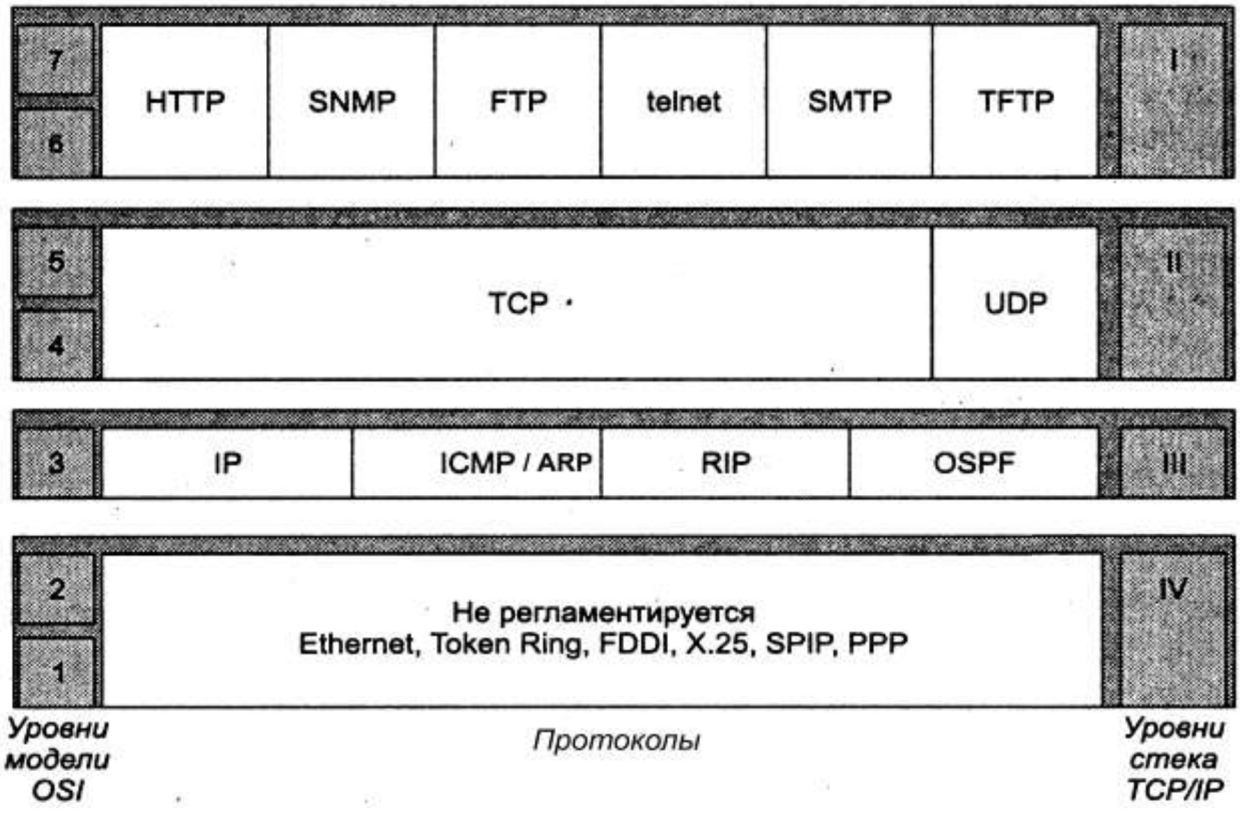
Достоинства стека *TCP/IP:*

- Способность эффективно фрагментировать пакеты.
- Гибкая система адресации, важная для построения больших гетерогенных сетей.
- Экономно используются возможности широковещательных рассылок.

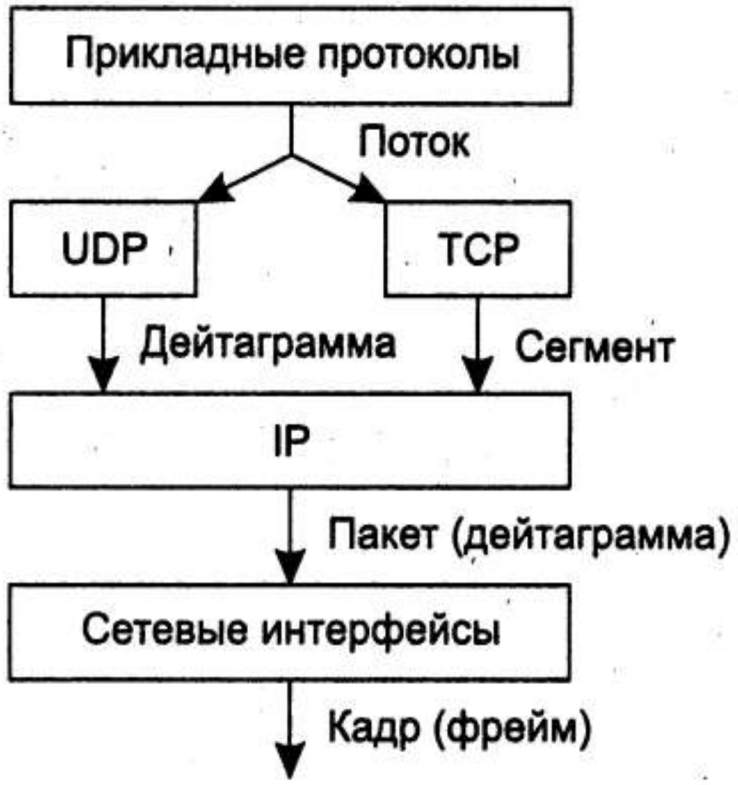
Однако платой за преимущества здесь оказываются высокие требования к ресурсам и сложность администрирования IP-сетей. Для реализации мощных функциональных возможностей протоколов стека *TCP/IP* требуются большие вычислительные затраты.

Лидирующая роль стека TCP/IP объясняется следующими его свойствами:

- Это наиболее завершенный стандартный и популярный стек сетевых протоколов.
- Почти все большие сети передают основную часть своего трафика с помощью протокола TCP/IP.
- Это метод получения доступа к сети Internet.
- Этот стек служит основой для создания intranet- корпоративной сети, использующей транспортные услуги Internet и гипертекстовую технологию WWW, разработанную в Internet.
- Все современные ОС поддерживают стек TCP/IP.
- Это гибкая технология для соединения разнородных систем как на уровне транспортных подсистем, так и на уровне прикладных сервисов.
- Это устойчивая масштабируемая межплатформенная среда для приложений клиент-сервер.



Архитектура стека протоколов TCP/IP и его связь с моделью ISO/OSI

Названия PDU в TCP/IP	№	Уровни стека протоколов TCP/IP	
	I	Прикладной	Application
	II	Транспортный	Transport
	III	Сетевой	Internet
	IV	Сетевых интерфейсов	Network access

Тема лекции:

Технологии физического уровня. Характеристики линий связи

План лекции:

1. Линии и каналы связи. Первичные сети. Физическая среда передачи данных.
2. Амплитудно-частотная характеристика.
3. Полоса пропускания.
4. Затухание.
5. Пропускная способность.
6. Помехоустойчивость. Перекрестные наводки на ближнем (дальнем) конце линии. Защищенность линии.
7. Достоверность передачи данных.

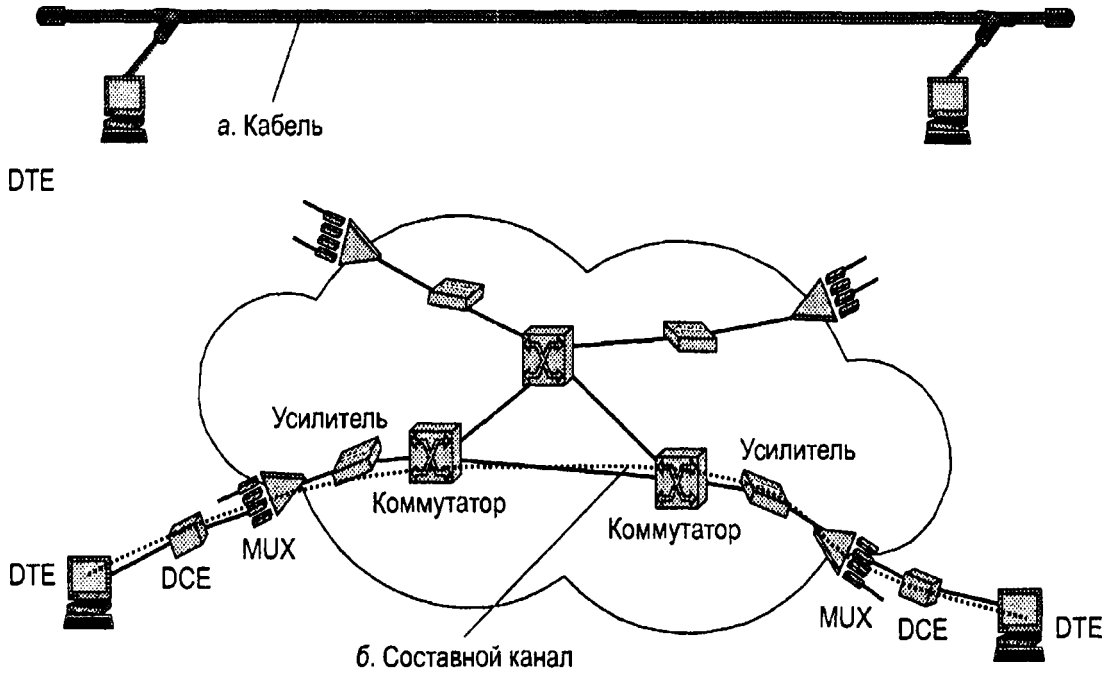
Первичные сети, линии и каналы связи

Канал (channel) – часть пропускной способности звена, используемая независимо при коммутации.

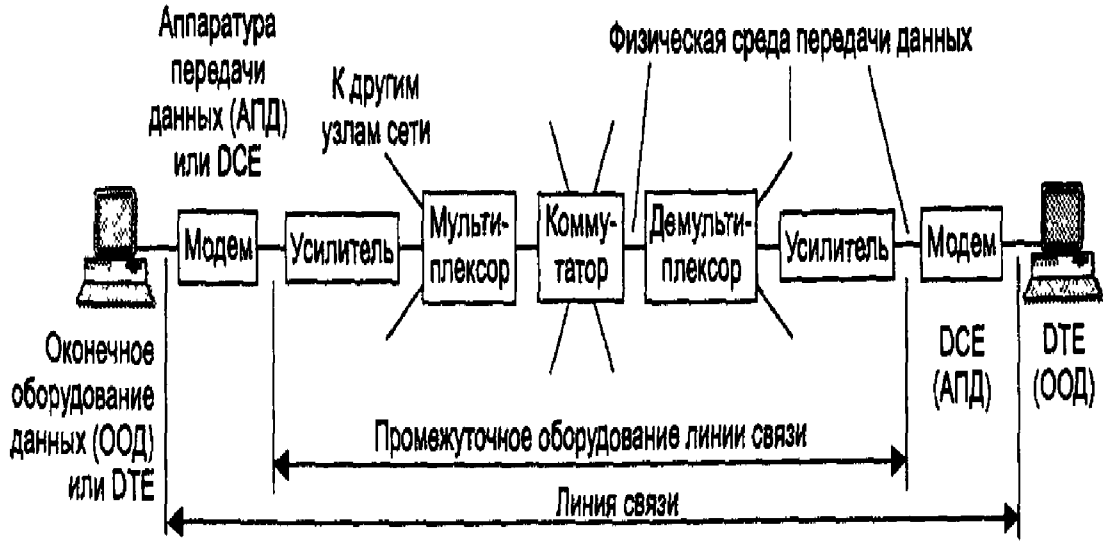
Составной канал (circuit) – путь между двумя конечными узлами сети (образуется отдельными каналами промежуточных звеньев и внутренними соединениями в коммутаторах).

Линия связи может использоваться как синоним для любого из двух описанных выше терминов.

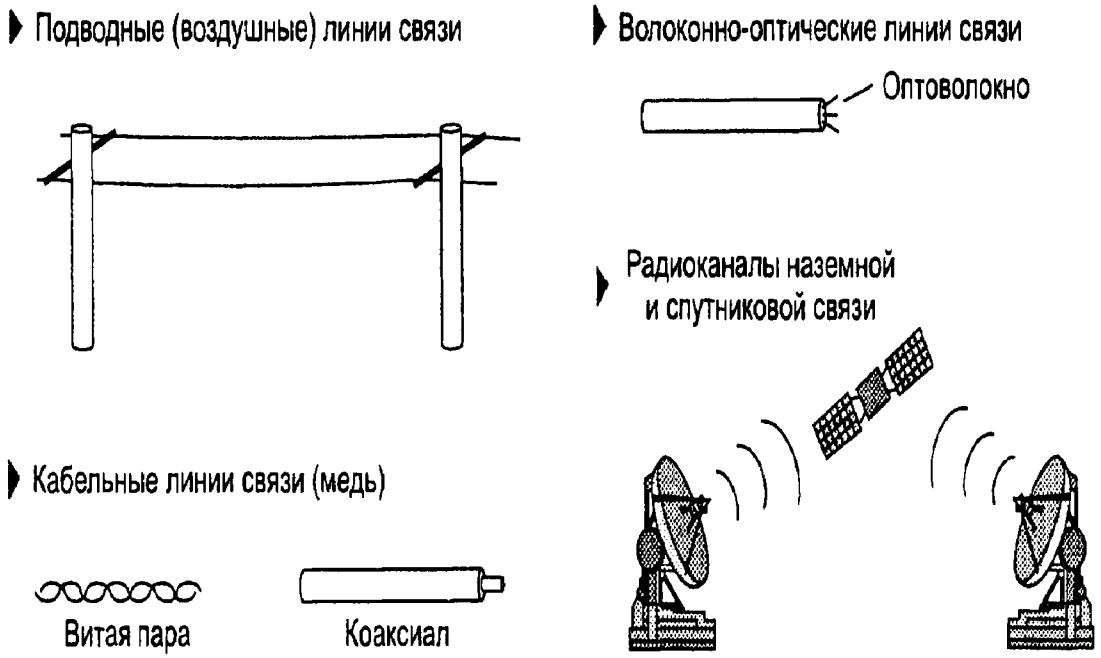
Первичные сети специально создают для того, чтобы предоставлять услуги каналов передачи данных для компьютерных и телефонных сетей, которые работают ***«поверх»*** первичных сетей и являются **наложенными сетями**.



Канал, составной канал, линия связи

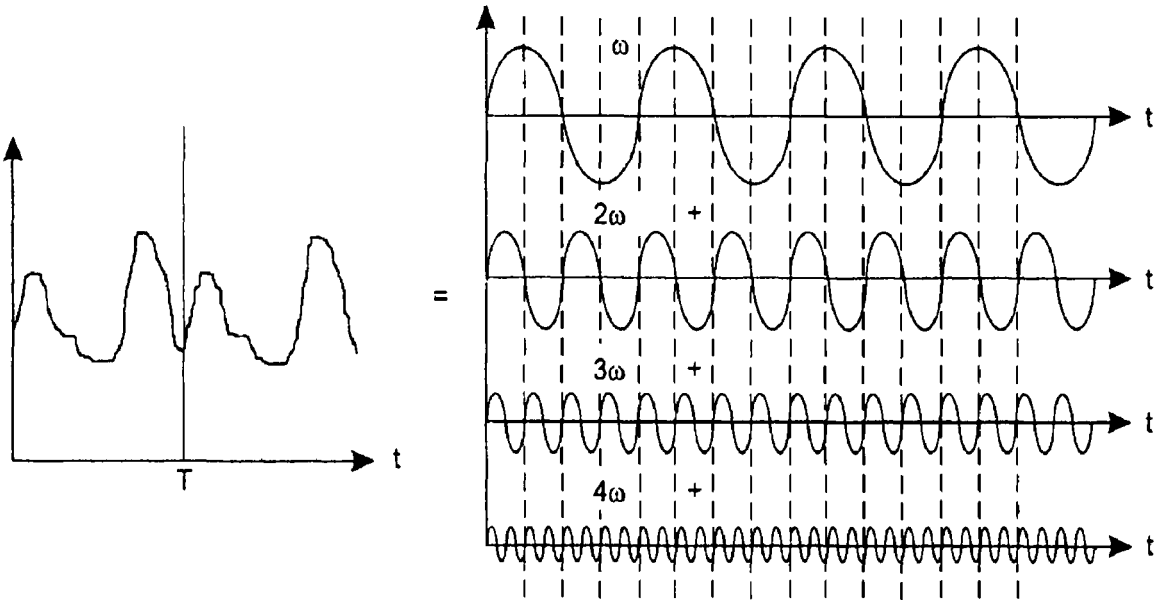


Оборудование линий связи

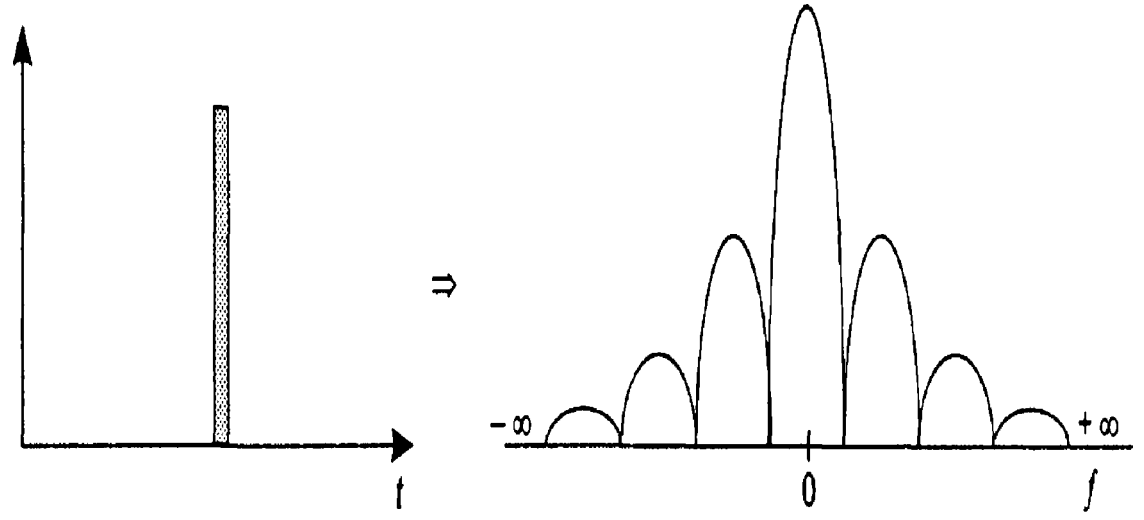


Физическая среда передачи данных

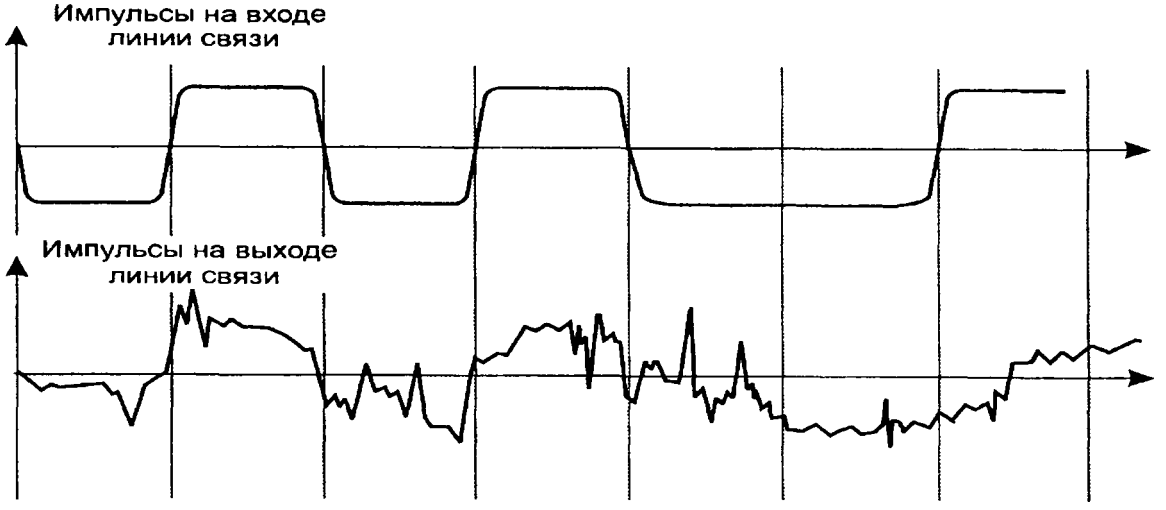
Для **определения** характеристик линий связи часто используют **анализ ее реакций не некоторые эталонные воздействия (синусоида, импульс, скачок).**



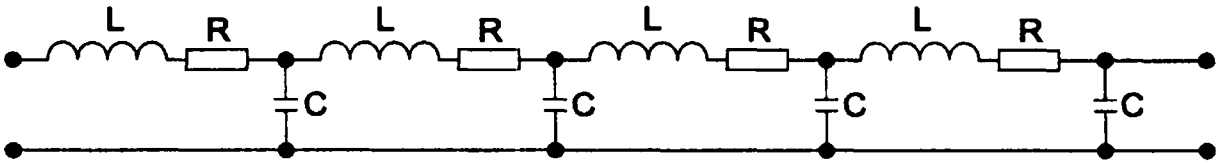
Представление периодического сигнала суммой синусоид



Спектральное разложение идеального импульса

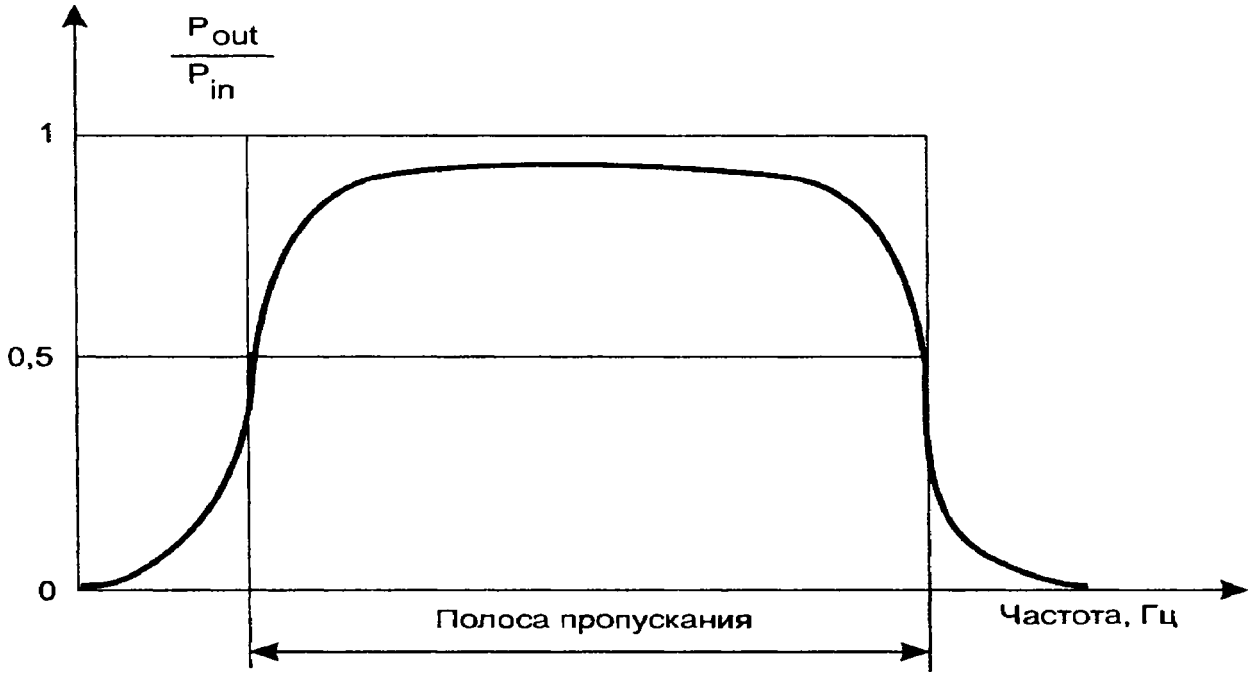


Искажение импульсов в линии связи



Представление линии как распределенной индуктивно-емкостной нагрузки

Амплитудно-частотная характеристика, полоса пропускания и затухание



Амплитудно-частотная характеристика, полоса пропускания и затухание. Зависимость затухания от частоты

Амплитудно-частотная характеристика (АЧХ) показывает как затухает амплитуда синусоиды на выходе линии связи по сравнению с амплитудой на ее входе (вместо амплитуды сигала может использоваться его мощность).

Зная АЧХ реальной линии связи можно определить форму выходного сигнала для любого входного сигнала. Но сделать это с достаточной точностью трудно, поэтому на практике вместо АЧХ применяются другие, упрощенные характеристики: **полоса пропускания** и **затухание.**

Полоса пропускания (bandwidth) – это непрерывный диапазон частот, для которого отношение амплитуды выходного сигнала ко входному превышает некоторый, заранее заданный предел, обычно 0,5. То есть полоса пропускания определяет диапазон частот синусоидального сигнала, в котором сигнал передается по линии связи без значительных искажений.

Ширина полосы пропускания в наибольшей степени влияет на максимальную возможную скорость передачи информации по линии связи.

Затухание (attenuation) определяется как ***относительное уменьшение амплитуды*** или ***мощности*** сигнала при передаче по линии связи сигнала

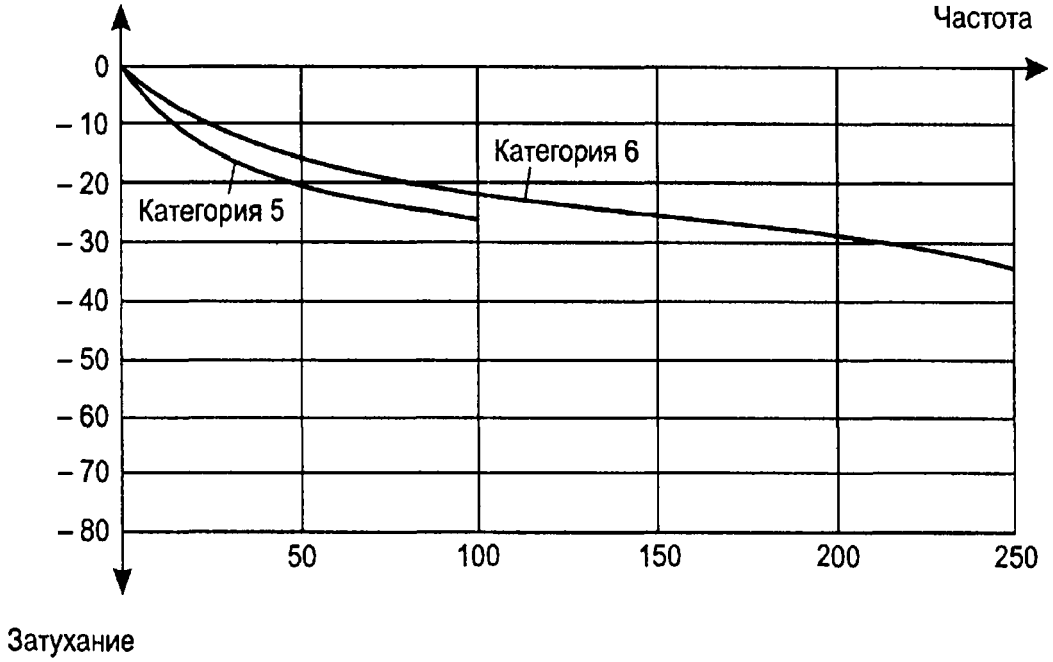
определенной частоты.

Затухание А обычно измеряется в децибелах (дБ, dB) и вычисляется по следующим формулам:

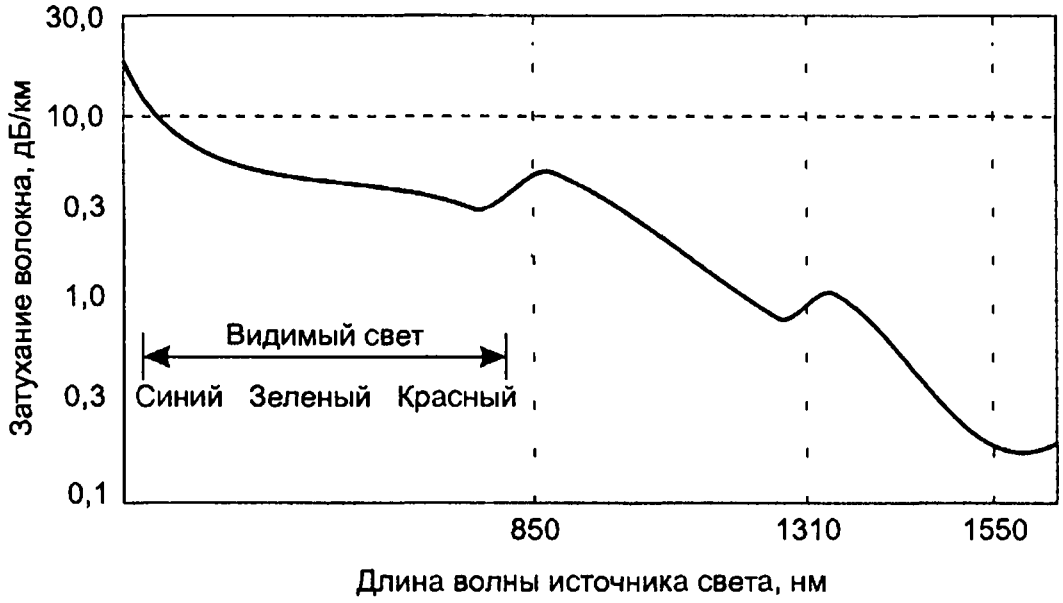
$$A = 10 \cdot \lg \frac{P_{out}}{P_{in}} \qquad A = 20 \cdot \lg \frac{U_{out}}{U_{in}}$$

P_{out} (U_{out})- мощность (амплитуда) сигнала на выходе линии;

P_{in} (U_{in})- мощность (амплитуда) сигнала на входе линии.



Затухание неэкранированного кабеля на витой паре

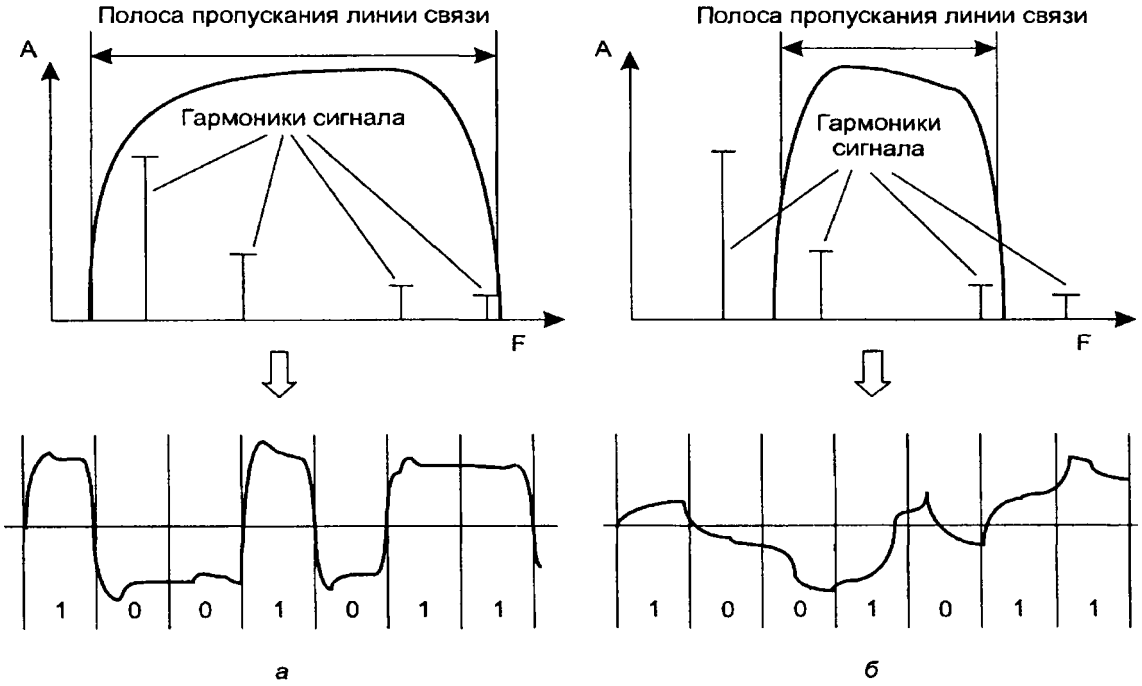


Затухание оптического сигнала. Окна прозрачности оптического волокна

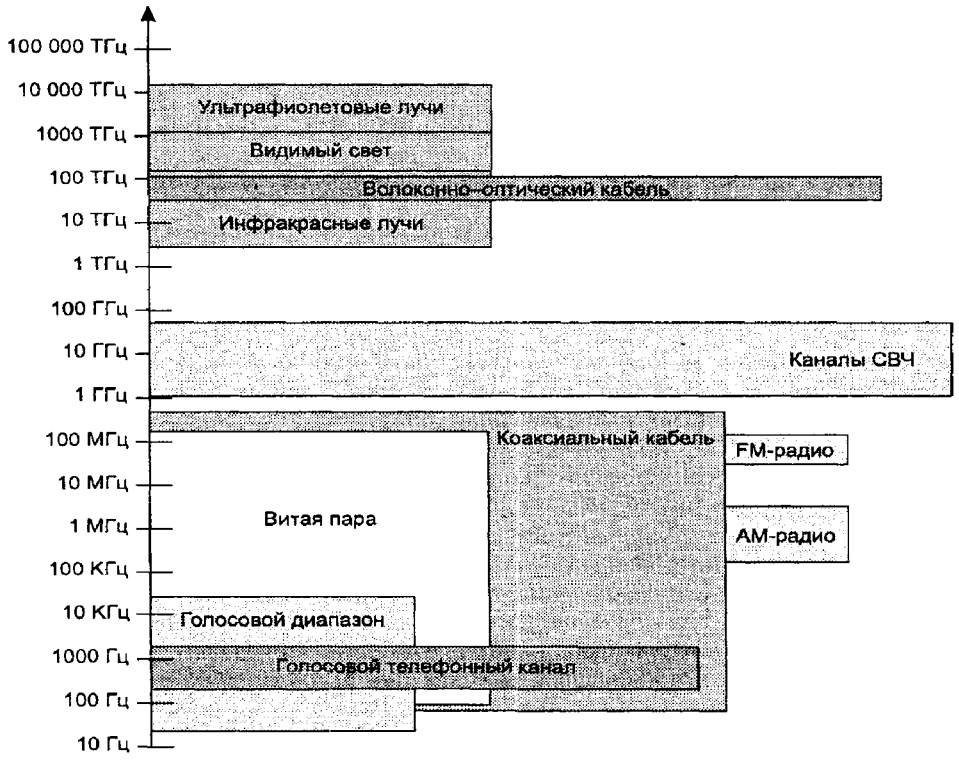
Пропускная способность (throughput) линии (канала связи) характеризует **максимально возможную скорость передачи данных по этой линии связи**.

Измеряется в битах в секунду (бит/с) и производных единицах (Кбит/с, Мбит/с, Гбит/с) кратных степеням 10 (а не 2).

Этот параметр зависит и от **параметров физической среды** и от **способа передачи данных**, то есть нельзя говорить с точностью о пропускной способности пока не определен **протокол физического уровня**.



Связь полосы пропускания, пропускной способности и спектра сигнала



Полосы пропускания линий связи и частотные диапазоны спектра электромагнитного излучения

Физическое и логическое кодирование

Выбор способа представления дискретной информации в виде сигналов, подаваемых на линию связи, называется **физическим**, или **линейным кодированием**. От выбранного способа кодирования зависит спектр передаваемого сигнала и, соответственно пропускная способность линии связи.

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.

Процесс изменения параметров несущего сигнала в соответствии с передаваемой информацией называется **модуляцией**.

Количество изменений информационного параметра несущего периодического сигнала в секунду измеряется в **бодах (baud)**. Период времени между соседними изменения информационного сигнала называется **тактом** работы передатчика.

Пропускная способность линии в битах в секунду в общем случае не совпадает с числом бод (может быть выше или ниже и зависит от способа кодирования).

Если сигнал изменяется так, что можно различить только два его состояния, то любое его изменение будет соответствовать наименьшей единице информации – биту. Если же сигнал имеет **более двух** различимых состояний, то любое его изменение будет **нести несколько битов** информации (то есть пропускная способность в битах в секунду будет выше, чем число бод).

Логическое кодирование, выполняемое до физического для повышения помехозащищенности или шифрования, обычно наоборот уменьшает пропускную способность в битах в секунду.

Чем выше частота несущего периодического сигнала, тем выше может быть частота модуляции и тем выше может быть пропускная способность линии связи. Однако с увеличением частоты периодического несущего сигнала увеличивается и ширина спектра этого сигнала.

Линия передает сигнал с таким спектром 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 возможная скорость передачи информации оказывается меньше.

Соотношение полосы пропускания и пропускной способности

Связь между полосой пропускания линии и ее ***максимально возможной пропускной способностью***, вне зависимости от принятого способа физического кодирования, установил **Клод Шэннон**:

$C=F\cdot\log_2(1+P_c/P_ш)$, где

C – макс. пропускная способность линии в бит/с,

F – ширина полосы пропускания линии в герцах,

P_c – мощность сигнала, $P_ш$ – мощность шума (помехи).

Близким по сути является соотношение, полученное **Найквистом**, которое также определяет максимально возможную пропускную способность линии связи (без учета шума в линии):

$C=2F \cdot \log_2 M$, где

M – количество различимых состояний информационного параметра.

Помехоустойчивость линии определяет способность линии **противостоять влиянию помех** (и даже их уменьшать), создаваемых во внешней среде или на внутренних проводниках самого кабеля.

Наименее помехоустойчивыми являются радиолинии, **хорошей** устойчивостью обладают кабельные линии и **отличной** – волоконно-оптические.

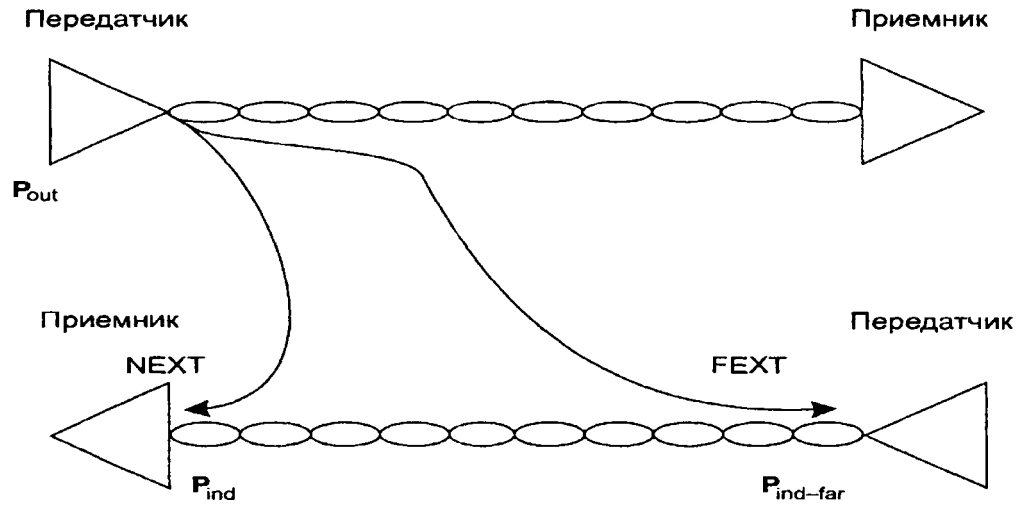
Для **уменьшения влияния помех** (наводок) от внешних (и внутренних) электромагнитных полей проводники **экранируют** и/или **скручивают**.

Перекрестные наводки на ближнем конце (Near End Cross Talk, NEXT) определяют помехоустойчивость кабеля, когда электромагнитное поле сигнала, передаваемого выходом передатчика по одной паре проводников, наводит на другую пару сигнал помехи:

$$NEXT = 10 \cdot \lg(P_{out} / P_{ind}) \quad (дБ).$$

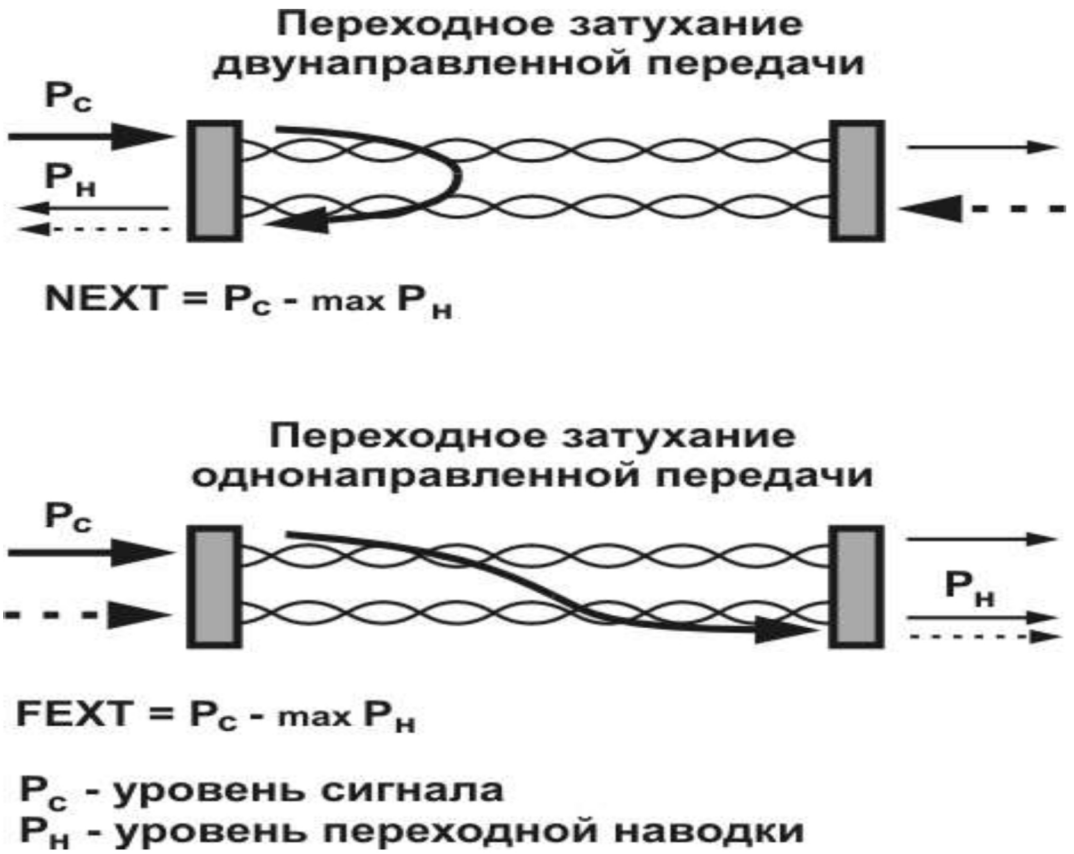
Перекрестные наводки на дальнем конце (Far End Cross Talk, FEXT) определяют помехоустойчивость кабеля, когда электромагнитное поле сигнала, передаваемого выходом передатчика по одной паре проводников, наводит на другую пару сигнал помехи, измеряемый на другом конце кабеля:

$$FEXT = 10 \cdot \lg(P_{out} / P_{ind\text{-}far}) \quad (дБ).$$



$P_{ind–far}$ – мощность наведенного сигнала на дальнем конце кабеля

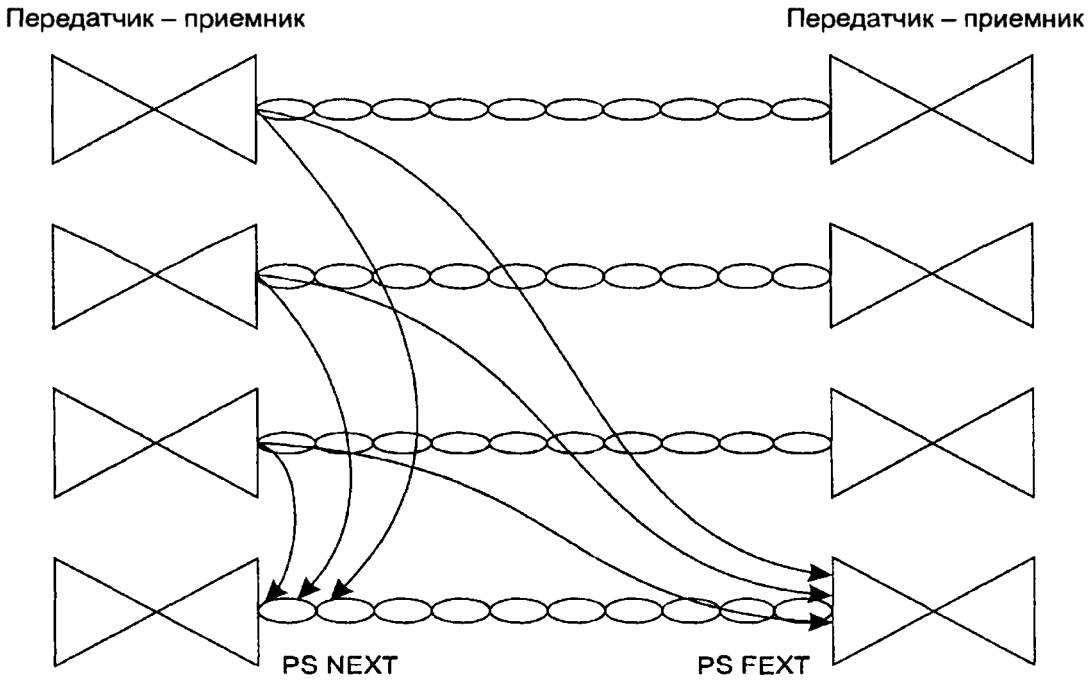
Перекрестные наводки на ближнем (NEXT) / дальнем конце (FEXT)



Перекрестные наводки на ближнем (NEXT) / дальнем конце (FEXT)

В современных технологиях данные передаются одновременно по нескольким витым парам, поэтому применяют показатели перекрестных наводок с приставкой **PS** (**PowerSUM** – суммарная наводка): **PS NEXT** и **PS FEXT.**

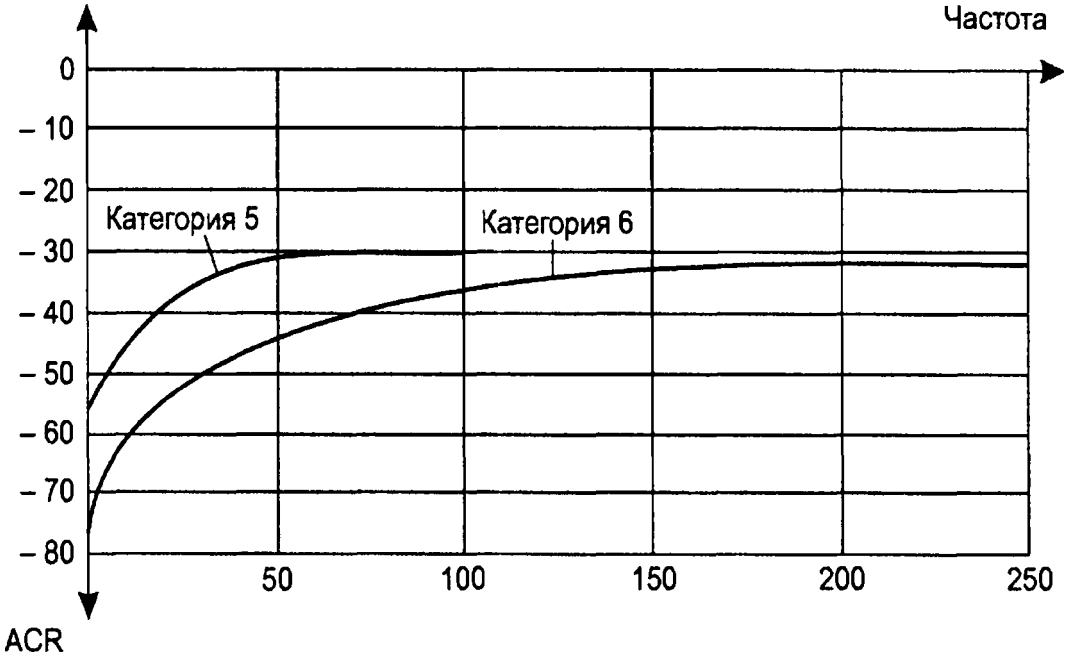
Эти показатели отражают устойчивость кабеля к суммарной мощности перекрестных наводок на одну из пар кабеля от всех остальных передающих пар.



Суммарное переходное затухание PS NEXT и PS FEXT

Другой важной характеристикой является **защищенность кабеля (Attenuation/Crosstalk Ratio, ACR)**, которая 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.



Защищенность (ACR) для витой пары

Несложно вывести критерии, напрямую показывающие соотношение сигнал/шум (а значит, и качество линии) в логарифмическом виде. В кабельных системах для этого используется следующая пара параметров:

ACR (attenuation to crosstalk ratio), дословно переводится как "отношение затухания к наводкам",

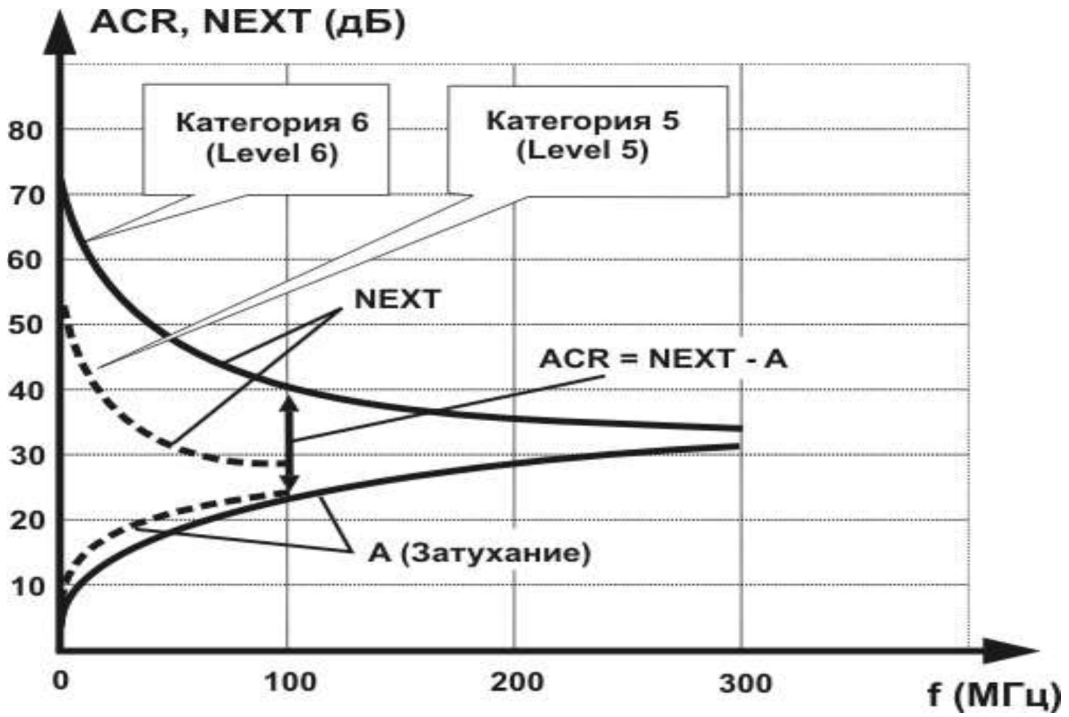
и **ELFEXT (equal level far end crosstalk)** - "равноуровневые наводки на дальнем конце".

Эти параметры не определяются путем измерений, а рассчитываются по следующим формулам:

ACR = NEXT - A, ELFEXT = FEXT - A.

Физический смысл этих параметров достаточно прост: **ACR** - это **превышение сигнала над уровнем собственных шумов** при **двунаправленной** передачи сигналов, а **ELFEXT - однонаправленной**.

Так как основным видом помех в кабелях компьютерных сетей являются наводки, то использование параметра ACR позволяет однозначно определить верхнюю границу частоты электрического тракта передачи (либо любой его части). Считается, что среда передачи может обеспечить устойчивую полнодуплексную работу любого приложения с такой граничной частотой, на которой параметр ACR составляет 10 дБ.



Защищенность (ACR) для витой пары

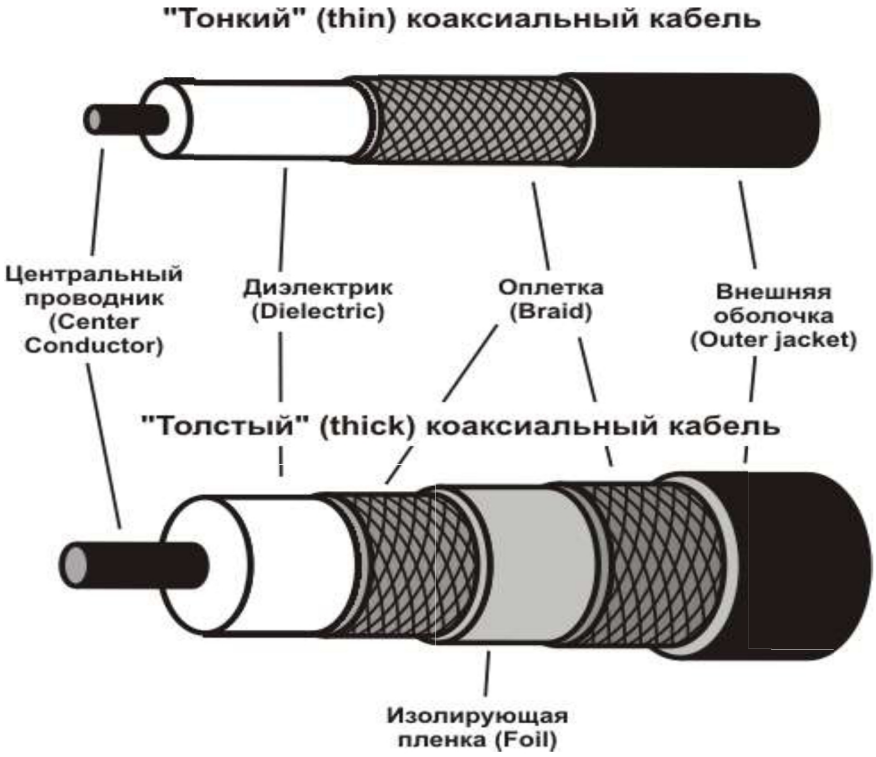
Достоверность передачи данных характеризует вероятность искажения для каждого передаваемого бита данных. Иногда этот показатель называют **интенсивностью битовых ошибок** (**Bit Error Rate, BER**).

Величина BER для линий 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}.

Тема лекции: **Технологии физического уровня. Типы, устройство и стандарты кабелей. Структурированная кабельная система (СКС)**

План лекции:

1. Коаксиальный кабель (медь, copper media).
2. Экранированная и неэкранированная витая пара (медь, copper media).
3. Волоконно-оптический кабель (fiber).
4. Структурированная кабельная система (СКС).



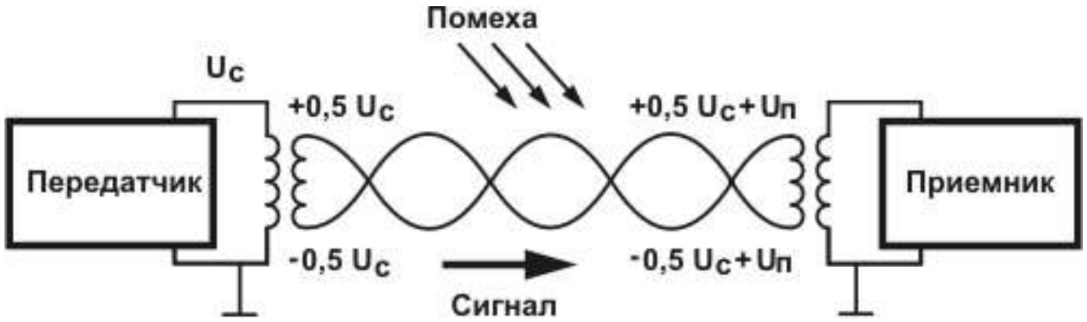
Конструкция коаксиального кабеля

Витая пара. Физические принципы.

Витая пара - это изолированные проводники, попарно свитые между собой некоторое число раз на определенном отрезке длины, что требуется для уменьшения перекрестных наводок между проводниками.

Такие линии как нельзя лучше подходят для создания симметричных цепей, в которых используется балансный принцип передачи информации.

Приемник и передатчик гальванически развязаны друг от друга согласующими трансформаторами. При этом во вторичные обмотки (сетевые адаптеры) подается только разность потенциалов первичной обмотки (непосредственно протяженной линии).



U_C - напряжение полезного сигнала
$U_П$ - напряжение помехи

Токи в любой точке идеальной витой пары равны по значению, и противоположны по направлению. Следовательно, векторы напряженности электромагнитного поля каждого из проводников противоположно направлены, и суммарное ЭМИ отсутствует. Под идеальной витой парой понимается линия, в которой проводники бесконечно плотно прилегают друг к другу, имеют бесконечно малый диаметр, и ток, протекающий через них, стремится к нулю.

Витая пара накладывает **ограничения** на протокол передачи (**невозможность передачи постоянной составляющей**), но значительно более устойчиво к внешним влияниям (по сравнению, например, с несимметричным RS-232). Из рисунка видно, что результирующее напряжение наводки на вторичной обмотке будет синфазным, соответственно не передастся на вторичную обмотку (сетевой адаптер).

Витая пара. История развития

Витая пара десятки лет успешно использовалась в телефонии, ее перенос на почву Ethernet произошел только сентябре 1990 года, когда был официально принят стандарт 10baseT. Это была витая пара 3 категории, с очень большим, в десятки сантиметров, шагом скрутки проводов в паре, и небольшой, до 20 МГц, полосой пропускания. Компьютерные кабели отличало только оформление - 4 пары под одной оболочкой.

Немного позже (с появлением Fast Ethernet в 1995 году), был введен новый стандарт на кабель Категории 5 (Level 5), с шагом скрутки, меняющемся для разных пар от 12 до 32 мм (например, ряд от Lucent - 15, 13, 20, 24 мм). Делается это для уменьшения перекрестных наводок (NEXT, FEXT) . Такой кабель обеспечивает передачу сигналов с частотой до 100 МГц. Далее появились: Категория 5e (до 125 МГц), Категория 6 (до 250 МГц) и Категория 7 (до 600 МГц).

Витая пара. Типы и особенности конструкции

Проводники изготовлены из монолитной медной проволоки толщиной 0,5 - 0,65 мм. Кроме метрической, применяется система AWG, в которой эти величины составляют 24 или 22 соответственно.

Толщина изоляции - около 0,2 мм, материал обычно поливинилхлорид

(английское сокращение PVC), для более качественных образцов 5 категории - полипропилен (PP), полиэтилен (PE). Особенно высококлассные кабели имеют изоляцию из вспененного (ячеистого) полиэтилена, которые обеспечивают низкие диэлектрические потери, или тефлона, который обеспечивает уникальный рабочий диапазон температур.

Разрывная нить (обычно капрон) используется для облегчения разделки внешней оболочки - при вытягивании она делает на оболочке продольный разрез, который открывает доступ к кабельному сердечнику, гарантированно не повреждая изоляцию проводников.

Внешняя оболочка имеет толщину 0,5-0,6 мм, и обычно изготавливается из привычного поливинилхлорида с добавлением мела, который повышает хрупкость. Это необходимо для точного облома по месту надреза лезвием отрезного инструмента. Кроме этого, начинают применяться так называемые "молодые полимеры", которые не поддерживают горения, и не выделяют при нагреве галогенов.

По наличию (или отсутствию) экрана, различают несколько типов кабелей:

1. **UTP (unshielded twisted pair),** что означает незащищенная витая пара (НЗВП), то есть кабель, витые пары которого не имеют индивидуального экранирования;
2. **FTP (Foiled Twisted Pair)** - фольгированная витая пара. Имеет общий экран из фольги, однако у каждой пары нет индивидуальной защиты;
3. **STP (shielded twisted pair)** - защищенная витая пара (ЗВП), каждая пара имеет экран;
4. **ScTP (Screened Twisted Pair)** - экранированный кабель, который может как иметь, так и не иметь защиту отдельных пар.

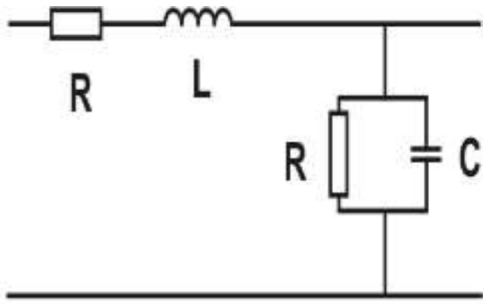
Экран выполняется либо плетеным из медной проволоки (хорошая защищает от низкочастотных наводок), либо из токопроводящей фольги (пленки), которая блокирует высокочастотное электромагнитное излучение. Так же на практике часто используют двойные экраны (HIGHT Screen), в которых используются оба способа.

Эффект от применения экрана на первый взгляд достаточно прост - уменьшение внешних наводок на экранированную пару (или несколько пар), и снижение уровня их электромагнитного излучения "наружу".

Но общий экран вызывает рост NEXT (перекрестных наводок, подробно рассмотренных ниже) из-за отражения от экрана, на 10-20%. Далее, экранирование увеличивает затухание в кабеле вследствие добавочной емкости между экраном и витыми парами. Но и это не все. Монтаж экранированной системы значительно более сложен (дорог), требует хорошего подбора всех элементов.

Витая пара. Электрические свойства

Электрические свойства витой пары характеризуются сопротивлением R, индуктивностью проводников L, емкостью C, и проводимостью изоляции G.



L - индуктивность
R - активное сопротивление
C - емкость

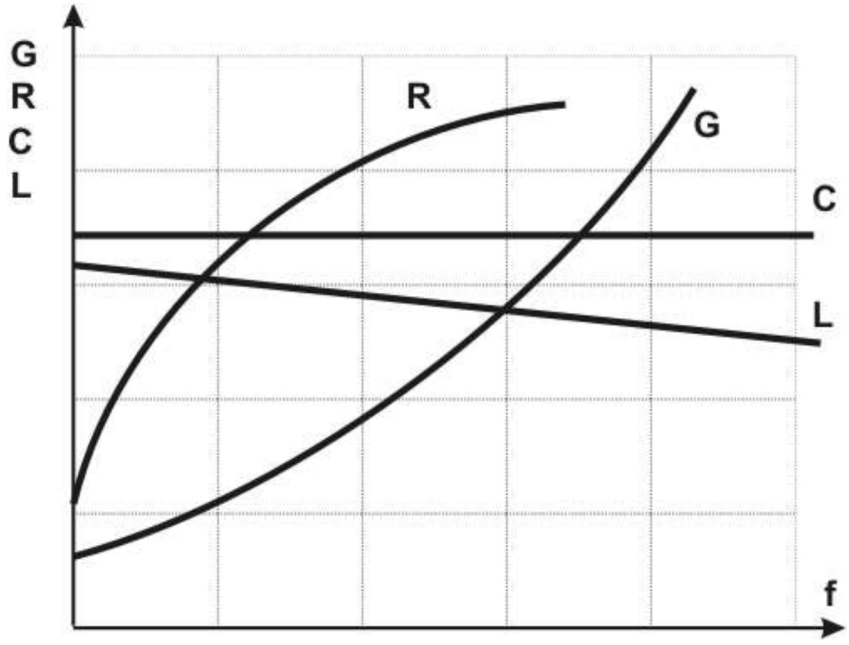
Величины R и G обуславливают тепловые потери в меди и диэлектрике соответственно. L и C определяют реактивность системы, или, иначе говоря, ее частотные свойства.

Активное сопротивление R постоянному току зависит от материала проводника, его геометрических размеров, и его температуры. По распространенному стандарту EIA/TIA-568A это значение не должно превышать 19,2 Ом на короткозамкнутом шлейфе длиной в 100 метров при температуре 20° C.

С увеличением частоты сигнала, активное сопротивление растет. Это обусловлено прохождением тока в основном по части, обращенной к другому проводнику (эффект близости). Вытеснение тока к поверхности проводника (скин-эффект) для проводов тоньше 0,8 мм мало заметен, но какое-то минимальное влияние на уменьшение эффективного сечения то же оказывает.

Проводимость изоляции G является мерой качества материала и его нанесения на поверхность отдельного проводника. Сопротивление току утечки связанное с несовершенством диэлектрика, может достигать нескольких ГигаОм, и на сегодня его можно не учитывать. Поэтому, в основном на проводимость изоляции влияют затраты на поляризацию диполей материала диэлектрика.

Особенно много их содержится в поливинилхлориде, часто используемом для витой пары низкой категории. В более качественных кабелях обычно используются полиэтилен или тефлон, рассеяние энергии в которых гораздо ниже. Еще ниже этот показатель для вспененных материалов, применяемых для кабелей высшего класса.



Индуктивность L можно разделить на внешнюю (определяемую геометрией и магнитными свойствами проводника), и внутреннюю (создаваемую магнитным полем протекающего тока). Внутренняя индуктивность имеет слабую тенденцию к уменьшению с ростом частоты.

Два проводника, составляющих пару, можно рассматривать как конденсатор, емкость которого, C, не зависит от частоты. Она определяется материалом изоляции, геометрическими размерами проводников, и расстоянием между ними. По стандарту, для современных кабелей, величина емкости составляет не более 5,6 нФ.

Особо нужно отметить, что применение экрана вызывает рост емкости примерно на 30%, что существенно снижает его эксплуатационные свойства такого кабеля.

На основании перечисленных электрических параметров, может быть рассчитано **волновое сопротивление**. Сделать это можно по формуле $Z = \sqrt{(R+jwL)/(G+jwC)}$, которую для высоких частот Ethernet можно упростить до $Z = \sqrt{L/C}$. **В рабочем диапазоне кабеля эта величина должна составлять 100 ± 15% Ом**.

Волновое сопротивление хорошо характеризует однородность тракта передачи электромагнитной энергии. Его неоднородности неизбежно вызывают отражения части сигнала, и ухудшение качества линии. Поэтом, достаточно очевидно, что все составляющие, включая сетевые адаптеры, должны иметь одинаковое волновое сопротивление, то есть должны быть согласованы.

Как правило, неоднородности волнового сопротивления на реальных коммуникациях являются следствием некачественного монтажа (изгиб, давление, растяжение, перекручивание).

Волоконно-оптический кабель. История

Передача информации через волновод из кварцевого стекла (SiO_2) применяется все более широко. Так, оптоволоконные кабели уже полностью вытеснили медные на магистральных каналах, и стремительно подбираются к конечному пользователю (Дом.ру – «оптика до дома»).

Физический принцип, лежащий в основе передачи сигналов по оптическому волокну прост и далеко не нов. Еще в 1870 г. в Лондонском Королевском обществе Дж. Тиндаль продемонстрировал непрямолинейное распространение света внутри струи жидкости, основанное на отражении света от границы сред (воздуха и воды).

Практическое применение этого эффекта стало возможно после нескольких принципиальных технологических "прорывов". Укажем два из них. В 1967 г. Жорес Алферов создал первые полупроводниковые гетеролазеры, работоспособные при комнатной температуре. Чуть позже, в 1970 г., на фирме "Корнинг" была получена первая миля сверхчистого кварцевого волокна, пригодного для оптической связи.

На основе этих технологий, в 1975 году было внедрено первое поколение передатчиков сигналов. Основу составлял светоизлучающий диод, работающий на длине волны 0,85 мкм в многомодовом режиме. Не прошло и трех лет, как появились одномодовые диодные лазеры на 1,3 мкм. А в 1982 году пошло в серию третье поколение, работающее на длине волны 1,55 мкм.

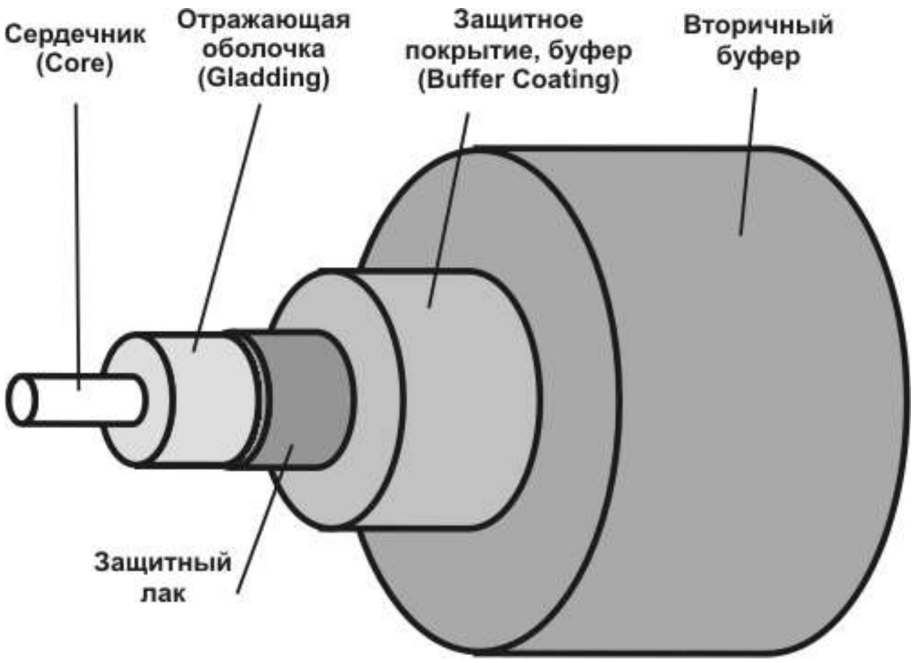
1988 год ознаменован вводом в действие первой трансатлантической ВОЛС ТАТ-8. В настоящее время близка к покорению другая часть рынка. Настольные системы, соединенные в локальную сеть при помощи оптоволокна, вполне реальны и лишь слегка экзотичны (пример: оптика на чипе).

Основными достоинствами оптоволокна является практически не ограниченная пропускная способность, индифферентность к электрическим (например, атмосферным) наводкам, и высокая долговечность. К недостаткам можно отнести относительно дорогой кабель и активное оборудование, и высокую сложность монтажа.

Волоконно-оптический кабель. Принцип работы

Источником распространяемого по оптическим кабелям света является светодиод (или полупроводниковый лазер), а кодирование информации осуществляется двухуровневым изменением интенсивности света (0-1). На другом конце кабеля принимающий детектор преобразует световые сигналы в электрические.

В используемых в настоящее время схемах луч от светодиода или лазера впускают в более плотную среду, ограниченную менее плотной. При правильном подборе материалов, происходит эффект полного отражения (преломление отсутствует). Таким образом, транспортируемый сигнал "идет" внутри замкнутой среды, проделывая путь от источника сигнала до его приемника.



Волоконно-оптический кабель. Устройство

Внешний диаметр отражающей оболочки унифицирован для всех типов кабелей и составляет 125±2 мкм. В этот размер входит и 2-3 мкм слой лака, который служит защитой от влаги и связанной с ней коррозии.

Первичную механическую прочность и гибкость рассматриваемой конструкции придает защитное покрытие из эпоксиакрилата, часто называемое буфером. Как правило, для удобства монтажа его окрашивают в разные цвета. Толщина покрытия составляет 250±15 мкм. Кроме этого, для лучшей защиты волокна и более удобного монтажа разъемов часто применяются конструкции с вторичным буфером диаметром 900 мкм, который без зазора уложен на первичный.

Волоконно-оптический кабель. Характеристики

Все распространенные типы волокон характеризуются двумя важнейшими параметрами: **затуханием и дисперсией**.

Дисперсия характеризует рассеяние во времени спектральных и модовых составляющих оптического сигнала. Существуют три типа дисперсии: межмодовая, материальная и межчастотная.

Все виды дисперсии, внося случайные задержки носителей сигнала, отрицательно влияют на пропускную способность оптоволоконного канала. Чем сильнее размыт по времени импульс на выходе (эффект дисперсии), тем сложнее его правильный прием. Иначе говоря, дисперсия накладывает ограничение на дальность передачи и на верхнюю частоту передаваемых сигналов.

Затухание характеризует потерю мощности передаваемого сигнала на заданном расстоянии, и измеряется в дБ/км. Волоконно-оптические линии как правило способны нормально функционировать при потерях в 30 дБ (прием всего 1/1000 мощности).

Есть два принципиально различных физических механизма, вызывающих

затухание:

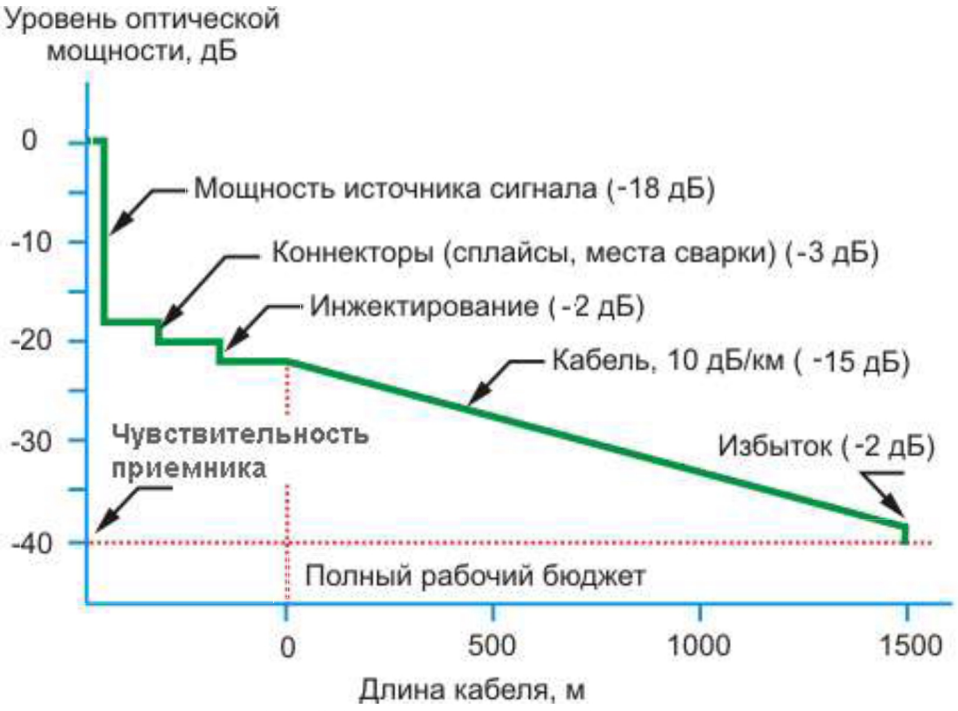
1. Потери на поглощение. Связаны с преобразованием одного вида энергии в другой. Электромагнитная волна определенной длины вызывает в некоторых химических элементах изменение орбит электронов, что, в свою очередь, ведет к нагреву волокна. Процесс поглощения волны тем меньше, чем меньше ее длина, и чем чище материал волокна.
2. Потери на рассеяние (выход части светового потока из волновода). Обусловлено это обычно неоднородностями показателя преломления материалов. Известно, что с уменьшением длины волны потери рассеивания возрастают.



В итоге образуются **три окна прозрачности**, в рамках которых затухание имеет наименьшее значение. Самые распространенные значения длины волны:

0,85 мкм; **1,3** мкм; **1,55** мкм.

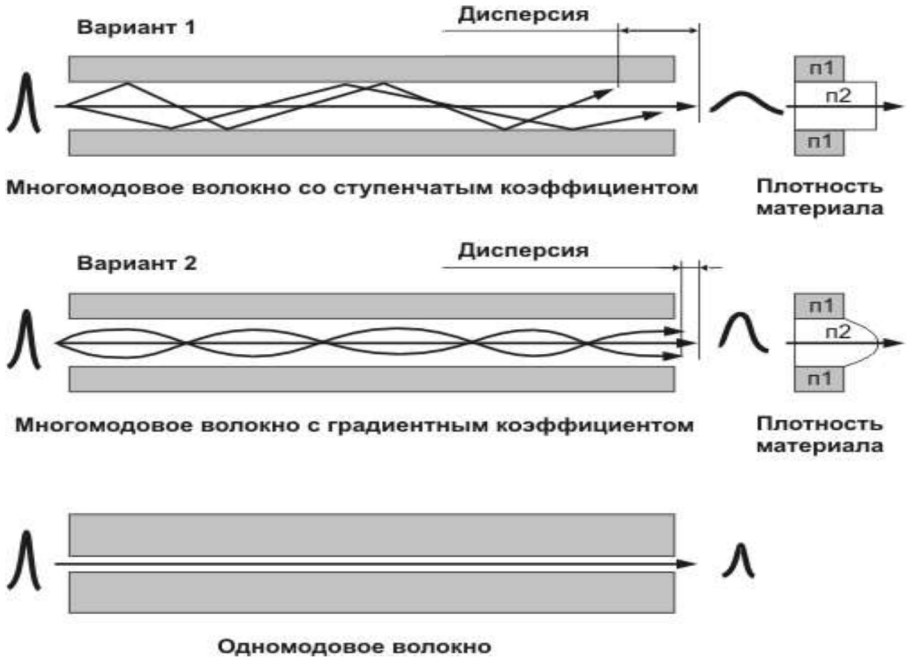
Именно под такие диапазоны разработаны специальные гетеролазеры, на которых основываются современные **ВОЛС** (волоконно-оптические линии связи). Влияние частоты сигнала на технологии сегодняшнего дня очень большое. Для примера, инфракрасный луч проходит в волокне с небольшим затуханием 10 км, красный свет (длина волны 0,65 мкм) пройдет лишь 0,5 км, а синий (0,43 мкм) и вообще меньше 50 м.



ВОЛС. Оптический бюджет

Волоконно-оптический кабель. Типы конструкции

По типу конструкции (по размеру серцевины) оптические волокна делятся на **одномодовые** (ОМ) и **многомодовые** (ММ).



В случае **многомодового** волокна диаметр сердечника (обычно 50 или 62,5 мкм) почти на два порядка больше, чем длина световой волны. Это означает, что свет может распространяться в волокне по нескольким независимым путям (модам). При этом очевидно, что разные моды имеют разную длину, и сигнал на приемнике будет заметно "размазан" по времени.

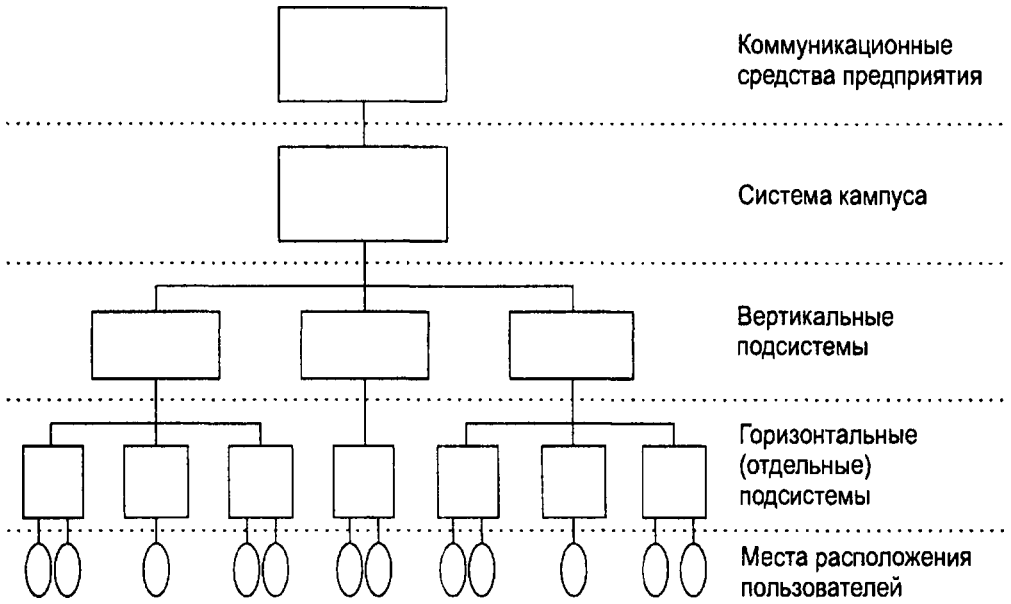
Одномодовое волокно с диаметром сердечника 8-9 мкм, что достаточно близко к обычно используемой длине волны 1,3 мкм. Межчастотная дисперсия при неидеальном источнике излучения остается, но ее влияние на передачу сигнала в сотни раз меньше, чем межмодовой или материальной. Соответственно, и пропускная способность одномодового кабеля намного больше, чем многомодового.

Параметры	Одномодовые	Многомодовые
Используемые длины волн	1,3 и 1,5 мкм	0,85 мкм, реже 1,3 мкм
Затухание, дБ/км.	0,4 - 0,5	1,0 - 3,0
Тип передатчика	лазер, реже светодиод	светодиод
Толщина сердечника.	8 – 9 мкм	50 или 62,5 мкм
Стоимость волокон и кабелей.	Около 70% от многомодового	-
Средняя стоимость конвертера в витую пару Fast Ethernet.	$300	$100
Дальность передачи Fast Ethernet.	около 20 км	до 2 км
Дальность передачи специально разработанных устройств Fast Ethernet.	более 100 км.	до 5 км
Возможная скорость передачи.	10 Гб/с, и более.	до 1 Гб/с. на ограниченной длине
Область применения.	телекоммуникации	локальные сети

Структурированная кабельная система (СКС)

Структурированная кабельная система (**Structured Cabling System, SCS**) здания – 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 компьютерных сетях.

Структура здания предопределяет структуру кабельной системы. СКС здания 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.



Иерархическая структура СКС

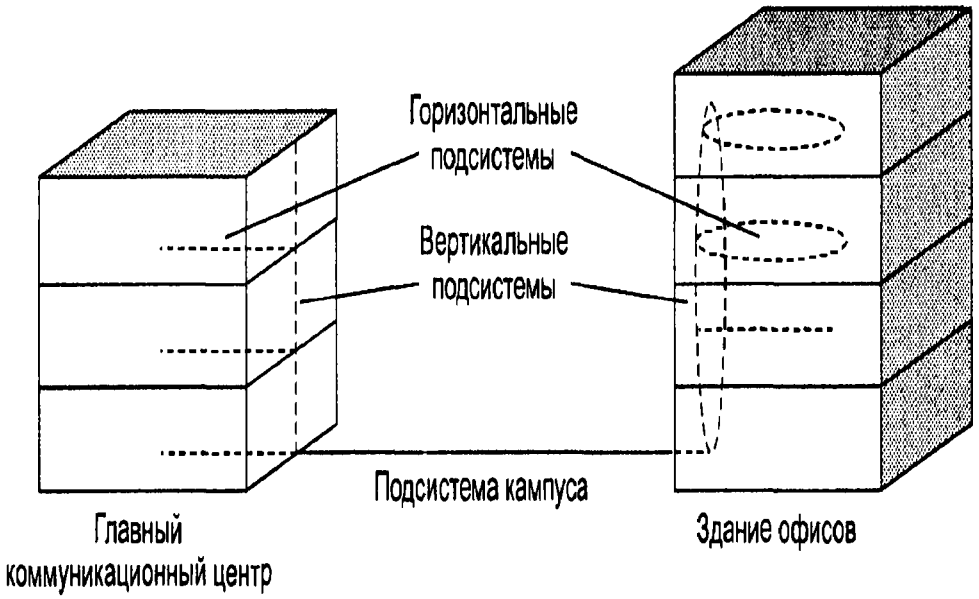
При необходимости конфигурацию связей можно легко изменить – добавить компьютер, сегмент, коммутатор, переместить или изъять оборудование.

Наиболее детально разработаны стандарты (гл. образом американские и международные) кабельных систем зданий, при этом иерархический подход к процессу создания такой кабельной системы позволяет назвать ее структурированной.

На основе СКС здания работает одна или несколько локальных сетей организаций или подразделений одной организации, размещенных в этом здании.

Типичная иерархическая СКС включает:

1. **горизонтальные подсистемы**, соответствующие этажам здания, - они соединяют кроссовые шкафы этажа с розетками пользователей;
2. **вертикальные подсистемы**, соединяющие кроссовые шкафы каждого этажа с центральной аппаратной здания;
3. **подсистему кампуса**, объединяющую несколько зданий с главной аппаратной всего кампуса; эта часть СКС часто называется **магистралью**.



Структурированная кабельная система (СКС)

СКС призвана стать **универсальной средой** для передачи компьютерных данных в локальной вычислительной сети, организации локальной телефонной сети, передачи видеоинформации, сигналов от датчиков пожарной безопасности и охранных систем.

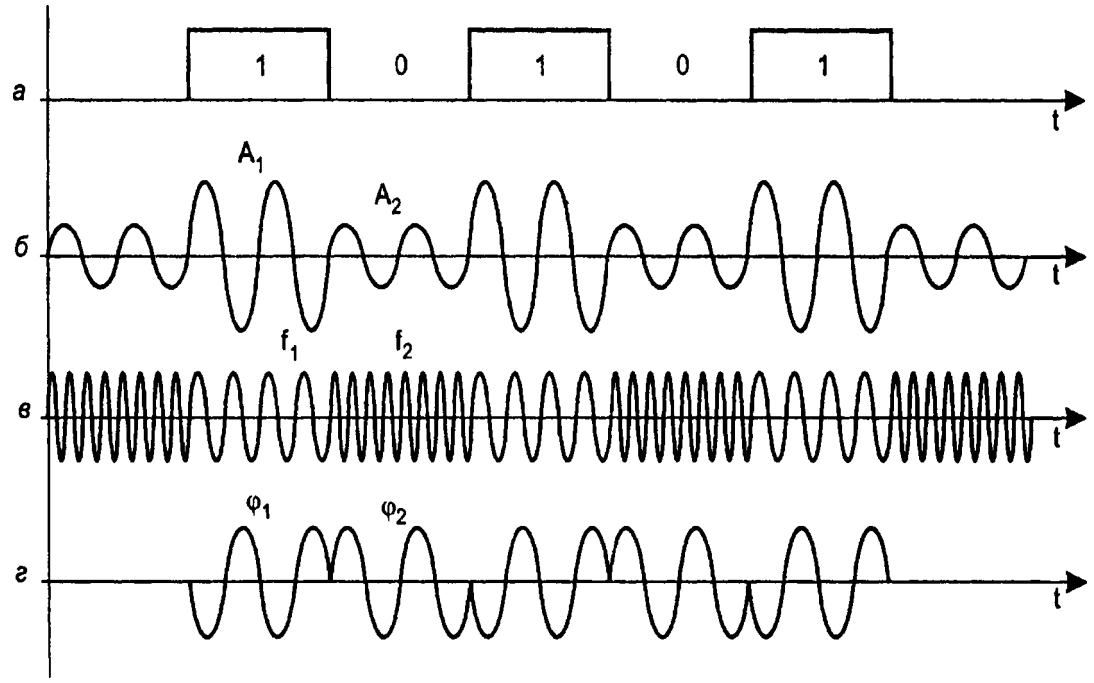
Подобная универсализация позволяет автоматизировать многие процессы контроля, мониторинга и управления хозяйственными службами и системами жизнеобеспечения предприятия.

Тема лекции: **Передача цифровых данных на физическом уровне. Методы модуляции, кодирования и мультиплексирования. Системы беспроводной передачи данных. Технологии первичных сетей**

План лекции:

1. Методы модуляции (манипуляции).
2. Проблемы дискретизации аналоговых сигналов. Аналого-цифровое (АЦП) и цифро-аналоговое преобразование (ЦАП).
3. Методы кодирования. Избыточное кодирование. Обнаружение и коррекция ошибок.
4. Методы мультиплексирования и коммутации.
5. Системы беспроводной передачи данных .
6. Технологии первичных сетей.

Методы модуляции

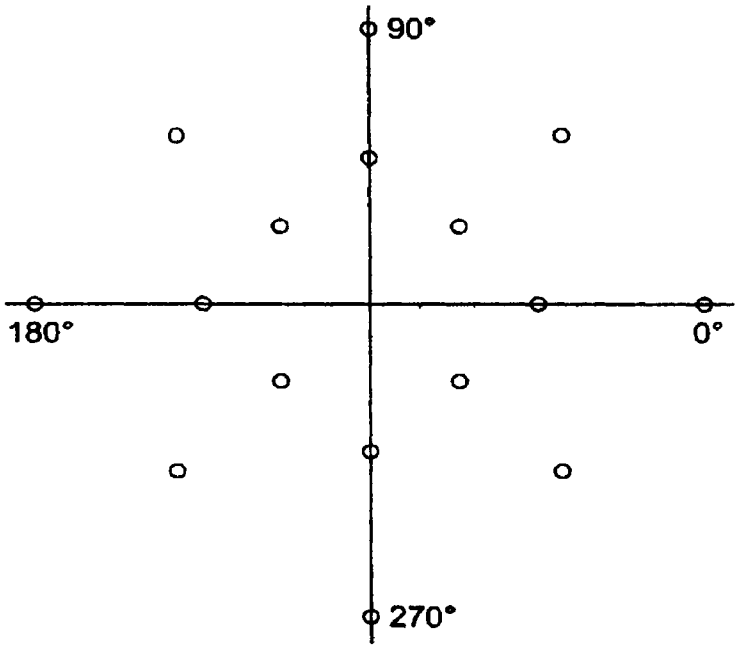


Методы модуляции

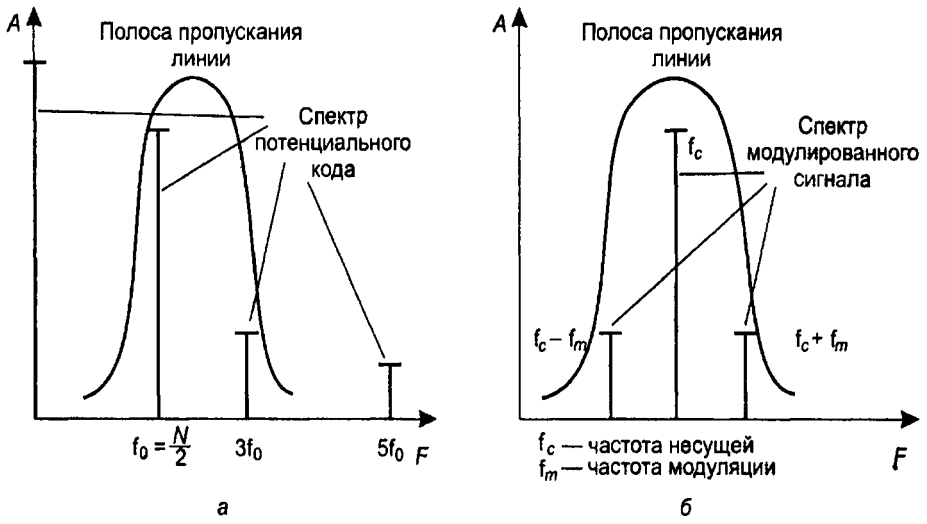
1. Амплитудная модуляция (б). В случае дискретной информации – амплитудная манипуляция (Amplitude Shift Keying, ASK).
2. Частотная модуляция (в). В случае дискретной информации (дискретный набор значения частоты) – частотная манипуляция (Frequency Shift Keying, FSK). Варианты: Binary FSK, BFSK, Multilevel FSK, MFSK.
3. Фазовая модуляция (г). В случае дискретной информации (дискретный набор значения фазы) – частотная манипуляция (Phase Shift Keying, PSK). Варианты: Binary PSK, BPSK, Quadrature PSK, QPSK.
4. Комбинированные методы: квадратурная амплитудная модуляция (Quadrature Amplitude Modulation, QAM)

Спектр модулированного сигнала зависит от типа модуляции и скорости

модуляции (т.е. скорости передачи битов исходной информации).



Квадратурная амплитудная модуляция



Спектр модулированного сигнала
а) потенциальное кодирование
б) амплитудная модуляция

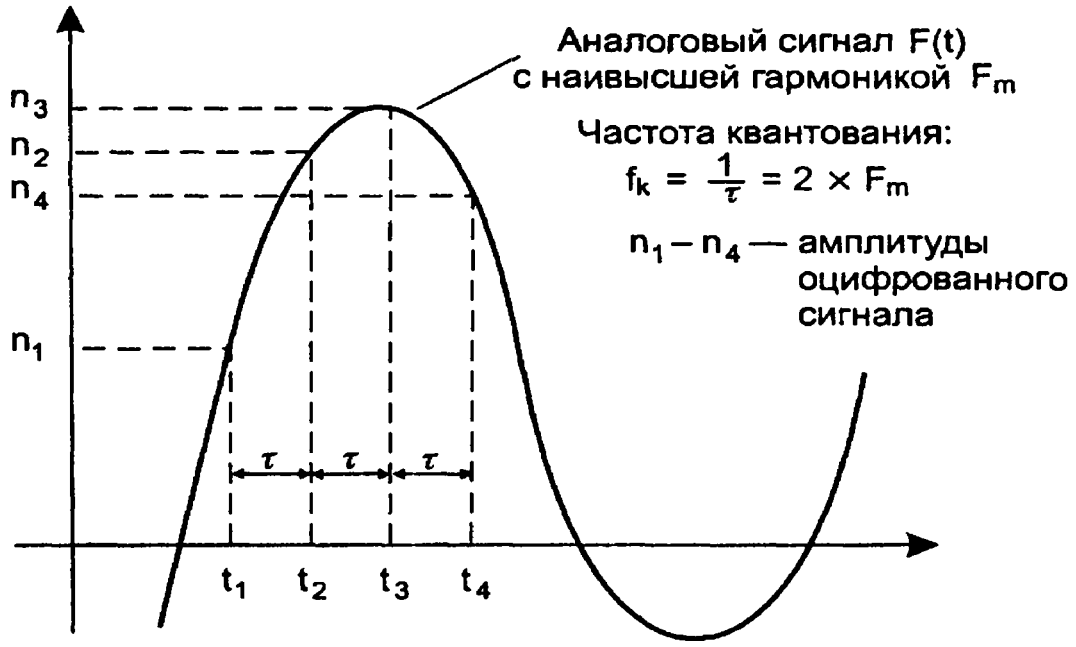
Дискретизация аналоговых сигналов

Аналого-цифровое преобразование (АЦП): амплитуда исходной непрерывной функции измеряется с заданным периодом (дискретизация по времени), затем каждый замер (отсчет, sample) представляется в виде двоичного числа определенной разрядности (квантование по уровню).

Цифро-аналоговое преобразование (ЦАП): обратный процесс – «демодуляция оцифрованных амплитуд» - восстановление исходной непрерывной функции.

Теорема Котельникова (Частота Найквиста):

$f_k >= 2F_m$.



Дискретизация (по времени) и квантование (по уровню) непрерывного процесса

Наиболее часто используются:

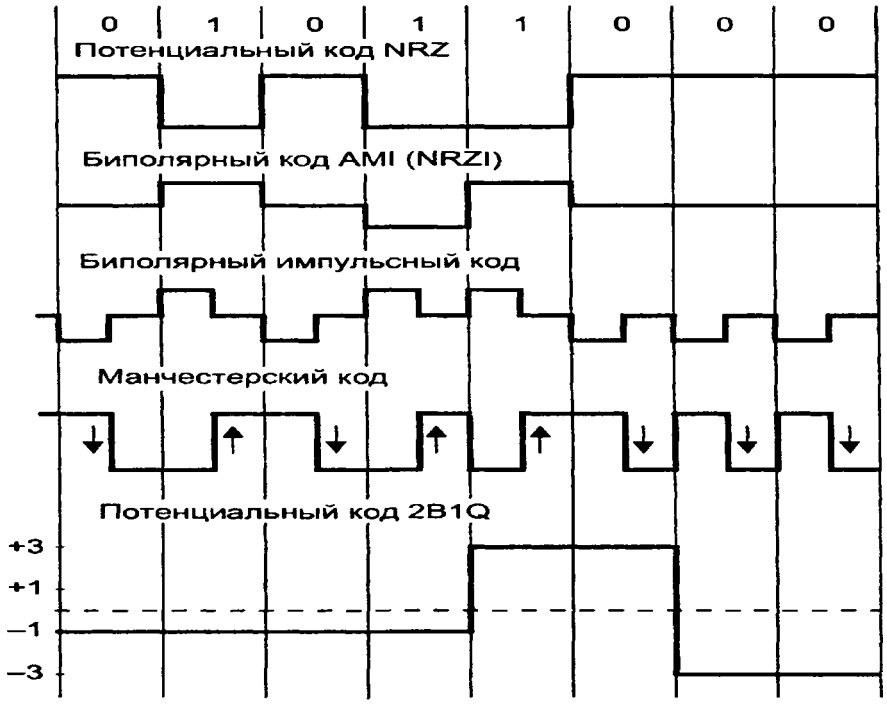
Импульсно-кодовая модуляция (ИКМ) – (Pulse Code Modulation, PCM).

Адаптивная дифференциальная импульсно-кодовая модуляция (АИКМ) – (Adaptive Differential Pulse Code Modulation, ADPCM).

Потенциальные и импульсные коды

В **потенциальных кодах** для представления логических единиц и нулей используется только значения потенциала сигнала (фронты – перепады не учитываются).

Импульсные коды позволяют представить двоичные данные импульсами определенной полярности или фронтами – перепадами определенного направления.



Примеры потенциальных и импульсных кодов

Требования к методам цифрового кодирования:

1. Наименьшая ширина спектра результирующего сигнала при заданной битовой скорости.
2. Обеспечение синхронизации.
3. Способность распознавать ошибки.
4. Низкая стоимость реализации.

Логическое кодирование. Избыточные коды

Соответствие исходных и результирующих кодов 4B/5B представлено ниже.

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

Код 4B/5B затем передается по линии с помощью 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 могут встретиться более трех нулей подряд.

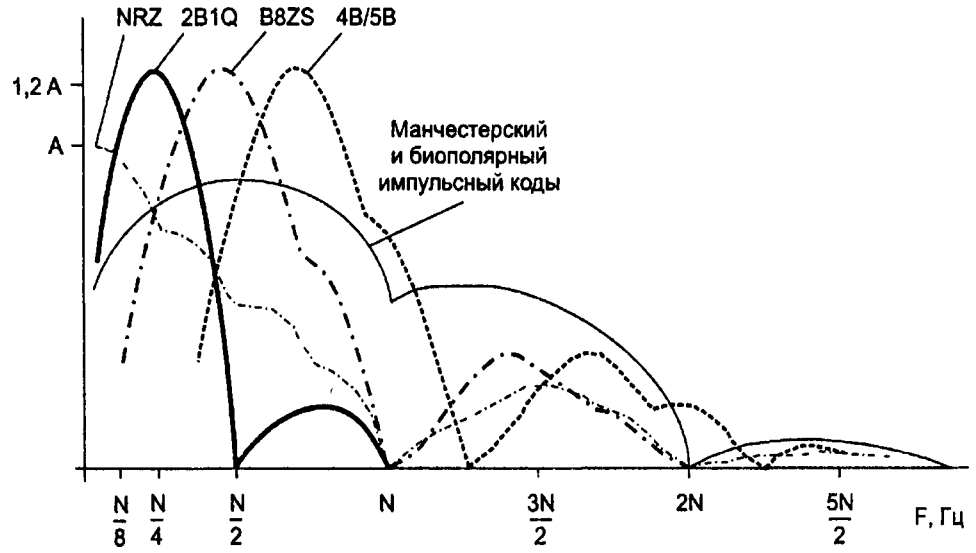
Буква B в названии кода означает, что элементарный сигнал имеет 2 состояния — от английского binary — двоичный. Имеются также коды и с тремя состояниями сигнала, например, в коде 8B/6T для кодирования 8 бит исходной информации используется код из 6 сигналов, каждый из которых имеет три состояния. Избыточность кода 8B/6T выше, чем кода 4B/5B, так как на 256 исходных кодов приходится 3^6=729 результирующих символов.

Логическое кодирование. Скрэмблирование

Перемешивание данных скрэмблером перед передачей их в линию с помощью потенциального кода является другим способом логического кодирования.

Методы скрэмблирования заключаются в побитном вычислении результирующего кода на основании бит исходного кода и полученных в предыдущих тактах бит результирующего кода. Например, скрэмблер может реализовывать следующее соотношение:

$$B_i = A_i \oplus B_{i-3} \oplus B_{i-5},$$



N — скорость передачи данных, бит/с;
A — амплитуда сигнала

Спектры потенциальных и импульсных кодов до и после логического кодирования

Методы компрессии данных

1. Десятичная упаковка.
2. Относительное кодирование.
   Адаптивная дифференциальная импульсно-кодовая модуляция (АИКМ) – (Adaptive Differential Pulse Code Modulation, ADPCM).
   Адаптивное сжатие.
3. Статистическое кодирование (алгоритм и коды Хафмана) .

Методы обнаружения и коррекции ошибок

Методы обнаружения и коррекции ошибок основаны на передаче избыточной служебной информации (контрольной суммы), по которой можно судить о достоверности принятых данных.

Контрольная сумма (контрольная последовательность кадра) – (Frame Check Sequence, FCS).

Алгоритмы вычисления FCS:

1. Контроль по паритету.

2. Вертикальный и горизонтальный контроль по паритету.
3. Циклический избыточный контроль (Cyclic Redundancy Check, CRC).

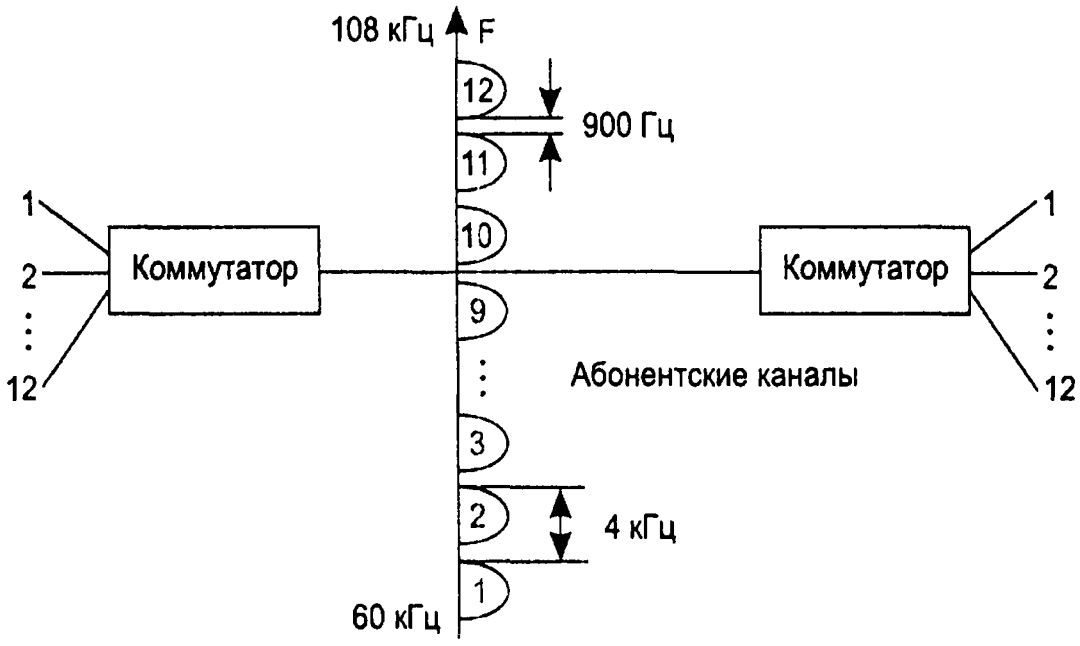
Методы коррекции ошибок: избыточные корректирующие коды (коды Хэмминга).

Методы мультиплексирования и коммутации

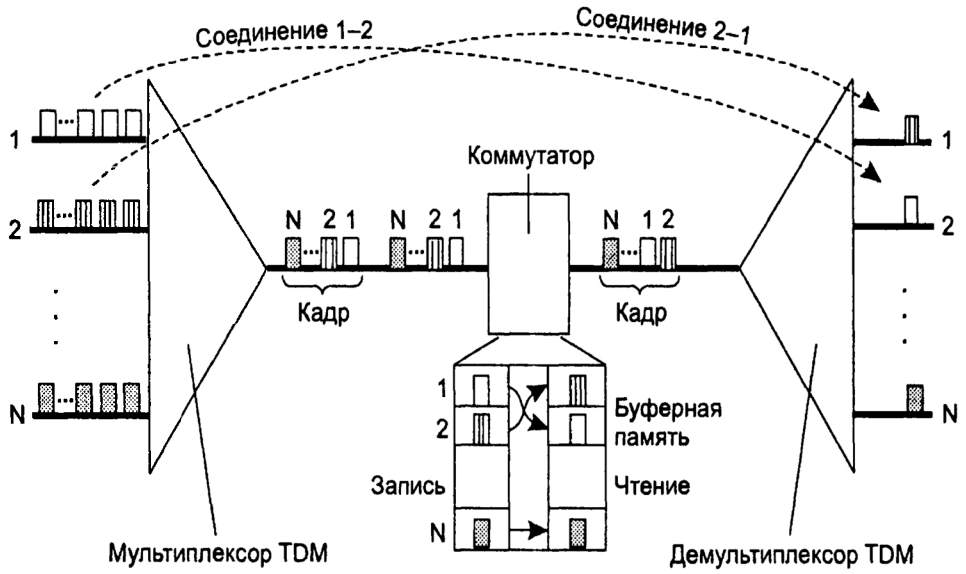
Задача: образовать в линии связи отдельные каналы передачи данных, которые можно использовать для коммутации информационных потоков пользователей.

Основные методы мультиплексирования абонентских каналов:

1. **Частотное мультиплексирование (Frequency Division Multiplexing, FDM).** Коммутация каналов.
2. **Волновое мультиплексирование (Wave Division Multiplexing, WDM) (DWDM)).** Коммутация каналов.
3. **Временное мультиплексирование (Time Division Multiplexing, TDM).** Коммутация каналов и пакетов**.**
4. **Множественный доступ с кодовым разделением (Code Division Multiple Access, CDMA).** Коммутация каналов и расширение спектра.



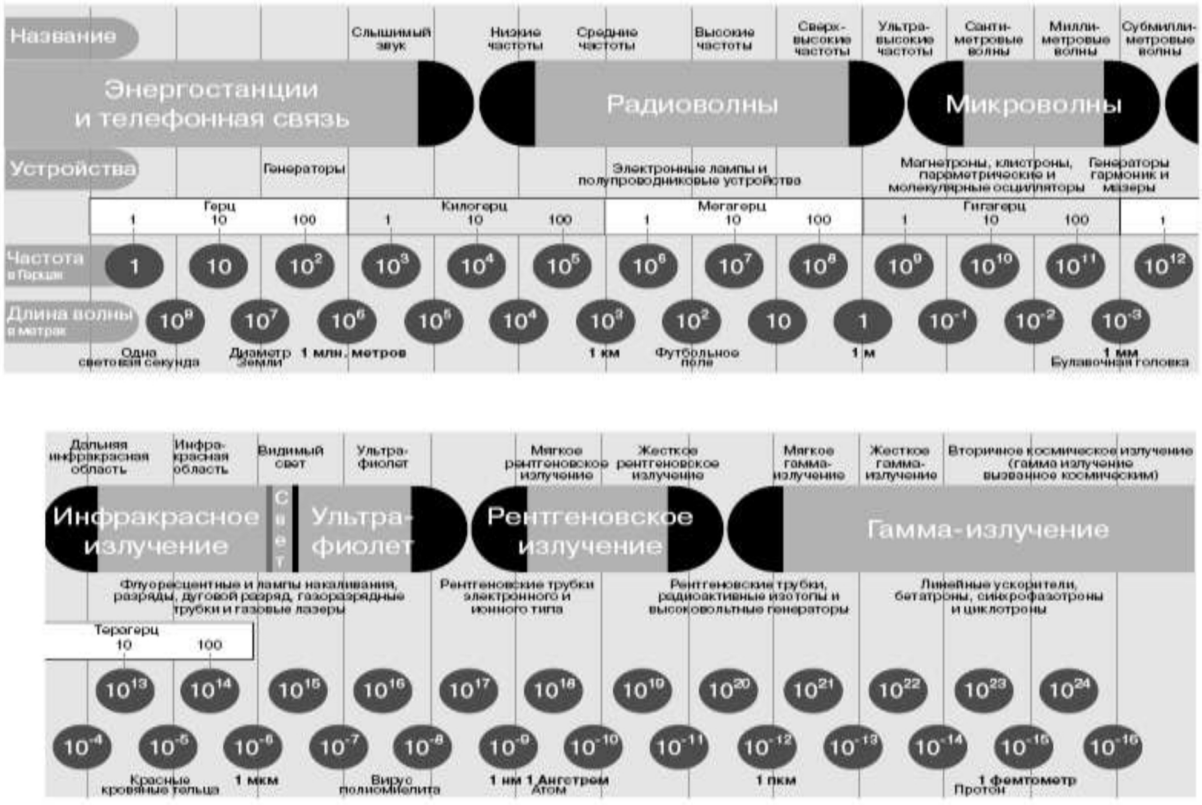
FDM – коммутация в телефонных сетях



TDM

Системы беспроводной передачи данных

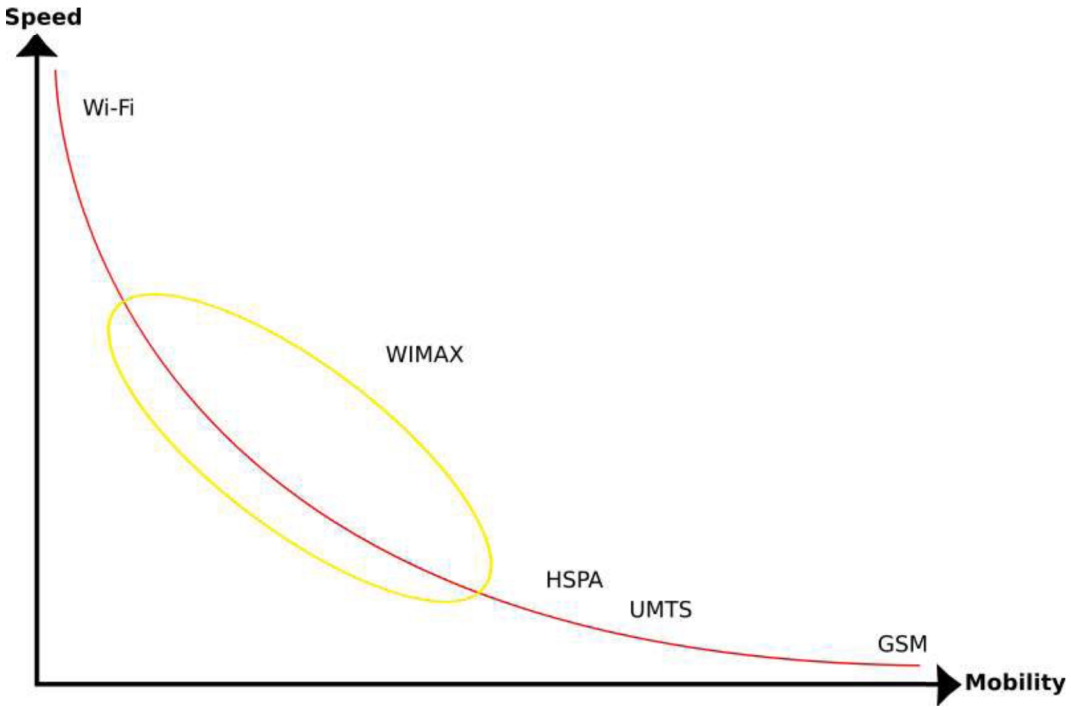
1. История и физические свойства беспроводных линий связи.
2. Диапазоны электромагнитного спектра. Лицензирование.
3. Типы беспроводных систем связи (двухточечные, многоточечные, радиорелейные, сотовые, спутниковые).
4. Технологии широкополосного сигнала (FHSS, DSSS, OFDM, CDMA).



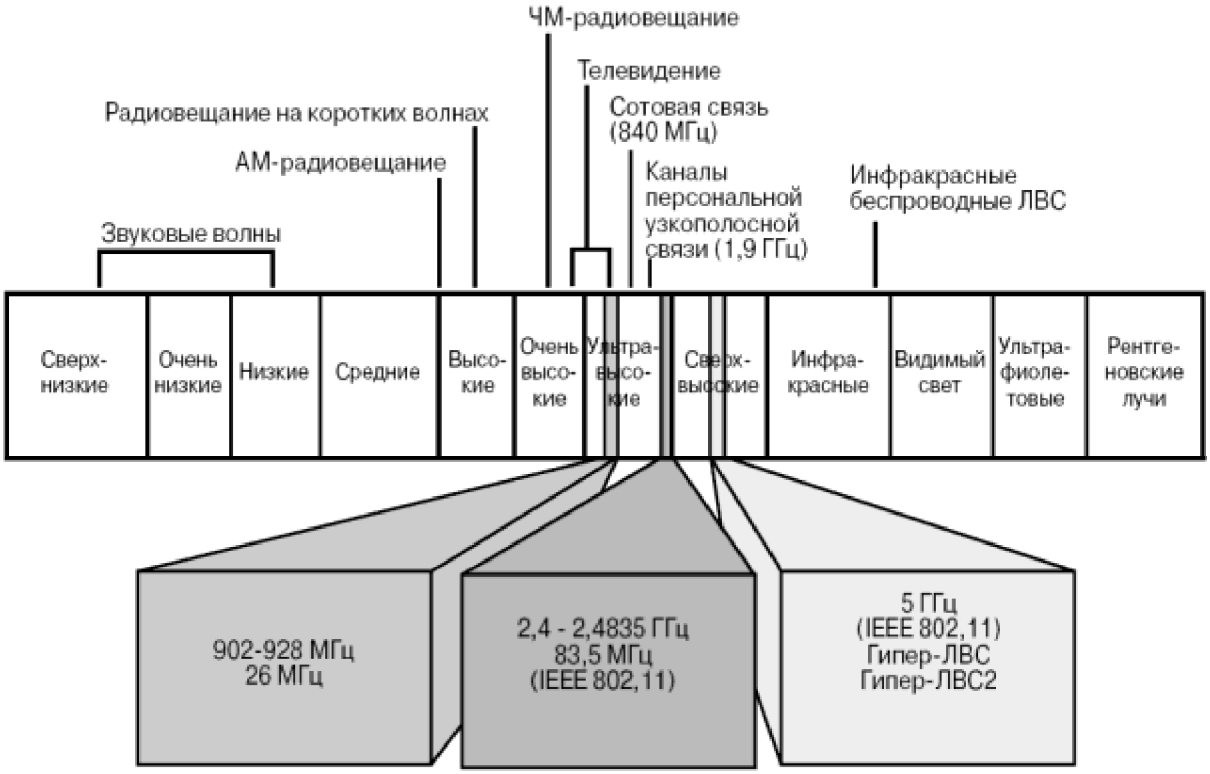
Диапазоны электромагнитного спектра



Сравнение технологий беспроводных сетей



Сравнение технологий беспроводных сетей



Диапазоны не требующие лицензирования (США)

Технологии широкополосного сигнала (FHSS, DSSS, OFDM, CDMA)

1. **FHSS**
2. **DSSS**
3. **OFDM**
4. **CDMA**

Первичные сети

Первичные сети предназначены для создания **универсальной коммутируемой инфраструктуры**, для гибкой организации **постоянного канала с двухточечной топологией между двумя** пользовательскими устройствами.

На основе каналов первичных сетей работают **наложенные** компьютерные и телефонные сети с пропускной способностью от 2 Мбит/с до 10 Гбит/с.

Три поколения технологий первичных сетей:

1. Плезиохронная цифровая иерархия (Plesiochronous Digital Hierarchy, **PDH**).
2. Синхронная цифровая иерархия (Synchronous Digital Hierarchy, **SDH**).
3. Уплотненное волновое мультиплексирование (Dense Wave Division Multiplexing, **DWDM**).

Тема лекции:

Технологии локальных сетей на основе разделяемой среды

План лекции:

1. Общие вопросы: важнейшие технологии; стандартные топологии; разделяемая среда.
2. Стек протоколов локальных сетей.
3. Структура стандартов IEEE 802.X.
4. Физические (MAC-) адреса.
5. Технология Ethernet: метод доступа; форматы кадров; спецификации физической среды.
6. Технологии Token Ring, FDDI.
7. Технология Fast Ethernet.
8. Технология Gigabit Ethernet.
9. Технология 10Gigabit Ethernet.

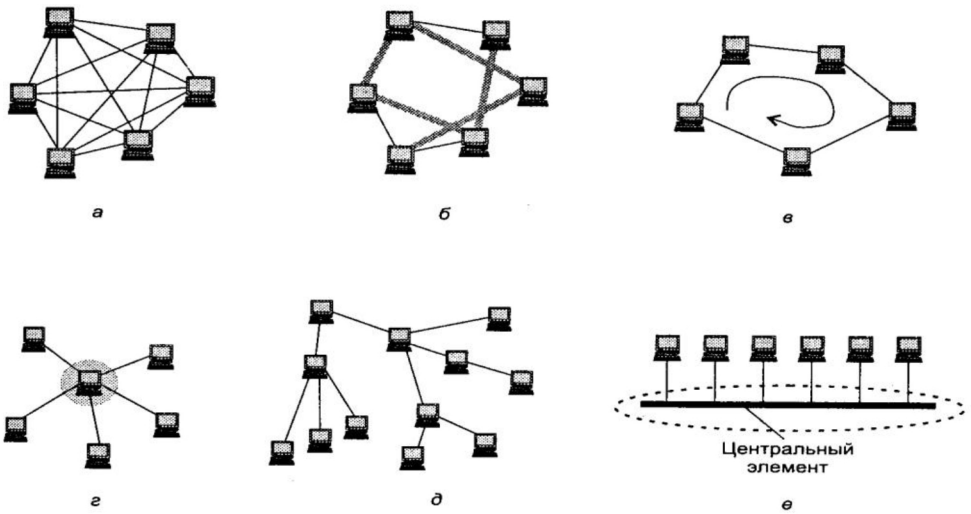
Технологии (протоколы) локальных сетей

Важнейшие технологии локальных сетей:

- Ethernet (Ethernet, Fast Ethernet, Gigabit Ethernet, 10G Ethernet – доминирует, развивается).
- Wi-Fi
- FDDI
- Token Ring

Технологии, расширяющие понятие классической локальной сети:

- HomePNA
- WiMax
- BlueTooth



Стандартные топологии локальных сетей а) полносвязная; б) ячеистая (mesh); в) кольцо (FDDI, Token Ring); г) звезда (Ethernet);

д) дерево; е) общая шина (Ethernet).

Разделяемая среда

Для упрощения и удешевления аппаратных и программных решений разработчики первых локальных сетей остановились на совместном использовании **общей среды** (shared media) передачи данных:

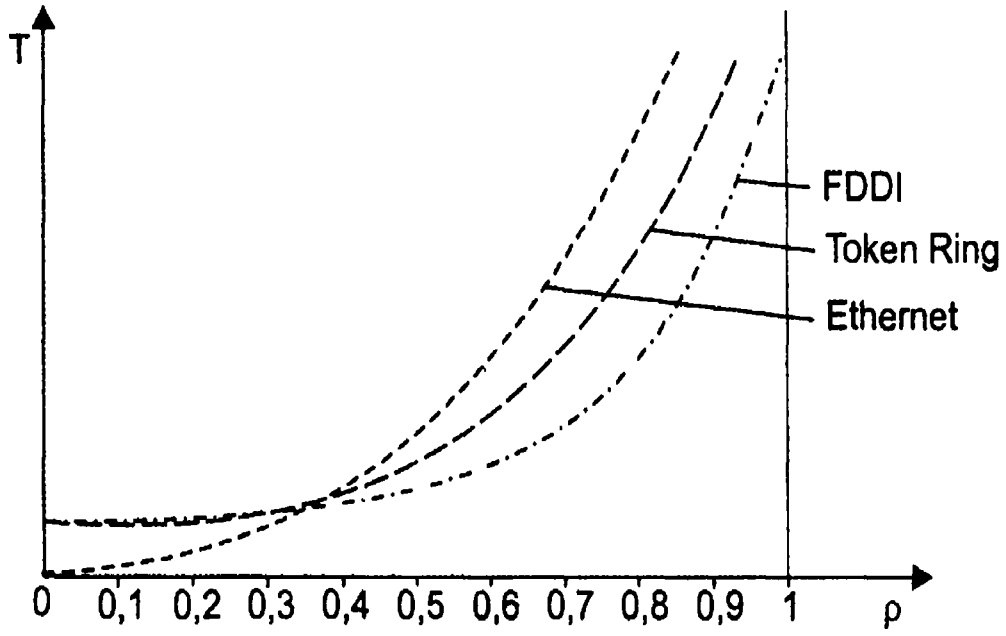
- Гавайский ун-т, радиосеть ALOHA, 1970. N. Abramson.
- Общая шина на коаксиальном кабеле, 1973, R. Metcalfe, Ethernet.

Методы доступа к разделяемой среде:

- Случайный (ALOHA, Ethernet).
- Детерминированный (FDDI, Token Ring).

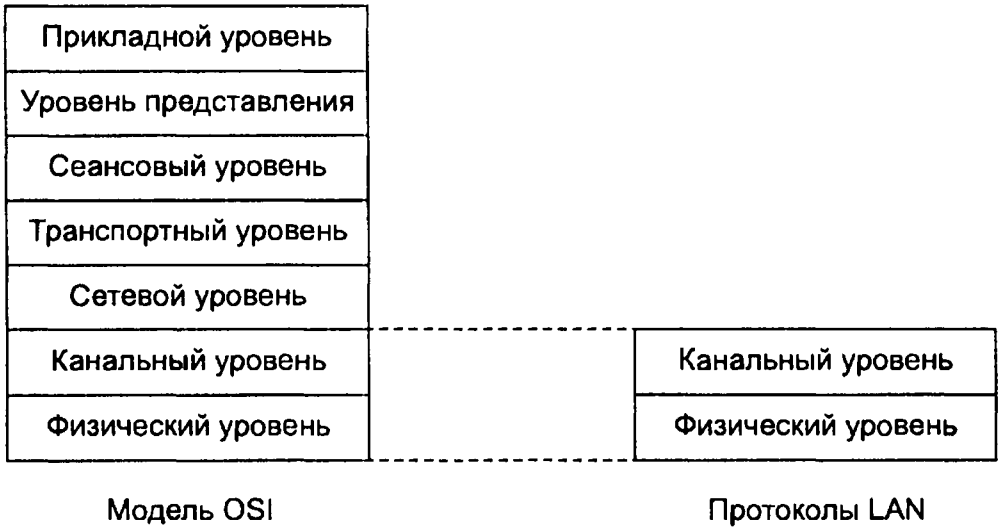
Основной недостаток разделяемой среды – плохая масштабируемость. Независимо от метода доступа к среде ее пропускная способность делится между всеми узлами сети.

Методы теории очередей, моделирование показывают (а практический опыт это подтверждает): как только коэффициент использования общей среды Ks (ρ на графике) превышает определенный порог, очереди к среде начинают расти нелинейно, и сеть становится практически неработоспособной (Ks для Aloha – 18%, для Ethernet – 30%, для FDDI, Token Ring – 60 – 70%).



Зависимость задержки доступа к среде от коэффициента использования общей среды ρ

Стек протоколов локальных сетей



Протоколы технологий локальных сетей реализуют, как правило, функции только двух нижних уровней модели OSI – физического и канального.

Канальный уровень локальных сетей

Канальный уровень локальных сетей делится на два подуровня:

- Уровень управления логическим каналом (**Logical Link Control, LLC**).
- Уровень управления управления доступом к среде (**Media Access Control, MAC**).

Функции уровня **LLC** обычно реализуются **программно**, соответствующим модулем операционной системы, а функции уровня **MAC** реализуются **программно-аппаратно**: сетевым адаптером и его драйвером.

Уровень MAC

Основными функциями уровня MAC являются:

- Обеспечение доступа к разделяемой среде.
- Передача кадров между конечными узлами, используя функции и устройства физического уровня.

Метод **случайного доступа** (захвата разделяемой среды), используемый Ethernet: узел, имеющий кадр для передачи, пытается его отправить без согласования времени использования разделяемой среды с другими узлами сети. Является децентрализованным, может приводить к коллизиям.

Методы **детерминированного доступа** (передача токена – децентрализовано, опрос – централизовано) имеют фиксированное и заранее известное ***максимальное время ожидания доступа к среде***.

Классический (legacy) Ethernet реализует **дейтаграммный полудуплексный** режим передачи данных.

Уровень LLC

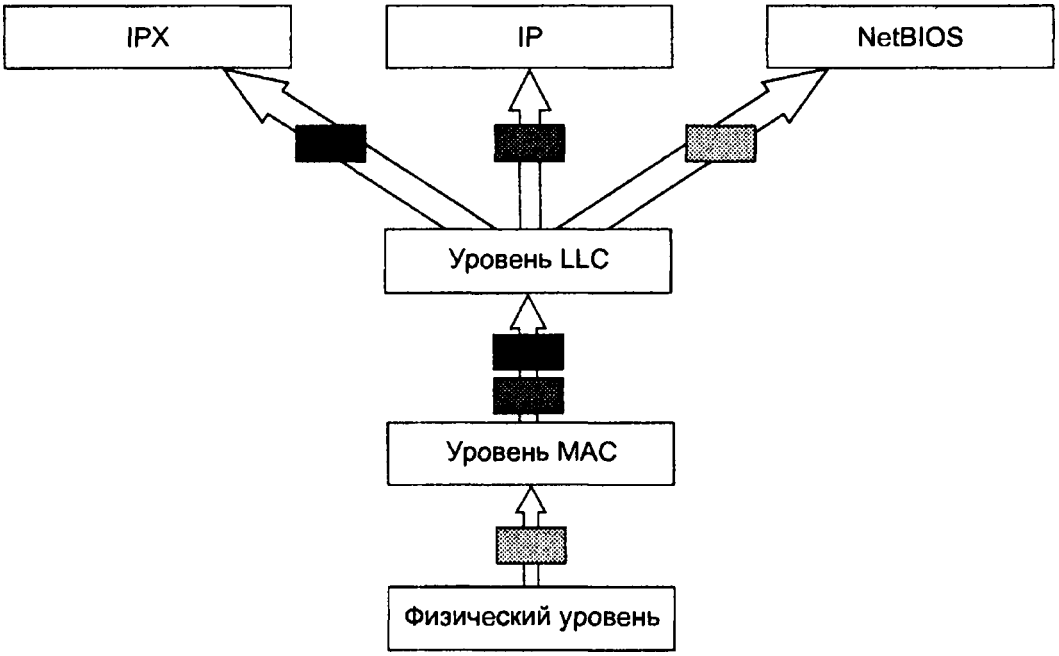
Уровень LLC реализует две функции:

- Организует интерфейс с прилегающим к нему сетевым уровнем.
- Обеспечивает доставку кадров с заданной степенью надежности.

Интерфейсные функции LLC заключаются в передаче пользовательских и служебных данных между уровнем MAC и сетевым уровнем (возможно с мультиплексированием / демультиплексированием).

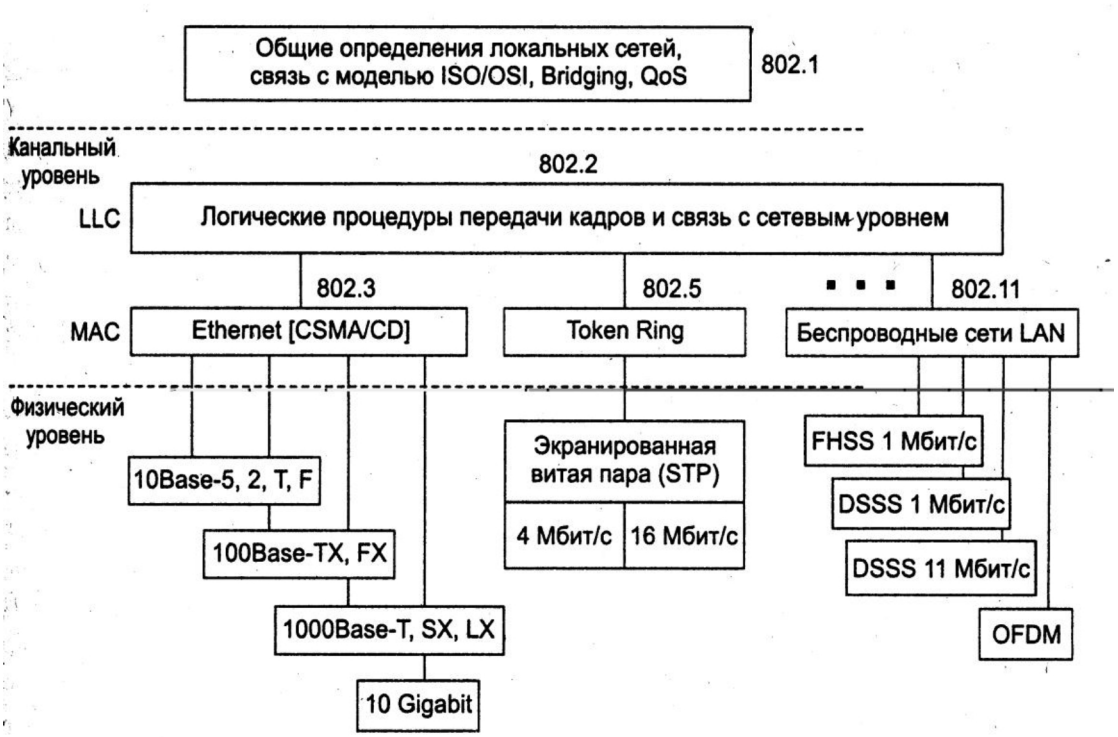
Уровень LLC предоставляет верхним уровням 3 типа транспортных услуг:

- Услуга LLC1 – без установления соединения и без подтверждения полученных данных.
- Услуга LLC2 – с установлением логического соединения и процедурой восстановления полученных данных (алгоритм скользящего окна).
- Услуга LLC3 – без установления соединения, но с подтверждением получения данных.



Демультиплексирование кадров протоколом LLC

Стандарты семейства IEEE 802.X



Структура стандартов IEEE 802.X

Перечень стандартов IEEE 802.X

IEEE 802.1 Bridging (networking) and Network Management **IEEE 802.2** Logical link control **IEEE 802.3** Ethernet IEEE 802.4 Token bus (disbanded) **IEEE 802.5** Defines the MAC layer for a Token Ring IEEE 802.6 Metropolitan Area Networks (disbanded) IEEE 802.7 Broadband LAN using Coaxial Cable (disbanded) IEEE 802.8 Fiber Optic TAG (disbanded) IEEE 802.9 Integrated Services LAN (disbanded) IEEE 802.10 Interoperable LAN Security (disbanded) **IEEE 802.11** Wireless LAN & Mesh (Wi-Fi certification) **IEEE 802.12** demand priority **IEEE 802.13** Cat.6 - 10Gb lan (new founded) IEEE 802.14 Cable modems (disbanded)	**IEEE 802.15** Wireless PAN **IEEE 802.15.1** (Bluetooth certification) **IEEE 802.15.4** (ZigBee certification) **IEEE 802.16** Broadband Wireless Access (WiMAX certification) **IEEE 802.16e** (Mobile) Broadband Wireless Access **IEEE 802.17** Resilient packet ring **IEEE 802.18** Radio Regulatory TAG **IEEE 802.19** Coexistence TAG **IEEE 802.20** Mobile Broadband Wireless Access **IEEE 802.21** Media Independent Handoff **IEEE 802.22** Wireless Regional Area Network

Перечень (история) стандартов IEEE 802.3

Ethernet Standard	Date	Description
Experimental Ethernet	1972	2.94 Mbit/s (367 kB/s) over coaxial cable (coax) cable bus
Ethernet II (DIX v2.0)	1982	10 Mbit/s (1.25 MB/s) over thin coax (thinnet) - Frames have a Type field. This frame format is used on all forms of Ethernet by protocols in the Internet protocol suite.
IEEE 802.3	1983	10BASE5 10 Mbit/s (1.25 MB/s) over thick coax - same as DIX except Type field is replaced by Length, and an 802.2 LLC header follows the 802.3 header
802.3a	1985	10BASE2 10 Mbit/s (1.25 MB/s) over thin Coax (thinnet or cheapernet)
802.3b	1985	10BROAD36
802.3c	1985	10 Mbit/s (1.25 MB/s) repeater specs
802.3d	1987	FOIRL (Fiber-Optic Inter-Repeater Link)
802.3e	1987	1BASE5 or StarLAN
802.3i	1990	10BASE-T 10 Mbit/s (1.25 MB/s) over twisted pair
802.3j	1993	10BASE-F 10 Mbit/s (1.25 MB/s) over Fiber-Optic
802.3u	1995	100BASE-TX, 100BASE-T4, 100BASE-FX Fast Ethernet at 100 Mbit/s (12.5 MB/s) w/autonegotiation
802.3x	1997	Full Duplex and flow control; also incorporates DIX framing, so there's no longer a DIX/802.3 split
802.3y	1998	100BASE-T2 100 Mbit/s (12.5 MB/s) over low quality twisted pair
802.3z	1998	1000BASE-X Gbit/s Ethernet over Fiber-Optic at 1 Gbit/s (125 MB/s)
802.3-1998	1998	A revision of base standard incorporating the above amendments and errata
802.3ab	1999	1000BASE-T Gbit/s Ethernet over twisted pair at 1 Gbit/s (125 MB/s)
802.3ac	1998	Max frame size extended to 1522 bytes (to allow "Q-tag") The Q-tag includes 802.1Q VLAN information and 802.1p priority information.
802.3ad	2000	Link aggregation for parallel links
802.3-2002	2002	A revision of base standard incorporating the three prior amendments and errata
802.3ae	2003	10 Gbit/s (1,250 MB/s) Ethernet over fiber; 10GBASE-SR, 10GBASE-LR, 10GBASE-ER, 10GBASE-SW, 10GBASE-LW, 10GBASE-EW
802.3af	2003	Power over Ethernet
802.3ah	2004	Ethernet in the First Mile
802.3ak	2004	10GBASE-CX4 10 Gbit/s (1,250 MB/s) Ethernet over twin-axial cable
802.3-2005	2005	A revision of base standard incorporating the four prior amendments and errata.
802.3an	2006	10GBASE-T 10 Gbit/s (1,250 MB/s) Ethernet over unshielded twisted pair(UTP)
802.3ap	2007	Backplane Ethernet (1 and 10 Gbit/s (125 and 1,250 MB/s) over printed circuit boards)
802.3aq	2006	10GBASE-LRM 10 Gbit/s (1,250 MB/s) Ethernet over multimode fiber
802.3ar	On Hold	Congestion management
802.3as	2006	Frame expansion
802.3at	exp. 2008	Power over Ethernet enhancements
802.3au	2006	Isolation requirements for Power Over Ethernet (802.3-2005/Cor 1)
802.3av	exp. 2009	10 Gbit/s EPON
802.3aw	2007	Fixed an equation in the publication of 10GBASE-T (released as 802.3-2005/Cor 2)
802.3ax	exp 2008	Move Link aggregation out of 802.3 to IEEE 802.1
802.3ay	exp 2008	Maintenance to base standard
802.3ba	exp. 2009	Higher Speed Study Group. 40 Gbit/s over 1m backplane, 10m Cu cable assembly (4x25 Gbit or 10x10 Gbit lanes) and 100 m of MMF and 100 Gbit/s up to 10 m or Cu cable assembly, 100 m of MMF or 40 km of SMF respectively

MAC – адреса

Для идентификации сетевых интерфейсов в локальной сети (может определять один или группу интерфейсов) используются регламентированные стандартом IEEE 802.3 уникальные 6 байтовые адреса (обычно записываются как шесть пар шестнадцатеричных цифр, разделенных тире или двоеточиями:

11-A0-17-3D-BC-01

Первый (младший) бит старшего байта: 0 – адрес индивидуальный; 1 – групповой (в частности 0xFFFFFFFFFFFF – все единицы – широковещательный). Второй бит старшего байта: 0 – централизованный, 1 – локальный.

Комитет IEEE распределяет между производителями сетевого оборудования организационно уникальные идентификаторы (Organizationally Unique Identifier, OUI), которые помещаются в старшие 3 байта MAC-адреса.

Локальные сети на основе разделяемой среды. Ethernet: метод доступа CSMA/CD

Метод **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection – коллективный доступ с распознаванием несущей и обнаружением коллизий).

Домен коллизий (Collision Domain).

Преамбула (самосинхронизирующиеся байты):
7 байт: 10101010 и 8-ой байт: 10101011.

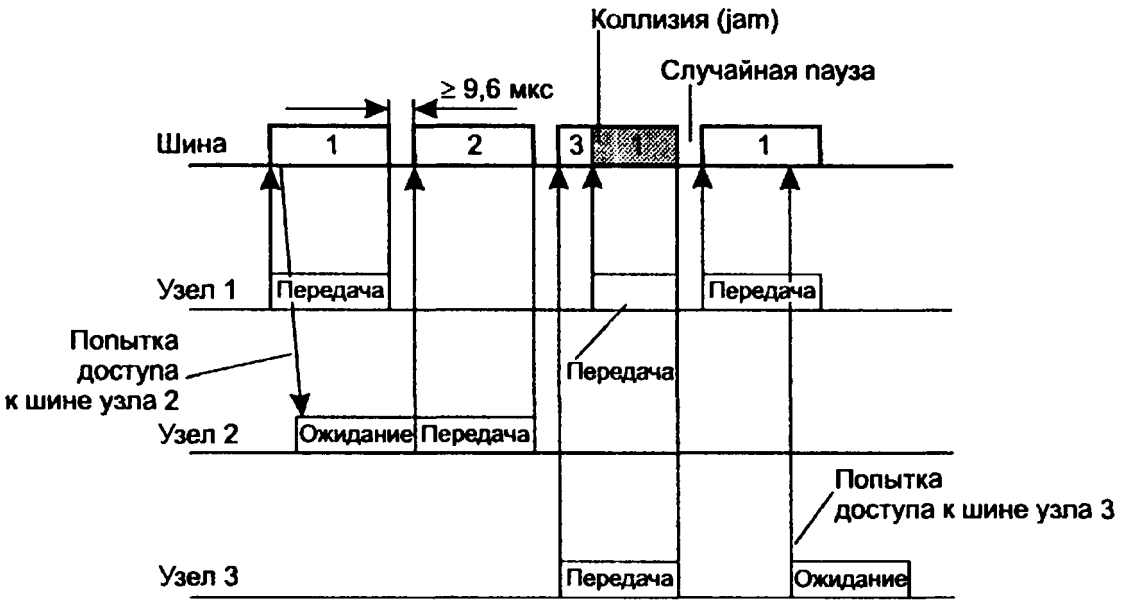
«Один передает – все остальные принимают».

Прослушивание среды (несущей частоты).

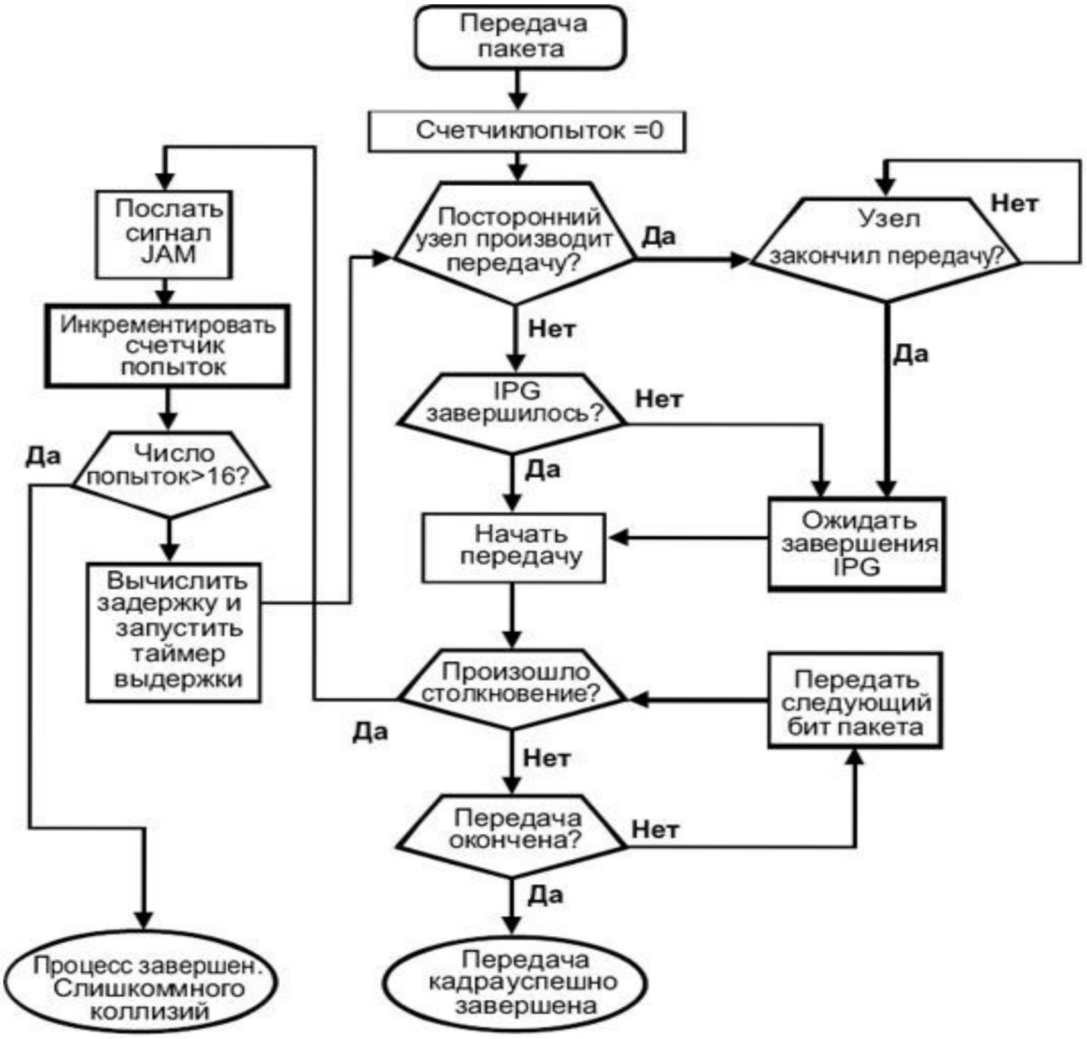
Возникновение и обнаружение коллизии, jam-последовательность (32 бита). Интервал отсрочки.

Время оборота (PDV).

Максимальное число сегментов и повторителей (хабов) в сети.



Ethernet: метод доступа CSMA/CD



Ethernet: метод доступа CSMA/CD

Ethernet: базовые параметры MAC-уровня

Параметры	**Значения**
Битовая скорость	10 Мбит/с
Интервал отсрочки	512 битовых интервала
Межкадровый, или межпакетный, интервал (IPG)	9,6 мкс
Максимальное число попыток передачи	16
Максимальное число возрастания диапазона паузы	10
Длина jam-последовательности	32 бит
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

Форматы кадров Ethernet

Кадр 802.3/LLC

6	6	2	1	1	1(2)	46–1497 (1496)	4
DA	SA	L	DSAP	SSAP	Управляющее поле	Данные	FCS
			Заголовок LLC				

Кадр Raw 802.3/Novell 802.3

6	6	2	46–1500	4
DA	SA	L	Данные	FCS

Кадр Ethernet DIX (II)

6	6	2	46–1500	4
DA	SA	T	Данные	FCS

Кадр Ethernet SNAP

6	6	2	1	1	1	3	2	46–1492	4
DA	SA	L	DSAP	SSAP	Управляющее поле	OUI	T	Данные	FCS
			AA	AA	03	000000			
			Заголовок LLC			Заголовок SNAP			

Спецификации физической среды Ethernet

10Base-5 – коаксиальный кабель диаметром 0,5 дюйма («толстый коаксиал»). Волновое сопротивление – 50 Ом. Максимум – 500м без повторителей.

10Base-2 – коаксиальный кабель диаметром 0,25 дюйма («тонкий коаксиал»). Волновое сопротивление – 50 Ом. Максимум – 185м без повторителей.

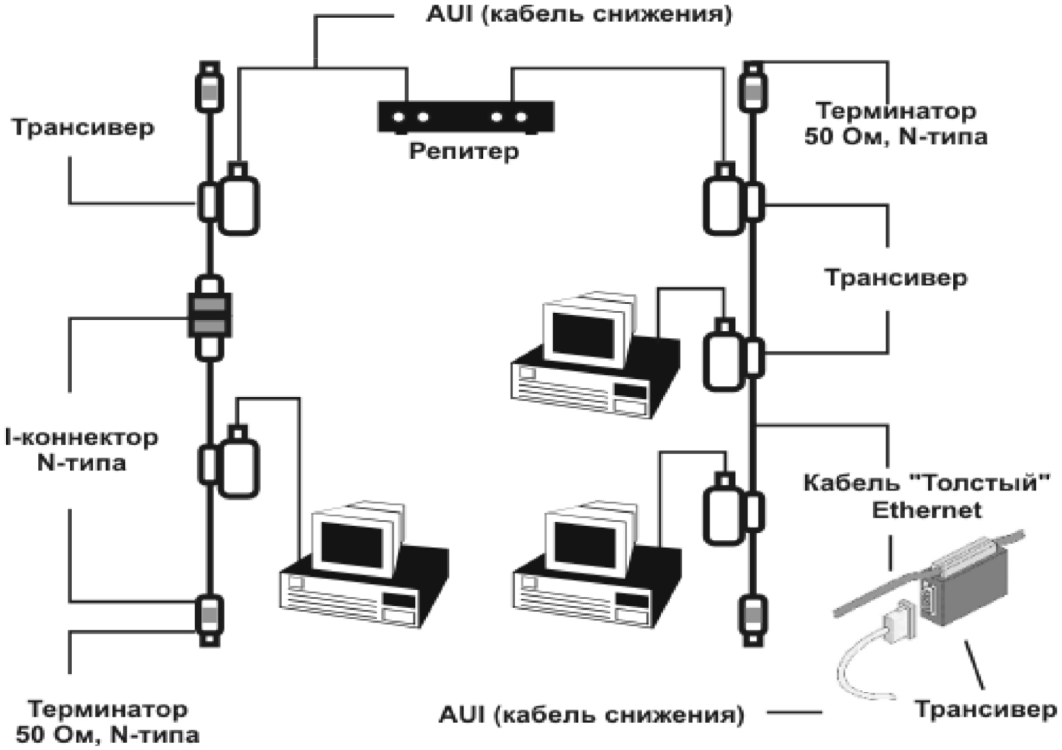
10Base-T – неэкранированная витая пара (UTP). Звезда на базе хаба (макс. – 100м). Волновое сопротивление – 100 Ом.

10Base-F – волоконно-оптический кабель.

Спецификации физической среды Ethernet

Параметр	10Base-5	10Base-2	10Base-T	10Base-F
Кабель	Толстый коаксиальный кабель RG-8 или RG-11	Тонкий коаксиальный кабель RG-58	Неэкранированная 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 станциями сети	4	4	4	4 (5 для 10 Base-FB)

Ethernet на «толстом» коаксиале 10Base-5



Каждый компьютер подсоединяется к главному кабелю специальным "кабелем снижения" (drop cable). Который, в свою очередь, присоединяется к AUI-порту сетевого адаптера.

Стандарт 10Base5 поддерживает до 100 узлов на сегмент (расстояние между

узлами кратно 2,5 метрам).

Максимальная длина сегмента не более 500 метров.

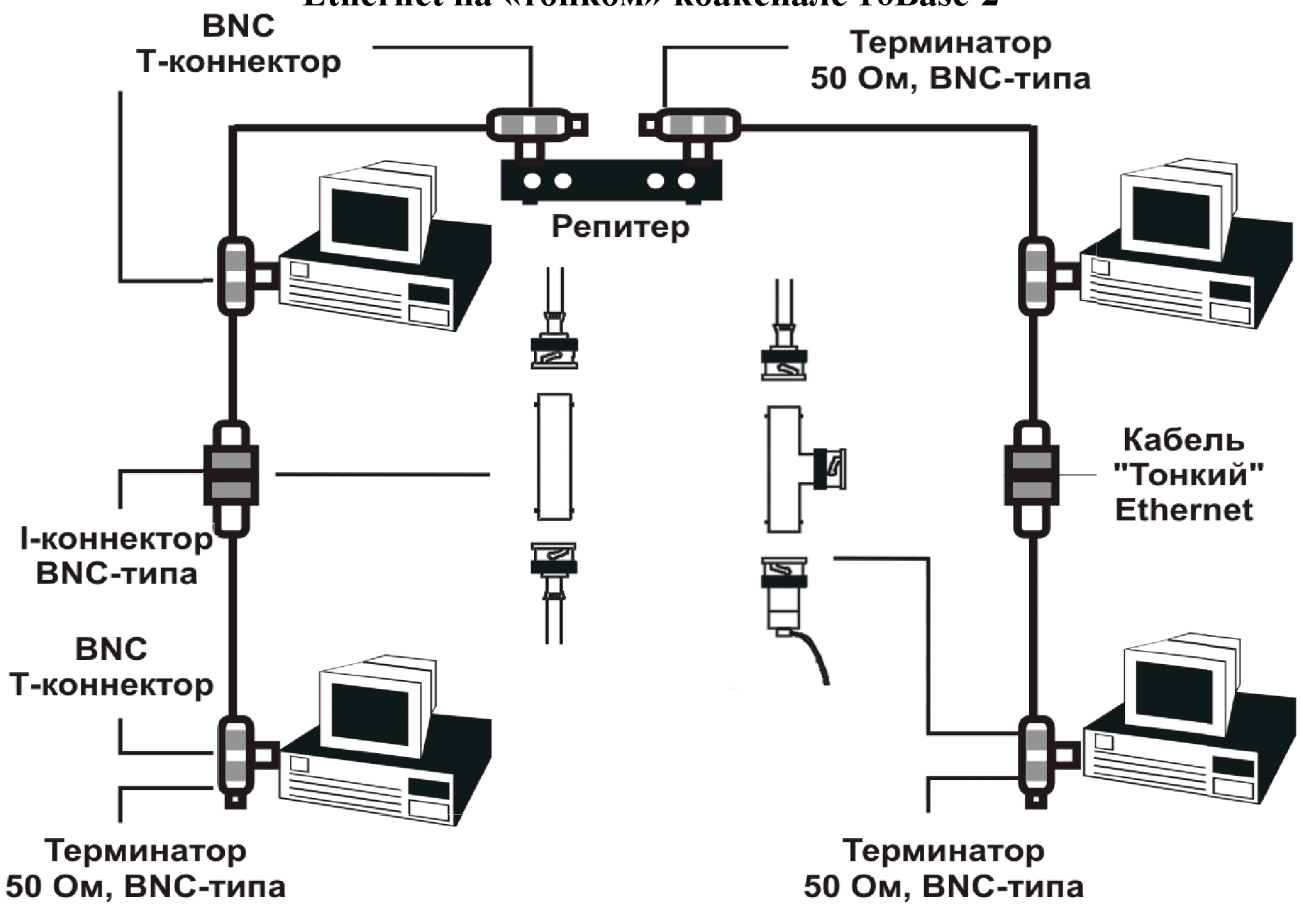
Главный кабель RG-8, RG-11. (RG от "Radio Grade" – волновод). Наружный диаметр 10 мм, центральный проводник - 2,17 мм, затухание на частоте 10 МГц в районе 70 dB/км.

Используются коннекторы N-типа.

Кабель снижения состоит из витых пар, может иметь длину до 50 метров. Используются разъемы типа DB15 (15 контактов), более известные под названием "AUI". Внешне они похожи на известные DB9 (RS-232, 9 контактов).

Правило 5-4-3: (не более) 5 сегментов – 4 повторителя – 3 нагруженных сегмента.

Ethernet на «тонком» коаксиале 10Base-2



К одному сегменту не может быть подключено более 30 устройств, длина которого должна составлять не более 185 м. Минимальное расстояние между ними составляет 0,5 метра. Таким образом, в локальной вычислительной сети может быть максимум 90 компьютеров.

Кабель RG58/U (одна центральная жила), RG58A/U, RG58C/U (негорючий материал диэлектрика).

Наружный диаметр около 5 мм, центральный проводник - 0,8 мм, затухание на частоте 10 МГц около 160 dB/км.

Применяются разъемы BNC-типа. Для подключения сетевых адаптеров к кабелю используются специальные Т-коннекторы (T-Connector).

Правило 5-4-3: (не более) 5 сегментов – 4 повторителя – 3 нагруженных сегмента.

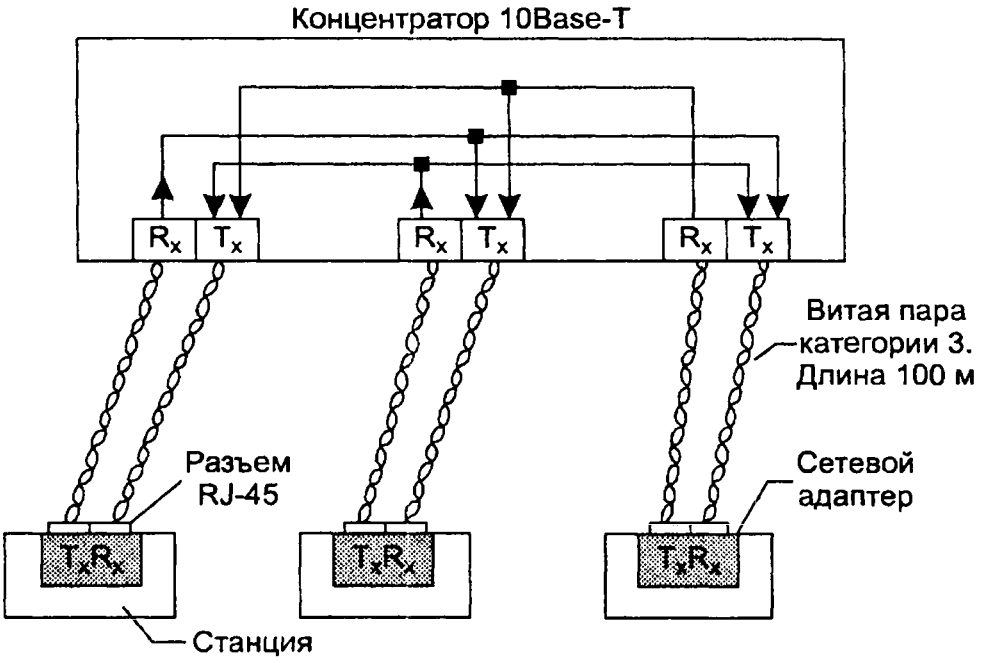
Ethernet на витой паре 10Base-T

Главное преимущество технологии 10Base-T – наличие между конечными узлами активного устройства (хаб, повторитель), которое может контролировать работу узлов и изолировать от сети некорректно работающие узлы.

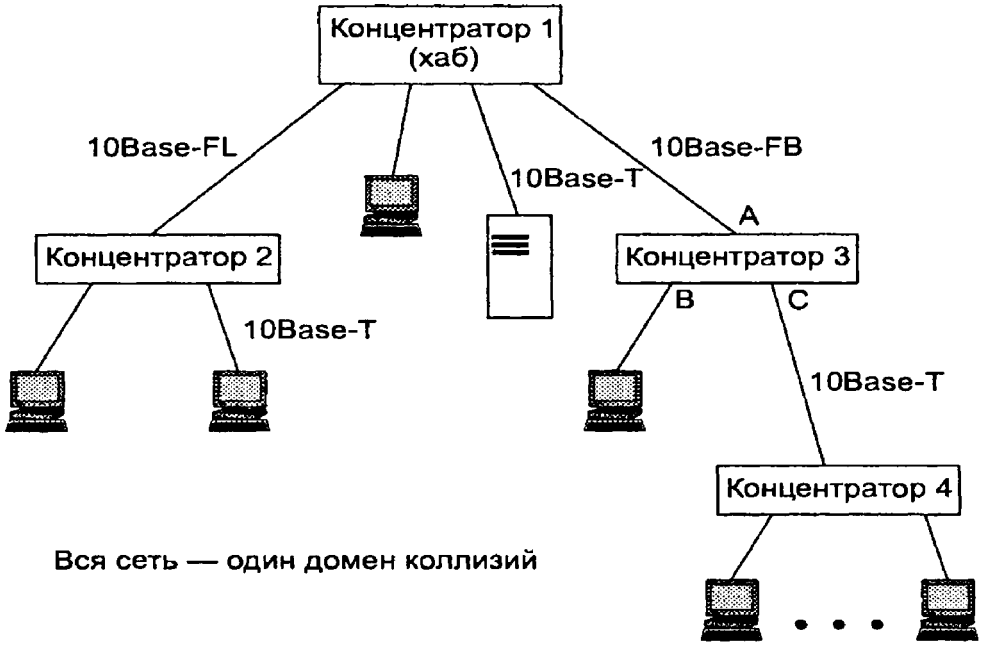
Физическая топология – звезда, логическая топология – общая шина (разделяемая среда).

Вместо правила 5-4-3 появилось правило «4-х хабов»: число хабов между двумя любыми хостами не может быть больше четырех.

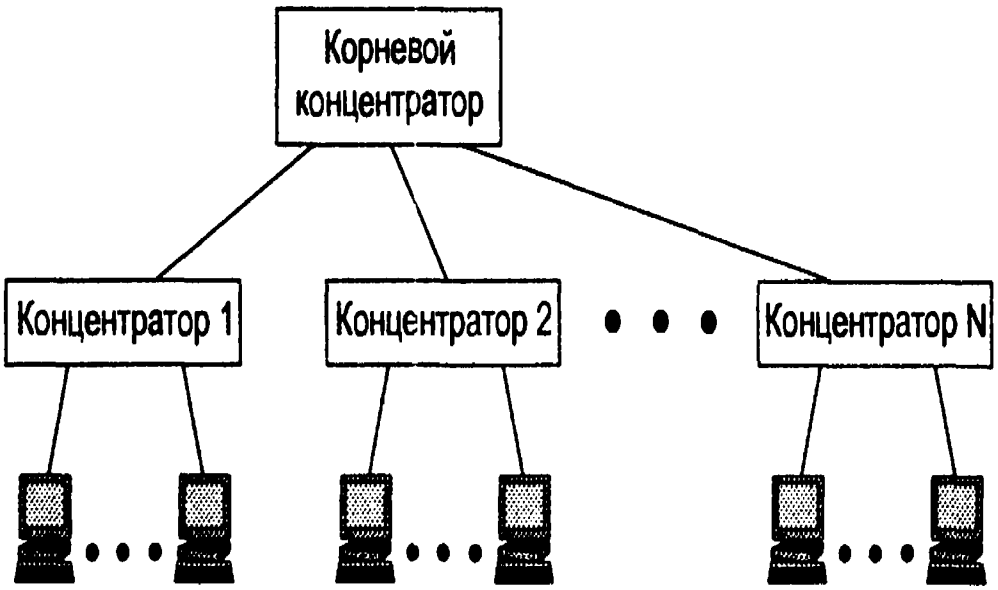
Главное осталось общим: поддержка **CSMA/CD** и **единый домен коллизий**: часть сети Ethernet, все узлы которой конкурируют за общую разделяемую среду, и, следовательно, каждый узел может создать коллизию с любым другим узлом этой части сети.



Ethernet на витой паре 10Base-T. Использование концентратора



Ethernet на витой паре 10Base-T. Структура сети



Ethernet на витой паре 10Base-T. Структура сети. Правило 4-х хабов

Token Ring. Доступ с передачей маркера

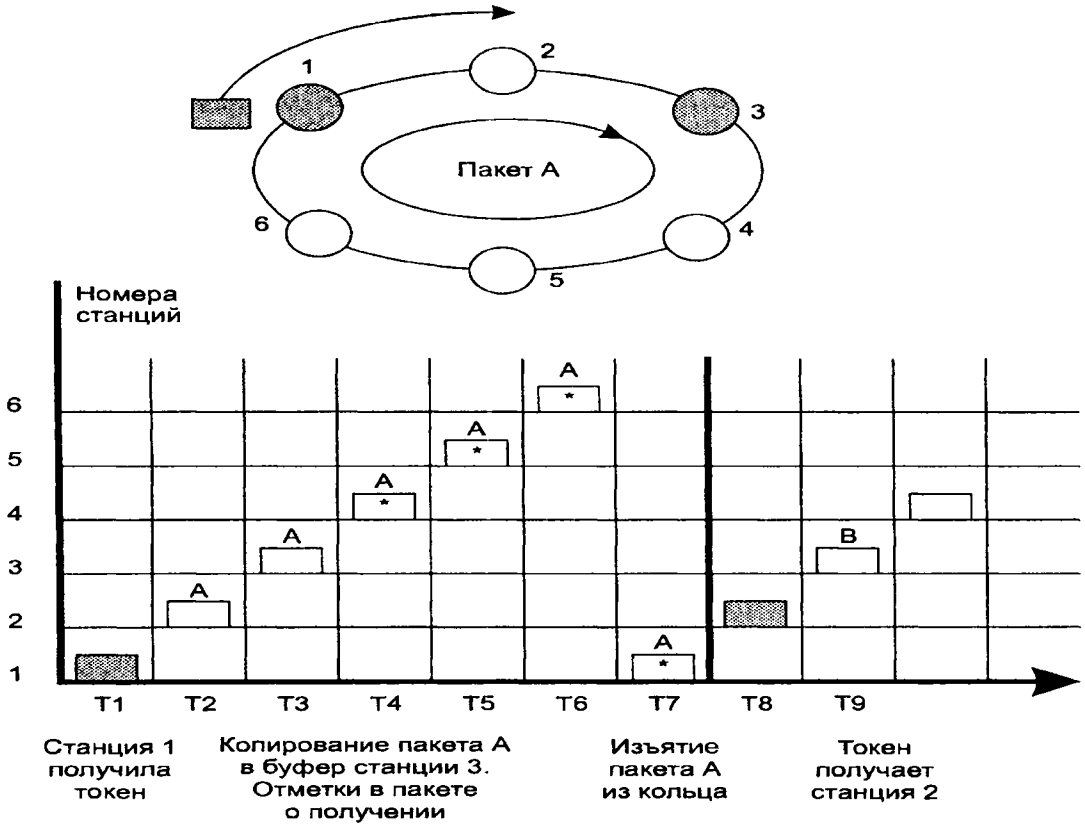
Разработка IBM (1984), основа стандарта IEEE 802.5 (1985).

Token Ring является классической сетью с передачей маркера. Логическая топология (и физическая в первых версиях сети) – кольцо. Более современные модификации построены на витой паре по топологии "звезда", и с некоторыми оговорками, совместимы с Ethernet.

Скорость передачи, описанная в IEEE 802.5, составляет 4 Мбит/с, однако существует более поздняя реализация на 16 Мбит/с. Из-за более упорядоченного

(детерминированного) метода доступа к среде, Token Ring на ранних этапах развития часто продвигался как более качественная замена Ethernet.

Для контроля сети одна из станций исполняет роль активного монитора.



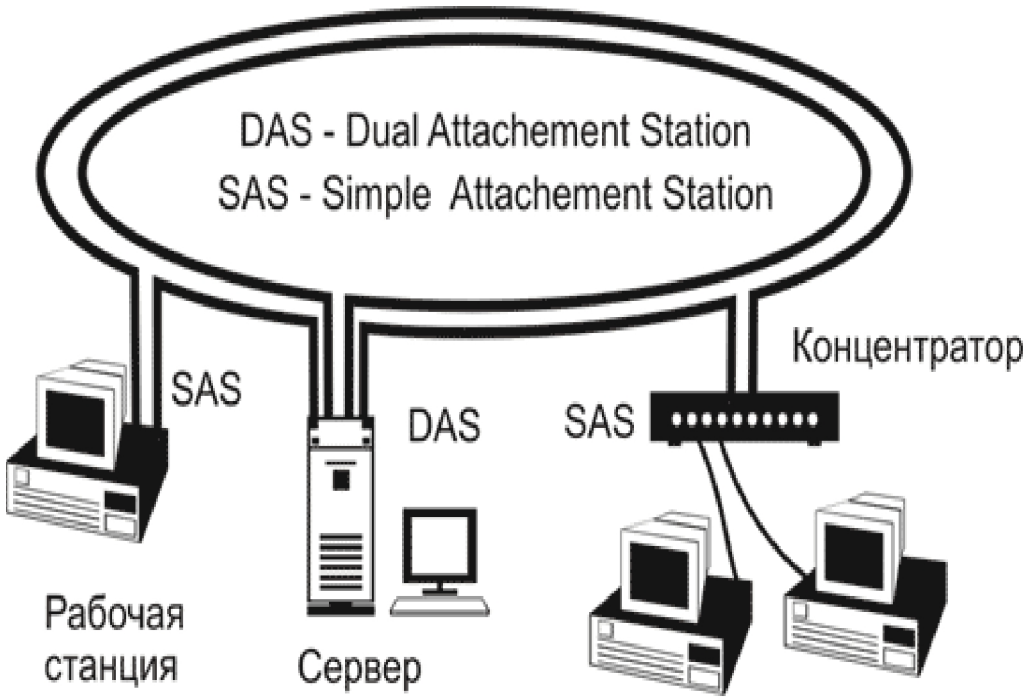
Token Ring. Доступ с передачей маркера

FDDI - первая локальная сеть на оптоволокне

Технология Fiber Distributed Data Interface (FDDI) разработана в 1988 году комитетом ANSI. Это была первая сеть, использовавшая в качестве среды передачи только оптоволоконный кабель. Причиной разработки была недостаточная в то время скорость (не более 10 Мбит/с) и надежность (отсутствие схем резервирования) локальных сетей.

Стандарт FDDI оговаривает передачу данных по двойному кольцу оптоволоконного кабеля со скоростью 100 Мбит/с, что позволяет получить надежный (зарезервированный) и быстрый канал. Расстояния вполне глобальные - до 100 км по периметру. Логически работа сети была построена на передаче маркера.

Дополнительно предусматривалась развитая схема приоритезации трафика. Сначала рабочие станции разделялись на два вида - синхронные (имеющие постоянную полосу пропускания), и асинхронные. Последние, в свою очередь, распределяли среду передачи с помощью восьмиуровневой системы приоритетов.



FDDI - первая локальная сеть на оптоволокне

Технология Fast Ethernet

IEEE 802.3u (1995) - дополнения к стандарту 802.3 (главы 21-30). Форматы кадров не изменились. Временные параметры уменьшились в 10 раз (битовый интервал стал 10 нс, однако, измеренные в битовых интервалах не изменились).

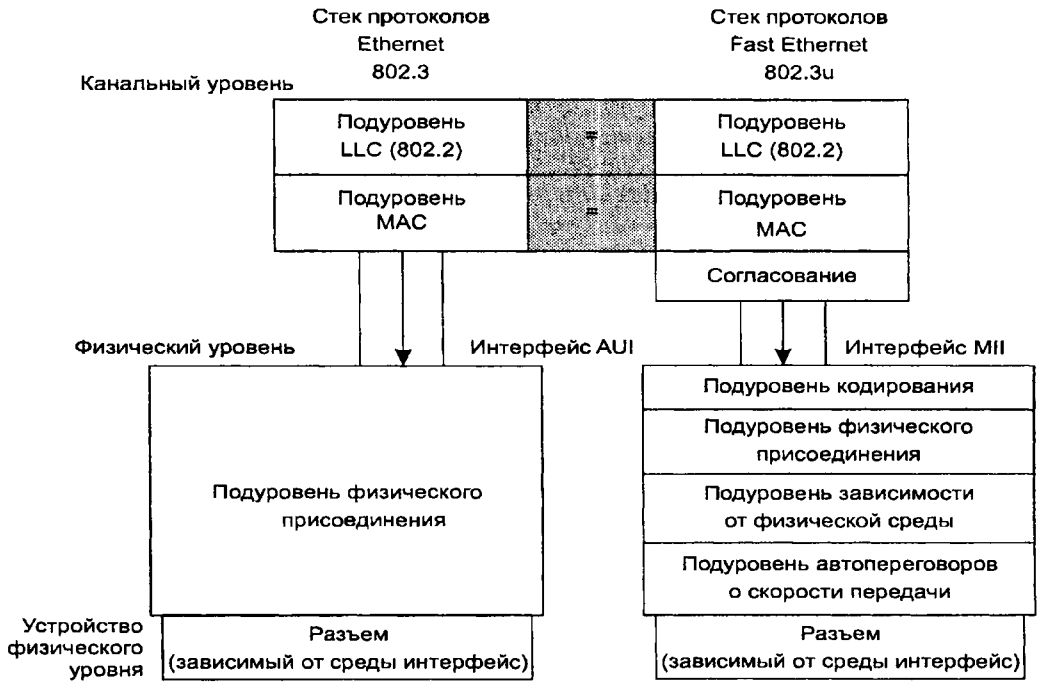
Изменился признак свободного состояния среды. Изменилось кодирование.

Появились «автопереговоры» и иерархия режимов для соединений на витой паре.

Появились хабы 1-го и 2-го класса и коммутаторы.

3 варианта кабельных систем:

- 100Base-FX - волоконно-оптический многомодовый кабель (2 волокна);
- 100Base-TX - витая пара категории 5 (2 пары);
- 100Base-T4 - витая пара категории 3,4,5 (4 пары).



Отличия Fast Ethernet от Ethernet

Технология Gigabit Ethernet

IEEE 802.3z -1998 (оптика).

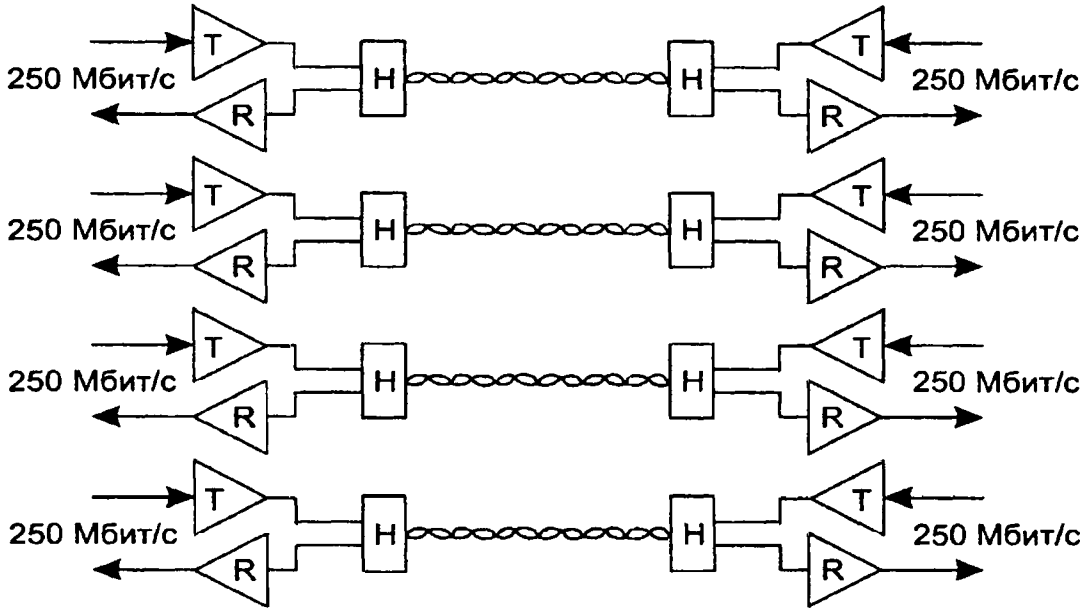
Форматы кадров не изменились. Временные параметры уменьшились в 10 раз (битовый интервал стал 10 нс, однако, измеренные в битовых интервалах не изменились).

Изменился признак свободного состояния среды. Изменилось кодирование.

«Автопереговоры» и иерархия режимов для соединений на витой паре.

Поддерживаются все прежние и новые варианты кабельных систем:

- 1000Base-SX/LX - волоконно-оптический одно- или -многомодовый кабель (62,5/125; 50/125);
- 1000Base-CX – сбалансированный медный кабель (волновое сопротивление 150 Ом), макс.-25м;
- 1000Base-T - витая пара категории не ниже 5 (4 пары, дуплекс). (IEEE 802.3ab – 1999г.).



Gigabit Ethernet (IEEE 802.3ab): двунаправленная передача по 4 парам UTP категории 5

10Gigabit Ethernet. IEEE 802.3ae (2003г.)

10 Гигабит Ethernet включает в себя семь стандартов физической среды для LAN, MAN и WAN. Он описывается поправкой IEEE 802.3ae и должен войти в следующую ревизию стандарта IEEE 802.3.

10GBASE-CX4 — Технология 10 Гигабит Ethernet для коротких расстояний (до 15 метров), используется медный кабель CX4 и коннекторы InfiniBand.

10GBASE-SR — Технология 10 Гигабит Ethernet для коротких расстояний (до 26 или 82 метров, в зависимости от типа кабеля), используется многомодовое оптоволокно. Он также поддерживает расстояния до 300 метров с использованием нового многомодового оптоволокна (2000 МГц/км).

10GBASE-LX4 — использует уплотнение по длине волны для поддержки расстояний от 240 до 300 метров по многомодовому оптоволокну. Также поддерживает расстояния до 10 километров при использовании одномодового оптоволокна.

10GBASE-LR и **10GBASE-ER** — эти стандарты поддерживают расстояния до 10 и 40 километров соответственно.

10GBASE-SW, **10GBASE-LW** и **10GBASE-EW** — Эти стандарты используют физический интерфейс, совместимый по скорости и формату данных с интерфейсом OC-192 / STM-64 SONET/SDH.

10GBASE-T, IEEE 802.3an-2006 — принят в июне 2006 года после 4 лет разработки. Использует экранированную витую пару. Расстояния — до 100 метров.

Тема лекции: **Коммутируемые локальные сети**

План лекции:

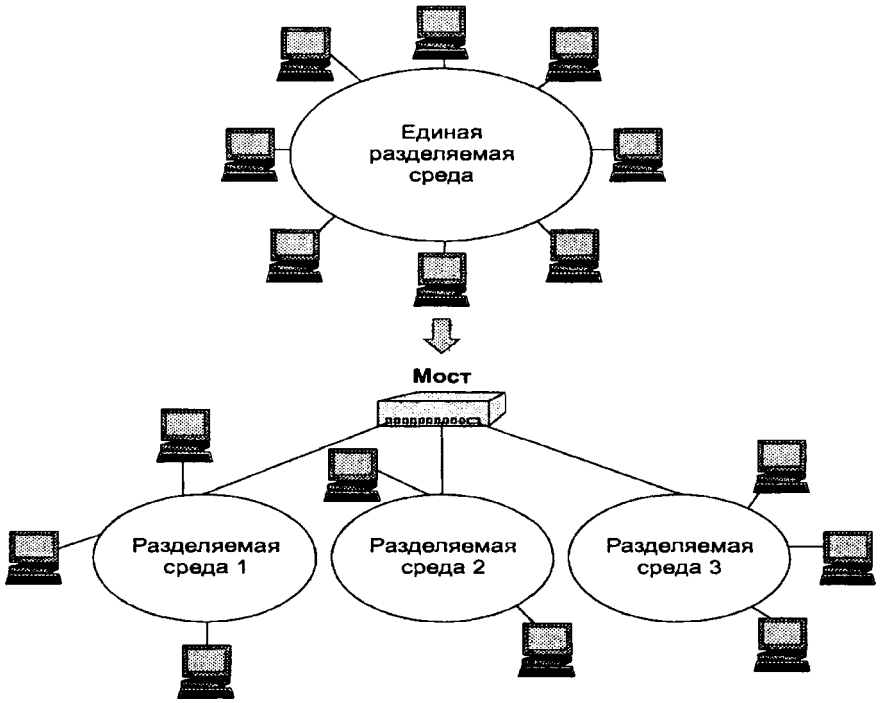
1. Логическая структуризация сети.
2. Алгоритм прозрачного моста.
2. Алгоритм покрывающего дерева.
3. Устройство коммутаторов.
4. Режимы работы коммутаторов.
5. Функции коммутаторов.

Логическая структуризация сети с помощью мостов и коммутаторов

Недостатки сетей на разделяемой среде: **дефицит пропускной способности** и **плохая масштабируемость**.

Исправить эти недостатки помогает **логическая структуризация сети**: **сегментирование** (разделение) единой среды на несколько сегментов, которые связаны такими **устройствами как мосты, коммутаторы, маршрутизаторы**.

Эти устройства передают кадры с одного своего порта на другой, анализируя помещенный в них адрес назначения. **Мосты и коммутаторы** делают это на основе анализа **MAC-адресов** (маршрутизаторы – на основе IP-адресов).



Логическая структуризация сети с помощью мостов и коммутаторов

Преимущества логической структуризации сети

1. Повышение производительности.
2. Повышение гибкости сети (обмен данными в гетерогенных сетях).
3. Повышение безопасности данных (изоляция одних сегментов сети или хостов от других).
4. Повышение управляемости сети (проблемы локализуются внутри сегмента. Сегменты образуют логические домены управления сетью).

5. Улучшение масштабируемости сети.

Алгоритм прозрачного моста IEEE 802.1D

Локальные сети разделяют на логические сегменты с помощью мостов и коммутаторов, которые являются функциональными аналогами.

Основное отличие коммутатора от моста: мост обрабатывает кадры последовательно, а коммутатор параллельно. Однако оба продвигают кадры на основе единого **алгоритма прозрачного моста**, описанного в стандарте **IEEE 802.1D.**

Термин «прозрачный» указывает на то, что устройства локальной сети (сетевые карты хостов, хабы, маршрутизаторы) функционируют «не замечая» присутствия в сети мостов (коммутаторов).

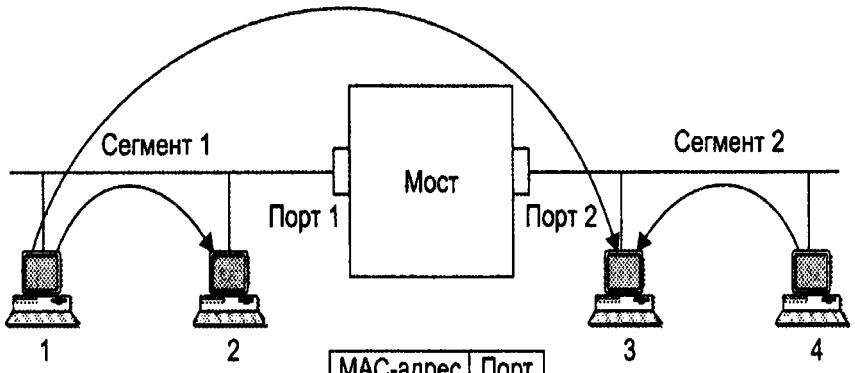
Алгоритм прозрачного моста не зависит от технологии локальной сети, в которой устанавливается мост/коммутатор. Поэтому коммутаторы Ethernet работают аналогично коммутаторам FDDI или Token Ring.

Коммутатор строит свою **адресную таблицу** на основании пассивного наблюдения за трафиком, циркулирующим в подключенных к его портам сегментам. При этом коммутатор учитывает адреса источников кадров данных, поступающих на порты коммутатора. По адресу источника коммутатор делает вывод о принадлежности узла источника конкретному сегменту сети.

Принцип работы прозрачного моста

Одновременно с передачей кадра на соответствующие порты коммутатор изучает адрес источника кадра и делает запись о его принадлежности к тому или иному сегменту в своей адресной таблице. Эту таблицу также называют (редко) таблицей фильтрации или таблицей маршрутизации.

Процесс обучения коммутатора никогда не заканчивается и происходит одновременно с продвижением и фильтрацией кадров. Коммутатор постоянно следит за адресами буферизуемых кадров, чтобы автоматически адаптироваться к изменениям в сети.



MAC-адрес	Порт
1	1
2	1
3	2
4	2

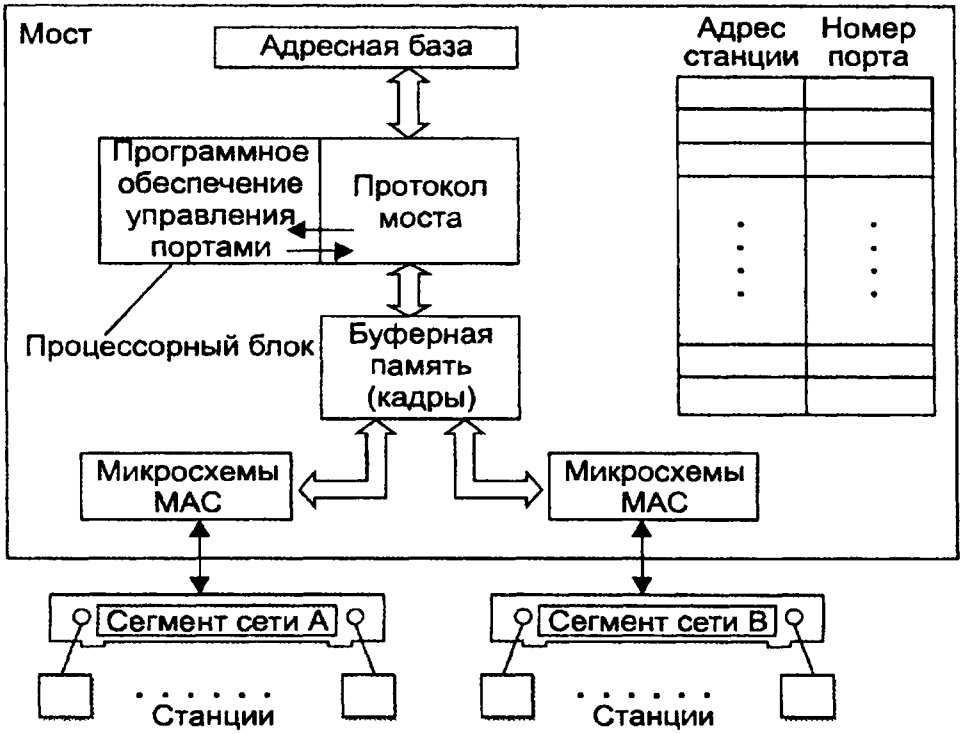
Принцип работы прозрачного моста

Входы адресной таблицы могут быть динамическими, создаваемыми в процессе самообучения коммутатора (имеют ограниченное время жизни), и статическими, создаваемыми вручную администратором (время жизни – не ограничено).

Кадры с широковещательными (broadcast) адресами, как и кадры с неизвестными адресами назначения передаются на все его порты (это режим затопления сети – flooding). Т.е. коммутаторы работают в едином широковещательном (broadcast) домене.

Однако при сбоях, некорректной работе одно из сетевых устройств может начать генерировать кадры с широковещательным адресом, а коммутатор начнет «тиражировать» ошибочный трафик на всех своих портах. Такая ситуация называется широковещательным штормом (broadcast storm).

Коммутаторы, по умолчанию, не защищают от широковещательного шторма. Это, по умолчанию, делают маршрутизаторы, которые делят сеть на несколько широковещательных (broadcast) доменов.



Структура моста/коммутатора

Топологические ограничения коммутаторов

В локальной сети на основе коммутаторов **не должно быть (активных)** петель (циклов). Т.е. должна поддерживаться топология – дерево.

Паразитные (возникшие в крупной сети непреднамеренно) или избыточные (созданные для повышения надежности и/или пропускной способности) связи необходимо блокировать, то есть переводить в неактивное состояние. В каждом коммутаторе для этих целей реализован стандартный алгоритм покрывающего дерева (Spanning Tree Algorithm, STA), реализуемый соответствующим

протоколом (Spanning Tree Protocol, STP).

Алгоритм покрывающего дерева

Протокол STP используется в коммутируемых сетях для создания логической топологии, свободной от петель, из физической топологии, в которой петли присутствуют. Алгоритм покрывающего дерева позволяет коммутаторам автоматически определять древовидную конфигурацию связей в сети при произвольном соединения портов между собой.

Поддерживающие алгоритм STA коммутаторы автоматически создают активную древовидную конфигурацию связей (то есть связную конфигурацию без петель) на множестве всех связей сети. Такая конфигурация называется покрывающим деревом - Spanning Tree (иногда ее называют основным деревом), и ее название дало имя всему алгоритму. Алгоритм Spanning Tree описан в стандарте IEEE 802.1D, том же стандарте, который определяет принципы работы прозрачных мостов.

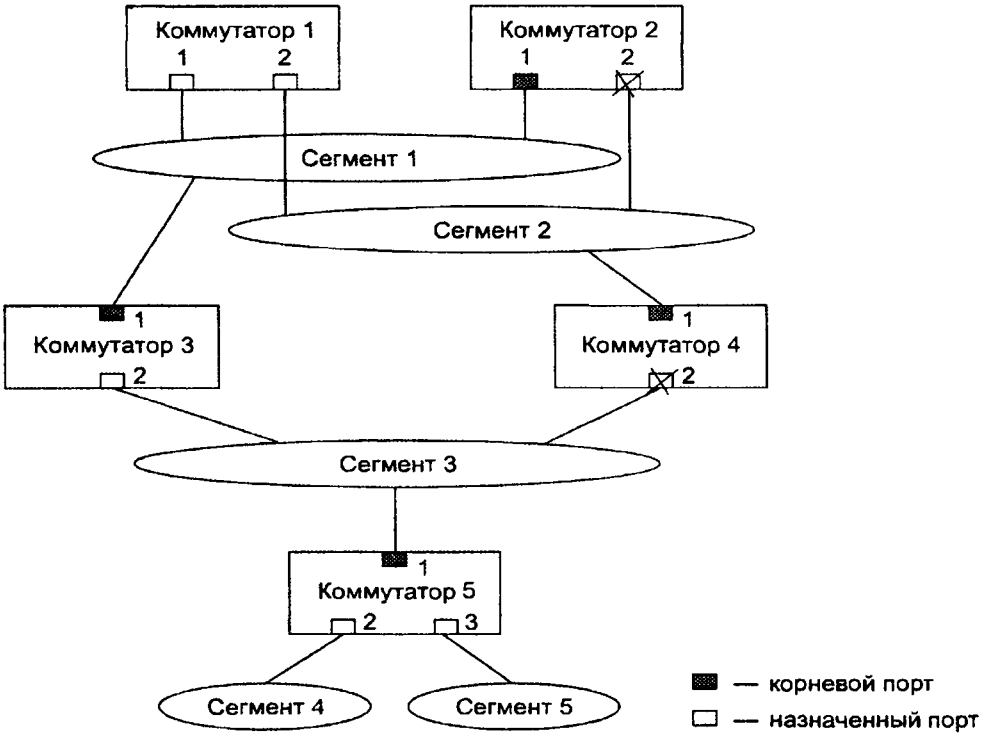
Коммутаторы находят покрывающее дерево адаптивно, с помощью обмена служебными пакетами. Реализация в коммутаторе алгоритма STA очень важна для работы в больших сетях - если коммутатор не поддерживает этот алгоритм, то администратор должен самостоятельно определить, какие порты нужно перевести в заблокированное состояние, чтобы исключить петли. К тому же при отказе какого-либо кабеля, порта или коммутатора администратор должен, во-первых, обнаружить факт отказа, а во-вторых, ликвидировать последствия отказа, переведя резервную связь в рабочий режим путем активизации некоторых портов.

При поддержке коммутаторами сети протокола Spanning Tree отказы обнаруживаются автоматически, за счет постоянного тестирования связности сети служебными пакетами. После обнаружения потери связности протокол строит новое покрывающее дерево, если это возможно, и сеть автоматически восстанавливает работоспособность.

Алгоритм Spanning Tree определяет активную конфигурацию сети за три этапа:

1. В сети определяется корневой коммутатор (root switch), от которого строится дерево. Корневой коммутатор может быть выбран автоматически или назначен администратором. При автоматическом выборе корневым становится коммутатор с меньшим значением MAC - адреса его блока управления.
2. Для каждого коммутатора определяется корневой порт (root port) - это порт, который имеет по сети кратчайшее расстояние до корневого коммутатора.
3. Для каждого сегмента сети выбирается назначенный порт (designated port) - это порт, который имеет кратчайшее расстояние от данного сегмента до корневого коммутатора.

После этого каждый коммутатор блокирует остальные порты, которые не попали в эти два класса портов. Можно математически доказать, что при таком выборе активных портов в сети исключаются петли и оставшиеся связи образуют покрывающее дерево (если оно может быть построено при существующих связях в сети).



Построение покрывающего дерева

Единственный порт, соединяющий каждый коммутатор с корневым коммутатором, и единственный порт, соединяющий каждый сегмент сети с корневым коммутатором, выбираются по критерию расстояния.

Расстояние до корня определяется как суммарное условное время на передачу одного бита данных от порта данного коммутатора до порта корневого коммутатора. При этом считается, что время внутренних передач данных (с порта на порт) коммутатором пренебрежимо мало, а учитывается только время на передачу данных по сегментам сети, соединяющим коммутаторы.

Условное время сегмента рассчитывается как время, затрачиваемое на передачу одного бита информации в 10 наносекундных единицах между непосредственно связанными по сегменту сети портами. Так, для сегмента Ethernet это время равно 10 условным единицам, а для сегмента Token Ring 16 Мбит/с - 6,25.

Для автоматического определения начальной активной конфигурации дерева все коммутаторы сети после их инициализации начинают периодически обмениваться специальными пакетами, называемыми протокольными блоками данных моста - BPDU (Bridge Protocol Data Unit).

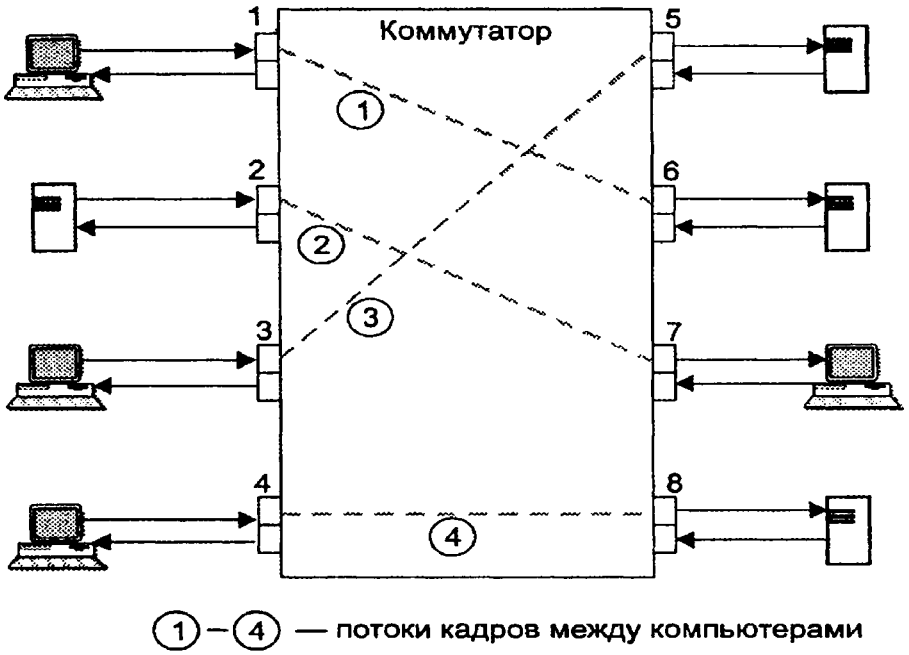
Устройство коммутаторов

Производительность коммутаторов (миллионы кадров в секунду) на несколько порядков выше, чем мостов. Добиваются высокой производительности за счет применения параллельной обработки кадров и/или конвейеризации.

В основе архитектуры коммутатора лежат быстродействующий узел обмена и набор процессоров портов (по числу портов маршрутизатора).

Узел обмена может строится на основе:

1. Коммутационной матрицы.
2. Общей шины.
3. Разделяемой многовходовой памяти.
4. Комбинации решений 1-3.



Параллельная передача кадров коммутатором

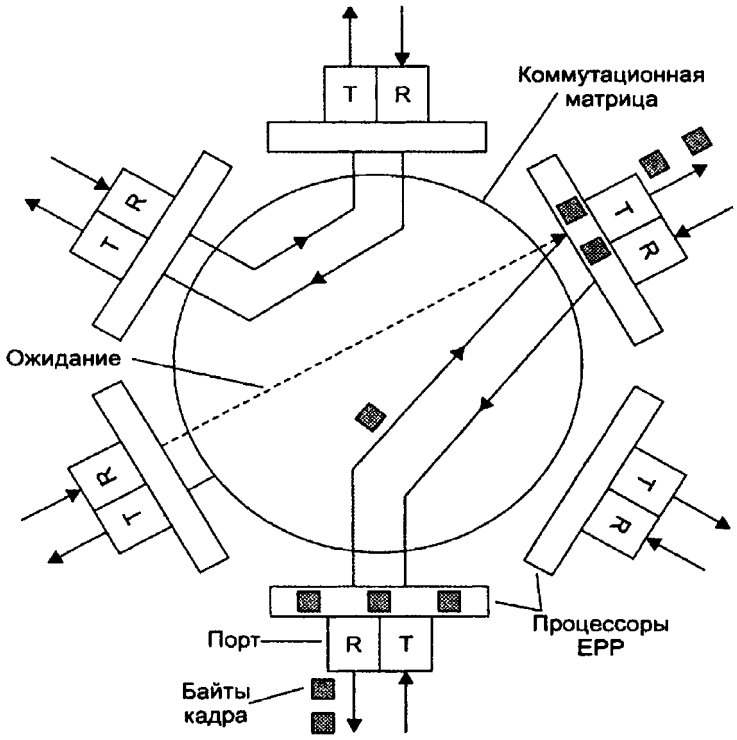
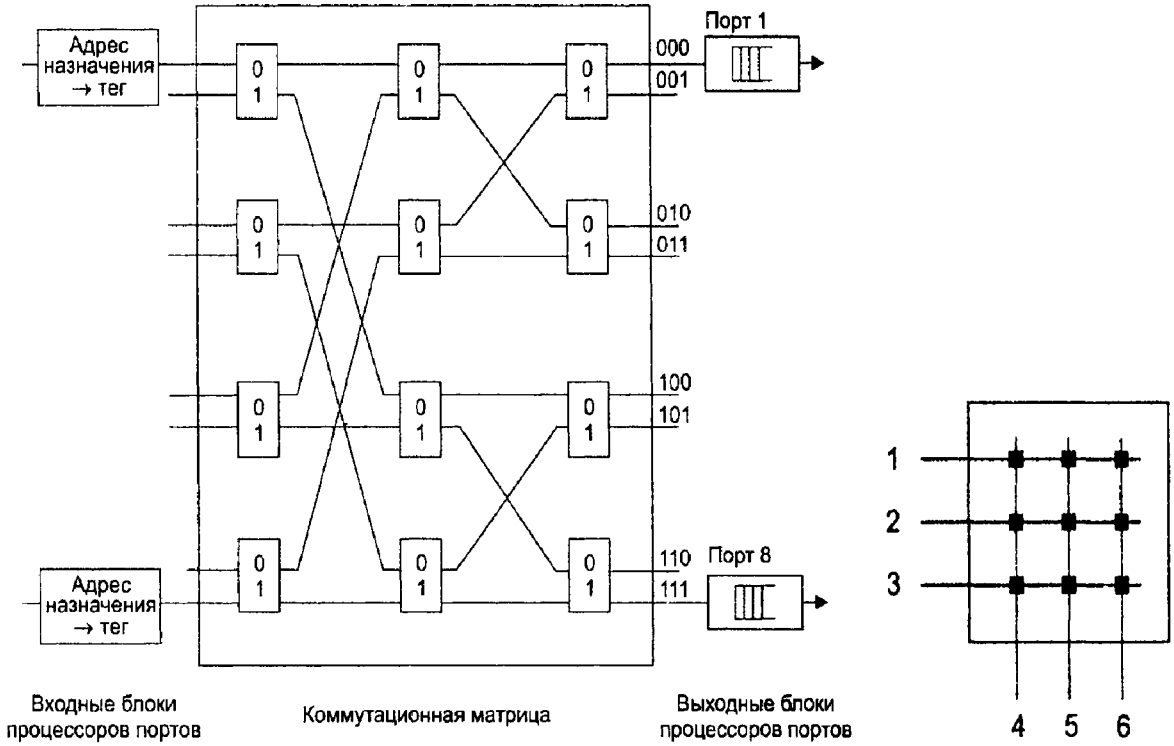
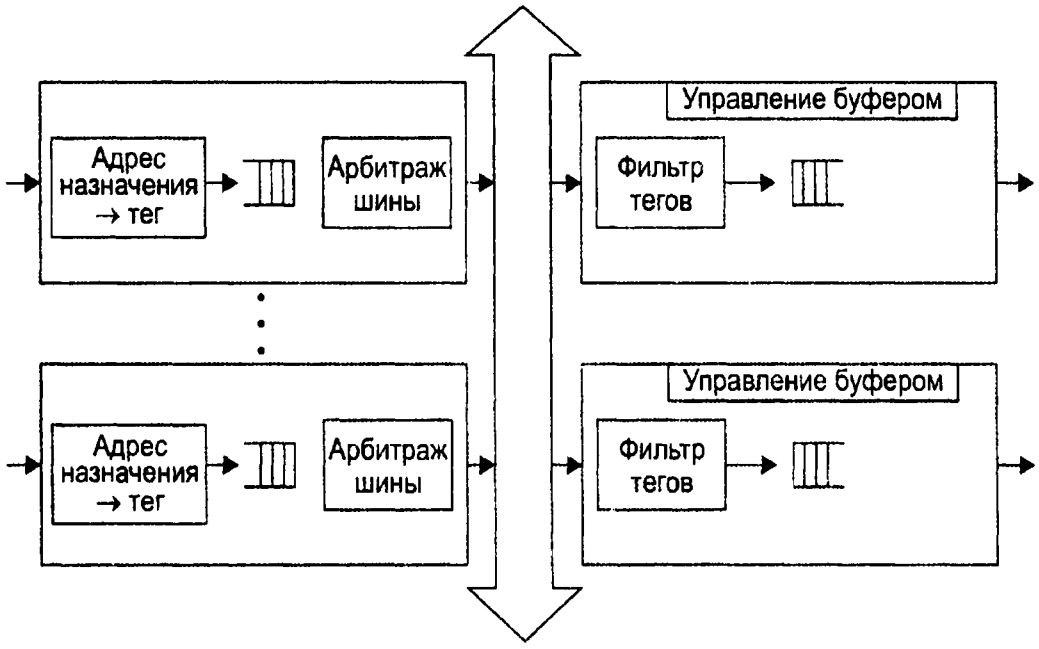


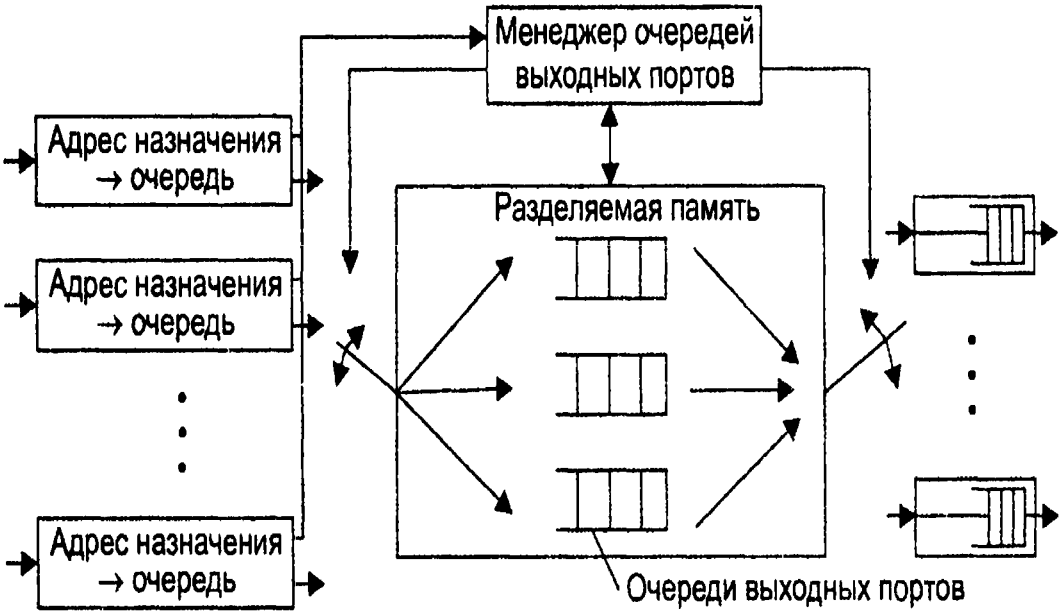
Схема первых коммутаторов фирмы Kalpana



Коммутационная матрица



Архитектура коммутатора с общей шиной



Архитектура коммутатора с разделяемой памятью

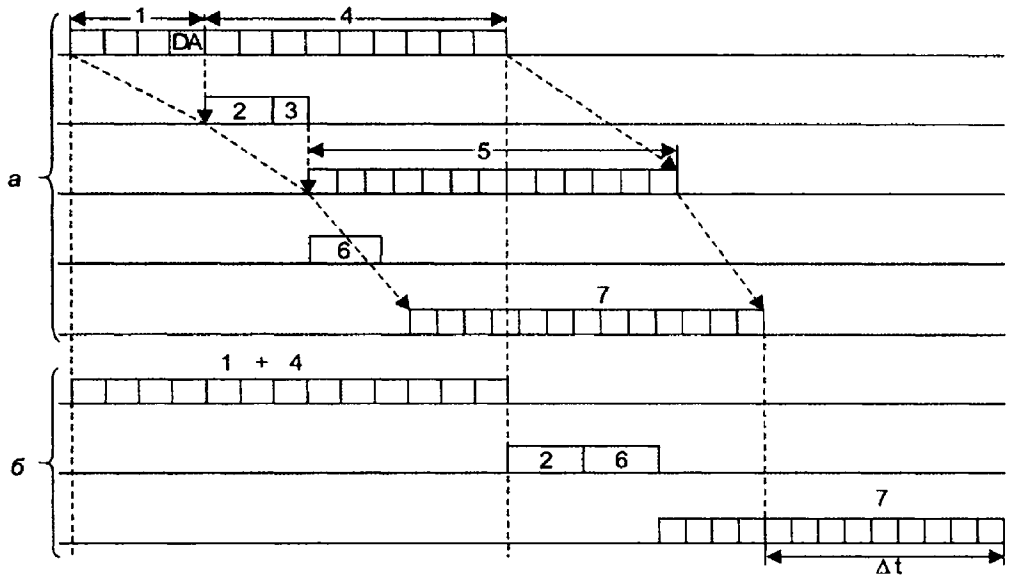
Этапы обработки кадра коммутатором:

1. Прием первых байтов кадра процессором входного порта (с адресом назначения).
2. Поиск адреса назначения в адресной таблице коммутатора (кэше процессора порта).
3. Коммутация матрицы.
4. Прием остальных байтов кадра процессором входного порта.
5. Передача байтов кадра процессору выходного порта через коммутационную матрицу.
6. Получение доступа к среде процессором выходного порта.
7. Передача байтов кадра процессором выходного порта в сеть.

Режимы работы коммутаторов

Два основных режима работы коммутатора:

1. Режим полной буферизации кадра с последовательным выполнением всех этапов.
2. Режим коммутации «**на лету**» (**on the fly**) или «**напролет**» (**cut-through**), который представляет собой, по сути, конвейерную обработку кадра, когда частично совмещаются во времени несколько этапов его передачи.



Экономия времени при конвейерной (cut-through) обработке кадра:
а) – конвейерная обработка; б) – обычная обработка с полной буферизацией.

Сравнение возможностей коммутаторов в разных режимах

Функция	На лету	С буферизацией
Защита от плохих кадров	Нет	Да
Поддержка разнородных сетей (Ethernet, Token Ring, FDDI, ATM)	Нет	Да
Задержка передачи кадров	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

Задержка передачи кадра: «на лету» - времени при конвейерной (cut-through) обработке кадра: 5-40 мкс; при полной буферизации: 50-20 мкс (для кадров минимальной длины).

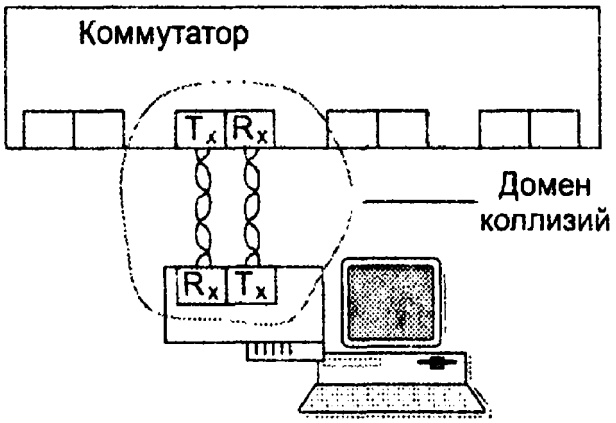
Неблокирующие коммутаторы

Коммутатор называют **неблокирующим**, если он может передавать кадры через свои порты с той же скоростью, с которой они на них поступают.

Устойчивый неблокирующий режим требует общей производительности не меньше полусуммы максимальной производительности портов.

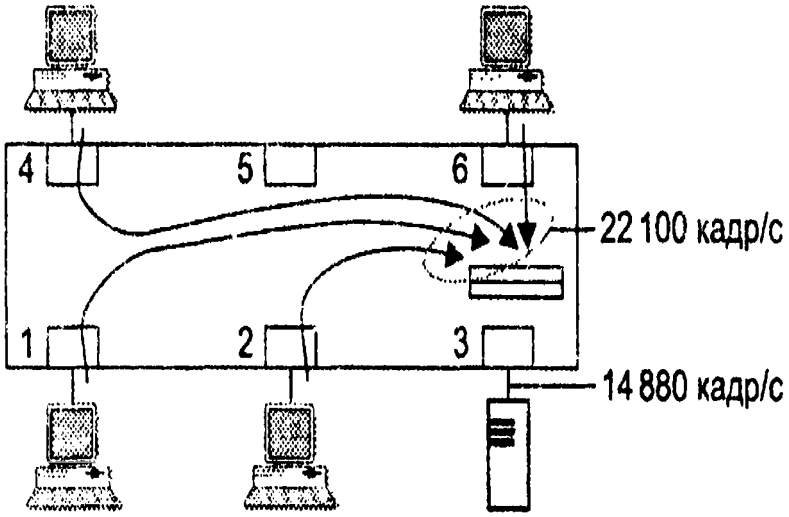
Мгновенный неблокирующий режим требует общей производительности не меньше суммы максимальной производительности портов.

Микросегментация и дуплексный режим



Домен коллизий, образуемый компьютером и портом коммутатора

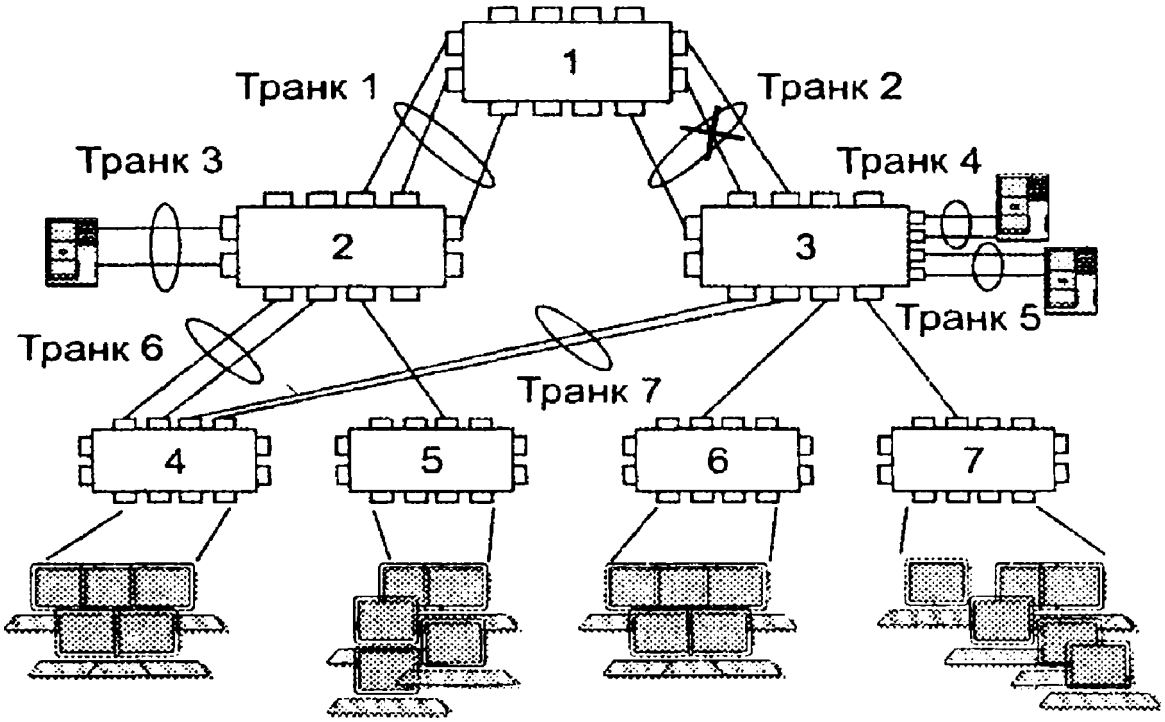
Подключение к портам коммутатора не сегментов а отдельных компьютеров называется микросегментацией.



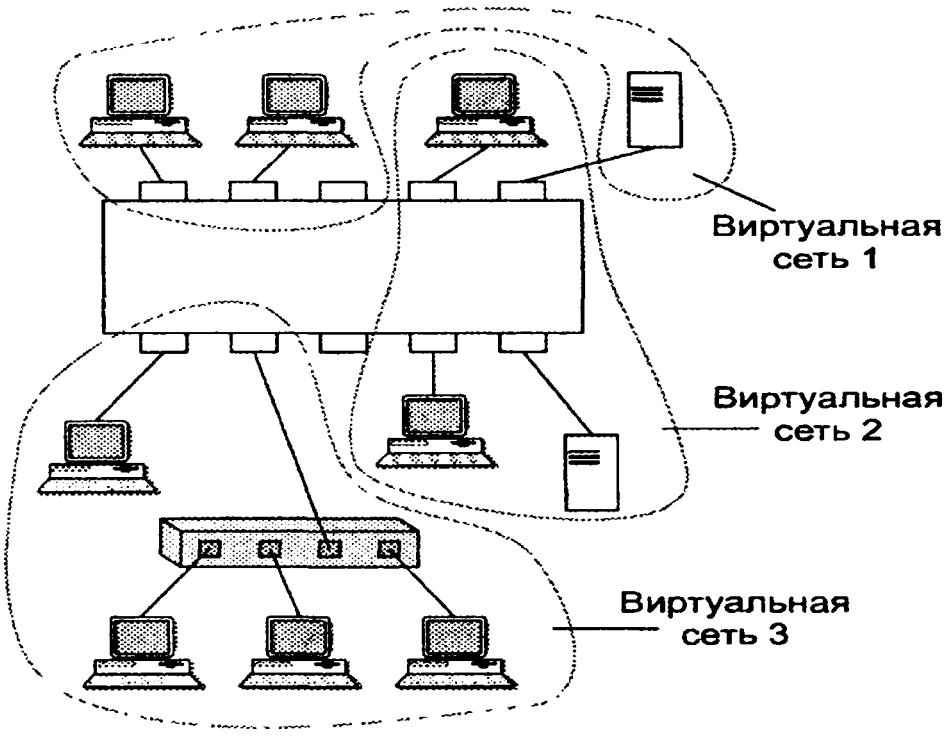
Переполнение буфера порта из-за несбалансированности трафика

Функции коммутаторов

1. Борьба с перегрузками (полудуплекс, разделяемая среда): метод обратного давления и метод агрессивного захвата среды.
2. Трансляция протоколов канального уровня.
3. Фильтрация трафика.
4. Агрегирование линий связи (**транки** и логические каналы).
5. **Виртуальные локальные сети (Virtual LAN, VLAN)** - группа узлов в сети, трафик которой, в том числе и широковещательный, на канальном уровне (без участия маршрутизаторов!) полностью изолирован от трафика других узлов сети.



Транки: агрегирование физических каналов



Пример виртуальных сетей (VLAN), построенных на одном коммутаторе

Технологии беспроводных локальных сетей

План лекции:

1. Перечень используемых технологий, их назначение и особенности применения.
2. Технические и юридические аспекты использования эфира.
3. Сети Wi-Fi.
4. Сети WiMAX.
5. Персональные беспроводные сети (Bluetooth, ZigBee, UWB).

Технологии беспроводных сетей

1. IEEE 802.11(a,b,g,n) - Wi-Fi (радиоэзернет) – LAN, локальные сети.

2. IEEE 802.16 WiMax:

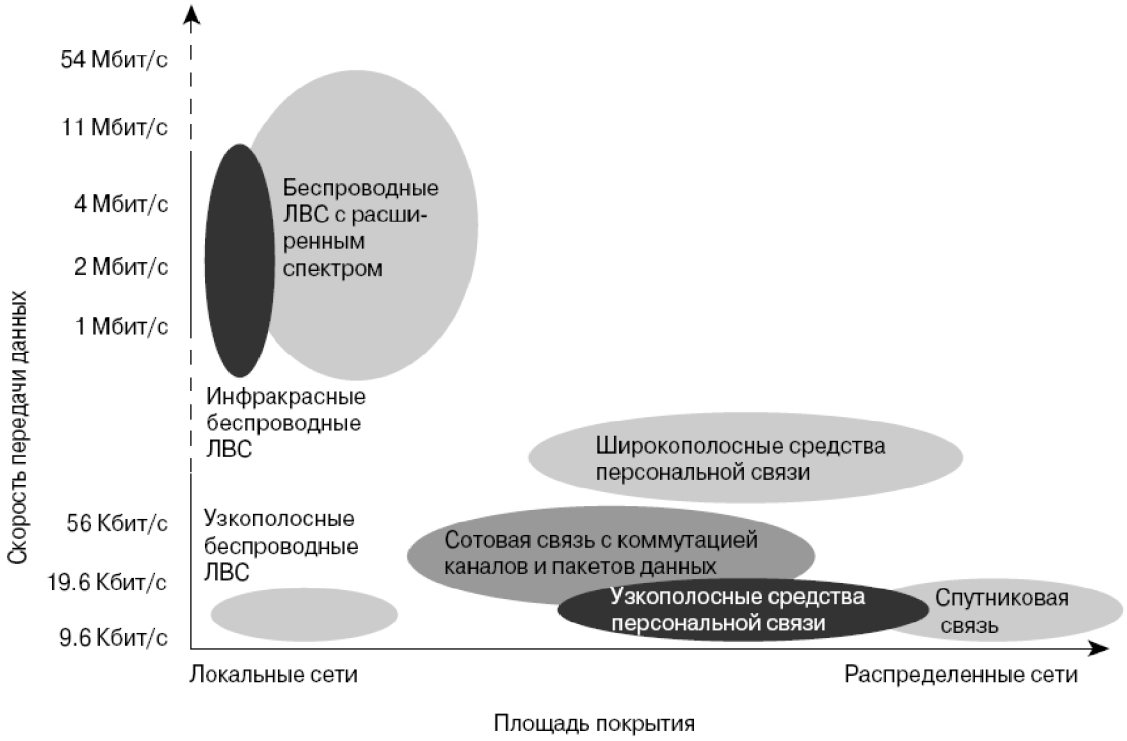
а) **IEEE 802.16d** стационарные, как сети доступа, + MAN;

б) **IEEE 802.16e** мобильные, как сотовые сети 4 поколения.

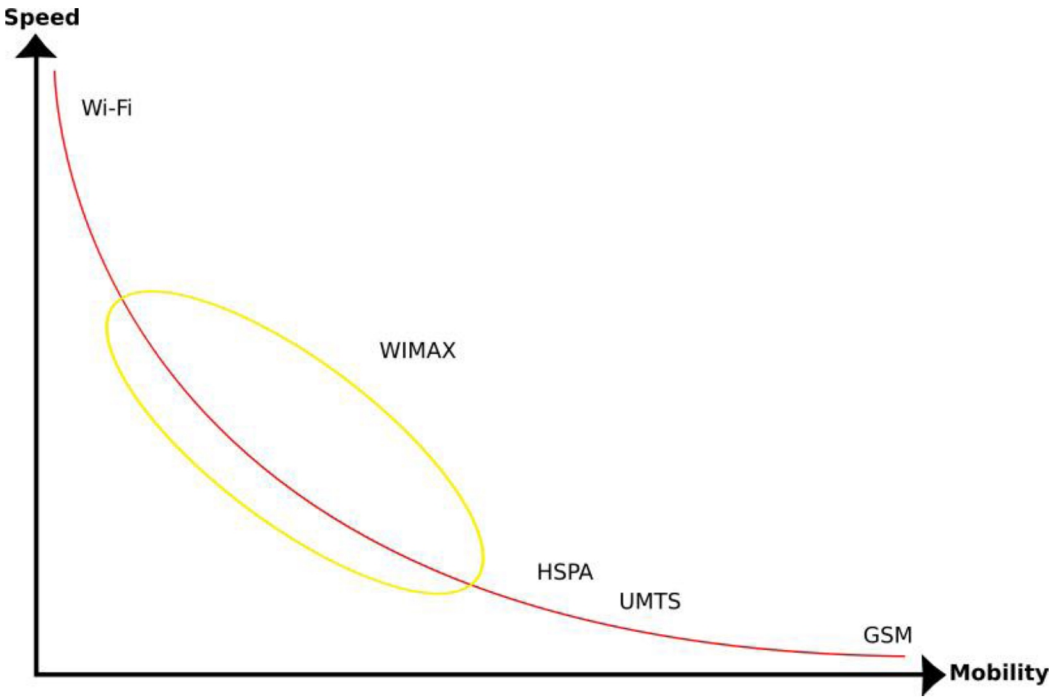
3. IEEE 802.15.1 Bluetooth –персональные (микро или пико) сети.

4. IEEE 802.15.4 ZigBee - распределенные сети.

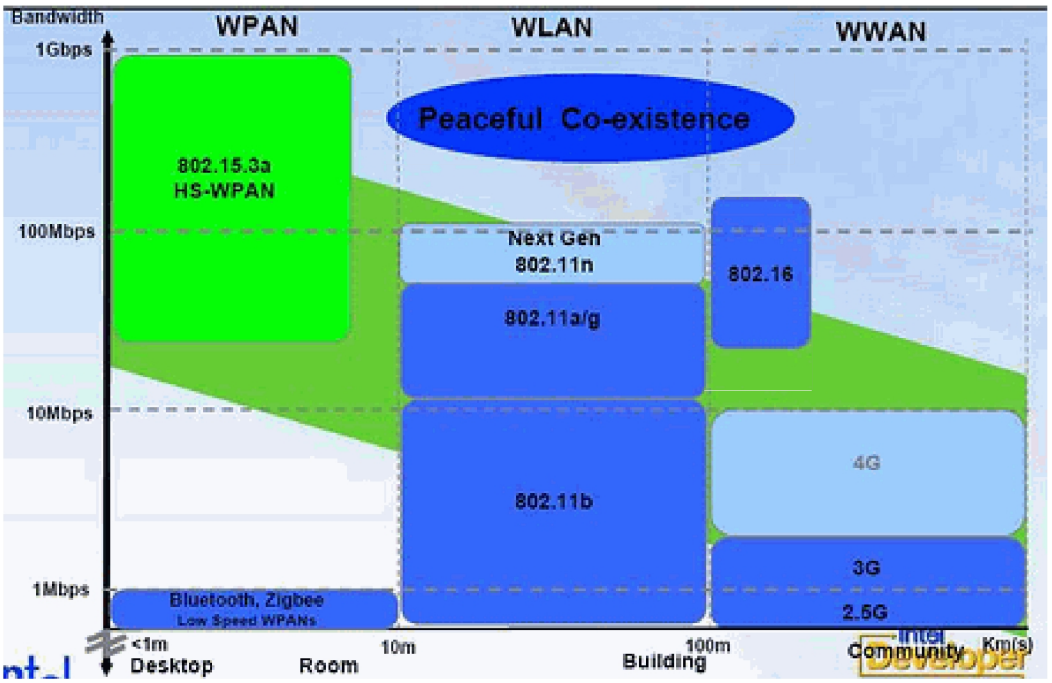
5. IEEE 802.15.3a UWB



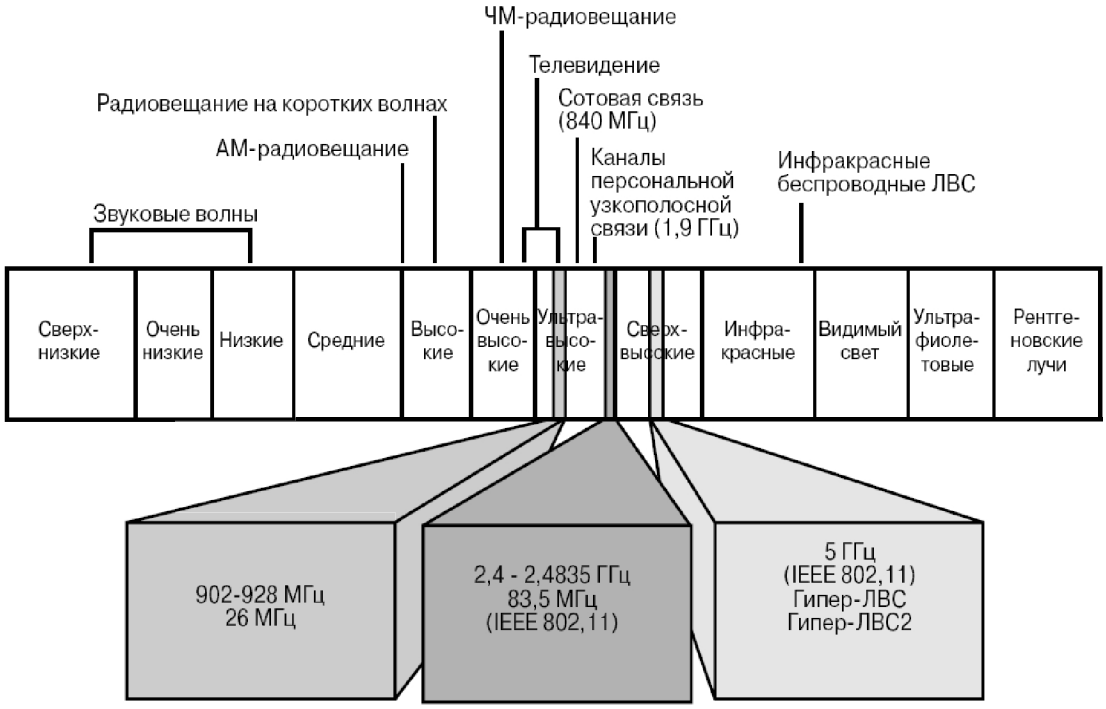
Сравнение технологий беспроводных сетей по производительности



Сравнение технологий беспроводных сетей по мобильности



Сравнение технологий беспроводных сетей площади покрытия



Диапазоны, не требующие лицензирования (США)



Пример распределение интенсивности сигнала в сетях WiFi

WiFi - IEEE 802.11(a,b,g,n)

1. Назначение: LAN, объединение LAN беспроводным мостом.

2. Характеристика стека протоколов: физический и канальный (только MAC) уровни модели OSI.

3. Дальность / скорость: ~100м / 1–600Мбит/с (7ГБит/с)

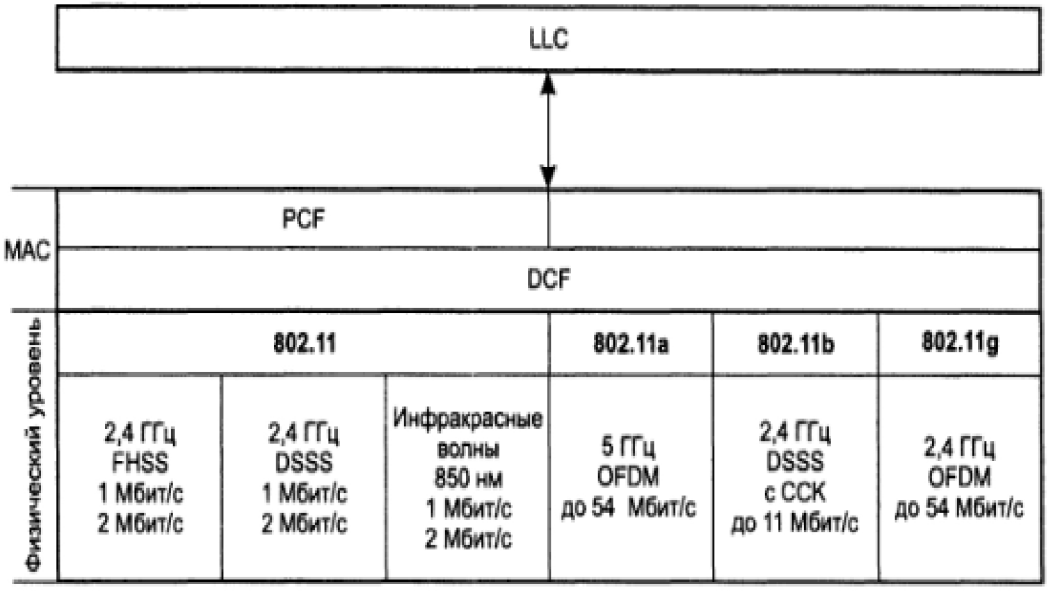
4. Диапазон частот: 2,4 ГГц или/и 5ГГц. (+60ГГц в будущем)

5. Топология: «звезда» / «дерево» (AdHoc – произвольная)

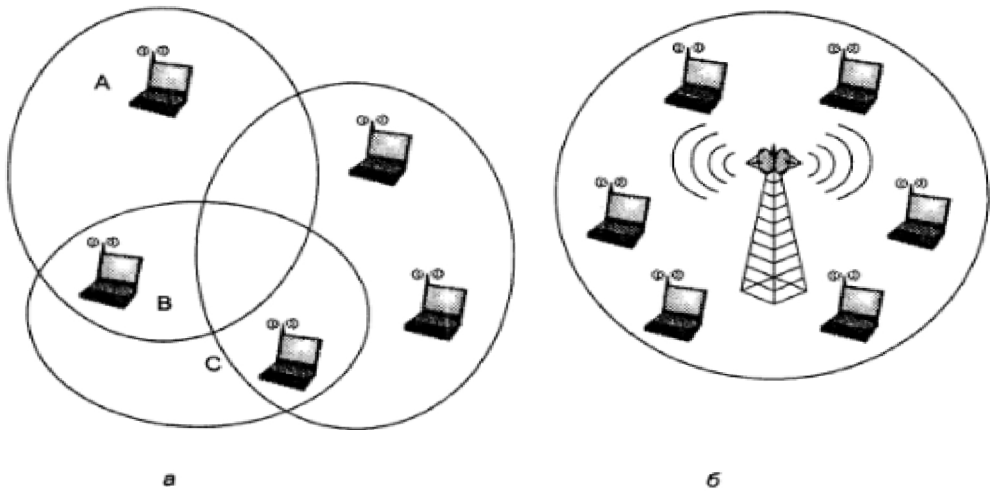
6. Доступ к среде / мультиплексирование: CSMA/CA временное разделение (+пространственное) для конкретного SSID, частотное (+территориальное) разделение для разных SSID
7. Безопасность: WEP, WPA, WPA2/IEEE802.11i
8. Другие особенности: MIMO, PoE
9. Перспективы: Wi-Fi Direct, Wireless Gigabit Alliance (WiGig)

Параметры основных стандартов WiFi

Параметр	**802.11 DSSS**	**802.11 FHSS**	**802.11b**	**802.11a**	**802.11g**	**802.11n**
Частотный диапазон (ГГц)	2,4	2,4	2,4	5	2,4	2,4 / 5
Макс. скорость передачи данных (Мбит/с)	2	2	11	54	54	600
Технология	DSSS	FHSS	CCK	OFDM	OFDM	OFDM
Тип модуляции (для макс. скорости передачи)	QPSK	GFSK	QPSK	64-QAM	64-QAM	64-QAM
Число неперекрывающихся каналов	3	3	3	15	3	?/?



Стек протоколов IEEE 802.11



а *б*

Топология и режимы работы сети WiFi: а) режим AdHoc (+иллюстр. проблемы «скрытого терминала»); б) инфраструктурный режим (основной).

Быстродействие сетей WiFi (802.11n-draft)

Стандарт беспроводной связи	Быстродействие, Мбит/с			
	Канал 20 МГц		Канал 40 МГц	
	1 поток	2 потока	1 поток	2 потока
802.11b, 2,4 ГГц	1; 2; 5,5; 11			
802.11a, 5 ГГц	6; 9; 12; 18; 24; 36; 48; 54			
802.11g, 2,4 ГГц	1; 2; 6; 9; 12; 18; 24; 36; 48; 54			
802.11n, 2,4 ГГц	6,5; 13; 19,5; 26; 39; 52; 58,5; 65	13; 26; 39; 52; 78; 104; 117; 130		
802.11n, 5 ГГц	6,5; 13; 19,5; 26; 39; 52; 58,5; 65	13; 26; 39; 52; 78; 104; 117; 130	13,5; 27; 40,5; 54; 81; 108; 121,5; 135	27; 54; 81; 108; 162; 216; 243; 270
802.11n, 2,4 и 5 ГГц	7,2; 14,4; 21,7; 28,9; 43,3; 57,8; 65; 72,2	14,4; 28,9; 43,3; 57,8; 86,7; 115,6; 130; 144,4	15; 30; 45; 60; 90; 120; 135; 150	30; 60; 90; 120; 180; 240; 270; 300

Беспроводные локальные сети 12

Параметры IEEE 802.11n (окончательные)

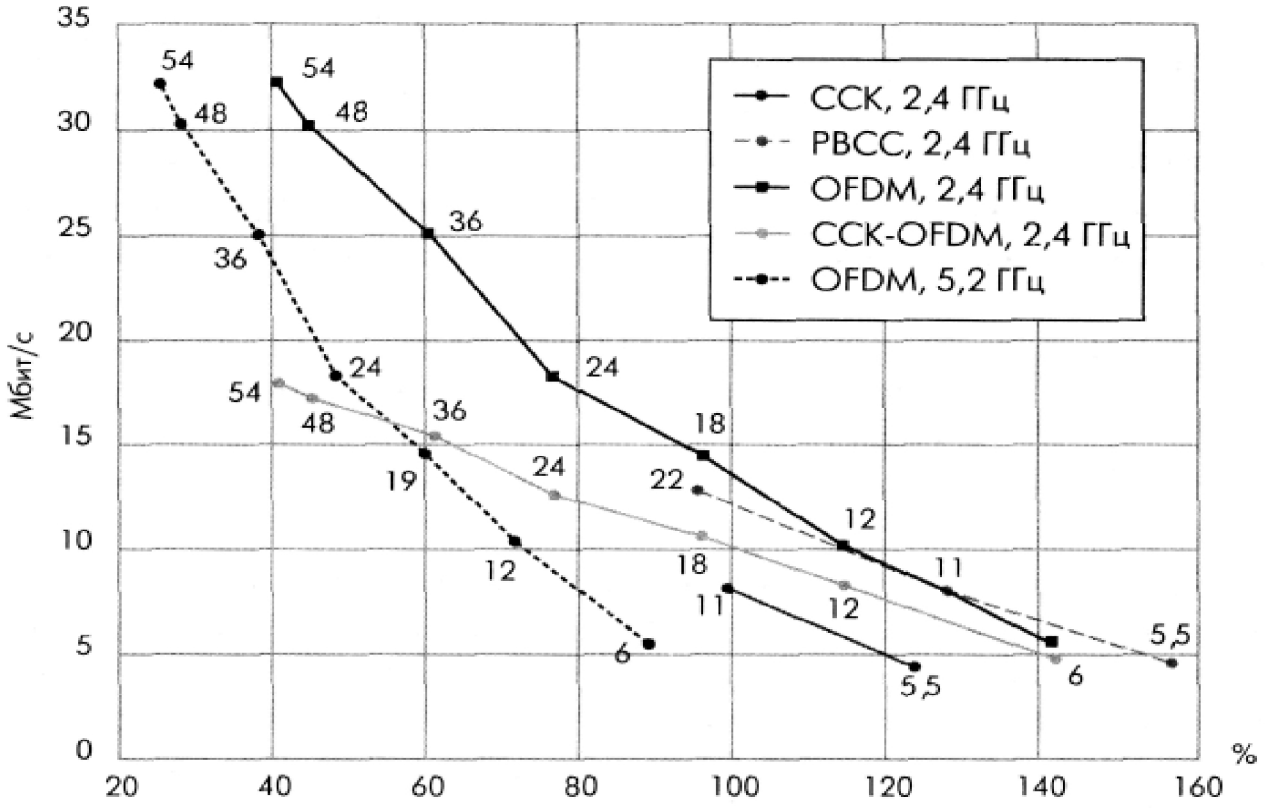
MCS Index	Spatial Streams	Modulation Type	Coding Rate	Data Rate Mb/s			
				20 MHz channel		40 MHz channel	
				800ns GI	400ns GI	800ns GI	400ns GI
0	1	BPSK	1/2	6.50	7.20	13.50	15.00
1	1	QPSK	1/2	13.00	14.40	27.00	30.00
2	1	QPSK	3/4	19.50	21.70	40.50	45.00
3	1	16-QAM	1/2	26.00	28.90	54.00	60.00
4	1	16-QAM	3/4	39.00	43.30	81.00	90.00
5	1	64-QAM	2/3	52.00	57.80	108.00	120.00
6	1	64-QAM	3/4	58.50	65.00	121.50	135.00
7	1	64-QAM	5/6	65.00	72.20	135.00	150.00
8	2	BPSK	1/2	13.00	14.40	27.00	30.00
9	2	QPSK	1/2	26.00	28.90	54.00	60.00
10	2	QPSK	3/4	39.00	43.30	81.00	90.00
11	2	16-QAM	1/2	52.00	57.80	108.00	120.00
12	2	16-QAM	3/4	78.00	86.70	162.00	180.00
13	2	64-QAM	2/3	104.00	115.60	216.00	240.00
14	2	64-QAM	3/4	117.00	130.00	243.00	270.00
15	2	64-QAM	5/6	130.00	144.40	270.00	300.00
...	3	…	…	…	…	…	…
23	3	64-QAM	5/6	195.00	216.60	405.00	450.00
...	4	…	…	…	…	…	…
31	4	64-QAM	5/6	260.00	288.90	540.00	600.00

Сравнение вариантов IEEE 802.11

802.11 Protocol	Release	Freq. (GHz)	Bandwidth (MHz)	Data rate per stream (Mbit/s)	Allowable MIMO streams	Modulation	Approximate indoor range[l]		Approximate Outdoor range[l]	
							(m)	(ft)	(m)	(ft)
	Jun 1997	2.4	20	1, 2	1	DSSS	20	66	100	330
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, 54	1	OFDM	35	115	120	390
		3.7[y]					--	--	5,000	16,000[y]
b	Sep 1999	2.4	20	1, 2, 5.5, 11	1	DSSS	38	125	140	460
g	Jun 2003	2.4	20	1, 2, 6, 9, 12, 18, 24, 36, 48, 54	1	OFDM, DSSS	38	125	140	460
n	Oct 2009	2.4/5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2[z]	4	OFDM	70	230	250	820[24]
			40	15, 30, 45, 60, 90, 120, 135, 150[z]			70	230	250	820[24]

• [y] IEEE 802.11y-2008 extended operation of 802.11a to the licensed 3.7 GHz band. Increased power limits allow for a range up 5000m. As of 2009, it is only being licensed in the United States by the FCC.

• [z] Assumes Short Guard interval (SGI) enabled, otherwise reduce each data rate by 10%.



Влияние технологии на скорость

WiMax - IEEE 802.16

1. Назначение: IEEE 802.16d - стационарные, как сети доступа, + MAN; **IEEE 802.16e** - мобильные, как сотовые сети 4 поколения

2. Характеристика стека протоколов: собственные стеки протоколов.

3. Дальность / скорость: ~1-10км / 30–75Мбит/с

4. Диапазон частот:

5. Топология: звезда

6. Доступ к среде / мультиплексирование:

7. Безопасность:

8. Перспективы: WiMax2 - IEEE 802.16m

Набор преимуществ присущ всему семейству WiMAX, однако его версии существенно отличаются друг от друга. Разработчики стандарта искали оптимальные решения как для фиксированного, так и для мобильного применения, но совместить все требования в рамках одного стандарта не удалось. Хотя ряд базовых требований совпадает, нацеленность технологий на разные рыночные ниши привела к созданию двух отдельных версий стандарта (вернее, их можно считать двумя разными стандартами). Каждая из спецификаций WiMAX определяет свои рабочие диапазоны частот, ширину полосы пропускания, мощность излучения, методы передачи и доступа, способы кодирования и модуляции сигнала, принципы повторного использования радиочастот и прочие показатели. А потому WiMAX-системы, основанные на версиях стандарта IEEE

802.16 e и d, практически несовместимы.

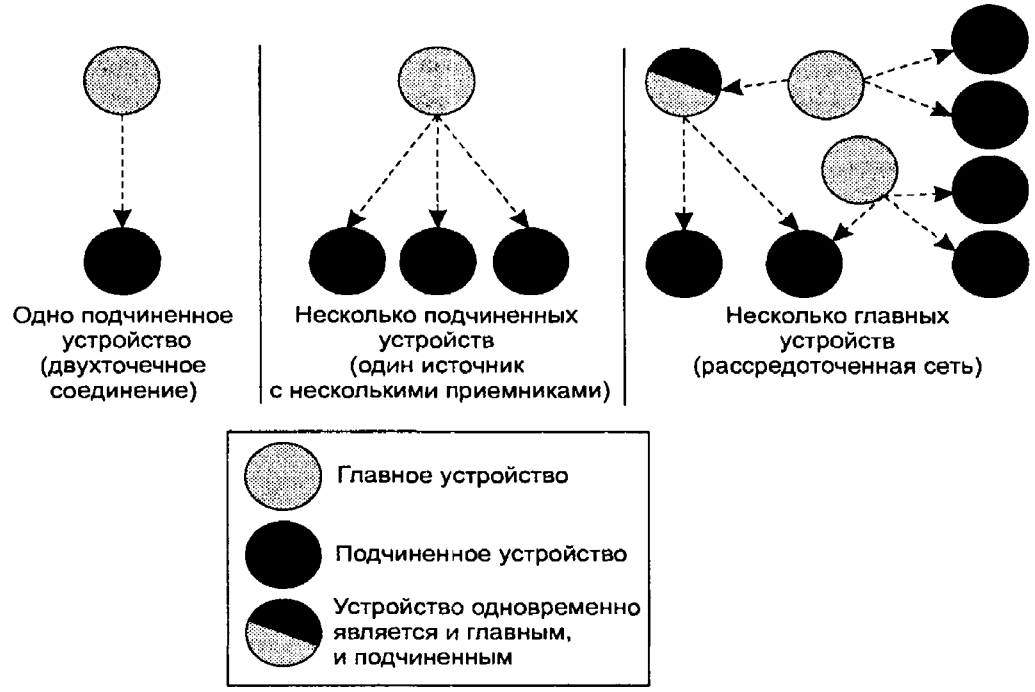
802.16-2004 (известен также как 802.16d и фиксированный WiMAX). Спецификация утверждена в 2004 году. Используется ортогональное частотное мультиплексирование (OFDM), поддерживается фиксированный доступ в зонах с наличием либо отсутствием прямой видимости. Пользовательские устройства представляют собой стационарные модемы для установки вне и внутри помещений, а также PCMCIA-карты для ноутбуков. В большинстве стран под эту технологию отведены диапазоны 3,5 и 5 ГГц. По сведениям WiMAX Forum, насчитывается уже порядка 175 внедрений фиксированной версии. Многие аналитики видят в ней конкурирующую или взаимодополняющую технологию проводного широкополосного доступа DSL.

802.16-2005 (известен также как 802.16e и мобильный WiMAX). Спецификация утверждена в 2005 году. Это — новый виток развития технологии фиксированного доступа (802.16d). Оптимизированная для поддержки мобильных пользователей версия поддерживает ряд специфических функций, таких как хэндовер и роуминг. Применяется масштабируемый OFDM-доступ (SOFDMA), возможна работа при наличии либо отсутствии прямой видимости. Планируемые частотные диапазоны для сетей Mobile WiMAX таковы: 2,3; 2,5; 3,4–3,8 ГГц. В мире реализованы несколько пилотных проектов, а недавно оператор Sprint анонсировал старт проекта национального масштаба. Конкурентами 802.16e являются все мобильные технологии третьего поколения (например, EV-DO, HSXPA).

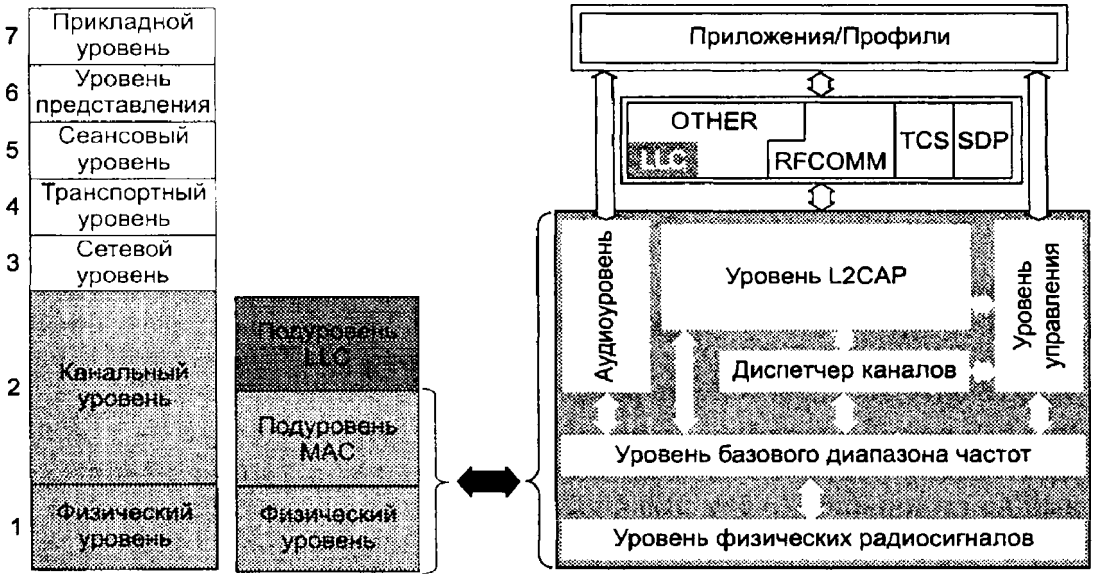
Основное различие двух технологий состоит в том, что фиксированный WiMAX позволяет обслуживать только «статичных» абонентов, а мобильный ориентирован на работу с пользователями, передвигающимися со скоростью до 120 км/ч. Мобильность означает наличие функций роуминга и «бесшовного» переключения между базовыми станциями при передвижении абонента (как происходит в сетях сотовой связи). В частном случае мобильный WiMAX может применяться и для обслуживания фиксированных пользователей.

Bluetooth - IEEE 802.15.1

1. Назначение: персональные (микро или пико) сети
2. Характеристика стека протоколов: собственный стек протоколов.
3. Дальность / скорость: ~10(100)м / 0,7-2,1Мбит/с
4. Диапазон частот: 2,4 ГГц. (79 каналов по 1МГц)
5. Топология: ячеистая (mesh) - 8 (255) устройств
6. Доступ к среде / мультиплексирование:
7. Безопасность:
8. Перспективы: Bluetooth 3 и 4



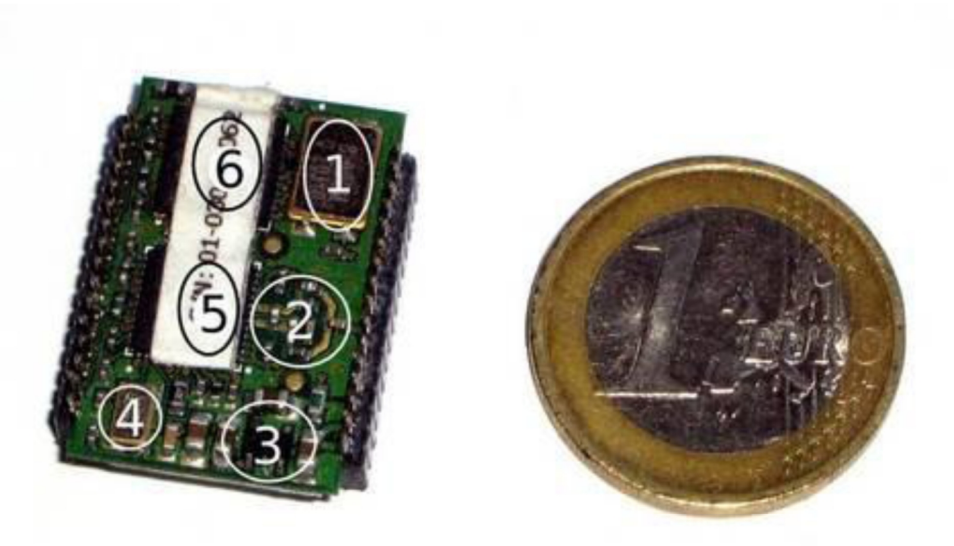
Структуры сети BlueTooth



Соответствие стеков протоколов OSI, LAN, BlueTooth

ZigBee - IEEE 802.15.4

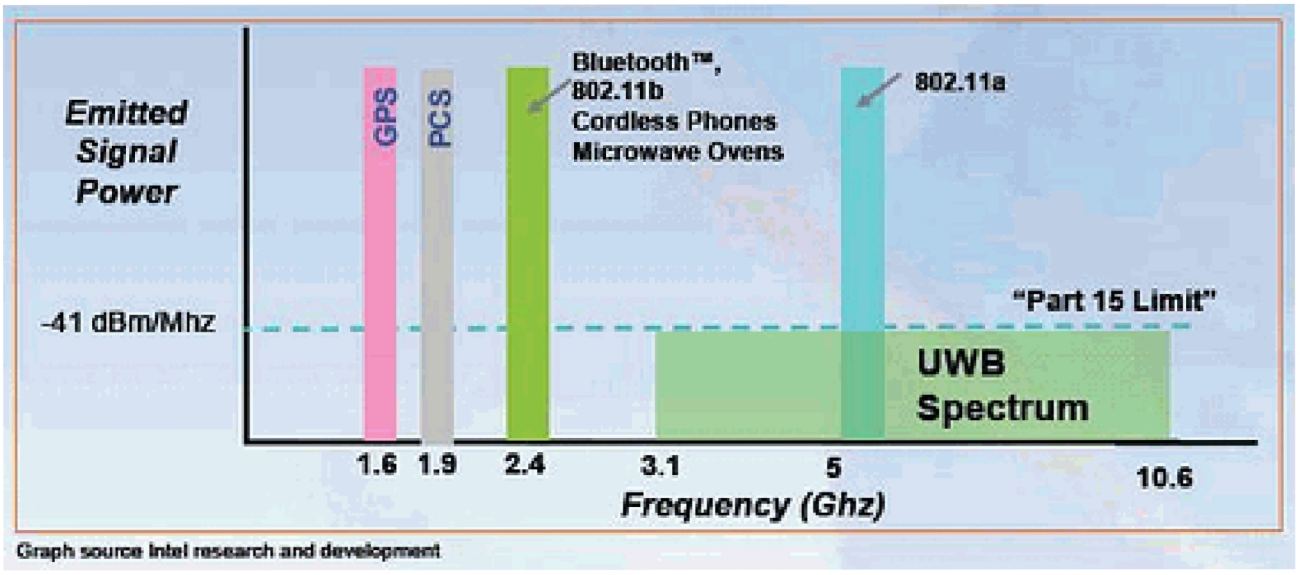
1. Назначение: персональные распределенные (микро или пико) сети
2. Характеристика стека протоколов: собственный стек протоколов.
3. Дальность / скорость: ~75(1500)м / 0,25Мбит/с
4. Диапазон частот: 2,4 ГГц (16 каналов) / 868МГц (1 канал) – Европа/ 915МГц (10 каналов– США).
5. Топология: ячеистая (mesh)
6. Доступ к среде / мультиплексирование:
7. Безопасность:
8. Другие особенности:



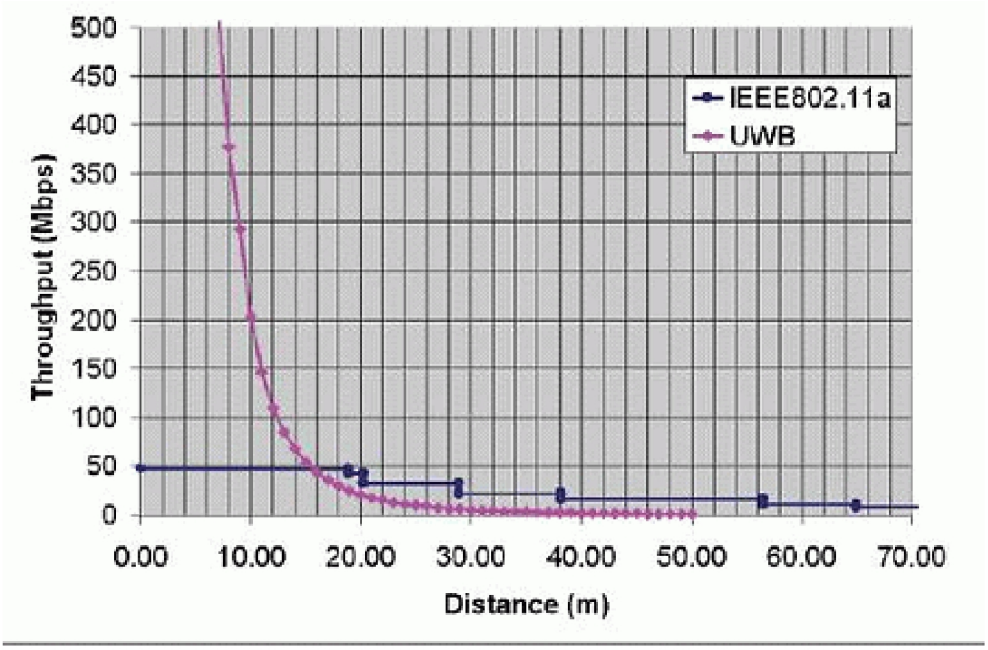
Размер оборудования ZigBee - IEEE 802.15.4

UWB - IEEE 802.15.3a

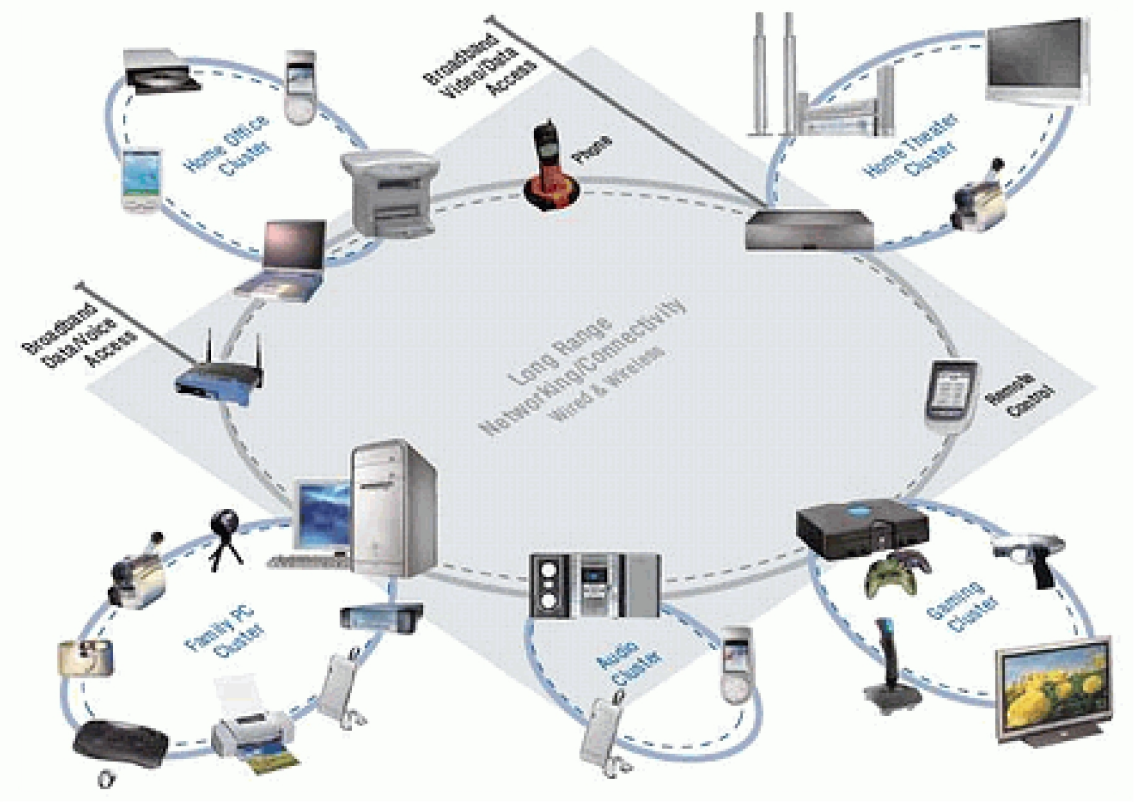
1. **Назначение:** персональные распределенные (микро или пико) сети
2. **Характеристика стека протоколов**: собственный стек протоколов.
3. **Дальность / скорость:**
4. **Диапазон частот:**
5. **Топология:**
6. **Доступ к среде / мультиплексирование:**
7. **Безопасность:**
8. **Другие особенности**:



Принцип работы UWB - IEEE 802.15.3a



Убывание пропускной способности с увеличением расстояния



Экосистема IEEE 802.15.3a

Сравнение беспроводных сетей

Wireless USB vs. 802.11a/b/g & Bluetooth[4]				
Specification	**Wireless USB Specification Rev. 1.0**	**Bluetooth 4.0 (proposed)**	**Wi-Fi (IEEE 802.11n)**	**Bluetooth 2.1 + EDR**
Frequency band	3.1 GHz–10.6 GHz	UWB (not decided)	2.4 GHz/5 GHz	2.4 GHz
Bandwidth	53 - 480 Mbit/s	53 - 480 Mbit/s	Max. 600 Mbit/s	Max. 3 Mbit/s
Distance	3 - 10 m	unknown distance	100 m	1 – 100 m, depending on output
Modulation	MB-OFDM	MB-OFDM	DSSS, DBPSK, DQPSK, CCK, OFDM	GFSK
Standardization	May 2005	pre-standard	September 2009	July 2007

Сравнение беспроводных сетей

Технология	**Стандарт**	**Использование**	**Пропускная способность,** Мбит/с	**Радиус действия**	**Частоты,** ГГц
Wi-Fi	802.11a	WLAN	до 54	до 100 метров	5,0
Wi-Fi	802.11b	WLAN	до 11	до 100 метров	2,4
Wi-Fi	802.11g	WLAN	до 54	до 100 метров	2,4
Wi-Fi	802.11n	WLAN	до 600	до 100 метров	2,4 / 5,0
WiMax	802.16d	WMAN	до 75	6-10 км	1,5-11
WiMax	802.16e	Mobile WMAN	до 40	1-5 км	2.3-3.6
WiMax	802.16m	WMAN, Mobile WMAN	до 1000, до 100	в разработке	в разработке
Bluetooth v. 1.1.	802.15.1	WPAN	до 1	до 10 метров	2,4
Bluetooth v. 1.3.	802.15.3	WPAN	11 - 55	до 100 метров	2,4
UWB	802.15.3a	WPAN	110-480	до 10 метров	2,5-10
ZigBee	802.15.4	WPAN	до 0,250	1-100 м	2,4/ 0,915/ 0,868

1. АДРЕСАЦИЯ В IP-СЕТЯХ

1.1. Три уровня адресации

Каждый **компьютер (узел, хост),** а точнее, его сетевой интерфейс в сети TCP/IP может иметь адреса трех уровней:

1. Локальный (аппаратный) адрес узла (MAC-адрес).
2. Сетевой или IP-адрес, состоящий из 4 байт (протокол IPv4).
3. Символьное (доменное) имя.

Локальный (аппаратный) адрес узла - в большинстве технологий локальных сетей - это MAC-адрес сетевого адаптера или порта маршрутизатора (используется на втором, канальном уровне модели OSI), например:

11-A0-17-3D-BC-01.

Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизовано. MAC-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем.

Сетевой или IP-адрес состоит из 4 байт (протокол IPv4), например:

109.26.17.100.

Этот адрес используется на сетевом уровне (третьем уровне модели OSI). Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов.

IP-адрес состоит из двух частей: **номера сети** и **номера узла**.

Номер сети может быть выбран администратором, либо назначен по рекомендации Network Information Center (NIC), если сеть должна работать как составная часть Internet. Обычно провайдеры услуг Internet получают диапазоны адресов у подразделений NIC, а затем распределяют их между своими абонентами. **Номер узла** в протоколе IPv4 назначается независимо от локального адреса узла.

Деление IP-адреса на поле номера сети и номера узла - гибкое, и граница между этими полями может изменяться в широких пределах (например, с помощью **масок**).

Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей (портов). Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение (интерфейс, порт).

Символьное (доменное) имя (идентификатор), например:

SERV1.IBM.COM

Адрес этого типа назначается администратором (не все узлы или порты имеют такой адрес) и состоит из нескольких частей, например, имени компьютера, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне модели OSI, например, в протоколах FTP или telnet.

Задачи адресации в стеке TCP/IP:

- Согласованное использование адресов различного типа (например, преобразование IP-адреса в MAC-адрес, доменного имени в IP-адрес, см. рис. 1).
- Обеспечение уникальности адресов (в пределах компьютера, подсети, корпоративной сети или Интернета).
- Конфигурирование сетевых интерфейсов и сетевых приложений.
- Обеспечение масштабируемости.

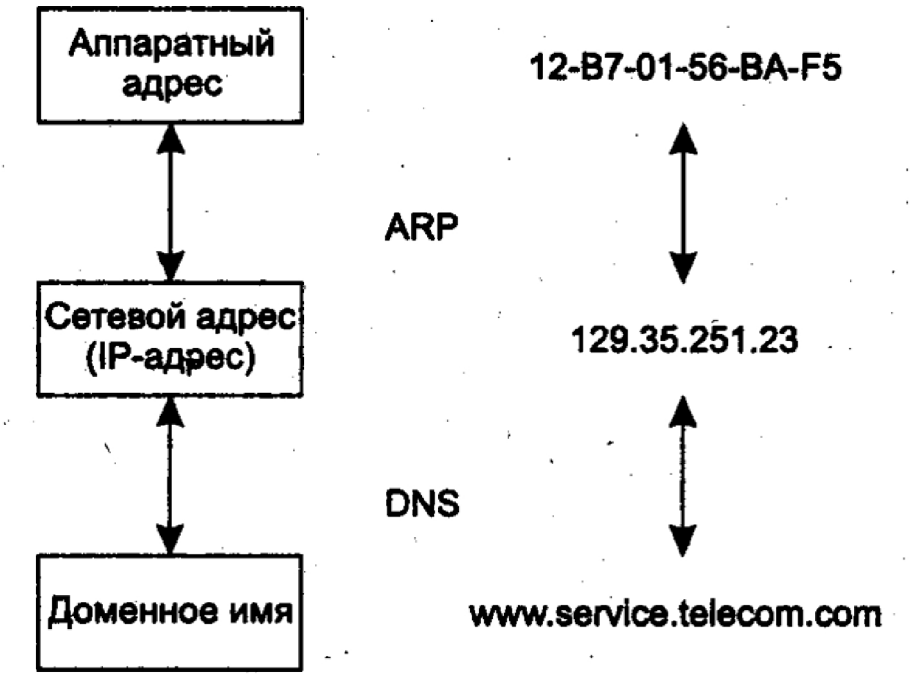


Рис. 1. Преобразование (отображение) адресов различных типов

1.2. Формат IP-адреса (нотация)

Наиболее распространенной формой представления IP-адреса является запись в виде четырех чисел, представляющих значения каждого байта в десятичной форме и разделенных точками:

128.10.2.30

Этот же адрес может быть представлен в двоичном формате, также разделенных точками (такая нотация используется редко, главным образом для пояснения способов преобразования адресов, выделения номеров сети и хоста с помощью масок):

1000 0000. 0000 1010. 0000 0010. 0001 1110

и в шестнадцатеричном формате (с точками):

80.0A.02.1D

В таблице 1 приведен более полный перечень различных форм представления IP-адреса и их связь между собой.

Таблица 1

Формы представления IP-адреса

Нотация	Вид адреса	Преобразование из точечно-десятичной нотации
Точечно-десятичная Dot-decimal notation (DDN)	192.0.2.235	не требуется
Точечно-шестнадцатеричная Dotted Hexadecimal	0xC0.0x00.0x02.0xEB	Каждый октет независимо преобразуется шестнадцатеричную форму
Точечно-восьмеричная Dotted Octal	0300.0000.0002.0353	Каждый октет независимо преобразуется восьмеричную форму
Шестнадцатеричная Hexadecimal	0xC00002EB	Из точечно-шестнадцатеричной удаляют точки
Десятичная Decimal	3221226219	32-битное двоичное число выражают в десятичной форме
Восьмеричная Octal	030000001353	32-битное двоичное число выражают в восьмеричной форме

Структура адресного пространства протокола IPv4 с разбивкой на классы, признаки принадлежности адреса к соответствующему классу, соответствующие каждому классу диапазоны номеров сетей, диапазоны адресов, максимально возможное число узлов в сети, стандартная маска и диапазоны частных адресов даны в таблице 2.

1.3. Особые (зарезервированные) IP-адреса

В системе адресации протокола IPv4 существует множество соглашений об особой интерпретации IP-адресов (см. таблицу 3). Опишем подробнее наиболее важные из них:

1. Сеть с номером **0**, и, следовательно, адреса вида **0.x.x.x** – явно не используются, кроме следующего случая. Если IP-адрес состоит только из двоичных нулей (**неопределенный** адрес): **0.0.0.0,** или в двоичной форме: **0000 0000. 0000 0000. 0000 0000. 0000 0000** ,то он обозначает адрес того узла, который сгенерировал этот пакет (этот режим используется только в некоторых сообщениях ICMP как адрес отправителя).
2. Если в поле **номера сети** стоят **нули**, например: **0.0.0.51** (для класса C), или в двоичной форме: **0000 0000. 0000 0000. 0000 0000. 0101 0001,** то по умолчанию считается, что этот узел принадлежит той же самой сети, что и узел, который отправил пакет (только как адрес отправителя).

Таблица 2

Структура адресного пространства IPv4

Класс	Первые биты слева	Наименьший / наибольший **номер сети**	Диапазон адресов	Максимальное число **узлов** в сети	Стандартная маска	Диапазон частных (**private**) адресов
A	**0**	**1.0.0.0 / 126.0.0.0**	**1.0.0.1 - 126.255.255.254**	**2^{24}-2**, поле 3 байта	**255.0.0.0**	**1** сеть: **10.0.0.0/8**: 10.0.0.0: **10.0.0.1 - 10.255.255.254**
B	**10**	**128.0.0.0 / 191.255.0.0**	**128.0.0.1 - 191.255.255.254**	**2^{16}-2**, поле 2 байта	**255.255.0.0**	**16** сетей: **172.16.0.0/12** 172.16.0.0 – 172.31.0.0: **172.16.0.1 - 172.31.255.254**
C	**110**	**192.0.0.0 / 223.255.255.0**	**192.0.0.1 - 223.255.255.254**	**2^{8}-2**, поле 1 байт	**255.255.255.0**	**256** сетей: **192.168.0.0/16**: 192.168.0.0 - 192.168.255.0: **192.168.0.1 - 192.168.255.254**
D	**1110**	**224.0.0.0 / 239.255.255.255**		Групповые адреса **multicast**	-	
E	**11110**	**240.0.0.0 / 247.255.255.255**		Зарезервировано	-	

3. Если все двоичные разряды IP-адреса равны 1, то есть: **255.255.255.255,** или в двоичной форме: **1111 1111. 1111 1111. 1111 1111. 1111 1111**, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется **ограниченным широковещательным** сообщением (**limited broadcast**).

4. Если в поле адреса узла назначения (host) стоят сплошные 1, например: **72.255.255.255,** (для класса А),
или в двоичной форме: **0111 0010. 1111 1111. 1111 1111. 1111 1111,**
то пакет, имеющий такой адрес, рассылается всем узлам сети с заданным номером. Такая рассылка называется **широковещательным** сообщением (**broadcast**).

5. Адреса вида **127.x.x.x** зарезервированы для организации обратной связи при тестировании работы программного обеспечения узла без реальной отправки пакета по сети (например, когда клиентская и серверная часть программы работают на одном компьютере). Этот адрес имеет название **loopback**.

6. Групповой IP-адрес - **multicast** - обеспечивает доставку пакета сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Один и тот же узел может входить в несколько групп. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом. Обеспечивает распространение информации по схеме «один – ко многим» (например

Интернет теле- и радиовещание). В противоположность multicast (класс D) адреса классов A,B,C называют иногда **unicast** адресами.

Таблица 3

Зарезервированные IP-адреса

Блок адресов CIDR	Описание	Документ
0.0.0.0/8	Эта же сеть (только как адрес отправителя)	RFC 1700
10.0.0.0/8	Частная сеть (в диапазоне класса A)	RFC 1918
14.0.0.0/8	Общедоступные сети данных (доступны до 2008-02-10)	RFC 1700
127.0.0.0/8	Сеть адресов для Loopback - тестирования	RFC 3330
128.0.0.0/16	Зарезервировано (IANA)	RFC 3330
169.254.0.0/16	Link-Local (автоконфигурирование хостов в сетях Microsoft)	RFC 3927
172.16.0.0/12	Частные сети (в диапазоне класса B)	RFC 1918
191.255.0.0/16	Зарезервировано (IANA)	RFC 3330
192.0.0.0/24	Зарезервировано (IANA)	RFC 3330
192.0.2.0/24	Документирование и примеры кода	RFC 3330
192.88.99.0/24	Передача (преобразование) данных из IPv6 в IPv4	RFC 3068
192.168.0.0/16	Частные сети (в диапазоне класса B)	RFC 1918
198.18.0.0/15	Сеть тестирования (benchmark tests)	RFC 2544
223.255.255.0/24	Зарезервировано (IANA)	RFC 3330
224.0.0.0/4	Multicasts (бывший класс D)	RFC 3171
240.0.0.0/4	Зарезервировано (бывший класс E)	RFC 1700
255.255.255.255	Широковещательная рассылка (Broadcast)	

1.4. Формат IP-адреса и маска подсети

IP-адрес состоит из двух частей: номера сети и номера узла (без специального разграничителя).

Для выделения адреса сети (подсети) из IP-адреса могут использоваться (например, в маршрутизаторах):

- **Классы адресов** (RFC 791) с фиксированными границами номера сети и номера узла (информацию о маске не нужно передавать). Со второй половины 90-х годов XX века классовая адресация повсеместно вытеснена бесклассовой адресацией, при которой количество адресов в сети определяется только и исключительно маской подсети.
- **Маски подсети** (**subnet mask, subnetting**) (RFC 950, RFC 1518); обеспечивают гибкое изменение границы между номером сети и номером узла (**VLSM** – variable length subnet mask – **маски подсети переменной длины**). Это позволяет экономно использовать адресное пространство для создания множества сетей разного размера (однако, информацию о маске **нужно** передавать). Сведения о масках подсети переменной длины показаны в таблице 4.

Таблица 4

Маски подсети переменной длины

Краткая запись (блок адресов CIDR)	**Точечно-десятичная запись**	**Точечно-двоичная запись**
IP-адрес/30	255.255.255.252	11111111.11111111.11111111.11111100
IP-адрес/29	255.255.255.248	11111111.11111111.11111111.11111000
IP-адрес/28	255.255.255.240	11111111.11111111.11111111.11110000
IP-адрес/27	255.255.255.224	11111111.11111111.11111111.11100000
IP-адрес/26	255.255.255.192	11111111.11111111.11111111.11000000
IP-адрес/25	255.255.255.128	11111111.11111111.11111111.10000000
IP-адрес/24	255.255.255.0	11111111.11111111.11111111.00000000
IP-адрес/23	255.255.254.0	11111111.11111111.11111110.00000000
IP-адрес/22	255.255.252.0	11111111.11111111.11111100.00000000
IP-адрес/21	255.255.248.0	11111111.11111111.11111000.00000000
IP-адрес/20	255.255.240.0	11111111.11111111.11110000.00000000
IP-адрес/19	255.255.224.0	11111111.11111111.11100000.00000000
IP-адрес/18	255.255.192.0	11111111.11111111.11000000.00000000
IP-адрес/17	255.255.128.0	11111111.11111111.10000000.00000000
IP-адрес/16	255.255.0.0	11111111.11111111.00000000.00000000
IP-адрес/15	255.254.0.0	11111111.11111110.00000000.00000000
IP-адрес/14	255.252.0.0	11111111.11111100.00000000.00000000
IP-адрес/13	255.248.0.0	11111111.11111000.00000000.00000000
IP-адрес/12	255.240.0.0	11111111.11110000.00000000.00000000
IP-адрес/11	255.224.0.0	11111111.11100000.00000000.00000000
IP-адрес/10	255.192.0.0	11111111.11000000.00000000.00000000
IP-адрес/9	255.128.0.0	11111111.10000000.00000000.00000000
IP-адрес/8	255.0.0.0	11111111.00000000.00000000.00000000

Маска подсети (subnet mask) – это число, применяемое в паре с IP-адресом, причем двоичная запись маски содержит слева непрерывную последовательность единиц в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети (подсети). Граница между последовательностями единиц и нулей в маске соответствует границе между номером сети (подсети) и номером узла в IP-адресе. Например:

255.255.128.0 ⇔ 1111 1111. 1111 1111. 1000 0000. 0000 0000

Иногда для пояснения использования масок переменной длины (VLSM, subnetting) в IP-адресе выделяют не 2, а 3 области:

- Биты адреса сети.
- Биты подсети, заимствованные (borrowed) у адреса узла (host), которые в сумме с предыдущим определяют реальный адрес сети (подсети).
- Биты адрес узла.

Например: **1000 0001. 0100 0000. 1000 0110. 0000 0101.**

Этот пример поясняет деление (условное) на сети и подсети: в «большой» сети класса B, образуется $32=2^5$ подсетей.

1.5. Пример использования маски при IP-адресации

Пусть заданы IP-адрес: **129.64.134.5** и маска подсети: **255.255.128.0**

Наложение маски для выделения адреса подсети реализуется логической операцией **«и» (AND)** над их двоичным представлением:

1000 0001. 0100 0000. 1000 0110. 0000 0101 ⇔ 129.64.134.5

AND

1111 1111. 1111 1111. 1000 0000. 0000 0000 ⇔ 255.255.128.0

=

(subnet; адрес подсети содержит нули в области номера узла**)**

1000 0001. 0100 0000. 1000 0000. 0000 0000 ⇔ 129.64.128.0

(host)

0000 0000. 0000 0000. 0000 0110. 0000 0101 ⇔ 0.0.6.5

В десятичной форме адреса подсети и узла, дополненные нулями до 32 бит, выглядят соответственно, как **129.64.128.0** и **0.0.6.5.**

Часто используют краткую запись: вместо IP-адрес: 129.64.134.5 и маска подсети: 255.255.128.0 записывают: **129.64.134.5/17,** то есть маска содержит 17 единиц, а значит в указанном IP-адресе: 129.64.134.5 под номер сети отведено 17 двоичных разрядов. В этом примере в «большой» сети класса B с помощью маски, содержащей 17 единиц, образуется $2=2^1$ подсети.

2. РАСПРЕДЕЛЕНИЕ IP-АДРЕСОВ В СЕТИ ИНТЕРНЕТ

В Интернете уникальность адресов гарантируется централизованной, иерархически организованной системой их распределения. Номер сети может быть назначен только по рекомендации специального подразделения Интернета.

Главным органом регистрации глобальных адресов в Интернете с 1998 г. является неправительственная некоммерческая организация ICANN (*Internet Corporation for Assigned Names and Numbers*), действующая через свой комитет IANA (*Internet Assigned Numbers Authority).* ICANN (IANA) координирует работу региональных отделов: ARIN - Америка; RIPE - Европа; APNIC - Азия и Тихоокеанский регион и т.п.

Региональные отделы выделяют блоки адресов крупным поставщикам услуг, а те, в свою очередь, распределяют их между своими клиентами, среди которых могут быть и более мелкие поставщики.

2.1. Дефицит IP адресов и его преодоление

Не смотря на усилия по увеличению размеров адресного пространства (см. рисунок 2), дефицит (public) адресов в Интернете ощущается (и нарастает) с начала 90-х годов 20-го века.

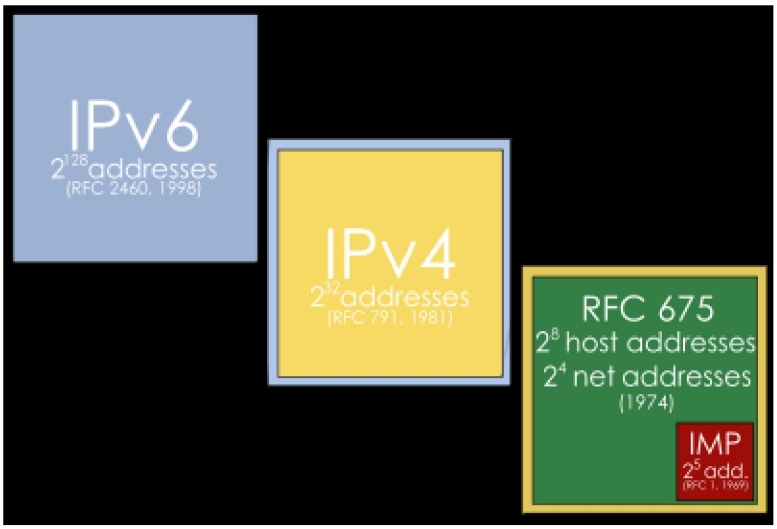


Рис. 2. Рост адресного пространства



Рис. 3. Пример нерационального использования пространства IP-адресов

Основные подходы для снятия проблемы дефицита адресов:

- Использование масок переменной длины VLSM.
- Использование протоколов CIDR (в определенном смысле, современный, усовершенствованный вариант VLSM) и NAT.
- Переход на IPv6 (наилучший вариант, но внедряется медленно).

2.2. Адресация и технология CIDR

Технология бесклассовой междоменной маршрутизации (**Classless Inter-Domain Routing, CIDR**), которая описана в документах RFC 1517 - 1520 (объявлена в 1993г.), позволяет центрам выдачи адресов гибко «нарезать» адреса на основе VLSM (главное условие - наличие непрерывных диапазонов адресов, такие адреса имеют одинаковый ***префикс***, то есть одинаковые цифры в старших разрядах).

Внедрение технологии CIDR обеспечивает:

- Более экономное расходование адресного пространства.
- Уменьшение числа записей в таблицах маршрутизации за счет объединения маршрутов (**supernetting**) – одна запись в таблице маршрутизации может представлять большое количество сетей.

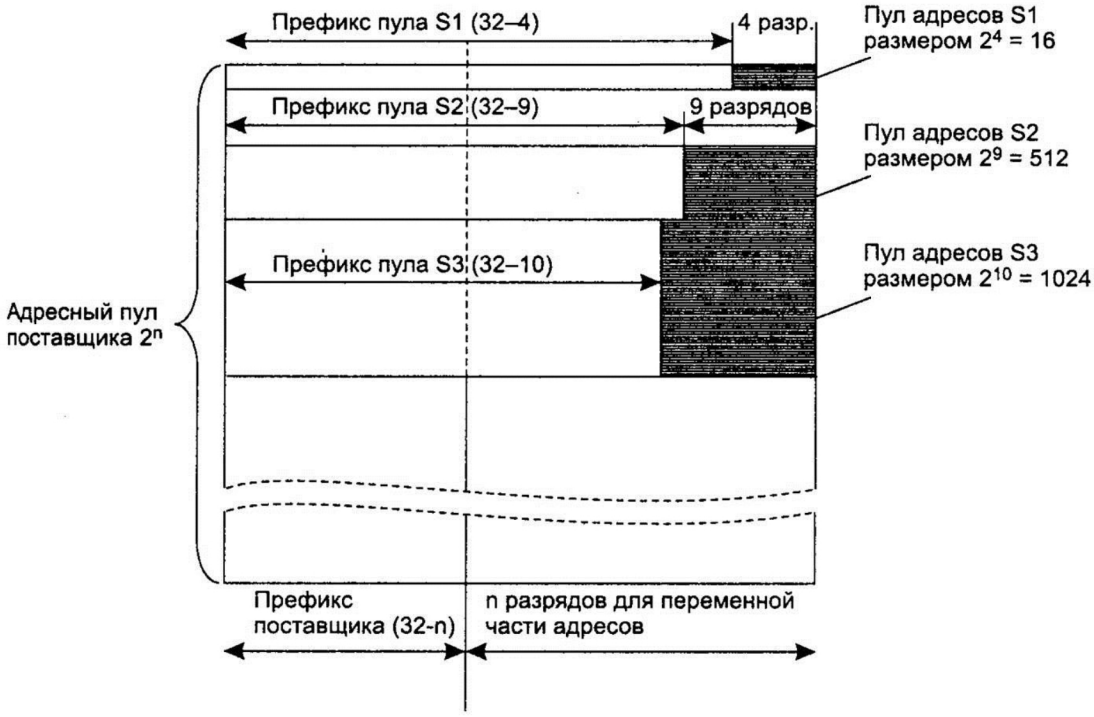


Рис. 4. Распределение IP-адресов в технологии CIDR

3. ОТОБРАЖЕНИЕ IP-АДРЕСОВ НА ЛОКАЛЬНЫЕ АДРЕСА

3.1. Протокол ARP

Одна из задач протокола IP – обеспечение совместной согласованной работы в сети, состоящей из подсетей, в общем случае использующих **разные** сетевые технологии.

Чтобы локальная технология сети смогла доставить пакет на следующий маршрутизатор, необходимо:

- Упаковать пакет в кадр нужного формата (например, Ethernet).
- Снабдить этот кадр локальным (MAC) адресом следующего маршрутизатора.

Для определения локального адреса по IP-адресу используется протокол разрешения адреса ***Address Resolution Protocol, ARP***. Протокол ARP работает 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 локальной сети, или же протокол глобальной сети (X.25, frame relay), как правило, не поддерживающий такой широковещательный доступ (ARP-сервер).

Протокол ReversARP. Существует также протокол, решающий обратную задачу - нахождение IP-адреса по известному локальному адресу. Он называется реверсивный ARP - ***RARP (Reverse Address Resolution Protocol)*** и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера.

3.2. Алгоритм работы протокола ARP

Если возникла необходимость послать пакет по адресу, например 194.85.135.65, то протокол IP сначала проверяет не содержится ли уже строка с таким IP-адресом в его ARP-таблице.

Если нет, то узел, которому нужно выполнить отображение IP-адреса на локальный адрес, формирует ARP-запрос (формат пакета протокола ARP показан на рисунке 5), вкладывает (инкапсулирует) его в кадр протокола канального уровня, указывая в нем известный IP-адрес, и рассылает запрос широковещательно.

В локальных сетях протокол ARP использует широковещательные (broadcast) кадры протокола канального уровня для поиска в сети узла с заданным IP-адресом.

<table>
<tr><th>0</th><th>8</th><th>16</th><th>31</th></tr>
<tr><td colspan="2">Тип сети
(для Ethernet – 1 (0x1))</td><td colspan="2">Тип протокола
(для IP – 2048 (0x800))</td></tr>
<tr><td>Длина локального адреса (для Ethernet – 6 (0x6))</td><td>Длина сетевого адреса
(для IP – 4 (0x4))</td><td colspan="2">Операция
(для ARP - 1 (0x1))
(для RARP - 2 (0x2))</td></tr>
<tr><td colspan="4">Локальный адрес отправителя (source) (байты 0 - 3) 00 80 48 EB</td></tr>
<tr><td colspan="2">Локальный адрес отправителя (source) (байты 4 - 5) 7E 60</td><td colspan="2">IP-адрес отправителя (source) (байты 0-1) 194.85.</td></tr>
<tr><td colspan="2">IP-адрес отправителя (source) (байты 2-3) 135.75</td><td colspan="2">Искомый локальный адрес (получателя, target) (байты 0 - 1) 00 00</td></tr>
<tr><td colspan="4">Искомый локальный адрес (получателя, target) (байты 2-5) 00 00 00 00</td></tr>
<tr><td colspan="4">Искомый IP-адрес (получателя, target) (байты 0 - 3) 194.85.135.65</td></tr>
</table>

Рис. 5. Формат пакета протокола ARP

Все узлы локальной сети получают ARP-запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес и отправляет его уже направленно, так как в ARP запросе отправитель указывает свой локальный адрес.

ARP-запросы и ответы используют один и тот же формат пакета. Так как локальные адреса могут в различных типах сетей иметь различную длину, то формат пакета протокола ARP зависит от типа сети.

Узел, отправляющий ARP-запрос, заполняет в пакете все поля, кроме поля искомого локального адреса, а в нем ставит нули (для RARP-запроса не указывается искомый IP-адрес). Значение этого поля заполняется узлом, опознавшим свой IP-адрес.

Для уменьшения числа ARP-запросов найденное соответствие между IP-адресом и MAC-адресом сохраняется автоматически в ARP-таблице соответствующего интерфейса в виде записи: 194.85.135.65 – 00 E0 F7 7F 19 20

Теперь, если вновь возникает необходимость послать пакет по адресу 194.85.135.65, то протокол IP прежде чем посылать широковещательный запрос сначала проверяет, не содержится ли уже строка с таким IP-адресом в его ARP-таблице.

Если да, то адрес 00 E0 F7 7F 19 20 сразу ставится в поле MAC-адреса назначения посылаемого IP пакета. А широковещательный ARP-запрос не рассылается.

ARP-таблица пополняется не только за счет поступающих на данный интерфейс ARP-ответов, но и путем анализа и извлечения полезной адресной информации из широковещательных ARP-запросов.

В ARP-таблицах существует два типа записей: динамические и статические. Статические (они существуют пока устройство включено) создают вручную с помощь утилиты (команды) ***arp.*** Если динамическая запись не обновлялась в течение определенного времени, то она исключается из таблицы. Поэтому ARP-таблицу называют иногда ***ARP-кэшем***.

3.3. Отображение IP-адресов на локальные адреса за пределами локальной сети

Протокол **ProxyARP** – разновидность протокола ARP, устанавливаемая на маршрутизаторах, позволяющая отображать IP-адреса на локальные в сетях с широковещанием даже в тех случаях, когда искомый узел находится за пределами данного широковещательного домена (домена коллизий).

3.4. Отображение IP-адресов на локальные адреса в глобальных сетях

В глобальных сетях, не поддерживающих широковещательную рассылку, администратору сети чаще всего приходится вручную формировать ARP-таблицы. В последнее время 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 (ARP-сервер), который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети.

4. ОТОБРАЖЕНИЕ СИМВОЛЬНЫХ АДРЕСОВ НА IP-АДРЕСА: СЛУЖБА DNS

DNS (Domain Name System) - это распределенная база данных, поддерживающая иерархическую систему имен для идентификации узлов в сети Internet. Служба DNS предназначена для автоматического поиска IP-адреса по известному символьному имени узла. Спецификация DNS определяется стандартами RFC 1034 и 1035. DNS требует статической конфигурации своих таблиц, отображающих имена компьютеров в IP-адрес.

Протокол DNS является служебным протоколом прикладного уровня. Этот протокол несимметричен - в нем определены DNS-серверы и DNS-клиенты. DNS-серверы хранят часть распределенной базы данных о соответствии символьных имен и IP-адресов. Эта база данных распределена по административным доменам сети Internet. Клиенты сервера DNS знают IP-адрес сервера DNS своего административного домена и по протоколу IP передают запрос, в котором сообщают известное символьное имя и просят вернуть соответствующий ему IP-адрес.

Если данные о запрошенном соответствии хранятся в базе данного DNS-сервера, то он сразу посылает ответ клиенту, если же нет - то он посылает запрос DNS-серверу другого домена, который может сам обработать запрос, либо передать его другому DNS-серверу.

Все DNS-серверы соединены иерархически, в соответствии с иерархией доменов сети Internet. Клиент опрашивает эти серверы имен, пока не найдет нужные отображения. Этот процесс ускоряется из-за того, что серверы имен постоянно кэшируют информацию, предоставляемую по запросам.

Клиентские компьютеры могут использовать в своей работе IP-адреса нескольких DNS-серверов, для повышения надежности своей работы.

База данных DNS имеет структуру дерева, называемого доменным пространством имен, в котором каждый домен (узел дерева) имеет имя и может содержать поддомены. Имя домена идентифицирует его положение в этой базе данных по отношению к родительскому домену, причем точки в имени отделяют части, соответствующие узлам домена.

Корень базы данных DNS управляется IANA. Домены верхнего уровня назначаются для каждой страны, а также для различных организаций. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, а для различных типов организаций используются следующие аббревиатуры:

- **com** - коммерческие организации (например, microsoft.com);
- **edu** - образовательные (например, mit.edu);
- **gov** - правительственные организации (например, nsf.gov);
- **org** - некоммерческие организации (например, fidonet.org);
- **net** - организации, поддерживающие сети (например, nsf.net).

Каждый домен DNS администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Каждый домен имеет уникальное имя, а каждый из поддоменов имеет уникальное имя внутри своего домена. Имя домена может содержать до 63 символов. Каждый хост в сети Internet однозначно определяется своим *полным доменным именем (fully qualified domain name, FQDN)*, которое включает имена всех доменов по направлению от хоста к корню. Пример полного DNS-имени: **citint.dol.ru**.

5. АВТОМАТИЗАЦИЯ ПРОЦЕССА НАЗНАЧЕНИЯ IP-АДРЕСОВ УЗЛАМ СЕТИ: ПРОТОКОЛ DHCP

При конфигурировании компьютеров и маршрутизаторов администратор должен задать: IP-адрес, маску, IP-адреса шлюза и серверов DNS, доменное имя и т.п.

Протокол динамического конфигурирования хостов (***Dynamic Host Configuration Protocol*** (***DHCP***)) автоматизирует процесс конфигурирования сетевых интерфейсов, гарантируя от дублирования адресов, за счет централизованного управления их распределением.

DHCP является расширением протокола BOOTP, использовавшегося ранее для обеспечения бездисковых рабочих станций IP-адресами при их загрузке. DHCP сохраняет обратную совместимость с BOOTP.

Стандарт протокола DHCP был принят в октябре 1993 года. Последняя версия протокола (март 1997 года) описана в RFC 2131 и 2132.. Новая версия DHCP, предназначенная для использования в среде IPv6, носит название DHCPv6 и определена в RFC 3315 (июль 2003 года).

Протокол DHCP реализует модель ***клиент-сервер***. Передача данных производится при помощи протокола UDP, при этом сервер принимает сообщения от клиентов на порт 67 и отправляет сообщения клиентам на порт 68. Сервер DHCP может работать в следующих режимах:

- ручное назначение статических адресов;
- автоматическое назначение статических адресов;
- автоматическое распределение динамических адресов.

5.1. Алгоритм работы протокола DHCP

Во время старта системы компьютер-клиент DHCP, находящийся в состоянии "инициализация", посылает сообщение **discover** (исследовать), которое широковещательно распространяется по локальной сети и передается всем DHCP-серверам частной интерсети. Каждый DHCP-сервер, получивший это сообщение, отвечает на него сообщением **offer** (предложение), которое содержит IP-адрес и конфигурационную информацию.

Компьютер-клиент DHCP переходит в состояние "выбор" и собирает конфигурационные предложения от DHCP-серверов. Затем он выбирает одно из этих предложений, переходит в состояние "запрос" и отправляет сообщение **request** (запрос) тому DHCP-серверу, чье предложение было выбрано.

Выбранный DHCP-сервер посылает сообщение **DHCP-acknowledgment** (подтверждение), содержащее тот же IP-адрес, который уже был послан ранее на стадии исследования, а также **срок аренды** для этого адреса. Кроме того, DHCP-сервер посылает параметры сетевой конфигурации.

Помимо IP-адреса, DHCP также может сообщать клиенту дополнительные параметры, необходимые для нормальной работы в сети. Эти параметры называются опциями DHCP. Список стандартных опций можно найти в RFC 2132.

Некоторыми из наиболее часто используемых опций являются:

- IP-адрес маршрутизатора по умолчанию;
- маска подсети;
- адреса серверов DNS;
- имя домена DNS.

Некоторые поставщики программного обеспечения могут определять собственные, дополнительные опции DHCP.

Некоторые реализации службы DHCP способны автоматически обновлять записи DNS, соответствующие клиентским компьютерам, при выделении им новых адресов. Это производится при помощи протокола обновления DNS, описанного в RFC 2136.

После получения подтверждения, клиент переходит в состояние "**связь**", находясь в котором он может принимать участие в работе сети TCP/IP. Компьютеры-клиенты, которые имеют локальные диски, сохраняют полученный адрес для использования при последующих стартах системы. При приближении момента истечения срока аренды адреса компьютер пытается обновить параметры аренды у DHCP-сервера, а если этот IP-адрес не может быть выделен снова, то ему возвращается другой IP-адрес.

Реализация **протокола DHCP.** Компания Microsoft впервые включила сервер DHCP в поставку серверной версии Windows NT 3.5, выпущенной в 1994 году. Начиная с Windows 2000 Server реализация DHCP-сервера от Microsoft позволяет динамически обновлять записи DNS, что используется в Active Directory.

Internet Systems Consortium выпустил первую версию ISC DHCP Server (для Unix-подобных систем) 6 декабря 1997 года. 22 июня 1999 года вышла версия 2.0, более точно соответствующая стандарту.

Компания Cisco включила сервер DHCP в Cisco IOS 12.0 в феврале 1999 года. Sun добавила DHCP-сервер в Solaris 8 в июле 2001 года.

ПРИЛОЖЕНИЕ

Таблица преобразования между двоичной/шестнадцатеричной и десятичной системами счисления для целых чисел от 0 до 255
(серым цветом выделены стандартные маски)

		Младшие (правые) разряды															
	Bin	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Старшие (левые) разряды	0000 0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	0001 1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0010 2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
	0011 3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
	0100 4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
	0101 5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
	0110 6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
	0111 7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
	1000 8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
	1001 9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
	1010 A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
	1011 B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
	1100 C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
	1101 D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
	1110 E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
	1111 F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Тема лекции:

Протокол IP (межсетевого взаимодействия). Маршрутизация

План лекции:

1. Функции протокола IP.
2. Формат IP-пакета (IPv4).
3. Фрагментация IP-пакетов.
4. Таблица маршрутизации.
5. Одношаговая маршрутизация и маршрутизация от источника.
6. Расстояние до сети назначения. Метрика.
7. Маршрут по умолчанию. Другие «особые» маршруты
8. Источники и типы записей в таблице маршрутизации.
9. Протокол IPv6.

Протокол межсетевого взаимодействия - IP

Основу транспортных средств стека протоколов TCP/IP составляет протокол межсетевого взаимодействия – **Internet Protocol (IP).**

Его характеризуют как:

- **дейтаграммный** (**best effort** – без гарантий доставки, ненадежный);
- **без установления соединения** (**connectionless**).

К основным функциям протокола IP относятся:

- перенос между сетями различных типов адресной информации в унифицированной форме;
- сборка и разборка пакетов при передаче их между сетями с различным максимальным значением длины пакета.

Формат IP-пакета (дейтаграммы)





Пакет IP состоит из заголовка и поля данных.

Формат заголовка IP-пакета



Поля заголовка IP-пакета

Заголовок пакета имеет ***следующие поля***:

- Поле **Номер версии** (**VERS**) (**4 бита**) - версия протокола IP. Сейчас повсеместно используется версия 4 и готовится переход на версию 6.
- Поле **Длина заголовка** (**HLEN**) пакета IP (**4 бита**) указывает длину заголовка 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 байт в поле **Резерв** (**IP OPTIONS**).

- Поле **Тип сервиса** (**SERVICE TYPE**) (1 байт) задает приоритетность пакета и вид критерия выбора маршрута. Первые три бита этого поля образуют подполе приоритета пакета (**PRECEDENCE**). Приоритет может иметь значения от 0 (нормальный пакет) до 7 (пакет управляющей информации). Маршрутизаторы и компьютеры могут принимать во внимание приоритет пакета и обрабатывать более важные пакеты в первую очередь. Следующие три бита определяют критерий оптимизации маршрута: бит **D=1** (delay) - минимизация задержки доставки данного пакета, бит **T=1** (traffic) - максимизация пропускной способности, а бит **R=1** (reliability) - максимизация надежности доставки.
- Поле **Общая длина** (**TOTAL LENGTH**) (**2** байта) указывает общую длину пакета (заголовка и поля данных).
- Поле **Идентификатор пакета** (**IDENTIFICATION**) (**2** байта) используется для распознавания пакетов, образовавшихся путем фрагментации исходного пакета. Все фрагменты должны иметь одинаковое значение этого поля.
- Поле **Флаги** (**FLAGS**) (**3** бита) указывает на возможность фрагментации пакета (бит **Do not Fragment – DF=1** - запрещает маршрутизатору фрагментировать данный пакет), а также на то, является ли данный пакет промежуточным или последним фрагментом исходного пакета (бит **More Fragments – MF=1** - говорит о том пакет переносит промежуточный фрагмент).
- Поле **Смещение фрагмента** (**FRAGMENT OFFSET**) (**13** бит) используется для указания в байтах смещения 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 максимальной длины пакета.
- Поле **Время жизни** (**TIME TO LIVE**) (**1** байт) указывает предельный срок, в течение которого пакет может перемещаться по сети. Время жизни данного пакета измеряется в секундах и задается источником передачи средствами протокола IP. На шлюзах и в других узлах сети по истечении каждой секунды из текущего времени жизни вычитается единица; единица вычитается также при каждой транзитной передаче (даже если не прошла секунда). При истечении времени жизни пакет аннулируется.
- Идентификатор **Протокола верхнего уровня** (**PROTOCOL**) (**1** байт) указывает, какому протоколу верхнего уровня принадлежит пакет (например, это могут быть протоколы TCP, UDP или RIP).
- **Контрольная сумма** (**HEADER CHECKSUM**) (2 байта) рассчитывается по всему заголовку.
- Поля **Адрес источника** (**SOURCE IP ADDRESS**) и **Адрес назначения** (**DESTINATION IP ADDRESS**) имеют одинаковую длину - 32 бита, и одинаковую структуру.

- Поле **Резерв** (**IP OPTIONS**) является необязательным и используется обычно только при отладке сети. Это поле состоит из нескольких подполей, каждое из которых может быть одного из восьми предопределенных типов. В этих подполях 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 Резерв должно быть добавлено несколько байт для выравнивания заголовка пакета по 32-битной границе.

Размер IP-пакета

Максимальная длина поля данных пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65535 байтов, однако 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 байтов, умещающиеся в поле данных кадра Ethernet.

Фрагментация IP-пакетов

Подчеркнем разницу между фрагментацией сообщений в узле-отправителе и динамической фрагментацией сообщений в транзитных узлах сети - маршрутизаторах. Во всех стеках протоколов есть протоколы, реализующие деление (фрагментацию) сообщений прикладного уровня на фрагменты, укладывающиеся в кадры канального уровня. При этом используется значение **MTU – (Maximum Transmission Unit – максимальная единица передачи),** характерное для данной технологии нижнего уровня.

Важное свойство протокола IP – реализация ***динамической фрагментации*** пакетов при передаче между сетями с различными максимально допустимыми значениями длины поля данных кадров.

В стеке TCP/IP протокол TCP разбивает поток байтов с прикладного уровня на сегменты (например по 1460 байт в случае использования на нижних уровнях протоколов семейства Ethernet). Поэтому протокол IP в узле-отправителе обычно не использует свои возможности по фрагментации. А вот на маршрутизаторе при передаче из сети с большим в сеть с меньшим значением **MTU** – используется как раз умение протокола IP выполнять дефрагментацию.

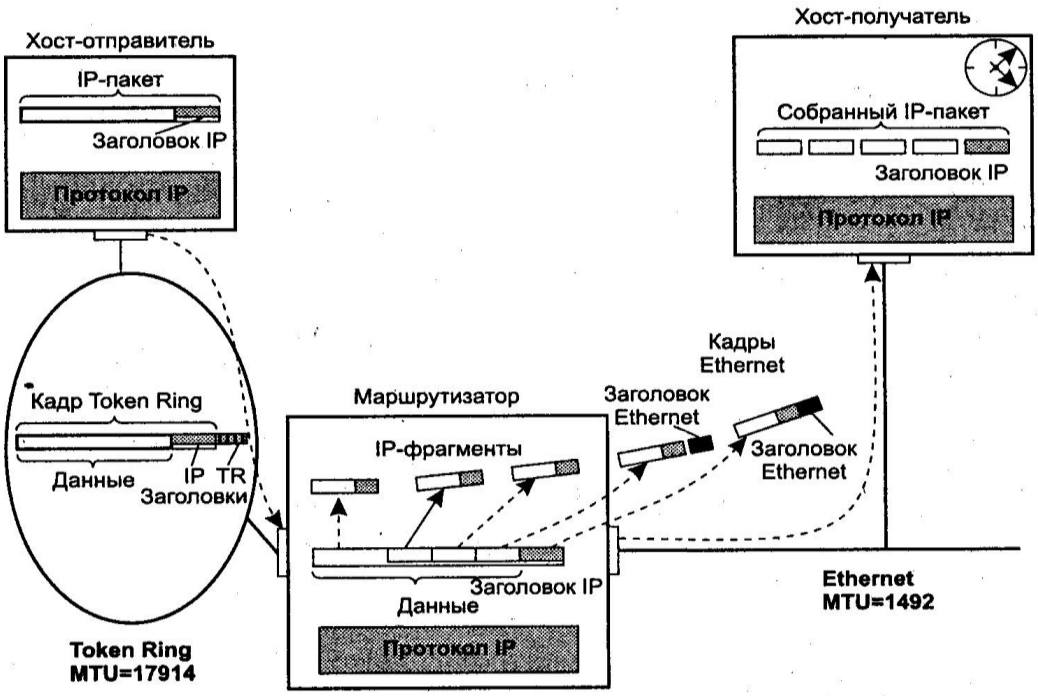
Значения MTU различных технологий

Технология	***MTU,*** *байт*
Ethernet DIX	1500
Ethernet 802.3	1492
Token Ring (IBM, 16 Мбит/с)	17914
Token Ring (802.5, 4 Мбит/с)	4464
FDDI	4352
X.25	576

Параметры фрагментации

Фрагмент при передаче по сети может неоднократно подвергаться фрагментации и каждый из фрагментов должен быть снабжен полноценным заголовком IP, содержащим ***Идентификатор, TTL***, флаги **DF** и **MF, смещение**.

Идентификатор *(уникальный) позволяет опознать все фрагменты одного и того же пакета.* ***TTL*** - задает время существования пакета в сети (ожидания при сборке). **Смещение фрагмента (Fragment Offset)** задает положение фрагмента в исходном пакете. Флаг **MF=1 - More Fragments** - фрагмент – не последний (для не фрагментированного пакета **MF=0).** Флаг **DF=1 - Do not Fragment** – не подлежит фрагментации (при невозможности доставить без фрагментации уничтожается и отправляется диагностическое ICMP-сообщение).



Пример фрагментации IP-пакетов

Сборка фрагментов IP-пакетов

IP-маршрутизаторы не собирают фрагменты пакетов в более крупные пакеты, даже если на пути встречается сеть, допускающая такое укрупнение. Это связано с тем, что отдельные фрагменты сообщения могут перемещаться по интерсети по различным маршрутам.

При приходе первого фрагмента пакета узел назначения запускает таймер, который определяет максимально допустимое время ожидания прихода остальных фрагментов этого пакета. Если таймер истекает раньше прибытия последнего фрагмента, то все полученные к этому моменту фрагменты пакета отбрасываются, а в узел, пославший исходный пакет, направляется сообщение об ошибке с помощью протокола ICMP.

Маршрутизация в IP сетях. Основные понятия

Таблица маршрутизации *(routing table)*

В стеке TCP/IP маршрутизаторы и конечные узлы принимают решения о том, кому передавать пакет для его успешной доставки узлу назначения, на основании так называемой ***таблицы маршрутизации (routing table).***

В этой таблице в столбце *"Адрес сети назначения"* указываются адреса всех сетей, которым данный маршрутизатор может передавать пакеты. В стеке TCP/IP принят так называемый ***одношаговый*** *подход* к оптимизации маршрута продвижения пакета (**next-hop routing**) - каждый маршрутизатор и конечный узел принимает участие в выборе только одного шага передачи пакета.

Одношаговая маршрутизация (next-hop routing)

Поэтому в каждой строке таблицы маршрутизации указывается не весь маршрут в виде последовательности IP-адресов маршрутизаторов, через которые должен пройти пакет, а только один IP-адрес - **адрес следующего маршрутизатора**, которому нужно передать пакет. Вместе с пакетом следующему маршрутизатору передается ответственность за выбор следующего шага маршрутизации. Одношаговый подход к маршрутизации означает распределенное решение задачи выбора маршрута. Это снимает ограничение на максимальное количество транзитных маршрутизаторов на пути пакета.

Пример таблицы маршрутизации

Адрес сети назначения	Адрес следующего маршрутизатора	Номер выходного порта	Расстояние до сети назначения
56.0.0.0	198.21.17.7	1	20
56.0.0.0	213.34.12.4	2	130
116.0.0.0	213.34.12.4	2	1450
129.13.0.0	198.21.17.6	1	50
198.21.17.0	-	2	0
213. 34.12.0	-	1	0
default	198.21.17.7	1	-

Маршрутизация от источника (source routing)

Альтернативой одношаговому подходу является указание в пакете всей последовательности маршрутизаторов, которые пакет должен пройти на своем пути. Такой подход называется **маршрутизацией от источника - Source Routing**. В этом случае выбор маршрута производится конечным узлом или первым маршрутизатором на пути пакета, а все остальные маршрутизаторы только отрабатывают выбранный маршрут, осуществляя коммутацию пакетов, то есть передачу их с одного порта на другой. Алгоритм Source Routing применяется в

сетях IP только для отладки, когда маршрут задается в поле Резерв (IP OPTIONS) пакета.

Расстояние до сети назначения

В случае, если в таблице маршрутов имеется более одной строки, соответствующей одному и тому же адресу сети назначения, то при принятии решения о передаче пакета используется та строка, в которой указано наименьшее значение в поле **"Расстояние до сети назначения"**.

При этом под расстоянием понимается любая **метрика**, используемая в соответствии с заданным в сетевом пакете классом сервиса. Это может быть количество транзитных маршрутизаторов в данном маршруте (количество **хопов** от **hop** - прыжок), время прохождения пакета по линиям связи, надежность линий связи, или другая величина, отражающая качество данного маршрута.

Маршрут по умолчанию и другие «особые» маршруты

Одношаговая маршрутизация позволяет сократить объем таблиц маршрутизации маршрутизаторов за счет использования в качестве номера сети назначения так называемого ***маршрута по умолчанию - default***.

Все пакеты с номерами сетей, которые отсутствуют в таблице маршрутизации, передаются маршрутизатору, указанному в строке ***default***. Поэтому маршрутизаторы хранят в своих таблицах ограниченную информацию о сетях интерсети, пересылая пакеты для остальных сетей в порт и маршрутизатор, используемые по умолчанию. Подразумевается, что маршрутизатор, используемый по умолчанию, передаст пакет на магистральную сеть, а маршрутизаторы, подключенные к магистрали, имеют полную информацию о составе интерсети.

Особенно часто приемом маршрутизации по умолчанию пользуются конечные узлы. При наличии одного маршрутизатора в локальной сети этот вариант - единственно возможный для всех конечных узлов.

Кроме маршрута **default**, в таблице маршрутизации могут встретиться два типа специальных записей - запись о **специфичном** для узла маршруте и запись об адресах сетей, **непосредственно подключенных** к портам маршрутизатора.

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 N и ее отдельного узла, имеющего адрес N,D, при поступлении пакета, адресованного узлу N,D, маршрутизатор отдаст предпочтение записи для N,D.

Непосредственно подключенные сети

Записи в таблице маршрутизации, относящиеся к сетям, непосредственно подключенным к маршрутизатору, в поле "Расстояние до сети назначения"

содержат нули.

Примеры таблиц маршрутизации

Таблица 18.5. Таблица программного маршрутизатора ОС Windows 2000

Сетевой адрес	Маска	Адрес шлюза	Интерфейс	Метрика
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	255.255.255.255	127.0.0.1	127.0.0.1	1
198.21.17.255	255.255.255.255	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	255.255.255.255	127.0.0.1	127.0.0.1	1
213.34.12.255	255.255.255.255	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	255.255.255.255	198.21.17.6	198.21.17.6	1

Таблица 18.6. Таблица маршрутизации аппаратного маршрутизатора

Адрес назначения	Маска	Шлюз	Метрика	Статус	TTL	Источник
198.21.17.0	255.255.255.0	198.21.17.5	0	Up	–	Подключена
213.34.12.0	255.255.255.0	213.34.12.3	0	Up	–	Подключена
56.0.0.0	255.0.0.0	213.34.12.4	14	Up	–	Статическая
116.0.0.0	255.0.0.0	213.34.12.4	12	Up	–	Статическая
129.13.0.0	255.255.0.0	198.21.17.6	1	Up	160	RIP

Таблица 18.7. Таблица маршрутизации Unix-маршрутизатора

Адрес назначения	Шлюз	Флаги	Число ссылок	Загрузка	Интерфейс
127.0.0.0	127.0.0.1	UH	1	154	lo0
Маршрут по умолчанию	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	U	44	132435	le1
129.13.0.0	198.21.1.7.6	UG	6	16450	le0
56.0.0.0	213.34.12.4	UG	12	5764	le1
116.0.0.0	213.34.12.4	UG	21	23544	le1

Источники и типы записей в таблице маршрутизации

Для маршрутизаторов существуют ***три*** основных источника появления записей в таблице:

1. **ПО стека TCP/IP** (при инициализации маршрутизатора создает ***минимальную таблицу маршрутизации:*** записи о непосредственно подключенных сетях, маршрутизаторах по умолчанию (информация о которых появляется в стеке при ручном конфигурировании интерфейсов компьютера или маршрутизатора), адресах особого назначения.
2. **Администратор**, формирующий записи с помощью системной утилиты (**route** в ОС семейств WindowsNT и Unix) или команд аппаратного маршрутизатора. Такие записи являются ***статическими*** (т.е. не имеют срока жизни). Могут быть постоянными (сохраняться при перезагрузке устройства) и временными (не сохраняются при перезагрузке).
3. **Протоколы маршрутизации** (RIP, EIGRP, OSPF). Такие записи являются ***динамическими,*** т.е. имеют ограниченный срок жизни.

Протокол IPv6

Протокол IP нового поколения (Next Generation Internet Protocol, IPng), сейчас используют название **IPv6.** Принят IETF в сентябре 1995 (RFC1752 + RFC2460, 2373 август 1998).

Обеспечивает модернизацию IPv4 по следующим **направлениям**:

1. Создание масштабируемой схемы адресации.
2. Сокращение объема работы маршрутизаторов.
3. Предоставление гарантий качества предоставляемых услуг.
4. Обеспечение защиты данных, передаваемых по сети.

Масштабируемая система адресации

IP-адрес IPv6 состоит из 16 байт (128 бит). Вместо прежних 2-х уровней иерархии адреса (номер сети и номер узла) в IPv6 имеется 4 уровня (3 используются для идентификации сетей, а один – для идентификации узлов сети).

Частные адреса в IPv6 имеют две разновидности:

1. Адреса локальных сетей, ***не разделенных*** на подсети (64 бита в поле идентификатора интерфейса).
2. Адреса локальных сетей, **разделенных** на подсети (содержат дополнительно двухбайтовое поле номера подсети).

Шестнадцатеричная форма записи адреса, например:

FEDC:0A98:0:0:0:0:7654:3210

Три основных типа адресов:

1. Индивидуальный адрес (unicast) (Классов – нет). Основной подтип – **глобальный агрегируемый уникальный адрес**.
2. Групповой адрес (multicast).
3. Адрес произвольной рассылки (anycast).

Гибкий формат заголовка IPv6-пакета снижает накладные расходы.

Снижение нагрузки на маршрутизаторы

Меры по освобождению маршрутизаторов от некоторых вспомогательных задач:

1. Перенесение функций фрагментации с маршрутизаторов на конечные узлы.
2. Агрегирование адресов (уменьшает размер адресных таблиц, сокращает служебный трафик).
3. Широкое использование маршрутизации от источника.
4. Отказ от обработки необязательных параметров заголовка.
5. Использование в качестве номера узла его MAC-адреса.

Новая версия протокола IP, являющаяся составной частью проекта IPv6, имеет ***встроенные средства защиты данных на сетевом уровне***.

Тема лекции: **Алгоритмы и протоколы стека TCP/IP**

План лекции:

1. Протоколы TCP и UDP и их общие свойства.
2. Протокол UDP .
3. Протоколы TCP, формат заголовка.
4. Логические соединения, сокеты.
5. Квитирование.
6. Алгоритм скользящего окна.
7. Протокол ICMP. Утилиты ping и traceroute.

Протоколы транспортного уровня TCP и UDP

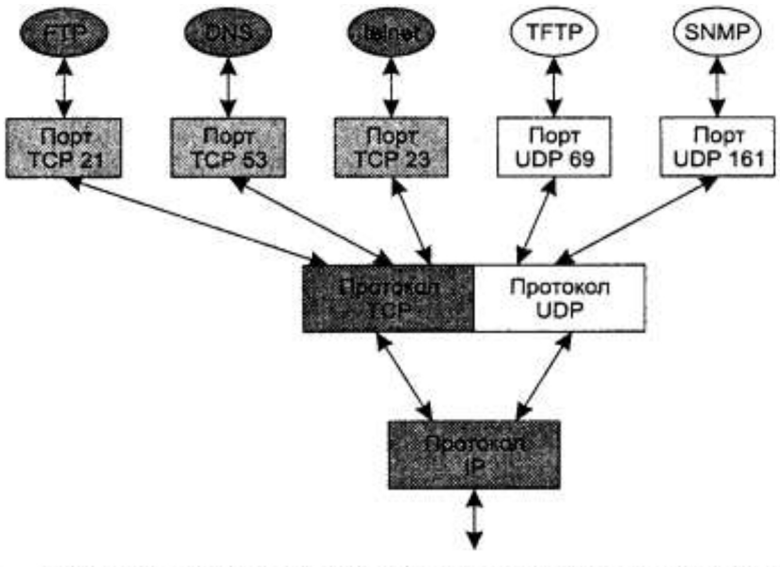
Главную задачу ***транспортного уровня*** – передачу данных между ***прикладными процессами*** решают:

- Протокол управления передачей (***Transmission Control Protocol, TCP***, RFC793).
- Протокол пользовательских дейтаграмм (***User Datagram Protocol, UDP***, RFC768).

Общие свойства: используют «порты» и «сокеты» для адресации процессов, упаковывают свои PDU в IP-пакеты. Протокольные сущности TCP и UDP устанавливаются только на конечных узлах.

Каждый компьютер может выполнять несколько процессов, прикладной процесс может иметь несколько точек входа. С другой стороны, пакеты, которые отправляют в сеть разные приложения, работающие на одном конечном узле, обрабатываются общим для них протоколом IP.

Процедура приема данных протоколами TCP и UDP, поступающих от нескольких различных прикладных служб, называется ***мультиплексированием***. Обратная процедура – распределение протоколами TCP и UDP поступающих от сетевого уровня пакетов между набором высокоуровневых служб – называется ***демультиплексированием***.



Мультиплексирование и демультиплексирование на транспортном уровне

Мультиплексирование и демультиплексирование на транспортном уровне

Порты, их номера и диапазоны портов

Протоколы TCP и UDP ведут для каждого приложения две очереди: очередь пакетов, поступающих к данному приложению из сети, и очередь пакетов, отправляемых данным приложением в сеть. Пакеты, поступающие на транспортный уровень, организуются операционной системой в виде множества очередей к точкам входа различных прикладных процессов. Такие системные очереди называют ***портами***, причем входная и выходная очереди одного приложения рассматриваются как один порт. Для однозначной идентификации портов им присваивают номера. Номера портов используют для адресации приложений.

Популярные общедоступные службы (FTP - 21, telnet - 23, HTTP, DNS) имеют закрепленные стандартные ***назначенные номера*** (***well-known***). Они являются уникальными в пределах Интернета и назначаются централизованно IANA из диапазона от 0 до 1023.

Диапазоны портов (IANA, RFC1700, 3232):

- **0 – 1023** – привилегированные или **зарезервированные** (The Well Known Ports);
- **1024 – 49151** – **зарегистрированные** (The Registered Ports);
- **49152 – 65535** – **динамические порты** (The Dynamic and/or Private Ports).

Протокол UDP

Протокол UDP, являясь дейтаграммным протоколом, реализует сервис по возможности (best effort), то есть не гарантирует доставку своих сообщений.

Единица данных протокола UDP называется UDP-дейтаграммой. Каждая дейтаграмма переносит отдельное пользовательское сообщение. Ее длина не может превышать длины поля данных протокола IP, которое в свою очередь ограничено размером кадра технологии нижнего уровня. Поэтому, если UDP-буфер переполняется, то данные приложения отбрасываются.

UDP-дейтаграмма и ее заголовок



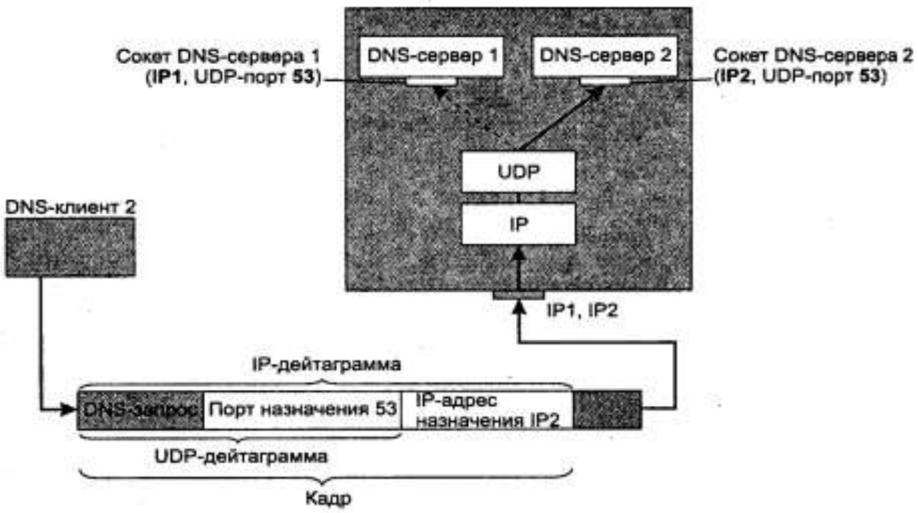
Формирование дейтаграммы протокола UDP

Заголовок UDP, состоящий из 4-х полей по 2 байта, содержит: номера портов отправителя и получателя, длину дейтаграммы и контрольную сумму.

Пример заголовка:

Source Port=0x0035
Destination Port=0x0411
Total length=132 (0x84) bytes
Checksum=0x5333

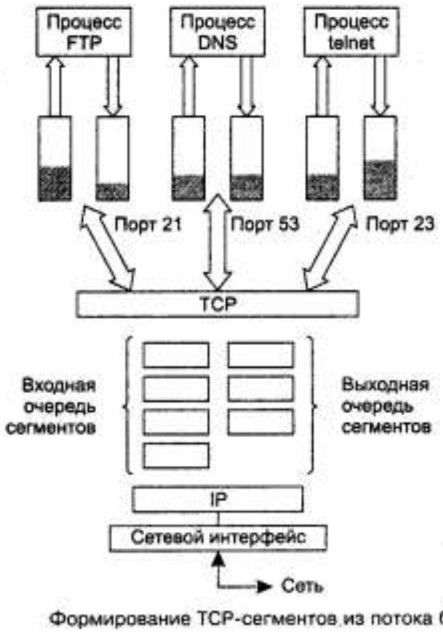
Демультиплексирование UDP, сокеты



Демультиплексирование протокола UDP на основе сокетов

Сокетом называют пару (IP-адрес, номер порта), которая однозначно определяет прикладной процесс в сети.

Схема работы протокола TCP



Формирование TCP-сегментов из потока байтов

Сегмент протокола TCP и его заголовок

Протокол TCP рассматривает информацию, поступающую от протоколов более высокого уровня, как ***неструктурированный поток байтов***. Из которого в буфере «нарезаются» **сегменты** и снабжаются **заголовками**.

Поля заголовка TCP-сегмента:

Порт источника (source port) – (2 байта) идентифицирует процесс отправитель.

Порт приемника (destination port) – (2 байта) идентифицирует процесс получатель.

Последовательный номер (sequence number) -

(4 байта) номер 1-го байта в сегменте, который определяет смещение сегмента относительно потока данных.

Подтвержденный номер (acknowledgement number) - (4 байта) максимальный номер байта в полученном сегменте, увеличенный на единицу (используется в качестве квитанции).

Длина заголовка (hlen) - (4 байта) длина заголовка TCP-сегмента в 32-битных словах.

Резерв (reserved) – 6 бит.

Кодовые биты (**code bits**) – 6 бит (уст. в 1):

URG - срочное сообщение; **ACK** - квитанция на принятый сегмент; **PSH** - запрос на отправку без ожидания заполнения буфера; **RST** – запрос на восстановление соединения; **SYN** – используется для синхронизации счетчиков переданных данных при установлении соединения; **FIN** – признак последнего байта в потоке передаваемых данных.

Окно (window) – (2 байта) – число байтов данных, ожидаемых отправителем.

Контрольная сумма (checksum) – 2 байта.

Указатель срочности (urgent pointer) – (2 байта), указывает на конец данных, которые надо срочно принять.

Параметры (options) – (не более 3 байт) могут отсутствовать, задает, например, макс. длину сегмента.

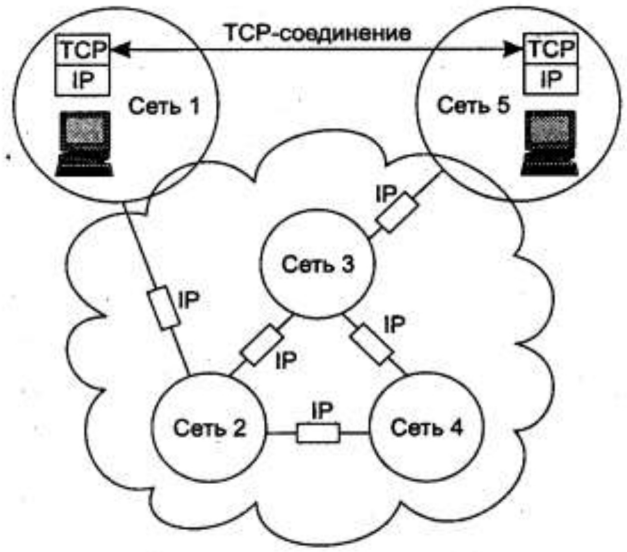
Заполнитель (padding) – фиктивное поле переменной длины делает размер заголовка кратным 32 битам.

Логические соединения

Протокол TCP обеспечивает ***надежную*** доставку сообщений, с помощью ***ненадежного*** протокола IP, устанавливая ***логическое соединение***.

При установлении логического соединения модули TCP договариваются между собой о ***параметрах*** процедуры обмена данными:

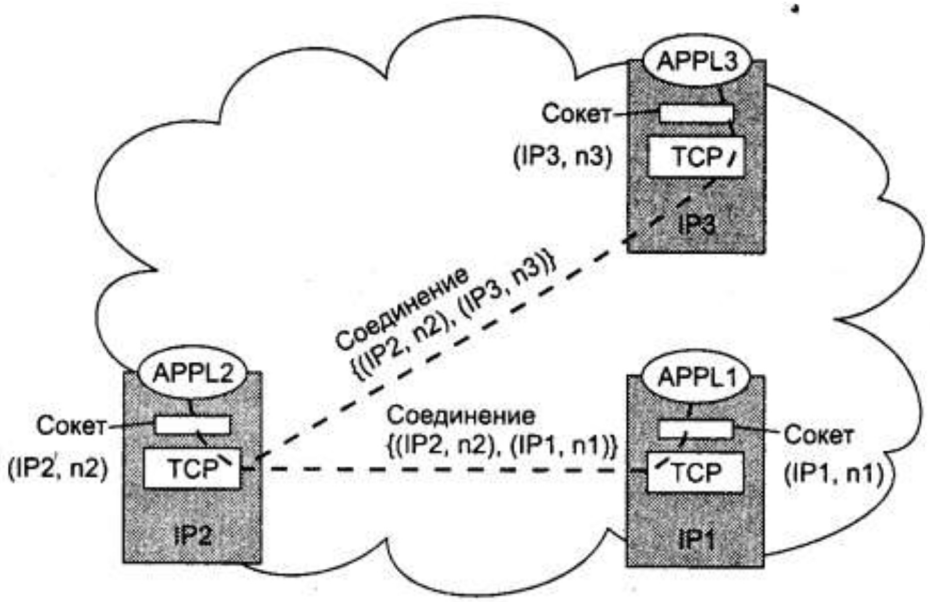
- максимальный размер сегмента, который сторона готова принимать;
- максимальный объем данных, который разрешается передавать в свою сторону (даже если квитанция на предыдущую порцию не получена);
- начальный порядковый номер байта, с которого сторона начинает отсчет потока данных в рамках данного логического соединения.



TCP-соединение создает надежный логический канал между конечными узлами

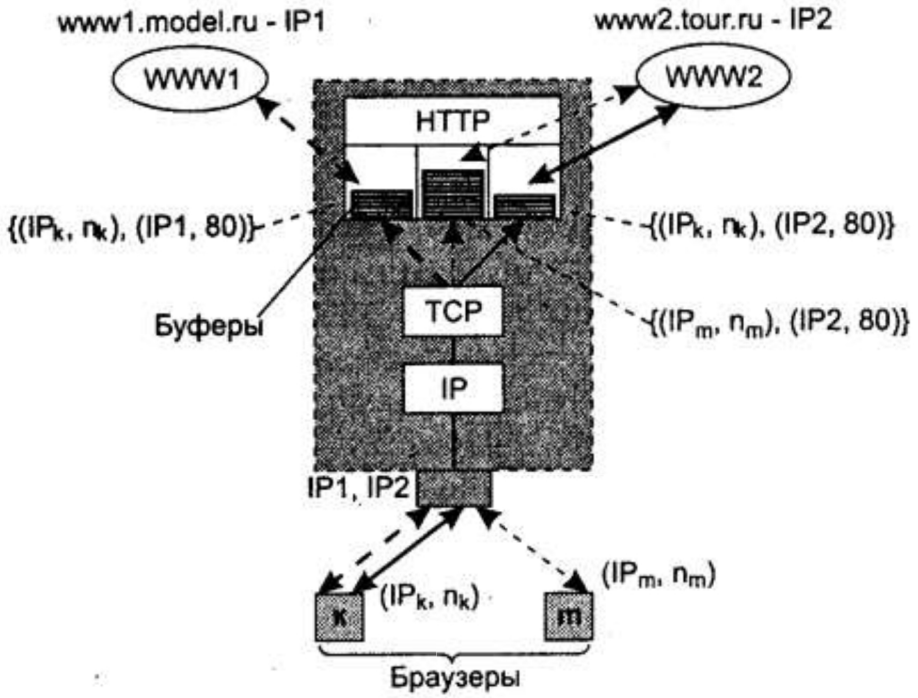
Идентификация логических соединений

Логическое TCP-соединение **однозначно** идентифицируется ***парой сокетов***.



Один сокет может участвовать в нескольких соединениях

Демультиплексирование на основе соединений



Демультиплексирование TCP на основе соединений



Передача без установления соединения (*а*) и с установлением соединения (*б*)

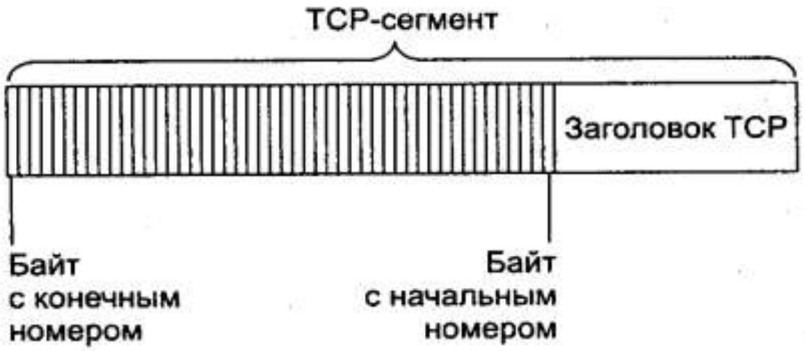
Алгоритм скользящего окна в протоколе TCP

Окно определено на множестве нумерованных ***байтов*** неструктурированного потока данных, буферизируемых протоколом TCP.

В рамках установленного логического соединения правильность передачи каждого сегмента должна подтверждаться квитанцией от получателя. ***Квитирование*** – традиционный способ обеспечения надежности связи, в TCP

реализуется алгоритмом ***скользящего окна.***

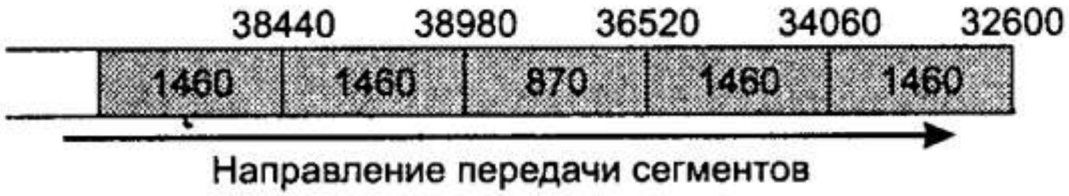
Последовательный и подтвержденный номер



Нумерация байтов в TCP-сегменте

Квитанция посылается только в случае правильного приема данных, отрицательные квитанции не высылаются. Таким образом отсутствие квитанции означает: либо потерю сегмента, либо прием искаженного сегмента, либо потерю квитанции.

В одном сегменте могут быть помещены и данные и квитанция, подтверждающая их получение.



Порядковый номер и номер квитанции

В качестве квитанции получатель сегмента отсылает ответное сообщение (сегмент), в которое помещает число (подтвержденный номер), на единицу превышающее максимальный номер байта в полученном сегменте.

Подтверждающий номер часто интерпретируется как номер следующего ожидаемого байта данных.

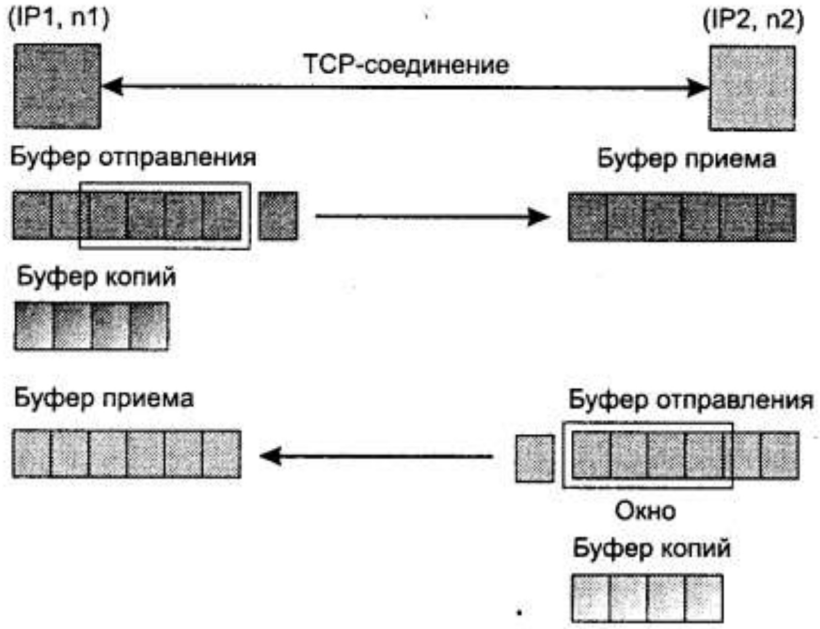
Окно приема

Протокол TCP является дуплексным: каждая сторона выступает и как отправитель, и как получатель. У каждой стороны 3 буфера: 1 - для хранения принятых сегментов, 2 – для хранения тех, которые предстоит отправить, 3 – для хранения копий сегментов, квитанции о получении которых еще не поступили.

Стороны посылают друг другу ***окна приема*** – сколько байтов принимающая сторона разрешает отправить с момента получения последней квитанции. Т.о. стороны пытаются регулировать поток байтов в свою сторону (сколько байтов,

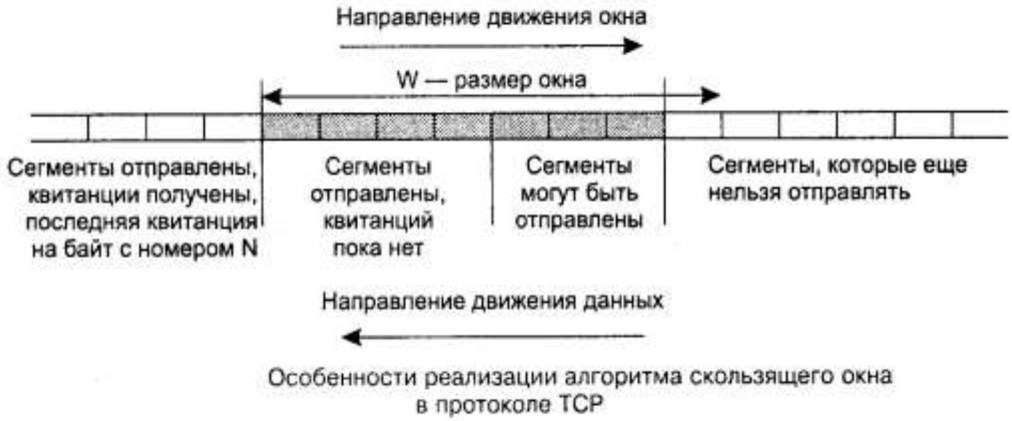
начиная с номера, на который была выслана квитанция, они готовы в настоящий момент принять).

Система буферов TCP-соединения



Система буферов TCP-соединения

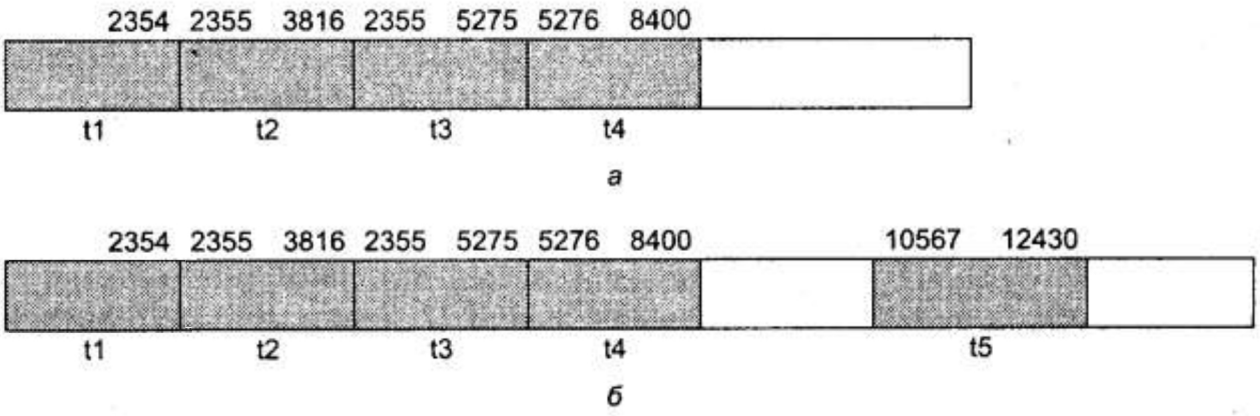
Алгоритм скользящего окна



Особенности реализации алгоритма скользящего окна в протоколе TCP

Накопительный принцип квитирования

Получатель может посылать квитанцию, подтверждающую получение сразу нескольких сегментов, но **только, если** они образуют ***непрерывный поток байтов***. Таким образом протокол TCP подтверждает получение не отдельных блоков данных, а непрерывной последовательности байтов.



Накопительный принцип квитирования: *а* — плотное заполнение буфера, в момент t4 передается квитанция, *б* — неплотное заполнение буфера, в момент t5 снова передается квитанция 8401

Время ожидания квитанции

Когда протокол TCP передает в сеть сегмент, он помещает его копию в очередь повторной передачи и запускает ***таймер***. Когда приходит квитанция на этот сегмент, его копия удаляется из очереди. Если квитанция не приходит до истечения срока, то сегмент посылается повторно.

Тайм-аут не должен быть слишком коротким (избыточные повторные передачи), или слишком длинным (простои).

При выборе величины тайм-аута должны учитываться скорость и надежность линий связи и др. факторы.

Алгоритм определения тайм-аута

При каждой передаче засекается время от момента отправки сегмента до прихода квитанции о его приеме (время оборота, $\mathbf{T_{об}}$). Получаемые значения $T_{об}$ усредняются с весовыми коэффициентами, возрастающими от предыдущего замера к последующему (чтобы усилить влияние последних замеров). В качества тайм-аута выбирается среднее $T_{об.сред.}$ умноженное не некоторый эмпирический коэффициент (> 2). В сетях с большим разбросом $T_{об}$ при выборе тайм-аута учитывается и дисперсия этой величины.

Управление окном приема

Окна приема на разных концах логического соединения могут иметь разный размер и динамически изменяться по инициативе каждой из сторон в зависимости от загруженности сети.

В качестве действующего размера окна выбирается минимальное из двух значений – значения, диктуемого приемной стороной, и значения, определяемого «на месте» отправителем.

При отказе от приема в квитанции указывается окно нулевого размера. Но и после этого приложение может послать сообщение на отказавшийся от приема порт, установив указатель срочности **URG.**

Протокол ICMP

Протокол межсетевых управляющих сообщений (Internet Control Message Protocol, ICMP, RFC792) дополняет протокол IP. Выполняет функции мониторинга и диагностики.

Сообщения об ошибках посылает не всегда.

Все типы ICMP-сообщений могут быть разделены на 2 класса:

1. **Диагностические** сообщения об **ошибках**.
2. **Информационные** сообщения типа **запрос/ответ**.

ICMP-сообщение инкапсулируется в поле данных пакета IP.

Заголовок ICMP состоит из 8 байт:

- ***Тип*** (1 байт) содержит код, определяющий тип сообщения (см. табл.).
- ***Код*** (1 байт) более тонко дифференцирует тип ошибки.
- ***Контрольная сумма*** (2 байта) всего ICMP-сообщения.
- Заголовок также включает поле из 4-х байт, содержимое которого зависит от значений полей типа и кода. В сообщениях типа запрос/ответ это поле содержит 2-х байтовые подполя ***идентификатора*** и ***порядкового номера***.

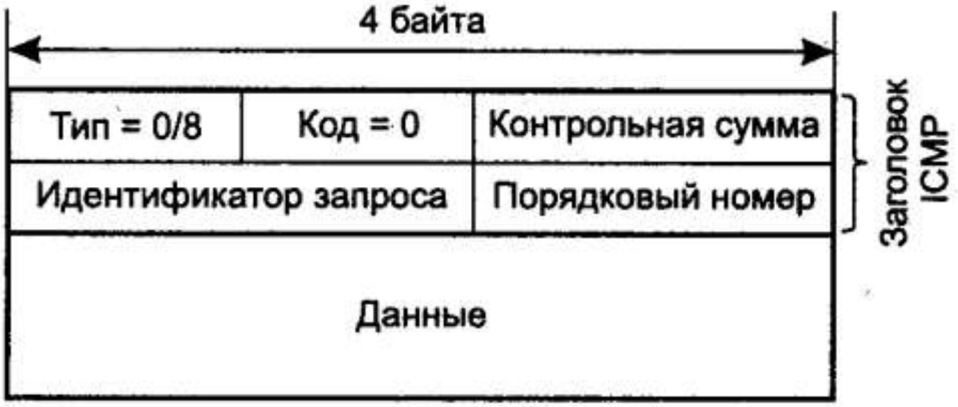
Возможные значения поля типа

Значение	Тип сообщения
0	Эхо-ответ
3	Узел назначения недостижим
4	Подавление источника
5	Перенаправление маршрута
8	Эхо-запрос
11	Истечение времени дейтаграммы
12	Проблема с параметром пакета
13	Запрос отметки времени
14	Ответ отметки времени
17	Запрос маски
18	Ответ маски

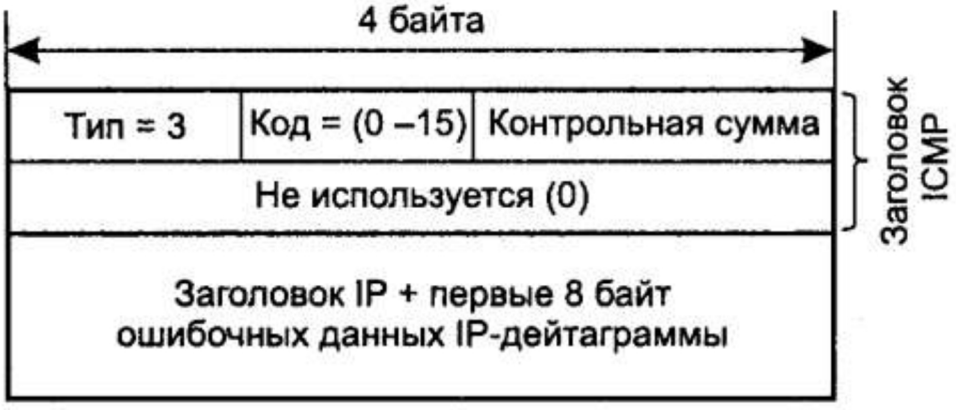
Коды, детализирующие причину ошибки о недостижимости узла назначения

Код	Причина
0	Сеть недостижима
1	Узел недостижим
2	Протокол недостижим
3	Порт недостижим
4	Требуется фрагментация, а бит DF установлен
5	Ошибка в маршруте, заданном источником
6	Сеть назначения неизвестна
7	Узел назначения неизвестен
8	Узел-источник изолирован
9	Взаимодействие с сетью назначения административно запрещено
10	Взаимодействие с узлом назначения административно запрещено
11	Сеть недостижима для заданного класса сервиса
12	Узел недостижим для заданного класса сервиса
13	Взаимодействие административно запрещено путем фильтрации

Формат эхо-запроса/эхо-ответа. Утилита *ping*



Формат сообщения об ошибке. Утилита *traceroute*



Тема лекции: **Алгоритмы и протоколы маршрутизации**

План лекции:

1. Иерархия маршрутизаторов в Интернете.
2. Классификация алгоритмов маршрутизации.
3. Дистанционно-векторные алгоритмы маршрутизации и алгоритмы состояния связей.
4. Протокол RIP.
5. Протокол OSPF.
6. Протокол BGP.

Иерархия маршрутизаторов в Интернете. Автономные системы

Следуя *принципу масштабируемости*, маршрутизация в Интернете функционирует в пределах ***автономных систем*** (Autonomous Systems, AS).

Маршрутизация в Интернете является иерархической. То есть, внутри каждой автономной системы может применяться любой из существующих протоколов маршрутизации (или их комбинация), но между автономными системами всегда применяется один и тот же протокол.

Автономная система (**AS**)– совокупность сетей под единым административным управлением, обеспечивающим общую для всех входящих в автономную систему маршрутизаторов политику маршрутизации. Регистрация автономных систем происходит централизованно.

Номер автономной системы состоит из 16 разрядов и никак не связан с префиксами IP-сетей, входящих в нее.

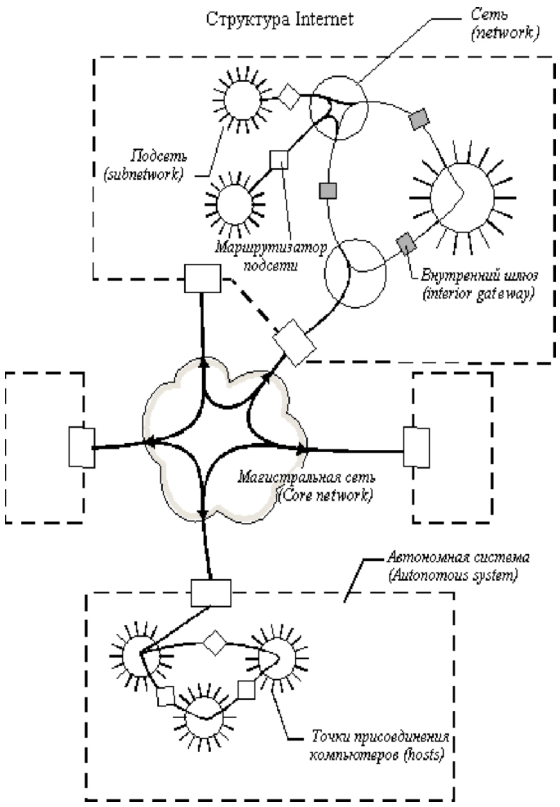
Автономные системы составляют магистраль Интернета. Концепция автономных систем скрывает от администраторов магистрали Интернета проблемы маршрутизации пакетов на уровне IP-сетей.

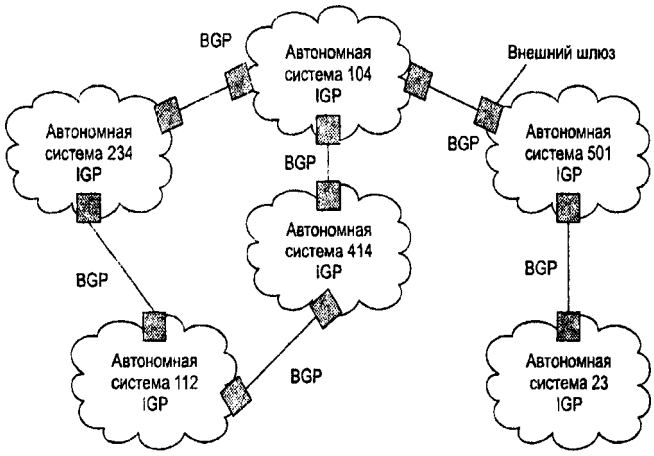
Внутренние и внешние шлюзовые протоколы (протоколы маршрутизации)

В IP-сетях в качестве ***внутренних шлюзовых протоколов (Interior Gateway Protocol, IGP),*** применяемых внутри автономных систем, сегодня наиболее активно используются три: **RIP, OSPF, IS-IS.**

Внешним шлюзовым протоколом (Exterior Gateway Protocol, EGP), то есть протоколом выбора маршрута между автономными системами, сегодня является протокол **BGPv4**.

До введения автономных систем предполагался двух уровневый подход – сначала ***маршрут определялся как последовательность сетей***, а затем вел непосредственно к заданному узлу в конечной сети.





Сейчас появился третий (верхний) уровень маршрутизации – теперь сначала ***маршрут определяется как последовательность автономных систем***, затем - как ***последовательность сетей***, а уж затем ведет к конечному узлу.

Классификация алгоритмов маршрутизации



Алгоритмы построения таблиц для одношаговой маршрутизации можно разделить на три класса:

- алгоритмы **фиксированной** маршрутизации,

- алгоритмы **простой** маршрутизации,
- алгоритмы **адаптивной** маршрутизации.

Результат работы любого из этих алгоритмов имеет единый формат. За счет этого в одной и той же сети различные узлы могут строить таблицы маршрутизации по своим алгоритмам, а затем обмениваться между собой недостающими данными, так как форматы этих таблиц фиксированы.

Поэтому маршрутизатор, работающий по алгоритму адаптивной маршрутизации, может снабдить конечный узел, применяющий алгоритм фиксированной маршрутизации, сведениями о пути к сети, о которой конечный узел ничего не знает.

Алгоритмы фиксированной маршрутизации

Фиксированная (статическая) маршрутизация применяется в сетях с простой топологией связей. Основана на ручном составлении таблицы маршрутизации администратором сети. Алгоритм часто эффективно работает также для магистралей крупных сетей, так как сама магистраль может иметь простую структуру с очевидными наилучшими путями следования пакетов в подсети, присоединенные к магистрали.

Различают одномаршрутные таблицы, в которых для каждого адресата задан один путь, и многомаршрутные таблицы, определяющие несколько альтернативных путей для каждого адресата. При использовании многомаршрутных таблиц должно быть задано правило выбора одного из них. Чаще всего один путь является основным, а остальные - резервными.

Алгоритмы простой маршрутизации

Алгоритмы ***простой маршрутизации*** подразделяются на три подкласса:

Случайная маршрутизация - пакеты передаются в любом, случайном направлении, кроме исходного.

Лавинная маршрутизация - пакеты передаются во всех направлениях, кроме исходного (применяется в мостах для пакетов с неизвестным адресом доставки).

Маршрутизация по предыдущему опыту - таблицы маршрутов составляются на основании данных, содержащихся в проходящих через маршрутизатор пакетах. Именно так работают прозрачные мосты, собирая сведения об адресах узлов, входящих в сегменты сети. Такой способ маршрутизации обладает медленной адаптируемостью к изменениям топологии сети.

Алгоритмы адаптивной (динамической) маршрутизации

Это основной вид алгоритмов маршрутизации в современных сетях со сложной топологией. ***Адаптивная маршрутизация*** основана на том, что маршрутизаторы периодически обмениваются специальной топологической информацией об имеющихся в интерсети сетях, а также о связях между маршрутизаторами.

Обычно учитывается не только топология связей, но и их пропускная способность и состояние.

Адаптивные протоколы позволяют всем маршрутизаторам собирать информацию о топологии связей в сети, оперативно отрабатывая все изменения конфигурации связей.

Эти протоколы имеют распределенный характер, который выражается в том, что в сети отсутствуют какие-либо выделенные маршрутизаторы, которые бы собирали и обобщали топологическую информацию: эта работа распределена между всеми маршрутизаторами.

Конвергенция сети

Протоколы маршрутизации должны создавать согласованные друг с другом таблицы маршрутизации. Протоколу маршрутизации нужно некоторое время, которое называется ***временем конвергенции,*** для того чтобы после нескольких итераций обмена служебной информацией все маршрутизаторы сети внесли изменения в свои таблицы и в результате все таблицы снова стали согласованными.

Различные протоколы маршрутизации обладают ***разным временем конвергенции***.

Протоколы и алгоритмы адаптивной (динамической) маршрутизации

Применяемые сегодня в IP-сетях протоколы маршрутизации относятся к адаптивным распределенным протоколам.

Адаптивные алгоритмы маршрутизации должны: быть простыми; обеспечивать рациональность маршрута; обеспечивать сходимость (конвергенцию) за приемлемое время.

Алгоритмы адаптивных протоколов обмена маршрутной информацией принято делить на две группы:

- ***Дистанционно-векторные алгоритмы (Distance Vector Algorithms, DVA).***
- ***Алгоритмы состояния связей (Link State Algorithms, LSA).***

Дистанционно-векторные алгоритмы

В **дистанционно-векторных алгоритмах (DVA)** каждый маршрутизатор периодически и широковещательно рассылает по сети вектор расстояний от себя до всех известных ему сетей. Под расстоянием обычно понимается число промежуточных маршрутизаторов через которые пакет должен пройти прежде, чем попадет в соответствующую сеть.

Может использоваться и другая метрика, учитывающая не только число перевалочных пунктов, но и время прохождения пакетов по связи между соседними маршрутизаторами или пропускная способность (или их комбинация).

Получив вектор от соседнего маршрутизатора, каждый маршрутизатор 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 рассылает новое значение вектора по сети.

В конце концов, каждый маршрутизатор узнает информацию об имеющихся в интерсети сетях и о расстоянии до них через соседние маршрутизаторы.

При наличии альтернативных маршрутов выбирается маршрут с наименьшей метрикой.

Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях. В больших сетях они засоряют линии связи интенсивным широковещательным трафиком, к тому же изменения конфигурации могут отрабатываться по этому алгоритму не всегда корректно.

Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол **RIP**.

Алгоритмы состояния связей

Алгоритмы состояния связей (***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.

Каждый маршрутизатор использует граф сети для нахождения оптимальных по некоторому критерию маршрутов до каждой из сетей, входящих в составную сеть.

Широковещательная рассылка используется здесь только при изменениях состояния связей, что происходит в надежных сетях не так часто.

Для того, чтобы понять, в каком состоянии находятся линии связи, подключенные к его портам, маршрутизатор периодически обменивается короткими пакетами (***HELLO***) со своими ближайшими соседями. Этот трафик также широковещательный, но он циркулирует только между соседями и поэтому не так засоряет сеть.

Объявления о состоянии связей не повторяются периодически, а передаются только в том случае, когда с помощью сообщений **HELLO** было установлено изменение состояния какой либо связи. Поэтому служебный трафик протоколов ***LSA*** гораздо менее интенсивный, чем у протоколов ***DVA***.

Протоколами, основанными на алгоритме состояния связей, в стеке TCP/IP являются протоколы **OSPF** и **IS-IS**.

Протокол RIP (Routing Information Protocol)

Протокол RIP является дистанционно-векторным (DVA) протоколом внутренней маршрутизации (IGP).

Протокол RIP использует ***эвристический алгоритм динамического программирования Беллмана-Форда***, и решение, найденное с его помощью является не оптимальным, а близким к оптимальному.

Преимуществом протокола RIP является его *вычислительная простота*, а недостатками - *увеличение трафика* при периодической рассылке широковещательных пакетов и *неоптимальность* найденного маршрута.

Процесс работы протокола состоит в рассылке, получении и обработке ***векторов расстояний*** до IP-сетей, находящихся в области действия протокола, то

есть в данной RIP-системе.

Результатом работы протокола на конкретном маршрутизаторе является таблица, где для каждой сети данной RIP-системы указано расстояние в ***хопах*** до этой сети и адрес следующего маршрутизатора.

Номер сети, адрес следующего маршрутизатора и расстоянии до этой сети вносятся в таблицу маршрутизации.

Алгоритм построения таблицы маршрутизации в протоколе RIP

Вектором расстояний называют набор пар (“IP-адрес сети", "Расстояние до этой сети").

Расстояние до сети, к которой маршрутизатор подключен непосредственно, примем равным 1.

Каждый маршрутизатор, на котором запущен модуль RIP, периодически широковещательно рассылает («объявляет») свой вектор расстояний. Вектор рассылается через все интерфейсы маршрутизатора, подключенные к сетям, входящим в RIP-систему.

Каждый маршрутизатор также периодически получает векторы расстояний от других маршрутизаторов.

Расстояния в этих векторах увеличиваются на 1, после чего сравниваются с данными в своей таблице маршрутизации, и, если расстояние до какой-то из сетей в полученном векторе оказывается меньше расстояния, указанного в таблице, значение из таблицы замещается новым (меньшим) значением, а адрес маршрутизатора, приславшего вектор с этим значением, записывается в поле "Следующий маршрутизатор" в этой строке таблицы.

После этого вектор расстояний, рассылаемый данным маршрутизатором, соответственно изменится.

Реализация протокола RIP

Существуют две версии протокола RIP: RIPv1 и RIPv2.

Версия 2 имеет некоторые усовершенствования, как то:

1. возможность маршрутизации сетей по модели CIDR (кроме адреса сети передается и маска),
2. поддержка мультикастинга,
3. возможность использования аутентификации RIP-сообщений и др.

Работа протокола RIP

Для каждой записи в таблице маршрутизации существует время жизни (TTL), контролируемое таймером.

Если для любой конкретной сети, внесенной в таблицу маршрутов, в течение 180 сек. не получен вектор расстояний, подтверждающий или устанавливающий новое расстояние до данной сети, то сеть будет отмечена как недостижимая (расстояние равно бесконечности, точнее 16).

Через определенное время модуль RIP производит "сборку мусора" - удаляет из таблицы маршрутов все сети, расстояние до которых бесконечно.

При получении сообщения типа "ответ" для каждого содержащегося в нем элемента вектора расстояний модуль RIP выполняет следующие действия:

1. проверяет корректность адреса сети и маски, указанных в сообщении;
2. проверяет, не превышает ли метрика (расстояние до сети) бесконечности (16);
3. некорректный элемент игнорируется;
4. если метрика меньше бесконечности, она увеличивается на 1;
5. производится поиск сети, указанной в рассматриваемом элементе вектора расстояний, в таблице маршрутизации;
6. если запись о такой сети в таблице маршрутов отсутствует и метрика в полученном элементе вектора меньше бесконечности, сеть вносится в таблицу маршрутов с указанной метрикой; в поле "Следующий маршрутизатор" заносится адрес маршрутизатора, приславшего сообщение; запускается таймер для этой записи в таблице;
7. если искомая запись присутствует в таблице с метрикой больше, чем объявленная в полученном векторе, в таблицу вносятся новые метрика и, соответственно, адрес следующего маршрутизатора; таймер для этой записи перезапускается;
8. если искомая запись присутствует в таблице и отправителем полученного вектора был маршрутизатор, указанный в поле "Следующий маршрутизатор" этой записи, то таймер для этой записи перезапускается; более того, если при этом метрика в таблице отличается от метрики в полученном векторе расстояний, в таблицу вносится значение метрики из полученного вектора;
9. во всех прочих случаях рассматриваемый элемент вектора расстояний игнорируется.

Сообщения типа "ответ" рассылаются модулем RIP каждые 30 с по широковещательному или мультикастинговому (только RIPv2) адресу.

Рассылка "ответа" может происходить также вне графика, если маршрутная таблица была изменена (triggered response). Стандарт требует, чтобы triggered response рассылался не немедленно после изменения таблицы маршрутов, а через случайный интервал длительностью от 1 до 5 с. Эта мера позволяет несколько снизить нагрузку на сеть.

В случае triggered response посылается информация только о тех сетях, записи о которых были изменены.

Информация о сетях с бесконечной метрикой посылается только в том случае, если она была недавно изменена.

При получении сообщения типа "запрос" с адресом 0.0.0.0 маршрутизатор рассылает в соответствующую сеть обычное сообщение типа ответ. При получении запроса с любым другим значением в поле (полях) "IP Address" посылается ответ, содержащий информацию только о сетях, которые указаны. Такой ответ посылается на адрес запросившего маршрутизатора (не широковещательно).

Протокол OSPF

Протокол OSPF (**Open Shortest Path First** – выбор кратчайшего пути первым, открытый протокол) принят в 1991 г. OSPF разбивает процесс

построения таблицы маршрутизации на два этапа:

1 этап. Каждый маршрутизатор строит граф связей сети (вершины – маршрутизаторы и IP-сети, ребра – интерфейсы маршрутизаторов), для чего маршрутизаторы обмениваются с соседями топологической информацией (без ее модификации) – **объявлениями о состоянии связей сети** (**Link State Advertisements, LSA**). В результате все маршрутизаторы сети располагают идентичными сведениями о графе сети, которые хранятся в ***базе данных о топологии сети***.

2 этап. Нахождение оптимальных маршрутов с помощью полученного графа (используется итеративный алгоритм Дийкстры). Каждый маршрутизатор считает себя центром сети и ищет оптимальный маршрут до каждой известной ему сети. В каждом найденном т.о. маршруте запоминается в табл. маршрутизации только один шаг – до следующего маршрутизатора (в соотв. с принципом одношаговой маршрутизации). При наличии нескольких маршрутов с одинаковой метрикой до сети назначения, в табл. маршрутизации запоминаются первые шаги всех маршрутов.

Корректировка таблиц маршрутизации

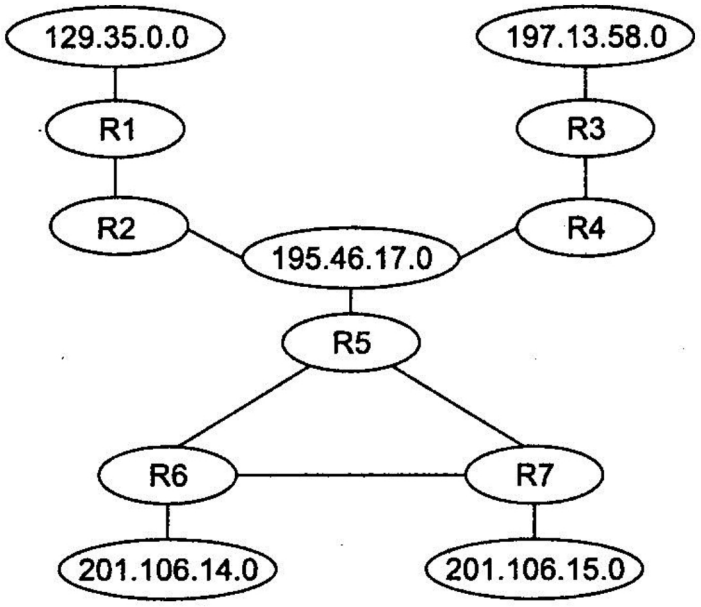
Для контроля состояния связей и соседних маршрутизаторов OSPF-маршрутизаторы регулярно (каждые 10 с.) передают друг другу сообщениями HELLO. При выявлении любых изменений (HELLO перестали поступать от одного из маршрутизаторов, начали приходить HELLO от нового, появилась новая связь) маршрутизатор вносит изменения в свою базу данных и рассылает непосредственным соседям LSA об этом изменении. После корректировки графа сети каждый маршрутизатор заново ищет оптимальные маршруты и корректирует свою таблицу маршрутизации.

Конвергенция таблиц маршрутизации к новому стабильному состоянию происходит достаточно быстро (время передачи LSA плюс время работы алгоритма Дийкстры).

Если же состояние сети не меняется, LSA не генерируются, таблицы маршрутизации не корректируются – экономится пропускная способность сети и вычислит ресурсы маршрутизаторов. Исключение: каждые 30 минут OSPF-маршрутизаторы обмениваются всеми записями базы данных топологической информации (т.е. синхронизируют их).

Протокол OSPF: связи и метрики

OSPF-маршрутизаторы соединены как с локальными сетями, так и непосредственно между собой глобальными двухточечными линиями связи. Протокол OSPF распространяет в своих объявлениях информацию о связях двух типов: маршрутизатор – маршрутизатор и маршрутизатор – сеть. Каждая связь характеризуется метрикой. В OSPF по умолчанию метрика учитывает пропускную способность каналов связи (допускается использовать две других: учитывающих задержки и надежность передачи). Для каждой из метрик протокол OSPF строит отдельную таблицу маршрутизации.



Граф сети, построенный протоколом OSPF

Выбор нужной таблицы маршрутизации происходит в зависимости от значений битов TOS в заголовке IP-пакета.

Стандартные для OSPF (и др. протоколов) значения расстояний для метрики, отражающей пропускную способность: сети Ethernet – 10, Fast Ethernet – 1, канал T1 – 65, канал 56 кбит/с – 1785. При использовании других, более скоростных каналов администратор задает другую шкалу, так что бы 1 соответствовала наиболее скоростному каналу.

При выборе оптимального пути на графе с каждым ребром графа связана метрика, которая добавляется к пути, если данная метрика в него входит.

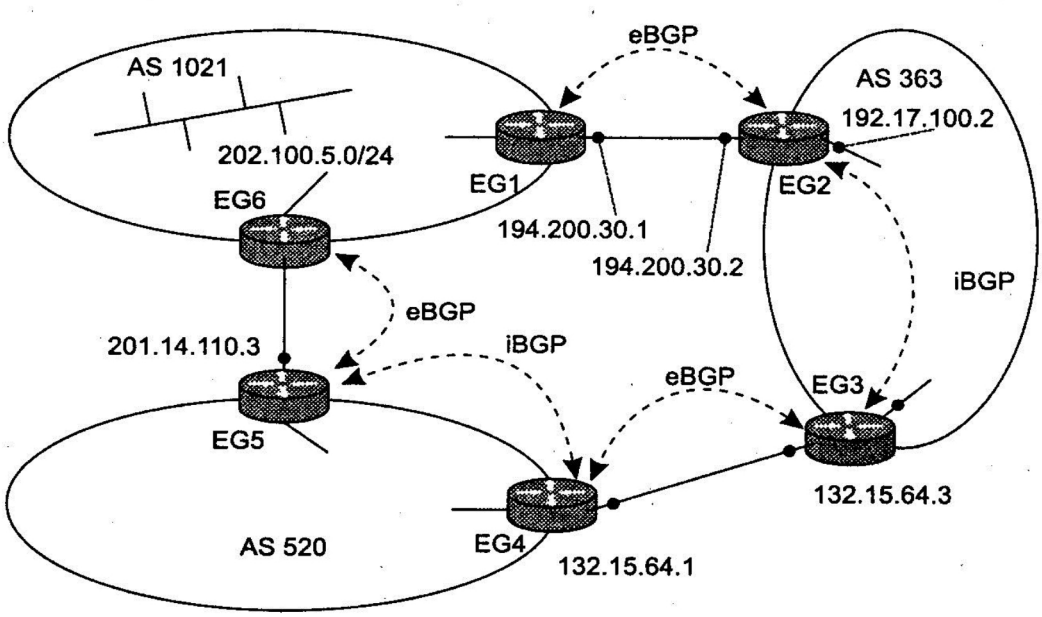
Протокол OSPF разрешает хранить в таблице маршрутизации несколько маршрутов к одной сети, если они обладают равными метриками. В этом случае маршрутизатор может реализовать режим баланса загрузки маршрутов, отправляя пакеты попеременно по каждому из маршрутов.

Вычислительная сложность протокола OSPF быстро растет с увеличением размерности сети. Поэтому сеть разбивают на области (***«области сети»*** не путать с AS). Только пограничные для областей маршрутизаторы обмениваются сокращенной информацией.

Протокол BGP

Пограничный шлюзовой протокол **(Border Gateway Protocol, BGPv4)** является сегодня основным протоколом обмена маршрутной информацией между автономными системами (AS) Интернета.

В каждой из AS имеется несколько маршрутизаторов, исполняющих роль внешних шлюзов. На каждом из них работает протокол BGPv4. Маршрутизатор взаимодействует с другими маршрутизаторами по протоколу BGPv4, только в том случае, если администратор ***явно*** указывает при конфигурировании, что эти маршрутизаторы являются его ***соседями***.



Поиск маршрута между автономными системами
с помощью протокола BGP

Для установления сеанса с указанными соседями BGP-маршрутизаторы используют протокол TCP (порт 179). При установлении BGP-сеанса могут применяться разнообразные способы аутентификации маршрутизаторов, повышающие безопасность работы AS.

Основным сообщением протокола BGP является сообщение ***UPDATE*** (обновить) - триггерное объявление (посылаемое только при возникновении изменений) о достижимости сетей, относящихся к его собственной AS.

В одном сообщении UPDATE можно объявить об одном новом маршруте или аннулировать несколько переставших существовать. Под маршрутом в BGP понимается последовательность AS, которую нужно пройти на пути к указанной в адресе сети.

Информация о маршруте (BGP-route) к сети (Network/Mask_length) задается так:

BGP-route = AS_Path; NextHop; Network/Mask_length;

Здесь **AS_Path** – набор номеров AS; **NextHop** – IP-адрес маршрутизатора, через который нужно передавать пакеты в сеть **Network/Mask_length.**

Тема лекции: **Дополнительные функции маршрутизаторов IP-сетей**

План лекции:

1. Фильтрация пользовательского трафика.
2. Списки доступа.
2. Стандарты QoS в IP-сетях.
3. Трансляция сетевых адресов (NAT).
4. Функции маршрутизаторов.
5. Классификация маршрутизаторов.

Фильтрация пользовательского трафика

Маршрутизатор – пограничное устройство, соединяющее сеть с внешним миром. Поэтому на него возлагают **функции защиты сети от внешних атак с помощью фильтрации пользовательского трафика** в соответствии с различными признаками: адресами отправителя и получателя, типа протокола, типа приложения, генерирующего трафик и т.п.

*Под **фильтрацией** понимается нестандартная обработка IP-пакетов, приводящая к отбрасыванию некоторых пакетов или изменению их маршрута.*

Типичные условия фильтрации в маршрутизаторах:

- IP-адрес отправителя и получателя.
- MAC-адрес отправителя и получателя.
- Идентификатор интерфейса с которого поступил пакет.
- Тип протокола, сообщение которого несет IP-пакет (TCP, UDP, ICMP, HTTP, OSPF).

При наличии фильтра маршрутизатор сначала **сравнивает условия фильтрации с признаками пакета** и при их **совпадении** выполняет предписанные действия, например:

- **отбрасывает** пакет (**drop**),
- **изменяет** его **маршрут**,
- **помечает** как вероятного **кандидата на удаление** при возникновении перегрузки.

Списки доступа (Access Control List, ACL)

Access control lists (ACL) - списки доступа широко используются современными маршрутизаторами, не ограничивающимися простой **проверкой адресов** и **продвижением пакетов**. Сейчас они выполняют **инкапсуляцию**, **туннелирование**, **преобразование протоколов**, действия по обеспечению некоторой безопасности.

Разнообразие каналов связи, увеличивающееся использование сети Интернет и появление, наряду с данными, передачи голоса, видео - все это приводит к усложнению проходящего через маршрутизатор трафика.

Проблемы **безопасности**, **балансировки нагрузки** и **распределения трафика по приоритетам** поставили вопрос о необходимости фильтрации пакетов.

В итоге для этого стали применяться списки доступа различных типов. С их

помощью маршрутизатор не становится полнофункциональным межсетевым экраном (для этого он должен уметь не только анализировать трафик, но и менять свои параметры в соответствии с накопленной статистикой трафика).

Назначение и принцип работы списка доступа

Список доступа - это упорядоченный набор операторов, каждый из которых разрешает или запрещает прохождение пакета через интерфейс.

Операторы списка доступа применяются к каждому пакету последовательно, начиная с первого, до тех пор, пока не будет найдено соответствие или достигнут конец списка. При первом же совпадении маршрутизатор определяет, сбросить пакет или принять его.

Поскольку после первого же совпадения прекращается проверка условий, то их порядок следования имеет важное значение. Поместив в начало списка наиболее часто встречающиеся условия, можно **существенно снизить нагрузку на маршрутизатор**.

Кроме фильтрации пакетов, списки доступа применяются для выделения трафика определенных хостов при его передаче по защищенному туннелю через Интернет, для перенаправления трафика по другому маршруту, для фильтрации изменений в таблицах маршрутизации, для управления доступом к виртуальным терминалам.

Списки доступа могут применяться **на входе** в маршрутизатор и на его **выходе**.

В **первом** случае сначала выполняется проверка пакета, а затем, если он пропускается, - его маршрутизация. При сбросе пакета (drop) по адресу его отправителя посылается ICMP-сообщение "хост недоступен" (host unreachable).

Во **втором** случае пакет сначала маршрутизируется и только потом его параметры сравниваются с условиями списка.

Списки доступа в маршрутизаторах Cisco

В качестве типичного примера рассмотрим синтаксис списков доступа, разработанный компанией **Cisco Systems** для операционной системы **Cisco IOS**.

Списки доступа (access-lists) позволяют управлять прохождением трафика через интерфейсы роутера, разрешая или запрещая передачу пакетов, удовлетворяющих указанным условиям. Списки доступа задают условия, которые роутер проверяет перед выполнением каких-либо действий.

Важнейшие области использования списков доступа:

1. Управление передачей пакетов на интерфейсах.
2. Управление доступом к виртуальным терминалам роутера и управлению через SNMP.
3. Ограничение информации, передаваемой протоколами маршрутизации.

Cisco IOS поддерживает несколько видов списков доступа для протокола IP:

- **Стандартные** списки доступа (проверяют адрес отправителя пакета).
- **Расширенные** списки доступа (проверяют адрес отправителя, адрес получателя и еще ряд параметров пакета).

- **Динамические** расширенные списки доступа (имеют конечное «время жизни» и условия применения).

Списки доступа либо **нумеруются**, либо **именуются**. Использование нумерованных, либо именованных списков доступа определяется областью их применения (некоторые протоколы требуют использования только нумерованных списков, некоторые - допускают как именованные, так и нумерованные списки).

Если используются нумерованные списки, то номера их должны лежать в определенных диапазонах, в зависимости от области применения списка (см. табл. на 18 слайде).

Задачи и правила построения списков доступа для различных протоколов различны, но, в общем, можно выделить **два этапа** работы с любыми списками доступа:

1. Сначала, необходимо **создать список доступа.**
2. Затем **применить** его к соответствующему **интерфейсу**, линии или логической операции, выполняемой роутером.

Создание списков доступа для протокола IP. Стандартный список доступа

Имеет ограниченные функциональные возможности: позволяет фильтровать пакеты только по адресу источника. Имеет следующий (базовый) формат (задается из режима глобальной конфигурации, то есть перед командой должен быть промпт: **Router(config)#**):

access-list access-list-number {**deny** | **permit**} {source-address [source-wildcard] | **any**} [log]

Жирным шрифтом выделены параметры оператора, которые вносятся в список в неизменном виде, а обычным шрифтом - которые заменяются символьными или числовыми значениями. В фигурных скобках находятся два параметра, один из которых обязательно должен присутствовать в операторе. Параметры, заключенные в квадратные скобки, необязательны.

Параметры стандартного списка доступа

Access-list-number (номер списка) - это число от 1 до 99 (стандартный список доступа протокола IP). Начиная с версии IOS 12.0 для номера стандартного списка IP доступен интервал 1300-1999. Используются и числа из диапазона 700-799 для фильтрации пакетов по 48-битным MAC-адресам.

Параметры **permit** или **deny** указывают, разрешается ли пакетам, удовлетворяющим условиям некоторого оператора списка, проходить через интерфейс. **Permit** - разрешить, **deny** - отказать.

Source-address (адрес источника) - это IP-адрес узла или группы узлов. Его задание осуществляется с использованием маски шаблона (**source-wildcard**). Действие этой маски противоположно по сравнению с маской подсети (subnet mask): *двоичный нуль предполагает строгое совпадение в соответствующей позиции, а двоичная единица означает произвольное значение*. Попросту: **source-wildcard – это инвертированная маска подсети.**

Ключевое слово **log** задает ведение журнала.

Примеры стандартных списков доступа

Все команды вводятся в режиме глобальной конфигурации маршрутизатора **Router(config)# :**

1. Разрешение прохождения через маршрутизатор пакетов от всех хостов сети класса C - 140.12.11.0

access-list 1 permit 140.12.11.0 0.0.0.255

2. Запрещение прохождения через маршрутизатор пакетов от всех хостов сети класса C - 140.12.11.0

access-list 1 deny 140.12.11.0 0.0.0.255

3. Разрешение прохождения через маршрутизатор пакетов хоста с IP-адресом 140.12.11.5

access-list 1 permit 140.12.11.5 0.0.0.0

или

access-list 1 permit host 140.12.11.5

4. Запрещение прохождения через маршрутизатор пакетов хоста с IP-адресом 140.12.11.5

access-list 1 deny 140.12.11.5 0.0.0.0

По умолчанию каждый список доступа в конце содержит неявный оператор **deny any**, который отсеивает весь трафик, кроме явно разрешенного, то есть:

access-list 1 deny any

Создание списков доступа для протокола IP.
Расширенный список доступа

Обеспечивает более гибкую фильтрацию пакетов. Он позволяет отбирать пакеты по адресу источника и по адресу назначения, а также по протоколу, порту источника, порту назначения и даже по значению разрядов в некоторых полях внутри пакета. Расширенный список доступа имеет следующий (базовый) формат (задается из режима глобальной конфигурации, то есть перед командой должен быть промпт: **Router(config)#**):

access-list access-list-number {**deny** | **permit**} {**protocol** | **protocol-keyword**}
{source-address [source-wildcard] | **any**}
[destination-address [destination-wildcard] | **any**]
[precedence precedence] [tos tos] [established] [log]

Параметры расширенного списка доступа

Access-list-number (номер списка) - число от 100 до 199, идентифицирующее операторы из расширенного списка доступа IP. Начиная с версии IOS 12.0 для номера списка доступен интервал 2000-2699.

Protocol-keyword задает протокол, подлежащий фильтрации, например, IP, TCP, UDP, ICMP и т.д.

Поддерживаются логические операции применительно к номерам портов (или другим параметрам): **lt** (**less than** - меньше), **gt** (**greater than** - больше), **eq**

(**equal** -равно), **neq** (**not equal** - не равно).

Чтобы задать порт источника - **source-port** или порт назначения - **destination-port**, можно указать номер или мнемонический код порта, либо использовать логическую операцию для задания диапазона портов.

Пример расширенного списка доступа

Следующий пример иллюстрирует использование логических операций:

access-list 101 permit TCP any host 140.12.11.10 eq WWW
access-list 101 permit ICMP any host 140.12.11.10 eq 8

Первый оператор разрешает передачу пакетов TCP с любого узла по адресу 140.12.11.10, если эти пакеты принадлежат типу Web. Вместо мнемонического кода WWW можно задать число 80, так как трафик Web обычно использует порт 80.

Второй оператор разрешает все адресованные указанному узлу сообщения ping - эхо запросы ICMP, которым назначен тип ICMP, равный 8 (синтаксис расширенных списков доступа позволяет задавать в конце оператора специфические для каждого протокола параметры; так, для протокола ICMP задаются не номера портов, как для протоколов UDP или TCP, а тип и если нужно - код сообщений ICMP). С учетом неявного оператора **deny any** в конце списка доступа весь остальной трафик (абсолютно весь) будет блокироваться.

Номера протоколов, используемые при создании списков доступа в маршрутизаторах Cisco Systems

Протокол	Диапазон номеров
IP	1-99 1300-1999 (с версии IOS 12.0)
Расширенный IP	100-199 2000-2699 (с версии IOS 12.0)
Коды Ethernet-type, алгоритм прозрачного моста (по типу протокола), алгоритм моста с маршрутизацией от источника (по типу протокола)**	200-299
DECnet и расширенный DECnet	300-399
XNS	400-499
Расширенный XNS	500-599
Apple Talk	600-699
Ethernet-адреса (MAC-адреса), алгоритм прозрачного моста (по коду производителя), алгоритм моста с маршрутизацией от источника (по коду производителя)**	700-799
IPX	800-899
Расширенный IPX	900-999
IPX SAP	1000-1099
Расширенный алгоритм прозрачного моста	1100-1199

** - В случае фильтрации пакетов на канальном уровне при назначении списка доступа интерфейсу вместо команды **access-group** используется команда **bridge-group**.

Назначение списков доступа на интерфейсы

Ошибкой многих начинающих (и не только) администраторов является то, что после создания списка доступа его просто "забывают" связать с каким-либо интерфейсом маршрутизатора. Список же начинает действовать только после связи его с интерфейсом.

К каждому интерфейсу можно применить стандартный и\или расширенный списки доступа, которые могут быть или **inbound (in)** - проверяется когда пакет поступает на вход интерфейса снаружи, или **outbound (out)** - проверка происходит, когда пакет приходит изнутри маршрутизатора на интерфейс (т.о.всего 4 ACL на интерфейс (порт)).

Для этого необходимо войти в конфигурацию нужного интерфейса и прописать список доступа (в данном случае как **group** - "группа"). Порядок внесения изменений в конфигурацию следующий (на примере протокола IP для интерфейса Ethernet 0/0):

```
Router# configure terminal
Router(config)# interface ethernet 0/0
Router(config-if)# ip access-group 1 in
```

Рекомендации по конфигурированию и применению списков доступа

1) использовать стандартный список доступа IP при фильтрации только по адресу источника;

2) в начало списка помещать наиболее часто встречающиеся условия для снижения нагрузки на роутер;

3) уделить внимание порядку следования операторов в списке доступа для исключения ошибок;

4) всегда применять список доступа к соответствующему интерфейсу и направлению фильтрации трафика;

5) список доступа отсеивает весь трафик, кроме явно разрешенного;

6) при составлении маски шаблона - двоичные нули означают совпадение, а двоичные единицы - произвольные символы;

7) операторы с более конкретными спецификациями всегда следует ставить перед операторами общего характера (для фильтрации по протоколу);

8) обращать внимание на правильность направления фильтрации трафика - "откуда-куда" местоположением оператора any.

Стандарты QoS в IP-сетях

Существует две системы стандартов **QoS** (**Quality of Service**) для IP-сетей:

Система интегрированного обслуживания (Integrated Services, **IntServ**), ориентированная на предоставление гарантий QoS для потоков конечных пользователей (поэтому технология IntServ используется в основном на периферии сети).

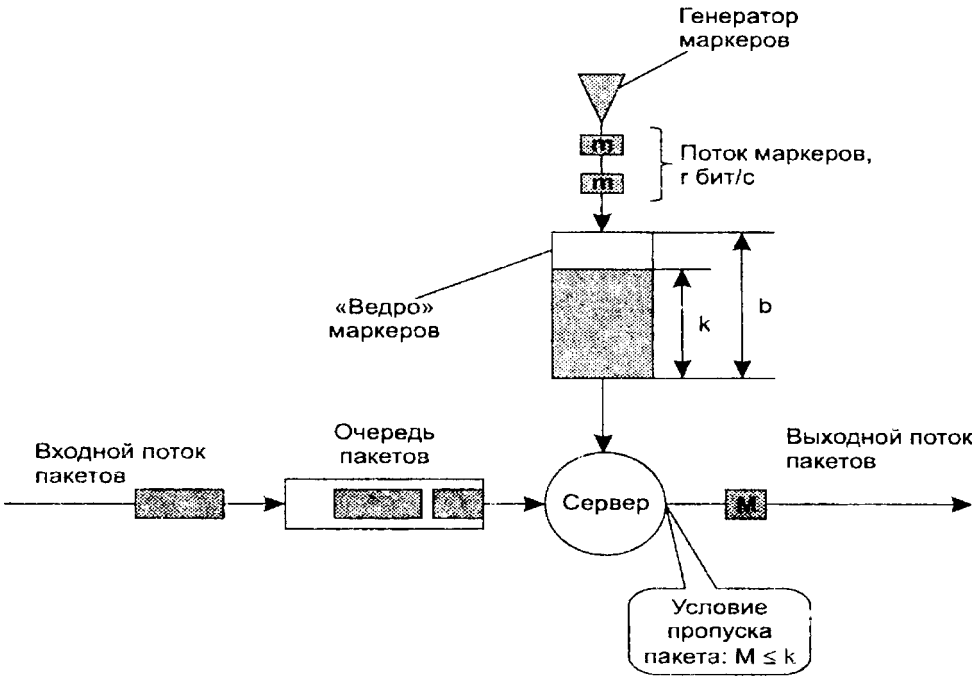
Система дифференцированного обслуживания (Differentiated Services, **DiffServ**) делает то же самое для классов трафика, и, следовательно, ее

предпочтительнее использовать на магистрали.

Обе системы используют все базовые элементы схемы поддержания параметров QoS, основанной на резервировании, то есть:

1) кондиционирование трафика;
2) сигнализацию для координации маршрутизаторов;
3) резервирование пропускной способности интерфейсов маршрутизаторов для потоков и классов;
4) приоритетные и взвешенные очереди.

Обе технологии (**IntServ** и **DiffServ**) используют в маршрутизаторах для профилирования и формирования трафика ***алгоритм ведра маркеров***.



Алгоритм ведра маркеров

Трансляция сетевых адресов

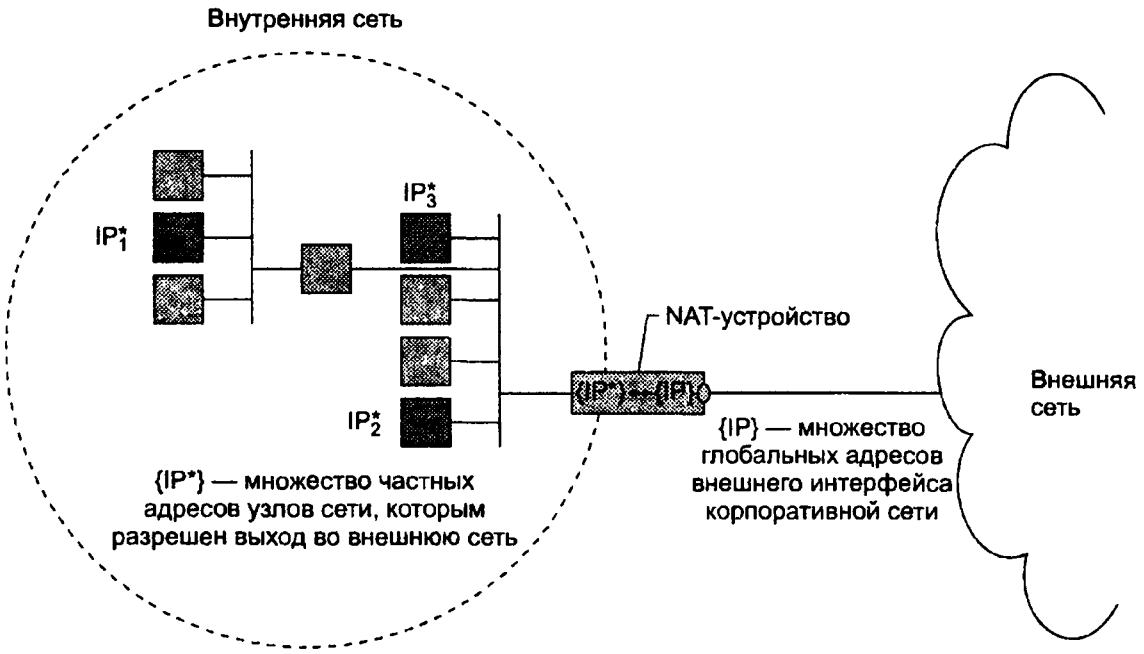
Причины подмены адресов:

- объединение «старых» (изолированных) сетей,
- дефицит IP-адресов,
- сокрытие истинных адресов из соображений безопасности.

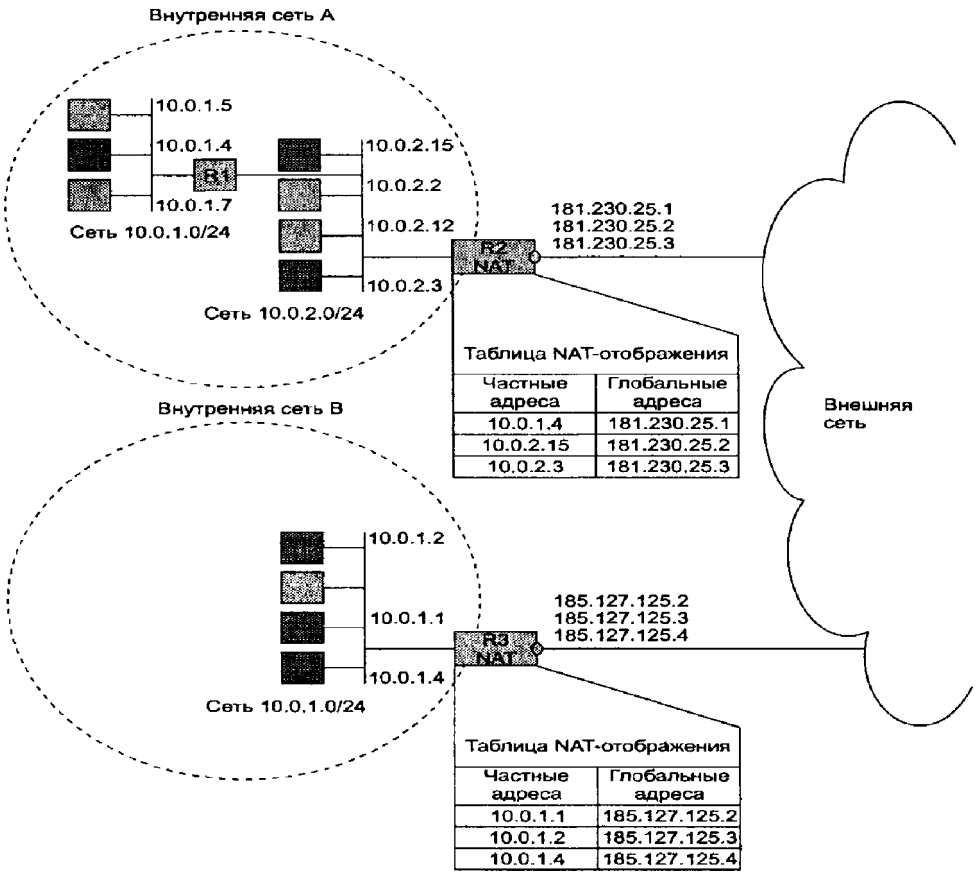
Традиционная технология **NAT (Network Address Translation)**, известна в двух вариантах:

Базовая трансляция сетевых адресов **(Basic Network Address Translation, Basic NAT).**

Трансляция сетевых адресов и портов **(Network Address&Ports Translation, NAPT)**.



Традиционная технология NAT



Базовая трансляция сетевых адресов для исходящих сеансов

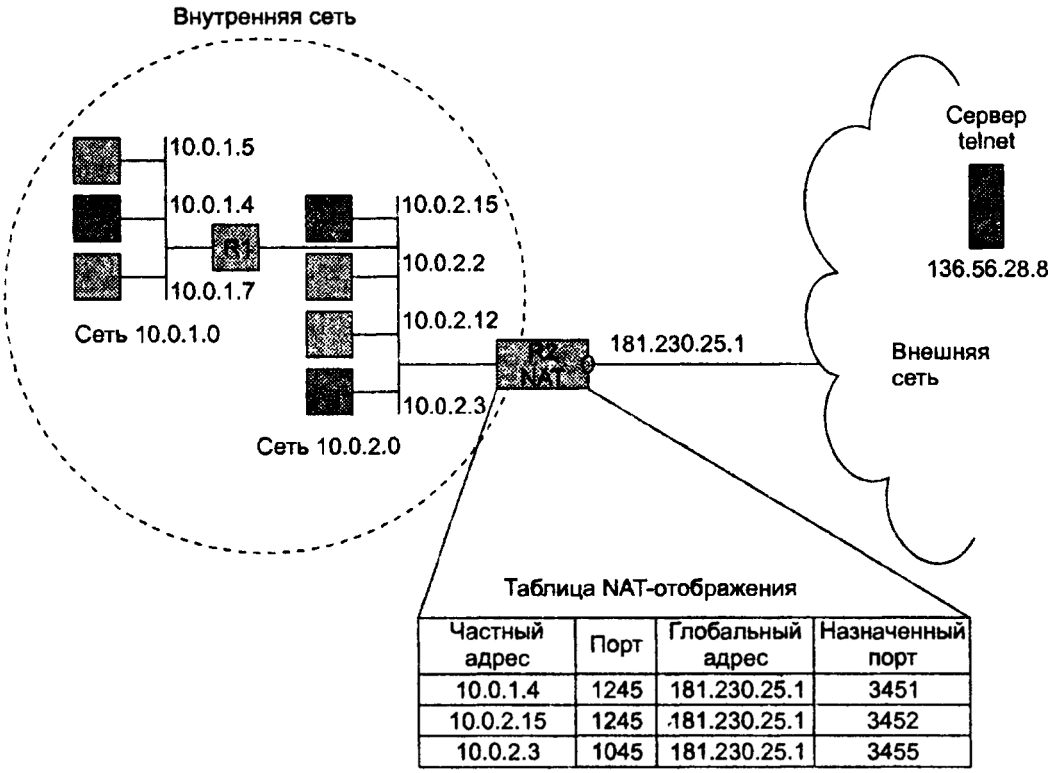


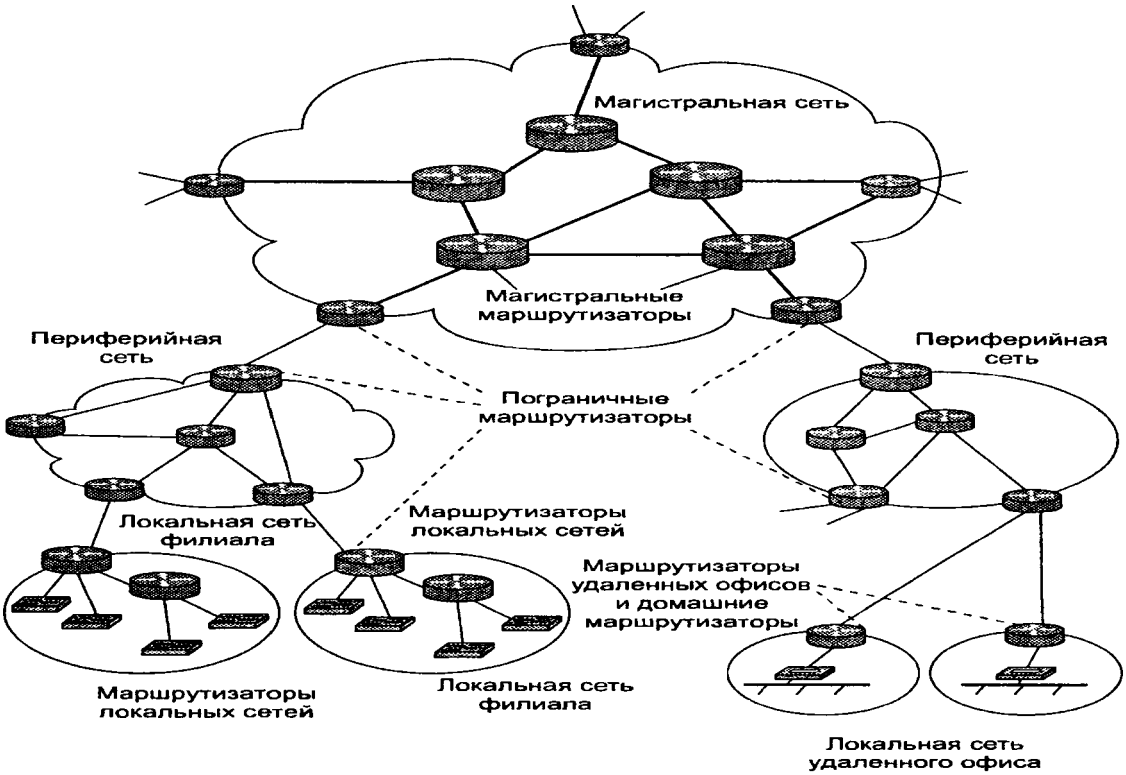
Таблица NAT-отображения

Частный адрес	Порт	Глобальный адрес	Назначенный порт
10.0.1.4	1245	181.230.25.1	3451
10.0.2.15	1245	181.230.25.1	3452
10.0.2.3	1045	181.230.25.1	3455

Трансляция сетевых адресов и портов NAPT



Функции маршрутизаторов



Классификация маршрутизаторов по областям применения