

The image features two large, thick black L-shaped brackets. One is positioned in the top-left corner, and the other is in the bottom-right corner. They frame the central text.

# ПРИНЦИПЫ ПЕРЕДАЧИ ИНФОРМАЦИИ ПО СЕТИ

# Принципы передачи информации по сети

---

Информация в локальных сетях, как правило, передается отдельными порциями, кусками, называемыми в различных источниках ***пакетами (packets), кадрами (frames)*** или блоками.

Причем **предельная длина этих *пакетов*** строго ограничена (обычно величиной в несколько килобайт). **Ограничена длина *пакета* и снизу** (как правило, несколькими десятками байт).

# Принципы передачи информации по сети

---



Важнейшим параметром является так называемое *время доступа* к сети (**access time**), которое определяется как временной интервал между моментом готовности абонента к передаче (когда ему есть, что передавать) и моментом начала этой передачи.

**Фактически, это время ожидания абонентом начала своей передачи.** Естественно, оно не должно быть слишком большим, иначе величина реальной, интегральной скорости передачи информации между приложениями сильно уменьшится даже при высокоскоростной связи.

Ожидание начала передачи связано с тем, что в сети не может происходить несколько передач одновременно.

# Принципы передачи информации по сети

---

Если бы вся требуемая информация передавалась каким-то абонентом сразу, непрерывно, без разделения на *пакеты*, то это *привело бы к монопольному захвату сети этим абонентом на довольно продолжительное время*. Все остальные абоненты вынуждены были бы ждать окончания передачи всей информации.

С тем чтобы уравнивать в правах всех абонентов, а также сделать примерно одинаковыми для всех них величину *времени доступа* к сети и *интегральную скорость передачи информации*, как раз и применяются *пакеты (кадры)* ограниченной длины.

# Принципы передачи информации по сети

---

Важно также и то, что при передаче больших массивов информации вероятность ошибки из-за помех и сбоев довольно высока.

Например, при характерной для локальных сетей величине вероятности одиночной ошибки в  $10^{-8}$  пакет длиной 10 Кбит будет искажен с вероятностью  $10^{-4}$ , а массив длиной 10 Мбит – уже с вероятностью  $10^{-1}$ .

К тому же выявить ошибку в массиве из нескольких мегабайт намного сложнее, чем в пакете из нескольких килобайт. А при обнаружении ошибки и невозможности ее исправить придется повторить передачу всего большого массива.

# Принципы передачи информации по сети

---

Каждый пакет помимо собственно данных, которые требуется передать, должен содержать некоторое количество **служебной информации**. Прежде всего, это адресная информация, которая определяет, от кого и кому передается данный пакет (как на почтовом конверте – адреса получателя и отправителя).

Если порция передаваемых данных будет очень маленькой (например, несколько байт), то доля служебной информации станет непозволительно высокой, что резко снизит интегральную скорость обмена информацией по сети.

# Принципы передачи информации по сети

---



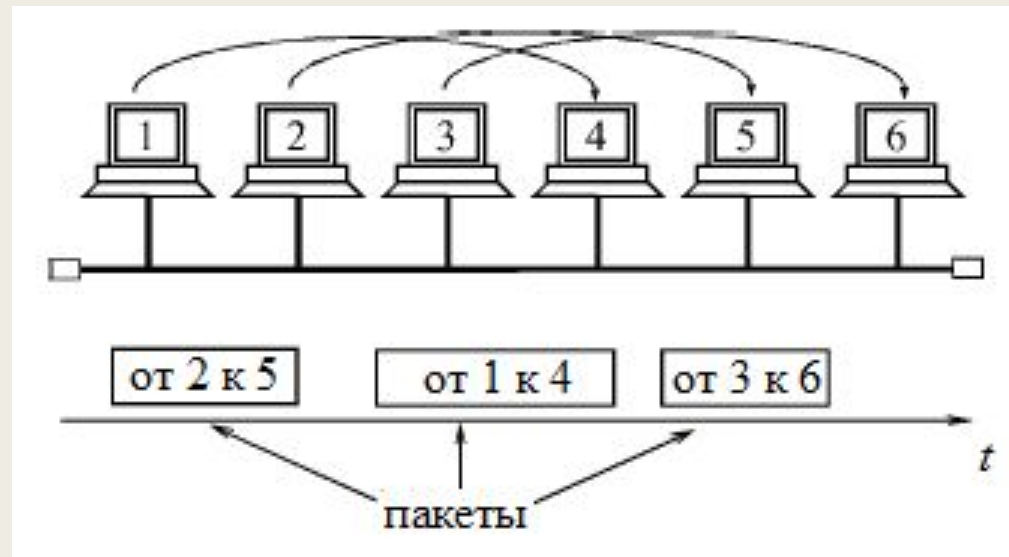
Существует некоторая оптимальная длина пакета (или оптимальный диапазон длин пакетов), при которой средняя скорость обмена информацией по сети будет максимальна.

Эта длина **не является неизменной величиной**, она зависит от уровня помех, метода управления обменом, количества абонентов сети, характера передаваемой информации, и от многих других факторов. **Имеется диапазон длин, который близок к оптимуму.**

# Принципы передачи информации по сети

Таким образом, процесс информационного обмена в сети представляет собой чередование *пакетов*, каждый из которых содержит информацию, передаваемую от абонента к абоненту.

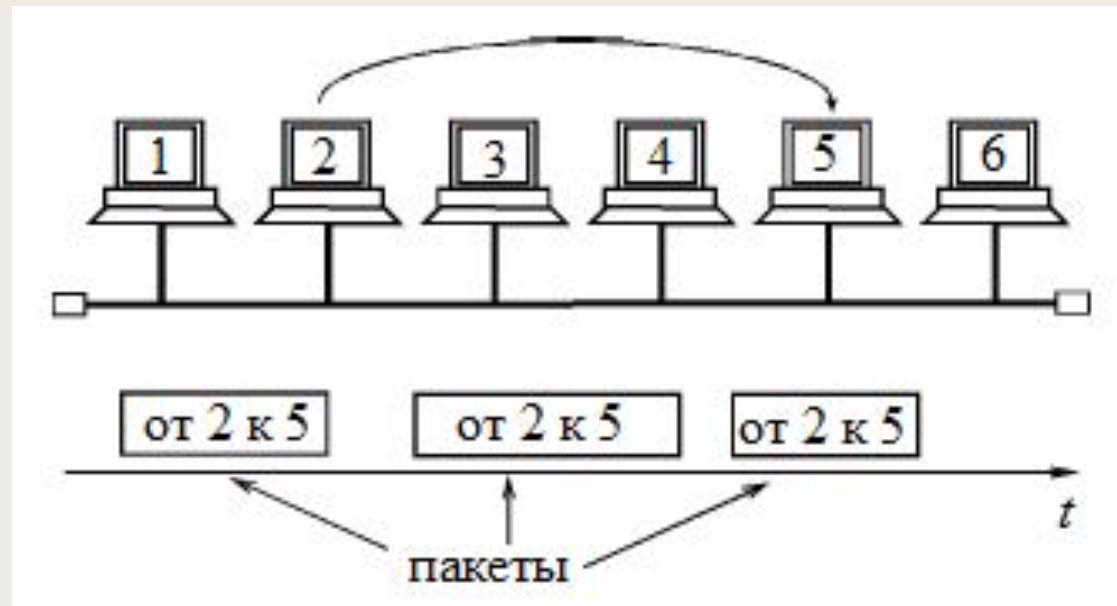
Обычно в сети чередуются *пакеты*, посланные разными абонентами





# Принципы передачи информации по сети

В частном случае все эти пакеты могут передаваться одним абонентом (когда другие абоненты не хотят передавать).



# Структура пакета

---

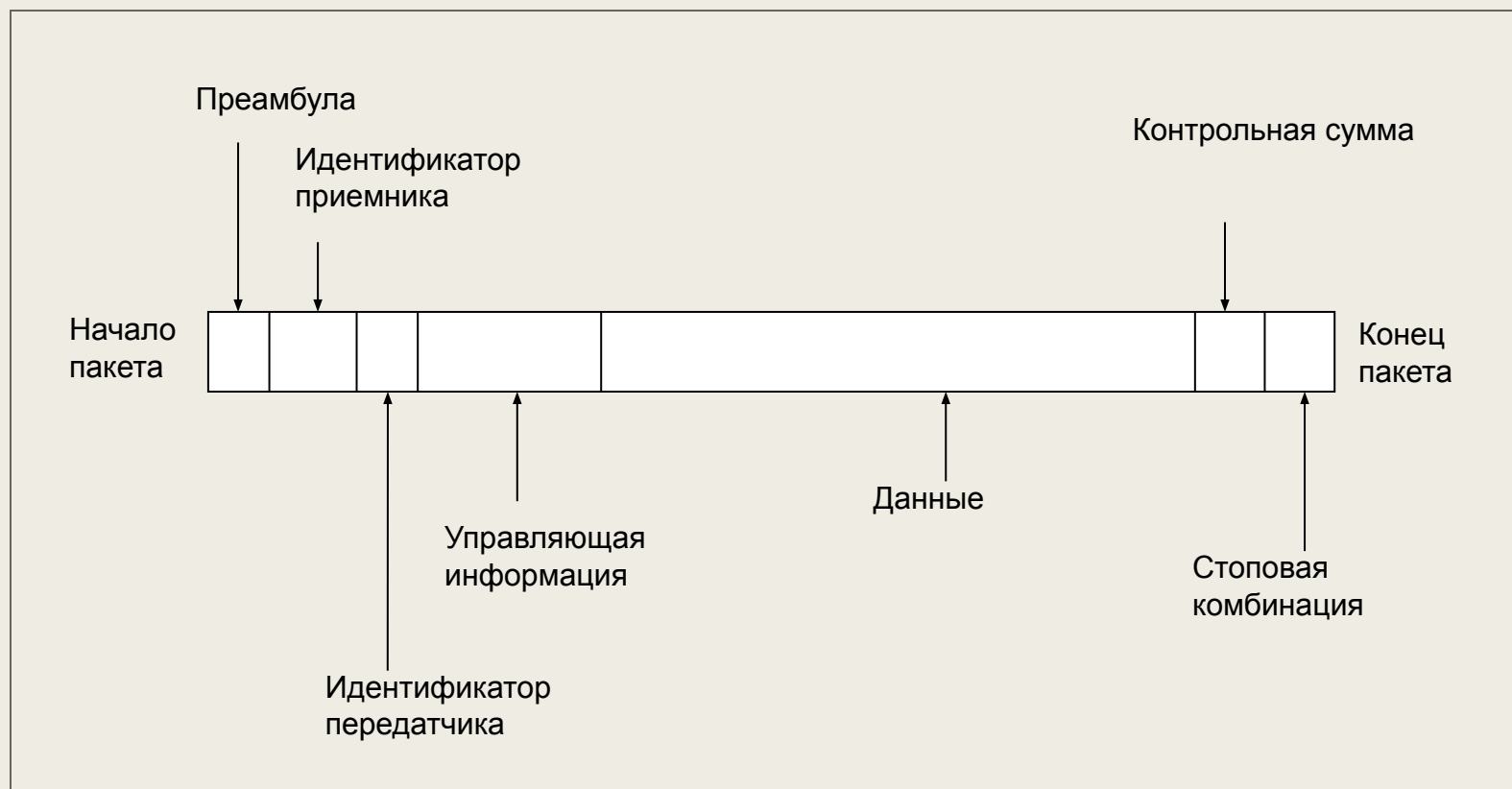
Структура и размеры *пакета* в каждой сети жестко определены стандартом на данную сеть и связаны, прежде всего, с аппаратурными особенностями данной сети, выбранной топологией и типом среды передачи информации. Кроме того, эти параметры зависят от используемого протокола (порядка обмена информацией).



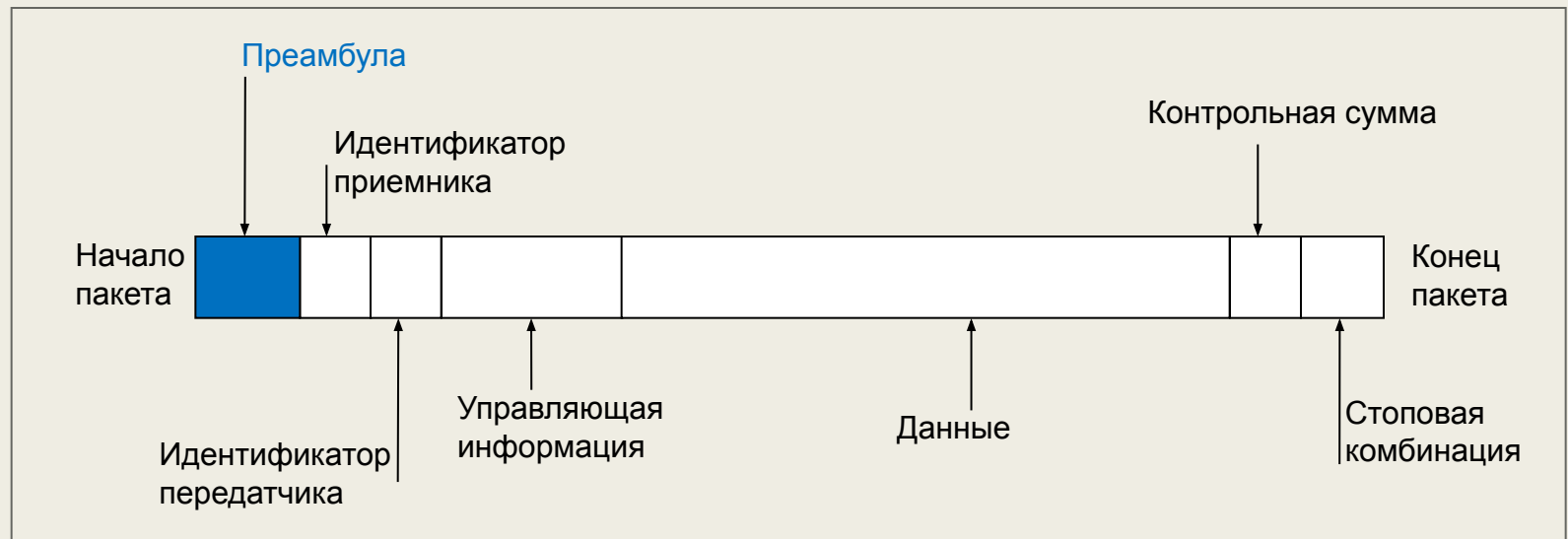
Но существуют некоторые **общие принципы формирования структуры пакета**, которые учитывают характерные особенности обмена информацией по любым локальным сетям.

# Структура пакета

В наиболее общем случае пакет содержит в себе следующие **7 основных полей**.



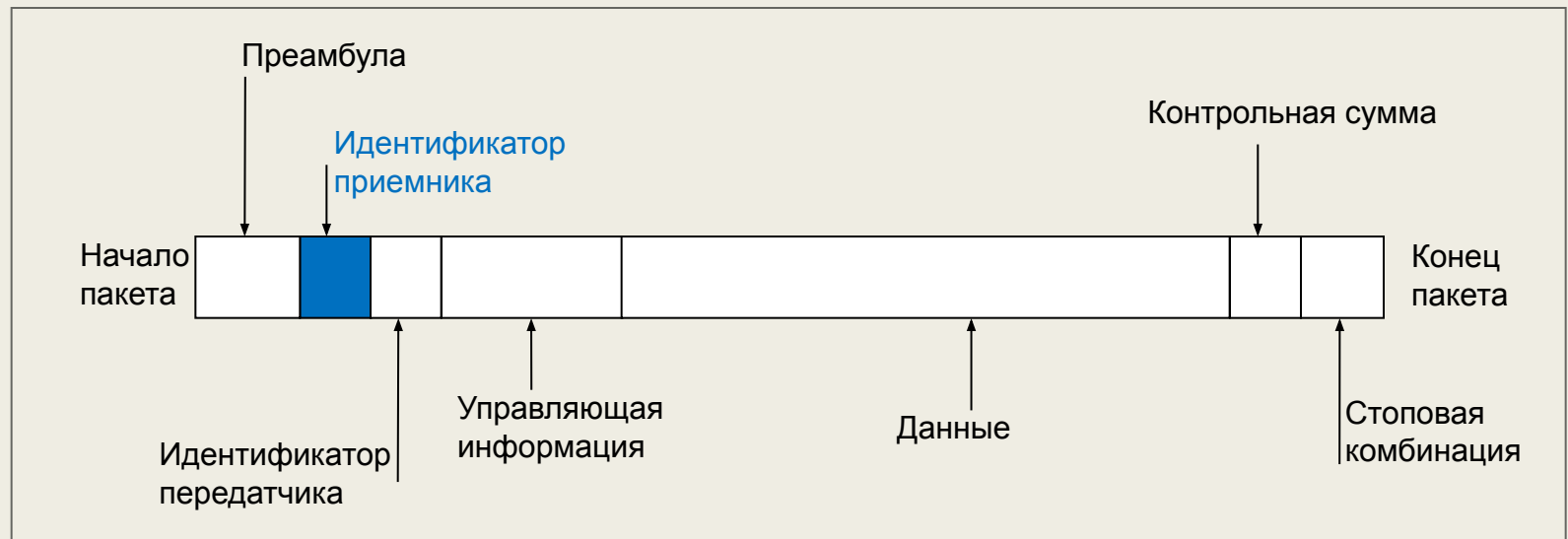
# Структура пакета. Преамбула



**Стартовая комбинация бит (преамбула)** - обеспечивает предварительную настройку аппаратуры адаптера или другого сетевого устройства на прием и обработку пакета.

Это поле может полностью отсутствовать или же сводиться к единственному стартовому биту.

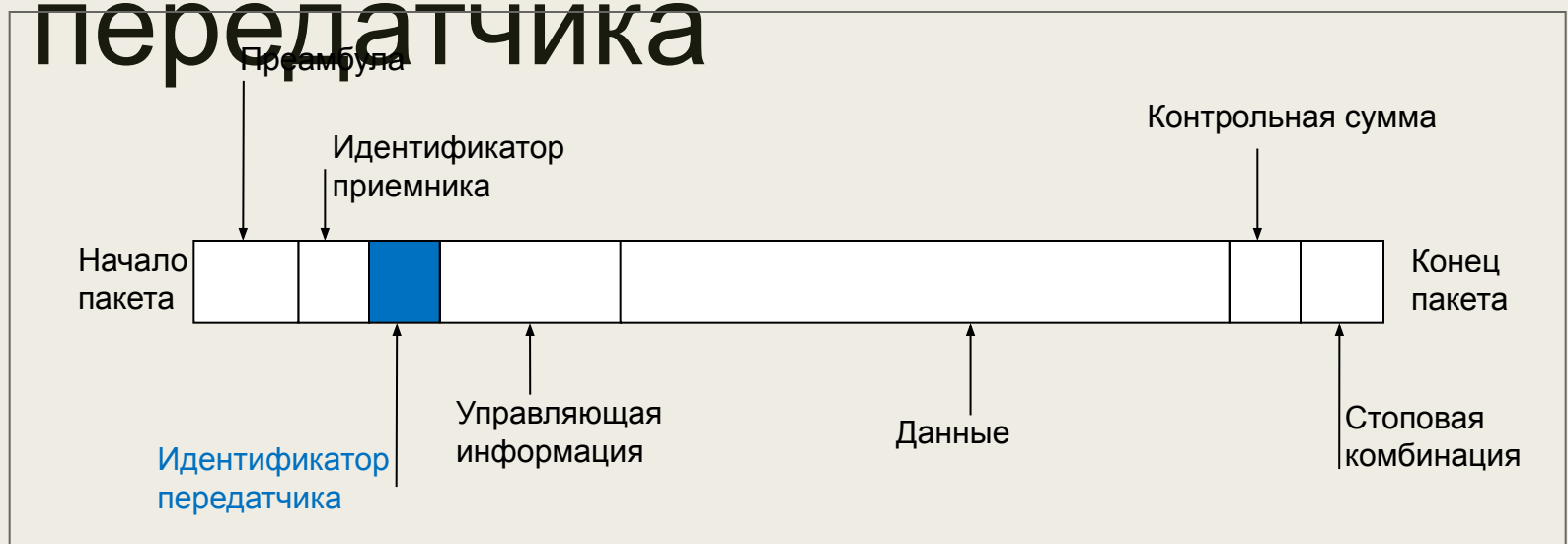
# Структура пакета. Идентификатор приемника



**Сетевой адрес (идентификатор) принимающего абонента,** то есть индивидуальный или групповой номер, присвоенный каждому принимающему абоненту в сети. Этот адрес позволяет приемнику распознать пакет, адресованный ему лично, группе, в которую он входит, или всем абонентам сети одновременно (при широком вещании).

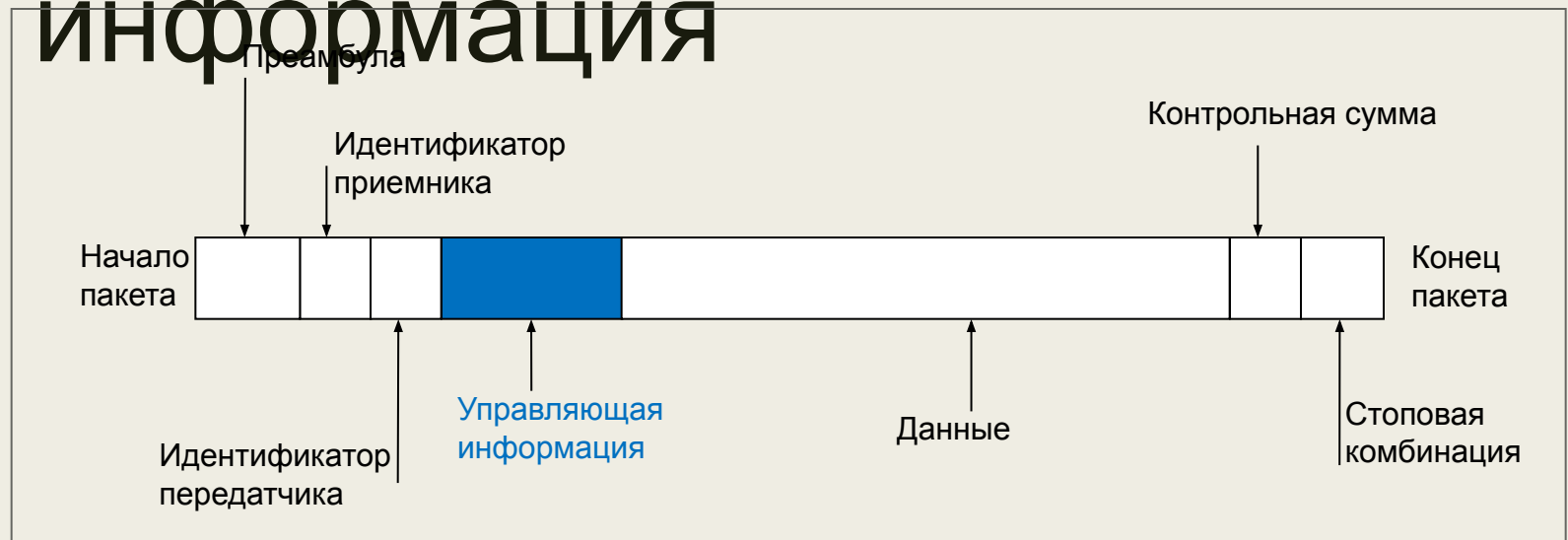
# Структура пакета. Идентификатор

## передатчика



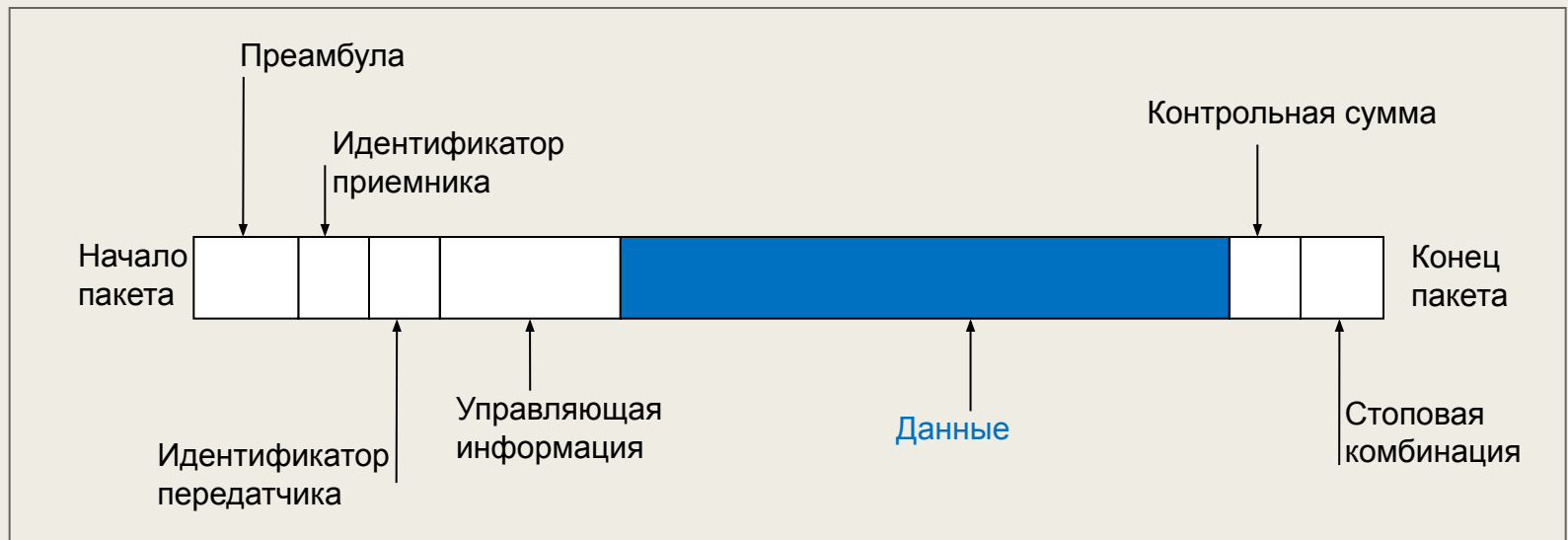
**Сетевой адрес (идентификатор) передающего абонента, то есть индивидуальный номер, присвоенный каждому передающему абоненту. Этот адрес информирует принимающего абонента, откуда пришел данный пакет. Включение в пакет адреса передатчика необходимо в том случае, когда одному приемнику могут попеременно приходить пакеты от разных передатчиков.**

# Структура пакета. Управляющая информация



**Служебная информация**, которая может указывать на тип пакета, его номер, размер, формат, маршрут его доставки, на то, что с ним надо делать приемнику и т.д.

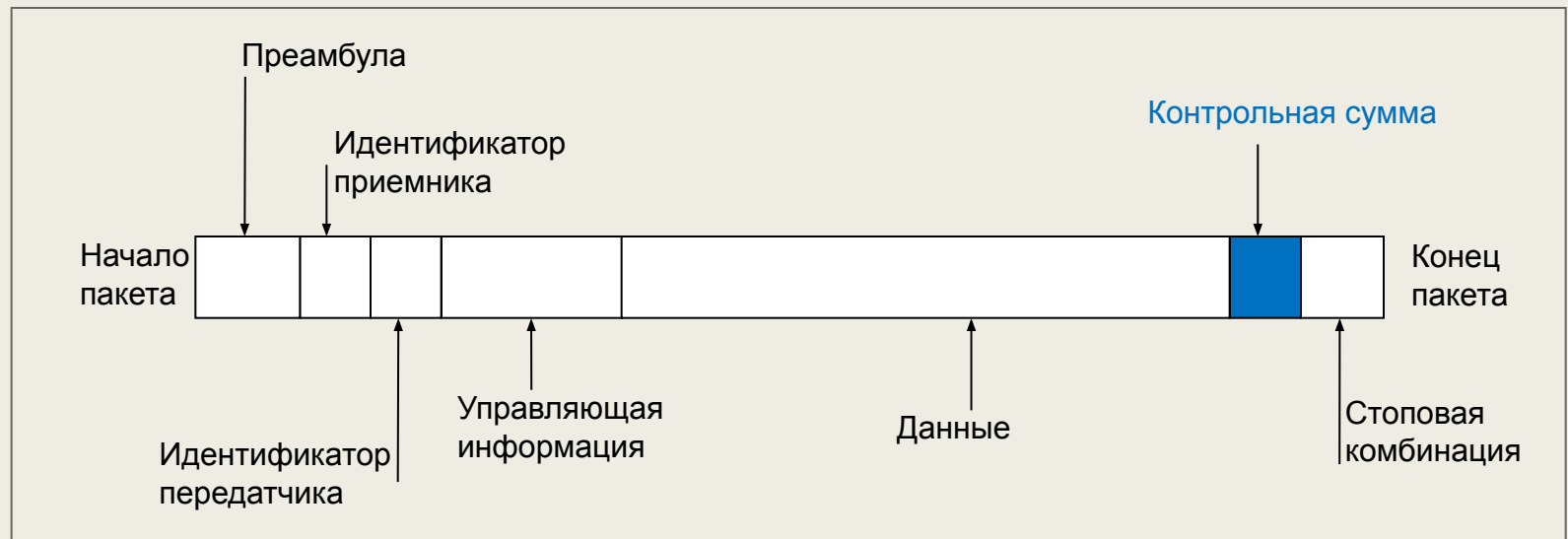
# Структура пакета. Данные



**Данные (поле данных)** – это та информация, ради передачи которой используется пакет. В отличие от всех остальных полей пакета поле данных имеет переменную длину, которая, собственно, и определяет полную длину пакета. Существуют специальные управляющие пакеты, которые не имеют поля данных.



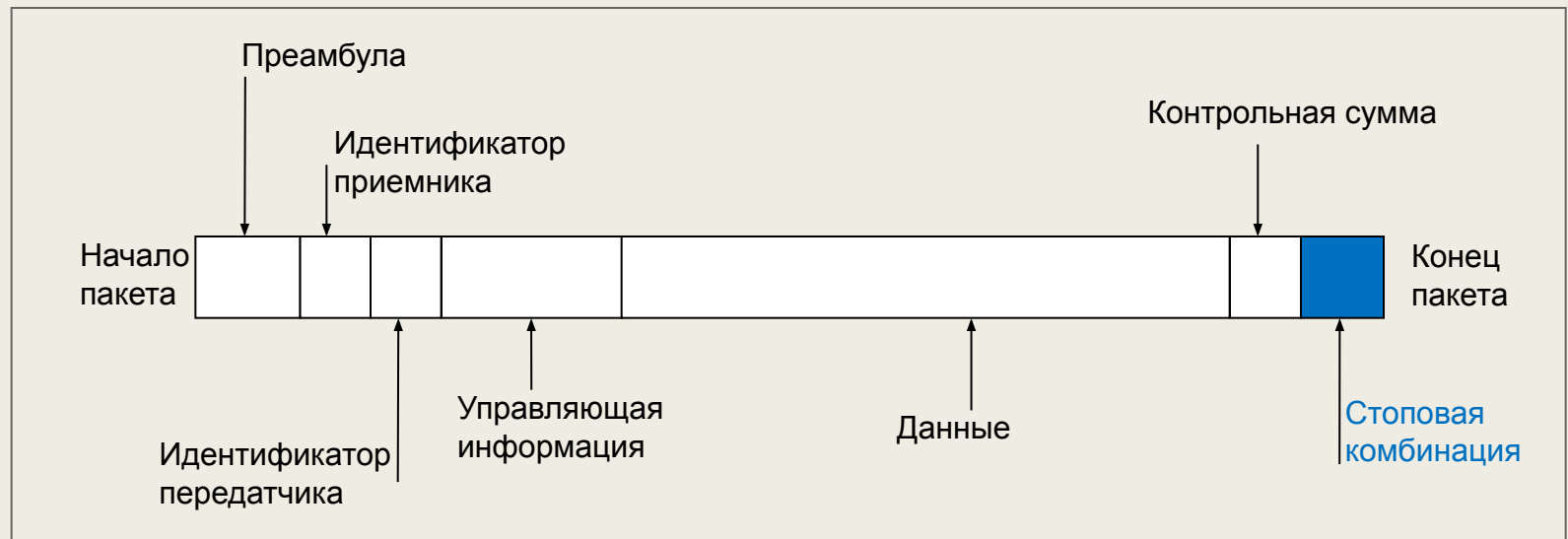
# Структура пакета. Контрольная сумма



**Контрольная сумма пакета** – это числовой код, формируемый передатчиком по определенным правилам и содержащий в свернутом виде информацию обо всем пакете.

Приемник, повторяя вычисления, сделанные передатчиком, с принятым пакетом, сравнивает их результат с контрольной суммой и делает вывод о правильности или ошибочности передачи пакета. **Если пакет ошибочен**, то приемник или **исправляет ошибки**, или **запрашивает повторную передачу пакета**.

# Структура пакета. Преамбула



**Стоповая комбинация** служит для информирования аппаратуры принимающего абонента об окончании пакета, обеспечивает выход аппаратуры приемника из состояния приема. Это поле может отсутствовать, если используется самосинхронизирующийся код, позволяющий определять момент окончания передачи пакета.

# Структура пакета

---

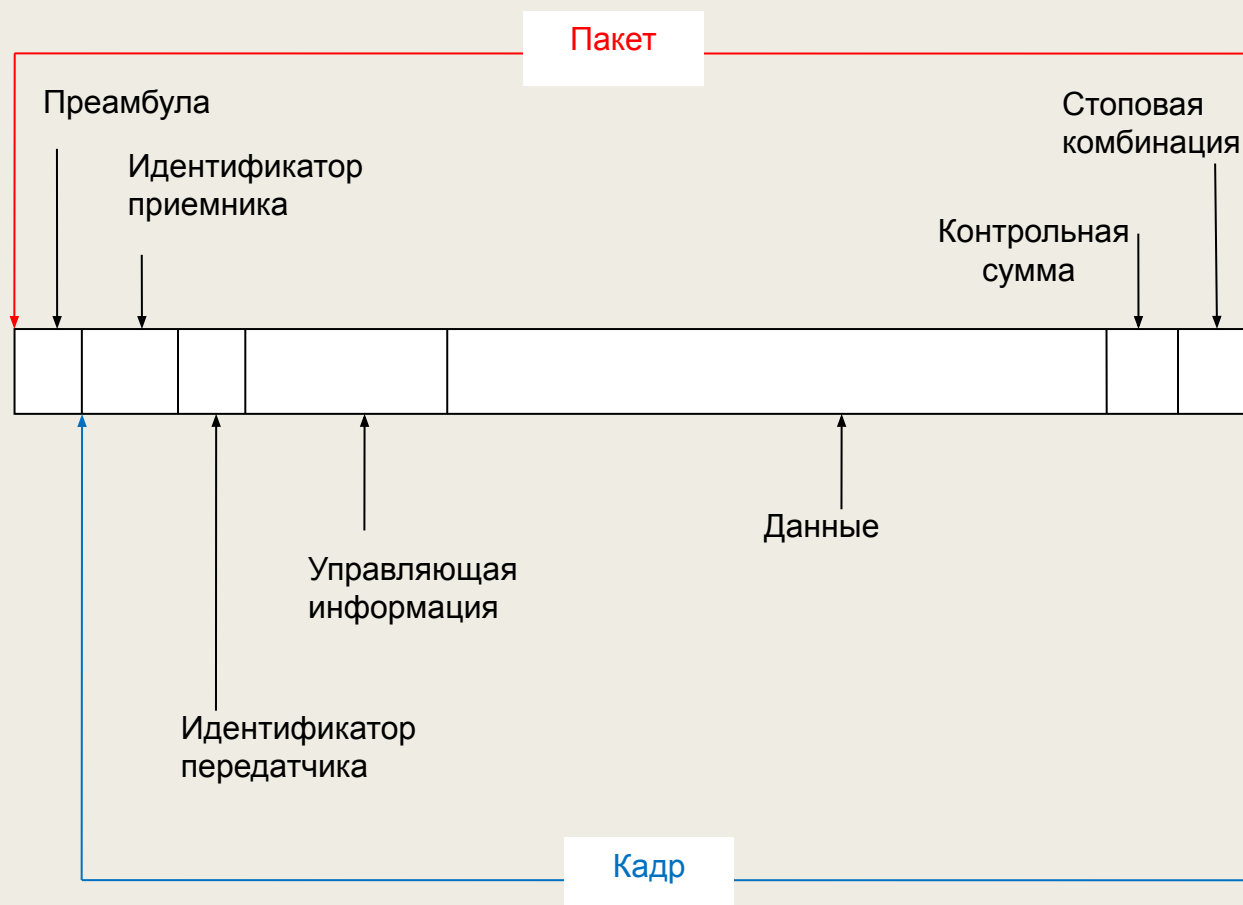
Нередко в структуре пакета выделяют всего **три поля**:

**Начальное управляющее поле** пакета (или **заголовок пакета**), то есть поле, включающее в себя стартовую комбинацию, сетевые адреса приемника и передатчика, а также служебную информацию.

**Поле данных** пакета.

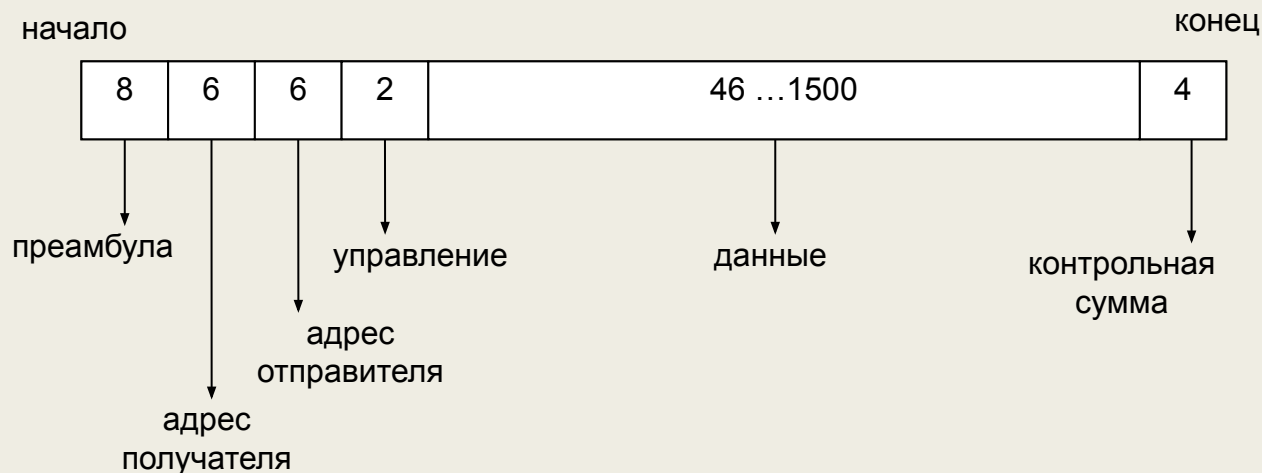
**Конечное управляющее поле** пакета (**заключение, трейлер**), куда входят контрольная сумма и стоповая комбинация, а также, возможно, служебная информация.

# Пакет или кадр?

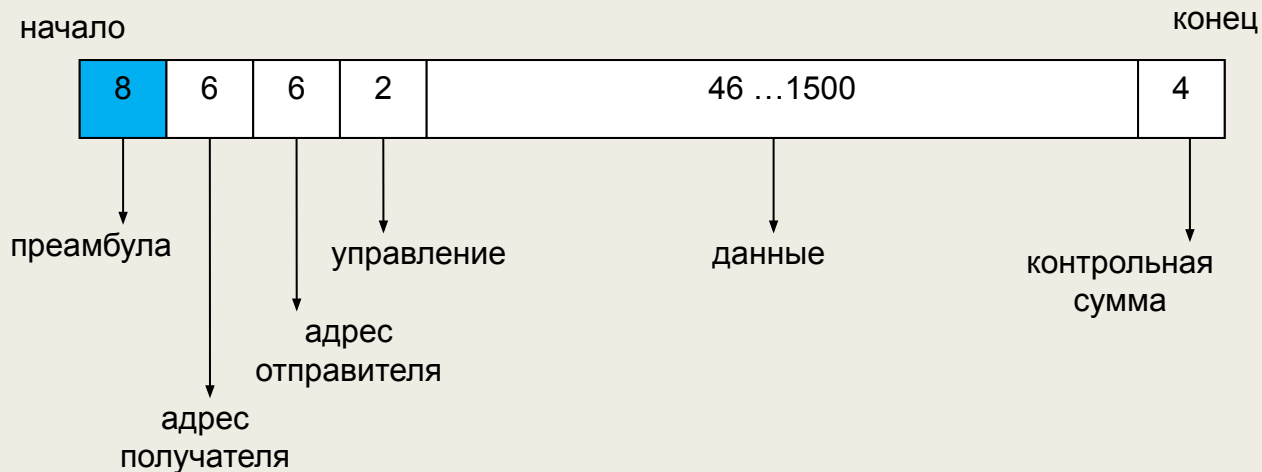


# Структура пакета Ethernet

Доступ к сети Ethernet осуществляется по методу CSMA/CD. В сети используются пакеты переменной длины с представленной на рисунке структурой (с указанием размеров полей). Предусмотрена **индивидуальная, групповая и широковещательная адресация**.

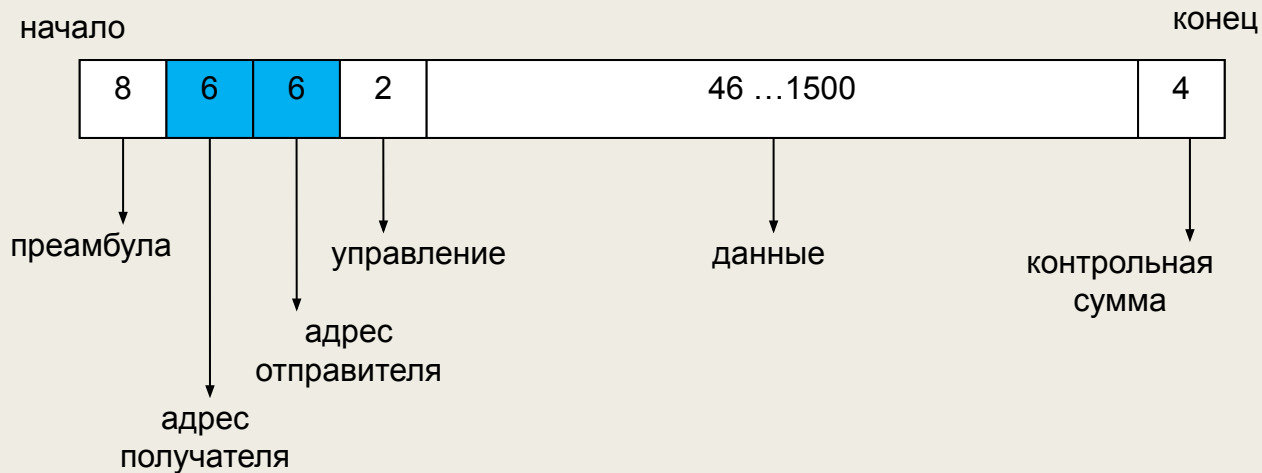


# Структура пакета Ethernet



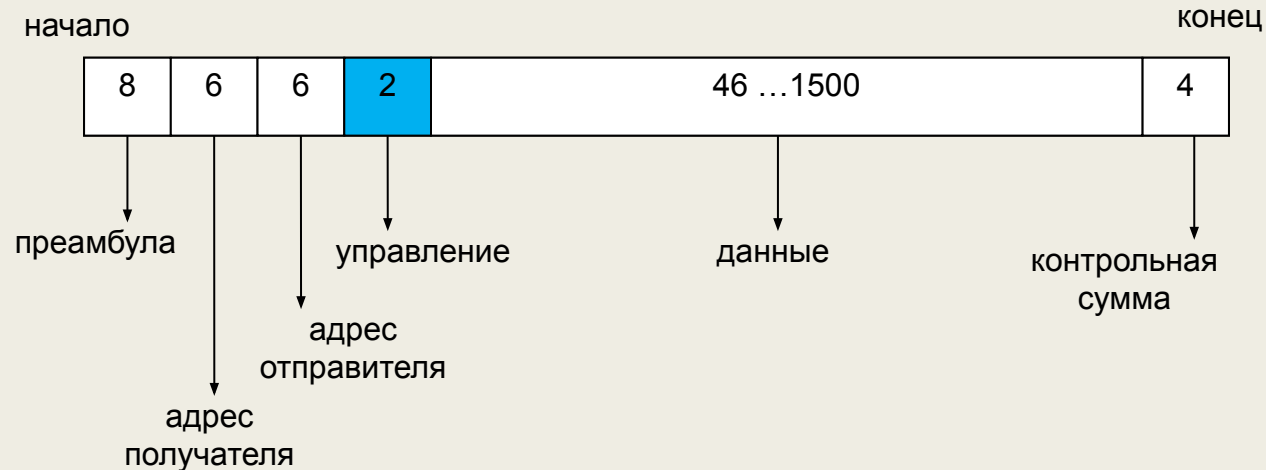
**Преамбула** состоит из 8 байтов, первые семь из которых представляют собой код **10101010**, а последний восьмой – код **10101011**. В стандарте IEEE 802.3 этот последний байт называется **признаком начала кадра** (SFD – Start of Frame Delimiter) и образует отдельное поле пакета.

# Структура пакета Ethernet



**Адрес получателя (приемника) и адрес отправителя (передатчика) включают по 6 байтов. Эти адресные поля обрабатываются аппаратурой абонентов.**

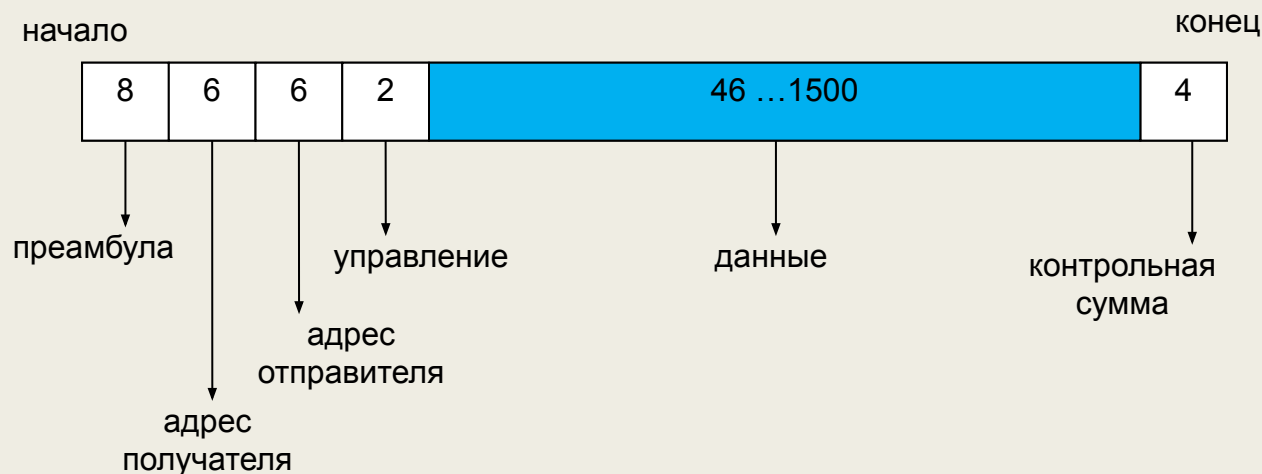
# Структура пакета Ethernet



**Поле управления** (L/T – Length/Type) содержит информацию о длине поля данных. Оно может также определять тип используемого протокола. Поле управления обрабатывается программно.

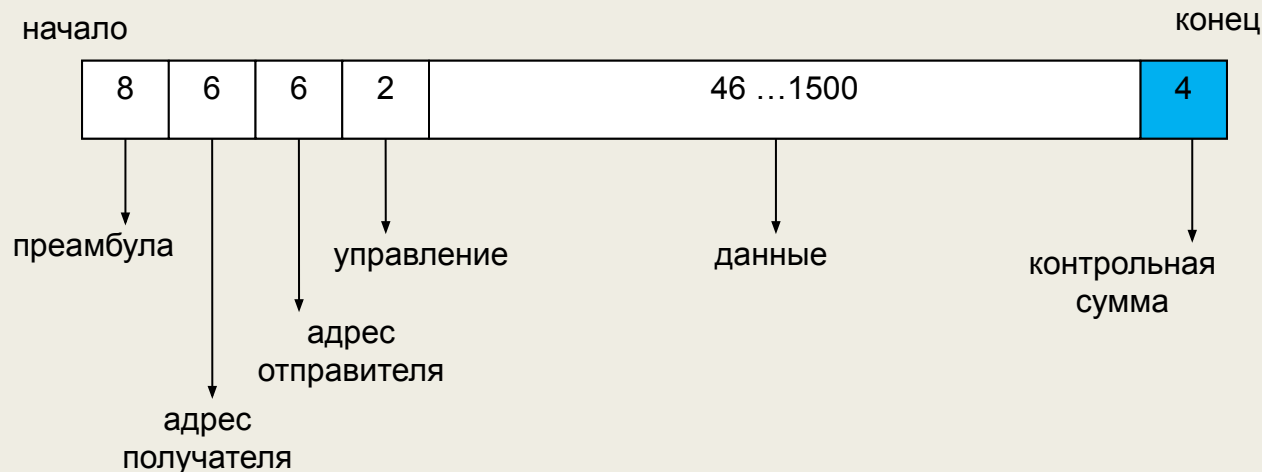


# Структура пакета Ethernet



**Поле данных** должно включать в себя от 46 до 1500 байт данных. Если пакет должен содержать менее 46 байт данных, то поле данных дополняется *байтами заполнения*. Согласно стандарту IEEE 802.3, в структуре пакета выделяется специальное поле заполнения (Pad Data – незначащие данные), которое может иметь нулевую длину, когда данных достаточно (больше 46 байт).

# Структура пакета Ethernet



**Поле контрольной суммы (FCS – Frame Check Sequence) содержит 32-разрядную циклическую контрольную сумму пакета (Cyclic Redundancy Check, CRC) и служит для проверки правильности передачи пакета (обнаружение и исправление ошибок).**

# Структура пакета Ethernet.

## Выводы

---

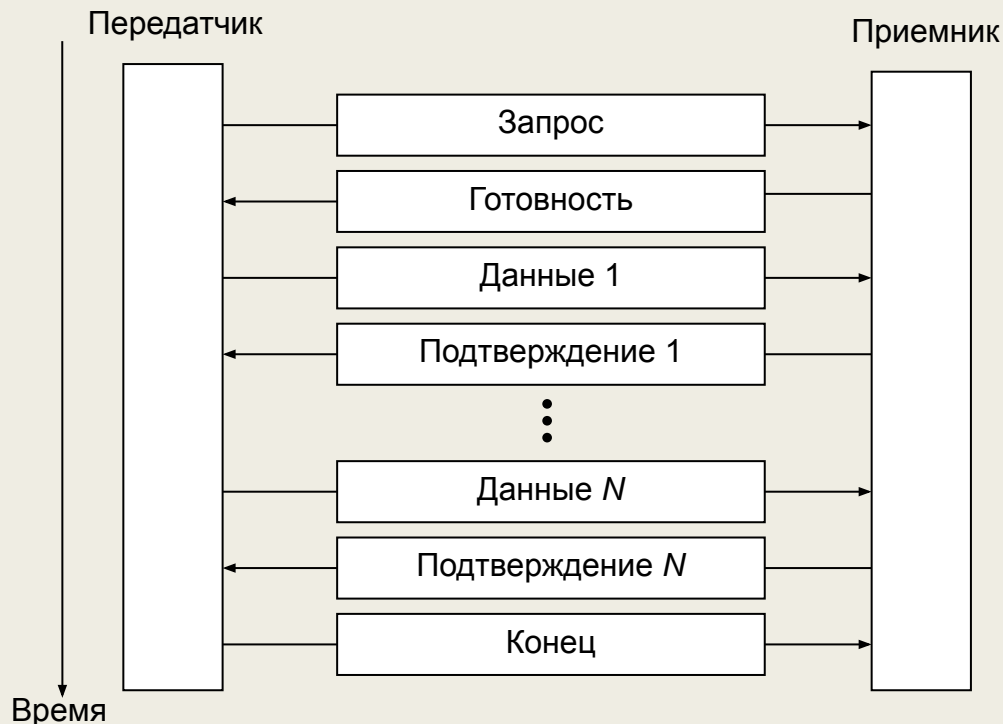
1. **Минимальная длина кадра** (пакета без преамбулы) составляет 64 байта (512 бит). Именно эта величина определяет максимально допустимую двойную задержку распространения сигнала по сети в 512 битных интервалов.

2. Стандарт предполагает, что **преамбула может уменьшаться** при прохождении пакета через различные сетевые устройства, поэтому она не учитывается.

3. **Максимальная длина кадра** равна 1518 байтам.

# Процесс обмена информацией

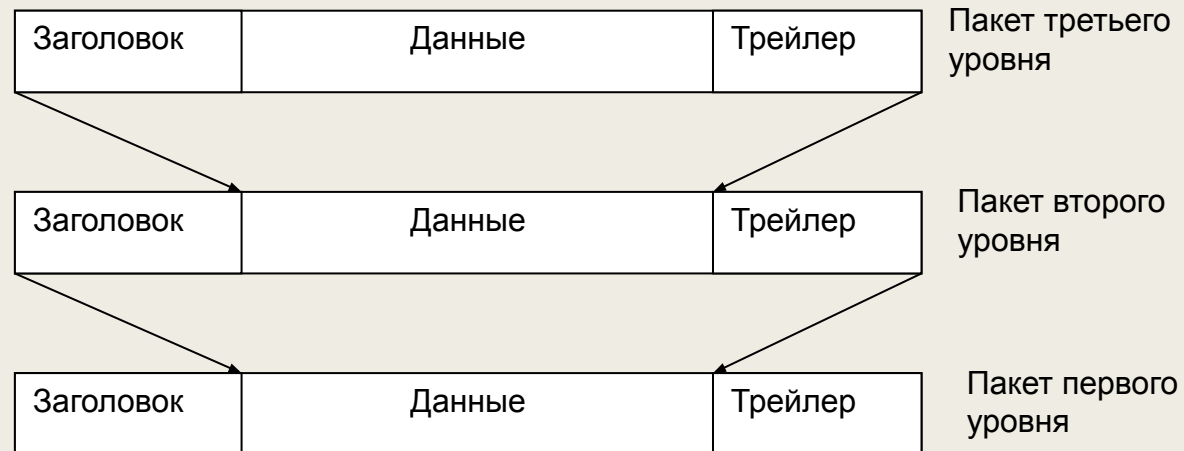
В процессе сеанса обмена информацией по сети между передающим и принимающим абонентами происходит обмен информационными и управляющими пакетами по установленным правилам, называемым *протоколом обмена*. Это позволяет обеспечить надежную передачу информации при любой интенсивности обмена по сети.



# Процесс обмена информацией

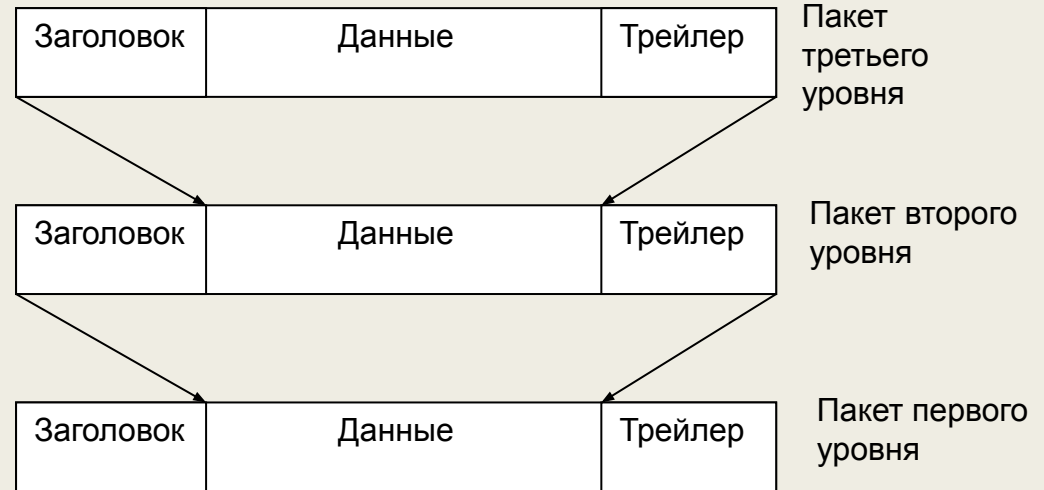
При реальном обмене по сети применяются **многоуровневые протоколы**, каждый из уровней которых предполагает свою структуру пакета (адресацию, управляющую информацию, формат данных и т.д.).

*Все пакеты более высоких уровней последовательно вкладываются в передаваемый пакет, точнее, в поле данных передаваемого пакета. Этот процесс последовательной упаковки данных для передачи называется также **инкапсуляцией пакетов**.*



# Процесс обмена информацией

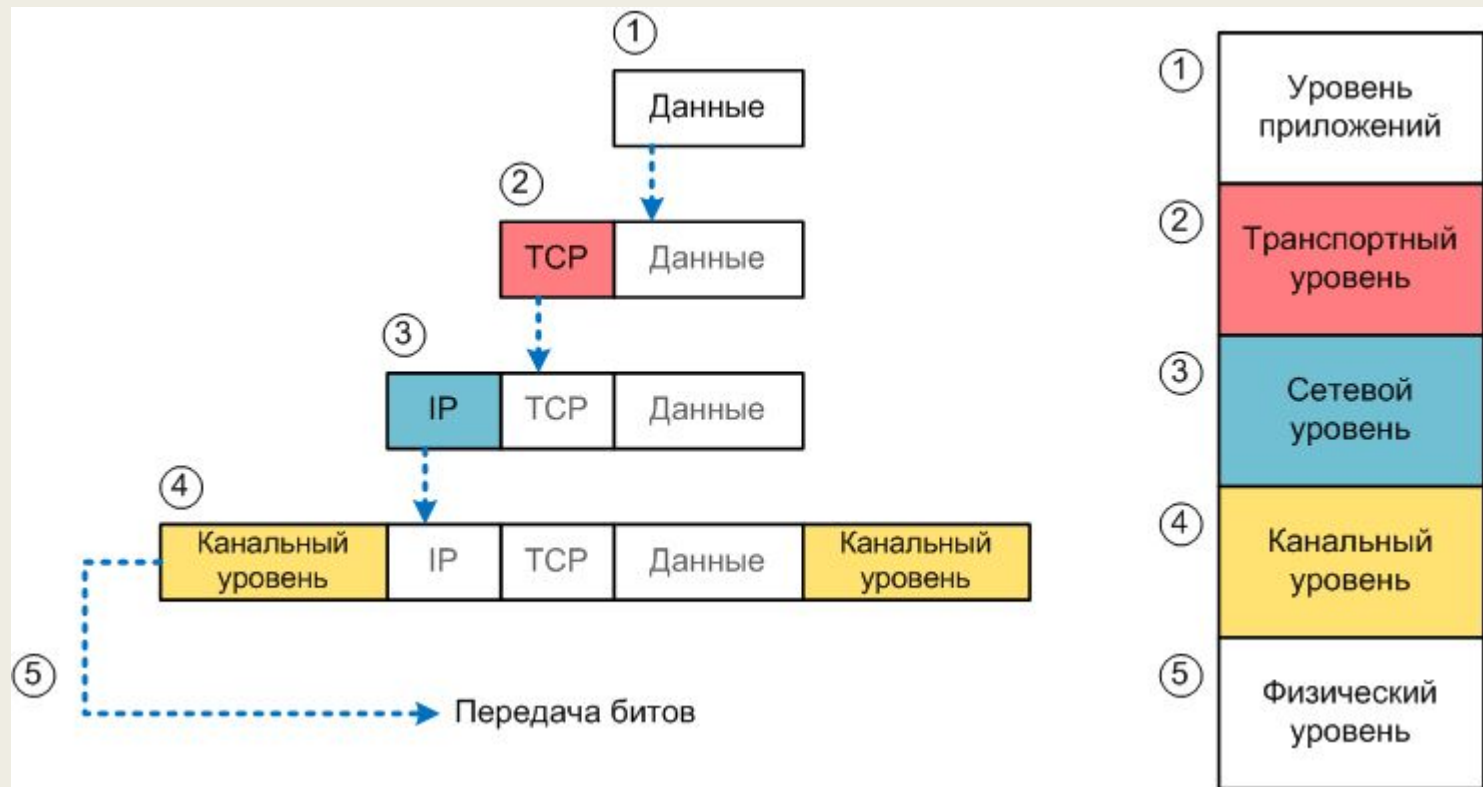
Каждый следующий вкладываемый пакет может содержать собственную служебную информацию, располагающуюся как до данных (заголовок), так и после них (трейлер), причем ее назначение может быть различным.



Доля вспомогательной информации в пакетах при этом возрастает с каждым следующим уровнем, что снижает эффективную скорость передачи данных. Для увеличения этой скорости предпочтительнее, чтобы протоколы обмена были проще, и уровней этих протоколов было меньше.

Обратный процесс последовательной распаковки данных приемником называется декапсуляцией пакетов.

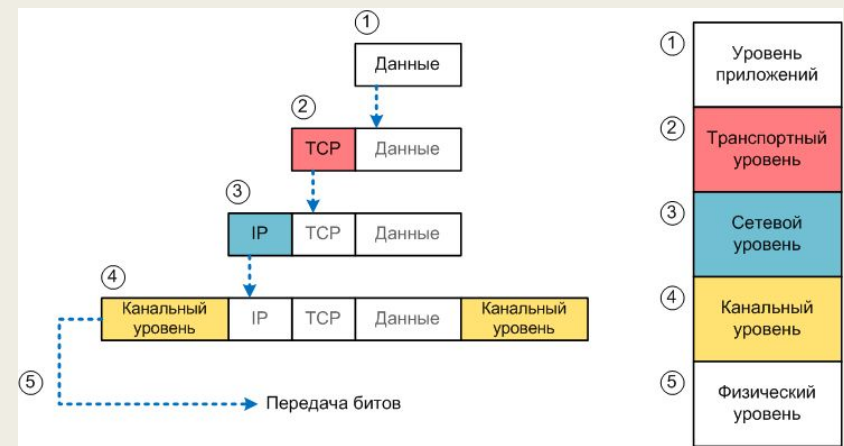
# Процесс обмена информацией. Инкапсуляция пакетов в стеке TCP/IP



# Инкапсуляция пакетов в стеке TCP/IP.

## Шаги

1. **Создание и инкапсуляция данных приложения с любыми необходимыми заголовками уровня приложений.** Например, HTTP-сообщение ОК может быть возвращено в заголовке HTTP, за которым следует часть содержимого веб-страницы.



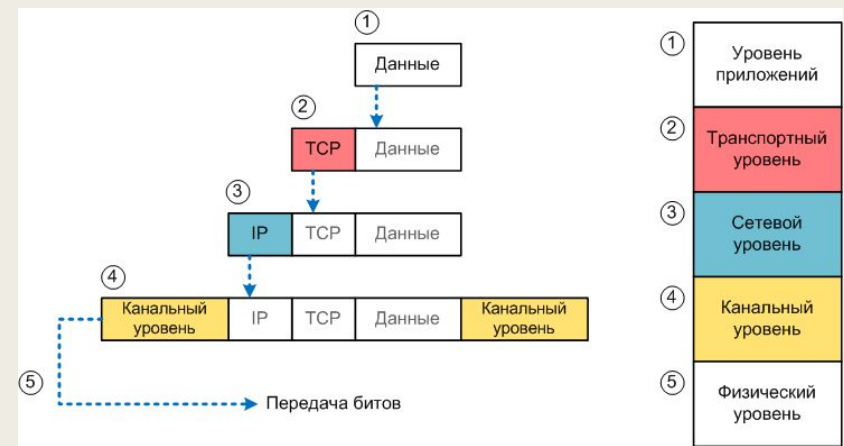
2. **Инкапсуляция данных, предоставленных уровнем приложений, внутри заголовка транспортного уровня.** Для приложений конечного пользователя обычно используется заголовок TCP или UDP.



# Инкапсуляция пакетов в стеке TCP/IP.

## Шаги

3. **Инкапсуляция данных, предоставленных транспортным уровнем, внутри заголовка сетевого уровня (IP).** IP определяет IP-адреса, которые однозначно идентифицируют каждый компьютер.



4. **Инкапсуляция данных, предоставленных сетевым уровнем, в заголовки и трейлер (концевик) канального уровня.** Этот уровень использует как заголовки, так и трейлер (концевик).

5. **Передача битов.** Физический уровень кодирует сигнал в среде для передачи кадра.