

Spanning Tree Protocols (STP)

IEEE 802.1D

Протокол распределенного связующего дерева

Источник: Кларк К, Гамильтон К. Принципы коммутации в локальных сетях Cisco. : Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 976 с. ISBN 5-8459-0464-1 (рус)

СОДЕРЖАНИЕ

1. Проблемы петлевой топологии
2. Решение проблемы петлевой топологии → Spanning Tree Protocol (STP)
3. Три этапа начальной сходимости
4. Форматы BPDU кадров
5. Обработка BPDU
6. Состояния и таймеры STP
7. RSTP, Rapid-PVST+

1. Проблемы петлевой топологии

Структура Ethernet кадра и MAC/SAP –адреса

Проблема петлевой топологии

Бесконечное движение по кругу

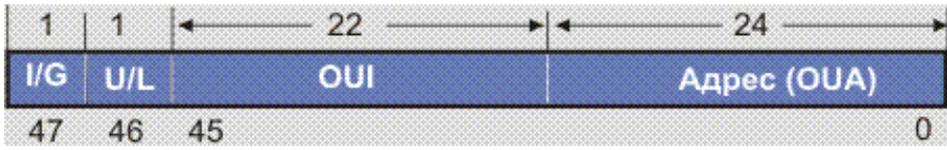
Вспомним структуру Ethernet кадра и MAC –адреса

Формат кадра 802.3/LLC [с 802.2 (SAP)]



- Пreamble (preamble) для кадровой синхронизации
- DA (Destination Address)..... MAC адрес приёмника (48 бит)
- SA (Source Address) MAC адрес источника (48 бит)
- Длина Длина кадра в байтах (16 бит),
(без поля FCS (46-1500))
- Данные полезная нагрузка
- FCS (Frame Check Sequence) Проверочная последовательность кадра (32 бита)

Формат и типы MAC-адресов



• Типы MAC-адресов:

- unicast (I/G = "0") – индивидуальный
- multicast (I/G = "1") – групповой
- broadcast (все биты = "1") – широковещательный

Петли порождают параллельные пути

- **В петлевой топологии существуют параллельные пути движения кадров между двумя сегментами LAN¹, которые являются причиной:**
 - бесконечного движения по кругу кадров с неизвестным адресом назначения (DA = неизвестный MAC, т.е. MAC-адрес отсутствует в таблицах коммутации моста)
 - бесконечного движения по кругу широковещательных кадров (DA=broadcast)
 - бесконечного движения по кругу unicast-кадров (DA=Unicast) в фазе наводнения (flooding)
 - переполнению буферных ресурсов
- **Параллельные пути в сложной топологии приводят к:**
 - переполнению всех буферных ресурсов и стагнации LANs
 - широковещательному шторму (Broadcast Storm)
- **Избежать этих эффектов можно посредством**
 - Протокол связующего дерева Spanning Tree Protocol (STP)

¹сегмент LAN – LAN без мостов

3.1 Bridged Local Area Network → (Мостовая локальная сеть)

«A concatenation of individual **IEEE 802 LANs** interconnected by MAC (Media Access Control) Bridges → Объединение отдельных локальных сетей IEEE 802, соединенных между собой мостами посредством управления доступом к среде

ПРИМЕЧАНИЕ - Если явно не указано иное, использование слова **сеть** (network) в 802.1D-2004 относится к мостовой локальной сети (**Bridged Local Area Network**). Термин «Мостовая локальная сеть» не сокращается иначе. *Термин «Локальная сеть» и аббревиатура «LAN» используются исключительно для обозначения отдельной ЛВС, определенной технологией MAC, без включения мостов»*

Вспомним, Мост для технологии MAC 802.3 (CSMA/CD, Ethernet) нужен был для сегментирования сети (деления на части) с целью уменьшения домена коллизии. Более широкое понятие сегмента сети (individual **IEEE 802 LANs**) пояснено в приводимом ниже разделе 3.4 IEEE 802.1D

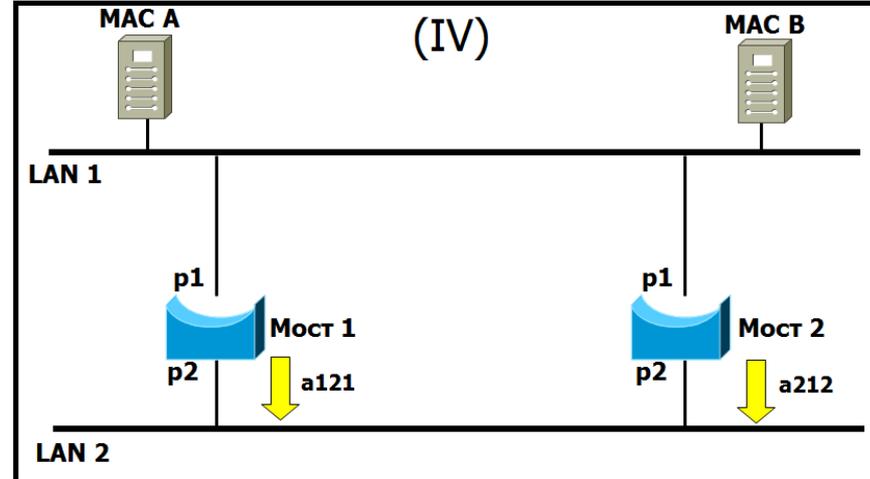
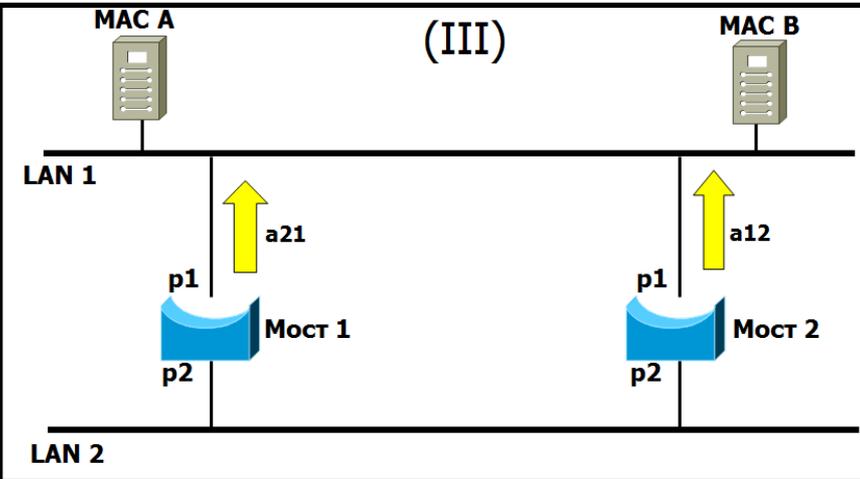
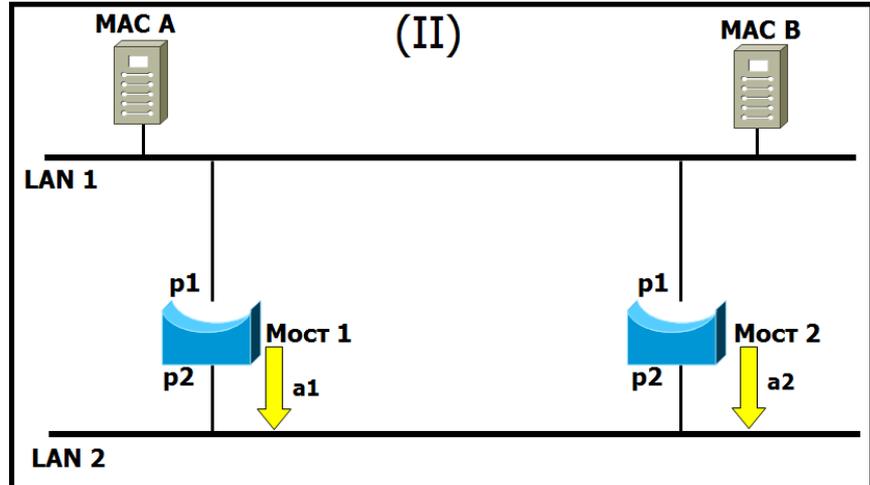
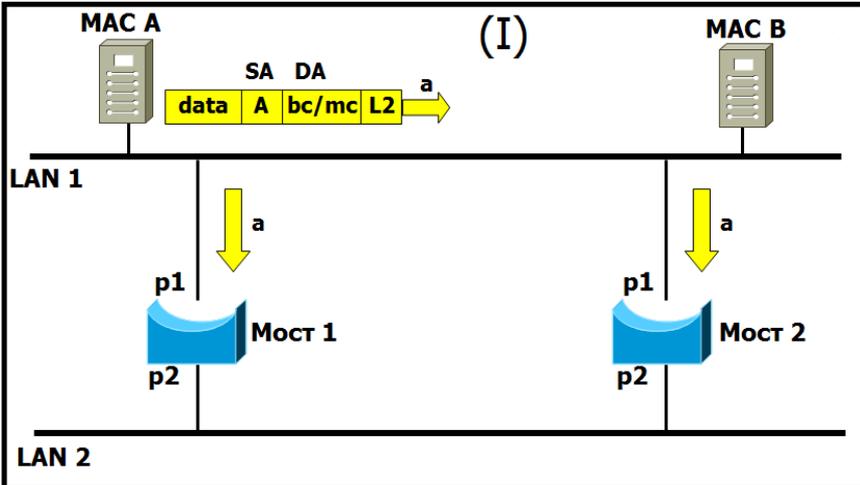
3.4 IEEE 802 Local Area Network (LAN) (Локальная сеть IEEE 802)

IEEE 802 LANs (also referred to in the text simply as LANs) are LAN technologies that provide a MAC Service equivalent to the MAC Service defined in ISO/IEC 15802-1. IEEE 802 LANs include IEEE Std 802.3 (CSMA/CD), IEEE Std 802.5 (Token Ring), IEEE Std 802.11 (Wireless), and ISO 9314-2 (FDDI) LANs. →

Локальные сети IEEE 802 (также называемые в тексте просто «LAN») - это технологии LAN, которые предоставляют сервис управления доступом к среде (MAC - Media Access Control), эквивалентный сервису MAC, определенному в ISO/IEC 15802-1. Локальные сети включают в себя локальные сети IEEE Std 802.3 (CSMA/CD), IEEE Std 802.5 (Token Ring), IEEE Std 802.11 (беспроводные) и ISO 9314-2 (FDDI).

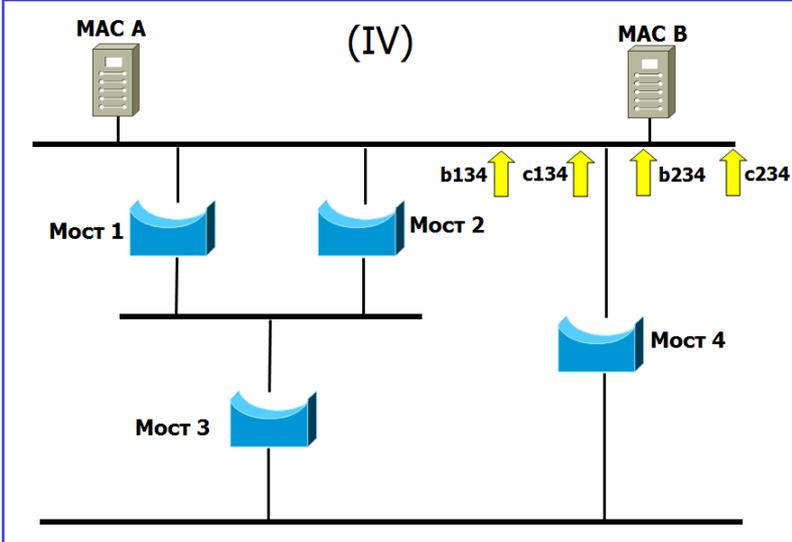
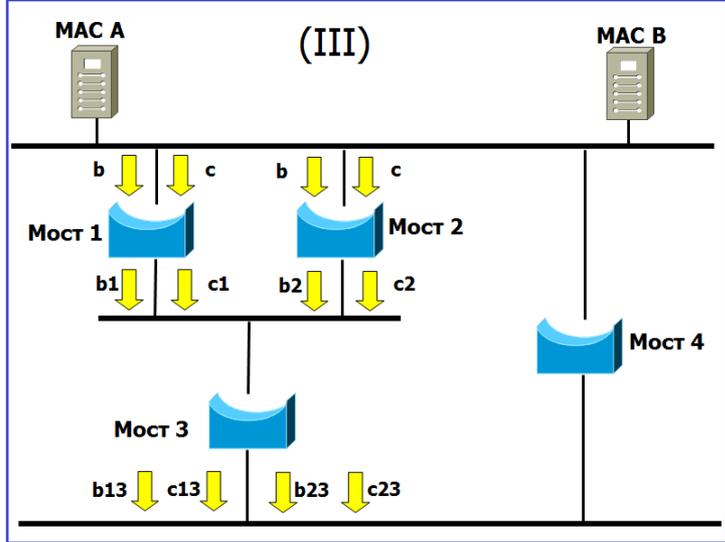
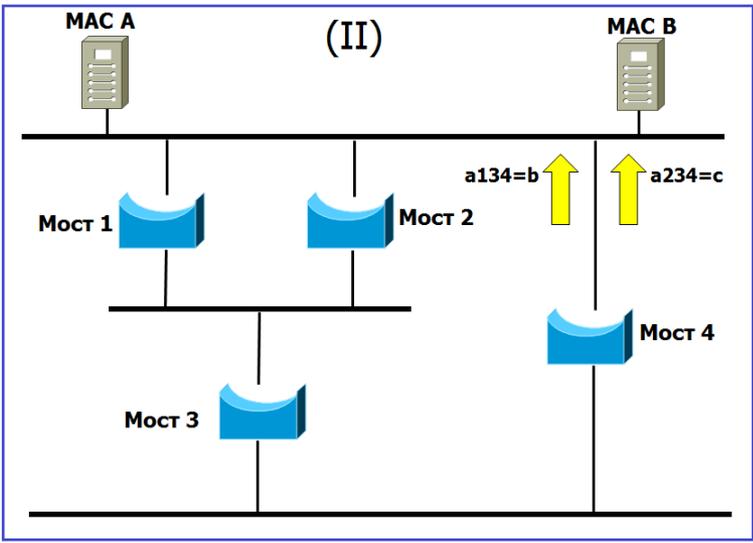
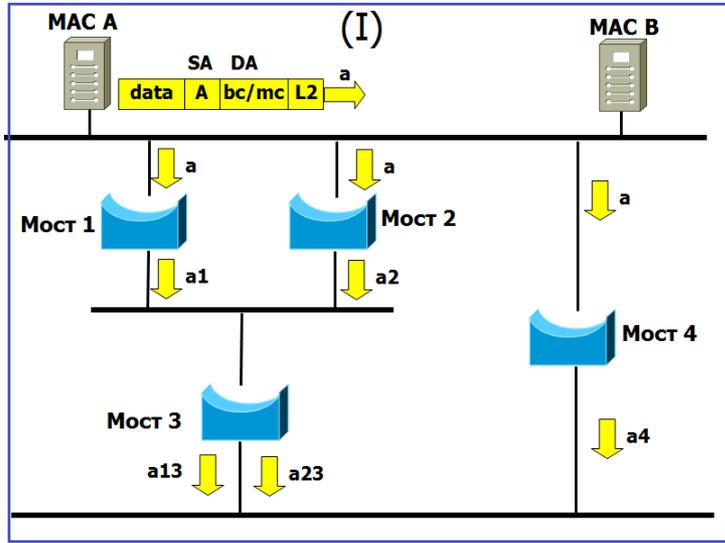
Бесконечное движение по кругу

В петлевой топологии broadcast (bc) / multicast (mc) и кадры с неизвестным мостам MAC-адресом назначения (поле DA в кадре) будут циркулировать вечно

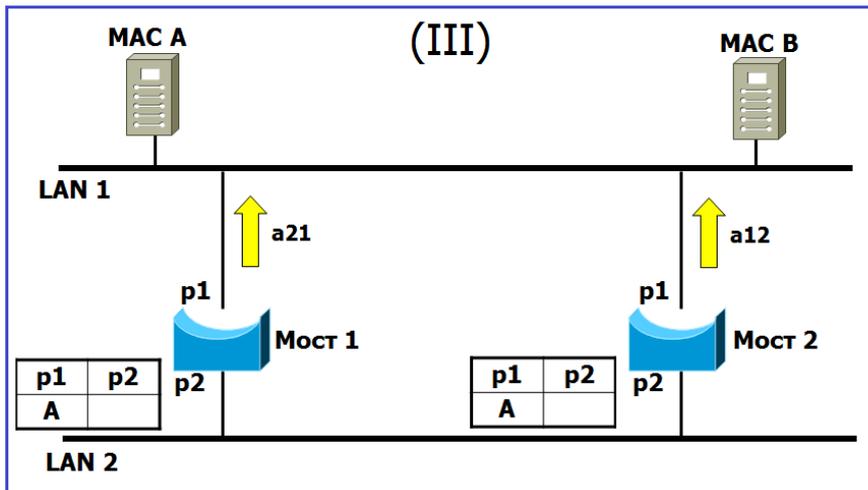
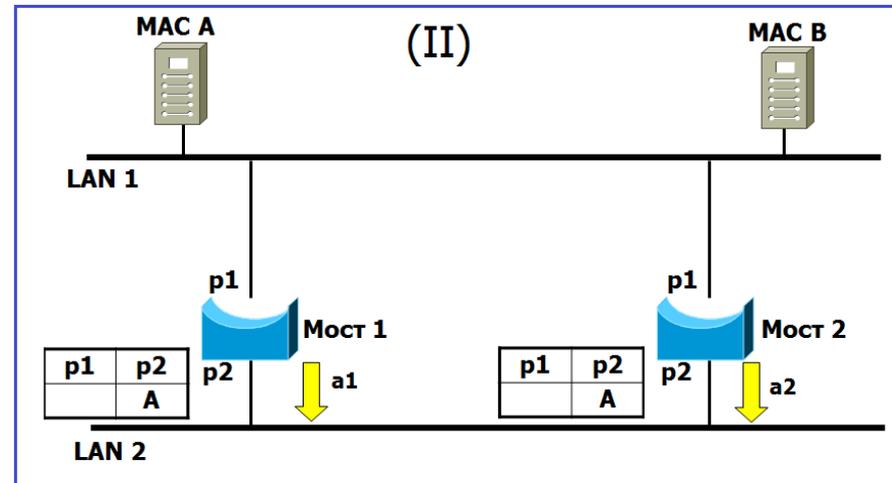
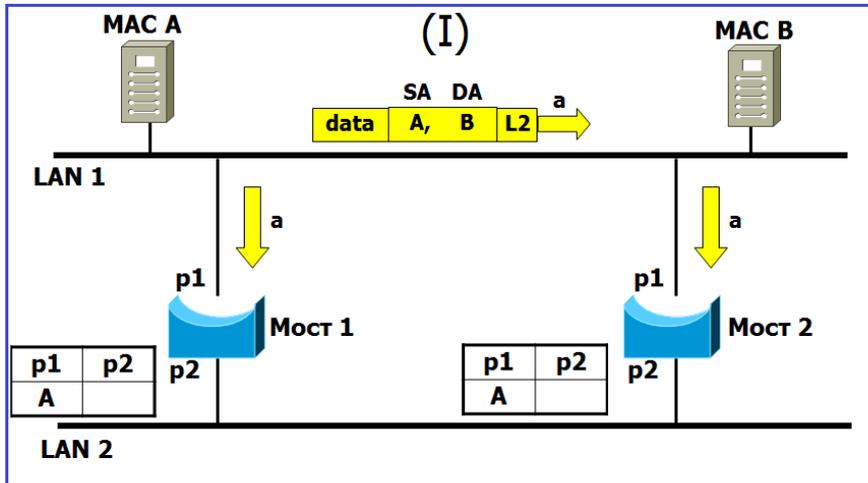


Широковещательный шторм

В топологии «петля в петле» broadcast (bc) / multicast (mc) кадры порождают широковещательный шторм, что приводит к коллапсу сети



Перезапись таблицы (неизвестное направление)



В петлевой топологии:

- кадры с неизвестным мостом MAC-адресом назначения (destination, поле DA в кадре) будут циркулировать вечно
- Процесс самообучения мостов по source MAC адресам (поле SA в кадре) приведет к непрерывным перезаписям в таблице коммутации сведений об отправителе кадра (компьютере A на рисунках)

Пример этой ситуации: компьютер B выключился, таблицы коммутации мостов через 5 минут удалили устаревшую запись о компьютере B

Почему возникают физические петли ?

● Топологические петли возникают по причине

- Намеренная попытка администратора повысить надежность сети за счет избыточных соединений
- Ошибка администратора
- Злой умысел атакующего

● Петли коммутации (L2 OSI RM) доставляют намного больше проблем, чем петли маршрутизации (L3 OSI RM)

- IP-пакет уничтожается маршрутизатором
 - ✓ При достижении времени жизни пакета TTL = 0
- Ethernet-кадр циркулирует в сети вечно, или пока:
 - ✓ Принудительно не будет отключено питание моста
 - ✓ Самостоятельно, в результате неисправности канала

2. Решение проблемы петлевой топологии → Протокол STP

STP сервис/функции

- создание из физической петлевой топологии логической/активной без петельной топологии
- путем “блокирования” части портов с целью построения дерева (корень, ствол, ветки, листья)

Автор распределенного алгоритма построения связующего дерева

- Радия Перлман (Radia Perlman), корпорация DEC, 1985 год

Spanning Tree Protocols (STP) (IEEE 802.1D)

● STP устраняет проблемы петлевой топологии сети

- Бесконечное движение по кругу кадров с неизвестным MAC адресом назначения и/или широковещательный шторм

● STP сервис/функции

- создание из физической петлевой топологии логической/активной топологии без петель

● STP механизм

- STP запускается/инициализируется в мостах администратором сети
- Мосты обмениваются BPDU-кадрами (Bridge Protocol Data Unit - протокольный блок данных моста), решая задачу построения логической беспетлевой топологии путем присвоения некоторым портам статуса "блокированный порт"
- BPDU-кадры передаются другим мостам в ethernet кадре с групповым адресом назначения (DA = 01-80-C2-00-00-00), закрепленным за STP

● STP основной недостаток

- Избыточные линии или избыточные сетевые компоненты, вводимый администратором для повышения надежности, не могут использоваться для балансирования загрузки

a) Конфигурирует активную топологию Bridge LAN произвольной топологии в единственное распределенное связующее дерево

- при наличии больше одного маршрута данных между любыми двумя конечными станциями устраняются циклы данных

b) Предусматривает автоматическую отказоустойчивость

- посредством реконфигурации топологии распределенного связующего дерева в результате неисправности Моста или неисправности в информационном канале, в пределах границ LAN без формирования циклов текущих данных

c) Полная активная топология стабилизируется в LAN любой размерности

- с высокой вероятностью и за известный ограниченный интервал времени
- в течении этого времени могут быть недоступны связи между любой парой конечных станций

d) Активная топология предсказуема и воспроизводима

- Активная топология может быть выбрана посредством управления параметрами алгоритма («Управление конфигурацией»), который на основе анализа потоков информации достигает цели

e) Конечные станции не сознают их принадлежности к единственной LAN при управлении доступом к среде

f) Используемая STP-протоколом полоса пропускания каналов при установлении и поддержании распределенного связующего дерева на любой конкретной LAN

- Использует малый процент от полной располагаемой полосы пропускания
- Не зависит от полного трафика, поддержанного Соединенной локальной сетью независимо от общего количества Мостов или локальных сетей

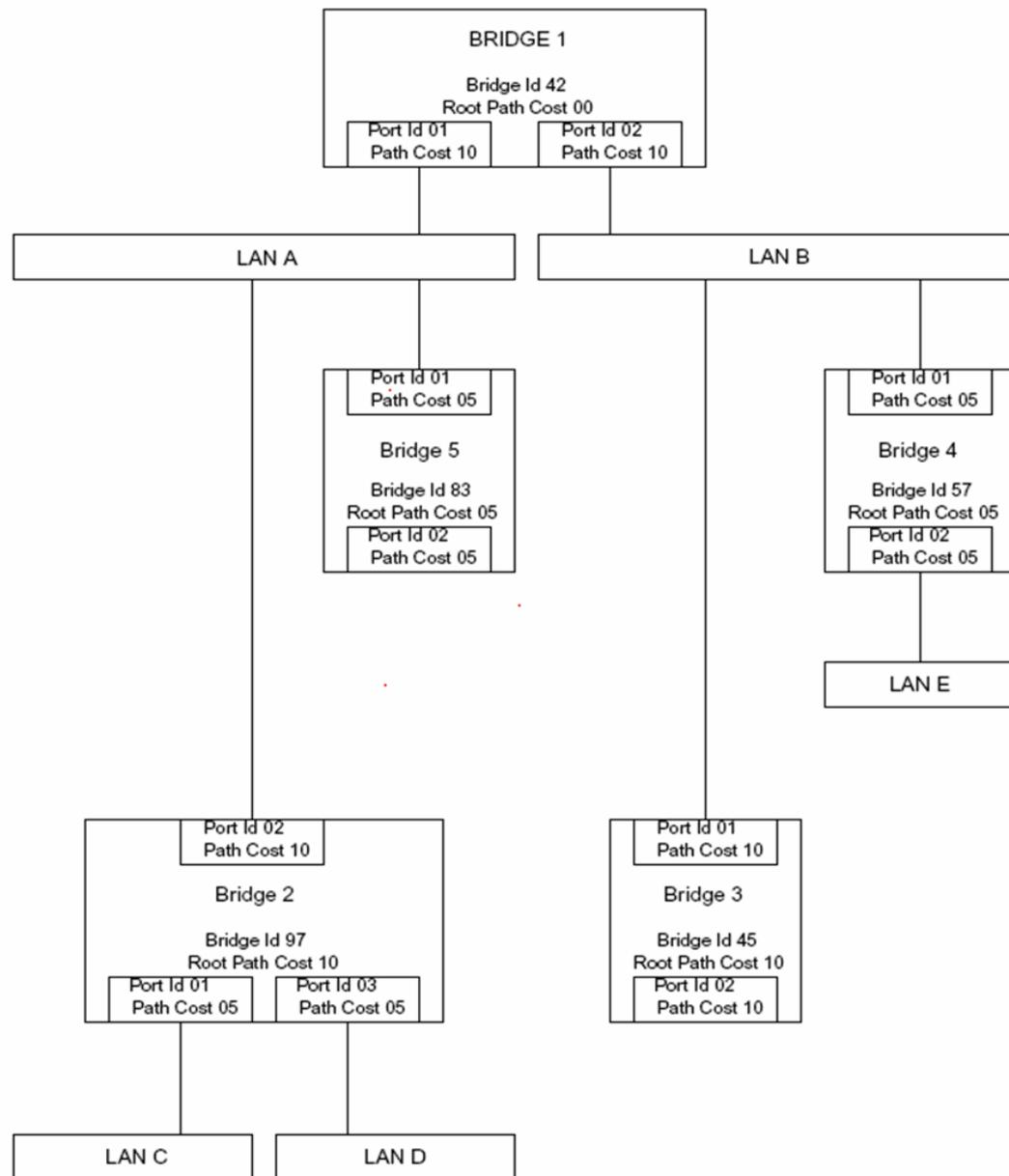
Параметры (сущности) STP

- **Алгоритм STP** разработан в 1985 году **Радией Перлман** (Radia Perlman) в корпорации DEC.
- **В вычислениях STP используются** следующие основные параметры (сущности):
 - **Bridge Id (BID)** - идентификатор моста
 - **Port Id (PID)** – идентификатор порта
 - **Path Cost** - СТОИМОСТЬ ПУТИ (стоимость порта)

На рисунке справа проиллюстрированы сущности STP из стандарта IEEE 802.1D, где

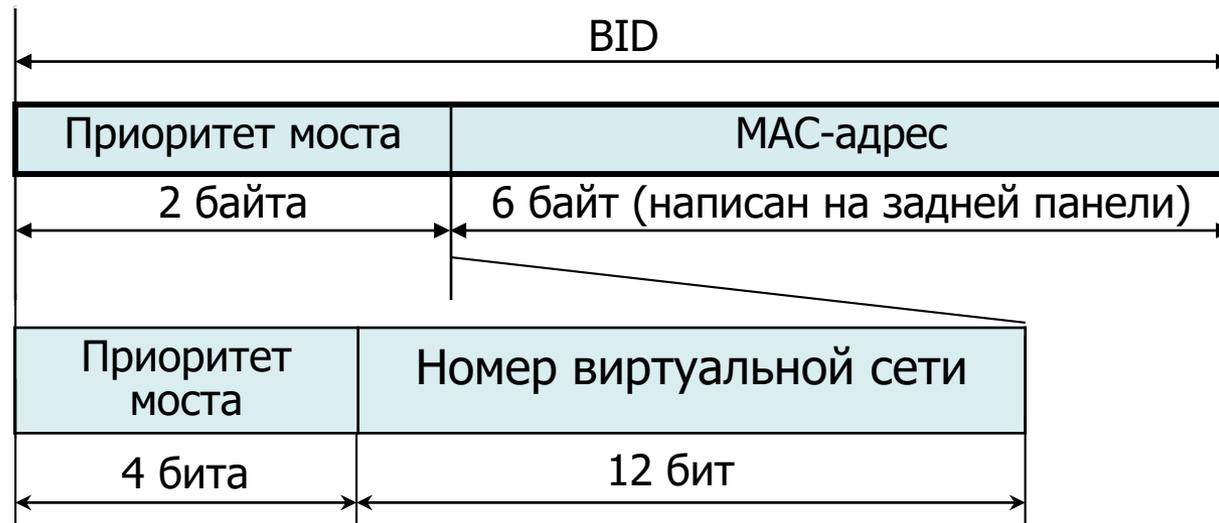
LAN (локальные сети) – это технологии построения локальных сетей IEEE 802 включают в себя локальные сети IEEE Std 802.3 (CSMA / CD), IEEE Std 802.5 (Token Ring), IEEE Std 802.11 (беспроводные) и ISO 9314-2 (FDDI).

Аббревиатура «LAN» в IEEE 802.1D используются исключительно для обозначения отдельной LAN, определенной технологией MAC (Media Access Control), без включения мостов.



Идентификатор моста (BID - Bridge ID)

- **BID это комбинация «Приоритета моста и MAC-адреса моста»**
- **Приоритет моста**, старшая часть BID, конфигурируется администратором



В редакции IEEE 802.1D–2004 в позицию «Приоритет моста» введен параметр «Номер виртуальной сети (VlanID)», уменьшив размер поля приоритета до с 16 до 4-х бит. Обеспечена возможность построения для каждой VLAN своего STP дерева – технология PVST (Per-VLAN spanning tree)

BID = «Приоритета моста + номер VLAN + MAC-адреса моста»

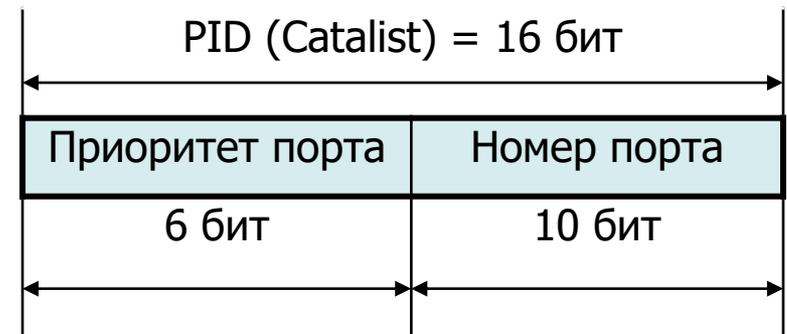
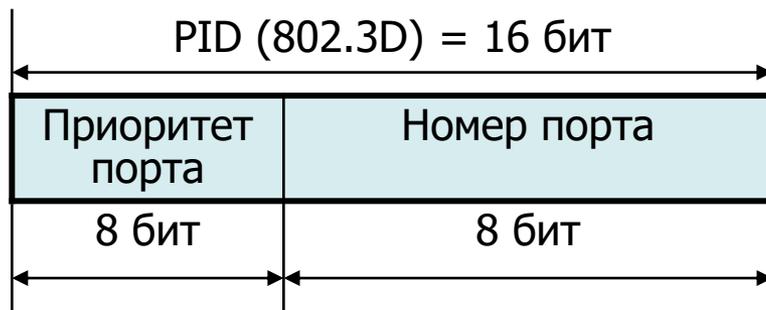
• **MAC адрес моста**

- К сведению, каждый порт моста имеет MAC-адрес
- MAC-адрес моста может назначаться администратором, или указан на задней панели устройства, или используется минимальный MAC-адрес из всех имеющихся на портах

• **Чем меньше BID моста, тем выше приоритет моста**

Идентификатор порта (PID - Port ID)

- **Port ID уникален среди всех других портов моста, является числом и состоит из двух полей:**
 - Приоритет порта (Port priority, старшая часть Port ID)
 - ✓ 1 байт в 802.3D (диапазон 0-255, по умолчанию – 128)
 - ✓ 6 бит в Catalyst (диапазон 0-63, по умолчанию 32)
 - ✓ Конфигурируется администратором
 - Номер порта (младшая часть Port ID), формируется аппаратурой
 - ✓ статически назначается для каждого порта: значение 1 присваивается порту 1/1, значение 2 - порту 1/2 и т.д
 - ✓ 1 байт в 802.3D (диапазон 0-255)
 - ✓ 10 бит в Catalyst (диапазон 0-1023, для коммутаторов с большой плотностью портов)
- **Чем меньше PID порта тем выше его приоритет относительно других портов моста**



Стоимость порта (Path Cost)

Табл. Значение стоимости пути (Path Cost) в IEEE версиях

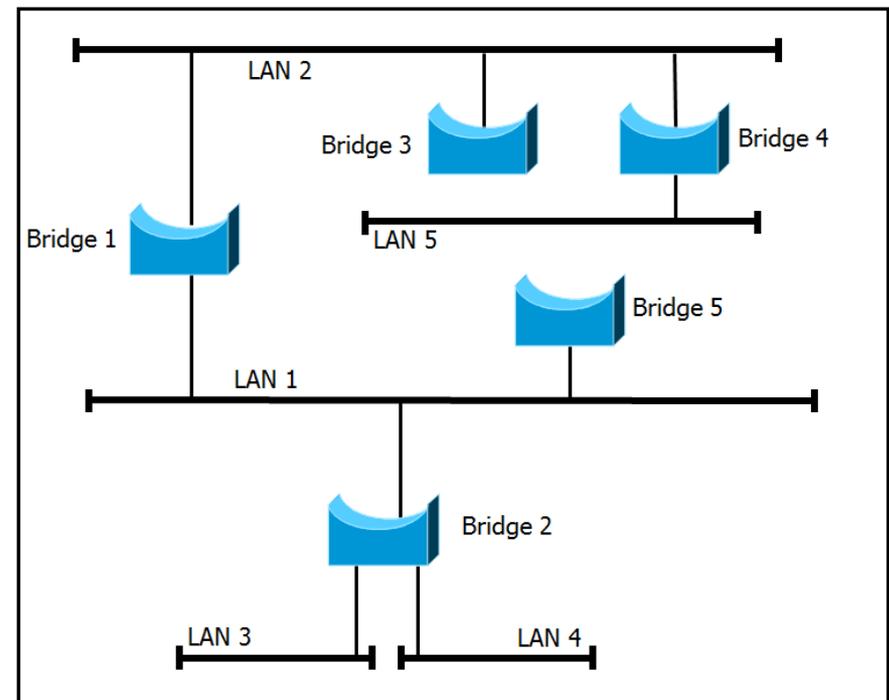
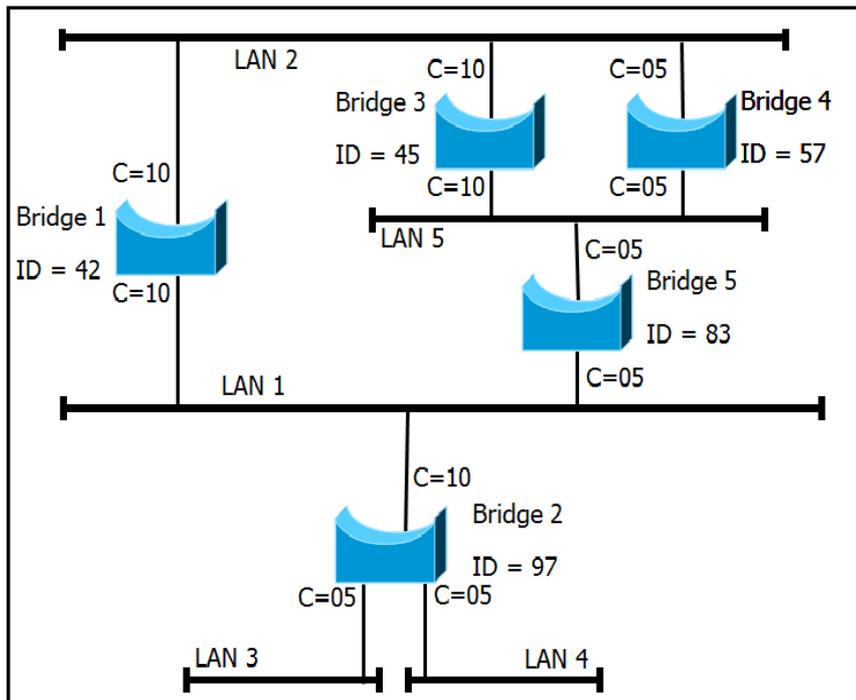
Скорость порта	Стоимость (802.1D-1998) [16 бит]	Стоимость (802.1D-2004) [802.1t - Long Path Cost, 32 бита]
4 Мбит/с	250	5 000 000
10 Мб/с	100	2 000 000
16 Мб/с	62	1 250 000
100 Мб/с	19	200 000
1 Гб/с	4	20 000
2 Гб/с	3	10 000
10 Гб/с	2	2 000
40 Гб/с	-	500
100 Гб/с	-	200
400 Гб/с	-	50
1 Тб/с	-	20
10 Тб/с	-	2

- Целое число, определяющее стоимость соединения (расстояния) между двумя смежными мостами
- Обратна пропорциональна скорости порта. Два варианта:
 - Patch Cost = нелинейная шкала стоимости
 - Long Patch Cost = 20Тбит/с : скорость интерфейса
- Стоимость порта по умолчанию определяется согласно таблице
- **ВАЖНО:** Администратором может быть установлено любое значение стоимости порта
 - Управление стоимостью следует использовать для реализации требований задания в лабораторной работе по STP

Суть работы STP

Протокол STP, инициированный в мостах, создает древовидную топологию, блокируя некоторые порты мостов согласно трем этапам начальной сходимости, реализуемым путем:

- **обмена** между мостами протокольными блоками данных **BPDU**, инкапсулированными в Ethernet кадр
- значение полей BPDU формируются STP протоколом на основе
 - Bridge Id (BID)** - идентификатор моста
 - Port Id (PID)** - идентификатор порта
 - Path Cost** - стоимость порта



Поля BPDU и роли Портов, используемые для построения дерева

BPDU - Bridge Protocol Data Unit (протокольный блок данных)

Существенные поля BPDU используются для:

- Root Identifier** → Какой BID примет статус корневого моста ???
- Root Path Cost** → Каково расстояние до корневого моста ???
- Sender Identifier** → BID моста отправителя BPDU-сообщения
- Port Identifier** → PID порта отправителя BPDU-сообщения

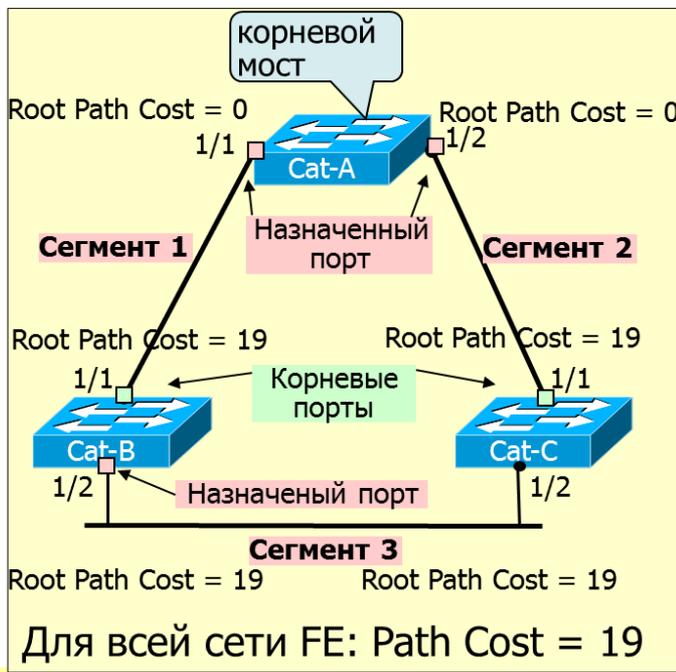
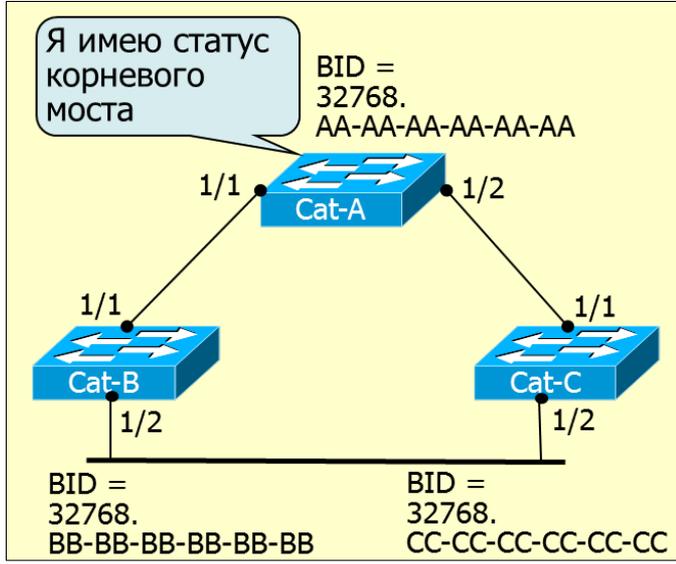
Количество	байт
Protocol Identifier	2
Protocol Version	1
BPDU Type	1
Flags	1
Root Identifier	8
Root Path Cost	4
Sender Identifier	8
Port Identifier	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2

В процессе обмена BPDU:

- 1) В поле **Root Identifier** будет формироваться **minBID** из всех существующих ("**корень дерева**")
- 2) После выбора корня в поле **Root Path Cost (RPC)** будет подсчитываться стоимость пути до корня для формирования по min RPC "**веток дерева**"
- 3) Ветка дерева должна "привиться" только к одной другой ветке путем **отрезания** ее от других веток для построения беспетельной топологии (Порт 1/2 Cat-C заблокируется)

Роли портов, определяют топологию сети

1. **Root (R)** – корневой
2. **Designated (D)** – назначенный
3. **Nondesignated (x)** – не назначенные (состояние порта блокированный)



3. Три этапа начальной сходимости STP

Три этапа начальной сходимости STP

Алгоритм принятия решения

Роли и Состояния портов

Примеры выбора корневого моста, корневых
и назначенных мостов

Состояния и таймеры STP

Три этапа начальной сходимости STP

● **Этап 1. Выбор корневого моста (Root Bridge)**

- Корневым выбирается мост с наименьшим Bridge ID (BID)

● **Этап 2. Выбор корневых портов (Root Ports)**

- Ближайшие к корневому мосту “порты некорневых мостов” становятся корневыми портами
 - ✓ Каждый некорневой мост должен иметь хотя бы один корневой порт
 - ✓ Совокупная стоимость всех каналов к корневому мосту называется корневой стоимостью (Root Path Cost)
 - ✓ Корневой мост отправляет сообщения BPDU, указывая корневую стоимость 0 (поле RPC=0)
 - ✓ Корневая стоимость пути (поле RPC в BPDU) увеличивается при получении мостом BPDU, а не при отправке

● **Этап 3. Выбор назначенных портов (Designated Port)**

- Ближайший к корневому мосту “порты сегментов сети” становятся назначенными портами
 - ✓ Каждый сегмент сети, подключенный к нескольким мостам, должен иметь хотя бы один назначенный порт. Через назначенный порт моста осуществляется обмен трафиком между сегментом сети и корневым мостом
 - ✓ Мост, имеющий назначенный порт, называется *назначенным мостом* для данного сегмента
 - ✓ Назначенным для данного сегмента становится тот мост, значение корневой стоимости пути (RPC, рассчитан на этапе 2) у которого меньше чем у других мостов, подключенных к данному сегменту
 - ✓ Назначенным становится порт назначенного моста, который подключен к данному сегменту

Алгоритм принятия решения

- **Алгоритм принятия решения** (при одинаковых значениях сравниваемых параметров, в порядке понижения приоритета)

1. По наименьшему идентификатору корневого моста (Root BID)
2. По наименьшей стоимости маршрута к корневому мосту (RPC)
3. По наименьшему идентификатору моста-отправителя (BID)
4. По наименьшему идентификатору порта-отправителя (Port ID)

Мосты используют Bridge PDU (BPDU) для обмена информацией протоколу STP.

Логика работы моста:

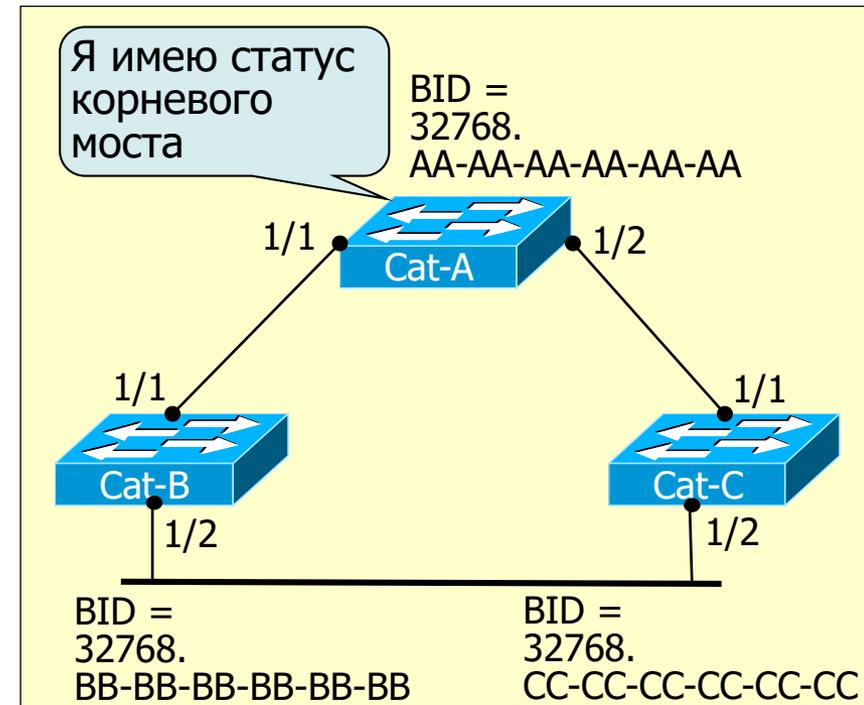
- Каждый мост сохраняет копии наиболее приемлемых BPDU, полученных на каждый порт, согласно алгоритма принятия решения:
 - По приходу в порт BPDU-сообщения от другого моста, алгоритм сравнивает его с существующим BPDU, сохраняя более приемлемое (имеющее наименьшие показатели)
- Этот же процесс управляет отправкой BPDU-сообщения
 - Если полученное через порт BPDU-сообщение более приемлемое, отправка через этот порт BPDU-сообщений прекращается
 - Если получение через порт более приемлемого BPDU-сообщения приостановилось на 20 с (стандартное значение) отправка BPDU-сообщения восстанавливается

Роли и Состояния портов

- **Роли портов**, определяют топологию сети
 - **Root** - корневой
 - **Designated** – назначенный
 - **Nondesigned** – не назначенные, блокирует передачу данных, разрывая петлю
- **Состояния портов** задают возможность передачи кадров
 - **Blocking** - блокирование. *Блокированный порт не передает кадры данных.* Назначение - предотвращение петель в сети. *Блокированный порт принимает кадры BPDU, чтобы разблокироваться в случае инициации процесс переконфигурации топологии*
 - **Listening** - прослушивание: *Порт принимает и отправляет только BPDU, кадры с данными не отправляет.* Из данного состояния порт не переходит в следующее состояние (Learning) для гарантированного выбора корневого моста в течении Forward delay timer, который по умолчанию равен 15 сек.
 - **Learning** – обучение. *Порт только принимает кадры данных с целью построения таблицы коммутации (CAM- таблицы) и не пересылает кадры данных.* Переход в следующее состояние (Forwarding) по истечении Forward delay timer.
 - **Forwarding** - пересылка: посылает/принимает BPDU кадры и кадры с данными, участвует в поддержании таблицы коммутации (MAC-адресов). *Это обычное состояние рабочего порта.*
 - **Disabled** - отключен (disabled): В это состояние порт переводит администратор сети, состояние administratively down, отключен командой shutdown. В этом состоянии порт ничего не передает, пока вручную его не включат

Этап 1. Выбор корневого моста (1)

- Корневым выбирается мост с наименьшим **Bridge ID (BID)**
 - Процесс выбора моста с наименьшим **BID** называется "корневой борьбой"
- Поле BID имеет длину 8 байт:**
 - Приоритет моста - 2 байта, диапазон: 0-65535, по умолчанию = 32768
 - MAC-адрес - 6 байт
- Процесс корневой борьбы**
 - BPDU-сообщения стандартно отправляются каждые 2 с
 - Вначале мосты в поля Root ID и Sender ID подставляют собственные значения BID
 - Затем мост в поле Root ID сохраняет лучшее из BPDU, полученных на каждый порт. Каждый проходящий BPDU сравнивается с ранее сохраненным. Если полученный BPDU более приемлем, то новое сообщение заменяет записанное
 - мосты Cat-A и Cat-B определяют, что наименьший BID у моста Cat
 - Поэтому Cat-A становится корневым



... Фрагмент BPDU-сообщения	
Root ID – BID корневого моста	→ Кто имеет статус корневого моста ?
Root path cost - Корневая стоимость	→ Каково расстояние до корневого моста ?
Sender ID - BID моста отправителя	→ Каков BID моста отправителя BPDU ?
Port Identifier - Идентификатор порта	→ С какого порта было отправлено BPDU
...	

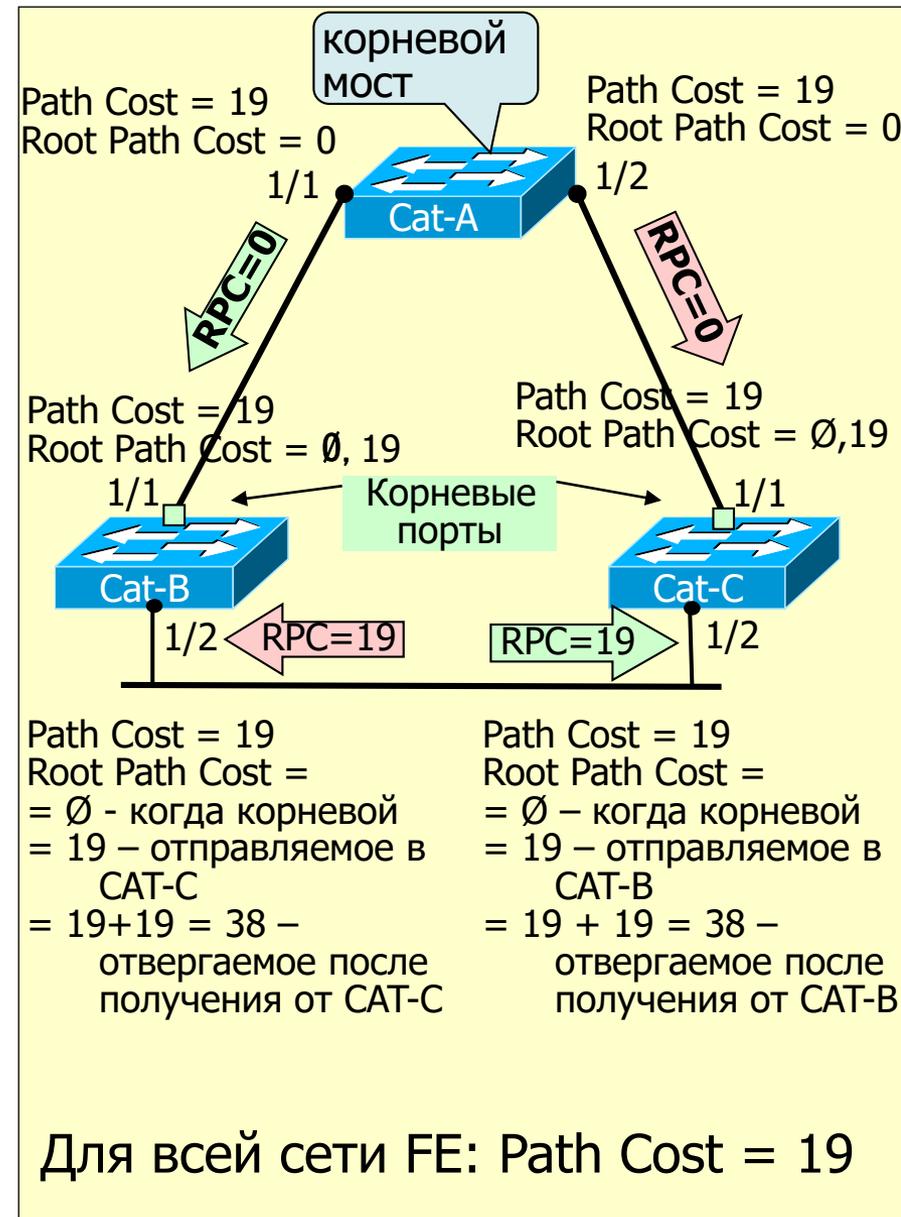
Этап 1. Выбор корневого моста (2)

Пример корневой борьбы

- Предположим, что мост Cat-B загружается в первый раз и каждые 2 с отправляет в сеть BPDU-сообщения, объявляя себя корневым мостом
- Через несколько минут загружается мост Cat-C и уверенно пытается назначить корневым мостом себя
 - ✓ Однако когда BPDU-сообщения моста Cat-C достигнут устройства Cat-B, они будут уничтожены, т.к. BID моста Cat-B меньше BID моста Cat-C
 - ✓ А когда BPDU-сообщения моста Cat-B достигнут Cat-C мост Cat-C обозначит Cat-B как корневой. Т.е с этого момента мост Cat-C будет подставлять в поле Root ID BPDU-сообщений BID моста Cat-B, а в поле Sender ID свой BID
 - ✓ Таким образом, все рабочие мосты в сети придут к соглашению, что на данный момент корневым мостом является устройство Cat-B.
- Пять минут спустя загрузится мост Cat-A
 - ✓ мост начнет сразу отправлять BPDU-сообщения по сети и попытается установить себя корневым
 - ✓ Как только сообщения от устройства Cat-A достигнут остальных мостов, мост Cat-A будет признан корневым
 - ✓ В таком случае все три моста будут отправлять сообщения BPDU, указывая
 - в поле Root ID идентификатор корневого моста Cat-A,
 - а в поле Sender ID — собственные значения идентификаторов

Этап 2: Выбор корневых портов (1)

- **Ближайшие к корневому мосту порты называются корневыми (Root Port)**
 - Корневой порт имеет по сети кратчайшее расстояние до корневого моста
- **Кратчайшее расстояние к корню**
 - Называется корневой стоимостью - Root path cost
 - Является совокупной стоимостью всех каналов к корневому мосту
 - Формируется в поле Root path cost BPDU-сообщения
 - Корневой мост отправляет BPDU-сообщение с **Root path cost (RPC) = 0**
- **Стоимость канала назначается для каждого порта - Path Cost**
 - Path Cost зависит от скорости канала (порта) между мостами
 - Например, для 100Мбит/с - Path Cost = 19
- **При получении BPDU сообщения через порт**
 - к полю Root path cost BPDU-сообщения прибавляется назначенное для этого порта Path Cost
 - *ИТАК*, стоимость RPC увеличивается при получении через порт моста сообщения BPDU



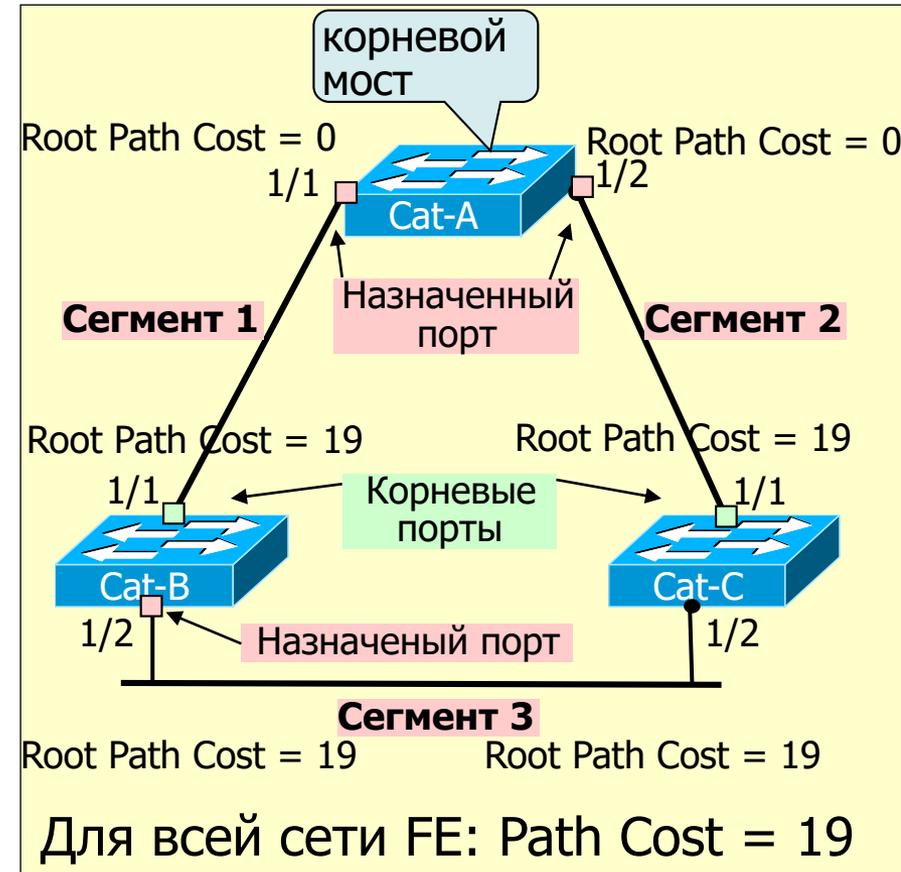
Этап 2: Выбор корневых портов (2)

Шаги возможного процесса

- 1. Cat-A (корневой мост) отправляет сообщения BPDU с корневой стоимостью равной 0 [Root Path Costs (RPC) = 0]**
 - $RPC_{Cat-A} = 0$
- 2. Получив такое BPDU мост Cat-B добавляет к значению корневой стоимости значение стоимости порта 1/1 равное для FE числу 19**
 - $RPC_{Cat-B} = RPC_{Cat-A} + Path\ COST_{Cat-B} = 0 + 19 = 19$
- 3. Cat-B отправляет BPDU с $RPC_{Cat-B} = 19$ через другие порты**
- 4. BPDU от моста Cat-B получает мост Cat-C на порту 1/2 и рассчитывает**
 - $RPC_{Cat-C(1/2)} = RPC_{Cat-B} + Path\ COST_{Cat-C(1/2)} = 19 + 19 = 38$
- 5. С другой стороны, на порт 1/1 Cat-C от корневого моста приходят сообщения со значением стоимости, равным 0, и затем мост Cat-C увеличивает ее на 19**
 - $RPC_{Cat-C(1/1)} = RPC_{Cat-A} + Path\ COST_{Cat-C(1/1)} = 0 + 19 = 19$
- 6. Мост Cat-C выберет порт 1/1 в качестве корневого со значением $RPC=19$**
- 7. Мост Cat-C отправляет BPDU с $RPC_{Cat-C} = 19$ через другие порты**
- Мост Cat-B выполняет аналогичные вычисления**
 - Стоимость пути от корневого моста до порта 1/1 устройства Cat-B равняется 19, в то время как стоимость пути от порта 1/2 моста Cat-B равняется 38, поэтому порт 1/1 становится корневым для коммутатора Cat-B

Этап 3: Выбор назначенных портов (1)

- **Каждый сегмент в сети должен иметь хотя бы один назначенный порт**
- **Основным критерием при выборе назначенного порта является наименьшая стоимость маршрута к корню (корневая стоимость)**
- **Функцию назначенного порта выполняет только один порт одного из мостов**
 - Основная идея назначенных портов состоит в том, что если только один порт обрабатывает трафик для определенного сегмента сети, то возможность появления замкнутых петель в сети исчезает
- **Мост, имеющий хотя бы один назначенный порт, называется назначенным мостом для данного сегмента**



Этап 3: Выбор назначенных портов (2)

ПРИМЕР выбора назначенных портов

- **Сегмент 1 имеет подключения по двум портам: порт 1/1 устройства Cat-A и порт 1/1 устройства Cat-B**
 - порт 1/1 Cat-A имеет корневую стоимость, равную 0
 - порт 1/1 Cat-B имеет корневую стоимость, равную 19
 - порт 1/1 Cat-A имеет меньшую корневую стоимость, поэтому становится назначенным
- **Сегмент 2 имеет подключения по двум портам:**
 - порт 1/2 Cat-A имеет корневую стоимость, равную 0, а порт 1/1 Cat-C — 19
 - Т.к. порт 1/2 Cat-A имеет меньшую корневую стоимость, то он становится назначенным

Ремарка: каждый активный порт корневого моста становится назначенным портом

- ✓ Исключением из правила являются физические петли к корневому мосту (например, когда два порта корневого моста подключаются к одному концентратору или два порта соединены кабелем).

Этап 3: Выбор назначенных портов (3)

● Сегмент 3 подключен к двум коммутаторам, имеющими одинаковое значение корневой стоимости, равное 19

- В такой ситуации срабатывает "Алгоритм принятия решения" протокола STP
 1. По наименьшему идентификатору корневого моста (Root BID)
 2. По наименьшей стоимости маршрута к корневному мосту
 3. По наименьшему идентификатору моста-отправителя (BID)
 4. По наименьшему идентификатору порта (Port ID)
- 1. Cat-B и Cat-C не корневые, поэтому переходим к пункту 2 алгоритма
- 2. Cat-B и Cat-C имеют одинаковые значения стоимости пути к корню, равные 19, поэтому переходим к пункту 3 алгоритма
- 3. Идентификатор коммутатора Cat-B (32768.BB-BB-BB-BB-BB-BB) меньше, чем идентификатор коммутатора Cat-C (32768.CC-CC-CC-CC-CC-CC), поэтому
 - порт 1/2 коммутатора Cat-B становится назначенным для сегмента 3
 - порт 1/2 коммутатора Cat-C получает статус не назначенного порта

● Три этапа начальной сходимости:

- Этап 1. Выбор корневого моста (Root Bridge).
- Этап 2. Выбор корневых портов (Root Port).
- Этап 3. Выбор назначенных портов (Designated Port).
 - ✓ На первом этапе в сети назначается один корневой мост.
 - ✓ На втором этапе, в каждом некорневом мосту назначается один корневой порт, т.е. порт, ближайший к корневому мосту.
 - ✓ На третьем этапе для каждого сегмента выбирается один назначенный порт.

● Алгоритм принятия решения (при одинаковых значениях сравниваемых параметров, в порядке понижения приоритета)

1. По наименьшему идентификатору корневого моста (Root BID)
2. По наименьшей стоимости маршрута к корневому мосту
3. По наименьшему идентификатору моста-отправителя (BID)
4. По наименьшему идентификатору порта-отправителя (Port ID)

● Суть процесса работы моста

1. Все сообщения BPDU, получаемые через порты моста сравниваются как между собой, так и сообщениями, отправленными из портов
2. Передается через порт (в порт записывается) только наиболее приемлемое BPDU сообщение
 - "Приемлемым" считается сообщение с наименьшими параметрами (RPC, BID, PID).
 - ✓ например, мост с наименьшим идентификатором BID становится корневым, а наименьшая стоимость маршрута RPC позволяет определить корневые и назначенные порты
3. Порт останавливает отправку сообщений BPDU, если от него было получено более приемлемое сообщение, чем отправляемое

4. Форматы BPDU кадров

Конфигурационные BPDU

Сообщения об изменении топологии TCN BPDU

Обработка конфигурационных BPDU

Два типа сообщений BPDU (1)

Существуют два типа сообщений BPDU:

1. Конфигурационные BPDU

- создаются корневым мостом и далее распространяются по ветвям дерева

2. Сообщения об изменении топологии TCN BPDU (TCN — Topology Change Notification)

- распространяются в обратном направлении, чтобы сообщить корневному мосту об изменении в топологии

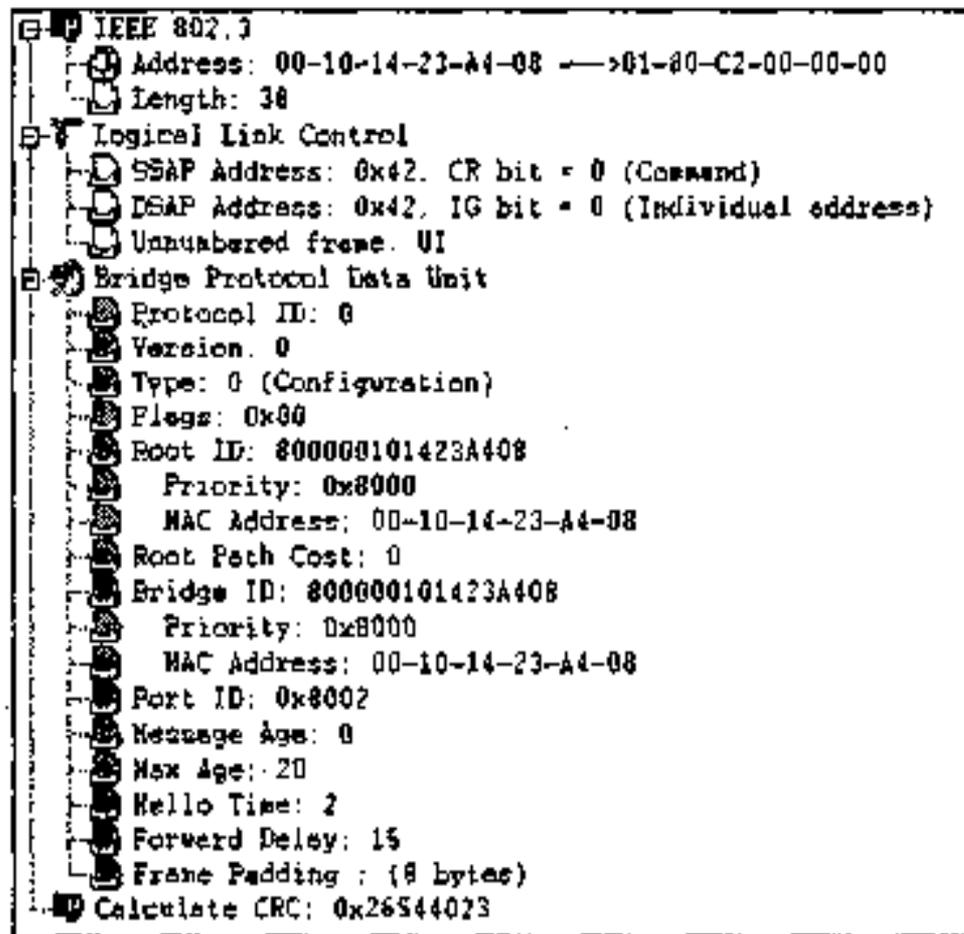


Рис. 6.15. Формат сообщений BPDU

MAC / SAP- адрес отправителя BPDU

- В кадре Ethernet 802.3 поле **Адрес отправителя (Source address)** — это **MAC-адрес порта коммутатора, который отправил BPDU**
 - Каждый порт коммутатора имеет свой уникальный (unicast) MAC-адрес
 - MAC-адрес порта используется для формирования поля SA в заголовке кадра, в котором отправляется сообщение BPDU
- В кадре Ethernet 802.2 поле **SSAP (source service access point)** — **сервисная точки доступа отправителя**
 - SSAP = DSAP = 0x42 (закреплены для STP)

ВНИМАНИЕ

- **Разница между MAC-адресом любого порта коммутатора и MAC-адресом используемого для идентификатора моста (BID)**
 - MAC-адрес моста, используемый при формировании поля BID, берется от Supervisor или аппаратный адрес, который написан на задней панели коммутатора
 - Если на задней панели нет этого адреса, выбирается минимальный адрес порта коммутатора
- **КОНКРЕТНАЯ РЕАЛИЗАЦИЯ**
 - Коммутаторы Catalyst всегда имеют на один MAC-адрес больше, чем количество портов: отдельный MAC-адрес на каждый порт, а также еще один дополнительный адрес — MAC-адрес всего устройства (для BID)

MAC / SAP- адрес получателя BPDU

● MAC-адрес получателя (DA) ← 802.3

- используется групповой (multicast) MAC-адрес = 01-80-C2-00-00-00

● **Сервисная точка доступа получателя DSAP = 0x42** ← LLC

- DSAP (destination service access point) — сервисная точка доступа получателя
- SSAP (source service access point) — сервисная точки доступа отправителя
- DSAP = SSAP = 0x42 (закреплены за STP)
- В двоичном исчислении 0x42 соответствует числу 01000010

● **В Тело Ethernet кадра вложена информация BPDU**

- Конфигурационное сообщение BPDU состоит из 12 полей, которые подробно описаны ниже

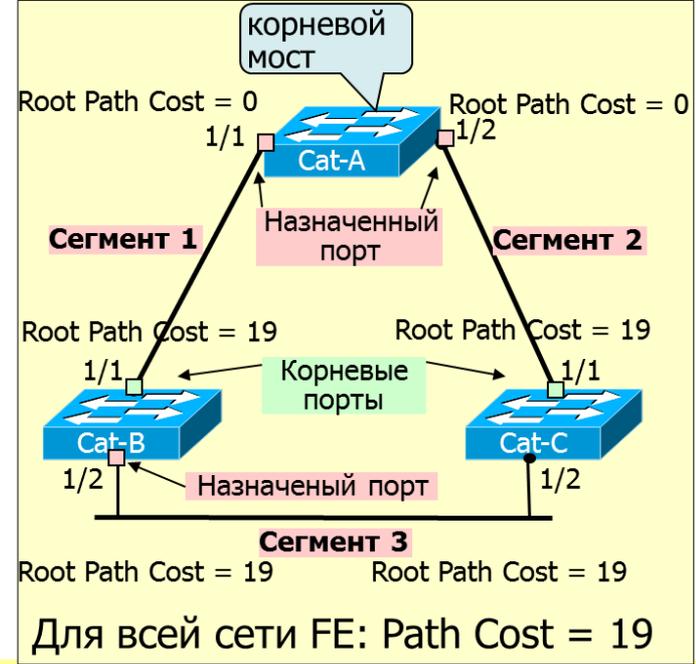
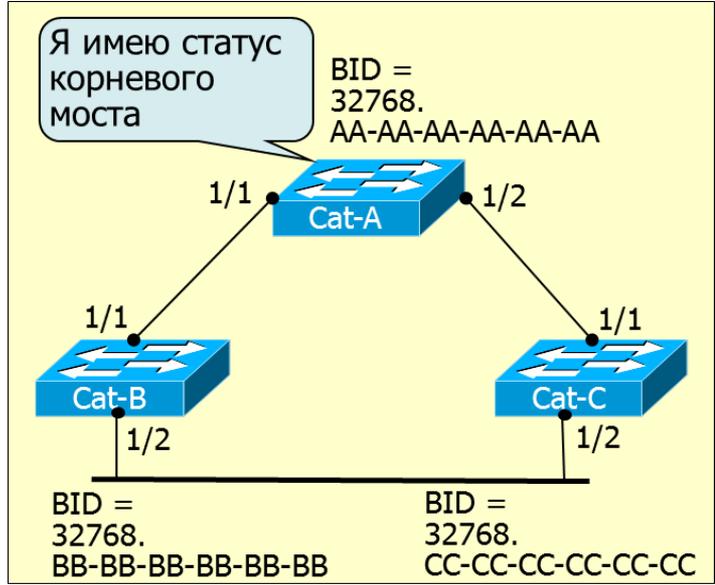
Формат конфигурационного сообщения BPDU (1)

BPDU - Bridge Protocol Data Unit (OSI термин для обозначения протокольных блоков данных)

Поля BPDU используются для:

- Root Identifier** → Какой BID примет статус корневого моста ???
- Root Path Cost** → Каково расстояние до корневого моста ???
- Sender Identifier** → BID моста отправителя BPDU-сообщения
- Port Identifier** → PID порта отправителя BPDU-сообщения

	количество	байт
Protocol Identifier	2	
Protocol Version	1	
BPDU Type	1	
Flags	1	
Root Identifier	8	
Root Path Cost	2	
Sender Identifier	8	
Port Identifier	2	
Message Age	2	
Max Age	2	
Hello Time	2	
Forward Delay	2	



Формат конфигурационного сообщения BPDU (2)

- **Protocol Identifier** — идентификатор протокола
 - 0000 (hex) для STP 802.1D
- **Protocol Version** — версия
 - 00 (hex) для версии 802.1D (1998)
 - 02 (hex) для версии 802.1D (2004)
- **BPDU Type** — тип сообщения
 - 00 (hex) для конфигурационных BPDU
 - 80 (hex) для сообщения об изменении топологии TCN
- **Root Identifier** — BID (идентификатор) корневого моста
 - 2 байта для приоритета (по умолчанию 32768)
 - 6 байт для MAC-адреса
- **Root Path Cost** — стоимость маршрута к корневому мосту
 - Диапазон 1-65535
- **Sender Identifier** — BID (идентификатор моста) отправителя BPDU сообщения
- **Port Identifier** — PID (идентификатор порта) отправителя BPDU сообщения
 - 1 байт приоритета (по умолчанию 128)
 - 1 байт номера порта

Формат конфигурационного сообщения BPDU (3)

- **Message Age** — возраст конфигурационного сообщения BPDU (1-10 секунд)
 - Корневой мост при передаче конфигурационного BPDU устанавливает переменную в 0
 - Каждая передача (через назначенный порт моста) увеличивает это число
- **Max Age** — максимальный срок хранения конфигурационных BPDU (6-40 с)
 - Ограничение жизни полученных конфигурационных BPDU
 - Основной параметр для обнаружения idle отказов (например, корневой мост мертв)
 - Поле оказывает влияние на таймер времени хранения таблицы моста в процессе уведомления об изменении топологии
 - По умолчанию 20 с.
- **Hello Time** (1-10 с) —
 - интервал времени между периодической отправкой корневым мостом конфигурационных сообщений BPDU (“configuration BPDU”)
 - ✓ Корневой мост отправляет конфигурационные сообщения каждые Hello Time секунд
 - ✓ Мосты, получившие такое сообщение, отправляют его дальше в сеть
 - по умолчанию 2 с
- **Forward Delay** (диапазон 4-30 с)
 - Время задержки перевода порта в состояние Forward (передача)
 - По умолчанию 15 с
 - Но это означает 15 сек состояний прослушивания + 15 сек изучения топологии

Ремарка: Hello Time, Max Age, Forward Delay специфицируются (конфигурируются) в корневом мосте

● **Flags — признаки,**

- Бит 8 – Подтверждение изменения топологии (TCA - Topology Change Acknowledgement)
- Бит 1 – Изменение топологии (TC - Topology Change)
- Используются в TCN BPDU для сигнализации об изменении топологии
 - ✓ Примечание: в случае изменения топологии MAC-адреса могут быть найдены в таблице моста на другом порту; поэтому необходимо переобучение динамических «таблиц коммутации» моста; переобучение запускается корневым мостом
 - ✓ Лекц Таненбаума

Сообщения BPDU об изменении топологии

- Сообщения об изменении топологии (Topology Change Notification – TCN)
- TCN BPDU играют ключевую роль в обработке изменений активной топологии

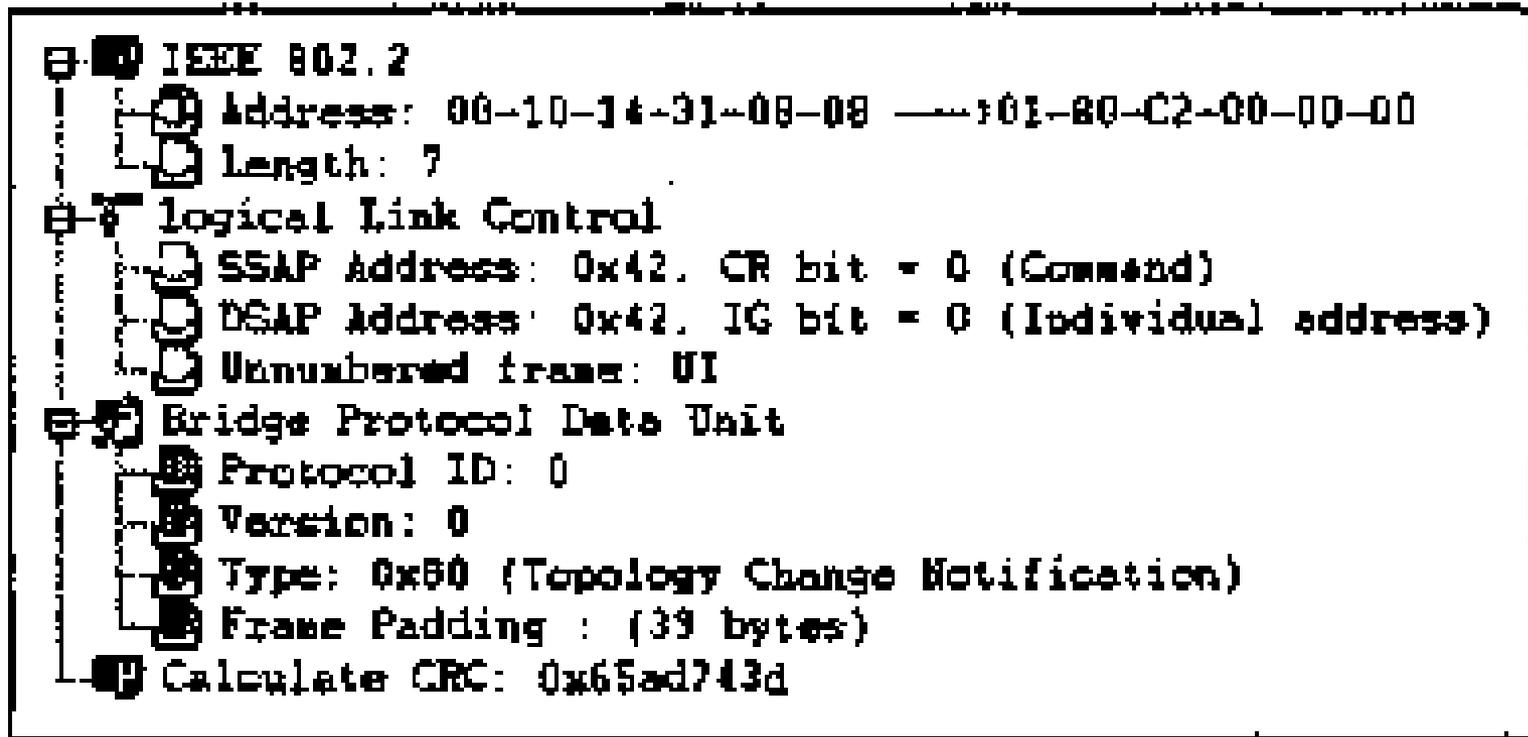


Рис. 6.16. Структура сообщений TCN BPDU

5. Обработка BPDU

Обработка конфигурационных BPDU

Обработка сообщения об изменении топологии TCN BPDU

Обработка конфигурационных BPDU

- **Вначале все мосты обмениваются хаотической информацией**
 - 4-х этапный алгоритм принятия решения позволяет мостам сформировать единое дерево
- **Определение наилучшего конфигурационного BPDU**
 - Если собственное конфигурационное сообщение BPDU порта является лучшим, отправление сообщений в сеть (порт) продолжается
 - Отправление конфигурационных сообщений BPDU начинается в тот момент, когда порт переходит в состояние прослушивания
 - В таком состоянии происходит обработка исключительно сообщений BPDU, пользовательский трафик не коммутируется.
 - По истечении таймера задержки распространения (15 с), порт переходит в состояние самообучения. В данном состоянии происходит построение таблицы коммутации по MAC-адресу user-кадров отправителя, однако входящие кадры от пользователей сети все еще игнорируются.
 - По истечении таймера самообучения, порт переходит в состояние передачи и начинается коммутация пользовательских данных
- **Если LAN (сеть) находится в устойчивом состоянии**
 - Конфигурационные BPDU распространяются от корневого моста вдоль ответвлений к каждому сегменту сети
- **Любое изменение физической топологии вызывает запуск "Процесса изменения топологии"**

Стандартная обработка конфигурационных BPDU (1)

Обработка конфигурационных сообщений BPDU: стандартная и исключительная

Стандартная

- корневой мост на каждом порту по hello-таймеру (2 с) генерирует конфигурационные BPDU
- некорневой мост получает конфигурационное BPDU непосредственно в корневой порт и отправляет измененные версии полученных сообщений на каждый из назначенных портов (перенаправление конфигурационных сообщений BPDU от корневого моста)

● **Таким образом, поток конфигурационных BPDU направлен от корневого моста;**

- конфигурационные BPDU получают только корневые порты;
- корневые порты не отправляют конфигурационных BPDU;

● **если корневой мост неисправен**

- отправление конфигурационных BPDU в сеть приостанавливается
- отсутствие конфигурационных сообщений в сети продолжается до тех пор, пока на другом коммутаторе не сработает таймер Max Age (максимальное время старения записи в кэше), и он примет статус корневого моста

Стандартная обработка конфигурационных BPDU (2)

- **если канал к корневому мосту неисправен** (а сам корневой мост останется активным),
 - отправление конфигурационных BPDU другим мостам приостанавливается
 - При наличии альтернативного маршрута к корневому мосту отсутствие конфигурационных сообщений BPDU наблюдается, пока один из оставшихся портов с рабочим каналом не выйдет из состояния блокировки.
 - Если альтернативный путь недоступен, сеть, состоящая из устройств канального уровня, будет разделена на части и для каждого из сегментов сети будет назначен свой корневой мост
- Таким образом, при стандартной обработке конфигурационных BPDU некорневые мосты отправляют в назначенные порты сообщения, полученные от корневого моста

Исключительная обработка конфигурационных BPDU (1)

Исключительная обработка конфигурационных BPDU

когда назначенный порт получает худшие BPDU (1) от некоторого другого устройства и отправляет в ответ (2) полученные сообщения.

STP использует исключительную ситуацию для подавления ложной информации

- Например, перед тем, как коммутатор Cat-C был подключен к концентратору (этап 1), корневой мост пришел в неисправность (этап 2), как показано на рис. 7.5.
- В начале работы Cat-C пытается назначить себя корневым мостом и сразу начинает отправлять конфигурационные сообщения BPDU, которые указывают на то, что он является корневым

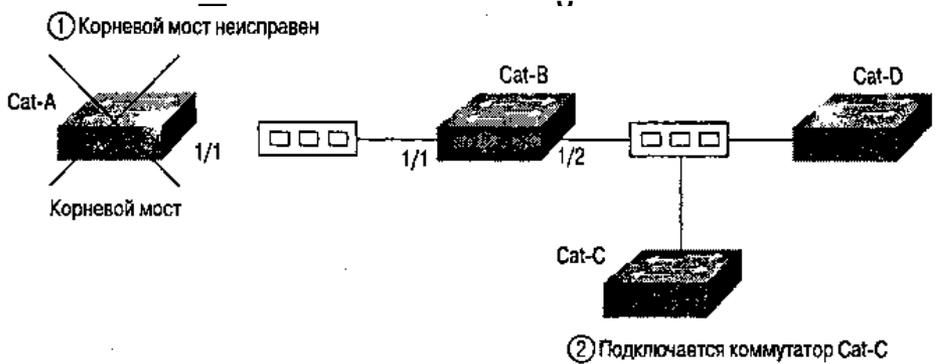


Рис. 7.5. Перед подключением коммутатора Cat-C корневой мост пришел в неисправность

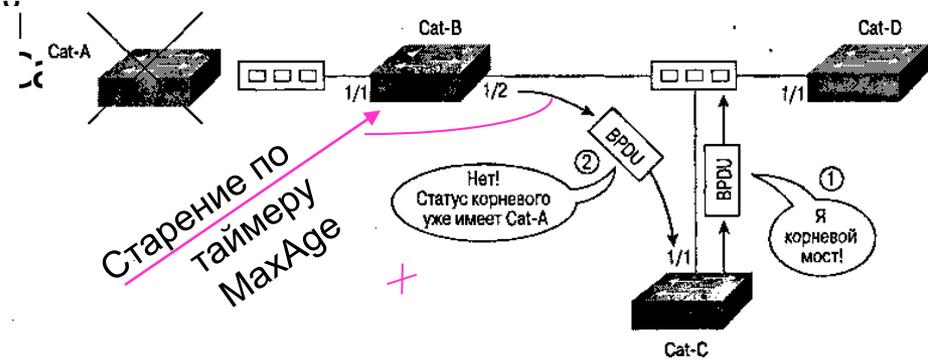


Рис. 7.6. Обработка конфигурационных сообщений BPDU в исключительных ситуациях

Исключительная обработка конфигурационных BPDU (2)

- Т.к. в настоящий момент корневой мост находится в неисправном состоянии, порт 1/2 коммутатора Cat-B приостанавливает отправку конфигурационных сообщений
- Поскольку порт 1/2 Cat-B имеет статус назначенного порта для указанного сегмента, он немедленно отвечает на конфигурационные сообщения Cat-C, объявляя корневым мостом Cat-A. Выполняя указанные операции, Cat-B предохраняет Cat-C от получения статуса корневого моста и создания логических петель в активной топологии сети

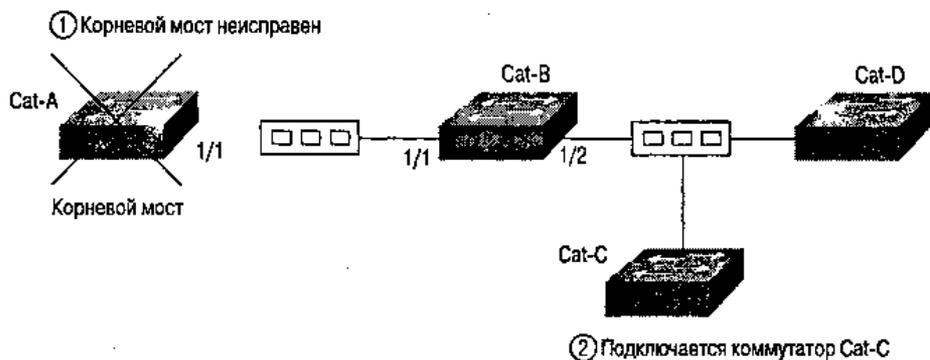


Рис. 7.5. Перед подключением коммутатора Cat-C корневой мост пришел в неисправность

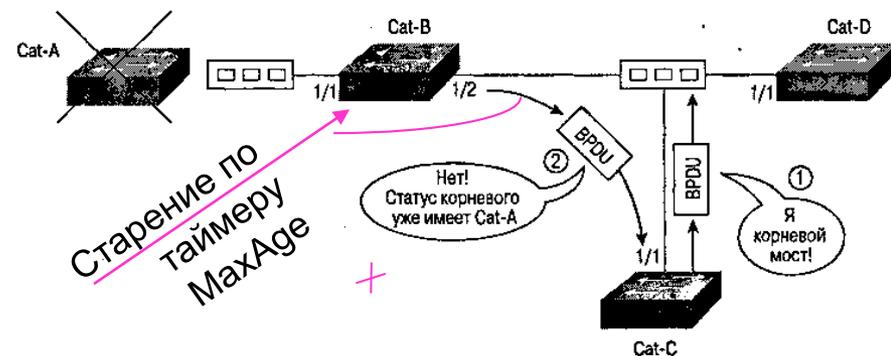


Рис. 7.6. Обработка конфигурационных сообщений BPDU в исключительных ситуациях

Исключительная ситуация (продолжение)

● Правила обработки конфигурационных BPDU в исключительных ситуациях

- Назначенные порты могут отвечать на ложные конфигурационные сообщения PDU в любое время
- Пока в время старения (MaxAge) информации в коммутаторе Cat-B о корневом мосте (Cat-A) не истекло, Cat-B будет отвергать ложные сообщения BPDU (в нашем случае от Cat-C)
- По истечении времени старения (MaxAge) информации о корневом мосте коммутатор Cat-B объявляет себя корневым

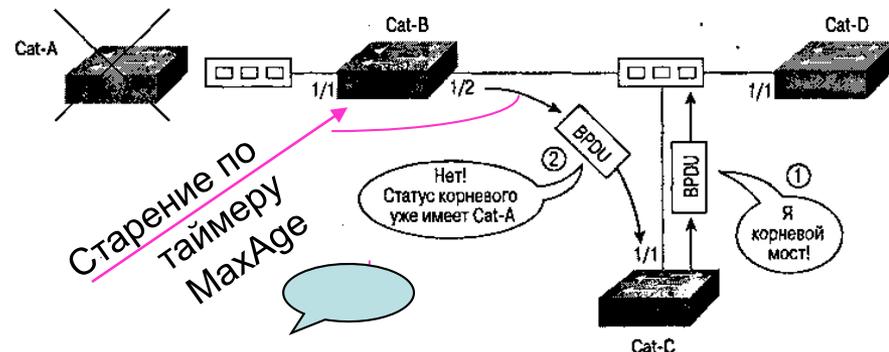


Рис. 7.6. Обработка конфигурационных сообщений BPDU в исключительных ситуациях

Исключительная ситуация (продолжение)

● Правила обработки конфигурационных BPDU в исключительных ситуациях

- Если устройства в сети быстро отвергают ложную информацию, то время сходимости такой сети будет минимальным. Можно представить, что случится, если Cat-B будет использовать стандартную обработку конфигурационных сообщений — устройство Cat-C будет иметь в своем распоряжении 20 с, чтобы неправильно назначить себе статус корневого и неумышленно создать логическую петлю в активной топологии сети. Даже если такая ситуация не приведет к возникновению петель, процесс выбора корневого и назначенных портов может неоднократно повториться и дестабилизировать состояние сети.

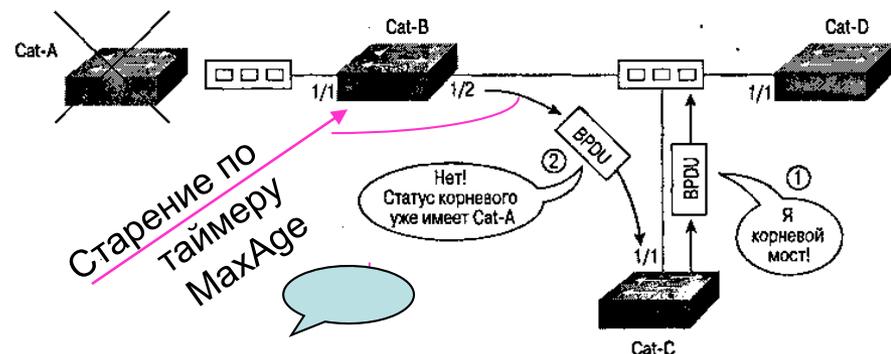


Рис. 7.6. Обработка конфигурационных сообщений BPDU в исключительных ситуациях

Конфигурационные сообщения отправляются в трех случаях:

- 1. По hello-таймеру** (стандартно — каждые 2 с), корневой мост отправляет конфигурационные сообщения BPDU на каждый порт (стандартная обработка)
- 2. Когда некорневые мосты получают конфигурационные сообщения на корневой порт**, полученное сообщение отправляется во все назначенные порты (стандартная обработка)
- 3. Когда назначенный порт получает ложное конфигурационное сообщение BPDU** от другого коммутатора, указанный порт сразу отвечает сообщением, которое записано в кэше порта, для того чтобы подавить ложную информацию.

Обработка TCN BPDU (1)

- 1. Когда некорневой мост определяет, что в активной топологии появились изменения, то корневому мосту отправляются уведомления об изменении активной топологии сети TCN BPDU**
- 2. Получив такие TCN BPDU- сообщения, корневой мост рассылает всем остальным мостам в сети конфигурационные BPDU-сообщения с меньшим значением поля времени старения (MaxAge) таблиц коммутации с 300 с до значения таймера задержки распространения**
- 3. Уведомления об изменении топологии TCN BPDU отправляются в трех случаях, но на практике можно выделить два основных случая их использования:**
 - обнаружение изменения в топологии
 - и их распространение

Обработка TCN BPDU (2)

- **Обработка обнаружения изменений топологии возникает в том случае**
 - порт моста переходит в состояние прослушивания и сам мост имеет хотя бы один назначенный порт
 - переходе порта из режима распространения или обучения в режим блокировки
- **Обработка распространения изменений топологии возникает в том случае, когда некорневой мост получает сообщение TCN в назначенный порт**

Обработка TCN BPDU (3)

1. Две ситуации рассылки сообщений объединяются в общий термин — **обнаружение изменений** и отображают перестройку активной топологии сети в таблицах коммутации мостов сети
2. Термин **распространение изменений** используется для описания процесса передачи TCN BPDU-сообщений через ветви связующего дерева, пока они не достигнут корневого моста

Обработка TCN BPDU (4)

Совет

В классическом описании алгоритма распределенного связующего дерева уведомления об изменении топологии TCN BPDU отправляются в трех случаях:

1. когда порт переходит в состояние передачи данных и сам мост имеет хотя бы один назначенный порт (часть процесса обнаружения изменений в сети);
2. когда порт моста переходит из состояния передачи данных или обучения в состояние блокировки (обнаружение изменения);
3. когда назначенный порт получает уведомление TCN BPDU он отправляет его в корневой порт (распространение информации об изменении)

Выводы

- 1. TCN BPDU отправляются только корневыми портами**
- 2. TCN BPDU отправляются только в назначенные порты**
- 3. TCN BPDU принимаются назначенными портами**
- 4. Поток TCN BPDU всегда направлен к корневому мосту**
- 5. TCN BPDU использует надежный механизм передачи данных, чтобы достигнуть корневого моста**
 - Передача TCN BPDU повторяется с интервалом hello-таймера до получения подтверждения (TCN BPDU с установленным флагом TCA (Topology Change Acknowledgment — подтверждение о получении сообщения об изменении топологии)).
 - *Новые сообщения TCN BPDU не будут генерироваться и отправляться вплоть до следующего изменения активной логической топологии сети (что может произойти через час, день, неделю и даже позже!)*
- 6. Сообщения TCN BPDU требуют отправки уведомления о получении, даже если процесс нормальной обработки конфигурационных сообщений BPDU в данный момент бездействует (например, в том случае, если поток конфигурационных сообщений BPDU от корневого моста отсутствует)**

6. Состояния и таймеры STP

Основные состояния протокола STP (1)

- После определения каждым мостом корневых, назначенных и обычных портов, топология без петель строится по правилу:
 - Корневые и назначенные порты передают данные (**Forwarding**)
 - Обычные порты блокируют передачу данных (**Blocking**)
 - ✓ Администраторы могут перевести порт в отключенное состояние (**Disabled**), это состояние не является составной частью процесса динамической обработки кадров
- **В стабильно работающей сети пять возможных состояний у STP**
 - Этот список можно представить как иерархию состояний от нижнего уровня (Blocking) до самого верхнего уровня (Forwarding)

Табл. Возможные состояния протокола STP

Состояние	Назначение
Передача данных (Forwarding)	Отправка и получение пользовательских данных через корневые и назначенные порты
Изучение топологии (Learning)	Построение таблиц коммутации (15 с)
Прослушивания (Listening)	Построение "активной" топологии (15 с)
Блокировка (Blocking)	Получение только сообщений BPDU
Отключен (Disabled)	Порт в неактивном состоянии (отключен админом)

Основные состояния протокола STP (2)

● Сразу после инициализации STP

- все порты переходят в состояние блокировки и обмениваются только сообщениями BPDU

● Некоторые события переводят мост в состояние прослушивания (**Listening**), например, отсутствуют сообщения BPDU определенный период времени

- мост принимает или отправляет только BPDU, а данные пользователя мостом игнорируются
- происходят три этапа начальной сходимости сети
- порты, потерявшие статус назначенных, определяются как неназначенные и снова переходят в состояние блокировки

● Порты, которые в течение 15 с (стандартное значение таймера) оставили за собой статус корневых или назначенных, переходят в состояние изучения топологии (**Learning**)

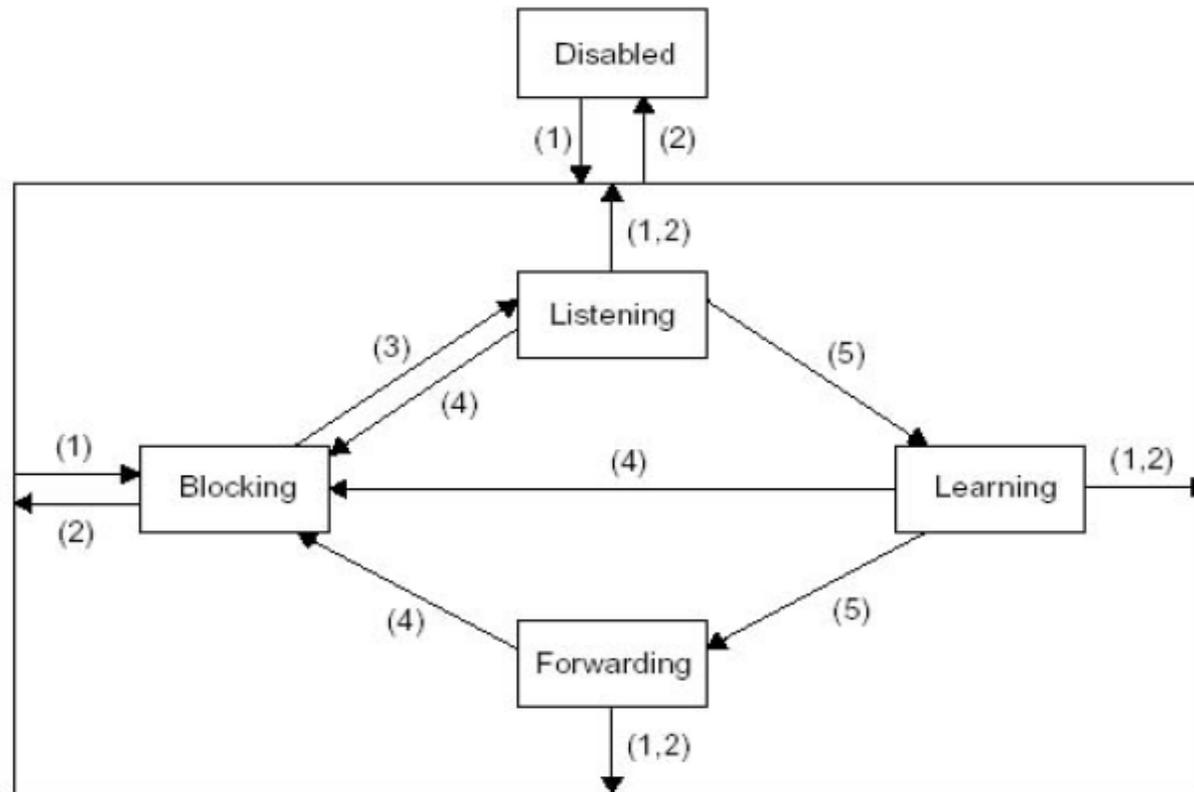
- это еще один 15-секундный период времени, когда мосты не пропускают пользователей
- строятся таблицы коммутации по MAC-адресам отправителя
- лавинное распространение сводится к минимуму, затем начинается передача данных пользователей

Основные состояния протокола STP (3)

- **Если порт к концу периода изучения топологии все еще имеет статус назначенного или корневого , происходит переход в состояние передачи информации (Forwarding) –** Продвижение - в этом состоянии порт может обрабатывать пакеты данных в соответствии с построенной таблицей коммутации. Также продолжают приниматься, передаваться и обрабатываться пакеты BPDU.
 - MAC-адресам отправителя
 - лавинное распространение сводится к минимуму, затем начинается передача данных пользователей

Основные состояния протокола STP (4)

- Таким образом, в процессе построения топологии сети каждый порт коммутатора проходит несколько стадий:



- 1) Порт активен или инициализация порта.
- 2) Порт отключен администратором или сбой порта.
- 3) Порт выбран в качестве корневого или назначенного порта.
- 4) Порт заблокирован.
- 5) Истек таймер смены состояний.

Рисунок 29. Состояния портов.

Основные состояния портов

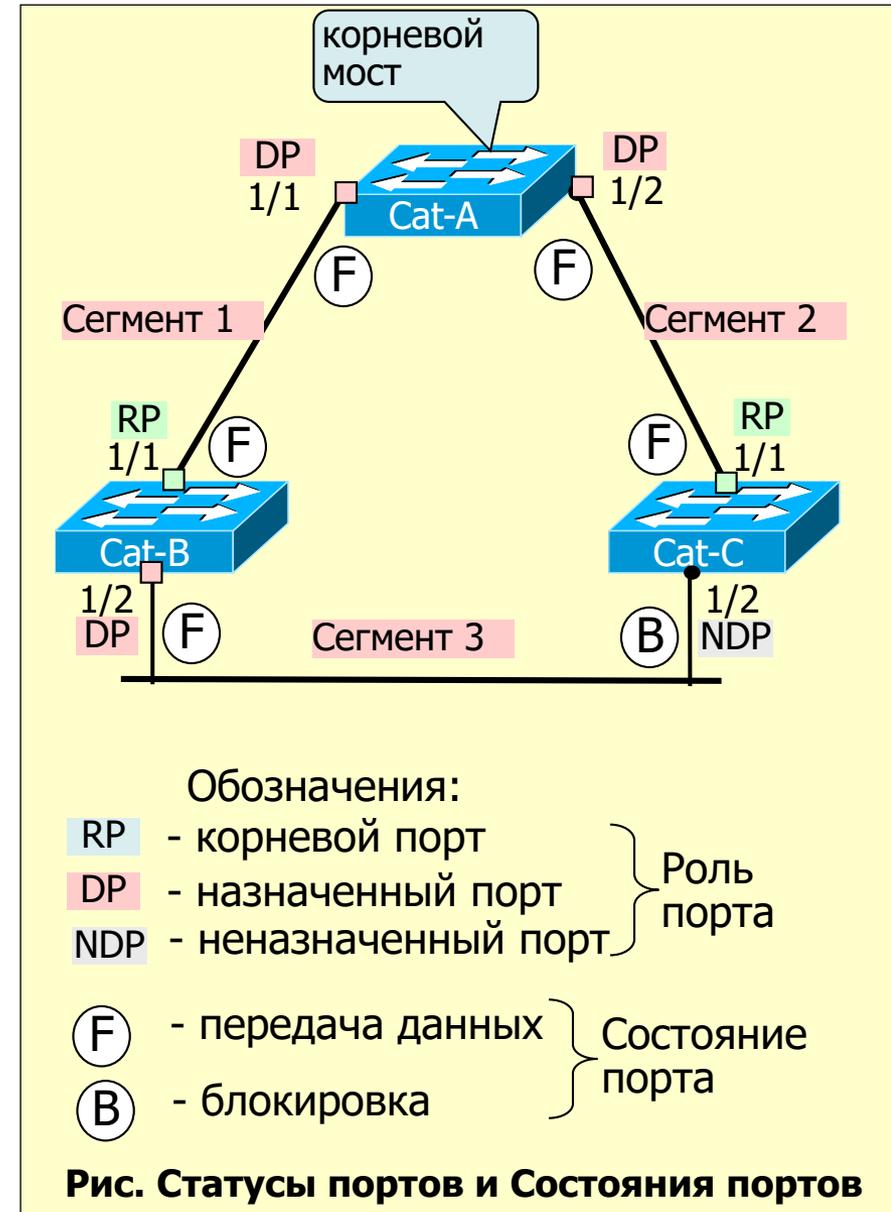
- **Blocking** - При инициализации коммутатора все порты (за исключением отключенных), а после построения дерева некоторые порты (не корневые и не назначенные) автоматически переводятся в состояние «Заблокирован». Блокированный порт не шлет ни чего, предотвращая петли в сети. Блокированный порт принимает и обрабатывает только кадры BPDU, чтобы разблокировать порт в случае необходимости. Все остальные пакеты отбрасываются.
- **Listening** – Прослушивание – в этом состоянии порт продолжает принимать и обрабатывать, а также начинает ретранслировать только кадры BPDU. Кадры с данными не передаются. Из этого состояния порт может перейти в состояние «Заблокирован», если получит BPDU с лучшими параметрами, чем его собственные (расстояние, идентификатор коммутатора или порта). В противном случае, при истечении таймера смены состояний, порт перейдет в следующее состояние «Обучение».
- **Learning** – Обучение – порт начинает принимать все пакеты и на основе адресов источника строить таблицу коммутации (CAM- таблицу, CAM-table — Content Addressable Memory Table). Порт в этом состоянии все еще не продвигает пакеты. Порт продолжает участвовать в работе алгоритма STA, и при поступлении BPDU с лучшими параметрами переходит в состояние «Заблокирован». В противном случае, при истечении таймера смены состояний, порт перейдет в следующее состояние «Продвижение».
- **Forwarding** – Продвижение - в этом состоянии порт может обрабатывать пакеты данных в соответствии с построенной таблицей коммутации. Также продолжают приниматься, передаваться и обрабатываться пакеты BPDU.
- **Disable** – Отключен – в это состояние порт переводит администратор. Отключенный порт не участвует ни в работе протокола STP, ни в продвижении пакетов данных. Порт можно также вручную включить и он сначала перейдет в состояние Blocking

Portfast

- **PortFast** – Предназначен для портов доступа и транковых портов не участвующих в формировании дерева STP.
- **PortFast** мгновенно переводит порт в состояние продвижения из заблокированного состояния, минуя состояния прослушивания и обучения.
- **Port Fast** применяется только для подключения компьютеров пользователей и серверов, а не коммутаторов, иначе может возникнуть петля. PortFast включается на уровне порта.
- **Portfast** следует включать только на интерфейсах, ведущих к конечным устройствам (рабочим станциям, серверам, телефонам и т.д.), но не к другим свичам.
- Репарка (<http://eucariot.livejournal.com/66249.html>): Есть очень удобная команда режима конфигурации интерфейса для включения нужных фиш на порту, в который будут включаться конечные устройства: `switchport host`. Эта команда разом включает PortFast, переводит порт в режим access (аналогично `switchport mode access`), и отключает протокол RAgP (об этом протоколе подробнее в разделе агрегация каналов).

Основные состояния протокола STP (6)

- В процессе нормальной работы корневой коммутатор продолжает генерировать служебные пакеты BPDU, а остальные коммутаторы продолжают их принимать своими корневыми портами и ретранслировать назначенными.
- Если по истечении максимального времени жизни сообщения (по умолчанию — 20 с) корневой порт любого коммутатора сети не получит служебный пакет BPDU, то он инициализирует новую процедуру построения связующего дерева
- На рис. Приведен пример сети с уже определенными ролями и состояниями портов



Таймеры протокола STP (1)

1. Hello-таймер - интервал между отправлением конфигурационных сообщений BPDU корневым мостом

- другие мосты транслируют BPDU, как только они получены от корневого
 - если в период времени от 2 до 20 с не поступают BPDU, то некорневые мосты приостанавливают распространение сообщений на указанный период времени
 - если время отсутствия сообщений превышает 20 с (Max Age), мост аннулирует ранее сохраненные портами сообщения BPDU и начинает поиск нового корневого порта
- **У всех мостов локально сконфигурированы свои hello-таймеры**
- используются как таймер повторной передачи уведомлений об изменении топологии (TCN — Topology Change Notification)

Табл. Таймеры протокола STP

Таймер	Назначение	Стандартное значение
Hello	Время между отправкой конфигурационных BPDU-сообщений корневым мостом	2 с
Forward Delay	Длительность состояний прослушивания и изучения топологии	15 с
Max Age	Время хранения BPDU-сообщений	20 с

Таймеры протокола STP (2)

2. Таймер задержки (Forward Delay)

- Контролирует длительность двух состояний STP — прослушивания (Listening) и изучения топологии (Learning)
- Стандартное значение равно 15 с и рассчитано на:
 - ✓ максимальный размер сети в семь переходов между мостами
 - ✓ количество утерянных сообщений BPDU не превышает трех
 - ✓ время hello-таймера установлено равным 2 с
- Кроме того, контролирует период времени хранения данных в таблице коммутации после изменения активной топологии

3. Таймер максимального срока хранения записей в таблице (Max Age)

- Контролирует время хранения сообщений BPDU до их удаления
- Каждый порт сохраняет лучшее из полученных сообщений BPDU
- Пока каждые две секунды мост получает непрерывный поток сообщений BPDU, копия наиболее приемлемого из сообщений хранится мостом отдельно

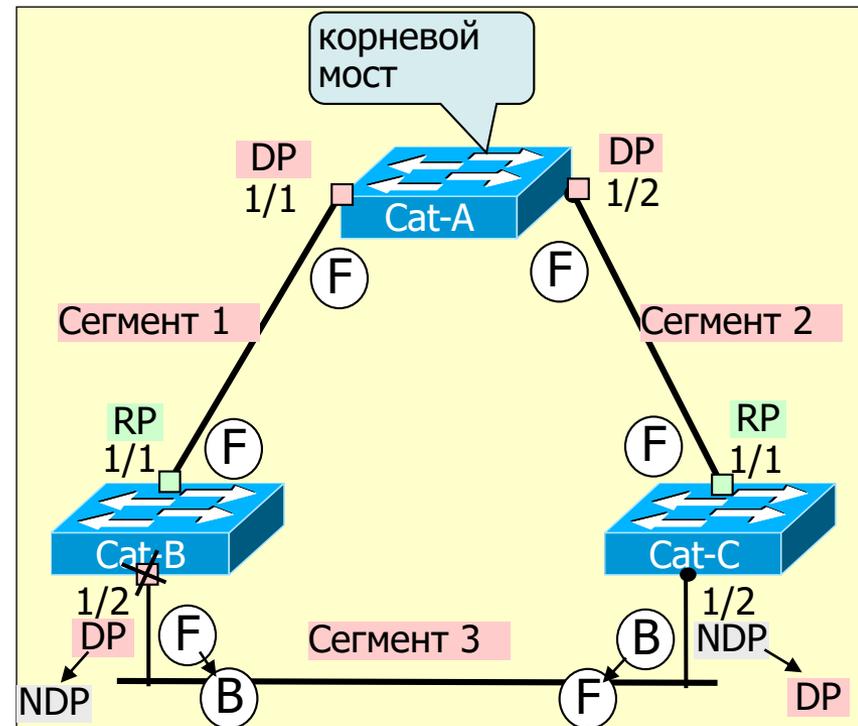
Таймеры протокола STP (3)

- **Максимальный диаметр сети при использовании мостов**
 - Max число соединения между любыми двумя конечными системами равно 7 при использовании по умолчанию переменных Hello Time, Max Age, Forward Delay
- **Предположим ситуацию, в которой устройство, отсылающее наиболее приемлемое сообщение, выходит из строя**
 - В протоколе STP существуют механизмы, позволяющие перехватывать функции неисправного устройства другим мостом

Таймеры протокола STP (4)

Предположим неисправность порта 1/2 Cat-B

- Через 2 секунды Cat-C зафиксирует отсутствие соответствующих BPDU
- Через 20 с (Max Age) у Cat-C устареет информация о Cat-B
 - порт 1/2 Cat-C перейдет в состояние прослушивания, чтобы попытаться присвоить себе статус назначенного порта
 - Порт 1/2 Cat-C наиболее приемлем для доступа к сегменту 3, становится назначенным, переходит в состояние передачи данных
- В этом случае сходимость STP-протокола равна 50 с
 - ✓ 20 с максимального времени хранения + 15 с в состоянии прослушивания + 15 с в состоянии изучения топологии



Обозначения:

RP - корневой порт
DP - назначенный порт
NDP - Не назначенный порт

Статус порта

(F) - передача данных
(B) - блокировка

Состояние порта

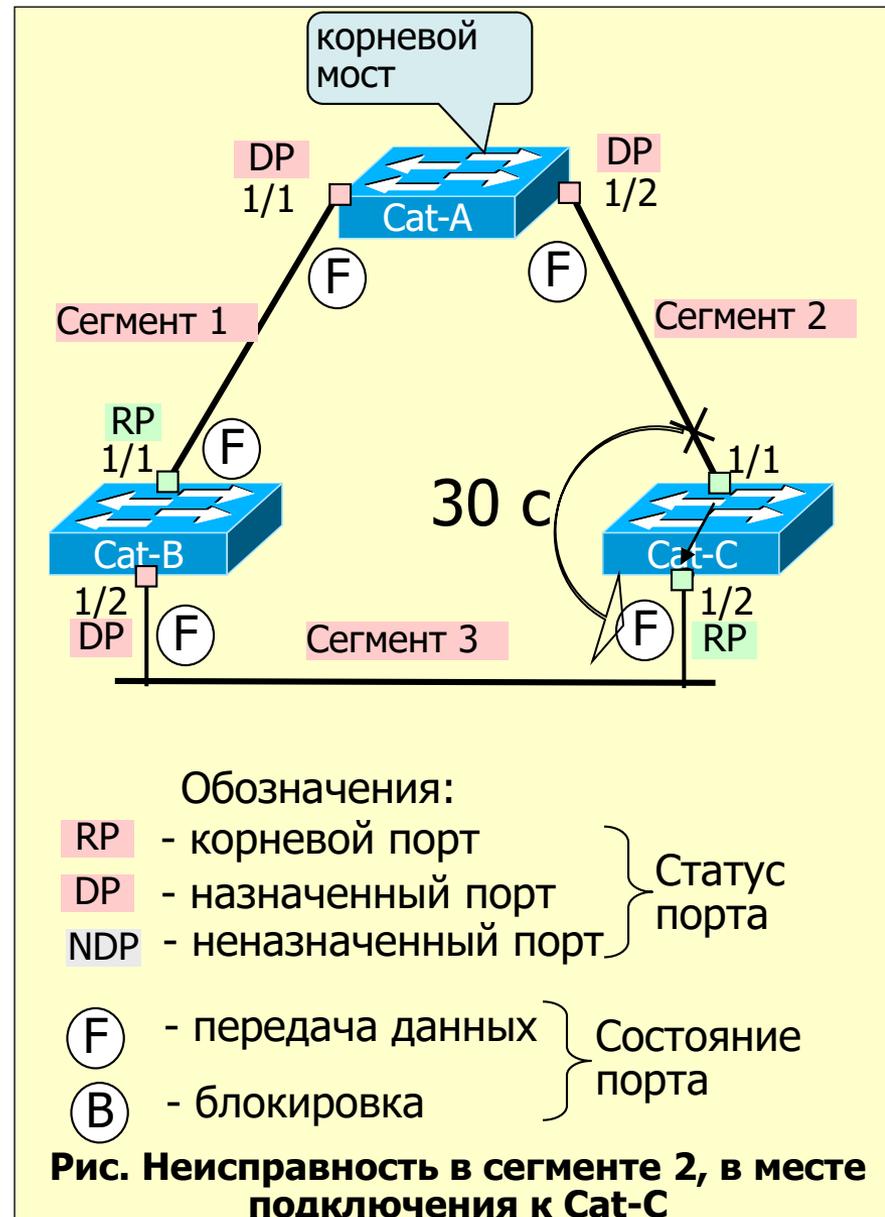
Рис. Неисправность порта 1/2 Cat-B

Таймеры протокола STP (5)

Неисправность в сегменте 2, в месте подключения к корневому порту Cat-C

Случай, когда мост может сразу перейти в состояние прослушивания без ожидания в течение 20 с

- Поскольку неисправность в сегменте с корневым портом, необходимость ожидать 20 с отпадает
- через 2 с порт 1/2 Cat-C переходит в состояние прослушивания в попытке назначить себя корневым портом
- Таким образом время сходимости STP-протокола уменьшается с 50 с до 30 с (15 с состояния прослушивания + 15 с состояния изучения топологии)



Таймеры протокола STP (6)

- **Стандартное значение времени сходимости протокола STP колеблется от 30 до 50 с.**
 - Существуют методы уменьшения времени сходимости
- **Важно: есть два ключевых правила для работы с таймерами протокола STP**
 - изменять стандартное значение таймера необходимо только в особых случаях
 - даже если возникла потребность в настройке таймеров, модифицировать значения необходимо только на корневом мосту
- **Пояснения**
 - Сообщения BPDU переносят значения временных параметров от корневого моста к остальным мостам в сети в трех полях
 - Если каждый мост в сети будет конфигурироваться отдельно, то может возникнуть ситуация, когда некоторые мосты могут перейти в состояние передачи данных, в то время как остальные еще находятся в состоянии прослушивания — т.е. сеть полностью дестабилизируется

Команда show spantree

Пример 6.1. Выполнение команды show spantree на коммутаторе Cat-B для модели сети на рис. 6.6

```

Cat-B (enable) show spantree
VLAN 1
Spanning tree enabled
Spanning tree type      ieee
                          [ ] Глобальная статистика

Designated Root          aa-aa-aa-aa-aa-aa
Designated Root priority 32768
Designated Root Cost     19
Designated Root port     1/1
Root Max Age 20 sec  Hello Time 2 sec  Forward Delay 15 sec
                          [ ] Статистика корневого моста

Bridge ID MAC ADDR       bb-bb-bb-bb-bb-bb
Bridge ID priority       32768
Bridge Max Age 20 sec  Hello Time 2 sec  Forward Delay 15 sec
                          [ ] Статистика локального моста

Port  Vlan Port-State  Cost  priority Fast-Start Group-method
-----
1/1   1   forwarding   19    32   disabled
1/2   1   forwarding   19    32   disabled
                          [ ] Статистика по портам
    
```

Значения таймеров, представленные в локальной статистике моста, не используются, пока мост не получит статус корневого

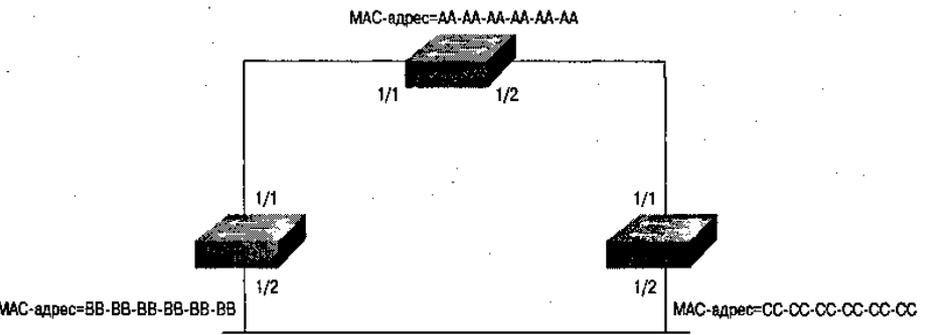


Рис. 6.6. Модель сети для дальнейших дискуссий по основам протокола STP

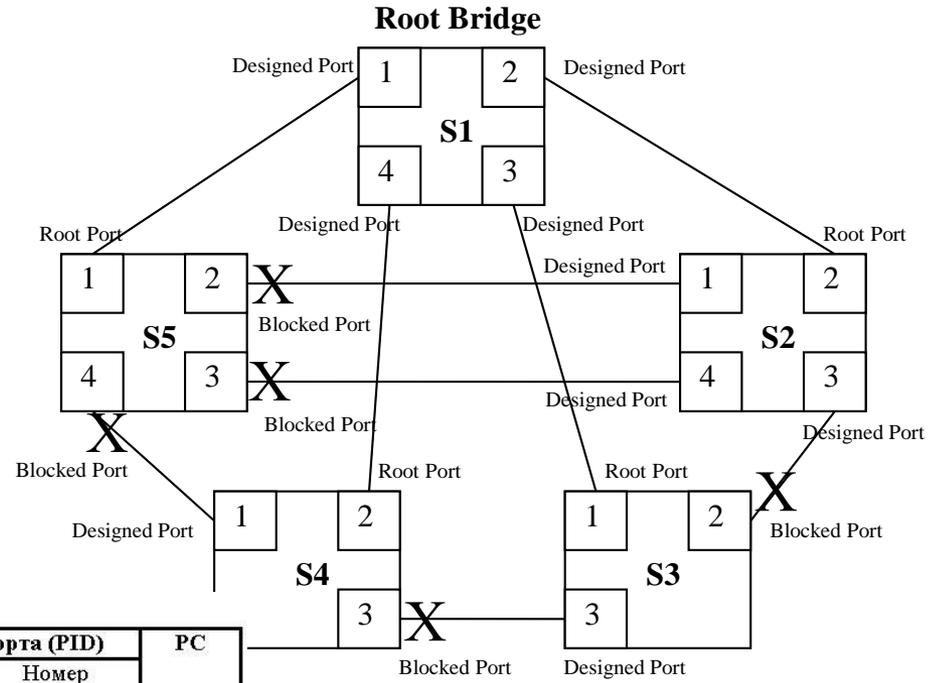
7. ПРОТОКОЛЫ RSTP, RAPID-PVST+

Rapid spanning tree protocol

RSTP, Rapid-PVST+

- **Время сходимости STP от 30 до 50 секунд для современных сетей неприемлемо.**
- **Для решения этой проблемы разработан протокол RSTP (802.1w).** Стандарт 802.1w позже был включен в 802.1D – 2004. RSTP обеспечивает более быструю сходимость сети по сравнению с STP.
- **Протокол Rapid-PVST+ строит дерево RSTP для каждого vlan.**
- **Существенное отличие STP и RSTP является способ перехода портов в состояние пересылки (Forwarding) и то каким образом этот переход влияет на роль порта в топологии. RSTP объединяет состояния Disabled, Blocking, Listening в одно Discarding, в котором порт не активен. Состояния портов в RSTP и STP представлены в Табл.3.**

Пример лабораторной работы



По коммутаторам:

Коммутатор	Идентификатор моста (BID)		Идентификатор порта (PID)		PC
	Приоритет (Bridge Priority)	MAC-адрес блока управления	Приоритет (Port Priority)	Номер (Port Number)	
S1	32768	00-11-5B-C6-E6-C3	32	1	19
			32	2	19
			32	3	19
			32	4	19
S2	32768	00-11-5B-C6-E6-C4	32	1	19
			32	2	19
			32	3	19
			32	4	19
S3	32768	00-11-5B-C6-E6-C5	32	1	19
			32	2	19
			32	3	19
S4	32768	00-11-5B-C6-E6-C6	32	1	19
			32	2	19
			32	3	19
S5	32768	00-11-5B-C6-E6-C7	32	1	19
			32	2	19
			32	3	19
			32	4	19

1. Кларк К, Гамильтон К. Принципы коммутации в локальных сетях Cisco. : Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 976 с. ISBN 5-8459-0464-1 (рус)