

# Протоколы маршрутизации

Внутренние и внешние протоколы  
маршрутизации

Протоколы RIP, OSPF, EGP, BGP

Фильтрация трафика и объявлений  
маршрутов

# Сети – совокупность подсетей. Можно представить графом.

- Сеть в общем случае рассматривается как **совокупность подсетей соединяемых между собой маршрутизаторами**, такая сеть называется составной сетью или интерсетью, internet. Компонентами интерсети могут быть как локальные (ethernet, fddi) так и глобальные сети (ISDN, x.25). Такие сети могут организовать работу и обмен между узлами своей сети, но не могут связать узлы разных сетей, за это отвечает сетевой уровень. **Сетевой уровень отсылает между узлами пакеты**. При этом **пакеты в локальных сетях оборачиваются в кадры** локальной сети, где они проходят, затем снова разворачиваются если при изменении сети, инкапсулируясь в кадр новой сети, в сетях где возможна транспортировка на более высоком уровне, данные пересылаются в пакетах более высоких уровней. В отличие от коммутаторов, маршрутизаторы имеют свой локальный адрес в сети.

# Адресация в сетях

- Для адресации узлов в интерсети очевидно необходимо ввести адрес в виде — **номер подсети и номер узла в подсети**. Один из способов использует в качестве адресов машин в подсети локальные адреса принятые в данной локальной сети, физические **mac** адреса — например, протокол `ipx`, другой способ использует собственную адресацию, независимую от локальных адресов — например, протокол `ip`.

# Алгоритмы маршрутизации

- Для маршрутизации применяются специальные алгоритмы и протоколы (например, RIP, OSPF, NLSP). С помощью протоколов маршрутизации создается карта сети или специальные таблицы маршрутизации, которые хранятся на каждом маршрутизаторе.
- Одношаговые алгоритмы маршрутизации определяют только следующий маршрутизатор в сети для отправки пакета, каждый маршрутизатор ответственен за выбор одного шага маршрута.
- В многошаговых алгоритмах узел источник задает полный маршрут следования пакета через промежуточные маршрутизаторы.

# Одношаговые алгоритмы делятся

## на:

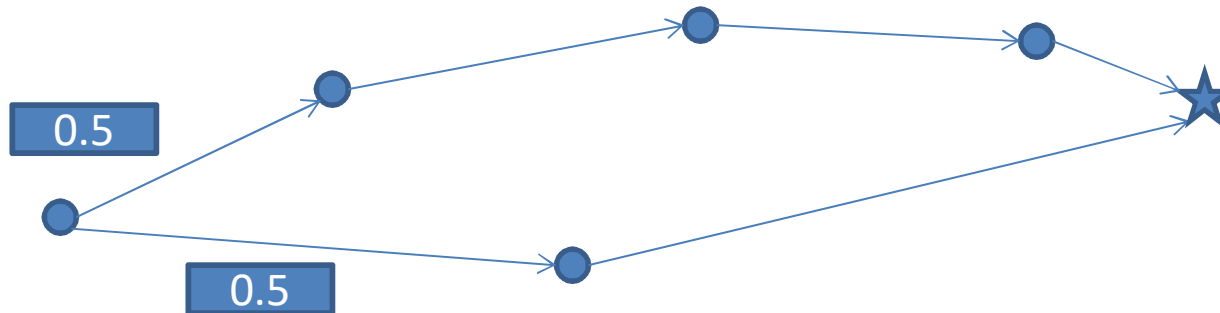
- Алгоритмы **фиксированной маршрутизации**:
- Все записи в таблице маршрутизации являются статическими. Администратор сети сам решает, на какие маршрутизаторы надо передавать пакеты с теми или иными адресами, например, с помощью команды `route` заносит записи в таблицу маршрутизации.
- Различают **одномаршрутные таблицы**, где всякий раз выбирается один и тот же маршрут для пакетов с одним адресом, в **многомаршрутных** могут определяться несколько путей, обычно это резервные пути, фиксированная маршрутизация подходит для сетей с не очень сложной топологией, например в магистральных каналах.
- Алгоритмы **простой маршрутизации**:
- В этом случае таблица маршрутизации вообще не строится, либо строится без участия маршрутизаторов.

# Бывает несколько видов простой маршрутизации:

- **Случайная маршрутизация** — пакет отсылается в любом случайном направлении, исключая направление с которого он поступил.
- **Лавинная маршрутизация** — пакет широковещательно отсылается по всем направлениям кроме исходного.
- **Маршрутизация на основе предыдущего опыта** — выбор маршрута осуществляется по таблице, таблица строится как в алгоритме настройки отсылки кадров прозрачными мостами и коммутаторами — на основе адресов появляющихся на входных портах.
- **Алгоритмы адаптивной (или динамической) маршрутизации.**
- Обеспечивают автоматическое обновление таблиц маршрутизации после изменения конфигурации сети. В таблицах маршрутизации для каждого маршрута обычно хранится время в течение которого данный маршрут может считаться действительным — время жизни маршрута. Адаптивные алгоритмы должны обеспечивать если не оптимальность то рациональность маршрута при достаточной скорости определения маршрутов и построения таблицы, при этом система построения маршрутов является децентрализованной — распределённой.

# Муравьиный алгоритм

- Муравей (пакет) выбирает случайное направление движения, и когда доходит до цели (пищи, адреса назначения) идет обратно оставляя «феромон», который усиливает вероятность выбора направления, со временем феромон испаряется, фактически значение веса выбора маршрута уменьшается.



Возвращаясь по более длинному пути шанс увеличить вес ребра меньше чем возвращаясь по короткому пути. И наоборот шанс увеличить вес ребра выше возвращаясь по короткому пути. Пока один муравей возвращается, на ребре с длинным путем вес уменьшится сильнее.

Адаптивные алгоритмы делятся на алгоритмы дистанционно векторного типа и алгоритмы состояния связей.

Протоколы обмена маршрутной информацией

дистанционно-векторные алгоритмы

*(Distance Vector Algorithms, DVA)*

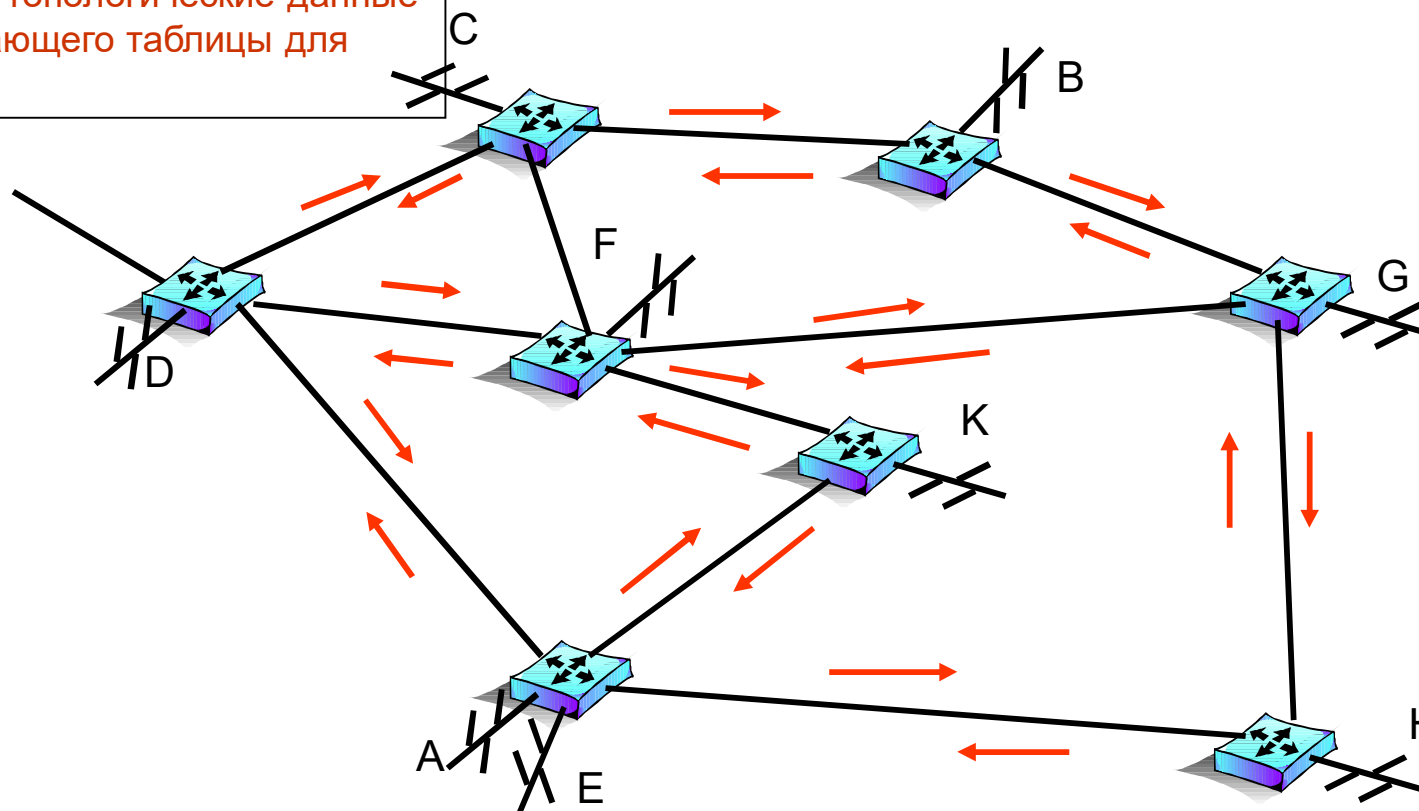
алгоритмы состояния связей

*(Link State Algorithms, LSA)*



Распределенный подход – нет центрального элемента, собирающего топологические данные и вырабатывающего таблицы для всех

## Постановка задачи

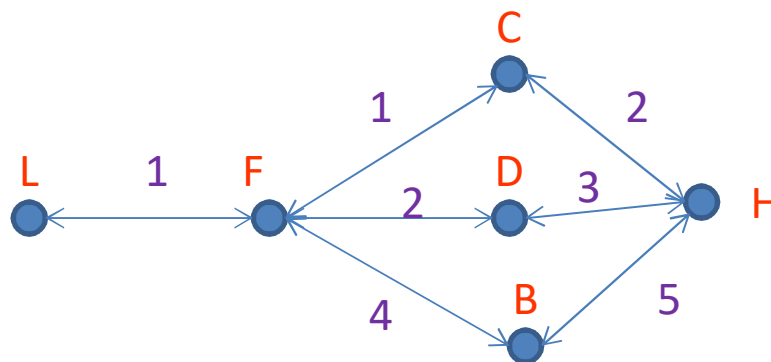


С помощью служебного протокола маршрутизаторы должны автоматически:

- построить свои таблицы маршрутизации (Net - Next Hop)
- постоянно поддерживать соответствие таблицы текущей топологии сети – добавление и удаление связей и маршрутизаторов. Пример где адресуются сети (интерфейсы маршрутизаторов).

# Дистанционно-векторные алгоритмы

## Distance-Vector Algorithms, DVA



Пример где адресуются маршрутизаторы.

L отключается. F шлет до C, D, B что L отключился. Они убирают запись об L. Но H успевает отправить свою таблицу раньше, чем C до H. Эта таблица приходит к C, D, B. И что же случается?

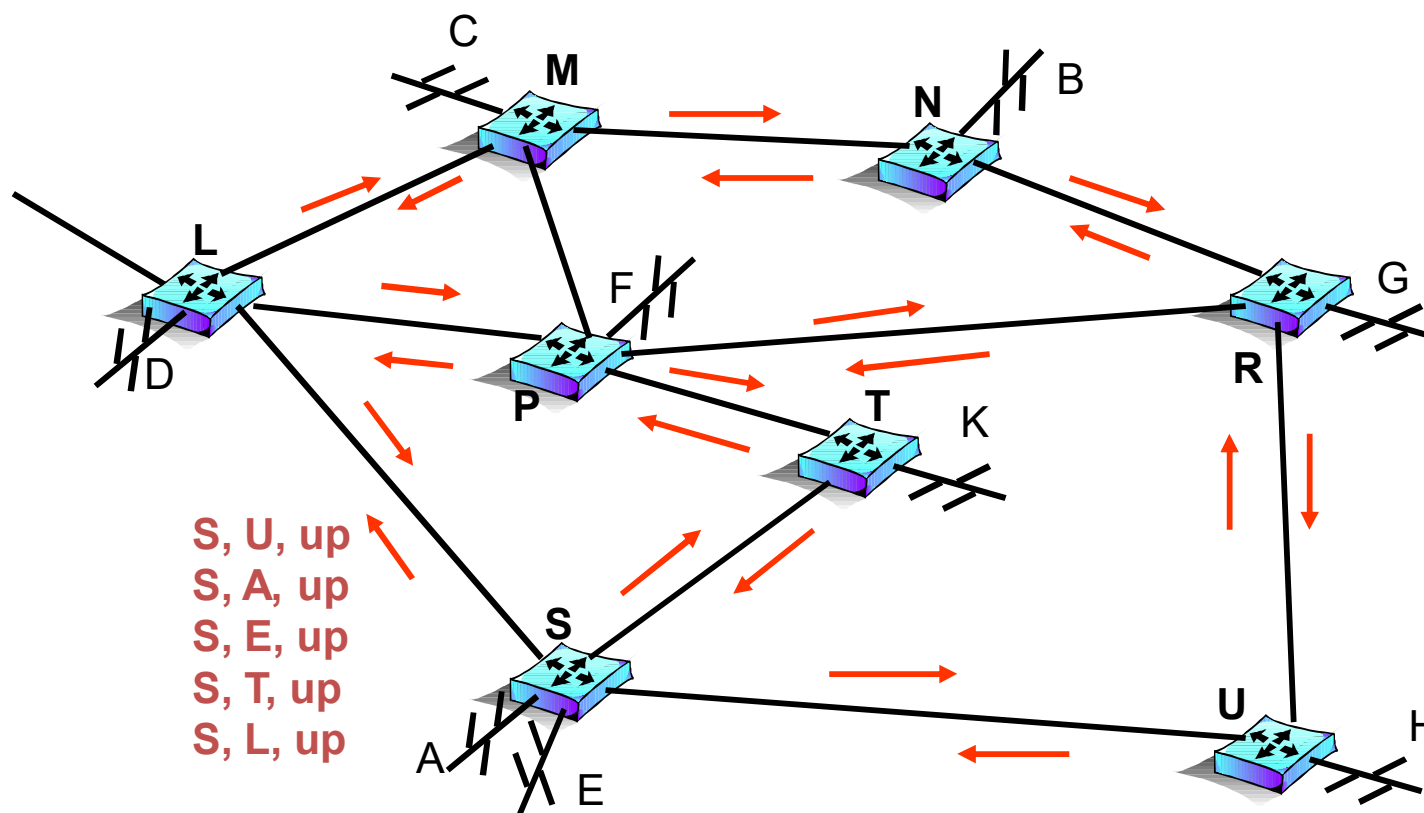
Что будет после того как D подумав что у него есть путь до L через H отправит таблицу до F, а тот в свою очередь до C. Очевидно что и C теперь будет думать о пути до L, хотя ранее не принимал этого от H.

1. Каждый маршрутизатор рассылает всем непосредственным соседям вектор расстояний до известных ему сетей. Расстояние (метрика) – в хопах; или в условных единицах, учитывающих пропускную способность каналов связи
2. Получив векторы от соседей, маршрутизатор добавляет новую информацию к своей базе данных о известных сетях – расстояние наращивается на 1 (или метрику канала связи) . Если к одной и той же сети можно попасть разными маршрутами, то выбирается один – с минимальным расстоянием
3. Процесс обмена векторами расстояний периодически повторяется – отслеживается динамика изменения топологии сети

## ***Алгоритм дистанционно-векторного типа***

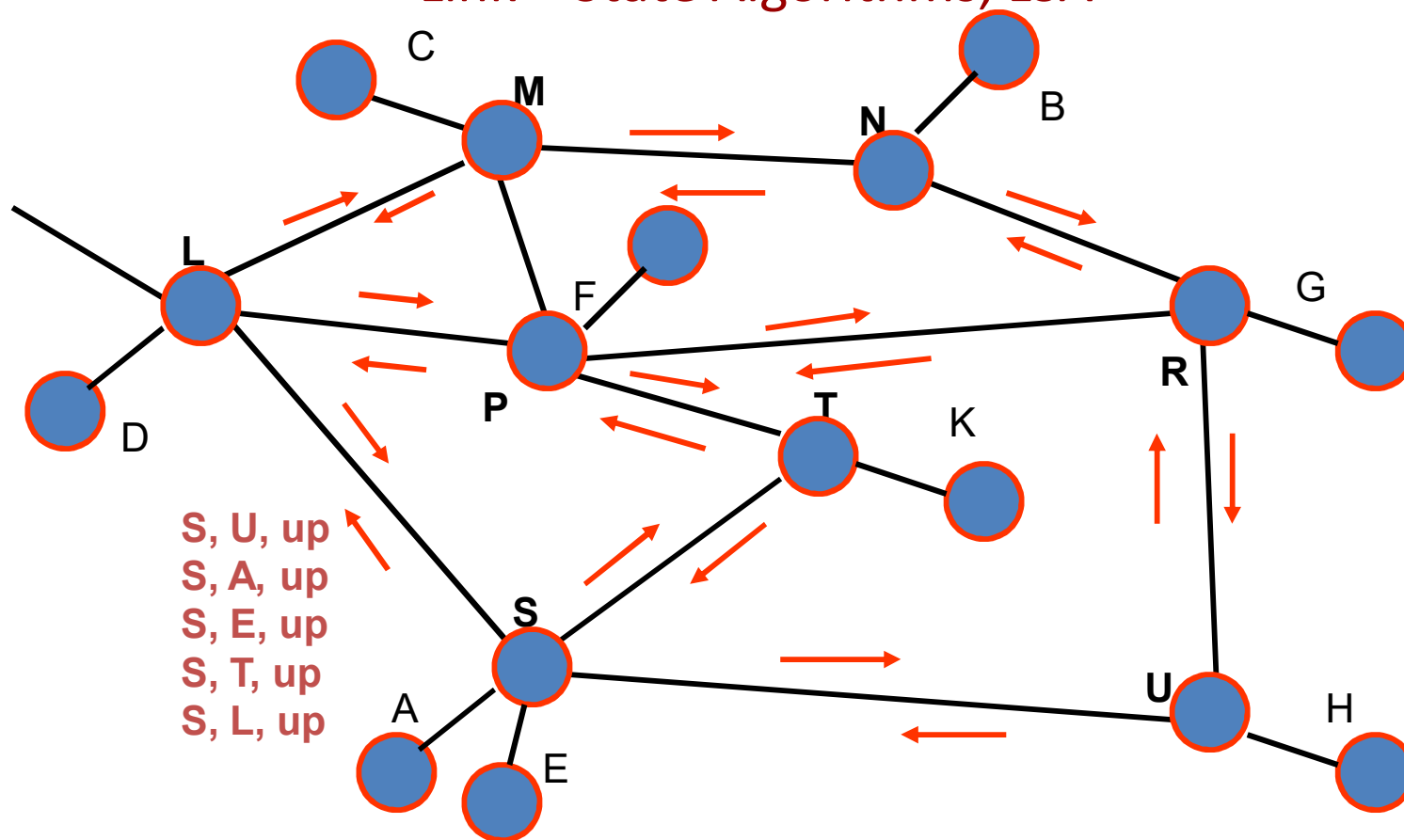
- Хорошо работают только в небольших сетях
- Засоряют линии связи интенсивным широковещательным трафиком
- Изменения конфигурации могут обрабатываться по этому алгоритму не всегда корректно — информация из «вторых» рук может уже устареть и «обмануть»
- Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол RIP

## Алгоритмы состояния связей Link – State Algorithms, LSA



1. Маршрутизаторы обмениваются информацией о топологии сети для построения точного графа.  
Вершины – маршрутизаторы и сети (A, B, C, ..., S, T, U). Передается информация о связях между вершинами графа. **Пример где адресуются сети.**

## Алгоритмы состояния связей Link – State Algorithms, LSA

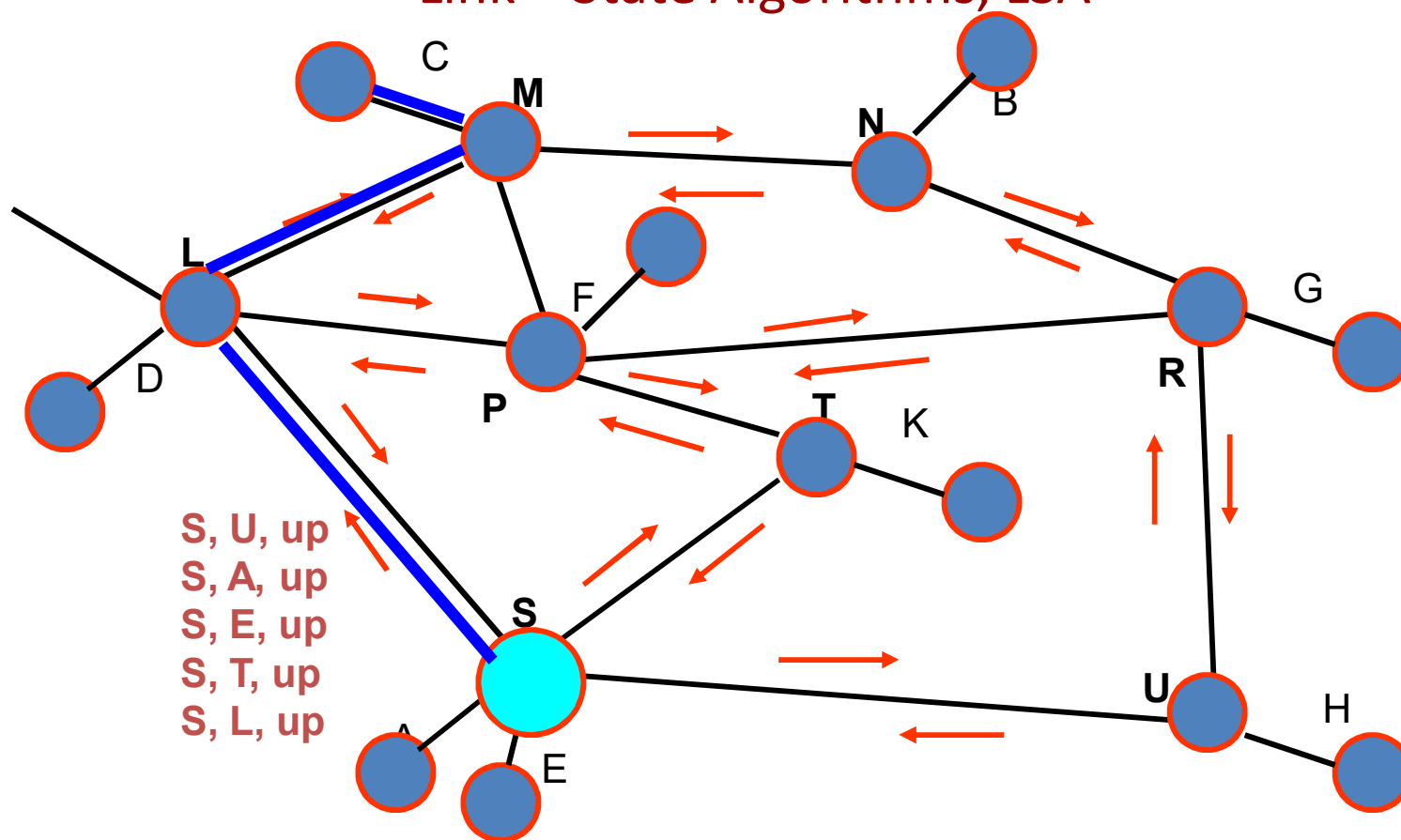


Каждый маршрутизатор через несколько итераций обмена располагает базой топологической информации – графом сети. Графы всех маршрутизаторов идентичны.

Новые топологические объявления – только при изменении состоянии связи (up -> down)

Периодическое тестирование связи – hello. Пример где адресуются маршрутизаторы.

## Алгоритмы состояния связей Link – State Algorithms, LSA



Каждый маршрутизатор на основе графа строит свою таблицу маршрутизации:

- находит от себя кратчайший маршрут до каждой сети
- запоминает из этого маршрута только следующий хоп

## ***Алгоритм состояния связей***

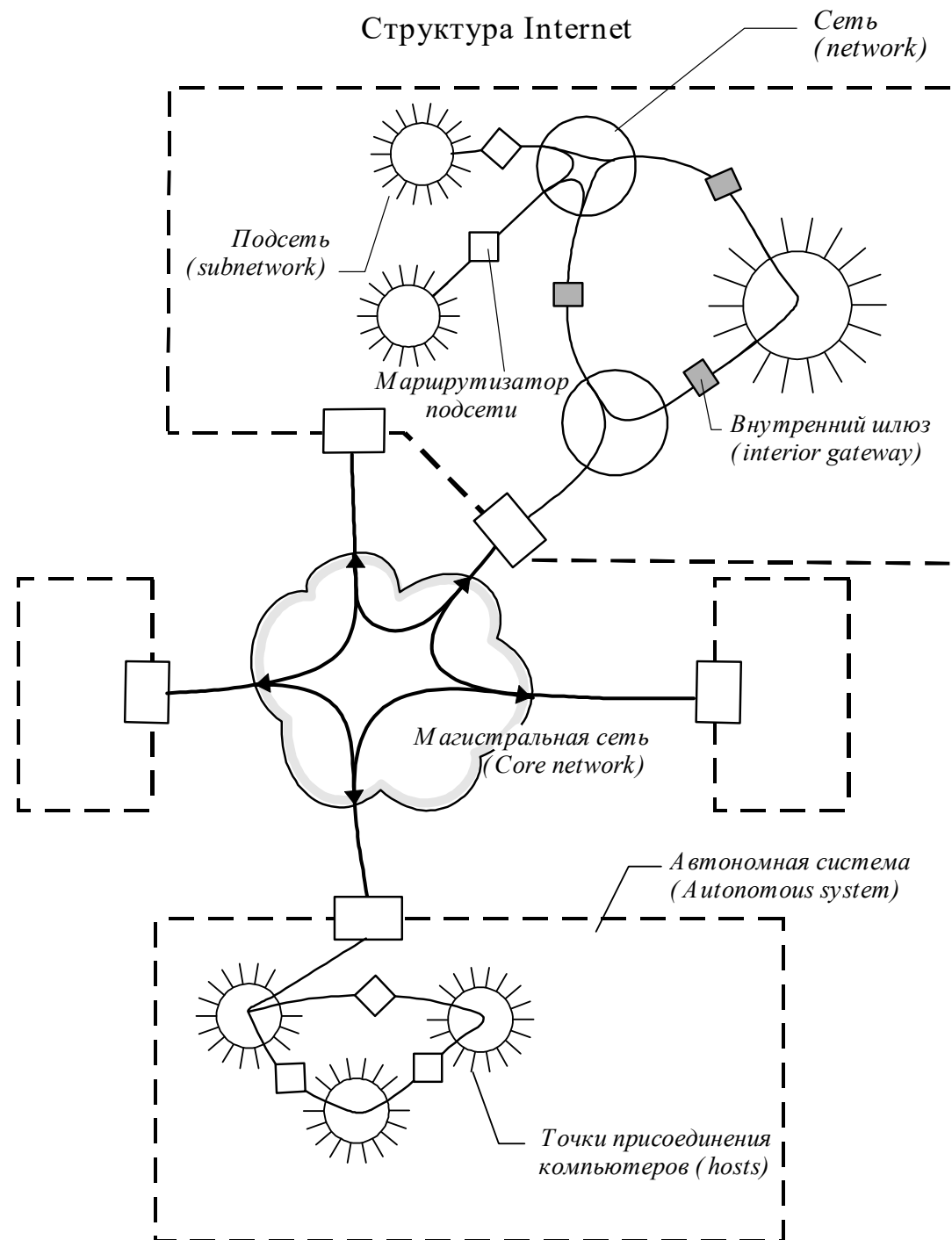
- Периоды нестабильной работы сети при изменении топологии существенно короче, чем у алгоритмов DVA
- Вычисления кратчайшего маршрута существенно сложнее – нагрузка на маршрутизатор растет
- Служебный трафик – меньше
- **Open Shortest Path First** – протокол типа LSA в стеке IP

## Внутренние и внешние протоколы маршрутизации

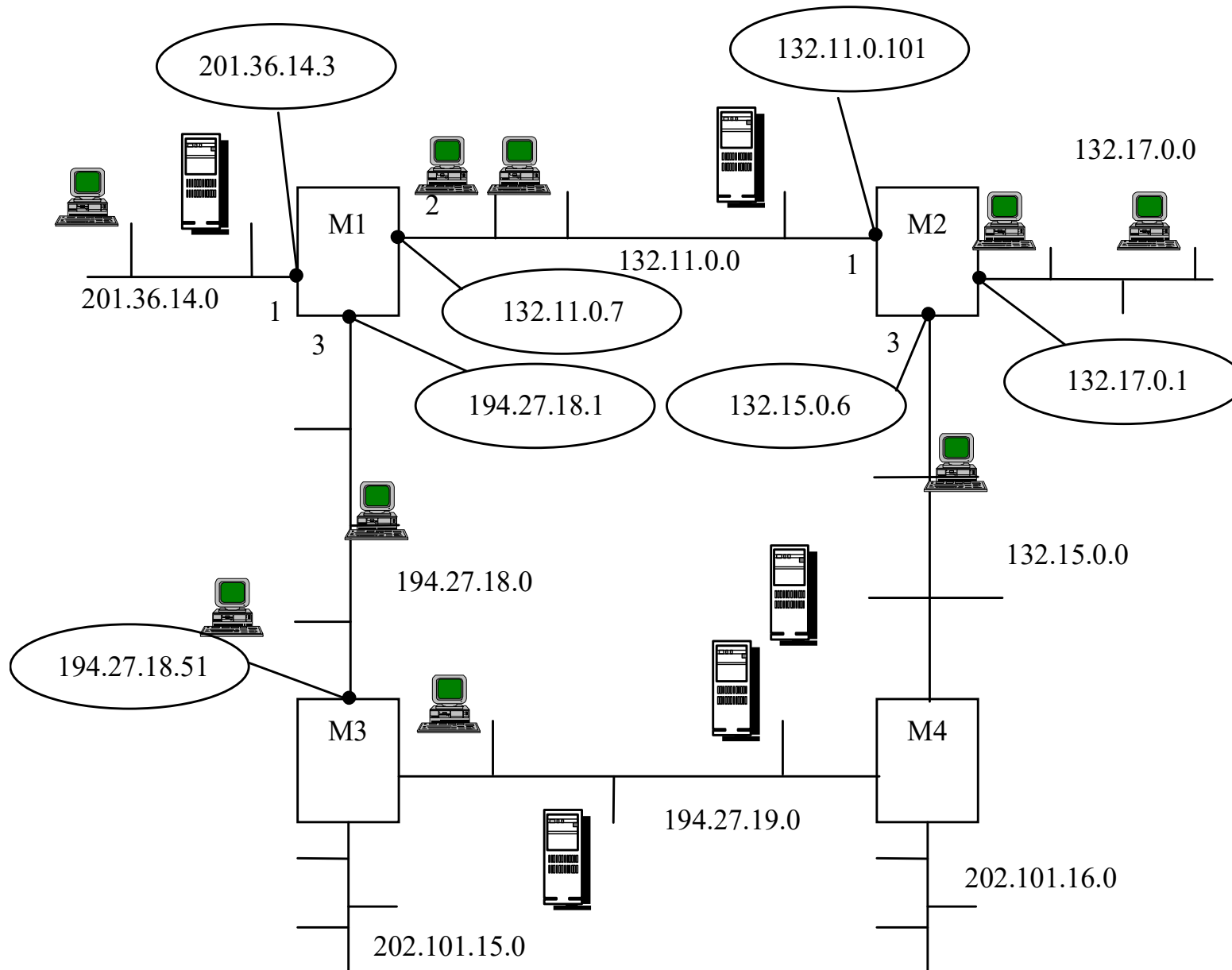
### ***Структура и терминология сети Internet:***

- магистральная сеть (core backbone network)
- автономные системы (autonomous systems)
- протоколы внутренних шлюзов (interior gateway protocol, **IGP**)
- протоколы внешних шлюзов (exterior gateway protocol, **EGP**)
- таблица достижимости (reachability table)





# Протокол маршрутизации RIP (Routing Information Protocol)



***Начальное состояние таблицы маршрутизации в M1:***

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1

***Начальное состояние таблицы маршрутизации в M2:***

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
132.11.0.0	132.11.0.101	1	1
132.17.0.0	132.17.0.1	1	1
132.15.0.0	132.15.0.6	1	1

RIP-объявления делаются каждым маршрутизатором каждые 30 секунд (То**б**ъявл.)

## Протокол маршрутизации RIP (продолжение)

**Таблица маршрутизации в M1 после одного шага:**

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
194.27.19.0	194.27.18.51	3	2
202.101.15.0	194.27.18.51	3	2

## Правила построения таблицы маршрутизации по RIP

- В таблице для каждой сети остается только одна запись с минимальным расстоянием
- Каждая запись имеет срок жизни  
 $6 \times \text{Тобъявл.} = 180 \text{ сек} = 3 \text{ мин}$
- Максимальное количество хопов в поле расстояния - 15
- 16- признак недостижимой сети
- Объявления о имеющейся в таблице сети, но с худшей метрикой учитываются только от того маршрутизатора, на основании объявления которого была сделана запись

# Протокол маршрутизации RIP

## (продолжение 2)

**Таблица маршрутизации в M1 после двух шагов:**

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
<del>132.15.0.0</del>	<del>194.27.18.51</del>	<del>3</del>	<del>3</del>
194.27.19.0	194.27.18.51	3	2
<del>194.27.19.0</del>	<del>132.11.0.101</del>	<del>2</del>	<del>3</del>
202.101.15.0	194.27.18.51	3	2
202.101.16.0	132.11.0.101	2	3
<del>202.101.16.0</del>	<del>194.27.18.51</del>	<del>3</del>	<del>3</del>

# Формат сообщения RIP

- Протокол RIP передает данные о номерах сетей в пакетах UDP (порт 520).
- В одном пакете RIP может передаваться до 25 номеров сетей.

Пакет RIP имеет следующий формат:

0	8	16	31
Команда	Версия	0 ...0	
Тип адресной информации (IP=2)		0...0	
IP адрес			
0...0			
0...0			
Метрика			

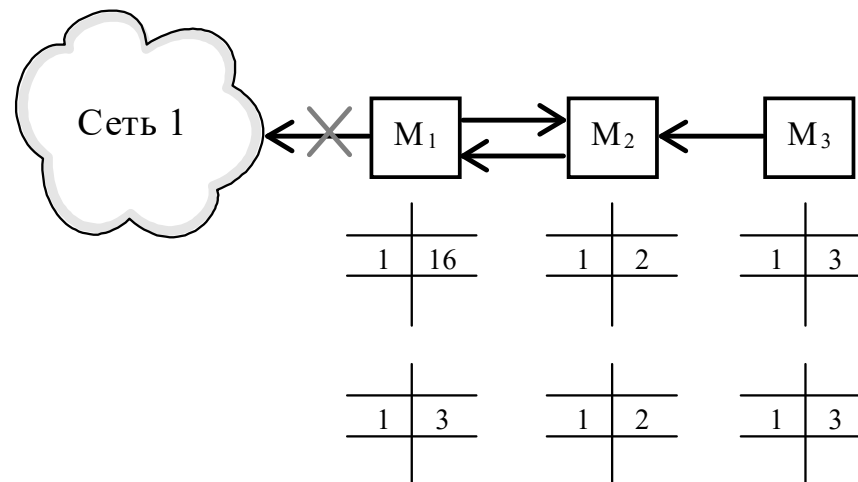
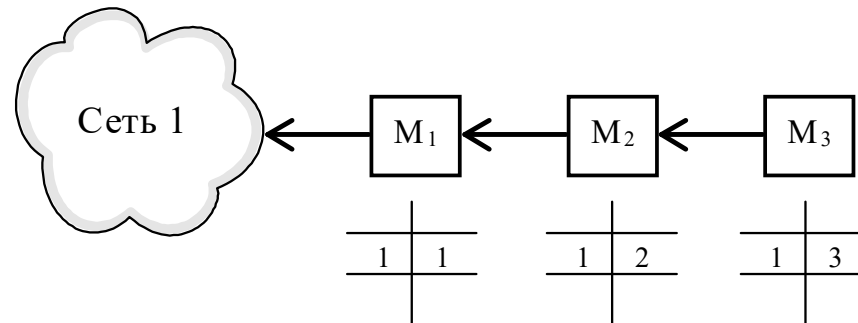
Протокол RIP использует две команды:

- 1 - запрос о посылке всей или части таблицы маршрутизации
- 2 - сообщение, содержащее всю или часть таблицы маршрутизации

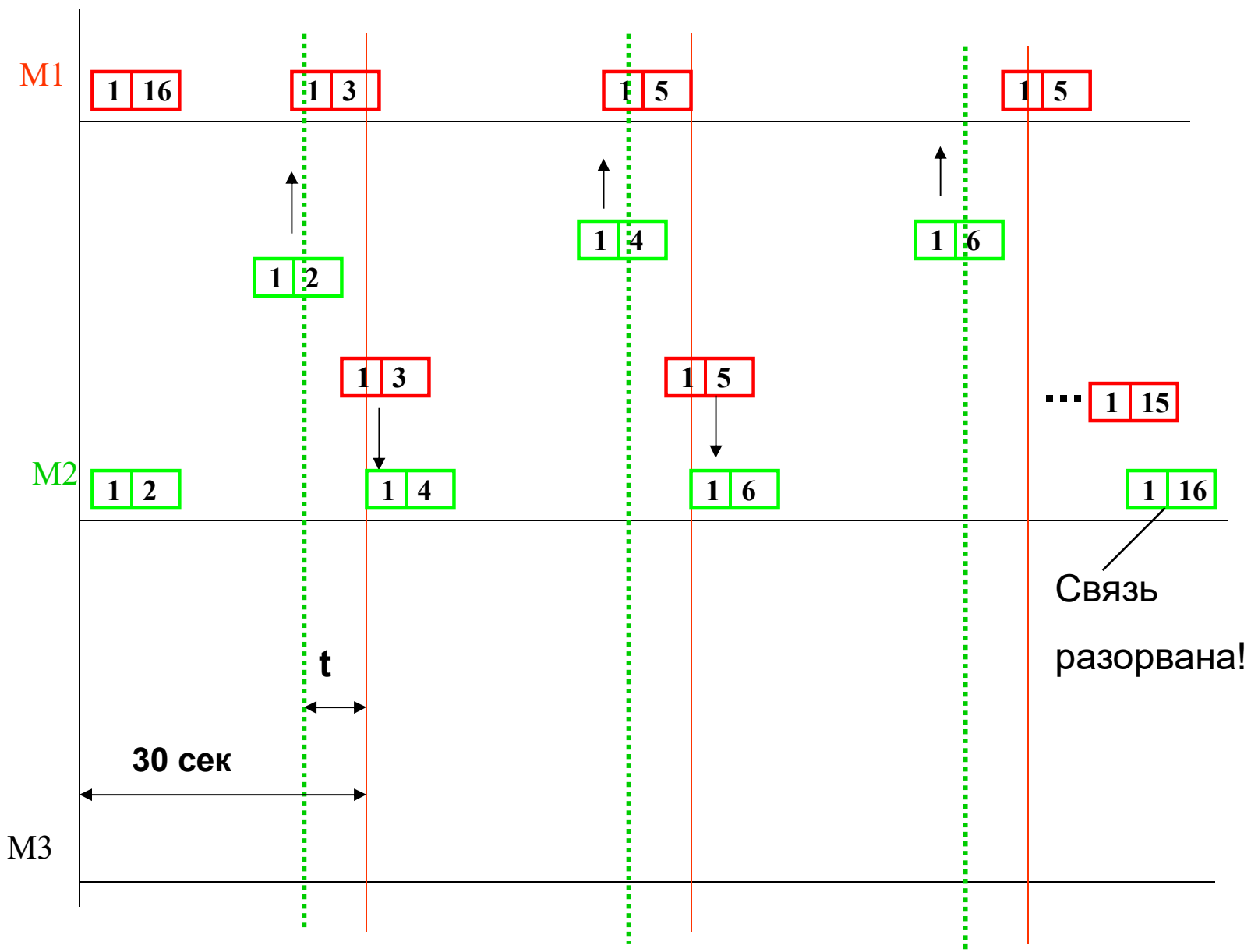
# Нестабильность работы RIP-маршрутизаторов при отказе связей

Нестабильность работы - заикливание пакетов, а также отправка пакетов по нерациональным маршрутам

Заикливание пакетов в маршрутных петлях:







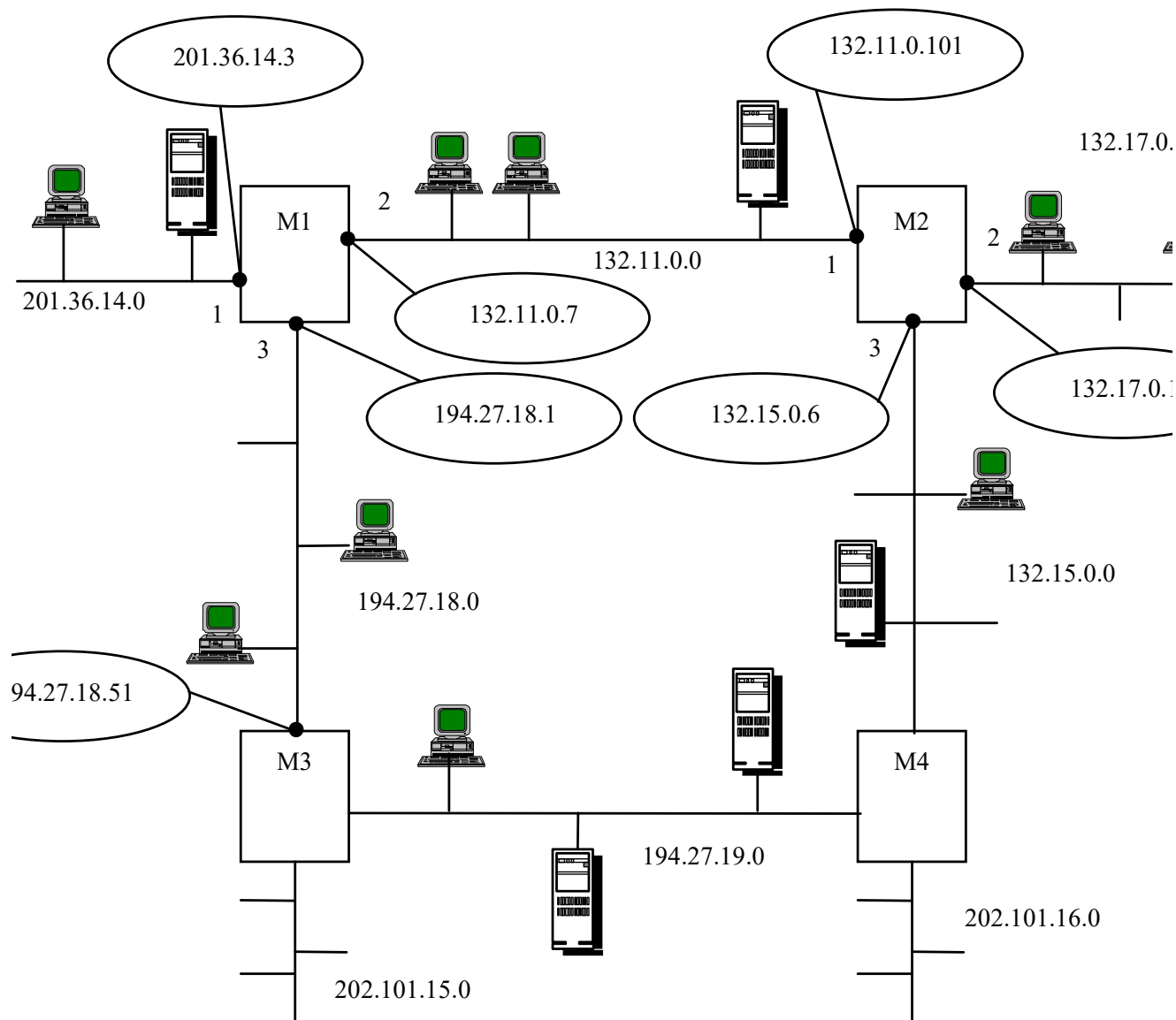
Время существования петли =  $(16/2-1) \times 30 = 210 \text{ с} = 3,5 \text{ мин}$

# **Методы борьбы с ложными маршрутами в протоколе RIP**

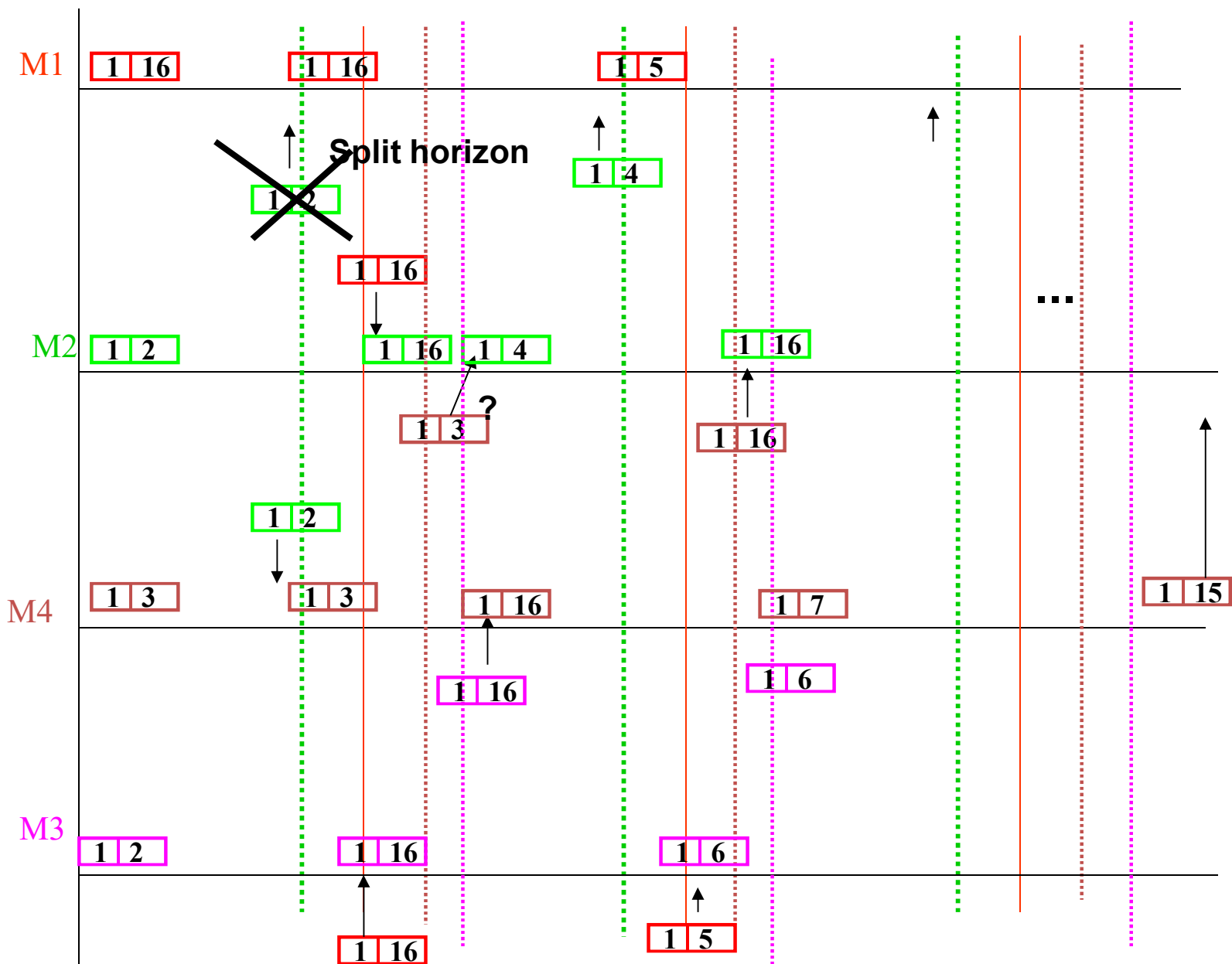
## **1) Split horizon - "расщепление горизонта"**

**Маршрутизатор никогда не возвращает данные о  
некоторой сети тому маршрутизатору, от которого узнал о  
этой сети**

- ♦ Защищает от заикливания пакетов в петлях, образованных соседними маршрутизаторами
- ♦ Не защищает от заикливания пакетов в маршрутных петлях, образованных 3-мя и более маршрутизаторами



При отказе связи M1 с сетью 201.36.14.0 информация о достижимости этой сети вернется в M1 через цепочку M2 - M4 – M3.



М4 узнал о  
1 от М3

Петля существовала  $4 \times 30 = 120$  сек

# **Методы борьбы с ложными маршрутами в протоколе RIP**

## **2) Triggered updates - триггерные обновления**

**При изменении состояния связи маршрутизатор немедленно делает объявление, не ожидая периода в 30 сек**

**Небольшая задержка (1-5 сек) – возможность решения проблемы на нижних уровнях**

- ♦ В многих случаях предотвращает использование устаревшей информации из вторых рук - быстро заменяет ее новой во всех маршрутизаторах сети
- ♦ Возможны сбои - когда регулярное объявление с устаревшей информацией опережает триггерное объявление

### **3) Hold down - "замораживанием изменений"**

**Вводится тайм-аут на принятие новых данных о сети, которая только что стала недоступной.**

- ♦ Тайм-аут предотвращает принятие устаревших сведений о некотором маршруте от тех маршрутизаторов, которые находятся на некотором расстоянии от отказавшей связи и передают устаревшие сведения о ее работоспособности
- ♦ Хорошо сочетается с триггерными объявлениями

Poison reverse

## Route Poisoning



# Протокол RIP v. 2

## ***Новые свойства:***

- ♦ поддержка масок подсетей
- ♦ аутентификация маршрутизаторов

## Формат сообщений RIP v.2:

Если "Тип адресной информации" = 0xFFFF, то в первой записи вместо маршрутной информации передается 16-байтовый пароль для аутентификации

"Метка маршрута" - для номера автономной системы

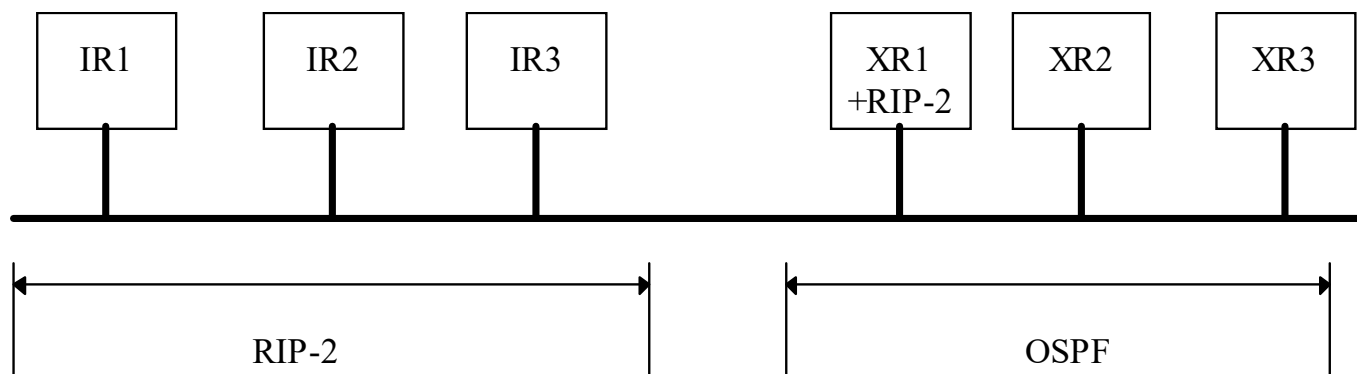
0	8	16	31
Команда	Версия	0 ...0	
Тип адресной информации (IP=2)		Метка маршрута	
IP адрес			
Маска подсети			
Следующий хоп			
Метрика			

# **Правила взаимодействия маршрутизаторов RIP-2 с маршрутизаторами RIP-1**

- информация, внутренняя для некоторой сети никогда не должна передаваться в другую сеть
- не должны передаваться данные о суперсетях, то есть сетях, имеющих маску более короткую, чем ее "естественная" длина

## Поле "Следующий хоп" - связь RIP-2 с OSPF

- ♦ Поле "Следующий хоп" используется для указания адреса маршрутизатора, которому нужно передавать пакеты для сети, адрес которой передается в поле IP адрес.
- ♦ Позволяет задать рациональный маршрут через сеть с другим протоколом маршрутизации, например, OSPF



# Протокол IGRP (Interior Gateway Routing Protocol) компании Cisco Systems

Протокол IGRP - дистанционно-векторный протокол

Использует приемы split horizon, triggered updates и hold down

Использует дополнительные по сравнению с RIP механизмы:

Использование более тонкой метрики:  $(K1/B_e + K2 \cdot D) / R$ ,

где

**B<sub>e</sub>** - пропускная способность канала,

**D** - задержка, вносимая каналом,

**R** - коэффициент готовности канала

Использование алгоритма **Route Poisoning** для борьбы с устаревшей информацией, получаемой из вторых рук через петли (Маршруты, метрика которых увеличивается при очередном обновлении более чем на 10%, отбрасываются. Использование этого алгоритма позволяет отказаться от "замораживания изменений", а это ускоряет установление в сети новой топологии после обрыва связи или отказа маршрутизатора, при первых обновлениях некоторые реальные маршруты могут быть отброшены, но при следующем обновлении все равно будут добавлены)

## Настройка IGRP

Ниже приведен пример типичной конфигурации протокола IGRP на маршрутизаторе фирмы Cisco.

```
router igrp 12
timers basic 15 45 0 60
network 162.8.0.0
network 193.1.1.0
no metric holddown
metric maximum-hop 50
```

Первая строка определяет, что IGRP является протоколом маршрутизации для автономной системы номер 12. В большинстве организаций все маршрутизирующие устройства имеют одинаковые номера автономной системы. Протокол IGRP не допускает обмена обновлениями между маршрутизаторами с разными номерами автономной системы.

Вторая строка устанавливает следующие значения таймеров IGRP:

15 с - основной временной интервал, определяющий периодичность широковещательных рассылок регулярных сообщений об обновлении;

45 с - время, по истечении которого маршрут считается недействительным, если о нем не поступает никакой новой информации;

0 с - интервал, в течение которого после удаления маршрута запрещается принимать сообщения о его обновлении (таймер hold-down);

60 с - время, по истечении которого происходит удаление маршрута из маршрутной таблицы.

Третья и четвертая строки идентифицируют сети, непосредственно подключенные к данному устройству маршрутизации. Пятая строка отключает механизм Hold-Down (значит, после удаления маршрута сообщение о его обновлении может быть принято незамедлительно). Шестая строка предписывает удалять пакеты, если они прошли через 50 маршрутизаторов. С одной стороны, это значение (число маршрутизаторов) должно быть достаточно большим, чтобы поддерживать все допустимые маршруты вашей сети, а с другой - его желательно сделать меньше, чтобы ускорить процесс удаления пакетов, попавших в маршрутную петлю.

EIGRP

## Протокол маршрутизации OSPF

Каждый маршрутизатор имеет исчерпывающую  
информацию о топологии сети -

база данных топологической информации

Все маршрутизаторы имеют идентификаторы

Сети идентифицируются IP-адресами

### **Этапы работы протокола**

- построение топологической базы отдельного маршрутизатора:
- построение общего графа путем обмена топологическими базами с соседями (подобно RIP)
- построение таблиц маршрутизации
- поддержание топологической базы



## **1 этап - построение топологической базы отдельного маршрутизатора:**

- ♦ Маршрутизатор - присоединенная соседняя сеть  
*(ручное конфигурирование)*
- ♦ Маршрутизатор - соседний маршрутизатор, непосредственно присоединенный  
*(ручное конфигурирование)*
- ♦ Маршрутизатор - соседний маршрутизатор, присоединенный через соседнюю сеть  
*(для широковещательных сетей - автоматический опрос, для нешироковещательных - ручное конфигурирование)*

Для каждой связи дополнительно определяется 3 типа метрики

#### **4 этап - поддержание топологической базы**

- проверка состояния связей - обмен hello (10 сек)
- объявления о изменении состояния связи
- для надежности изредка обмен полными топологическими базами – раз в 30 мин

#### ***Дополнительные свойства:***

- аутентификация
- разбиение на OSPF-области
- внешние объявления

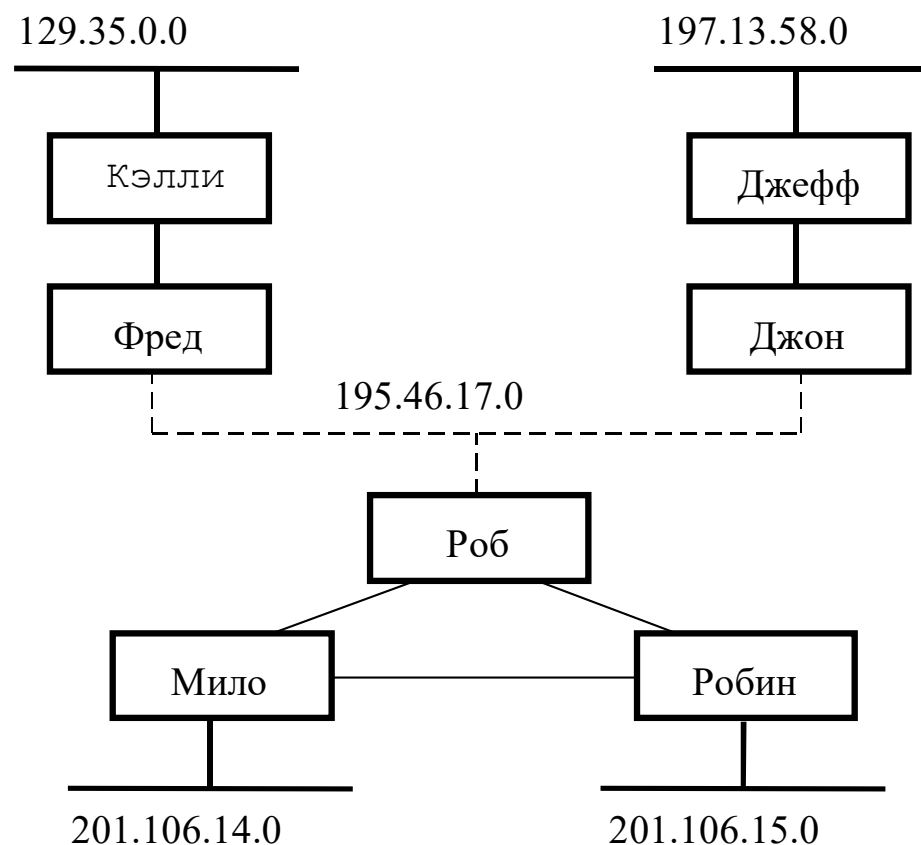
# Пример сети с маршрутизаторами OSPF

## *Связи типа "маршрутизатор - маршрутизатор":*

- Мило - Робин
- Мило - Роб
- Роб - Робин
- Кэлли - Фред
- Джеф - Джон

## *Связи типа "маршрутизатор - сеть":*

- Мило - 201.106.14.0
- Робин - 201.106.15.0
- Фред - 195.46.17.0
- Джон - 195.46.17.0
- Роб - 195.46.17.0
- Кэлли - 129.35.0.0
- Джеф - 197.13.58.0.



## Маски переменной длины в OSPF

- ♦ Протокол OSPF передает длину маски вместе с номером сети
- ♦ Поддерживаются маски переменной длины
- ♦ IP-подсети могут перекрываться - одна сеть является подмножеством адресов другой
- ♦ Если адрес принадлежит нескольким подсетям в таблице маршрутизации, то маршрутизатор использует наиболее специфический маршрут

# Метрика и оптимизация транзитных локальных сетей

В OSPF три категории сетей :

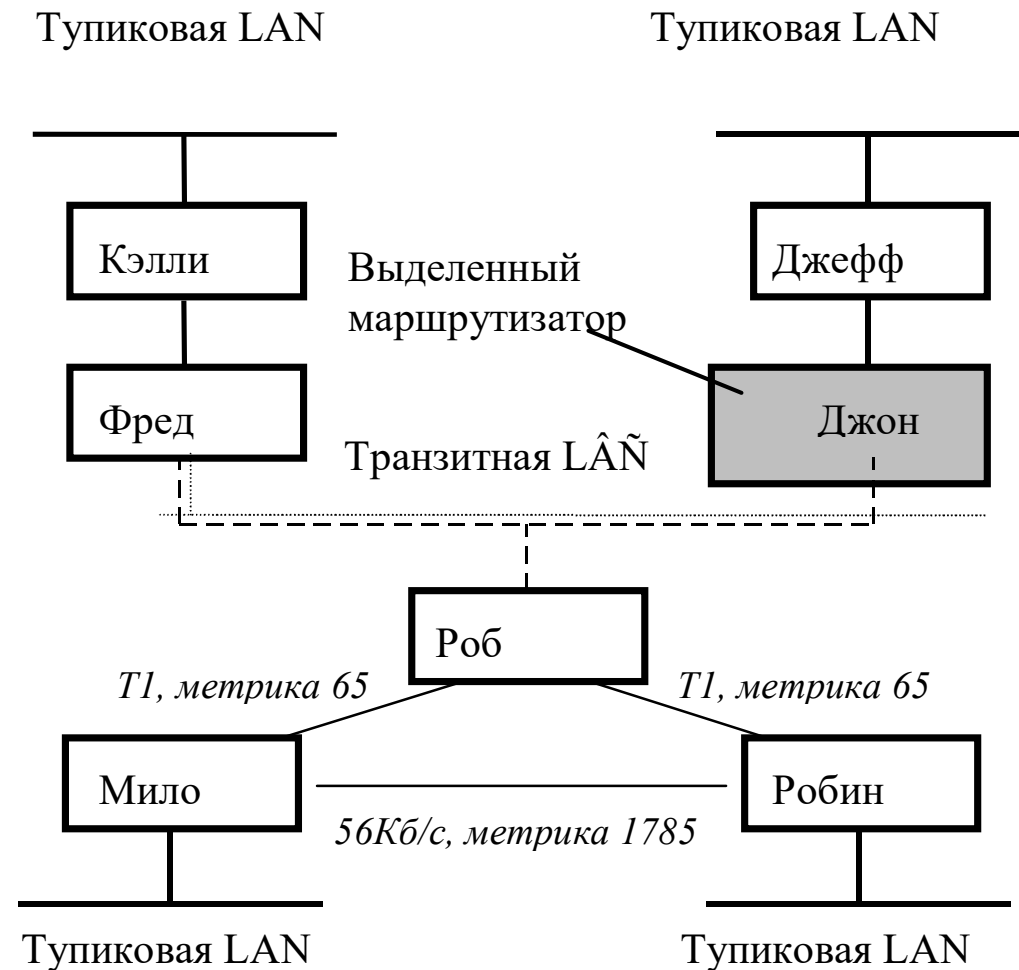
- 1) "хост-сеть» - подсеть из одного адреса
- 2) "тупиковая сеть» - подсеть, подключенная только к одному маршрутизатору
- 3) "транзитная сеть» - подсеть, подключенная к более чем одному маршрутизатору.

◆ Синхронизация маршрутизаторов "каждый с каждым" на транзитной сети:

передача базы  $N*(N-1)$  раз

◆ Синхронизация с "выделенным" маршрутизатором:

передача базы  $2*N$  раз



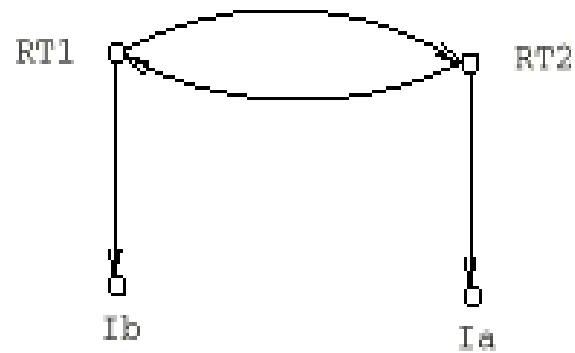
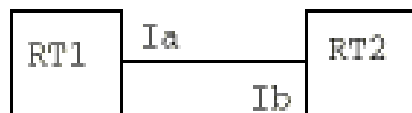
# Топологическая база OSPF

**Три типа вершин:**

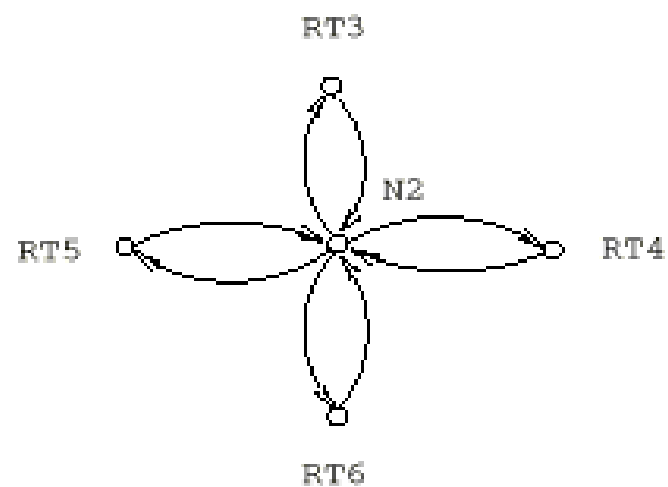
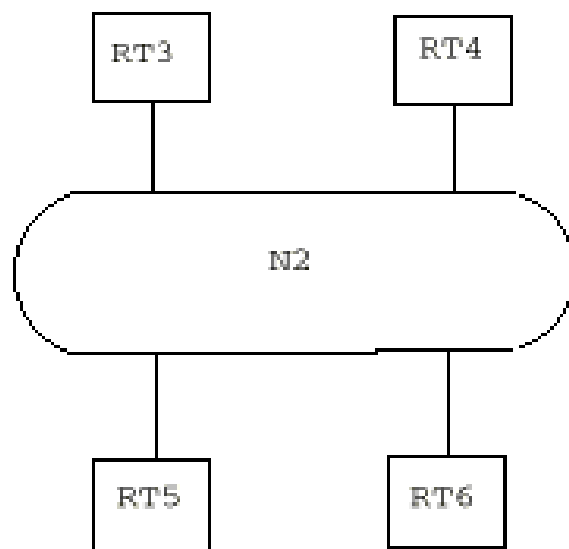
- Маршрутизатор (транзитная)
- Сеть (транзитная)
- Тупиковая (stub) сеть (не транзитная)

## Преобразование физических связей:

А) физическая связь "точка – точка"

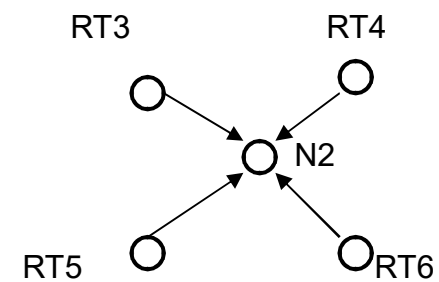
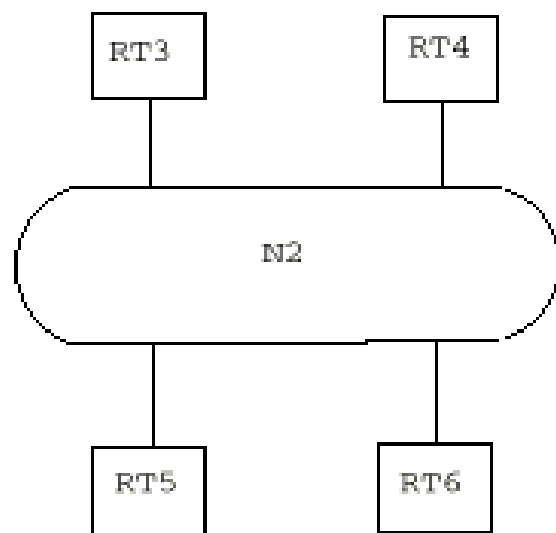


## В) Связи через сеть с множественным доступом





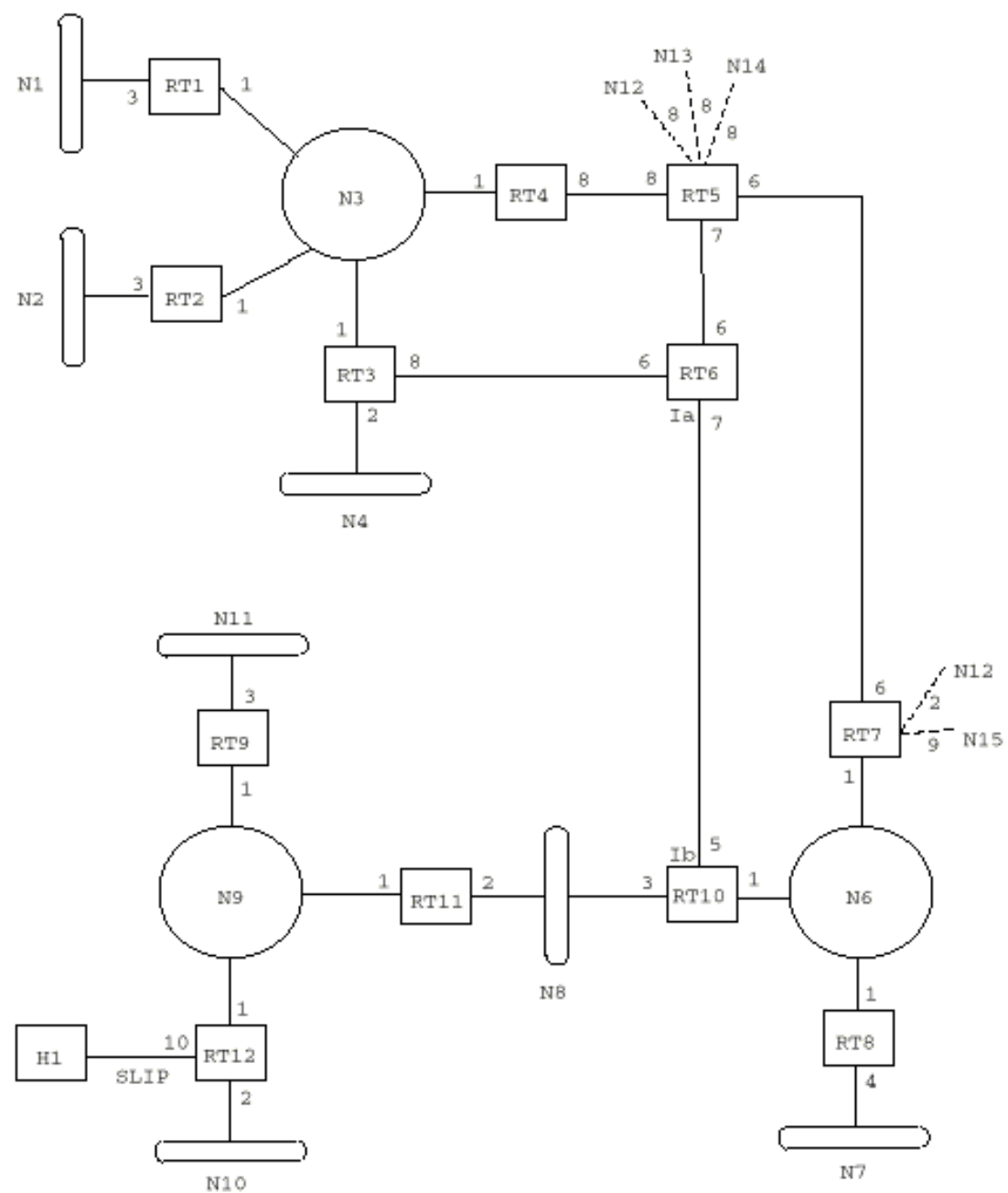
### С) Связь с тупиковой сетью с множественным доступом



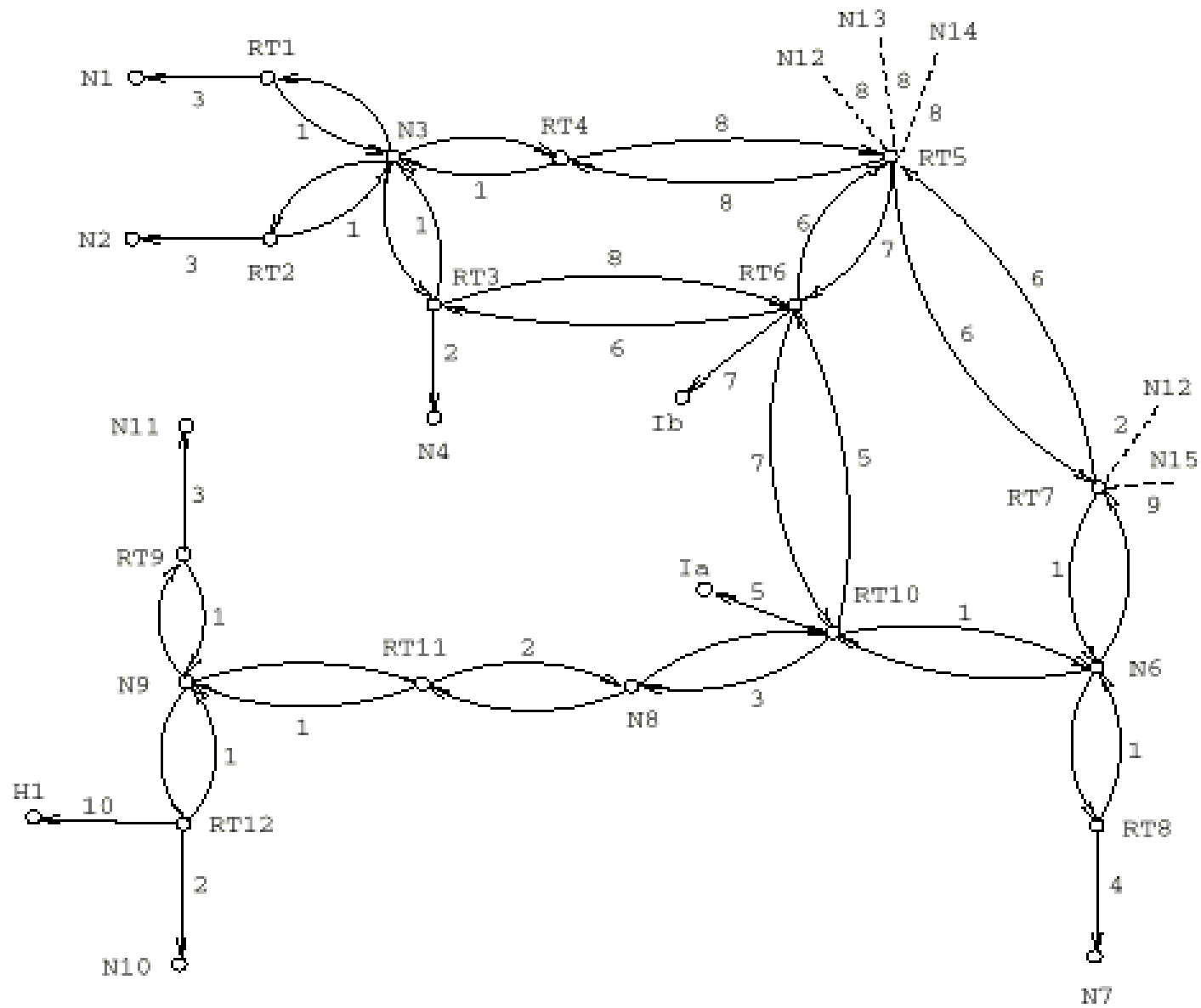
## **Пример построения топологической базы и таблицы маршрутизации**

- A) Исходная сеть**
- B) Результирующий граф**
- C) Объявления о связях маршрутизатора**
- D) Объявления о связях сети**
- E) OSP-дерево маршрутизатора RT6**
- F) Локальные маршруты в таблице RT6**
- G) Локальные маршруты в таблице RT6**

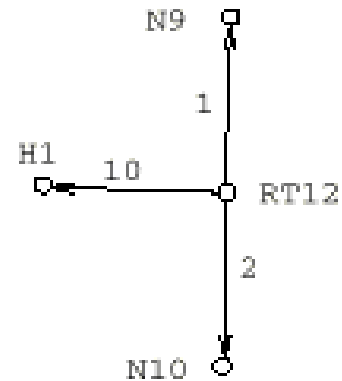
# А) Исходная сеть



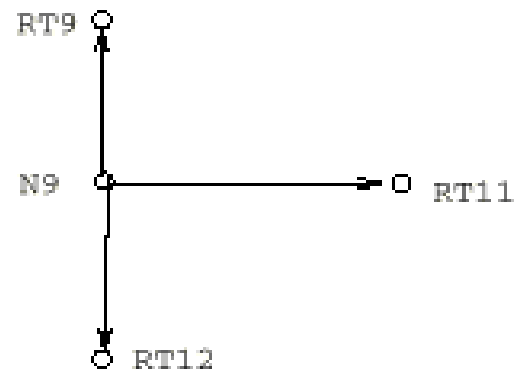
## В) Результирующий граф



### С) Объявления о связях маршрутизатора

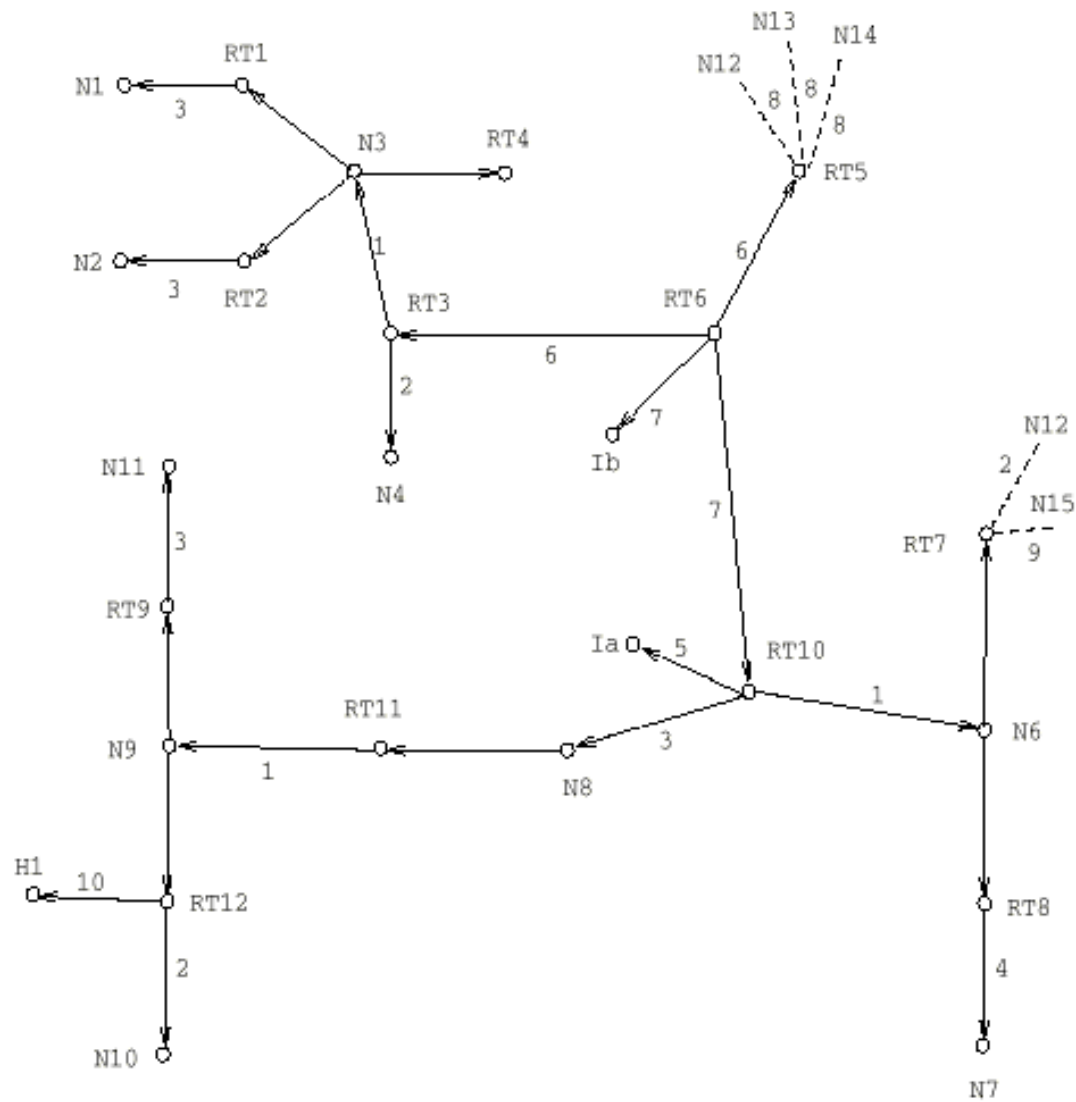


### D) Объявления о связях сети



*Метрика в направлении от  
сети до маршрутизатора  
всегда равна 0*

### Е) OSP-дерево маршрутизатора R16



**F) Локальные маршруты в таблице RT6**

<i>Destination</i>	<i>Next Hop</i>	<i>Distance</i>
N1	RT3	10
N2	RT3	10
N3	RT3	7
N4	RT3	8
Ib	*	7
Ia	RT10	12
N6	RT10	8
N7	RT10	12
N8	RT10	10
N9	RT10	11
N10	RT10	13
N11	RT10	14
H1	RT10	21
RT5	RT5	6
RT7	RT10	8

**G) Внешние маршруты в таблице RT6**

<i>Destination</i>	<i>Next Hop</i>	<i>Distance</i>
N12	RT10	10
N13	RT5	14
N14	RT5	14
N15	RT10	17



## Обработка внешних маршрутов в OSPF

- Внешние маршруты (полученные маршрутизатором по протоколам EGP или BGP) распространяются по AS без изменений
- Каждый внешний маршрут содержит IP-адрес назначения и IP-адрес маршрутизатора данной AS, через который следует передавать пакеты к сети назначения (Forwarding Address)
- Существует 2 типа внешней метрики: 1-й и 2-й:
  - Метрика типа 1 эквивалентна метрике, используемой OSPF внутри данной AS
  - Метрика типа 2 несоизмерима с внутренней метрикой OSPF — ее значения всегда больше любых значений внутренней метрики
- При использовании метрики типа 1 оптимальный маршрут к внешней сети находится путем суммирования внутренней и внешней метрик
- При использовании метрики типа 2 оптимальный маршрут к внешней сети находится путем использования только внешней метрики

## Пример:

- Если внешняя метрика к сети N12 — типа 1, то цена маршрута к сети N12 через маршрутизатор RT5 будет равна  $6+8=14$ , а через маршрутизатор RT7  $\rightarrow 1+7+2=10$ . Выбирается маршрут через маршрутизатор RT7
- Если внешняя метрика к сети N12 — типа 2, то цена маршрута к сети N12 через маршрутизатор RT5 будет равна 8, а через маршрутизатор RT7  $\rightarrow 2$ . Выбирается маршрут через маршрутизатор RT7

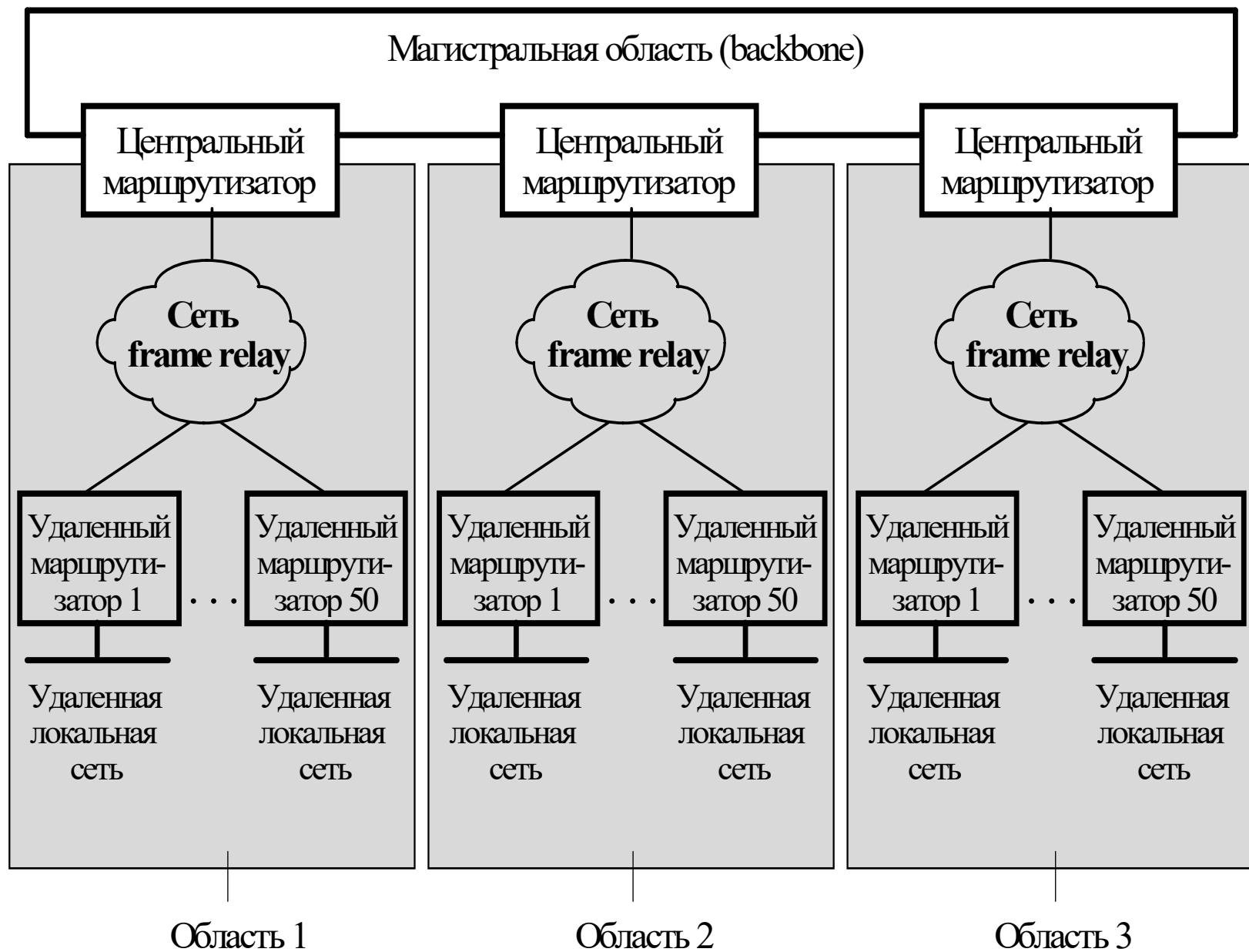
## Разделение своей автономной системы на области

- ♦ Цель - сократить размерность графа связей сети до приемлемых величин
- ♦ Граф строится только в пределах области
- ♦ Между областями передаются вектора расстояний
- ♦ Области соединены через общую область - магистраль (нет маршрутных петель между областями)

*Пограничный маршрутизатор области (ABR, area border router)*  
- это маршрутизатор с интерфейсами в двух или более областях, одна из которых является магистральной (backbone area)

**ABR** распространяет обобщенные объявления (summary links advertisements) в формате Net – NextHop (адрес пограничного маршрутизатора) :

- ♦ объявления об отдельном маршруте
- ♦ обобщения нескольких маршрутов в наименее специфический адрес
- ♦ маршрут по умолчанию



## Разделение AS на области (Areas)

Автономная система разбита на 4 области:

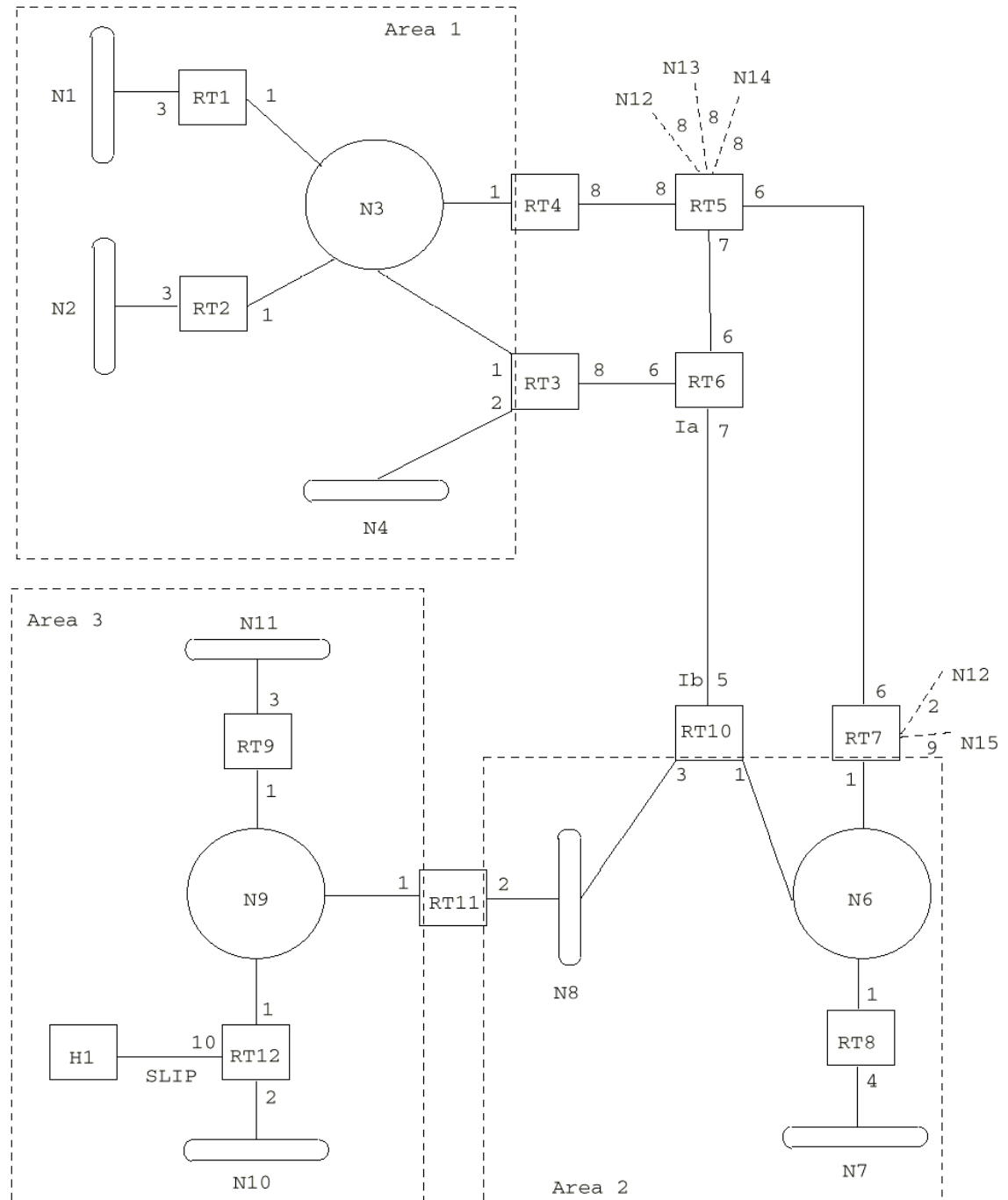
**Area 1:** RT1, RT2, **RT3**, **RT4**, N1, N2, N3, N4

**Area 2:** **RT10**, **RT7**, RT8, **RT11**, N6, N7, N8

**Area3:** **RT11**, RT9, RT12, N9, N10, N11, H1

**Backbone (магистраль):** RT3, RT4, RT5, RT6, RT7, RT10, RT11, Ia, Ib

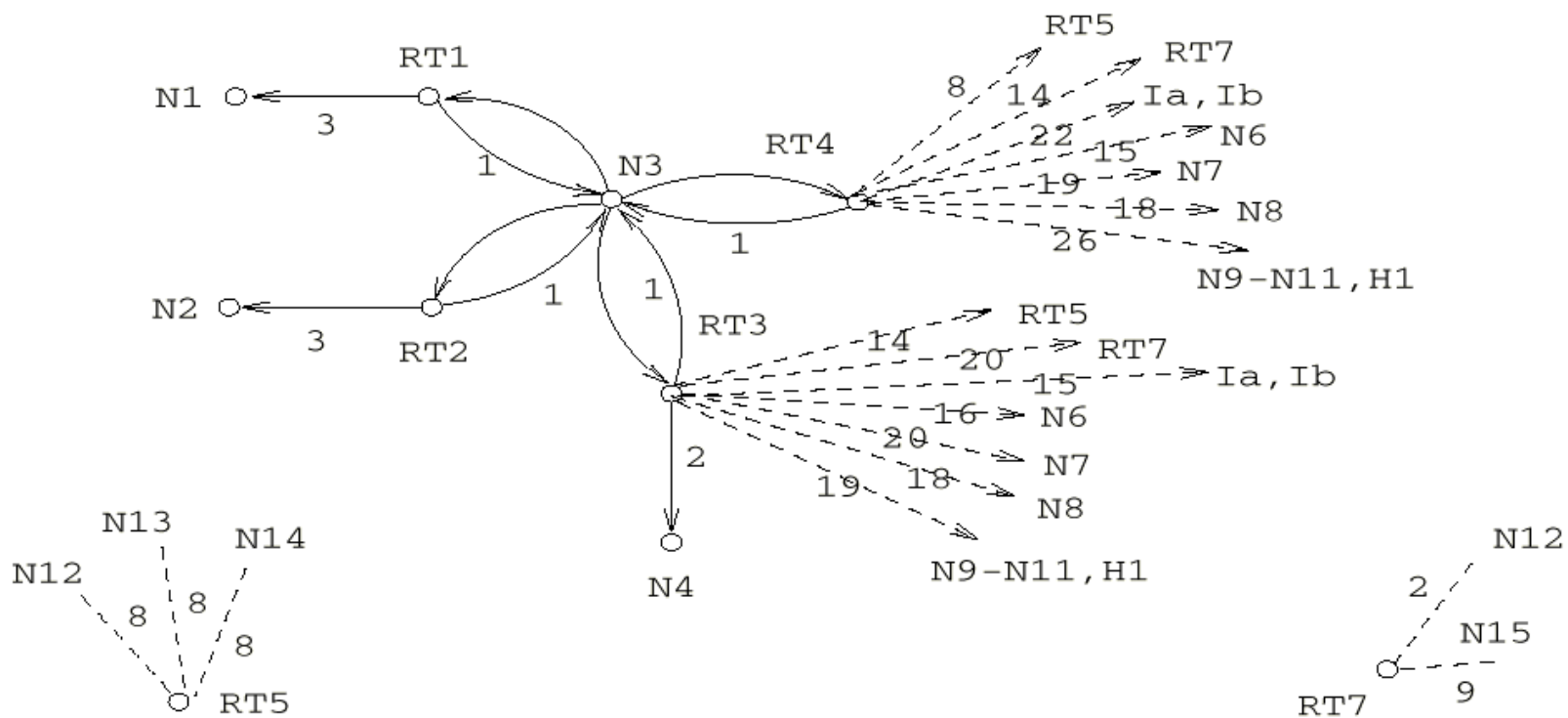
**RT<sub>x</sub>** — пограничные маршрутизаторы области (Area Border Router, ABR)



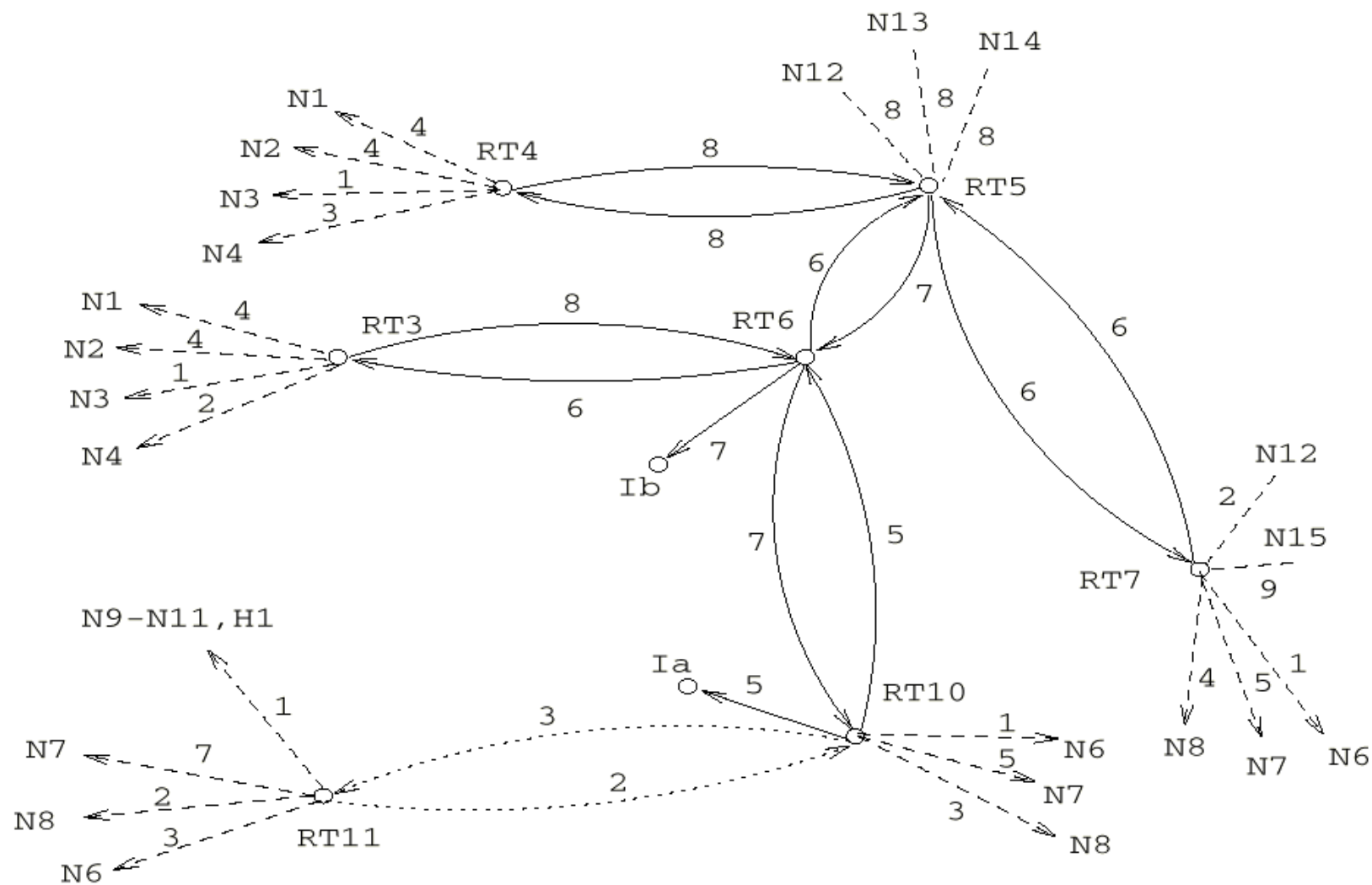
- Области обмениваются объявлениями только через магистраль
- Пограничные маршрутизаторы области распространяют обобщенные (summary) объявления о маршрутах к сетям в других областях
- Магистраль может быть образована как физическими, так и виртуальными связями

# Топологические базы областей

## База области 1:



## База магистрالی:





## Топологические базы областей

Объявления, передаваемые пограничными маршрутизаторами из области 1 в магистраль:

<i>Network</i>	<i>RT3 adv.</i>	<i>RT4 adv.</i>
N1	4	4
N2	4	4
N3	1	1
N4	2	3

Расстояния, вычисленные RT3 и RT4 до остальных пограничных маршрутизаторов

<i>Area border router</i>	<i>dist from RT3</i>	<i>dist from RT4</i>
to RT3	*	21
to RT4	22	*
to RT7	20	14
to RT10	15	22
to RT11	18	25
to Ia	20	27
to Ib	15	22
to RT5	14	8
to RT7	20	14

Объявления, передаваемые пограничными маршрутизаторами RT3 и RT4 внутренним маршрутизаторам области 1:

<i>Destination</i>	<i>RT3 adv.</i>	<i>RT4 adv.</i>
Ia,Ib	15	22
N6	16	15
N7	20	19
N8	18	18
N9-N11,H1	19	26
RT5	14	8
RT7	20	14

# Сравнение протоколов RIP и OSPF по затратам на широковещательный трафик

## Трафик, создаваемый протоколом RIP:

$$F_{RIP} = (\text{число объявляемых маршрутов}/25) \times 528 \text{ (байтов в сообщении)} \times \\ (\text{число копий в единицу времени}) \times 8 \text{ (битов в байте)}$$

## Трафик, создаваемый протоколом OSPF:

$$F_{OSPF} = \{ [20 + 24 + 20 + (4 \times \text{число соседей})] \times \\ (\text{число копий HELLO в единицу времени}) \} \times 8 + \\ [\text{число объявлений} \times \text{средний размер объявления} \times \\ (\text{число копий объявлений в единицу времени})] \times 8,$$

где

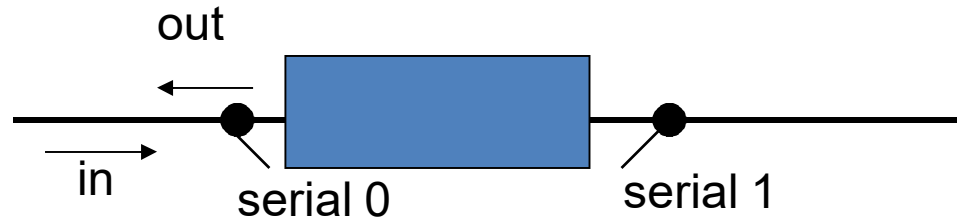
- 20 - размер заголовка IP-пакета,
- 24 - заголовок пакета OSPF,
- 20 - размер заголовка сообщения HELLO,
- 4 - данные на каждого соседа
- HELLO - каждые 10 секунд
- Объявления OSPF - каждый 30 минут

Пример: 10 маршрутизаторов, 100 сетей, маршрутизаторы соединены друг с другом через сеть frame relay

$$\mathbf{F_{RIP}: (100 \text{ маршрутов} / 25 \text{ маршрутов в объявлении})} \times 528 \times (10 \text{ копий} / 30 \text{ сек}) \times 8 = 5\,632 \text{ б/с}$$

$$\mathbf{F_{OSPF:} \{[20 + 24 + 20 + (4 \times 10)] \times (10 \text{ копий} / 10 \text{ сек})\} + [100 \text{ связей} \times (32 + 24 + 20) \times (10 \text{ копий} / 30 \times 60 \text{ сек})\} \times 8 = 1\,170 \text{ б/с}}$$

## Фильтрация пакетов с помощью маршрутизаторов



Задача – не принимать (in) или не отправлять (out) пакеты, удовлетворяющие определенным признакам

Язык фильтрации маршрутизаторов Cisco Systems:

1. Стандартный список доступа:

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

```
interface serial 0
```

```
ip access-group 1 in – применение списка доступа
```

Определение списка доступа

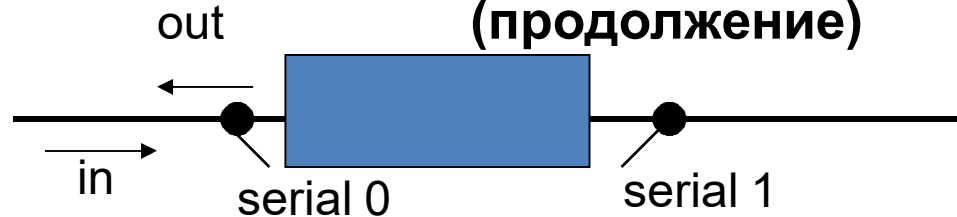
```
access-list 1 deny 192.78.46.0 0.0.255
```

```
access-list 1 deny host 193.24.15.8
```

```
access-list 1 permit any
```

[*access-list 1 deny any*] – неявный оператор в конце каждого списка

## Фильтрация пакетов с помощью маршрутизаторов (продолжение)



Задача – не принимать (in) или не отправлять (out) пакеты, удовлетворяющие определенным признакам

Язык фильтрации маршрутизаторов Cisco Systems:

1. Расширенный (extended) список доступа:

```
access-list [list number] [permit|deny]
[protocol|protocol key word]
[source address source-wildcard mask] [source port]
[destination address destination-wildcard mask]
[destination port] [log options]
```

**Запрещает ли список 101 доступ к серверу 192.78.46.8 по TCP, разрешая остальной доступ по IP?**

```
access-list 101 permit IP any host 192.78.46.8
access-list 101 deny TCP any host 192.78.46.8
```

**Запрет ping хоста 192.78.46.8:**

```
access-list 101 deny ICMP any host 192.78.46.8 eq 8
```

## Фильтрация объявлений протоколов маршрутизации



Маршрутизаторы Cisco Systems:

1. Подавление отправки объявлений на определенном интерфейсе:

```
passive-interface serial 0
```

2. Управление объявлением определенных маршрутов

```
distribute-list 2 out serial 1
```

```
access-list 2 deny 194.12.34.0 0.0.0.255
```

```
access-list 2 deny 132.7.0.0 0.0.255.255
```

```
access-list 2 permit any
```

3. Управление приемом определенных маршрутов

```
distribute-list 2 in serial 1
```

В ОС Unix – **gated** поддерживает язык фильтрации объявлений

## Связь между различными протоколами маршрутизации



```
interface serial 0
  ip address 130.93.1.1 255.255.255.0
!
interface ethernet 0
  ip address 130.94.1.1 255.255.255.0
!
router ospf 9000
  network 130.93.0.0 0.0.255.255 area 0
  redistribute rip metric 1
!
router rip
  network 130.94.0.0 0.0.255.255 area 0
  redistribute ospf 9000 metric 1
```



# Протокол EGP

- ◆ Протокол EGP (Exterior Gateway Protocol) - протоколом обмена маршрутной информацией между автономными системами (AS)
- ◆ EGP успешно работает только для иерархических связей AS
- ◆ При существовании маршрутных петель - возможно за-  
цикливание пакетов
- ◆ Каждая автономная система имеет уникальный 16-  
битный идентификатор
- ◆ Магистраль Internet рассматривается как одна автономная  
система, которая использует свой собственный внутрен-  
ний протокол маршрутизации
- ◆ Маски и подсети не поддерживаются

Протокол EGP определяет обмен маршрутной информацией между *непосредственными соседями (direct neighbor)*

Два EGP-маршрутизатора могут стать непосредственными соседями в том случае, если:

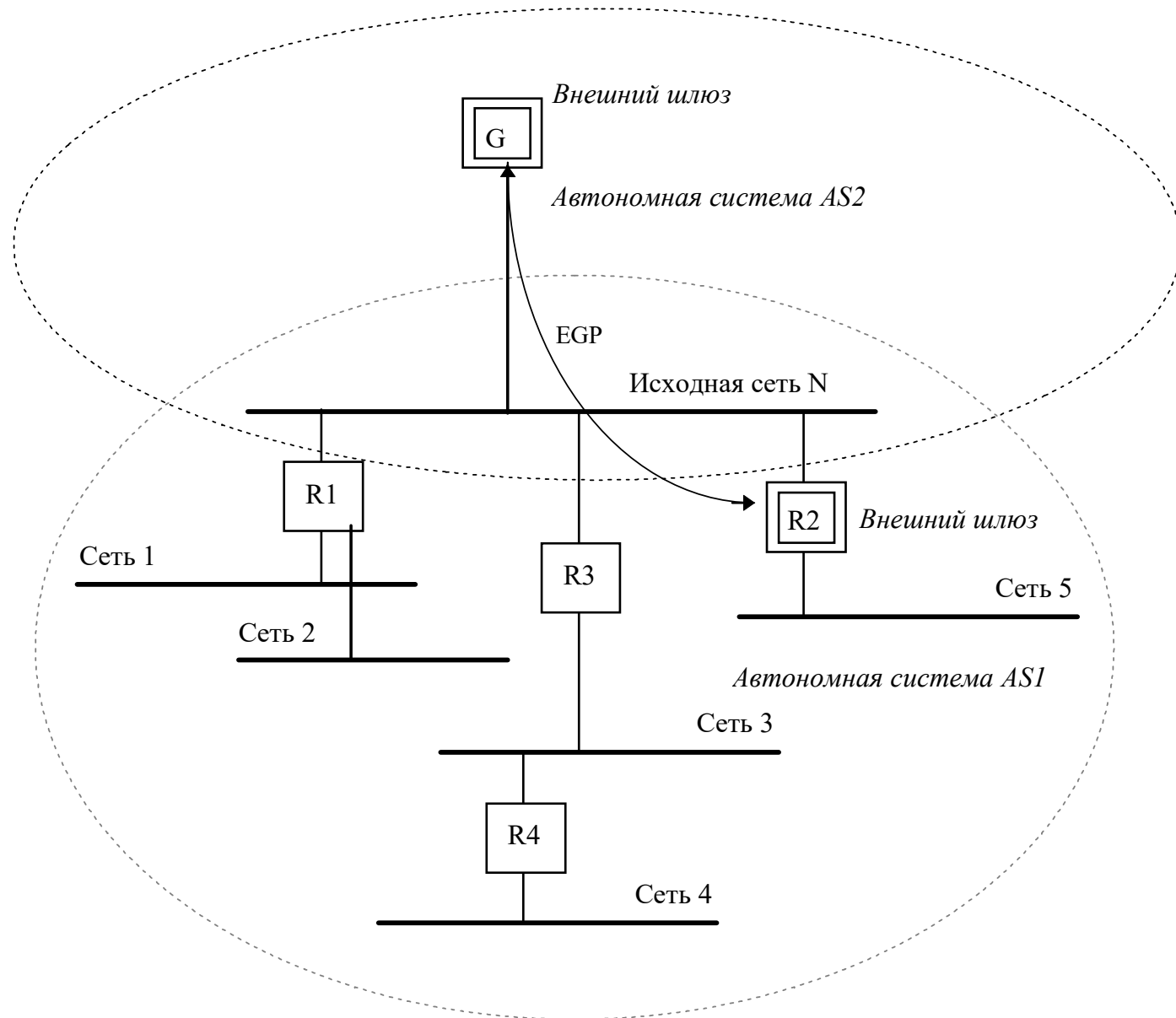
- ♦ они связаны через общую сеть (локальную или канал "точка-точка") или
- ♦ они связаны через интерсеть, которая прозрачна для них, то есть для обмена сообщениями маршрутизаторам не нужно знать внутреннюю топологию связей этой интерсети

В протоколе EGP определены три основные функции:

- ♦ установление соседских отношений
- ♦ подтверждение достижимости соседа
- ♦ обновление маршрутной информации о достижимости сетей в AS

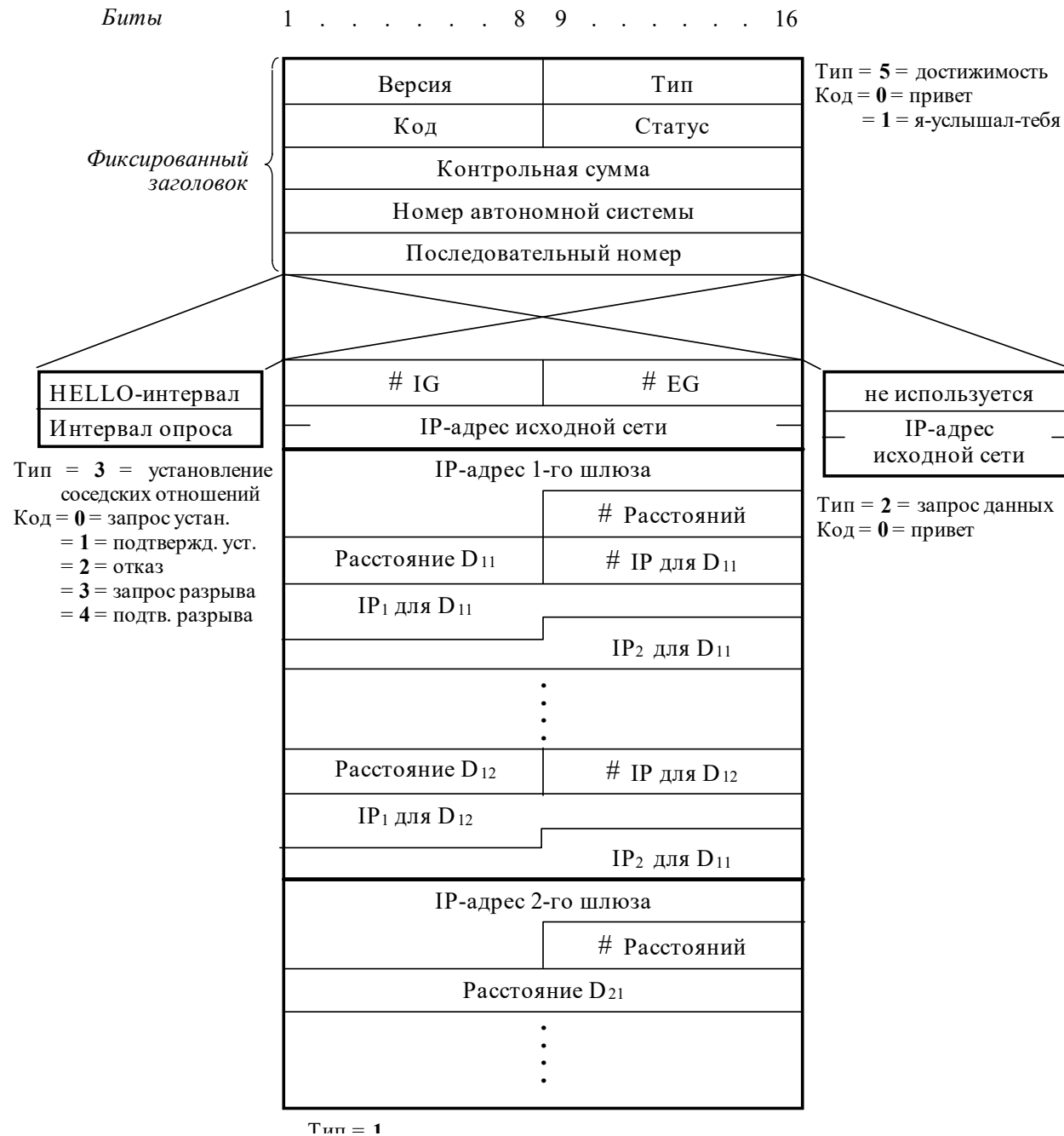
С помощью протокола достижимости сетей внешний маршрутизатор передает список сетей, для которых данный маршрутизатор является наилучшим входным маршрутизатором относительно исходной общей сети N

# Пример двух соседних автономных систем, объединяемых протоколом EGP



- ♦ Маршрутизаторы R2 и G - внешние "соседи" по EGP
- ♦ Маршрутизаторы R2, R1 и R3 - внутренние "соседи" по EGP
- ♦ Маршрутизатор R1 и R3 - "косвенные" соседи G через R2
- ♦ Маршрутизатор R2 передает G свои данные и данные маршрутизаторов R1 и R3
- ♦ "Соседи" назначаются вручную
- ♦ Список сетей для объявлений формируется вручную или берется от внутреннего протокола RIP
- ♦ Объявления о достижимости сетей выполняются на основе схемы "запрос - ответ"

# Общий формат сообщений протокола EGP



# Протокол BGP (Border Gateway Protocol) v.4

- ◆ Поддерживает произвольную структуру связей между автономными системами, в том числе и петли
- ◆ Передает маршрут достижимости - список AS, которые нужно пересечь для достижения некоторых сетей - для исключения петель
- ◆ Поддерживает "бесклассовую" маршрутизации, основанную на стратегии CIDR
- ◆ Использует надежный транспортный протокол TCP для обмена сообщениями
- ◆ Таблица маршрутизации пересылается целиком только при инициализации соединения между маршрутизаторами, а затем пересылаются только обновления маршрутов

- Протокол BGP поддерживает "**policy based**" маршрутизацию
- выбор маршрутов для объявления в другие AS производится на основании данных о сетях + набора правил, задаваемых администратором системы
- В одной AS может быть несколько BGP-маршрутизаторов, которые должны поддерживать согласованность своих таблиц с помощью внутреннего протокола маршрутизации (например, OSPF)

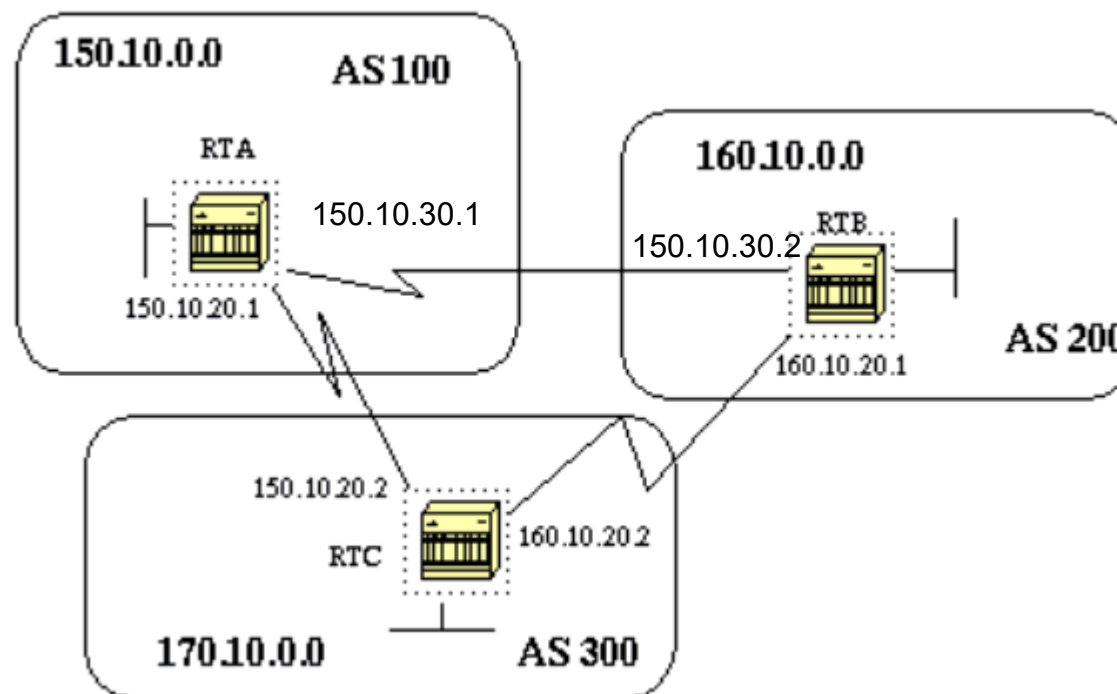
Основной единицей информации, которой оперируют маршрутизаторы BGP, является маршрут (route)
--

**Маршрут** характеризуется IP-адресами нескольких сетей назначения и путем к этим сетям.

**Путь** - это последовательность промежуточных автономных систем и IP-адрес пограничного маршрутизатора в автономной системе, к которой принадлежит сеть назначения



## Исключение петель в BGP

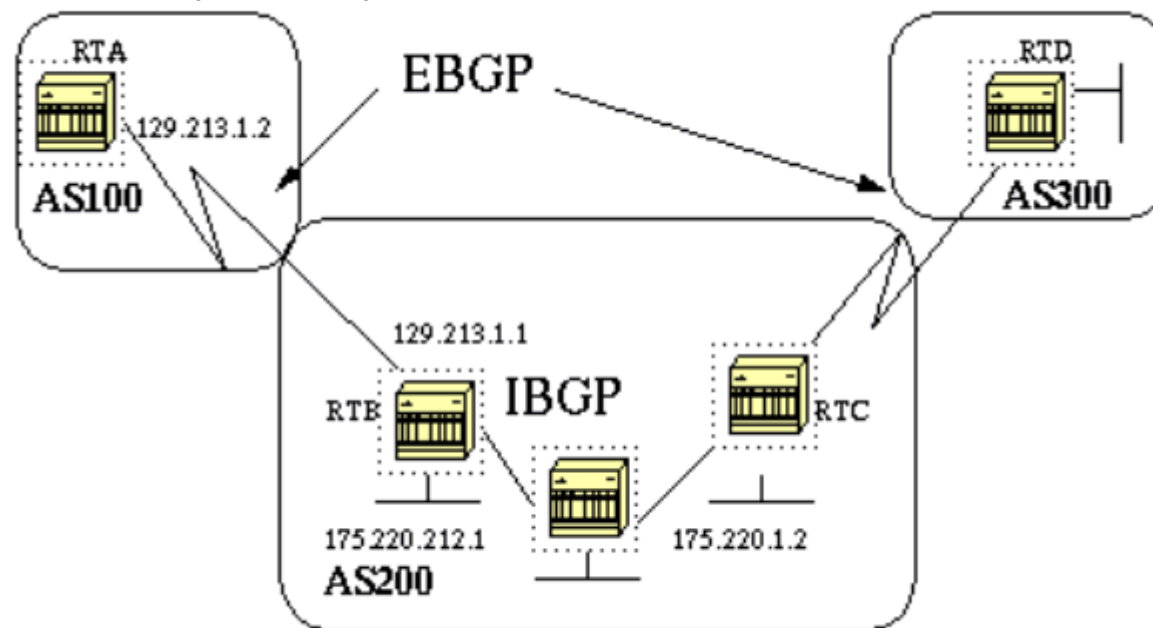


1. Объявление RTA->RTB: AS100; NH=150.10.30.1; Net=150.10.0.0 255.255.0.0
2. Объявление RTB->RTC: AS200, AS100; NH = 160.10.20.1; Net=150.10.0.0 255.255.0.0
3. Объявление RTC>RTA: AS300, AS200, AS100; NH = 150.10.20.2; Net=150.10.0.0 255.255.0.0 – отбраковывается, так как в пути встретился номер своей AS

## Внешнее и внутреннее использование BGP

Exterior BGP – между AS

Interior BGP – между маршрутизаторами одной AS



RTA#

```
router bgp 100 ( – номер AS)
neighbor 129.213.1.1 remote-as 200
```

RTB#

```
router bgp 200
neighbor 129.213.1.2 remote-as 100
neighbor 175.220.1.2 remote-as 200
```

RTC#

```
router bgp 200
neighbor 175.220.212.1 remote-as 200
```

## Фильтрация объявлений в BGP

RTC#

```
router bgp 300
```

```
neighbor 2.2.2.2 remote-as 100
```

```
neighbor 2.2.2.2 route-map  
STOPUPDATES out
```

```
route-map STOPUPDATES permit 10
```

```
match ip address 1
```

```
access-list 1 deny 170.10.0.0  
0.0.255.255
```

```
access-list 1 permit 0.0.0.0  
255.255.255.255
```

match as-path

match community

match clns

match interface

match ip address

match ip next-hop

match ip route-source

match metric

match route-type

match tag

## Организация базы маршрутной информации в BGP

Вся маршрутная информация хранится в базе данных маршрутов - Routing Information Base

RIB состоит из трех частей:

- ◆ Adj-RIBs-In - необработанные маршруты, поступающие из других AS и из своей собственной AS в сообщениях UPDATE протокола BGP.
- ◆ Loc-RIB - локальная маршрутная информацию, которую маршрутизатор BGP выбрал в соответствии с его локальной политикой из базы Adj-RIBs-In.
- ◆ Adj-RIBs-Out - маршруты, выбранные в соответствии с внешней политикой для объявлений другим BGP маршрутизаторам

# Типы сообщений BGP

Протокол BGP использует четыре типа сообщений:

- |                       |  |
|-----------------------|--|
| <b>OPEN -</b>         | открывает логическое соединение между BGP-маршрутизаторами                                     |
| <b>UPDATE -</b>       | содержит данные о новом маршруте или отзываемых маршрутах                                      |
| <b>NOTIFICATION -</b> | предназначены для уведомления об ошибках, обнаруженных в полученных сообщениях UPDATE или OPEN |
| <b>KEEPALIVE -</b>    | служат для поддержания сессии между соседями в открытом состоянии                              |

## Формат сообщения UPDATE

В одном сообщении UPDATE маршрутизатор может объявить один достижимый маршрут или отозвать несколько недостижимых маршрутов:

Unfeasible Routes Length (2 octets)
Withdrawn Routes (variable)
Total Path Attribute Length (2 octets)
Path Attributes (variable)
Network Layer Reachability Information (variable)

- ♦ **Withdrawn Routes** - описывает отзываеые недостижимые маршруты

Каждый маршрут в поле Withdrawn Routes описывается парой чисел:

- ◇ Длина префикса (1 байт)
- ◇ Префикс (переменной длины)

- ♦ **Network Layer Reachability Information** - содержит перечень сетей, к которым ведет новый маршрут, описанный полем Path Attributes

Перечень сетей задается парами значений "Длина префикса - Префикс"

- ♦ **Path Attributes** содержит данные о атрибутах путей между AS:

- AS\_PATH представляет собой последовательность номеров автономных систем, которые должен пересечь маршрут, описываемый сообщением UPDATE. Исходная информация для отбраковки петель
- NEXT\_HOP - определяет IP-адрес пограничного маршрутизатора, через который нужно передавать пакеты сетям, которые указаны в поле Network Layer Reachability

## ◆ Обработка маршрутов по протоколу BGP

**Этап 1.** Проверяется, не совпадает ли новый маршрут с одним из имеющихся в базе Adj-RIB-In. Для этого совпадение списка достижимых сетей, указанных в поле Network Layer Reachability, сравнивается с списком сетей каждого маршрута, хранящегося в Adj-RIB-In

- ◆ Если список сетей маршрута совпадает со списком какого-либо маршрута, но путь, указанный в параметре Path Attributes, отличается от пути этого маршрута, то такой маршрут называется **новым** и он вытесняет старый из базы Adj-RIB-In. Затем запускается Decision Process, так как старый маршрут больше не может использоваться



- ◆ Маршрут называется перекрывающимся (overlapping route), если его список сетей имеет общее подмножество со списком сетей какого-либо другого маршрута
  - ◇ Если новый маршрут является перекрывающимся и более специфическим, то вызывается Decision Process. Например, существует маршрут 194.27.0.0/16, новый - 194.27.192.0/18.
  - ◇ Если новый маршрут имеет параметры пути, содержащиеся в каком-либо имеющемся в Adj-RIB-In маршруте, и является более специфическим, то никакие действия не выполняются

- ◆ Если новый маршрут содержит список сетей, который не совпадает ни с одним списком, уже имеющимся в Adj-RIB-In, то он должен быть помещен в Adj-RIB-In, а затем должен быть запущен Decision Process
- ◆ Если новый маршрут является перекрывающимся и менее специфическим, чем имеющийся, то нужно запустить Decision Process над набором сетей назначения, которые описываются только менее специфическим маршрутом

**Этап 2. Decision Process** выбирает маршруты для последовательного объявления другим BGP-маршрутизаторам путем применения к маршрутам, хранящимся в Adj-RIB-In, правил из локальной базы Policy Information Base - PIB.

Decision Process формализуется путем определения функции предпочтения маршрутов:

Функция предпочтения не должна принимать во внимание:

- ⇒ существование других маршрутов
- ⇒ параметры других маршрутов
- ⇒ отсутствие других маршрутов
- ⇒ некоторый маршрут должен быть оценен "сам по себе"

После оценки всех маршрутов из совпадающих маршрутов выбирается тот, у которого степень предпочтения выше

## **Decision Process ответственен за:**

- ♦ выбор маршрутов для объявлений BGP-маршрутизаторам, находящимся в своей автономной системе
- ♦ выбор маршрутов для объявлений BGP-маршрутизаторам, находящимся в соседних автономных системах
- ♦ агрегирование маршрутной информации за счет поглощения более специфических маршрутов менее специфическими в том случае, если пути у этих маршрутов совпадают

## Decision Process состоит из трех фаз

### **Фаза 1:**

- ♦ вычисление степени предпочтения для каждого маршрута, полученного от соседней автономной системы
- ♦ объявление BGP-маршрутизаторам из своей автономной системы всех маршрутов, которые имеют высшую степень предпочтения
- ♦ Стандарт BGP не определяет вида этих правил выбора маршрута, за исключением одного - если в маршруте уже имеется номер своей автономной системы, то это значит, что маршрут содержит петлю и поэтому он должен быть отбракован
- ♦ Любые другие правила администратор AS может сконфигурировать в том виде, в каком их позволяет конфигурировать разработчик BGP-маршрутизатора

### **Примеры правил:**

- ◇ запрещение использования некоторых AS как транзитных в путях до сетей назначения
- ◇ предпочтение одним AS перед другими для транзита

## Фаза 2

- ◆ Вызывается по завершению фазы 1
- ◆ Ответственна за выбор наилучшего маршрута из всех, имеющихся для каждого отличного от других набора сетей назначения, и установки этого маршрута в локальную базу Loc-RIB. В Loc RIB включаются также и сети своей AS
- ◆ При установке маршрута BGP-маршрутизатор ДОЛЖЕН определить непосредственный следующий маршрутизатор к адресу, приведенному в поле NEXT\_HOP, с помощью внутреннего протокола маршрутизации, то есть протокола IGP, используемого в данной автономной системе

### **Фаза 3**

- ♦ Вызывается после завершения работы фазы 2
- ♦ Ответственна за распространение маршрутов из Loc-RIB BGP-маршрутизаторам из соседних AS в соответствии с правилами, содержащимися в базе RIB

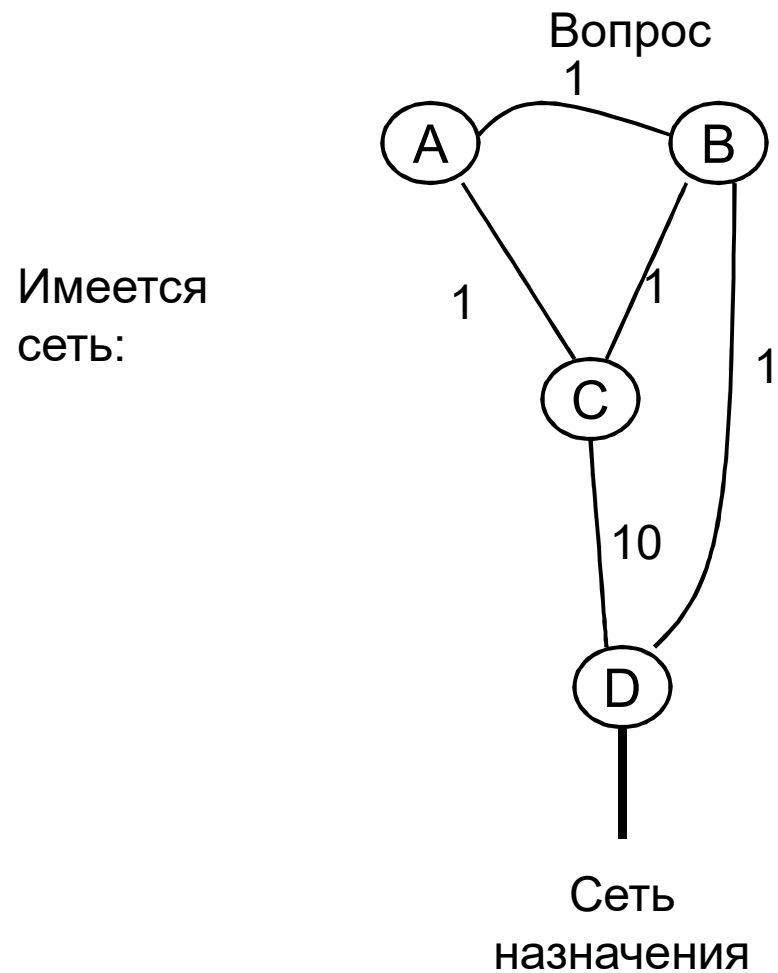
Правила могут запрещать:

⇒ объявление некоторых сетей из своей или других областей

⇒ объявления своих сетей для некоторых AS

⇒ объявления сетей всех других AS для некоторой AS

- ♦ Фаза 3 выполняет также агрегирование маршрутов
- ♦ При передаче маршрута, описывающего сети назначения, находящиеся в другой автономной системе, BGP-маршрутизатор должен модифицировать значения полей:
  - ⇒ Path Attributes - добавить своей номер AS
  - ⇒ NEXT\_HOP.- указать маршрутизатор, который соединен с общей исходной сетью с той AS, куда идет объявление



Записи о пути к сети назначения

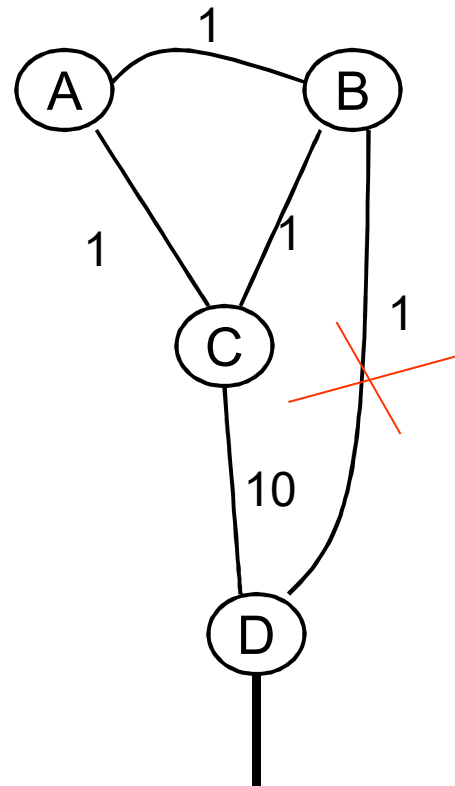
D: direct, metric 1

B: route via D, metric 2

C: route via B, metric 3

A: route via B, metric 3





Связь B-D отказывает

Маршрутизаторы не поддерживают Split Horizon

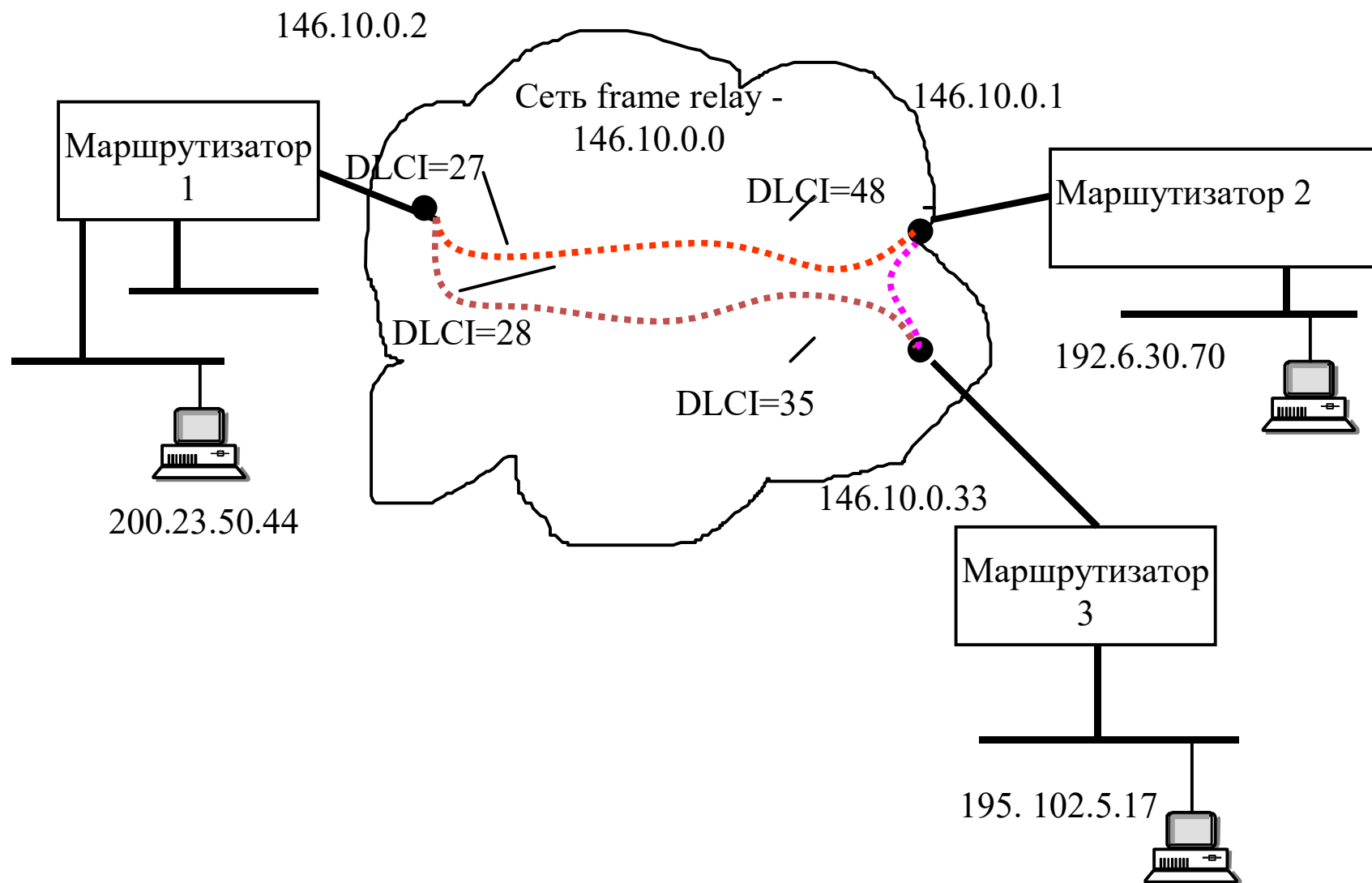
Что происходит в сети при наихудшей синхронизации объявлений?

Сеть  
назначения

Записи о пути к сети назначения

D:	dir, 1	dir, 1
B:	16	
C:	B, 3	
A:	B, 3	

## Конфигурирование маршрутизаторов при работе через нешироковещательные (NBMA) сети



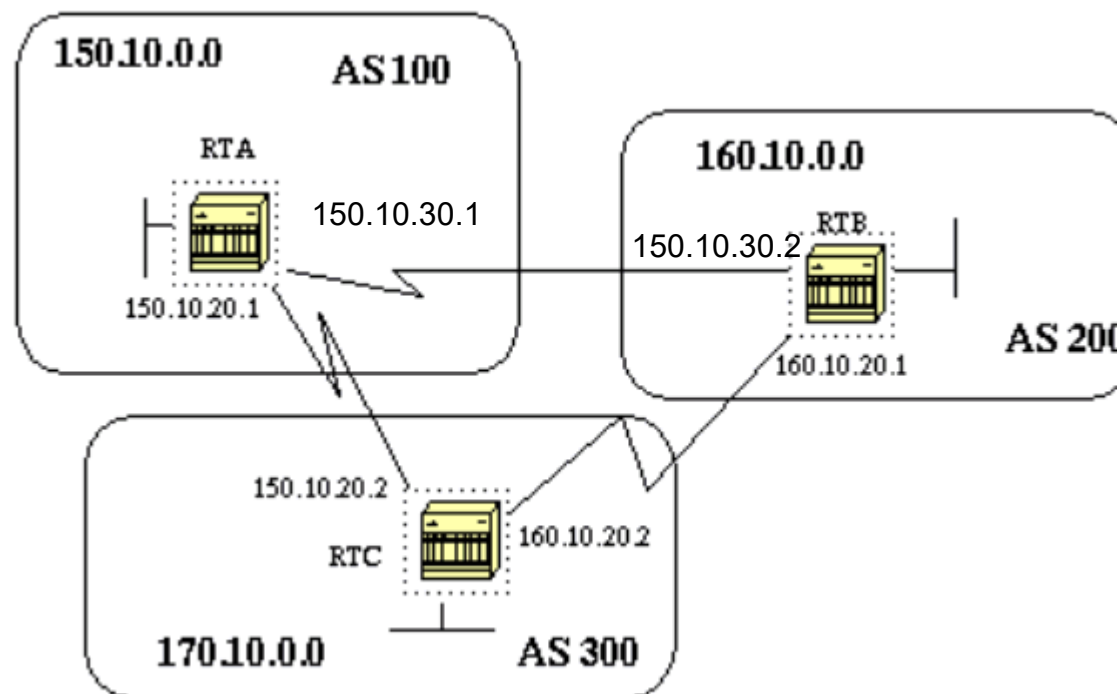
**Таблица маршрутизации M1:**

Сеть	Маска	Следующий маршрутизатор	Интерфейс	Метрика
146.10.0.0	255.255.0.0	146.10.0.2	146.10.0.2	1
200.23.50.0	255.255.255.0	146.10.0.2	146.10.0.2	1
192.6.30.0	255.255.255.0	146.10.0.1	146.10.0.2	2
195. 102.5.0	255.255.255.0	146.10.0.33	146.10.0.2	2

**ARP-таблица маршрутизатора M1:**

146.10.0.1	27
146.10.0.33	28

## Исключение петель в BGP

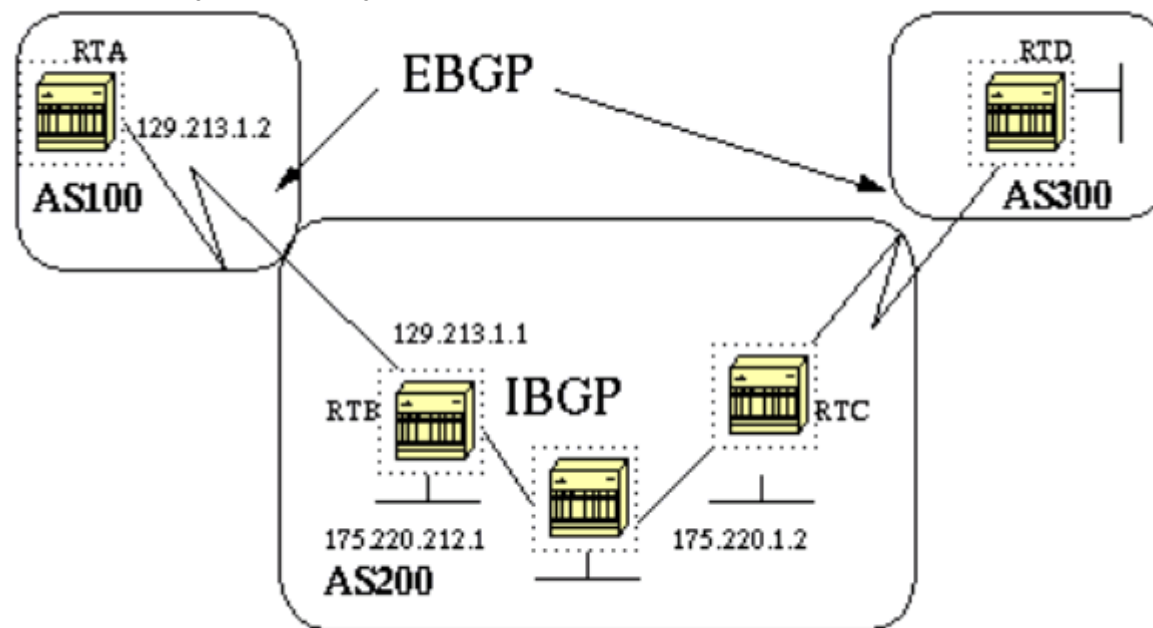


1. Объявление RTA->RTB: AS100; NH=150.10.30.1; Net=150.10.0.0 255.255.0.0
2. Объявление RTB->RTC: AS200, AS100; NH = 160.10.20.1; Net=150.10.0.0 255.255.0.0
3. Объявление RTC->RTA: AS300, AS200, AS100; NH = 150.10.20.2; Net=150.10.0.0 255.255.0.0 – отбраковывается, так как в пути встретился номер своей AS

## Внешнее и внутреннее использование BGP

Exterior BGP – между AS

Interior BGP – между маршрутизаторами одной AS



RTA#

```
router bgp 100 ( – номер AS)
neighbor 129.213.1.1 remote-as 200
```

RTB#

```
router bgp 200
neighbor 129.213.1.2 remote-as 100
neighbor 175.220.1.2 remote-as 200
```

RTC#

```
router bgp 200
neighbor 175.220.212.1 remote-as 200
```

## Фильтрация объявлений в BGP

RTC#

```
router bgp 300
```

```
neighbor 2.2.2.2 remote-as 100
```

```
neighbor 2.2.2.2 route-map  
STOPUPDATES out
```

```
route-map STOPUPDATES permit 10
```

```
match ip address 1
```

```
access-list 1 deny 170.10.0.0  
0.0.255.255
```

```
access-list 1 permit 0.0.0.0  
255.255.255.255
```

match as-path

match community

match clns

match interface

match ip address

match ip next-hop

match ip route-source

match metric

match route-type

match tag