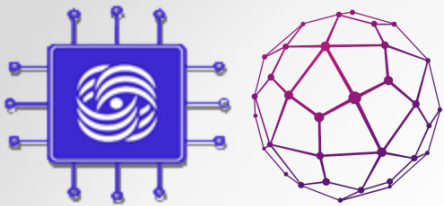


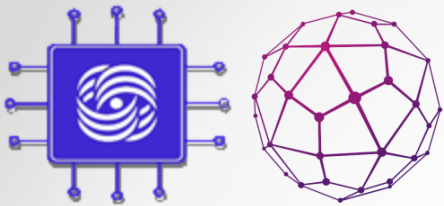
Модели распределения сетевых ресурсов

Доп. главы Компьютерных сетей и
телекоммуникации
к.ф.-м.н. Чемерицкий Е.В.



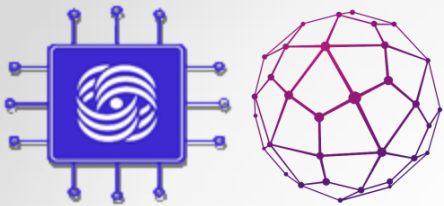
Вопросы:

- Каковы преимущества и недостатки архитектуры коммутатора с буферизацией на выходе?
- Сколько очередей внутри коммутатора с N портами, виртуальной буферизацией на входе, WRR из 5 очередей и 3-х уровневый Random Early Detection необходимо для поддержания требований качества сервиса?



План лекции

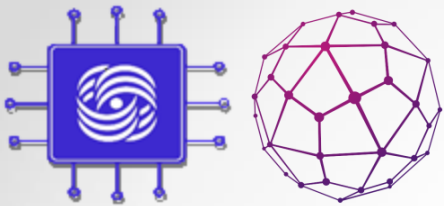
- Модель Интегрированных Сервисов
- Протокол Резервирования Ресурсов (RSVP)
- Модель Дифференцированных Сервисов
- Маршрутизация с учётом качества сервиса



Качество Сервиса в сети Интернет

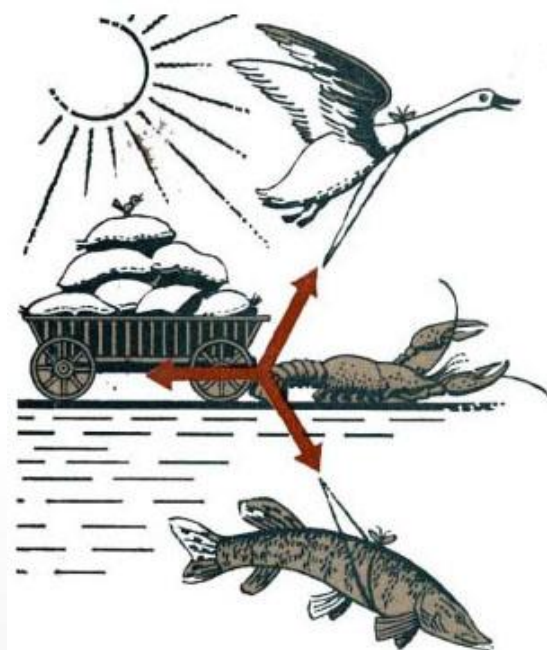
- Если управление качеством сервиса не поддерживается, то сетевые соединения обслуживаются по принципу ***best effort***
- Потоки данных обслуживаются одинаково
 - Зачем нужно разное качество сервиса?
 - Дифференцирование потоков по качеству:
всегда ли это хорошая идея?

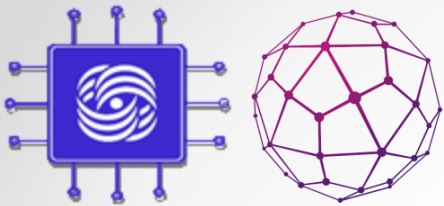
David D. Clark, K. Sollins, J. Wroclawski, R. Braden,
Tussle in Cyberspace: Defining Tomorrow's Internet
Proceedings of SIGCOMM 2002, ACM Press, 2002



Методы управления качеством сервиса в сети

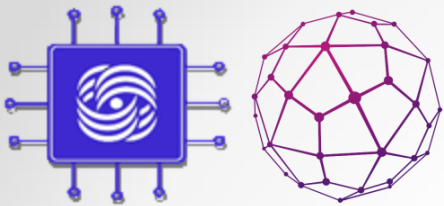
- Выполнение потребностей приложений
 - Какие параметры соединения можно задавать?
 - Даётся ли гарантия выполнения требований?
- Сложность реализации и эксплуатации
 - Какими дополнительными функциями должно обладать оборудование?
 - Какая поддержка требуется от хостов?
- Накладные расходы на использование
 - Насколько эффективна работа сети?
 - Сколько ресурсов будет задействовано?
 - Сколько будет простаивать?





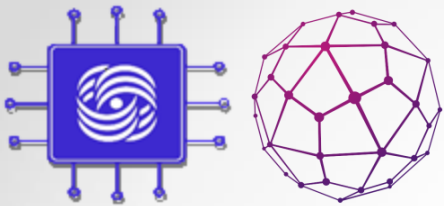
Проблема эффективного распределения ресурсов

- Проблема обеспечения качества связана с проблемой распределения сетевых ресурсов между потоками данных
 - Проблему распределения ресурсов можно формализовать как задачу оптимизации
 - Чем больше ресурсов вовлечено в обслуживание потока, тем выше качество его соединения
 - Чем большее количество ресурсов позволяет задействовать модель распределения, тем выше эффективность сети, и тем больший **уровень утилизации** достигается



Почему высокий уровень утилизации – хорошо?

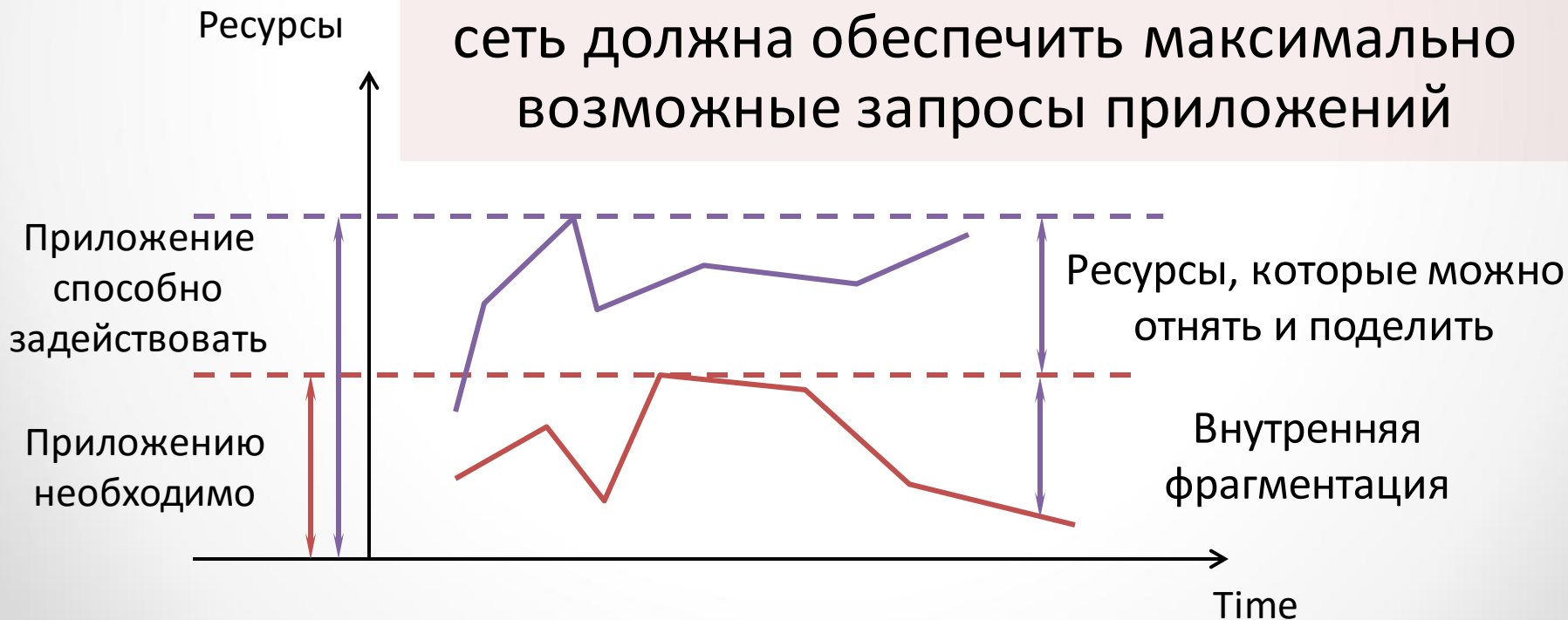
- Приложения не всегда способны задействовать все ресурсы сети
 - Spanning Tree Protocol – в топологии Fat Tree
 - Backbone networks – запас производительности
 - Разница в характеристиках аппаратуры
- Чем выше уровень утилизации, тем лучше отношение производительности сети к стоимости сетевой инфраструктуры
 - Уменьшение затрат провайдеров

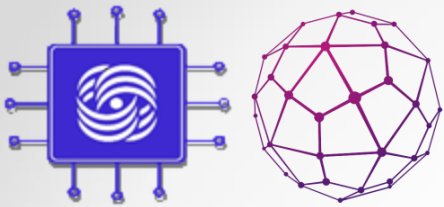


Низкая утилизация при отсутствии управления QoS

Сеть работает корректно, если она предоставляет каждому приложению необходимые ему ресурсы

При отсутствии управления качеством сеть должна обеспечить максимально возможные запросы приложений



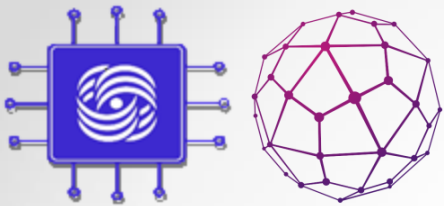


Модель Интегрированных Сервисов (IntServ)

Мультимедийный трафик в сети:

- Как оградить TCP трафик от мультимедийных данных, передающихся через UDP?
- Как обеспечить качество соединения?

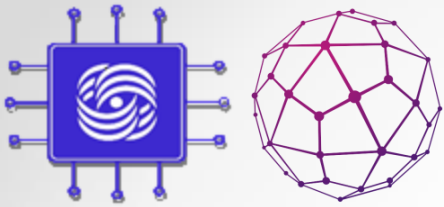
Гарантированный уровень качества можно обеспечить лишь с помощью **резервирования ресурсов** – закрепления части ресурсов сети за конкретным потоком данных



Модель Интегрированных Сервисов (IntServ)

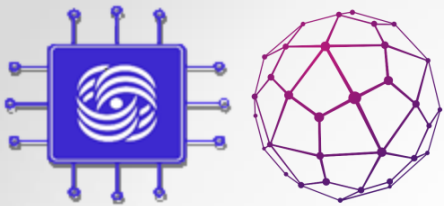
Основная идея – прокладывание маршрутов с заданным качеством путём предварительного резервирования ресурсов на оборудовании

- Модель лишь расширяет архитектуру Интернета, и способна работать по принципу *best-effort*
- Модель особенно эффективна при многоадресной передаче данных
- Допускаются накладные расходы на предварительное прокладывание маршрута
- ***всё или ничего*** – модель или гарантирует соединение нужного качества, или отказывается предоставить какое-либо соединение



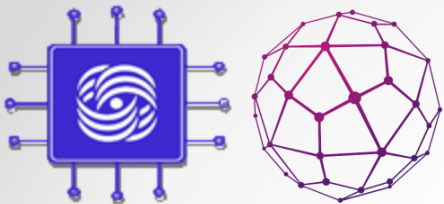
Основные компоненты модели IntServ

- Классификатор (classifier)
 - Разделение пакетов на классы обслуживания
- Планировщик (scheduler)
 - Обеспечение выполнения требований QoS
- Контроль доступа (admission control)
 - Оценка возможности добавления потоков
- Протокол резервирования ресурсов
 - Резервирование ресурсов вдоль маршрута

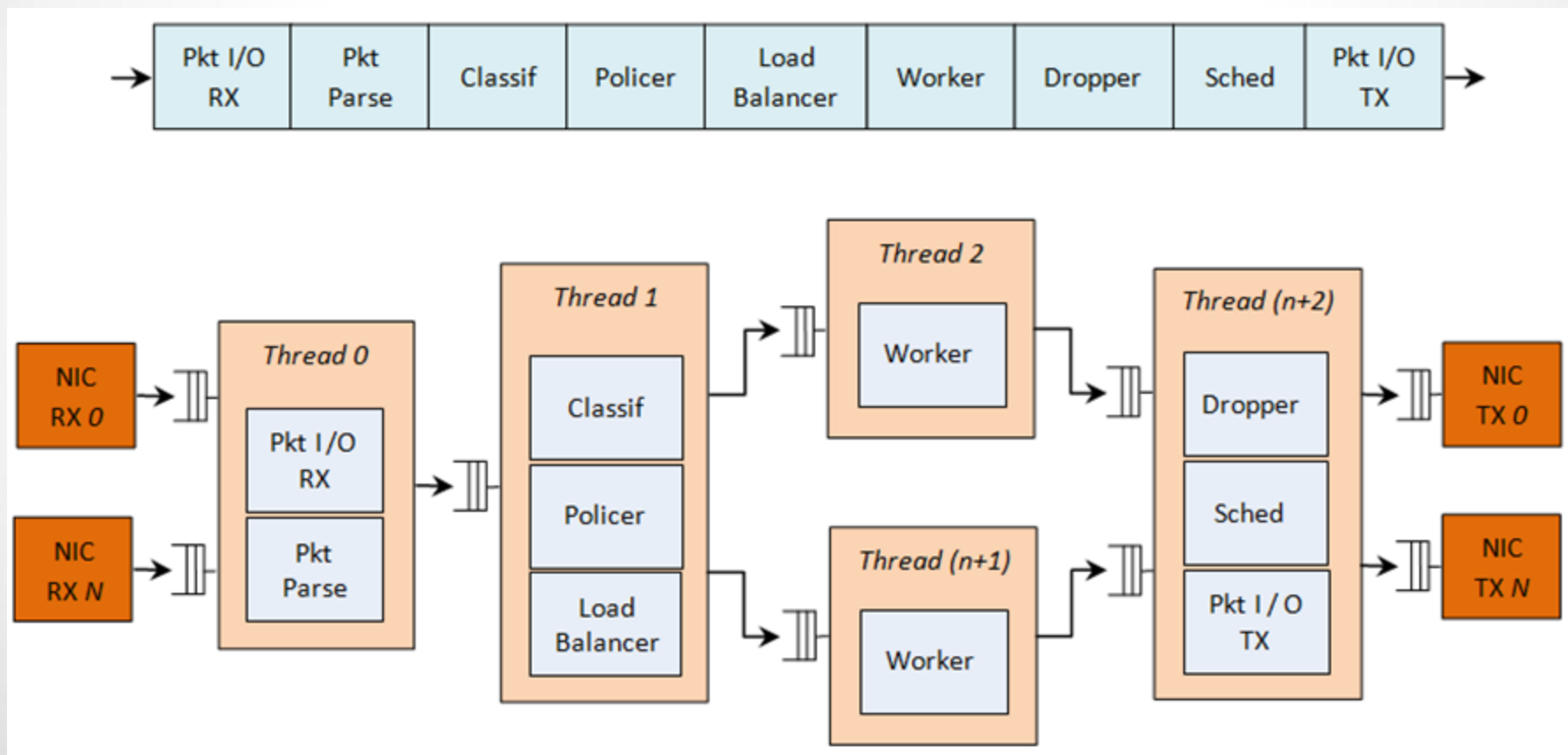


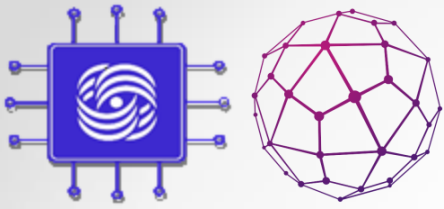
Расчёт и обеспечение качества соединений

- Распределение по очередям на входящих и исходящих интерфейсах коммутатора
- Использование policing & shaping для формирования нужного профиля потоков
- Установка надлежащих дисциплин сброса пакетов при их постановке в очереди и выборки пакетов из очередей
- Настройка алгоритмов планирования коммутационной матрицы



Расчёт и обеспечение качества соединений

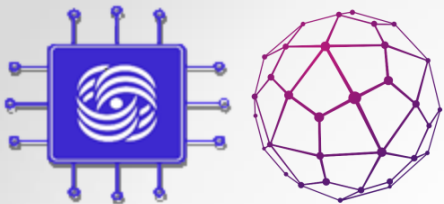




Протокол резервирования ресурсов RSVP

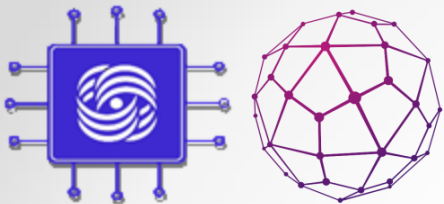
Поддерживаемые типы трафика:

- Регулярный (*best-effort*)
 - Передача файлов, просмотр почты, etc
- Чувствительный к скорости (*rate-sensitive*)
 - Потокковое вещание аудио и видео
- Чувствительный к задержке (*delay-sensitive*)
 - Voice Over IP, online игры

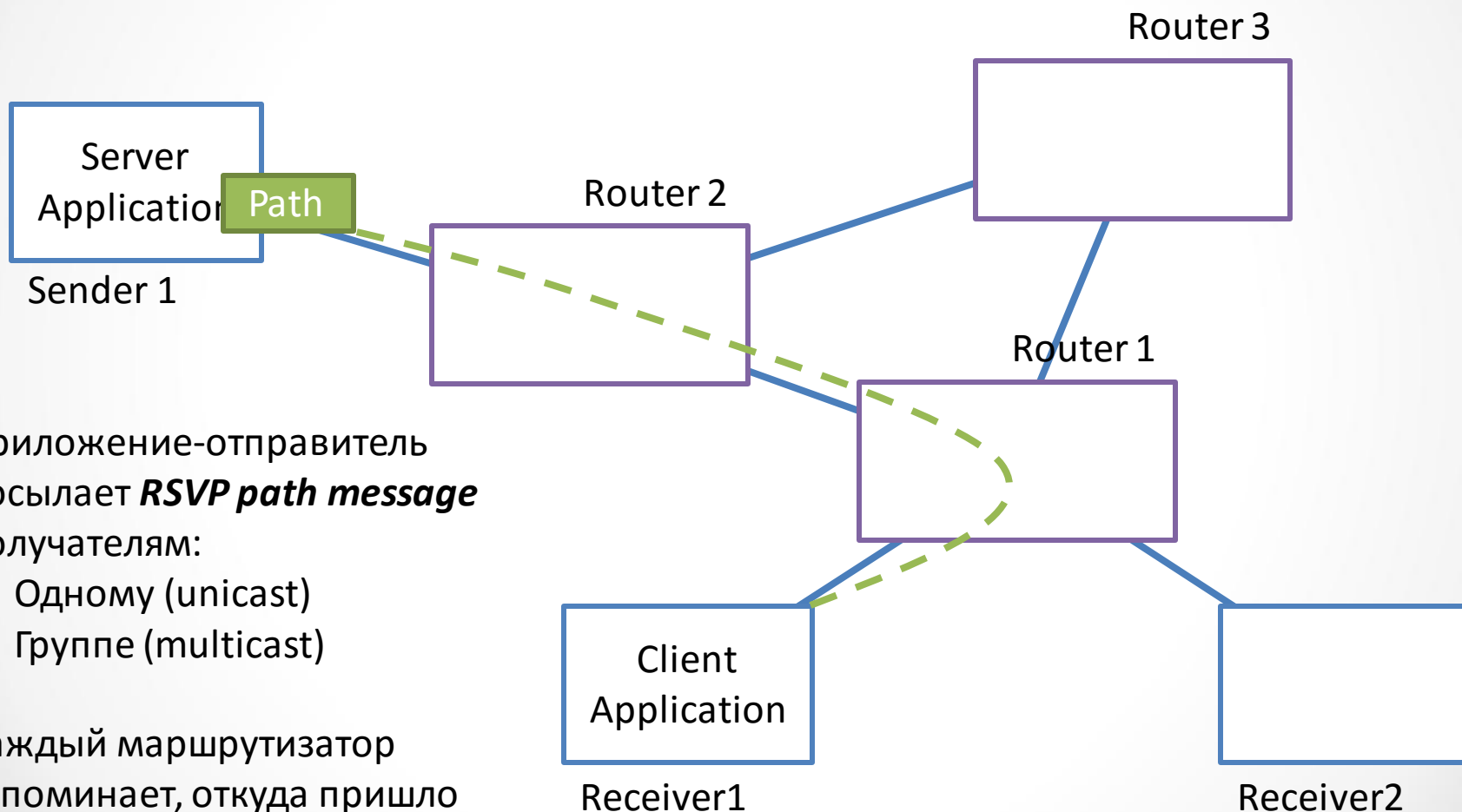


Протокол резервирования ресурсов RSVP

- **Поток данных** – последовательность пакетов, имеющих общего отправителя, единый набор получателей, и требующих одинакового уровня качества сервиса
- **Спецификация потока (*flowspec*)**
 - определяет поток данных, тип трафика, требования к качеству соединения
- **Спецификация фильтра (*filterspec*)**
 - Определяет правила доступа пакетов к зарезервированным ресурсам
- RSVP работает на уровне **сессий** – множеств потоков с одинаковым набором получателей



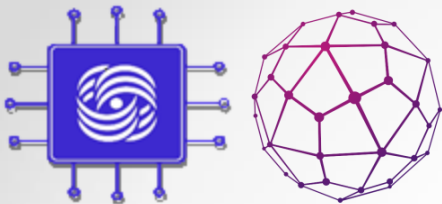
Пример работы RSVP



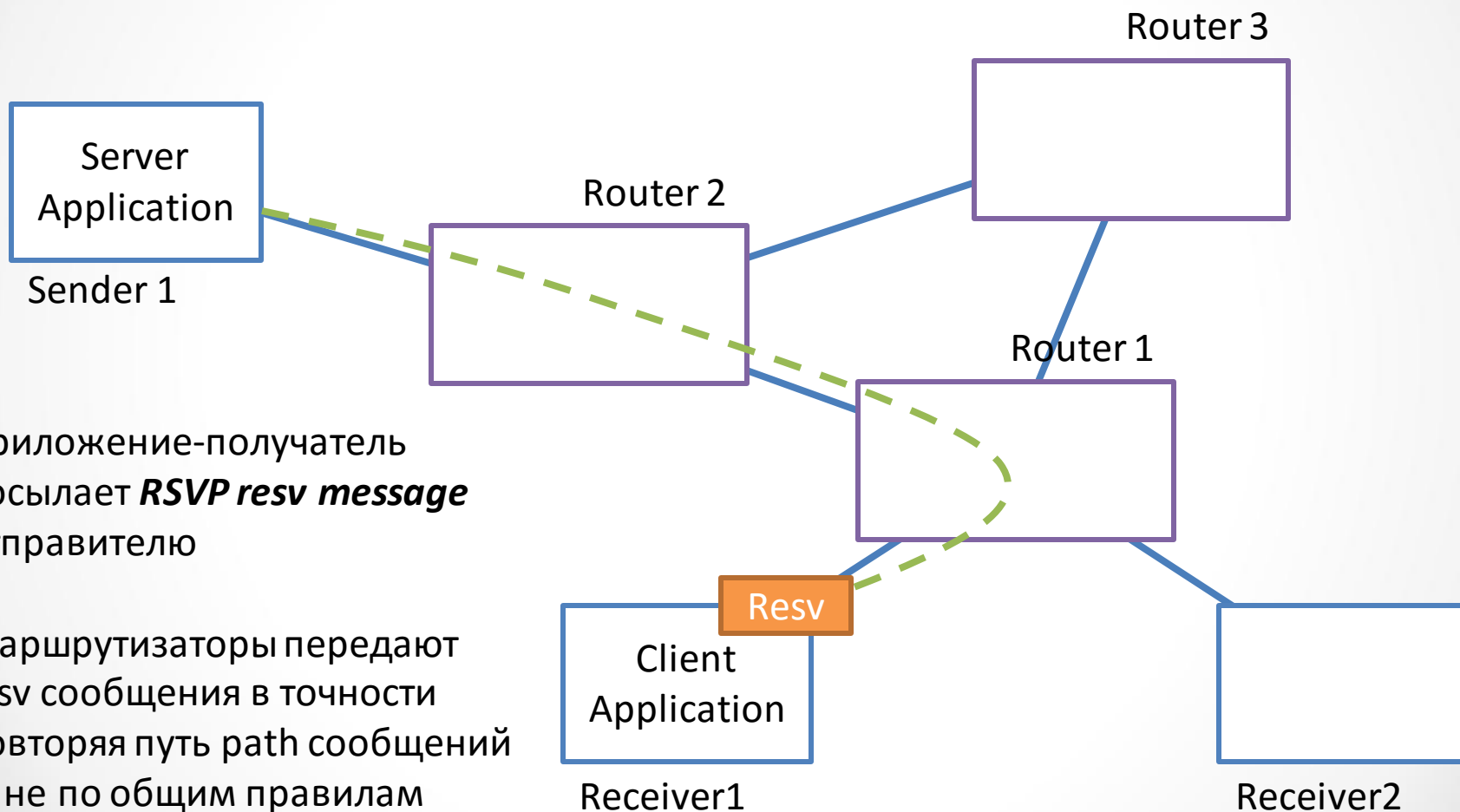
Приложение-отправитель
посылает **RSVP path message**
получателям:

- Одному (unicast)
- Группе (multicast)

Каждый маршрутизатор
запоминает, откуда пришло
сообщение

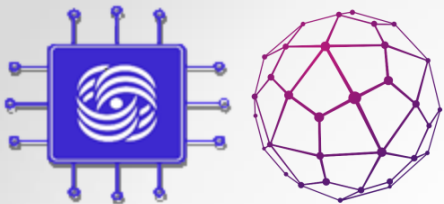


Построение маршрута

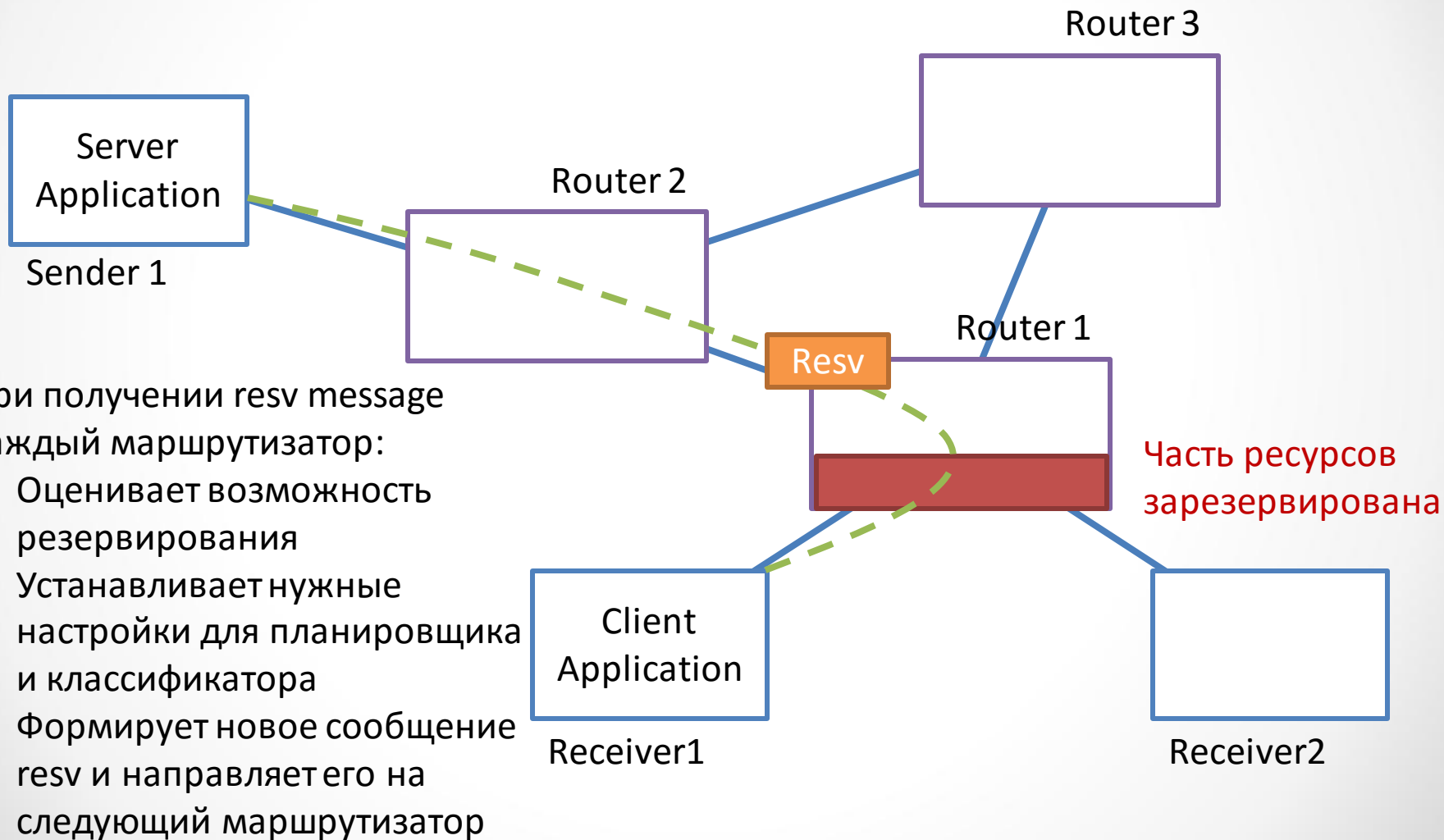


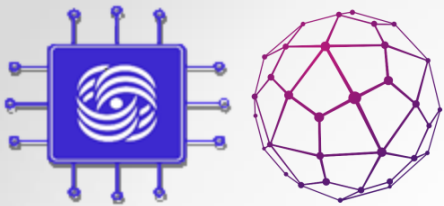
Приложение-получатель
посылает ***RSVP resv message***
отправителю

Маршрутизаторы передают resv сообщения в точности повторяя путь path сообщений (а не по общим правилам маршрутизации пакетов)

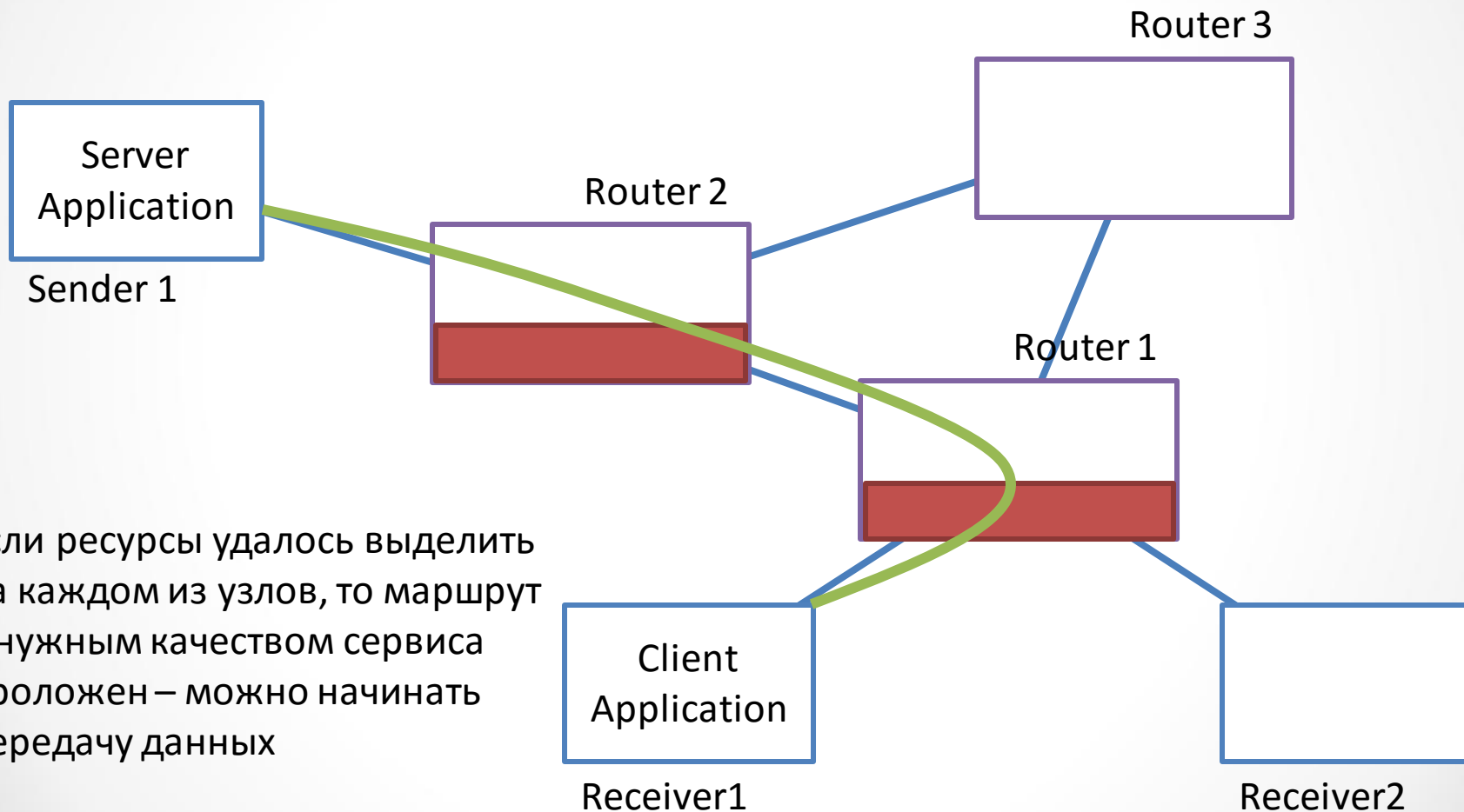


Резервирование ресурсов

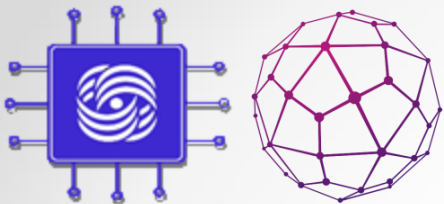




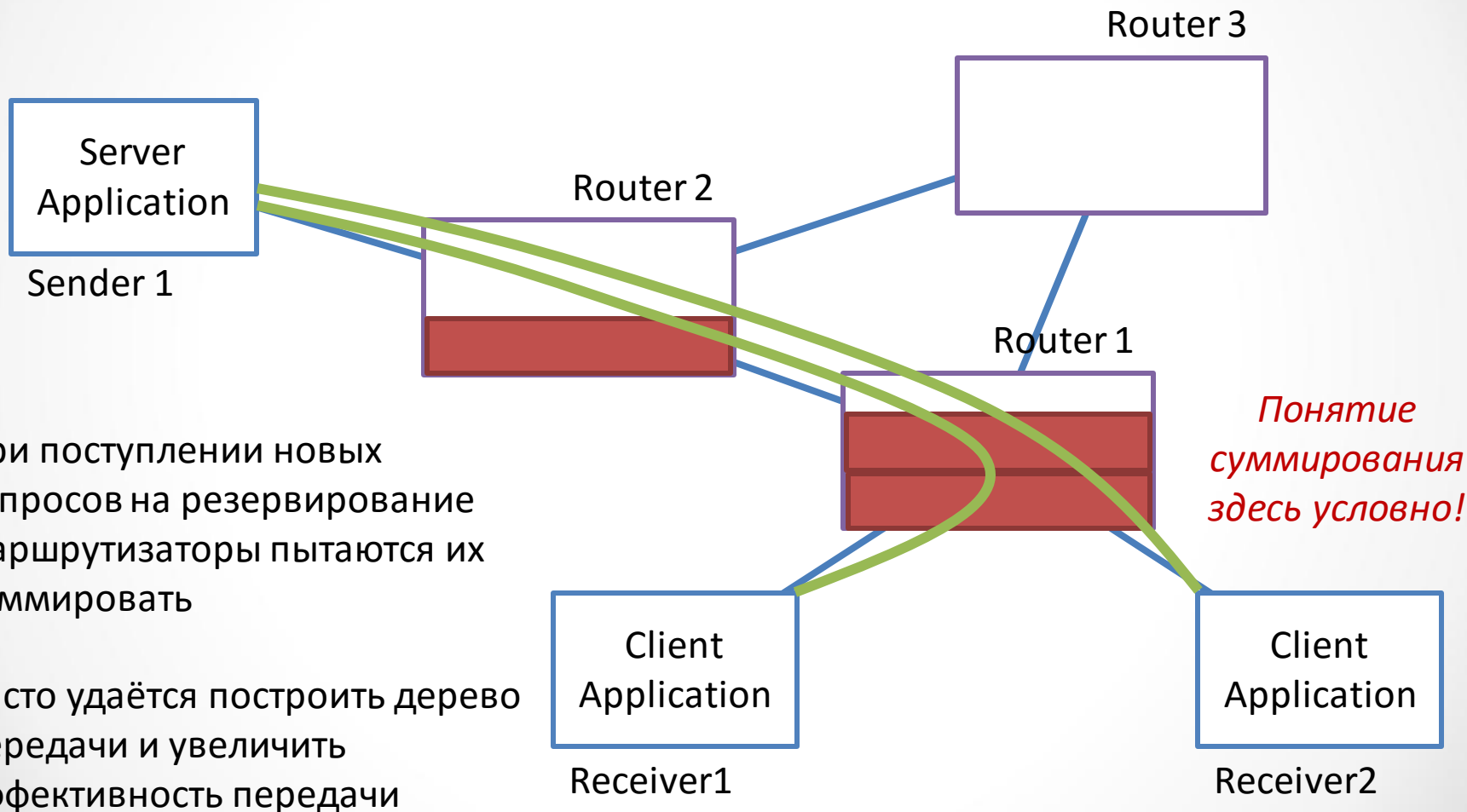
Начало передачи данных



Если ресурсы удалось выделить на каждом из узлов, то маршрут с нужным качеством сервиса проложен – можно начинать передачу данных

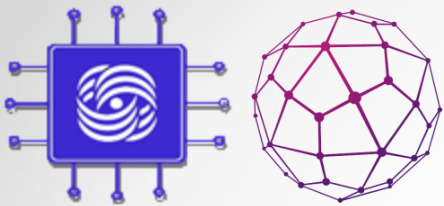


Использование filterspec



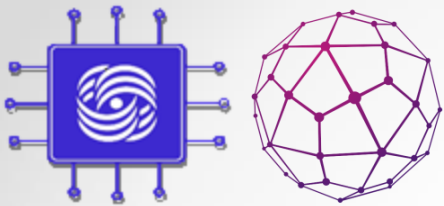
При поступлении новых запросов на резервирование маршрутизаторы пытаются их суммировать

Часто удаётся построить дерево передачи и увеличить эффективность передачи



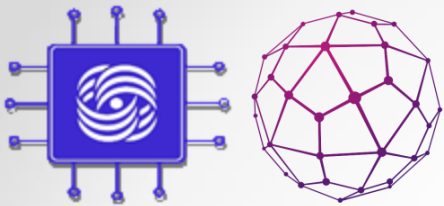
Стили резервирования FilterSpec

- Заданным фильтром (Fixed Filter)
 - Ресурсы выделяются отправителю для индивидуального пользования
 - Обычный видео стриминг
- Шаблонное (Wildcarded Filter)
 - Ресурсы разделяются между группой отправителей по заданному предикату
 - Во время аудио конференции одновременная передача данных маловероятна
- Разделяемое явно (Shared-Explicit)
 - Ресурсы разделяются между группой отправителей
 - Члены группа могут изменяться со временем



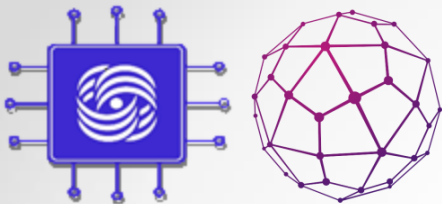
Особенности RSVP

- RSVP – **сигнальный протокол**: обеспечивает лишь резервирование вдоль маршрута: первичный выбор маршрута – забота протоколов маршрутизации
- Протокол экономно расходует ресурсы при частичном совпадении маршрутов
- Иницирует резервирование получатель
- Одни и те же ресурсы могут использоваться сразу несколькими отправителями
- Резервирование поддерживается отправителем и получателем (***Soft State***)



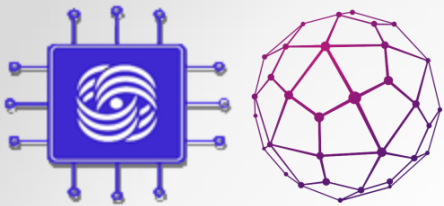
Недостатки RSVP

- Необходимо вести маршрутизацию на уровне потоков – для каждого потока нужно хранить его состояние (soft state)
- Выделение ресурсов в индивидуальном порядке слишком требовательно к железу
- Плохая масштабируемость решения
- ***Внутренняя фрагментация*** ресурсов при статическом резервировании – низкая утилизация оборудования



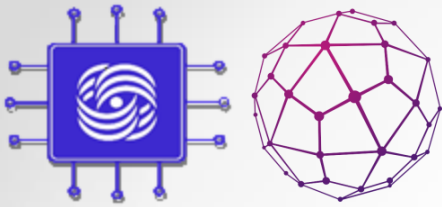
Наследие RSVP

- Способность RSVP прокладывать стабильные маршруты послужила основой для разработки **коммутации потоков**
- Протокол IFMP (1996, Ipsilon Networks) – **коммутация по тегам** в сетях ATM
- Протокол MPLS (1997, Cisco Systems) – **коммутация по меткам** в сетях IP
- Проект Ethane & протокол OpenFlow (2008)



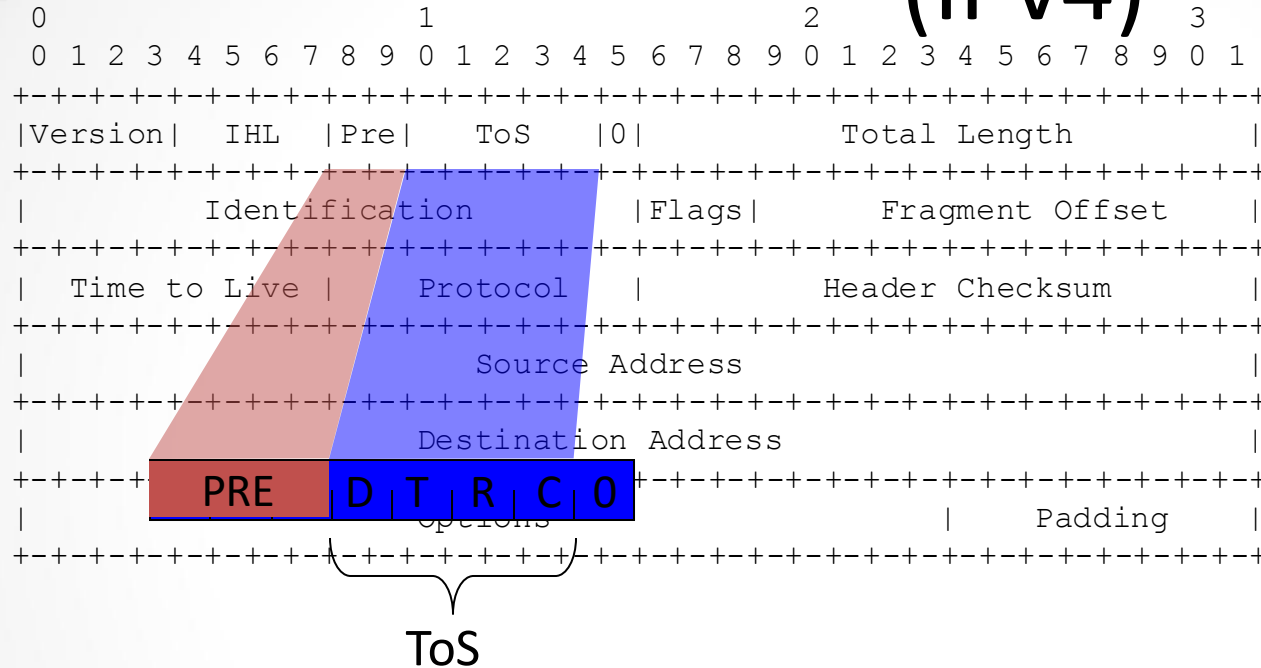
Модель Дифференцированных Сервисов (DiffServ)

- Каждый маршрутизатор имеет несколько predetermined классов обслуживания
- Пограничные маршрутизаторы определяют класс потока, маркируют его пакеты **dscp метками** и проводят **traffic conditioning** – используют инструменты policing & shaping для установки нужного профиля трафика
- На внутренних маршрутизаторах пакеты с более высоким приоритетом получают большую долю ресурсов, и наоборот



Internet Protocol version 4 (IPv4)

[RFC1349]



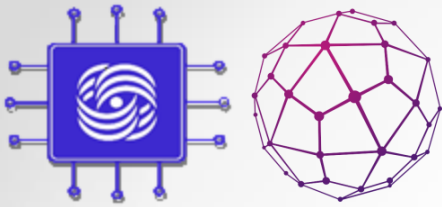
ToS

☐ Type of Service

- ☐ D – minimize delay
- ☐ T – maximize throughput
- ☐ R – maximize reliability
- ☐ C – minimize cost

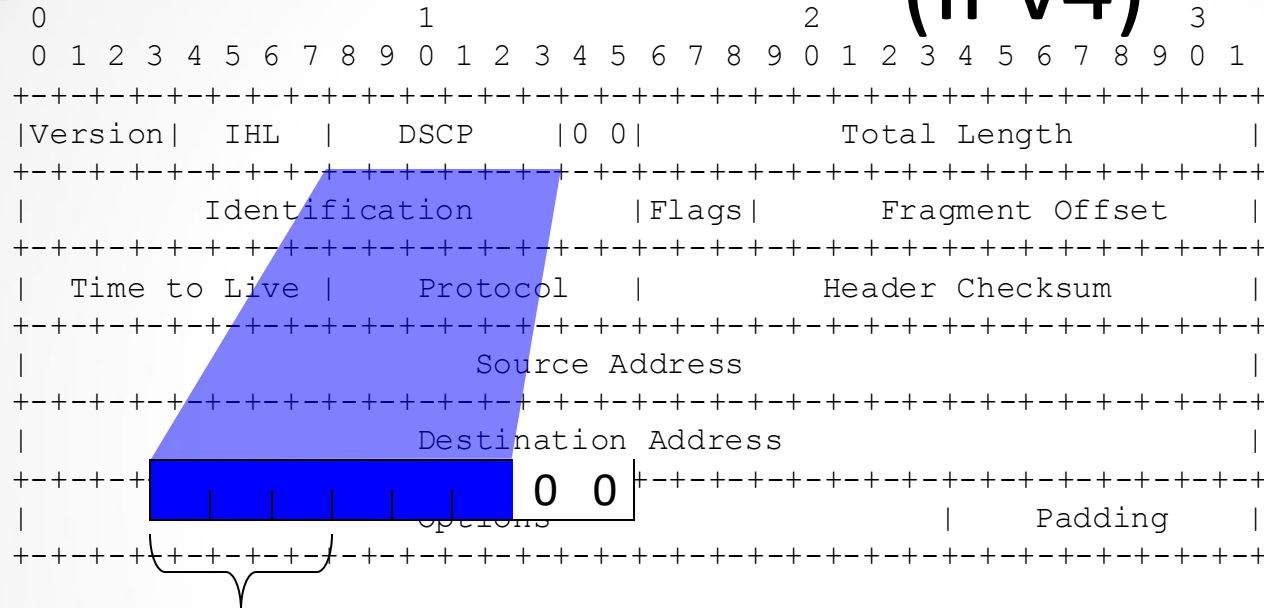
PRE

- Precedence Field
 - Priority of the packet



Internet Protocol version 4 (IPv4)

[RFC2474]



Class selector codepoints
of the form xxx000

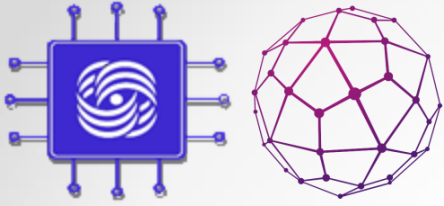
DSCP

❑ Differentiated Services Codepoint

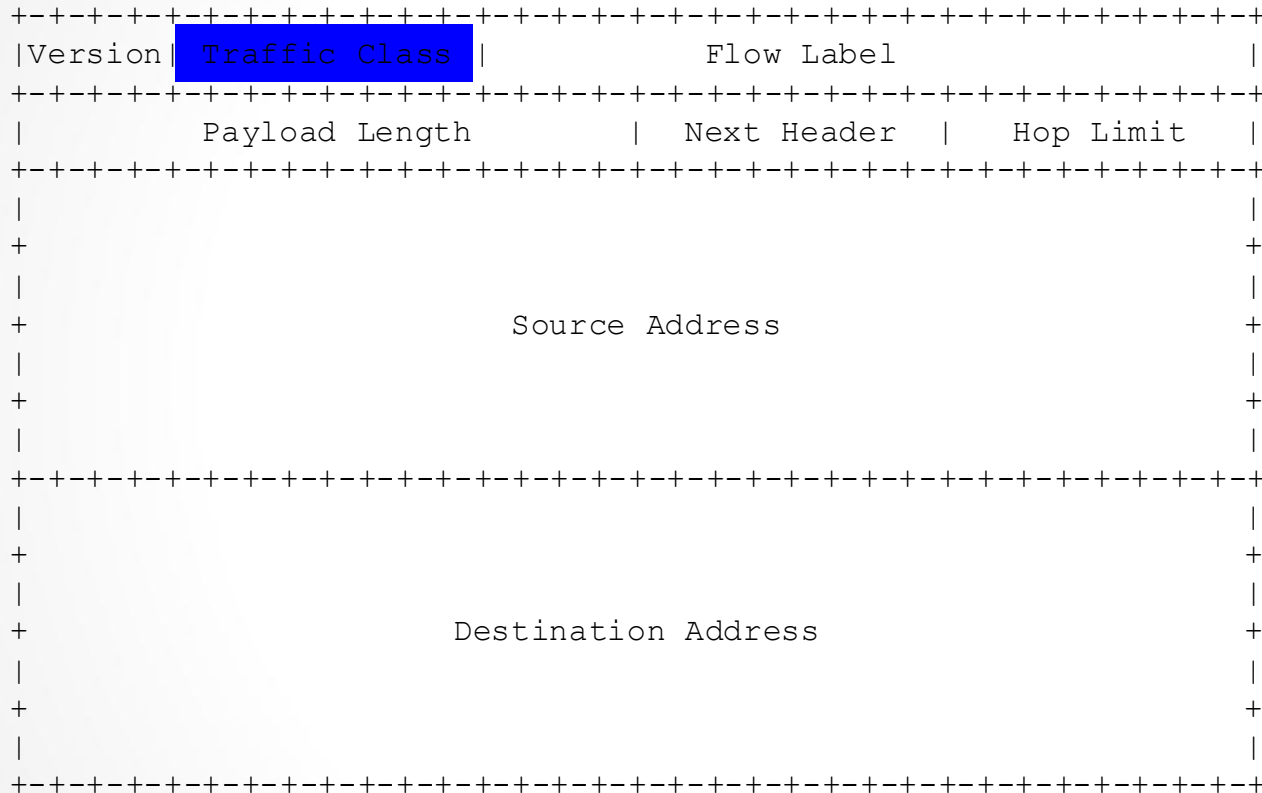
xxxxx0 reserved for standardization

xxxx11 reserved for local use

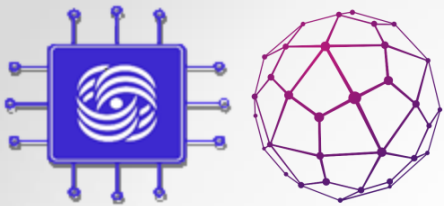
xxxx01 open for local use, may be standardized later



Internet Protocol version 6 (IPv6)

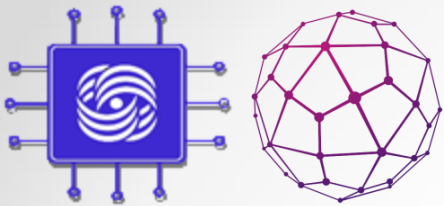


- Traffic class
 - Interpret like IPv4's DS field



Стандартные классы обслуживания DiffServ

- Default Forwarding (DF)
 - Обычно обслуживается по best-effort
- Expedient Forwarding (EF)
 - Идёт через очередь с высшим приоритетом
 - Маленькие delay, jitter & loss
- Assured Forwarding (AF)
 - Идёт через менее приоритетную очередь
 - Охватывает несколько классов с разной политикой сброса при заполнении очереди



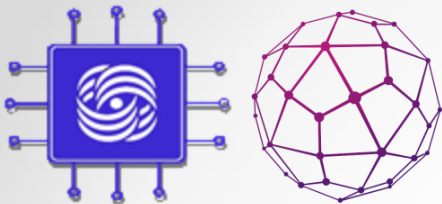
Модель DiffServ

Преимущества

- Отсутствие внутренней фрагментации
- Высокая степень утилизации оборудования
- Простота реализации в аппаратуре
- Хорошая масштабируемость

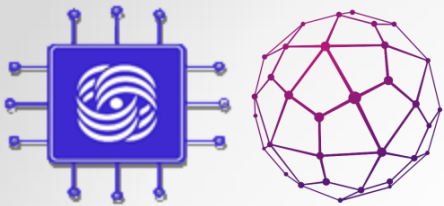
Недостатки

- Не различает метрик качества
- Не предоставляет гарантий качества
- Распределяет ресурсы пропорционально



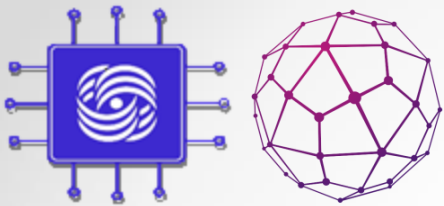
QoS Routing

- Прокладывание индивидуальных маршрутов для потоков с учётом их требований качества сервиса
 - Маршруты между двумя точками в сети могут быть построены разными способами, если их требования качества для потоков не совпадают
- Появляется возможность вовлечь ресурсы, расположенные вне основных маршрутов передачи данных
- Метод совместим с обеими моделями управления ресурсами



Проблемы QoS Routing

- Необходим гранулярный контроль за коммутационными устройствами
 - Управление на уровне потоков
- Необходимо централизованное управление
 - Алгоритмы маршрутизации не будут сходиться
- Нужна возможность получения требований качества от приложений
- Построение маршрутов – тяжёлая задача

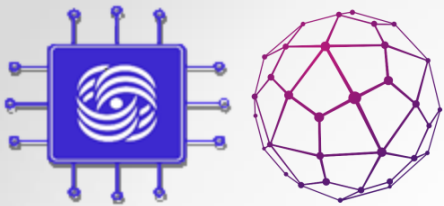


Проблема наилучшего распределения ресурсов

Управлять выполнением требований качества и уровнем утилизации сети можно за счёт особого алгоритма построения маршрутов

Критерии оптимизации:

- Увеличение устойчивости к пиковым нагрузкам – равномерное распределение
- Обслуживание потоков наименьшим количеством устройств – можно выключить часть оборудования
- Максимизация надёжности сети



Управление QoS с помощью Flow (De) Multiplexing Protocol

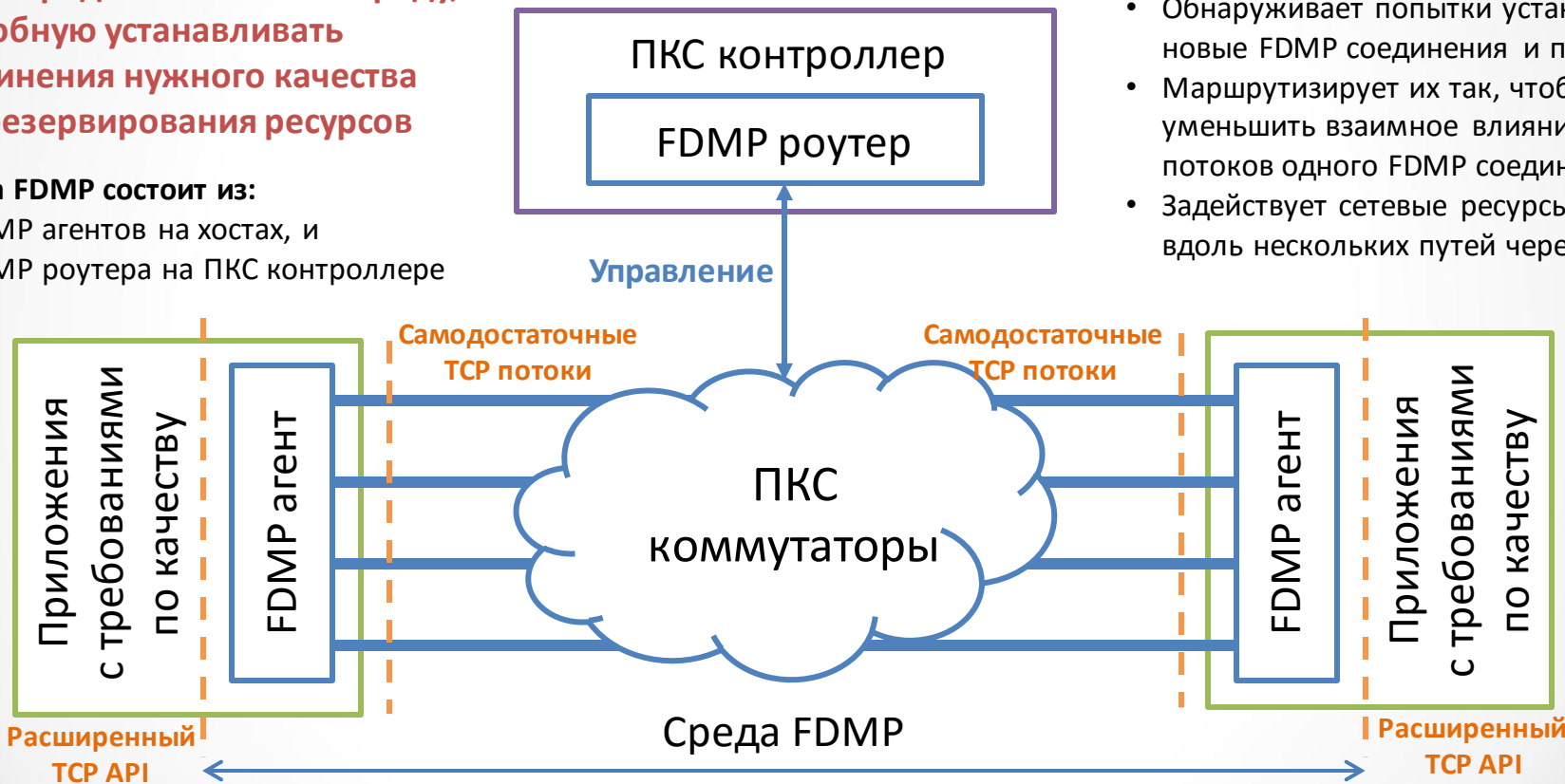
FDMP представляет собой среду, способную устанавливать соединения нужного качества без резервирования ресурсов

Среда FDMP состоит из:

- FDMP агентов на хостах, и
- FDMP роутера на ПКС контроллере

FDMP роутер:

- Обнаруживает попытки установить новые FDMP соединения и потоки
- Маршрутизирует их так, чтобы уменьшить взаимное влияние потоков одного FDMP соединения
- Задействует сетевые ресурсы вдоль нескольких путей через сеть



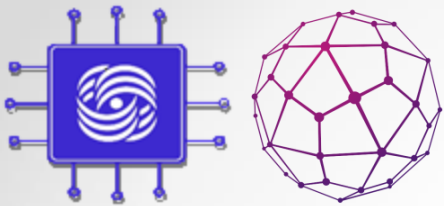
FDMP агент на стороне отправителя:

Разделяет поток байтов от отправителя на несколько TCP потоков и передаёт их независимо друг от друга

Число TCP потоков изменяется динамически в зависимости от актуального состояния сети и текущих требований приложений

FDMP агент на стороне получателя:

Собирает из пакетов, приходящих через TCP потоки от отправителя, оригинальный поток байтов



Демонстрация FDMP

Динамическая адаптация соединения к перегрузкам в сети:
пропускная способность от h1 к h2 не меньше 80 Mbps

