



# Программные методы управления сетями передачи и обработки данных (СПОД)

Пашков Василий Николаевич

Кафедра АСВК, ВМК МГУ

# Проблемы традиционных сетей



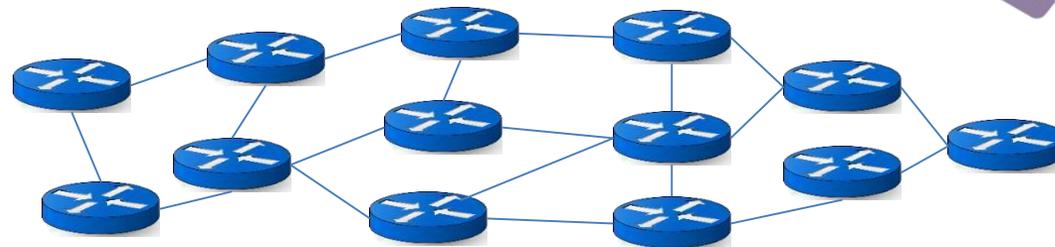
Функция

...

Функция

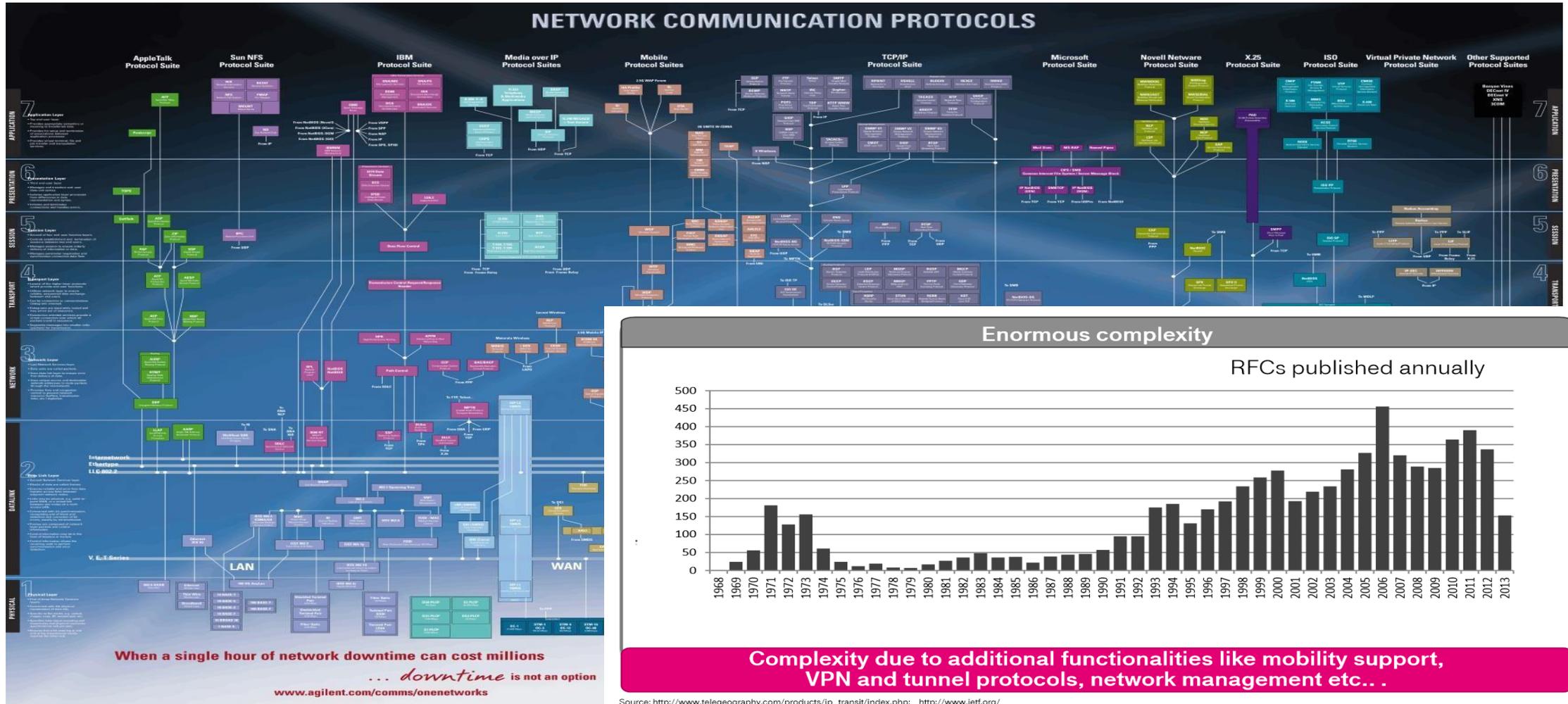
**Операционная  
система**

**Специальное  
устройство передачи  
данных**



- Зависимость от производителя
- Ошибки в реализациях сетевых протоколов
- Миллионы строк закрытого проприетарного кода (6000+ RFC)
- Высокая стоимость оборудования
- Высокая стоимость эксплуатации
- Сложность управления большими сетями
- Сложность отладки
- “Закрытость” оборудования и программного обеспечения
- Сложность внедрения новых идей
- Неэффективность использования аппаратных ресурсов, энергоэффективность

# Постоянный рост сложности

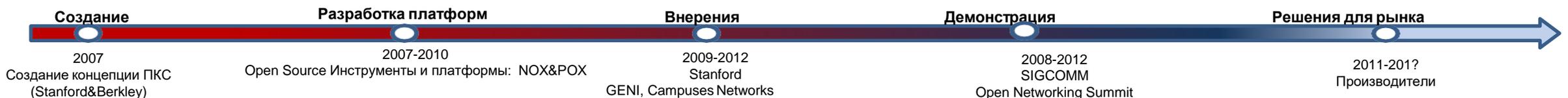




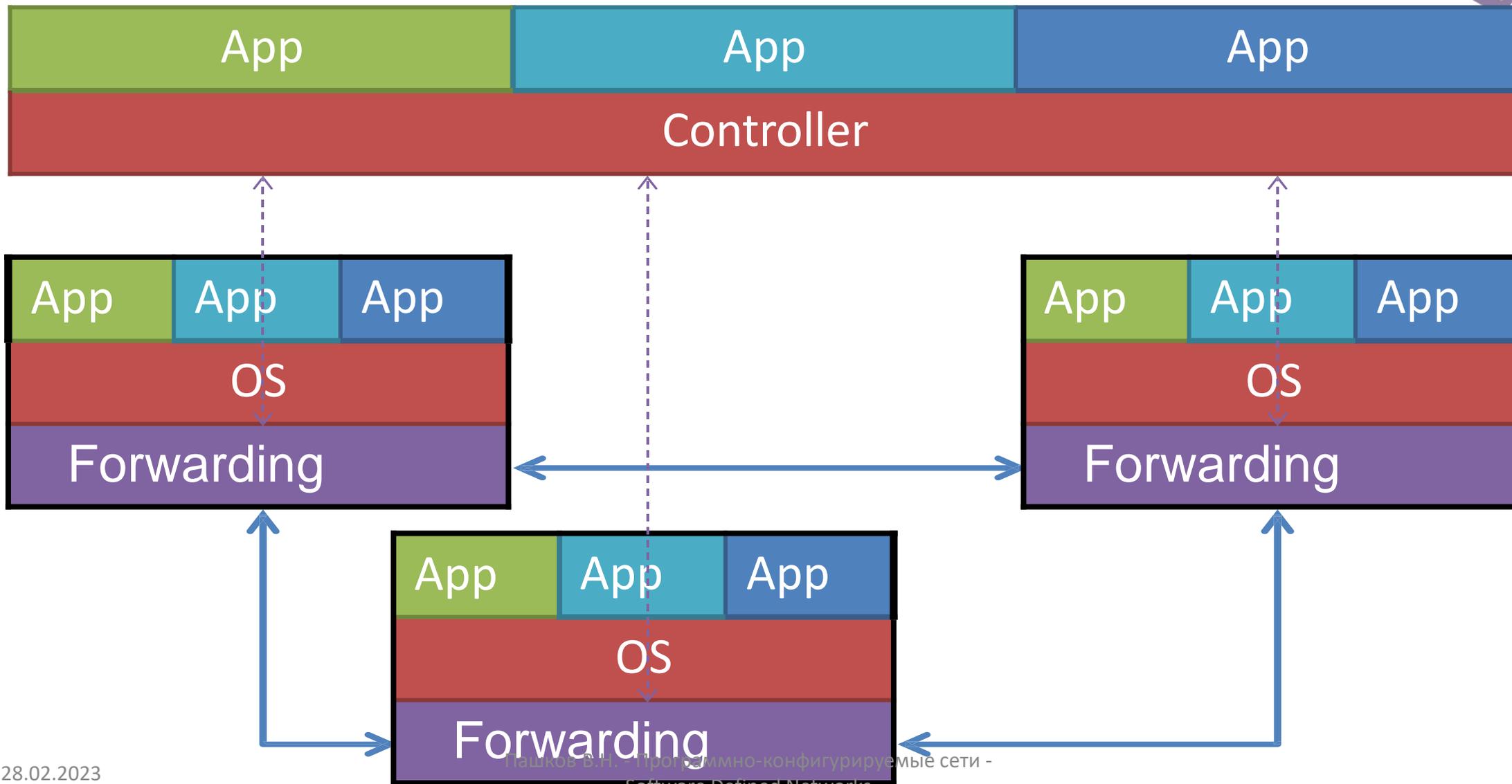
# Основные принципы SDN

**Программно-Конфигурируемые Сети (Software Defined Networking/SDN)** – это разделение контура передачи данных и контура управления данными, что позволяет осуществлять программное управление сетевым оборудованием и потоками данных в сети.

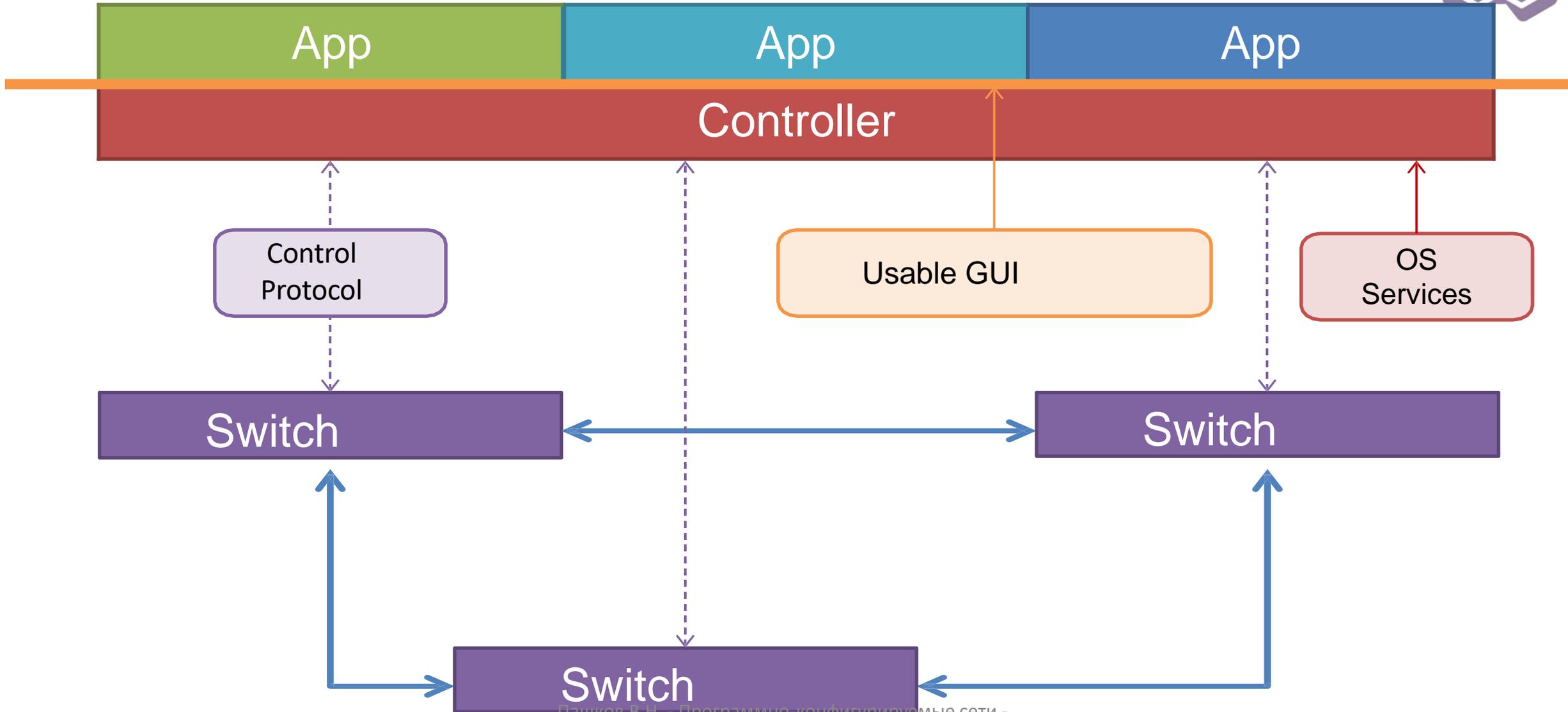
- 1. Физическое разделение контуров:** контура управления сетевым оборудованием и контура передачи данных
2. Переход от управления отдельными экземплярами сетевого оборудования к управлению сетью в целом – **логически централизованное управление**
- 3. Открытый унифицированный программно-управляемый интерфейс** между сетевыми приложениями и транспортной сетью



# Переход к SDN

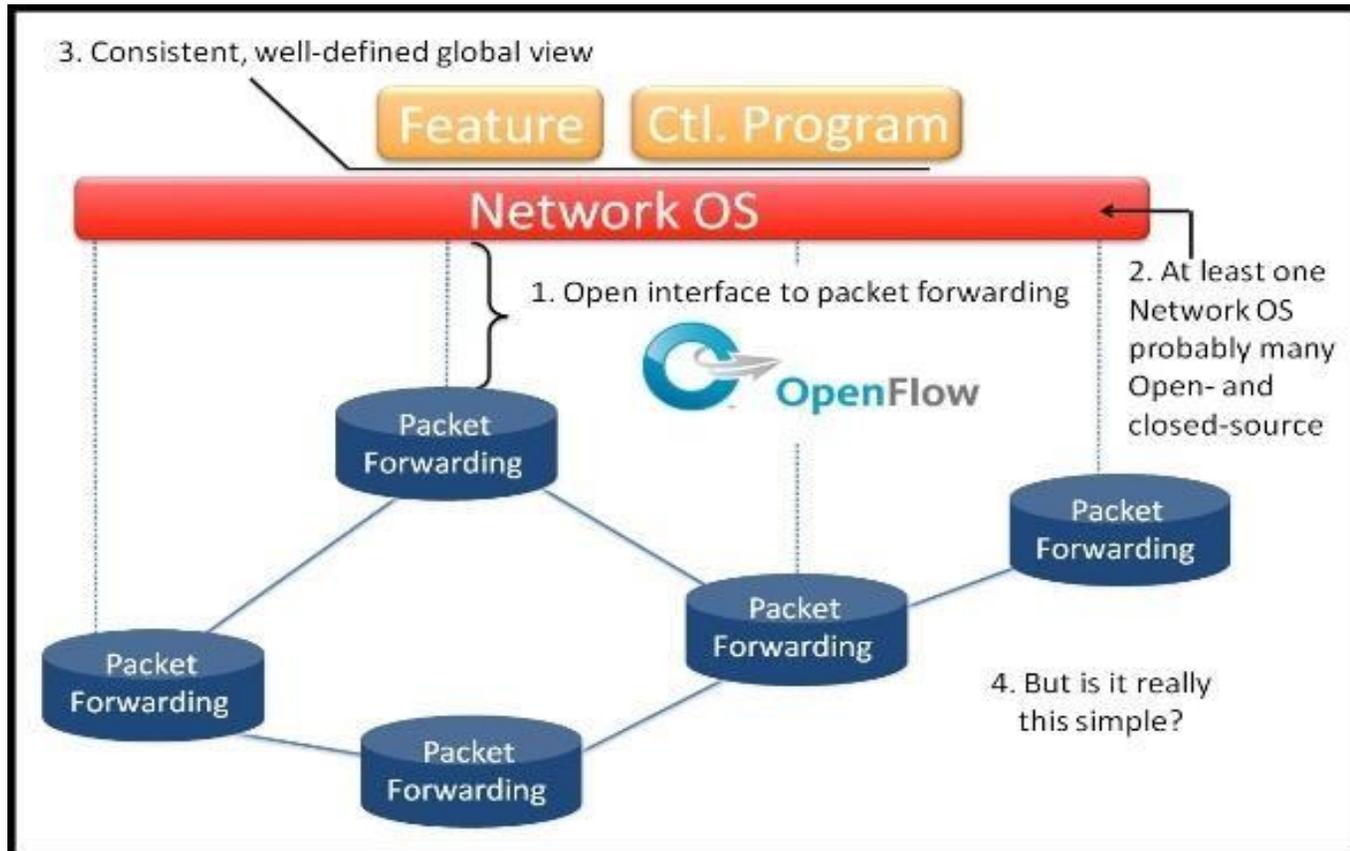


# Архитектура SDN





# Достоинства ПКС



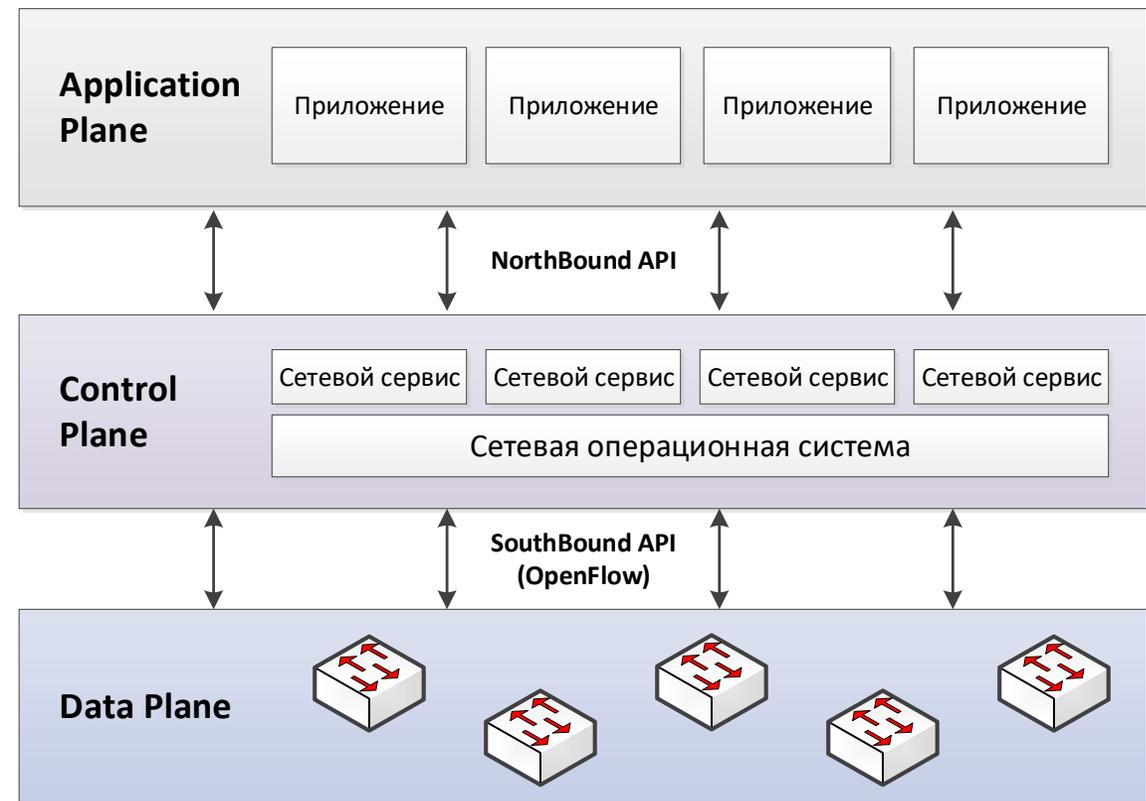
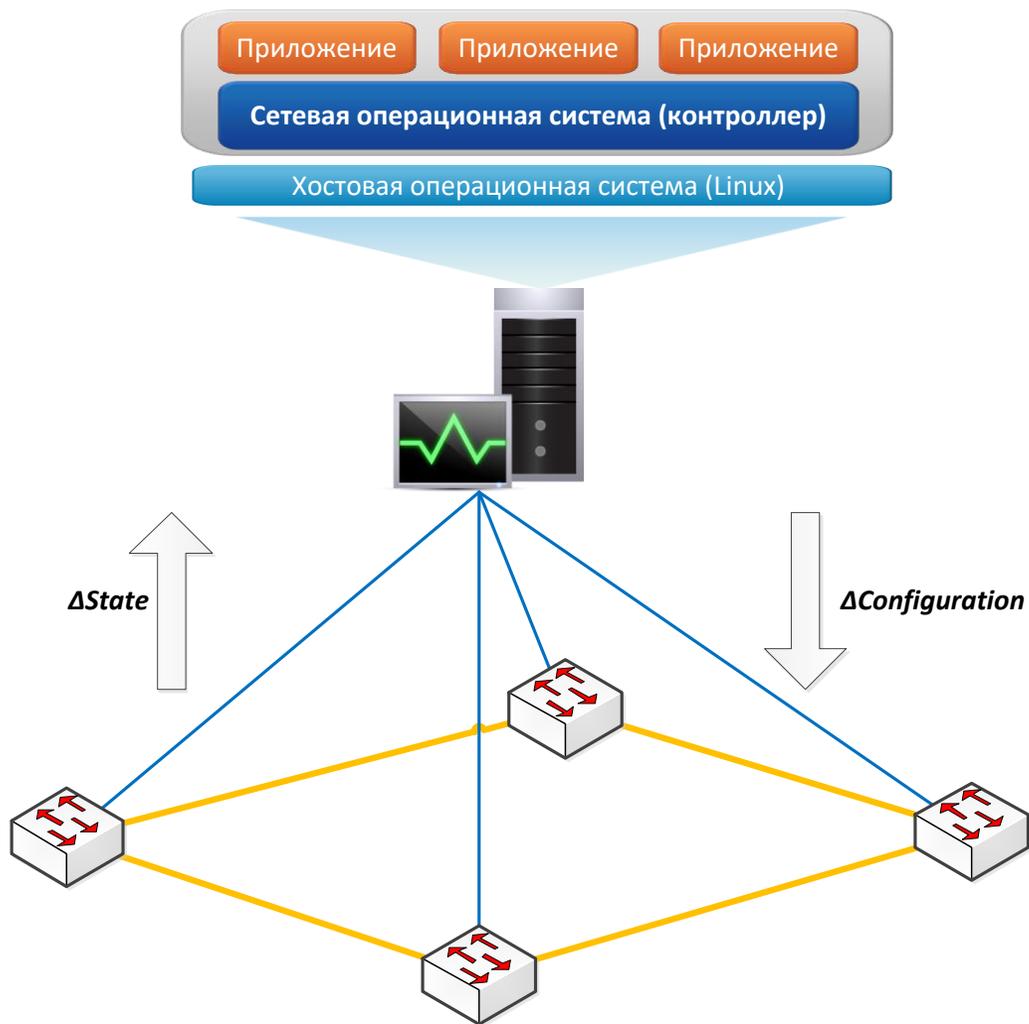
- Гибкость управления и высокая скорость реакции на события
- Программируемость
- Снижение стоимости сетевого оборудования (CAPEX)
- Снижение стоимости обслуживания сетей (OPEX)
- Открытость к инновациям, возможность создавать новые сетевые сервисы



# Как работает SDN?



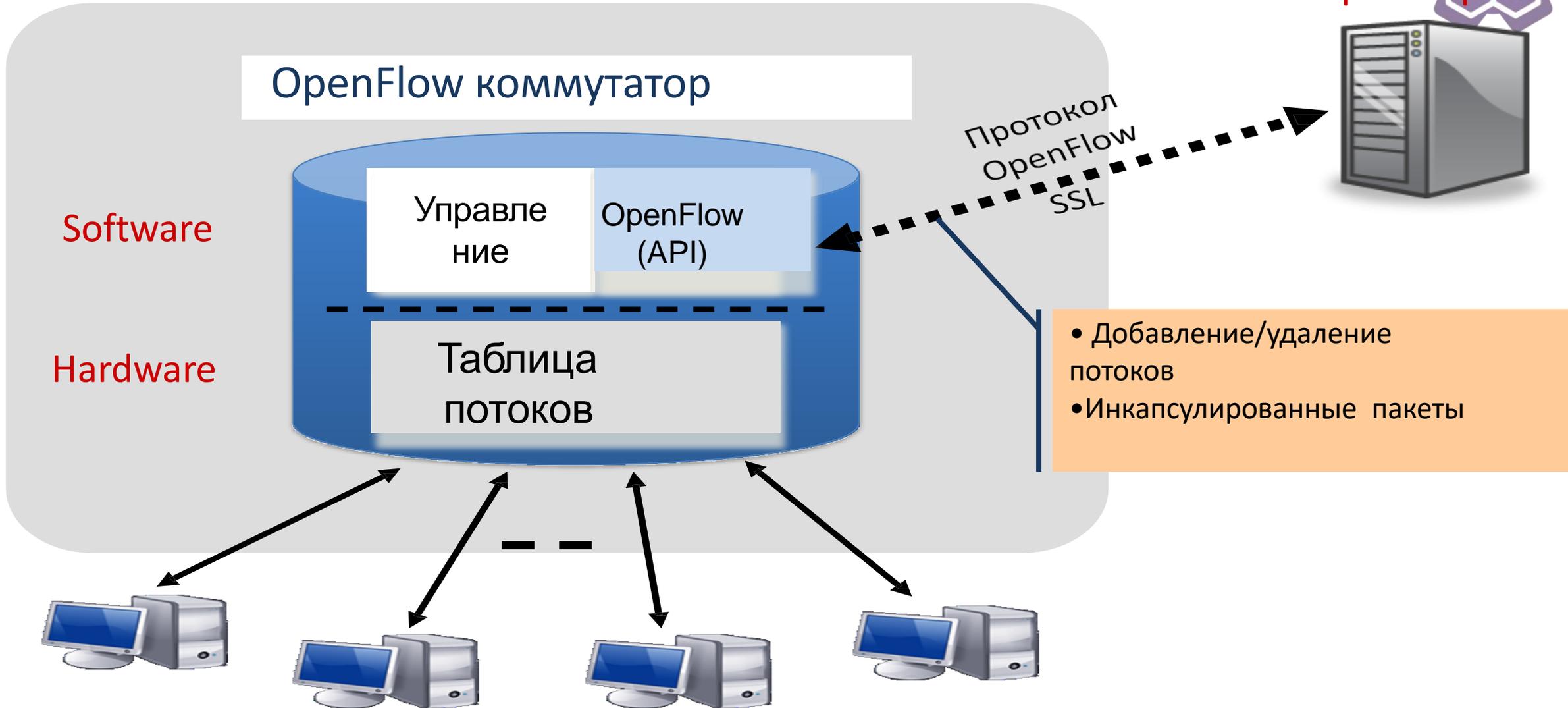
# SDN



# OpenFlow



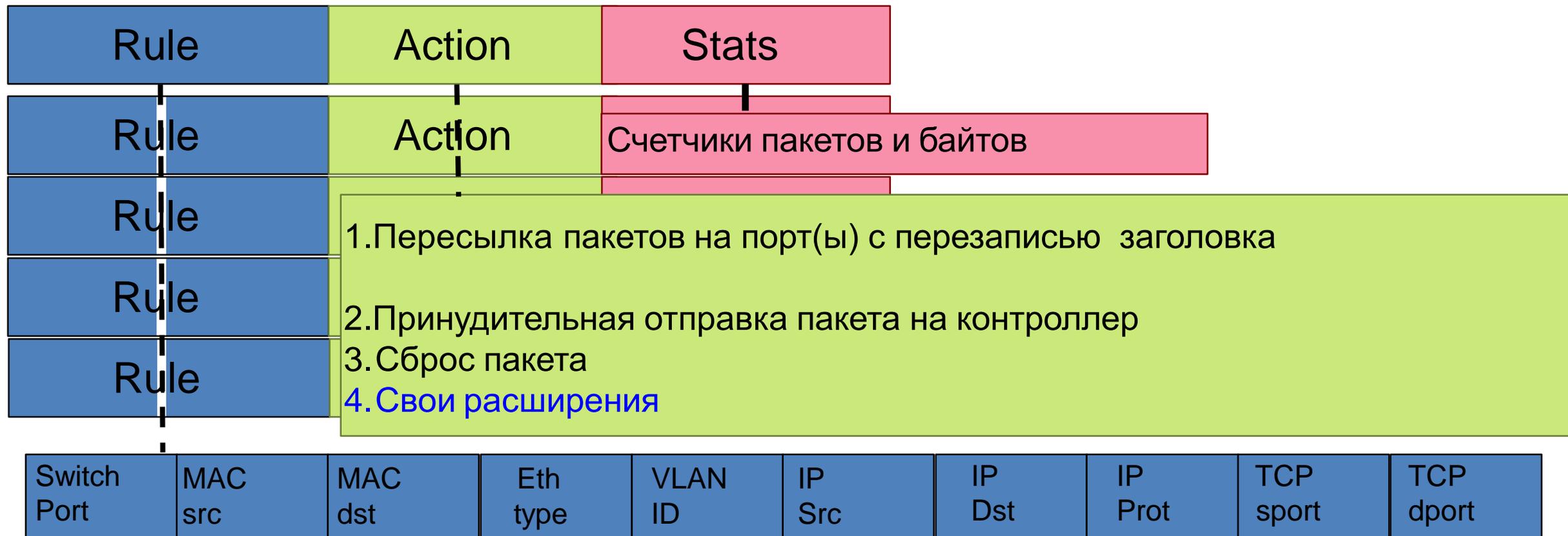
Контроллер





# OpenFlow 1.0

## Flow Table



+ маска по полям



# Примеры правил OpenFlow

## Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f:...	*	*	*	*	*	*	*	port6

## Flow Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
port3	00:20..	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6

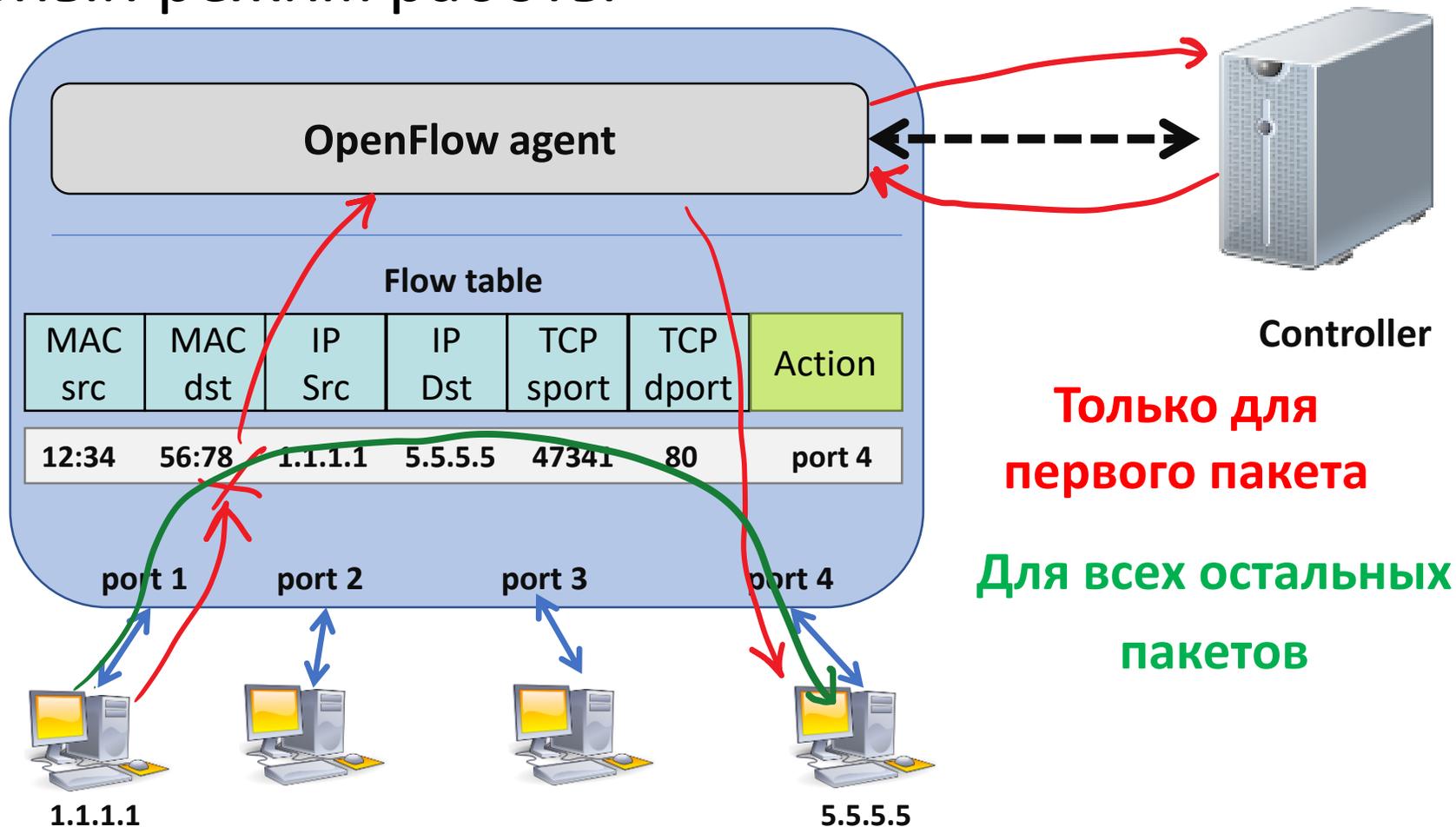
## Firewall

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	*	*	*	22	drop



# Схема работы OpenFlow

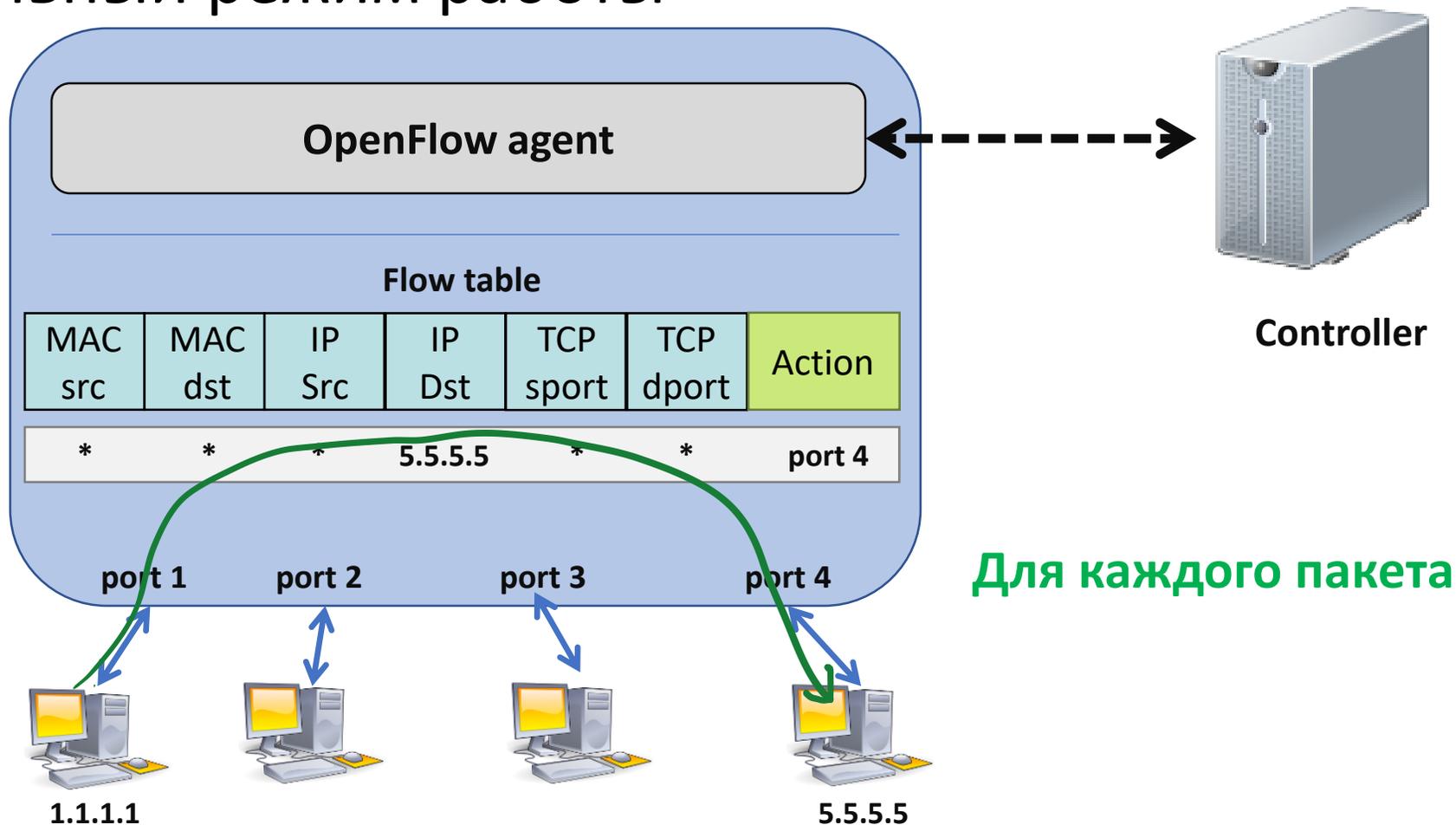
## Реактивный режим работы





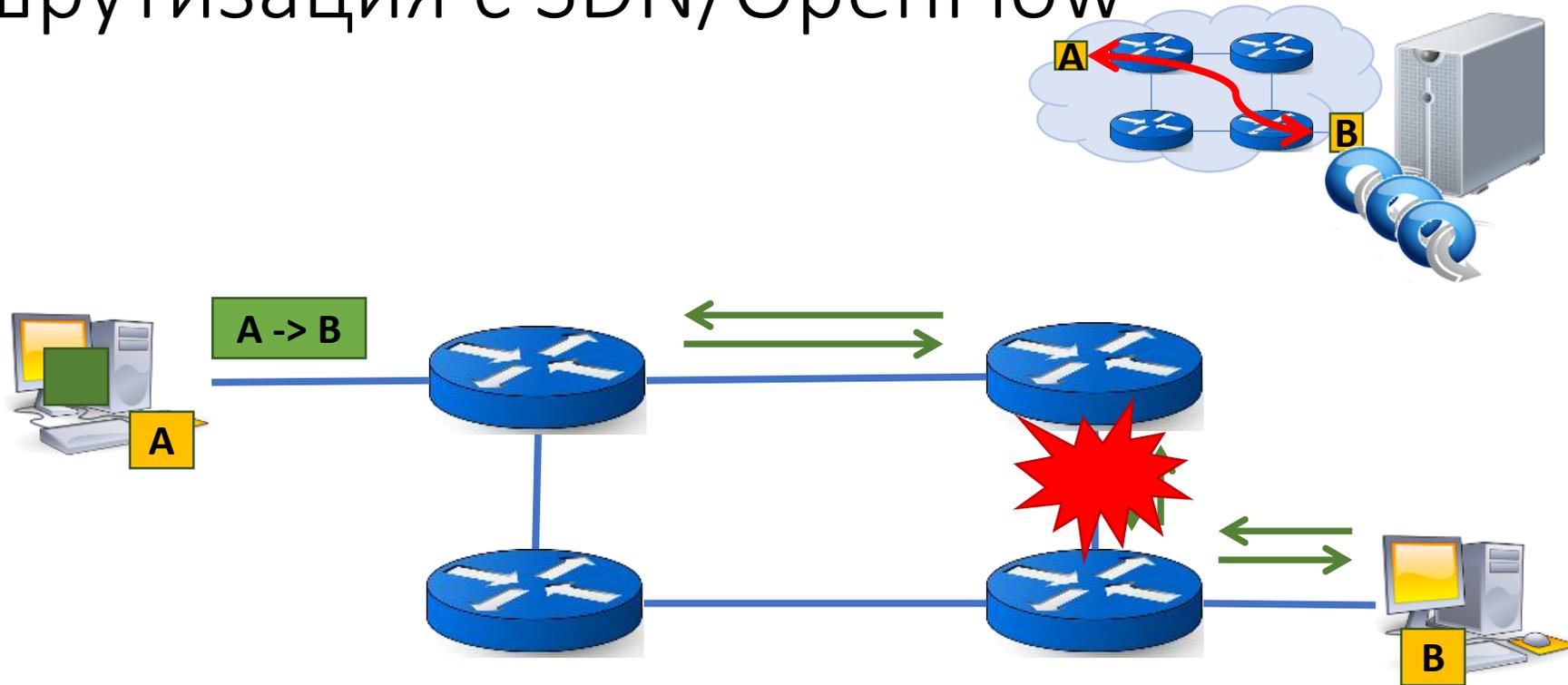
# Схема работы OpenFlow

## Проактивный режим работы





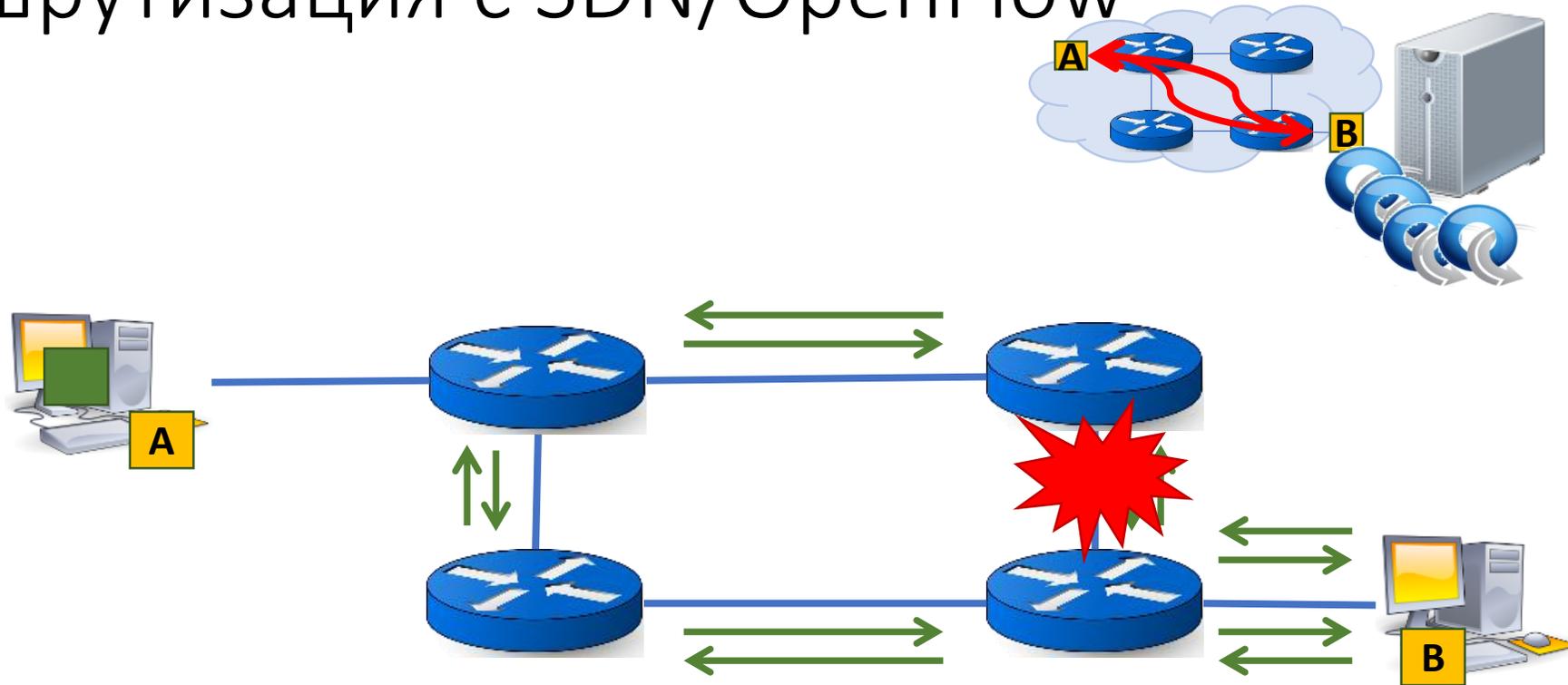
# Маршрутизация с SDN/OpenFlow



- Известный пакет отправляется на контроллер (OF\_PACKET\_IN).
- Контроллер вычисляет лучший маршрут через всю сеть (с наименьшей стоимостью и удовлетворяющий политикам маршрутизации).
- Соответствующие правила OpenFlow устанавливаются на коммутаторы + сразу и обратный маршрут (OF\_PACKET\_OUT/FLOW\_MOD).



# Маршрутизация с SDN/OpenFlow



- Известный пакет отправляется на контроллер (OF\_PACKET\_IN).
- Контроллер вычисляет лучший маршрут через всю сеть (с наименьшей стоимостью и удовлетворяющий политикам маршрутизации).
- Соответствующие правила OpenFlow устанавливаются на коммутаторы + сразу и обратный маршрут (OF\_PACKET\_OUT/FLOW\_MOD).
- **Динамическая переконфигурация в случае ошибки сети.**

# SDN/OpenFlow контроллеры

- Большое количество реализаций:
  - Nox, Pox, MUI, Ryu, Beacon, OpenDaylight, Floodlight, Maestro, McNettle, Flower, **Runos**
- Для образовательных целей - POX
- Два больших комьюнити
  - ONOS (Stanford)
  - OpenDayLight (Cisco)
- В России – контроллер Runos
  - [arccn.github.io/runos](https://arccn.github.io/runos)



Floodlight





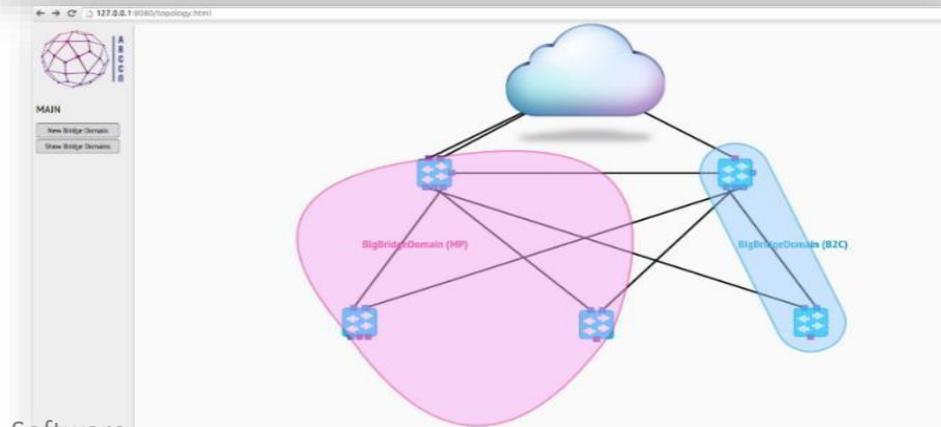
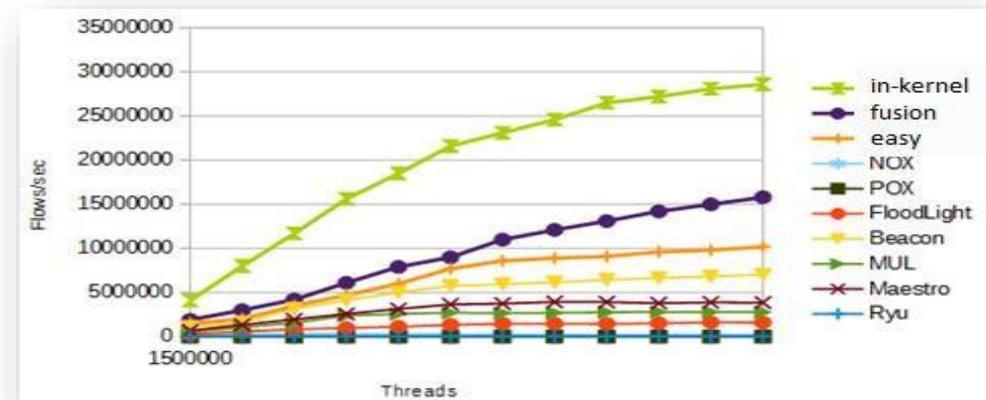
# Сетевая операционная система RUNOS 2.0



Система управления сетью первый российский SDN-контроллер RUNOS  
*Russian Network Operation System*

Есть разные варианты контроллера с единой базой и различным набором сервисов и приложений

- **Открытая версия на Github** <http://arccn.github.io/runos/>
  - Своя база на C++11/14, а не Java
  - цель: упростить разработку сетевых приложений и не забывать о производительности
  - приложения: топология, маршрут, перестроение в случае обрыва, REST, WebUI, проактивная загрузка правил, резервирование Active-Passive
- **Внутренняя ядерная версия**
  - Супер-производительность 30 млн событий в секунду
  - Разработка приложений под заказчика
- **Внутренняя версия с приложениями под оператора связи**
  - База такая же, как и на Github. Заказчики сами могут разрабатывать приложения. Учиться по доступным материалам
  - Сервисы B2C, B2B (p2p, mp2mp, multicast, и т.п.)
  - Active-Standby режим



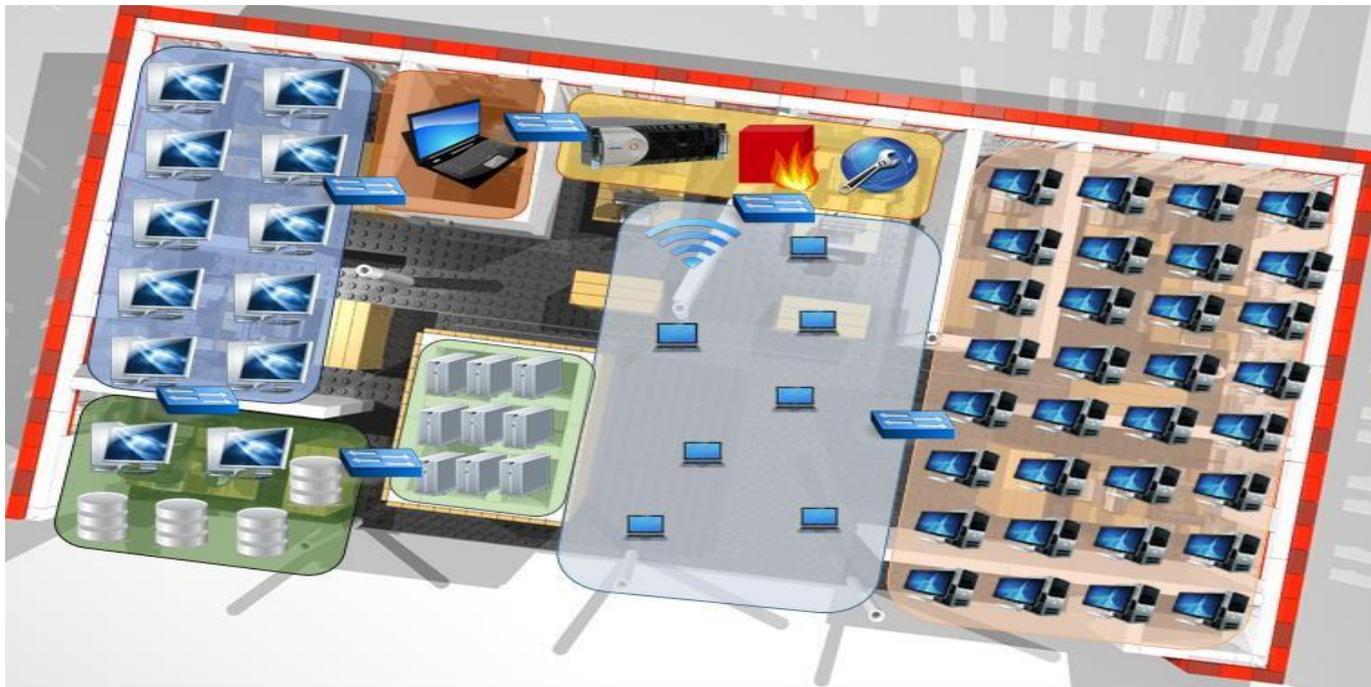


# Примеры применения SDN и задачи управления в корпоративных сетях, сетях операторов, сетях ЦОД



# SDN в корпоративных сетях

- Современные компания имеют сложную сетевую инфраструктуру:
  - Большое количество сетевых элементов
  - Разветвленная топология
  - Набор различных политик маршрутизации и безопасности



Централизованное управление политиками маршрутизации и политиками безопасности



# Трудности администрирования

- Сетевые администраторы отвечают за поддержание работы сетевой инфраструктуры:
  - Сетевые инженеры руками переводят высокоуровневые политики в низкоуровневые команды
  - Ручная настройка всех сетевых устройств
  - Ограниченный инструментарий по управлению сетевыми устройствами
  - Переучивание под каждого вендора
- Существуют т.н. системы управления по SNMP, но только мониторинг состояния, а не управление – настройка все равно в ручном режиме.

```
Router Management
  1.  Configure Static-routes/ACLs
  2.  Configure RIP
  3.  Configure OSPF
  0.  Exit

Select Menu Number [0-3]: 1

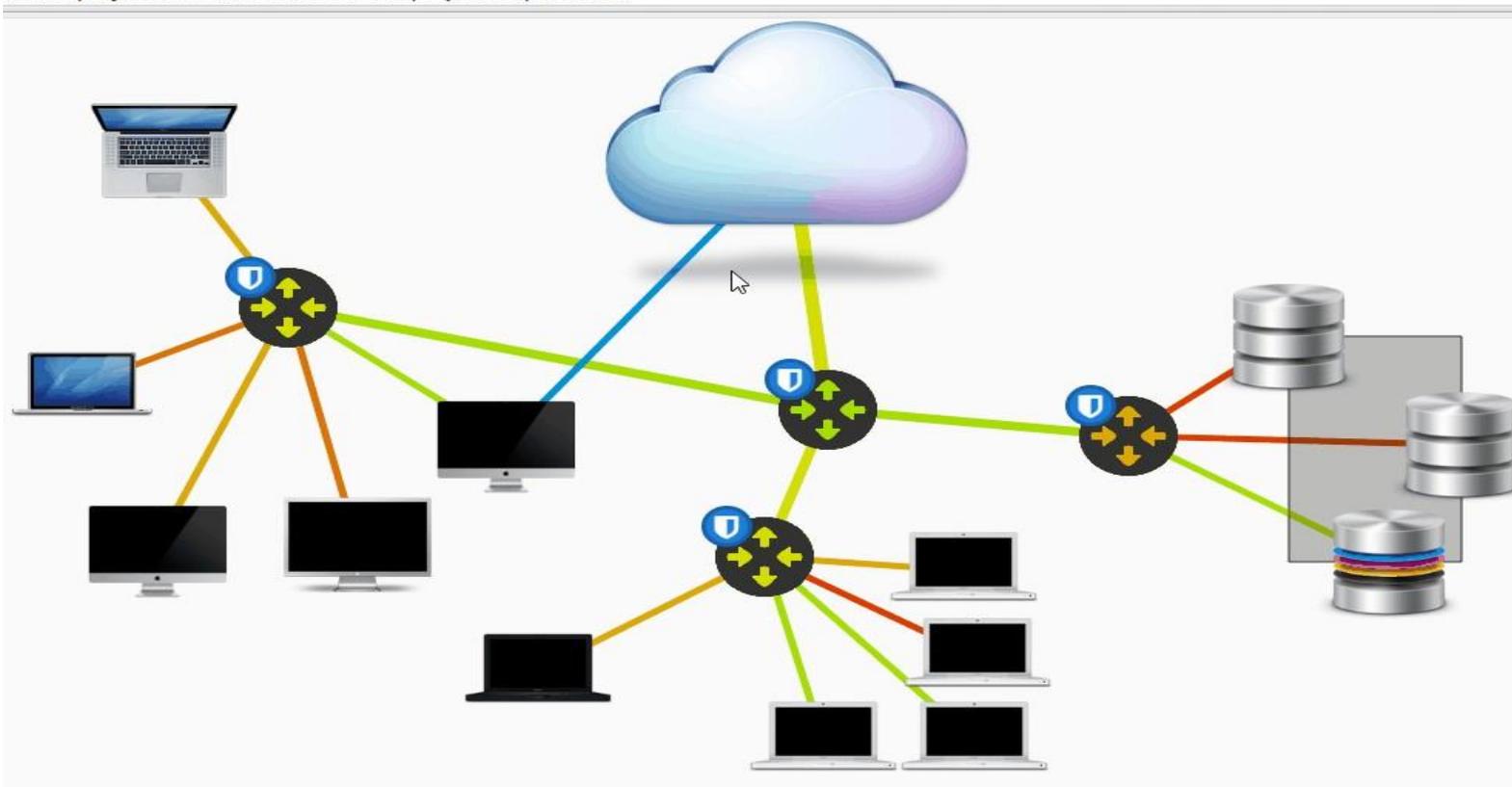
router> enable
router# configure terminal
router(config)# ip route 5.5.5.5 255.255.255.255 2.2.2.2
router(config)# write
Configuration saved...
router(config)# _
```

Автоматизация управления,  
конфигурирования и  
настройки сетевого  
оборудования



# Мониторинг и управление из единого центра

science/projects/arccn/2015/ross15/deploy/enterprise.html



Централизованный мониторинг состояния сети и контроль использования ресурсов сети

Централизованное управления сетевыми устройствами и потоками данных

Быстрая/автоматическая реакция на отказы, перегрузки и другие события в сети



# Возникающие задачи

- **Разработка/доработка существующих контроллеров**
- **Проектирование контура управления SDN сети с учетом требований заказчика**
- **Разработка приложений для контроллеров и соответствующих алгоритмов и методов:**
  - Разработка приложений маршрутизации в SDN/OpenFlow сетях.
  - Разработка приложений для управления трафиком IoT устройств в SDN/OpenFlow сетях.
  - Разработка приложений мониторинга состояния контура передачи данных и контура управления в SDN/OpenFlow сетях.
  - Разработка приложений ядра контроллера по поддержке PCEP протокола взаимодействия с контроллером.
  - Разработка приложений ядра контроллера по поддержке NETCONF протокола взаимодействия с контроллером.
  - Разработка приложений балансировки нагрузки в контуре передачи данных SDN сетей.
  - Разработка приложений обеспечения отказоустойчивого подключения коммутаторов к контроллеру через контур передачи данных.
  - Разработка приложений обнаружения и противодействия DDoS атакам на контроллер.

# Пример 1: разработка приложения для контроллера



```
of13::FlowMod fm2; // Table 0: in_port,VLAN=ar.ep.stag -> output(ar.ep.port)
fm2.table_id(FORWARDING_TABLE);
fm2.priority(100);
fm2.cookie(cookie);
fm2.xid(mgr->impl->xid);
fm2.buffer_id(OFP_NO_BUFFER);
fm2.add_oxm_field(new of13::InPort(main_route[0].port));
fm2.add_oxm_field(new of13::VLANVid(end_switch_list[0].ep_list[0].stag | of13::OFPVID_PRESENT));
of13::ApplyActions acts2;
acts2.add_action(new of13::OutputAction(end_switch_list[0].ep_list[0].port, 0));
fm2.add_instruction(acts2);
mgr->get_connection(sw1_dp_id)->send(fm2);

/* DR */
/* rules for the last one switch in the route */
of13::FlowMod fm3; // Table 0: VLAN=ar.ep.stag -> META=bbd_id, GOTO 1
fm3.table_id(FORWARDING_TABLE);
fm3.priority(100);
fm3.cookie(cookie);
fm3.xid(mgr->impl->xid);
fm3.buffer_id(OFP_NO_BUFFER);
fm3.add_oxm_field(new of13::VLANVid(end_switch_list[0].ep_list[0].stag | of13::OFPVID_PRESENT));
fm3.add_instruction(new of13::WriteMetadata(domain_id, 0xFFFF));
fm3.add_instruction(new of13::GoToTable(LEARNING_TABLE));
mgr->get_connection(sw2_dp_id)->send(fm3);

// Table 1: META=bbd_id, in_port=dr.ep0.port -> TO_CONTROLLER
for (auto ep : end_switch_list[1].ep_list) {
of13::FlowMod fm4; // Table 1: META=bbd_id, in_port=dr.ep0.port -> TO_CONTROLLER
fm4.table_id(LEARNING_TABLE);
fm4.priority(100);
fm4.cookie(cookie);
fm4.xid(mgr->impl->xid);
fm4.buffer_id(OFP_NO_BUFFER);
fm4.add_oxm_field(new of13::Metadata(domain_id, 0xFFFF));
fm4.add_oxm_field(new of13::InPort(ep.port));
of13::ApplyActions acts4;
```

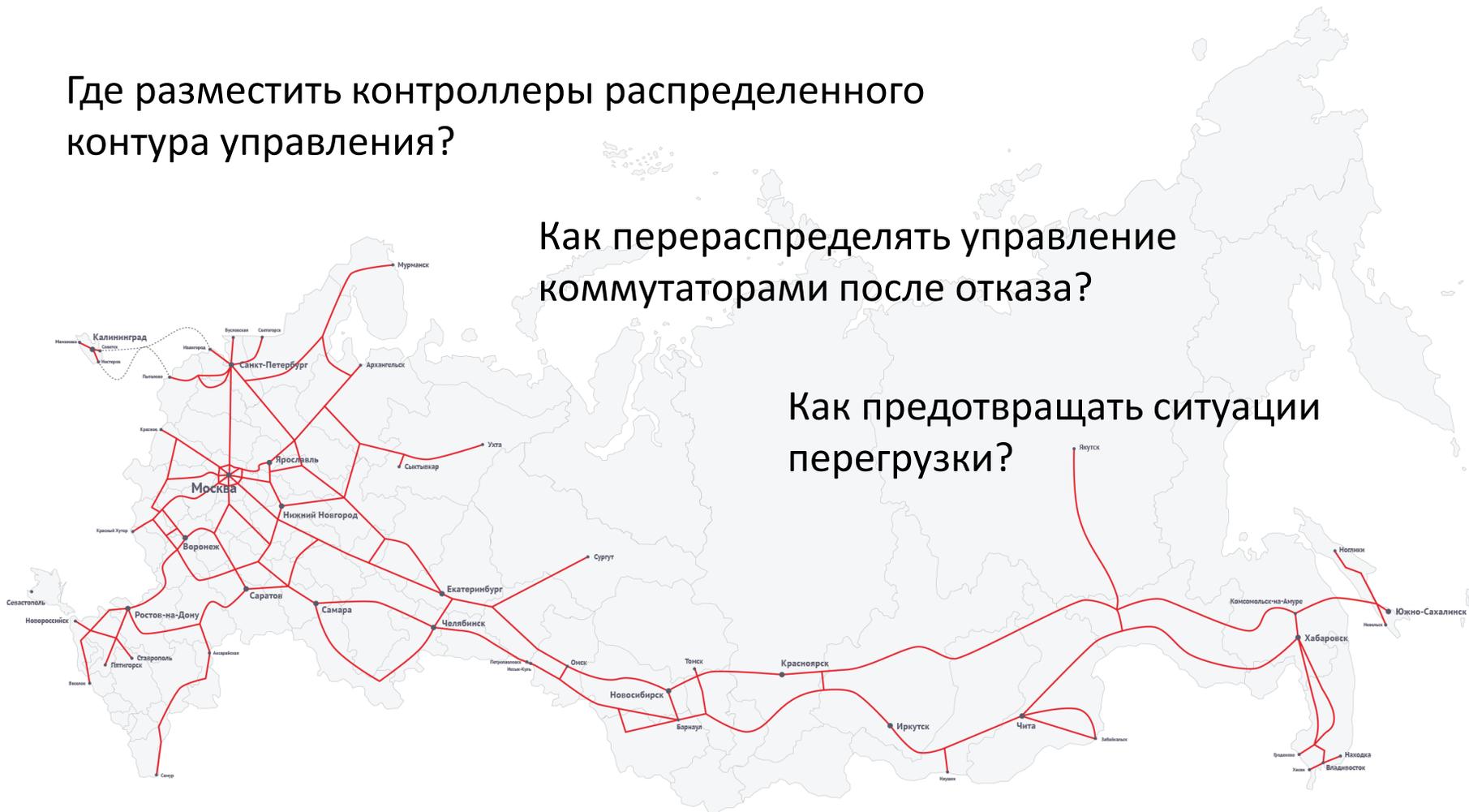


# Пример: магистральная сеть ТрансТелеком (ТТК)

Где разместить контроллеры распределенного контура управления?

Как перераспределять управление коммутаторами после отказа?

Как предотвращать ситуации перегрузки?





# Применение SDN в ЦОД



# Применение SDN для управления сетями ЦОД

## Особенности:

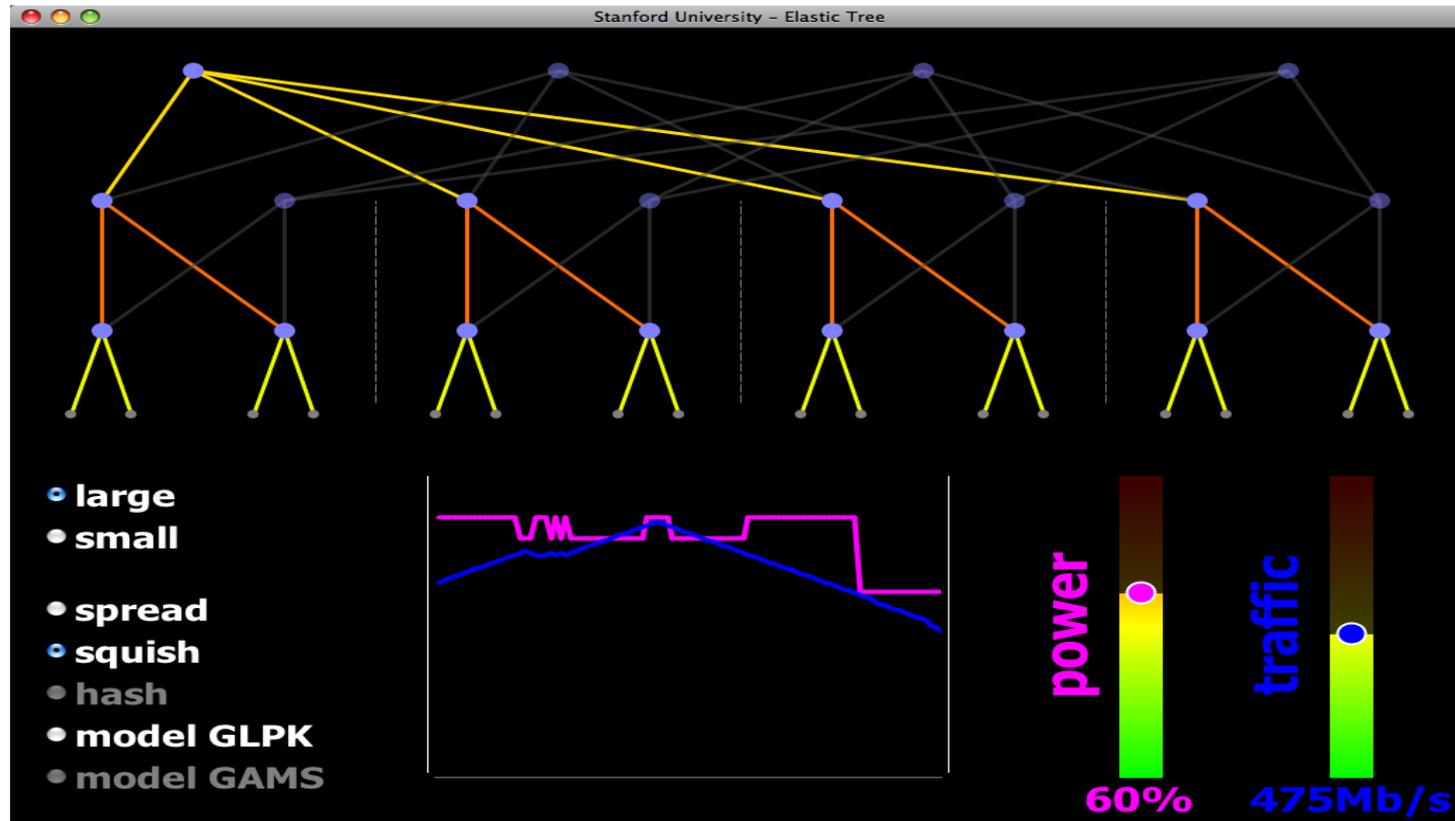
- Высокая плотность приложений и сервисов
- «Паутина соединений»

## SDN:

- Управление «зоопарком» оборудования из одной точки
- Политики настраиваются единократно, затем применяются автоматически



# Применение SDN для управления сетями ЦОД /2

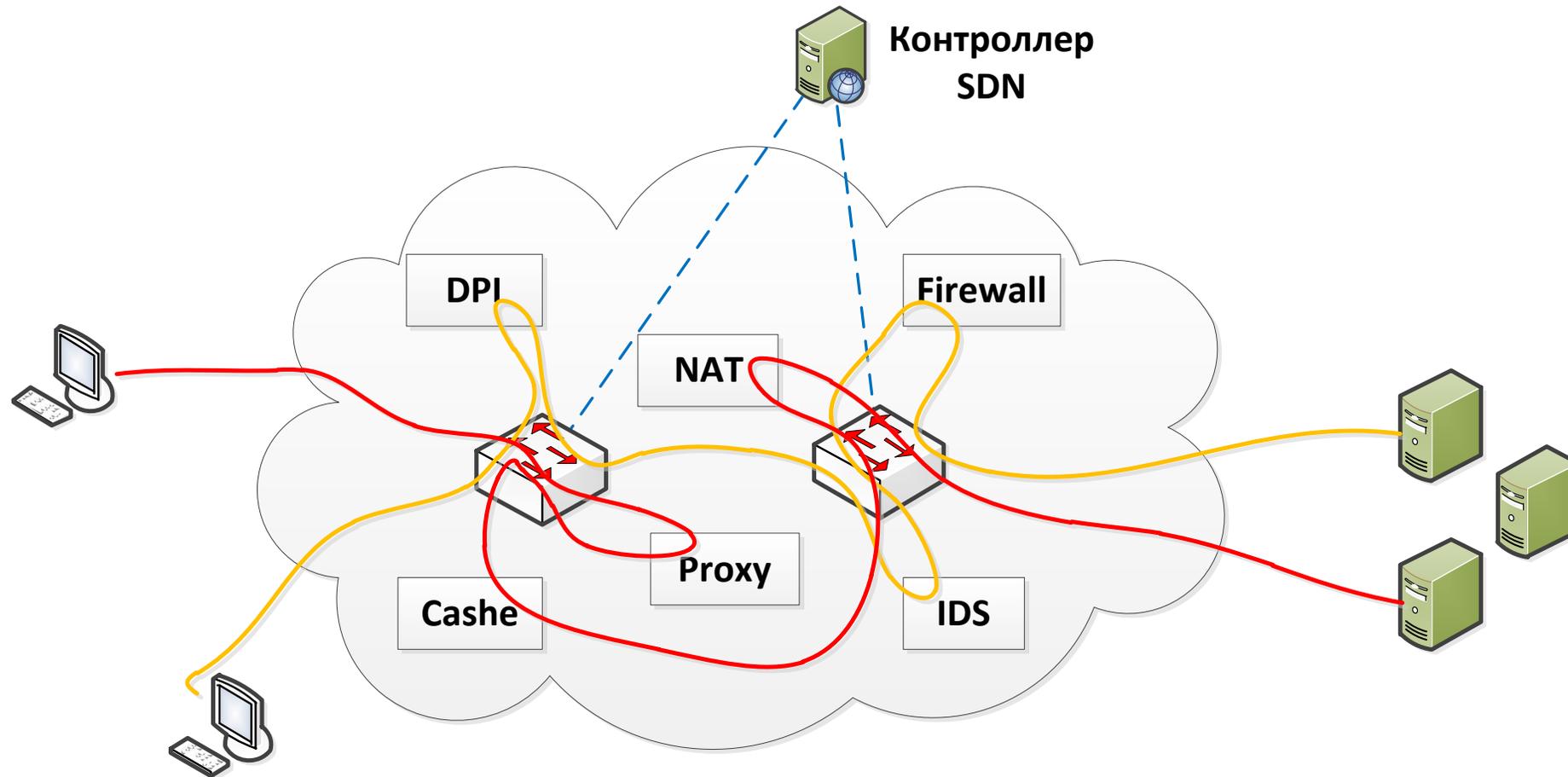


Уменьшение энергопотребления в ЦОД

- Отключение неиспользуемых коммутаторов и каналов на основе собранной информации о сети
- ElasticTree (Stanford): сокращение энергопотребления до 60%
- Применение в Google



# Управление потоками: SDN + NFV = Service Chaining





# Применение AI для задач управления сетями: SDN + AI

## Intent-Based Networking, IBN

- «Сеть на основе намерений» - сеть которая дает возможность автоматизировать задачи управления сетью за счет элементов искусственного интеллекта и машинного обучения.
- С точки зрения администратора это означает более высокий уровень абстракции при управлении сетью: достаточно назначить доступ группе пользователей к приложению, а необходимая для этого настройка сетевого оборудования будет выполнена автоматически.



# SDN в России

- «Ростелеком» начал работу над внедрением перспективных технологических направлений «**Программно-конфигурируемых сетей**» (SDN) и «**виртуализации сетевых функций**» (NFV).
- «Ростелеком» поддерживает **стартапы**, которые занимаются **разработкой технологий** в области SDN и NFV





# SDN в мире

## Open Networking Foundation

Проект коммутатора (white-box): **Stratum**

Проект контроллера: **ONOS**

Cloud Providers



Telecom Operators



Networking Vendors



White Box ODM Vendors



Silicon Vendors





# Примеры тем курсовых и дипломных работ

- Разработка приложений для распределенного контура управления ПКС
- Разработка и исследование метода прогнозирования нагрузки на контроллер в ПКС сетях
- Применение методов ИИ для задач управления в ПКС сетях
- Мониторинг состояния сетевого соединения на ПКС-контроллере
- Исследование алгоритмов консенсуса для распределенного контроллера в программно-конфигурируемой сети
- Разработка и исследование адаптивного алгоритма контроля работоспособности для Active/Active кластера контроллеров RUNOS
- Разработка и оптимизация алгоритмов размещения и сортировки записей в тернарной ассоциативной памяти в OpenFlow коммутаторах



Спасибо за внимание!

**Пашков Василий Николаевич**

[pashkov@lvk.cs.msu.su](mailto:pashkov@lvk.cs.msu.su)

каб. 768