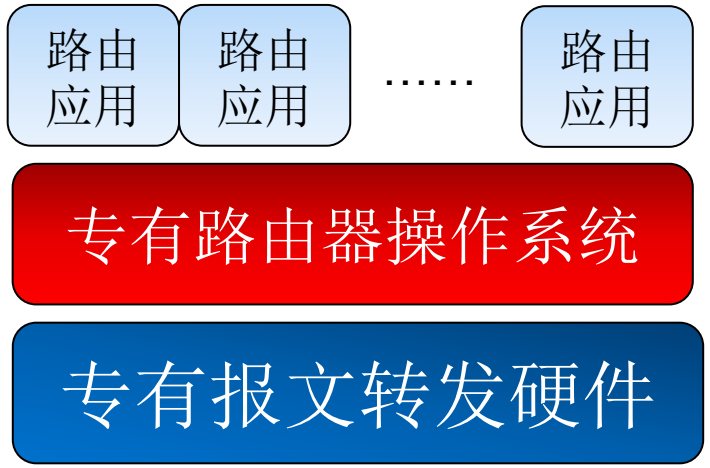
第5章 SDN网络

華中科技大學
网络空间安全学院
School of Cyber Science and Engineering, HUST

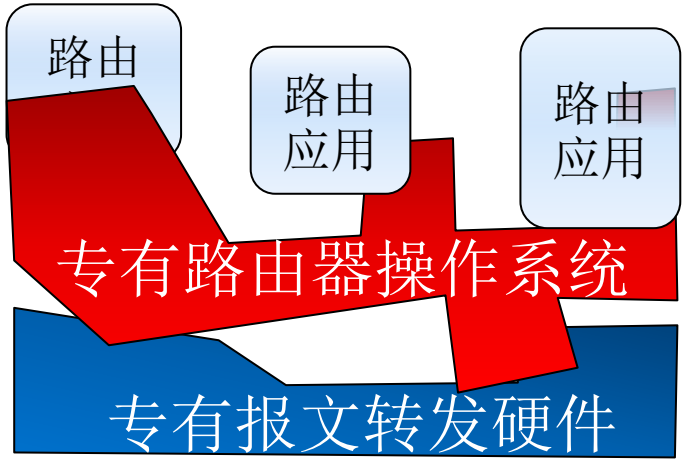
•传统网络存在的问题

•各设备厂家网络层实现方式和架构封闭。

- 封闭架构导致层次和接口模糊封闭，功能单元界面不清晰。
- 封闭架构导致设备制造商对新技术的驱动力不强，协议更新慢。



标准网络层实现

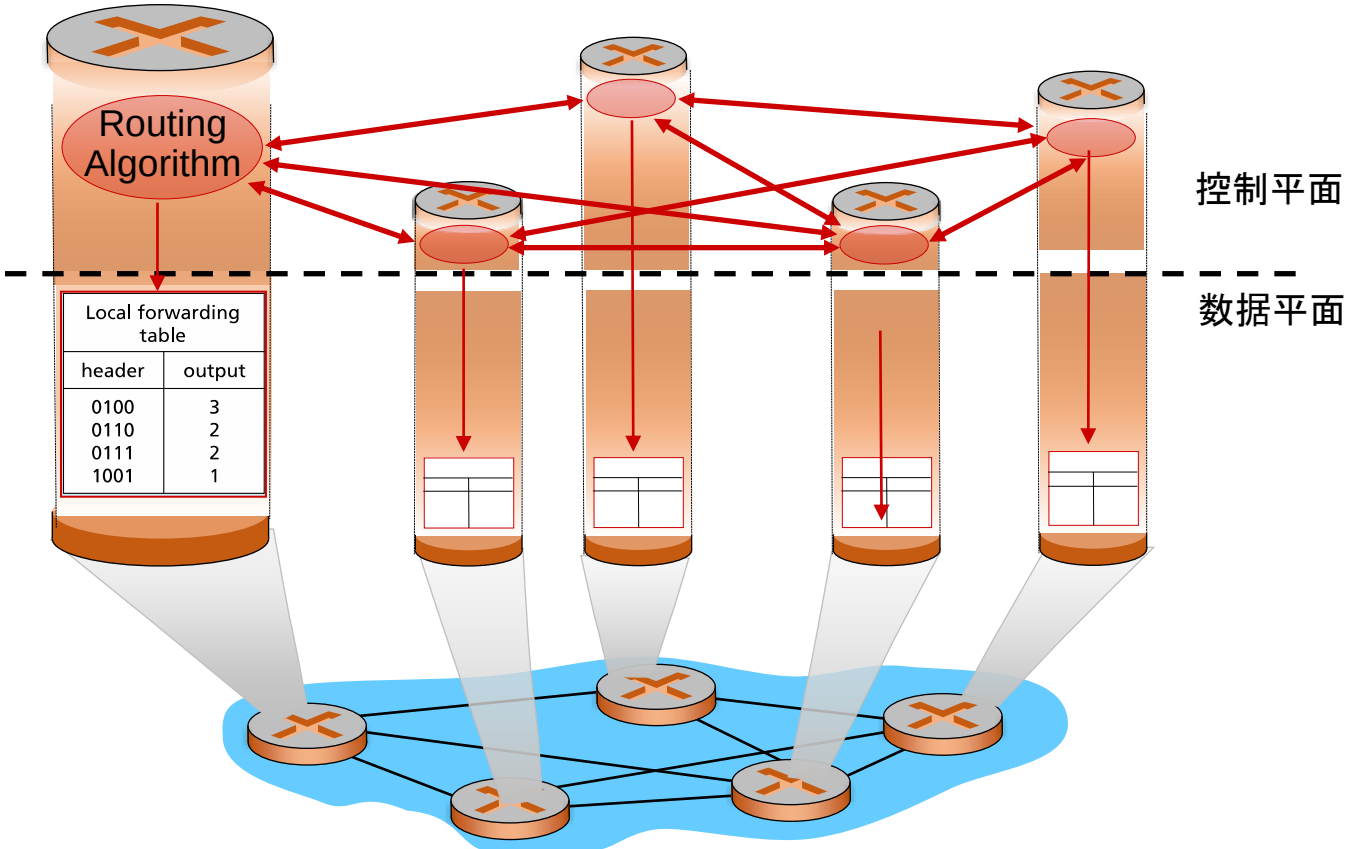


实际网络层的封闭实现

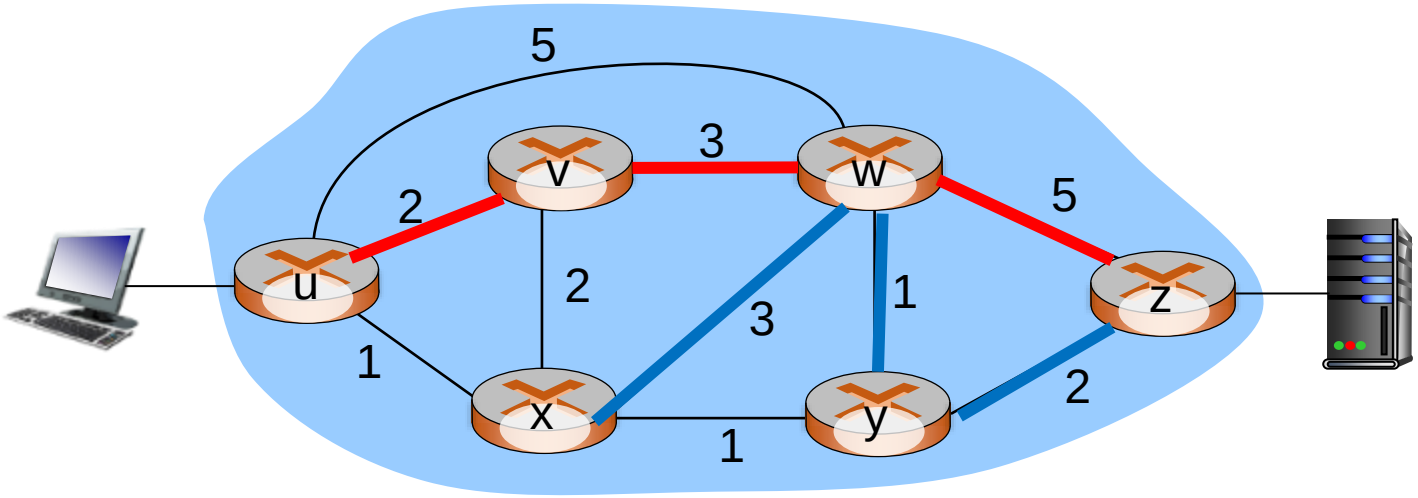
- **回顾: 网络层的主要功能**
  - 在全局范畴为主机之间的通信进行**选路**，选路的结果反映为分组交换机（设备）上的转发表 **(控制平面)**
    - 控制数据报沿着从源主机到目的主机的端到端路径中路由器之间的路由方式
  - 分组交换机（设备）上的网络层根据转发表以及分组头部信息，将分组向适当链路进行**转发（数据平面）**
    - 决定到达路由器输入链路之一的数据包如何转发到路由器的输出链路之一

•路由器实现的控制平面

•每个路由器具备独立的路由功能和数据转发功能，路由算法组件分布在不同路由器上，彼此交互，计算生成转发表，构成**分布式**的控制平面。



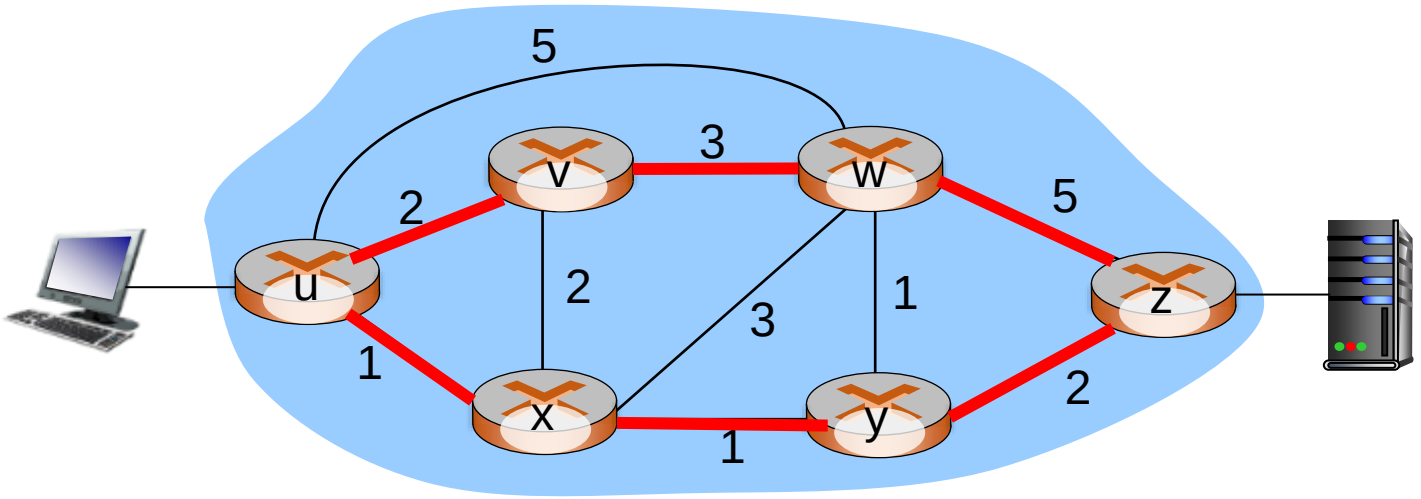
- 挑战：确定性路由



- 需求：网络运营商希望u-z的流量经过uvwz，x-z的流量经过xwyz

- 方案：
- 1) 定义链路权重，以便流量路由算法计算相应路由。
- 2)采用新的路由算法.

•挑战：负载均衡路由



需求：

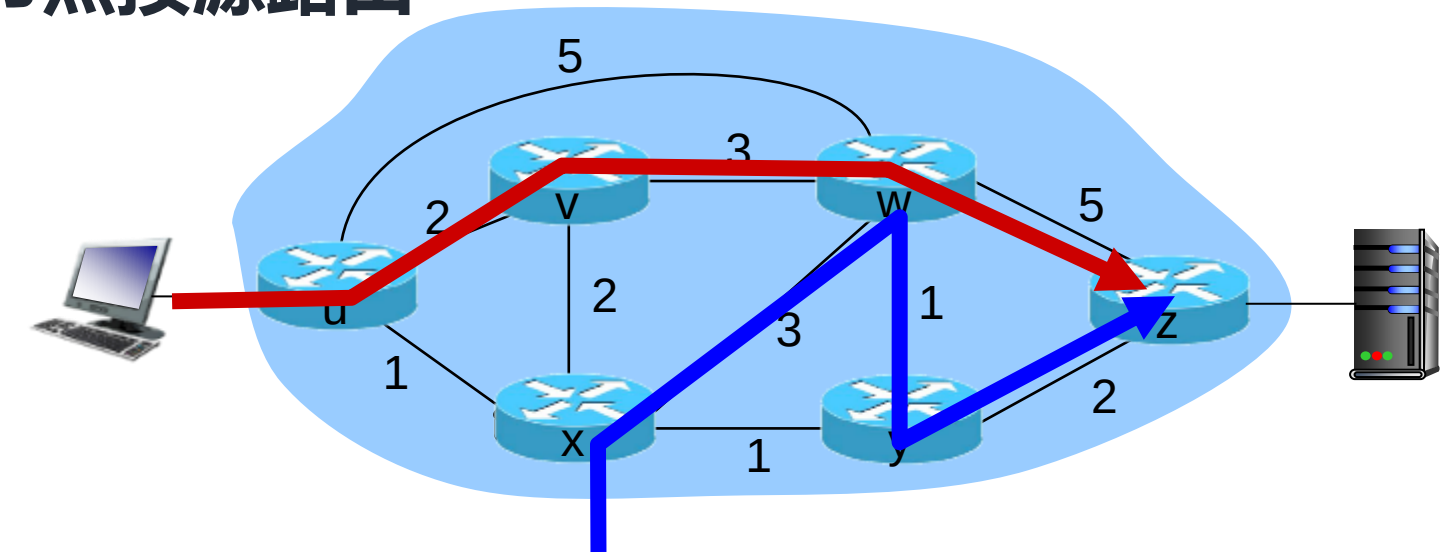
网络运营商希望由uvwz和uxyz两条路径来分担u-z的流量

•方案：

•1）保证多条路径为等价路由（代价相同），不支持按流指定负载均衡。

•2) 采用新的路由算法.

•挑战：中间节点按源路由



需求：

W以不同方式对来源不同的蓝色和红色流进行路由

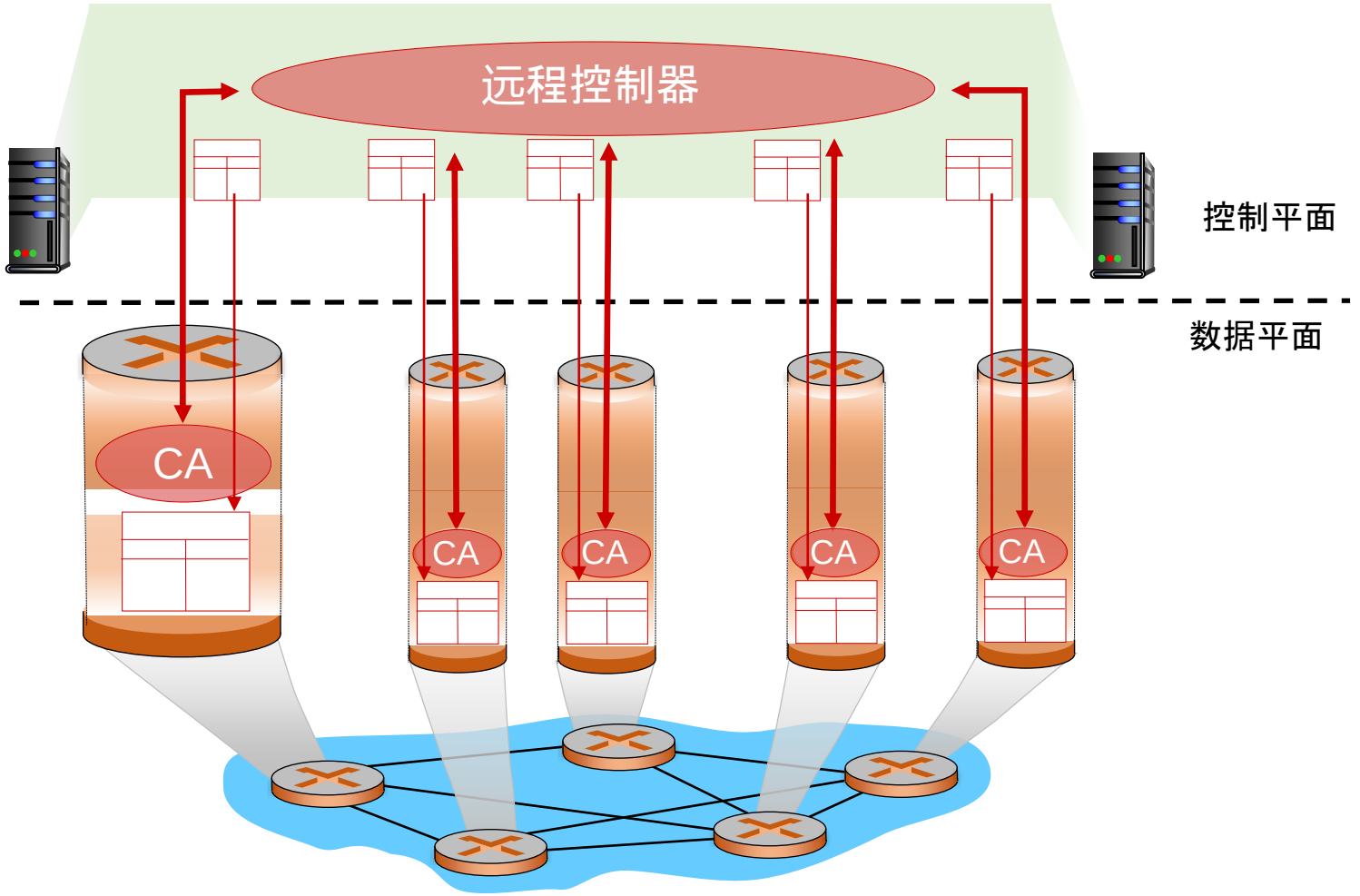
方案：

1)现有基于目的地址的路由协议（LS和DV路由协议）无法支持。

2)采用新的路由算法。

•解决方案：分层（功能分离）

路由控制功能从本地路由器分离，汇聚到远程控制器，与路由器中的本地控制代理(CAs)进行交互，以计算转发表



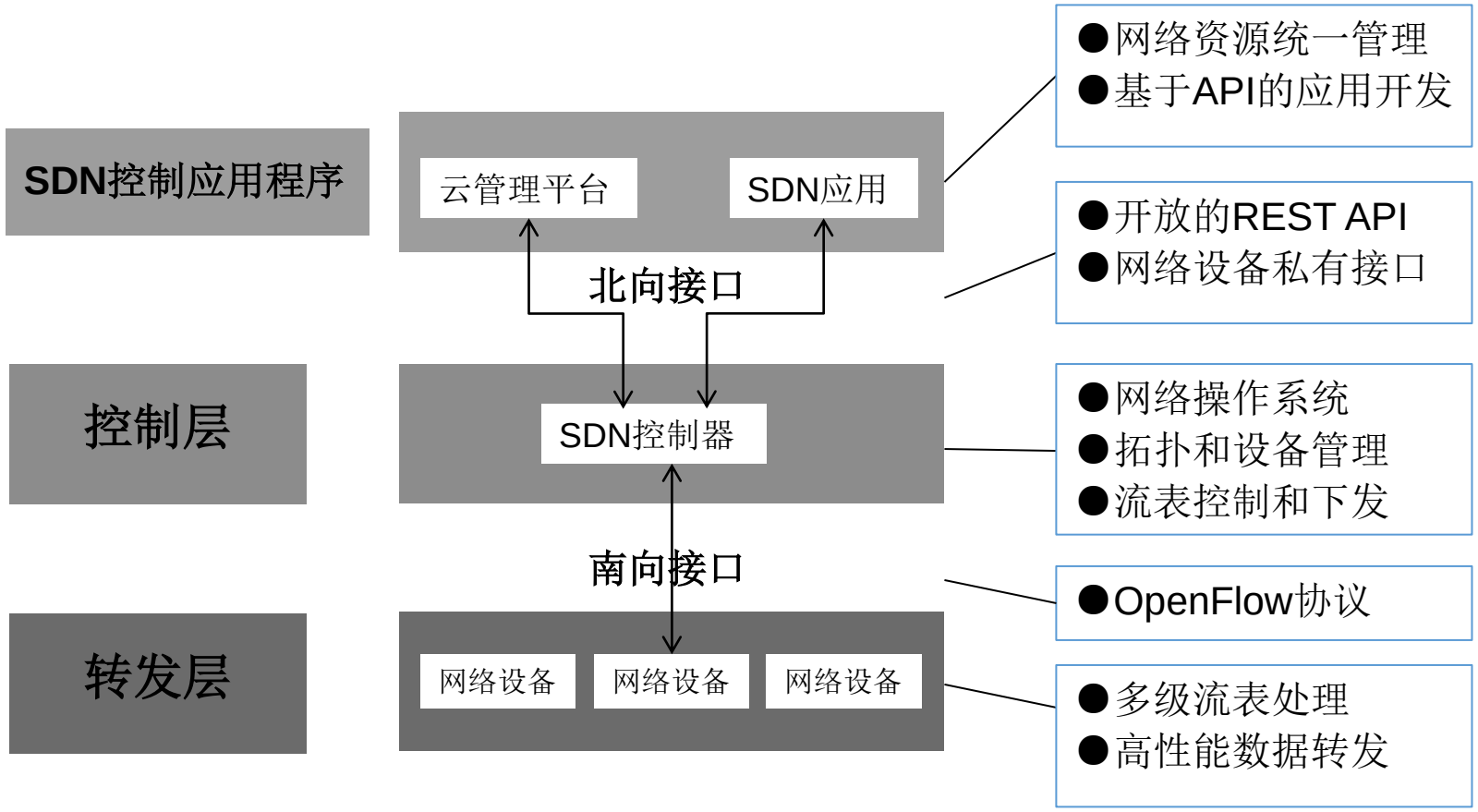
数据平面

- 本地，每个路由器功能
- 决定从路由器输入端口到达的分组如何转发到输出端口
- 转发功能：
  - 传统方式：基于目标地址+转发表
  - SDN式：基于多个字段+流表

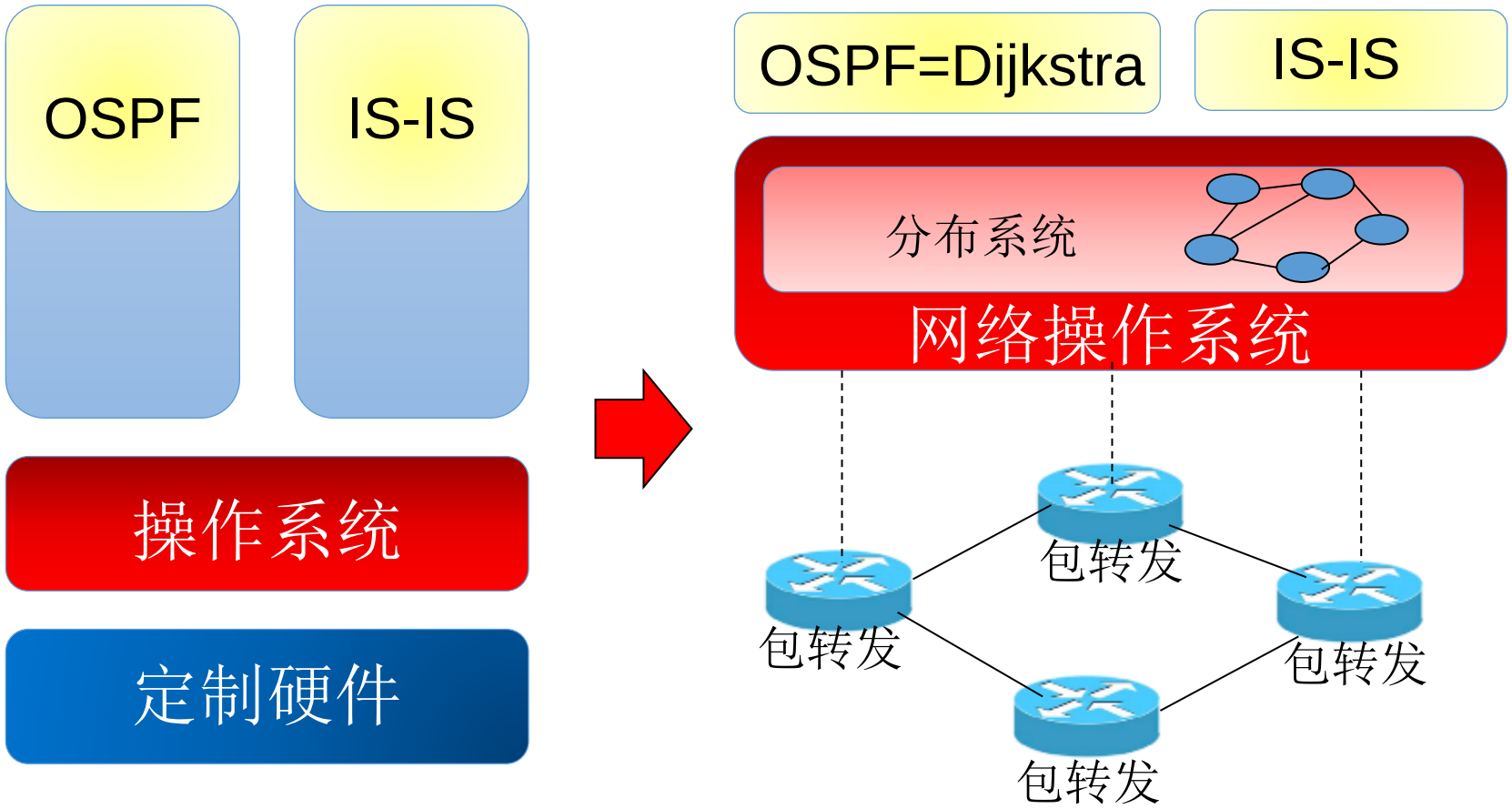
控制平面

- 网络范围内的逻辑
- 决定数据包如何在路由器之间路由，决定数据报从源到目标主机之间的端到端路径
- 2个控制平面方法：
  - 传统的路由算法：在路由器中实现
  - SDN：在远程服务器中实现

•SDN架构：控制与转发分离



•SDN架构：SDN化示例



•SDN架构：数据平面与控制平面

- 数据平面
  - 处理和转发数据包
  - 转发状态+数据报头->转发决策
- 控制平面
  - 计算路由器的转发状态
  - 确定数据报如何转发和转发到哪里
  - 路由、流量工程和防火墙状态管理控制
  - 实现分布路由协议、手工配置或集中计算
- 不同平面需分别进行抽象，以满足SDN需求

数据平面的变化趋势

- 路由器转发的传统方案
  - 基于数据包的目的IP地址转发。

- 网络层变化
  - NAT重写首部IP和端口。
  - 防火墙根据首部字段阻断和重定向。
  - 负载均衡转发到目的地址簇中某个地址。

- 以L3数据为基础转发

- L2/L3/ACL/QoS/组播/安全防护的转发和处理

- 单一的转发处理

- 包括转发、丢弃阻断和重定向等多种处理方式

- 以主机为粒度
- 仅考虑目标IP地址

- 以流为粒度
- 考虑MAC地址、IP、端口
- 同时考虑源和目的

•数据平面：通用转发

匹配

•对协议栈的多个首部字段进行匹配

•包括链路层/网络层/传输层

动作

转发到一个或多个输出端口（路由转发）

跨越多个服务端口进行负载均衡（负载均衡）

重写首部值（NAT）

阻断/丢弃分组（防火墙）

向特定服务器发送分组（DPI）

统一成流表（Openflow）

首部字段值的集合

计算器集合

分组匹配流表项后的动作组合

•OpenFlow流表

•流表由多个流条目组成，流条目包括

- •头域：用于匹配规则确定输入报文是否与本条目匹配。
- •计数器：用于与本流相关的跟踪统计
- •动作：描述交换机针对匹配报文采取的动作。

	头域	计数器	动作
流条目0	输入端口12 192.168.1.0 端口 1012	val	val
流条目1	输入端口 * 209.* 端口 *	val	val
		val	val
流条目k	输入端口 2 192.168.2.0 端口 995	…	…
		val	val
流条目n	输入端口 2 192.168.3.0 端口 995	…	…

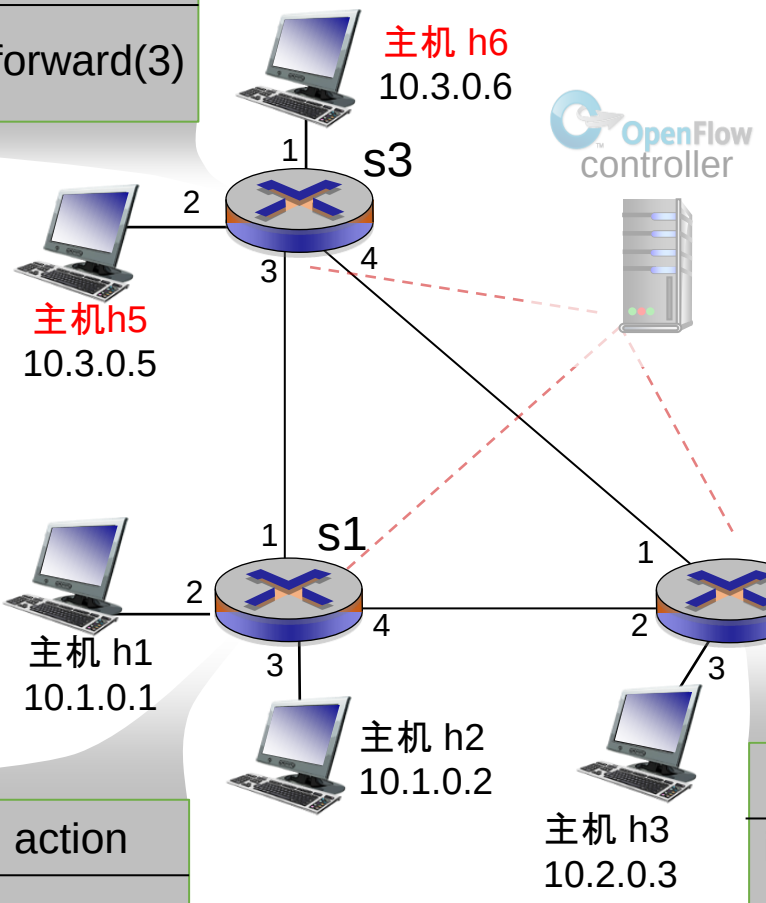
OpenFlow通过功能抽象统一交换转发设备处理方式

- 路由器
  - ***匹配***：最长目的IP前缀
  - ***动作***：通过某链路端口转发
- 交换机
  - *匹配*：目的MAC地址
  - ***动作***：转发或者洪泛

- 防火墙
  - *匹配*：IP地址和UDP/TCP端口号
  - ***动作***：允许或者拒绝
- NAT
  - *匹配*：IP地址和UDP/TCP端口号
  - ***动作***：重写地址和端口

OpenFlow 流表匹配示例

匹配	动作
IP Src = 10.3.*.* IP Dst = 10.2.*.*	forward(3)



举例: 来自主机h5和h6的数据报应该通过s1发送到s2, 再从s2发送到h3或h4

匹配	action
ingress port = 1 IP Src = 10.3.*.* IP Dst = 10.2.*.*	forward(4)

匹配	动作
ingress port = 2 IP Dst = 10.2.0.3	forward(3)
ingress port = 2 IP Dst = 10.2.0.4	forward(4)

OpenFlow 流表匹配示例

交换

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

按流交换

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

防火墙

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

OpenFlow 流表匹配示例

路由

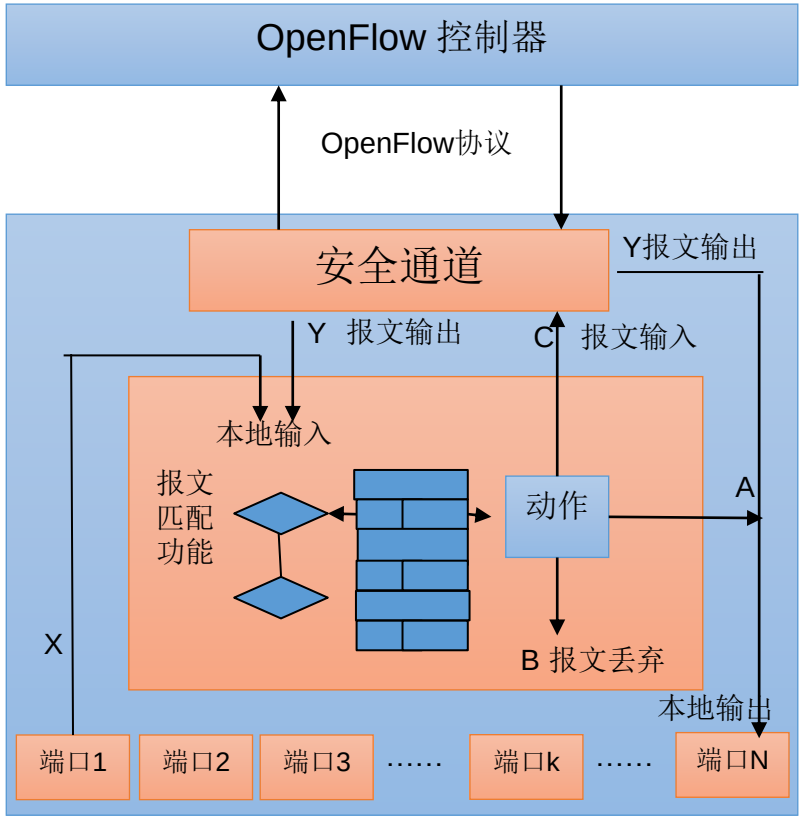
Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	5.6.7.8	*	*	*	port6

VLAN交换

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f..	*	vlan1	*	*	*	*	*	port6, port7, port9

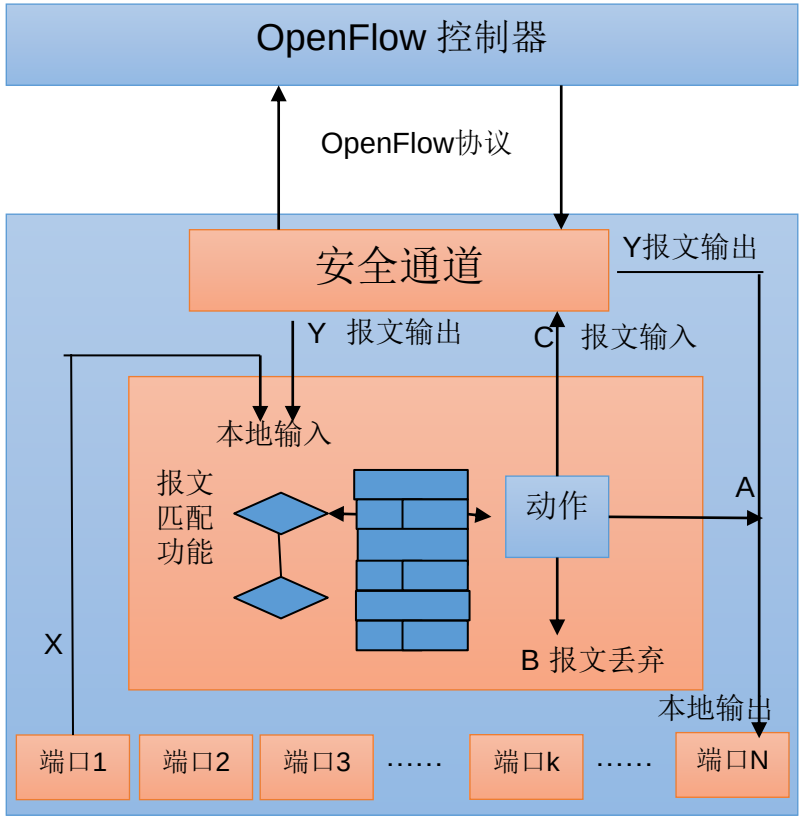
•OpenFlow交换机

- •OpenFlow交换机由OpenFlow协议、安全通道、报文匹配、流表和动作等组成。
- •报文匹配功能基于流表对输入报文（X）进行匹配，将其引导至动作箱。
- •动作包括三种可选操作
  - •转发报文输出，可能先修改头域字段
  - •丢弃报文
  - •通过报文输入（PAKCET_IN）消息将报文转发至控制器



•OpenFlow交换机

- 控制器和交换机之间的报文通过安全通道传输。
- 当控制器有报文需要通过交换机输出时，采用PACKET_OUT消息，图中所示的两条Y路径
- 控制器直接指定输出端口
- 控制器通过报文匹配逻辑决定转发策略



•SDN控制平面

- SDN控制平面包括SDN控制器和SDN网络控制应用程序。
- 控制平面基于网络的抽象，简化了网络编程控制
- SDN控制平面包括两个层面的抽象
  - 流抽象-交换机（南向）API，通过Openflow协议与数据平面交互
  - 映射抽象-网络（北向）API，控制器与网络控制应用程序之间交互