CS 3700

Networks and Distributed Systems

Lecture 12: Quality of Service (QoS)

Motivation

- Internet is designed to give best-effort service
  - i.e. all packets are treated the same

Motivation

- Internet is designed to give best-effort service
  - i.e. all packets are treated the same
- However, not all packets are the same
  - HTTP – delay sensitive

Motivation

- Internet is designed to give best-effort service
  - i.e. all packets are treated the same
- However, not all packets are the same
  - HTTP – delay sensitive
  - Voice/Video streaming – delay and jitter sensitive
  - Online Games – delay and jitter sensitive

Motivation

- Internet is designed to give best-effort service
  - i.e. all packets are treated the same
- However, not all packets are the same
  - HTTP – delay sensitive
  - Voice/Video streaming – delay and jitter sensitive
  - Online Games – delay and jitter sensitive
  - BitTorrent – totally insensitive
    - Delay, jitter, bandwidth do not matter
    - File transfer will finish eventually

Motivation

- Internet is designed to give best-effort service
  - i.e. all packets are treated the same
- However, not all packets are the same
  - HTTP – delay sensitive
  - Voice/Video streaming – delay and jitter sensitive
  - Online Games – delay and jitter sensitive
  - BitTorrent – totally insensitive
    - Delay, jitter, bandwidth do not matter
    - File transfer will finish eventually
- Should the network give better quality to some packets?

Three Relevant Factors

1. Application performance
   - How much bandwidth do you need?
   - What about delay and jitter?

Three Relevant Factors

1. Application performance
   - How much bandwidth do you need?
   - What about delay and jitter?
2. Bandwidth required to provide performance
   - How to meet performance goals…
   - While still offering general service to all applications

Three Relevant Factors

1. Application performance
   - How much bandwidth do you need?
   - What about delay and jitter?
2. Bandwidth required to provide performance
   - How to meet performance goals…
   - While still offering general service to all applications
3. Complexity/cost of required mechanisms
   - How to modify the network to meet perf. goals?
   - Political concerns, e.g. network neutrality
   - Security

QoS: Quality of Service

- Idea: build some unfairness into the network
  - Some traffic is high priority, gets better service
  - Some traffic is low priority, gets reduced service

QoS: Quality of Service

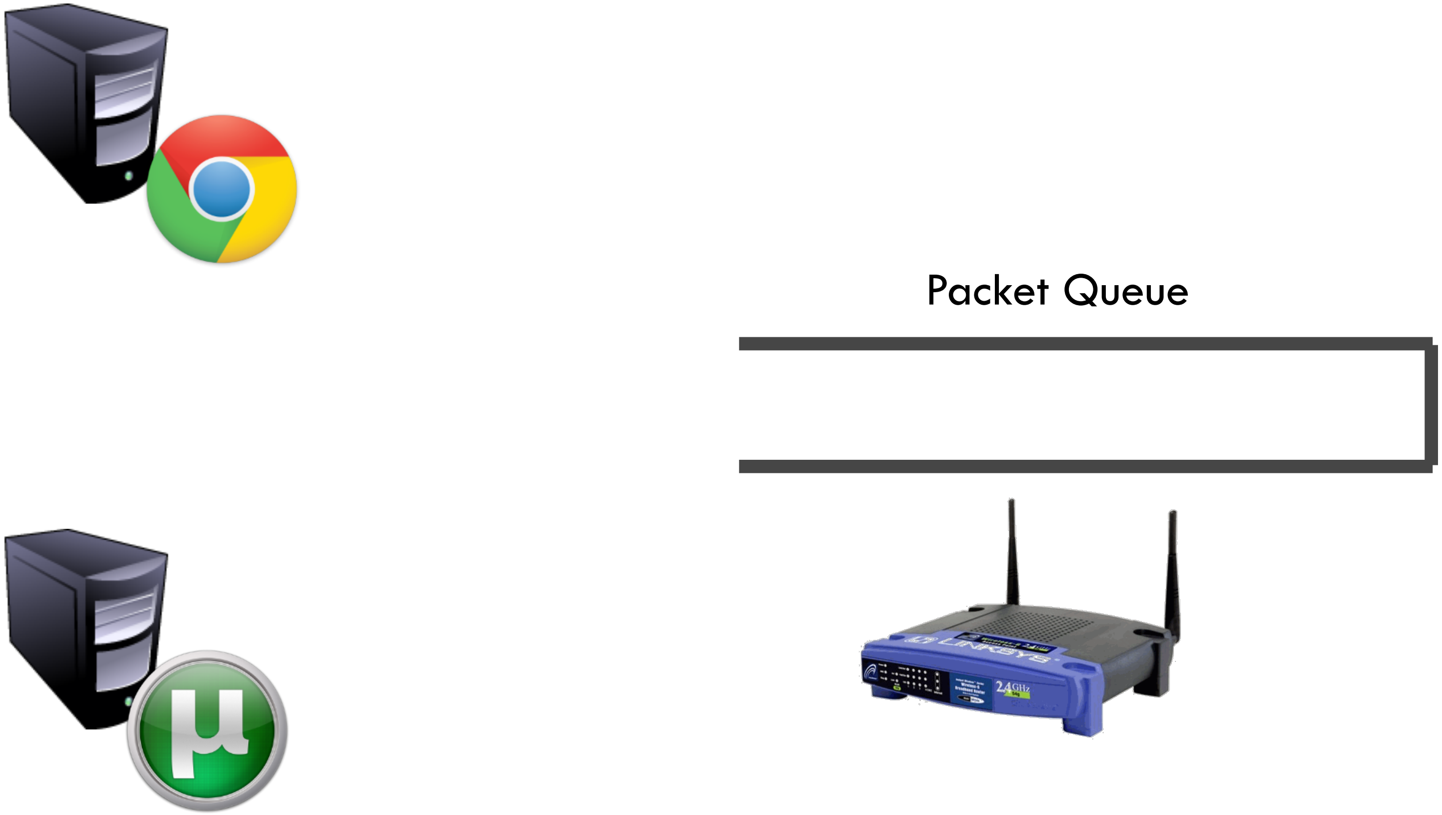
- Idea: build some unfairness into the network
  - Some traffic is high priority, gets better service
  - Some traffic is low priority, gets reduced service
- Thus, “important” traffic receives “better” service
  - What traffic is important?
  - What do we mean by “better” service?
    - Is the gain guaranteed and strictly defined?
    - Is the gain relative and fungible?

Outline

- “Soft” QoS
  - Packet shaping/prioritization
  - DiffServ
- “Hard” QoS
  - IntServ

The Problem at the Edge

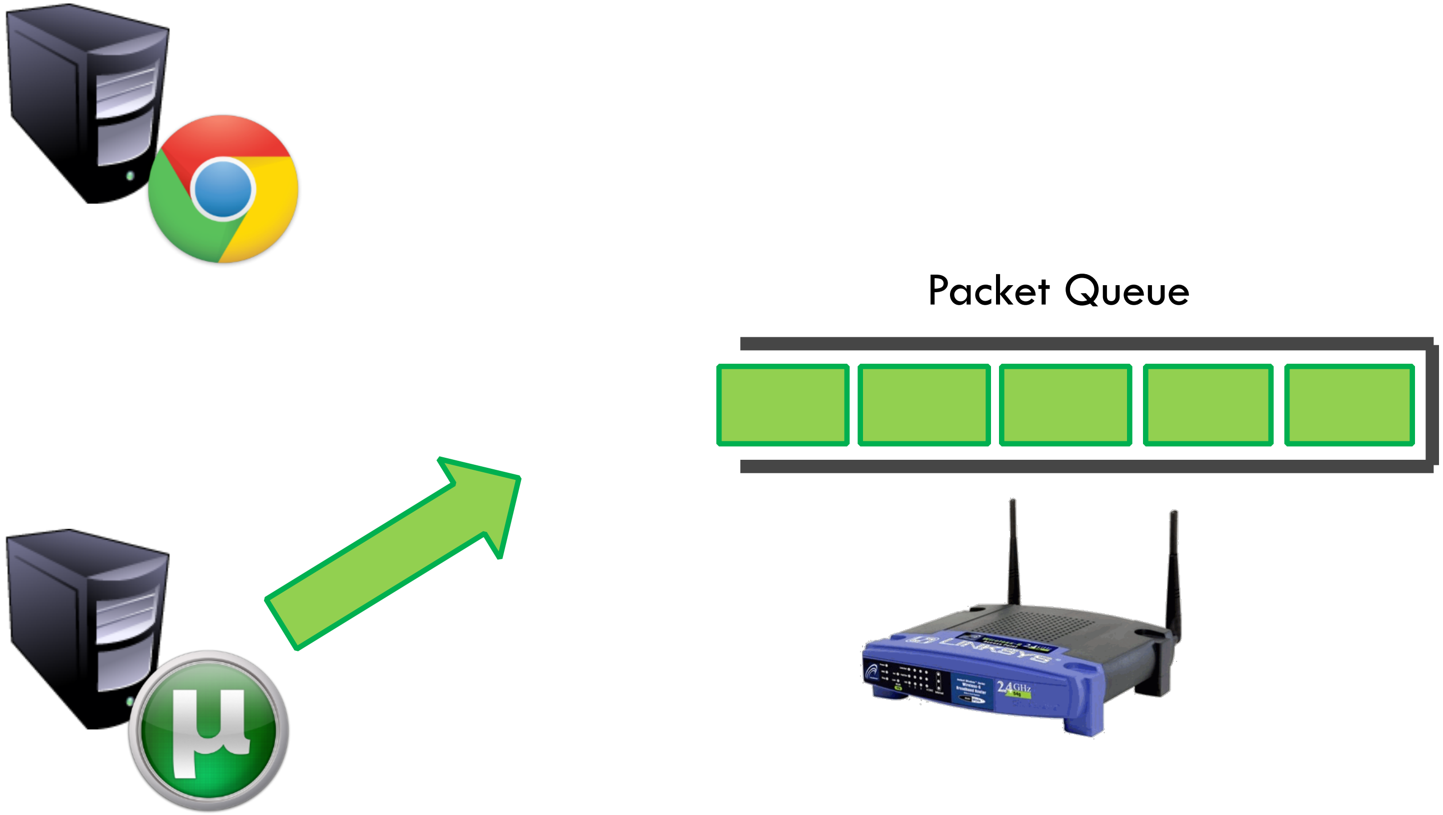
□ Problem: sharing resources between applications



Packet Queue

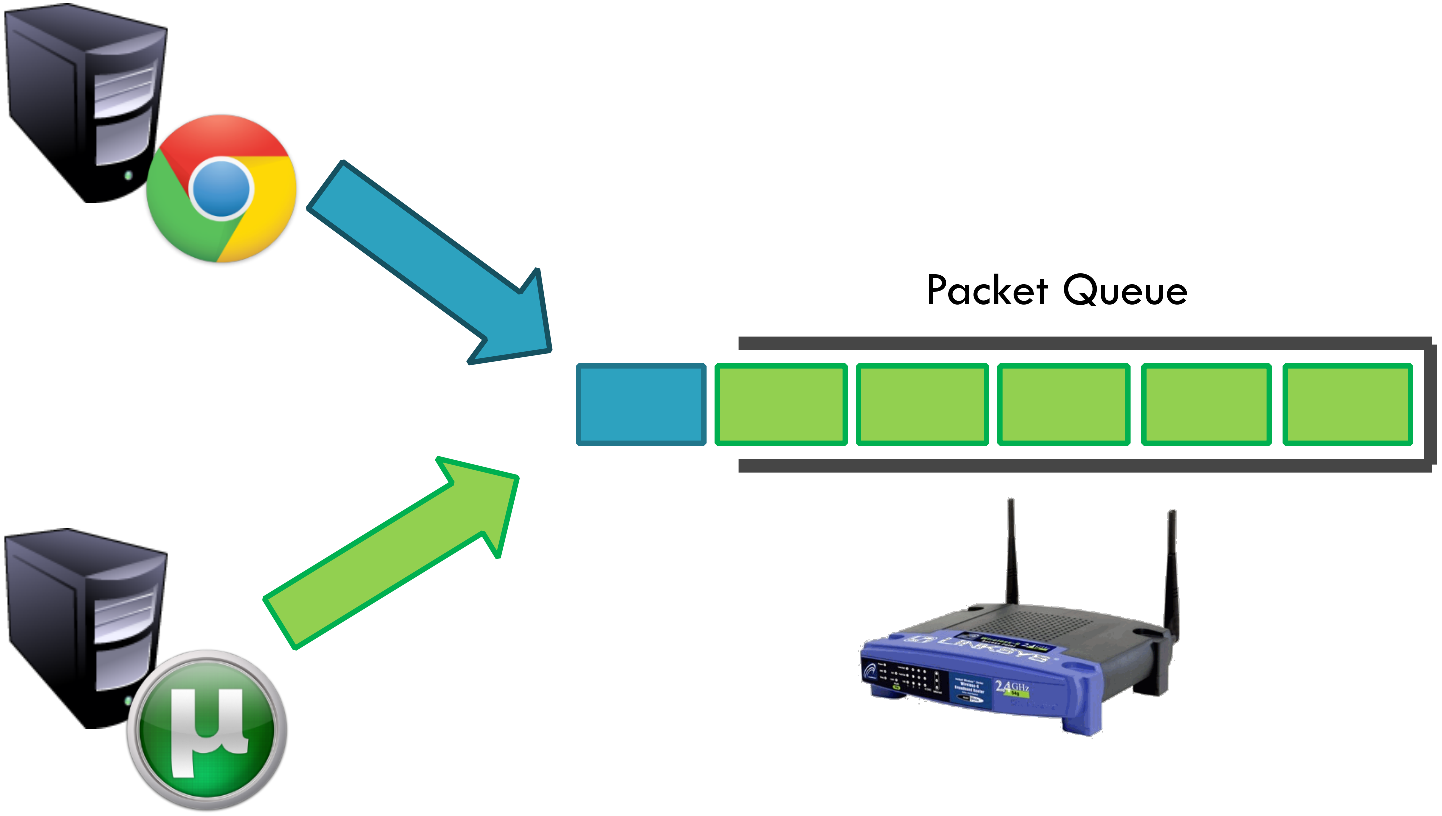
The Problem at the Edge

- Problem: sharing resources between applications



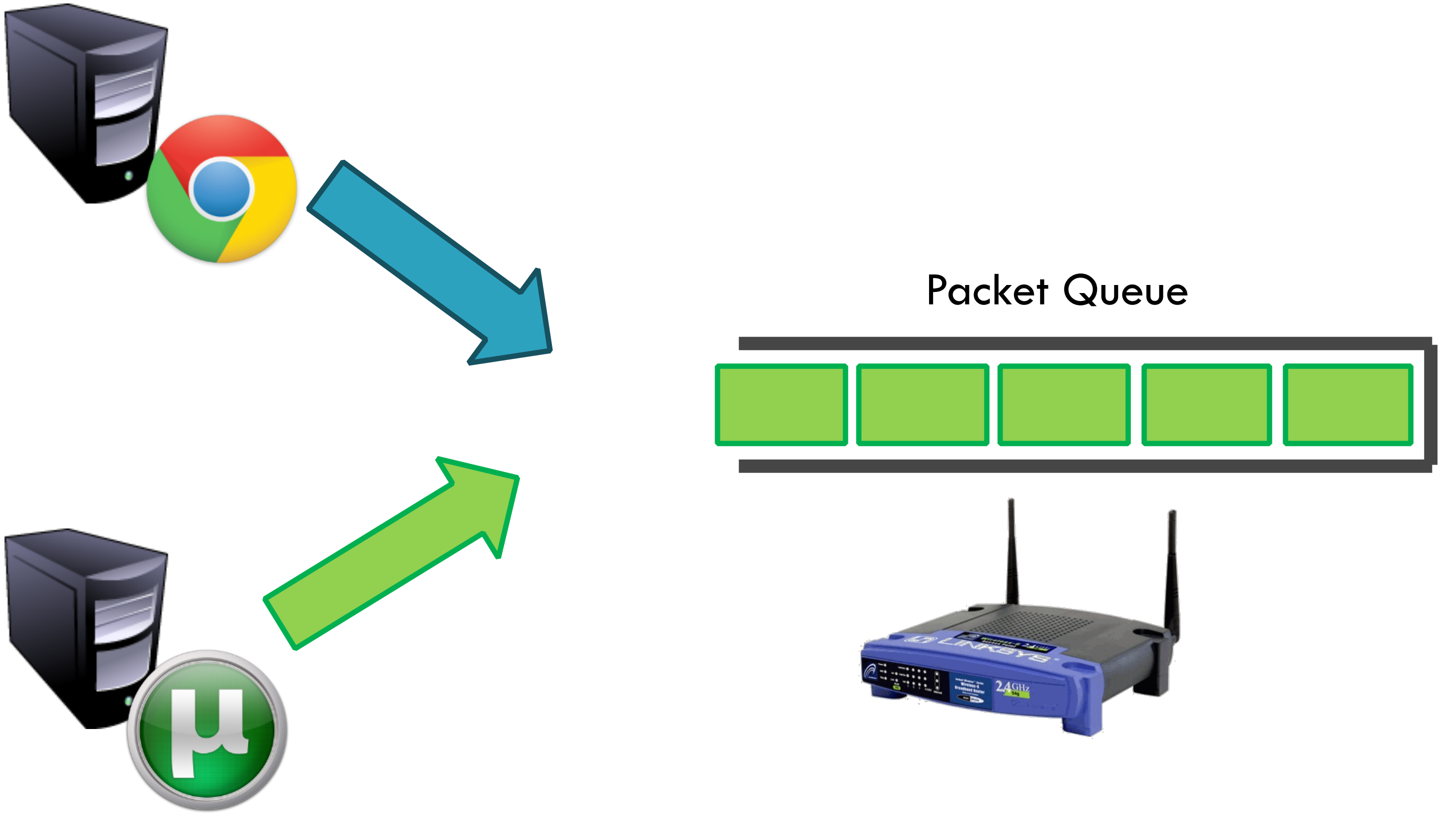
The Problem at the Edge

- Problem: sharing resources between applications



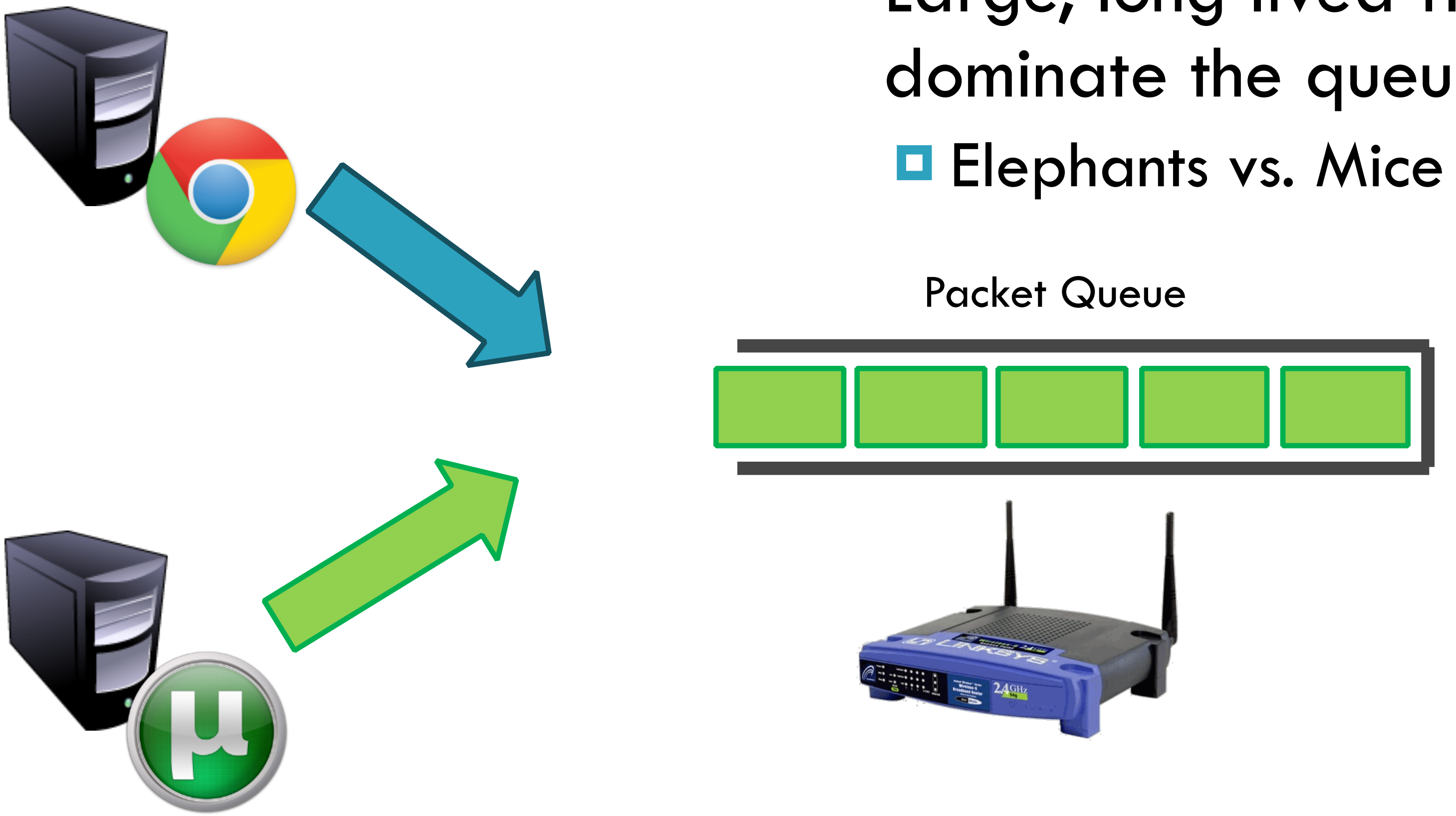
The Problem at the Edge

- Problem: sharing resources between applications



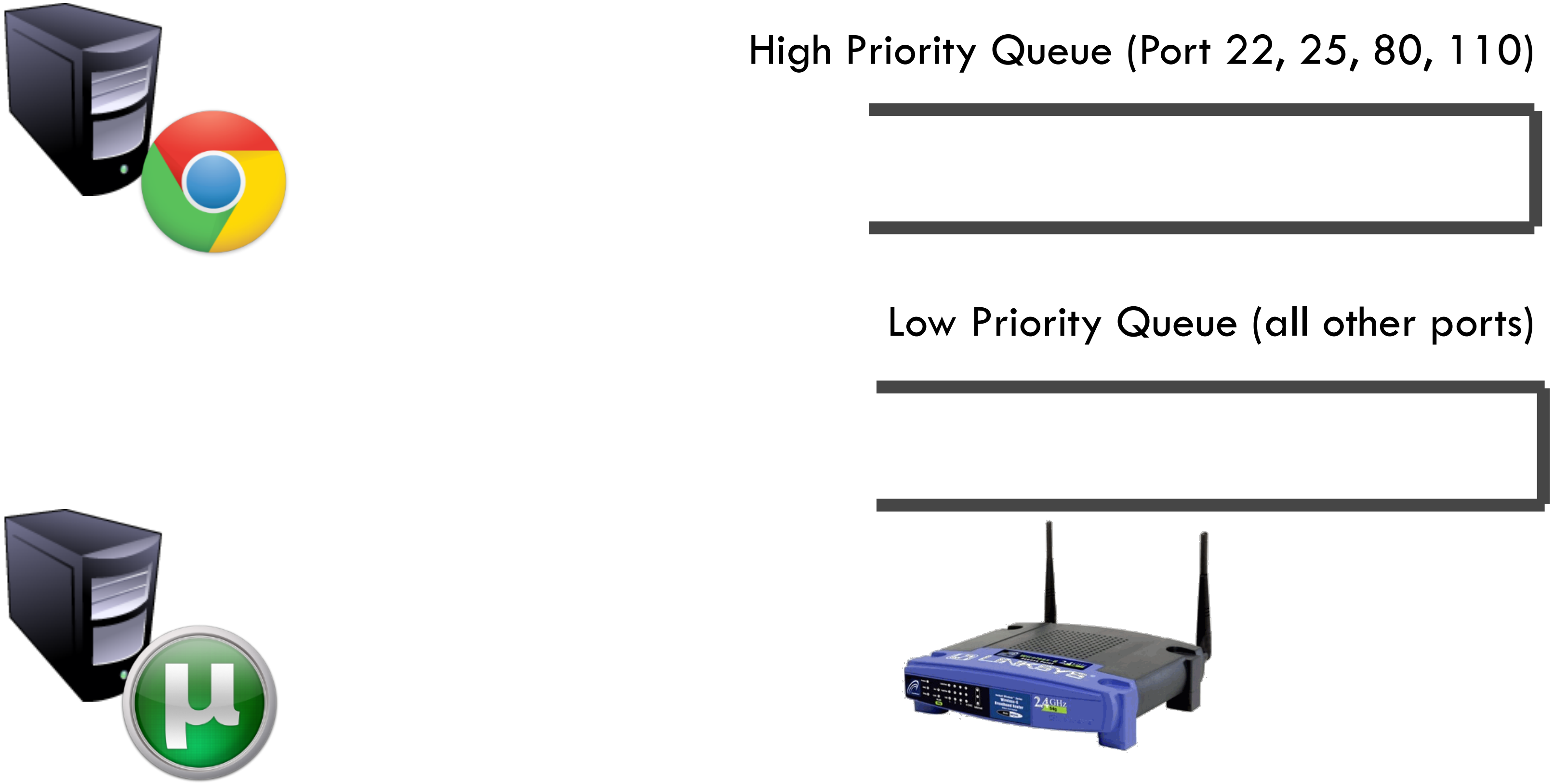
The Problem at the Edge

- Problem: sharing resources between applications
  - Large, long lived flows can dominate the queue
    - Elephants vs. Mice



Port-Based Priority Queues

- Port-based QoS
  - Very common in home routers

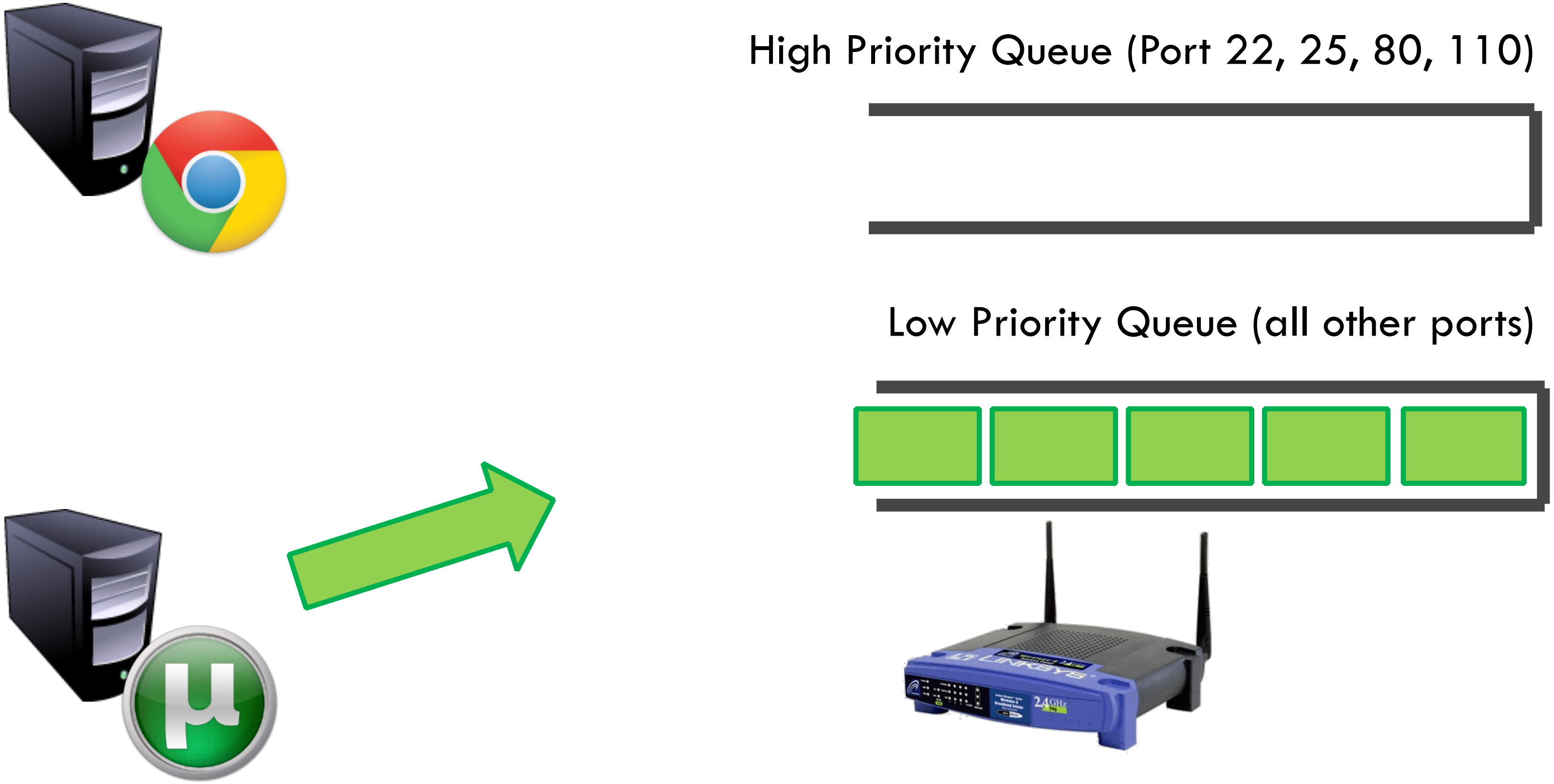


High Priority Queue (Port 22, 25, 80, 110)

Low Priority Queue (all other ports)

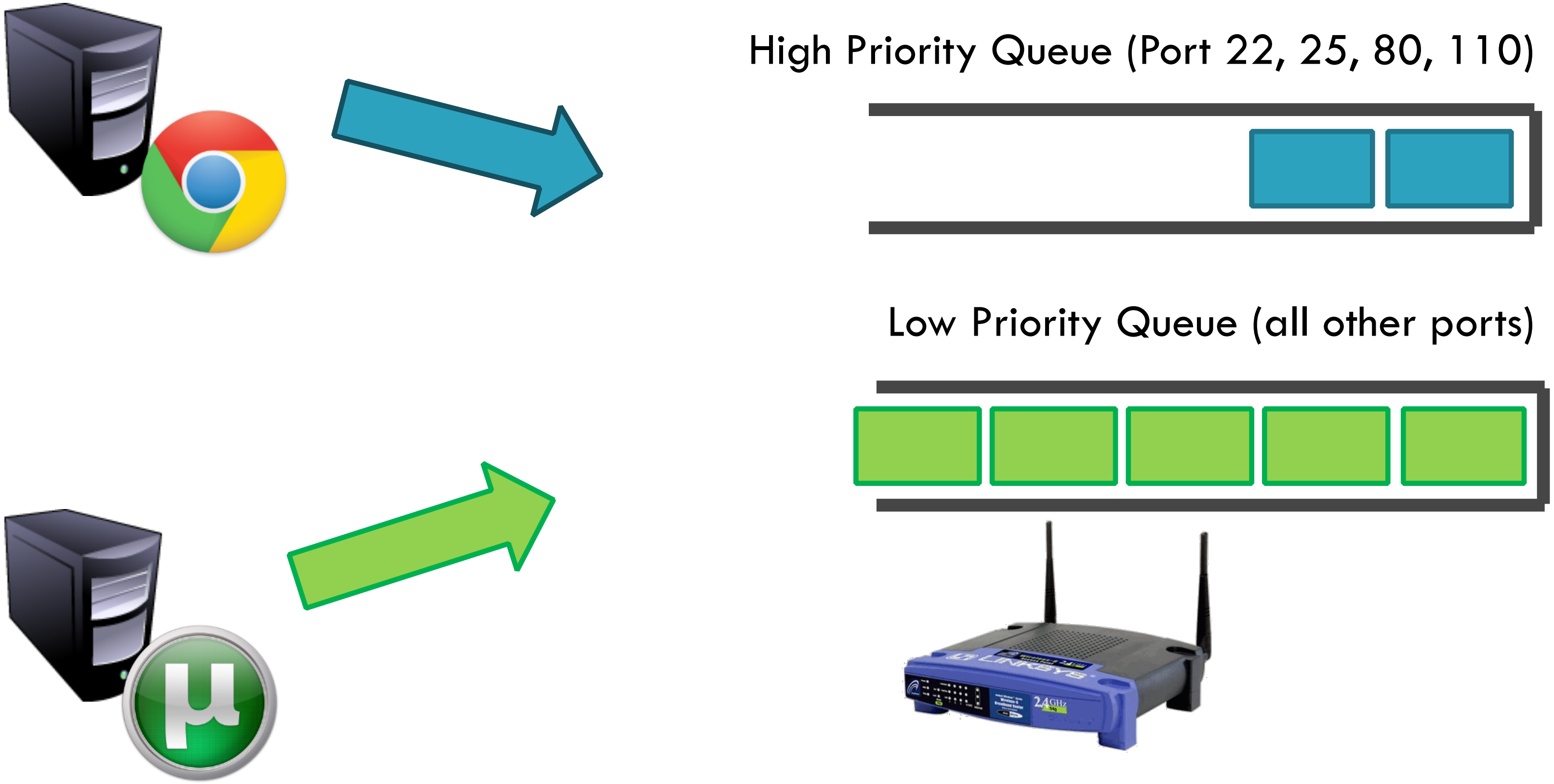
Port-Based Priority Queues

- Port-based QoS
  - Very common in home routers


High Priority Queue (Port 22, 25, 80, 110)
Low Priority Queue (all other ports)

Port-Based Priority Queues

- Port-based QoS
  - Very common in home routers

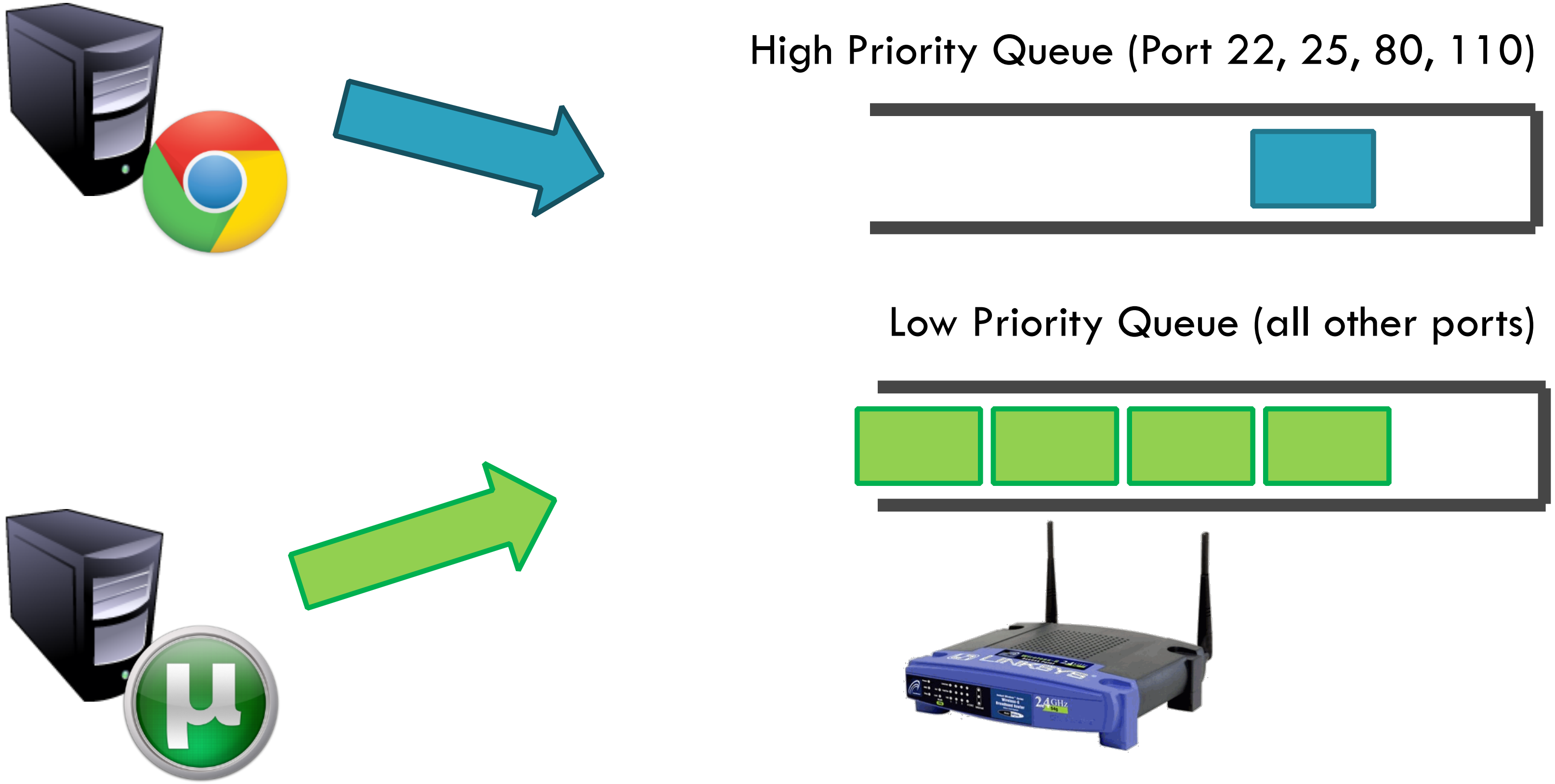


High Priority Queue (Port 22, 25, 80, 110)

Low Priority Queue (all other ports)

Port-Based Priority Queues

- Port-based QoS
  - Very common in home routers



High Priority Queue (Port 22, 25, 80, 110)

Low Priority Queue (all other ports)

QoS at Internet Scale

- Priority queues at the edge of the network help
  - … but what about QoS across the entire Internet?

QoS at Internet Scale

- Priority queues at the edge of the network help
  - … but what about QoS across the entire Internet?
- Popular area of research in the 1990's
  - Differentiated Service (DiffServ)
    - Class-based traffic management mechanism
    - Coarse grain control
    - Relative performance improvements / lower overhead

QoS at Internet Scale

- Priority queues at the edge of the network help
  - … but what about QoS across the entire Internet?
- Popular area of research in the 1990's
  - Differentiated Service (DiffServ)
    - Class-based traffic management mechanism
    - Coarse grain control
    - Relative performance improvements / lower overhead
  - Integrated Service (IntServ)
    - Flow-based traffic management mechanism
    - Fine grained control
    - Guaranteed performance / high overhead

Differentiated Services (DiffServ)

- Goal: offer different levels of service to packets
  - Organized around domains (ASs)
  - Involves edge and core routers (sometimes hosts too)

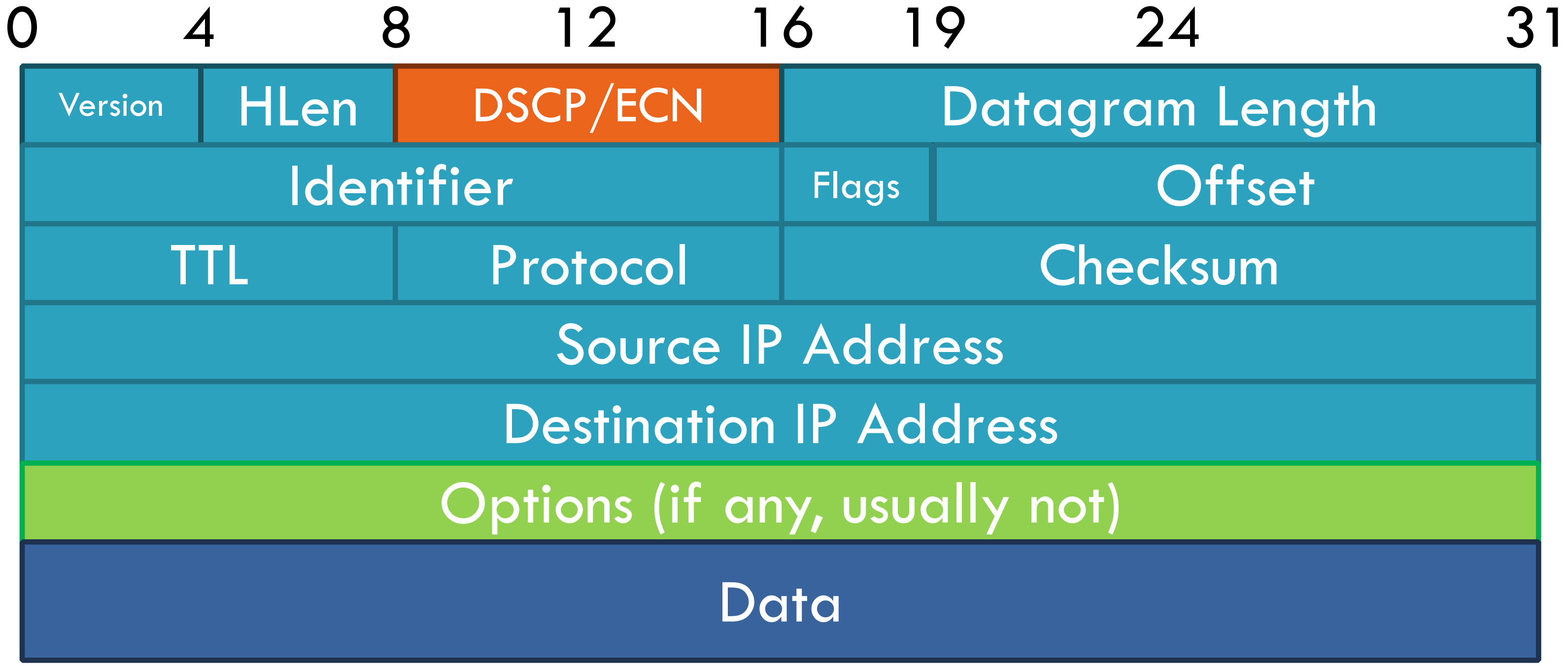
Differentiated Services (DiffServ)

- Goal: offer different levels of service to packets
  - Organized around domains (ASs)
  - Involves edge and core routers (sometimes hosts too)
- Edge routers
  - Sort packets into classes (based on many factors)
  - Set bits (**D**iff**S**erv **C**ode **P**oint) in packet headers
  - Police/shape traffic

Differentiated Services (DiffServ)

- Goal: offer different levels of service to packets
  - Organized around domains (ASs)
  - Involves edge and core routers (sometimes hosts too)
- Edge routers
  - Sort packets into classes (based on many factors)
  - Set bits (**D**iff**S**erv **C**ode **P**oint) in packet headers
  - Police/shape traffic
- Core Routers
  - Handle per-hop packet behavior based on DSCP

IP Header, Revisited



0 4 8 12 16 19 24 31

Version HLen DSCP/ECN Datagram Length

Identifier Flags Offset

TTL Protocol Checksum

Source IP Address

Destination IP Address

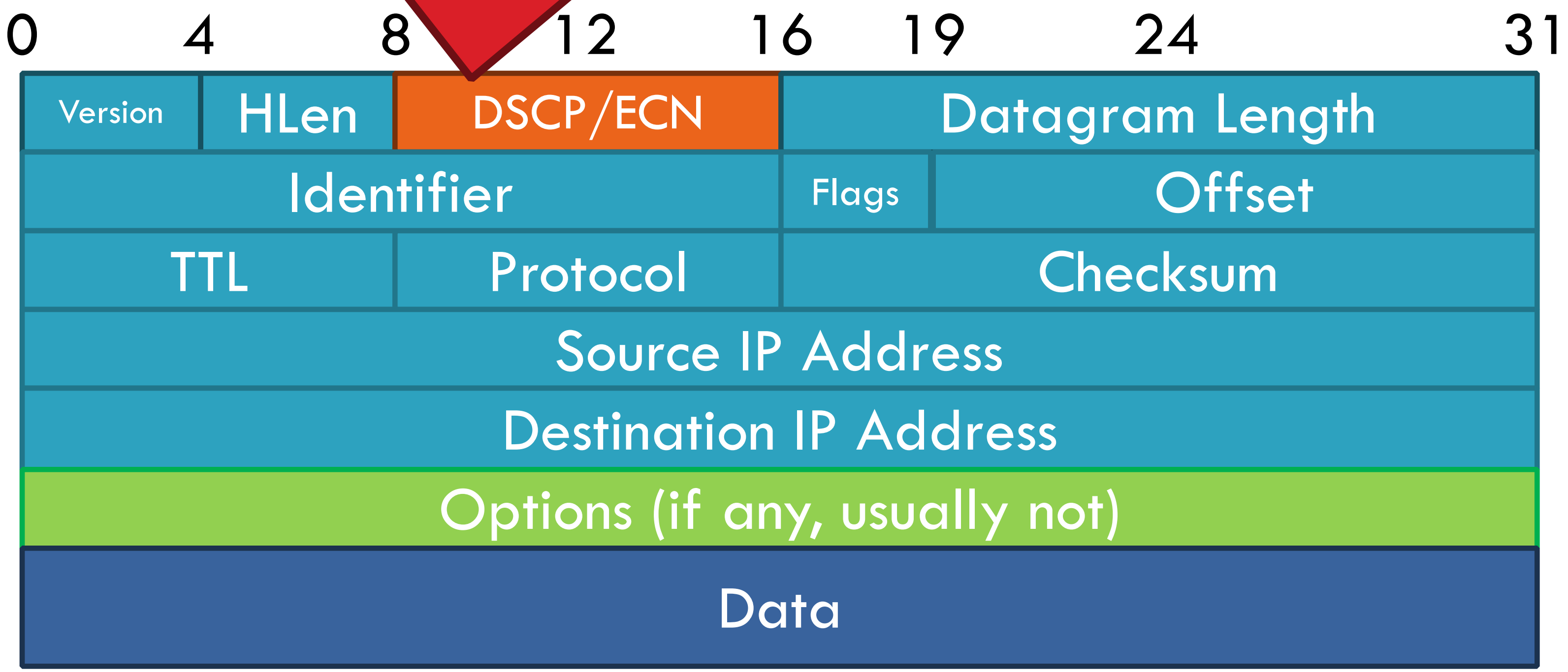
Options (if any, usually not)

Data

IP Header, Revisited

DiffServ Code Point

Used to label the class of the packet



0 4 8 12 16 19 24 31

Version | HLen | DSCP/ECN | Datagram Length

Identifier | Flags | Offset

TTL | Protocol | Checksum

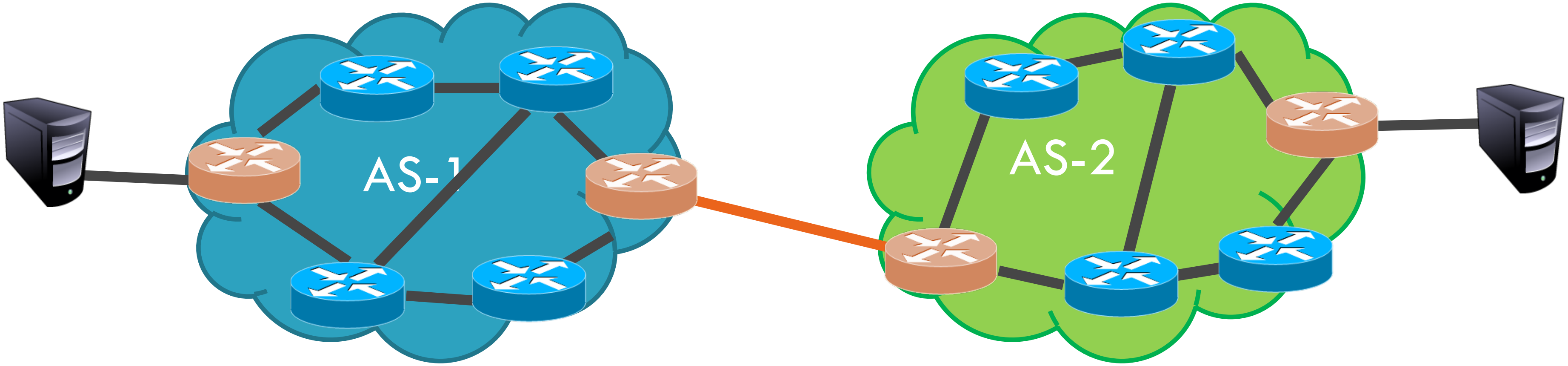
Source IP Address

Destination IP Address

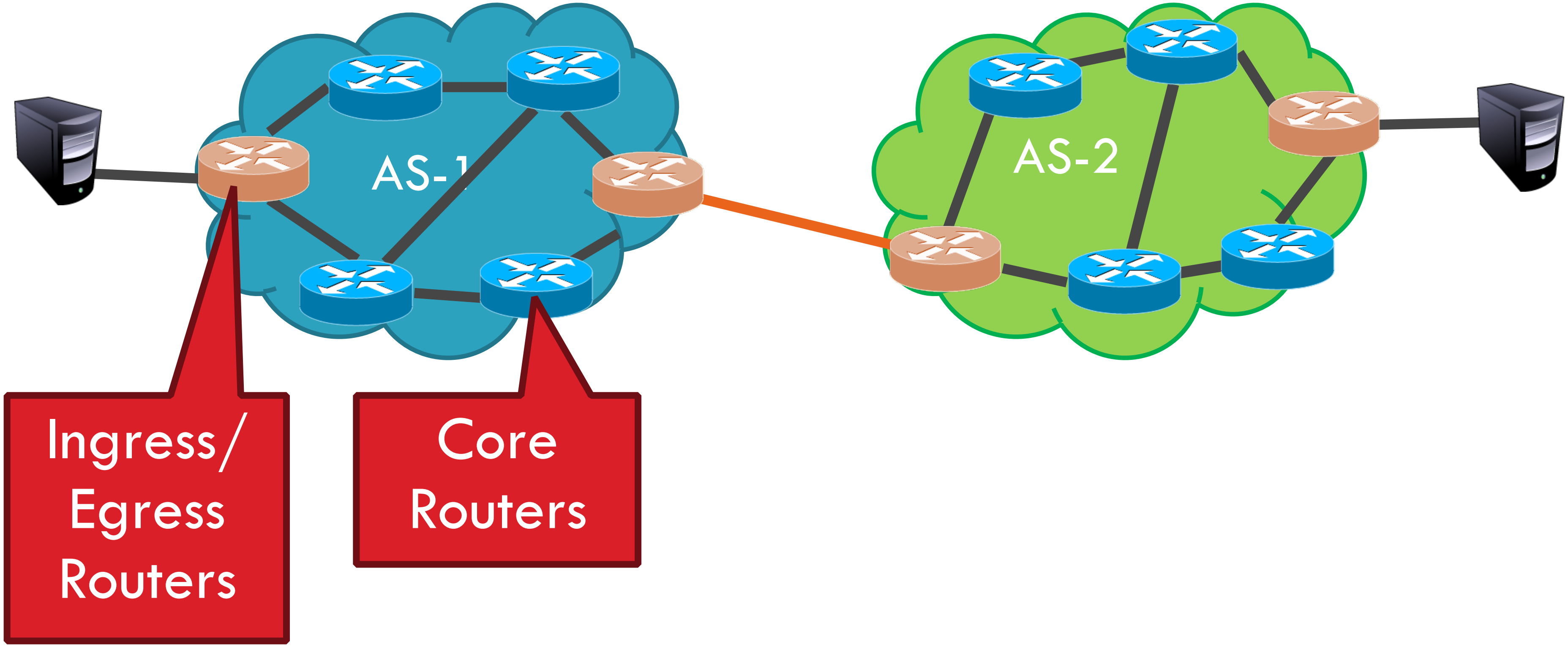
Options (if any, usually not)

Data

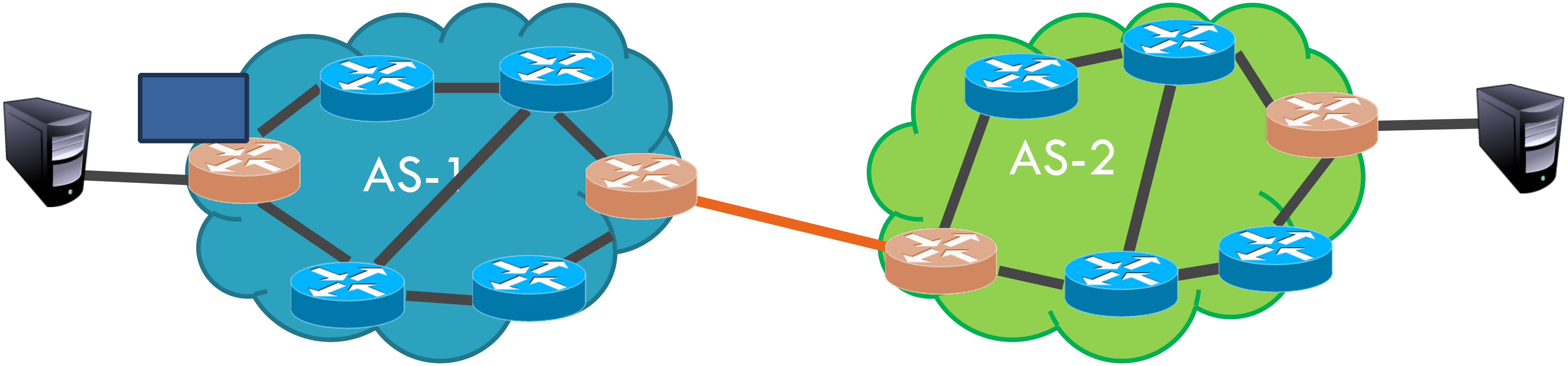
DiffServ at a High-Level



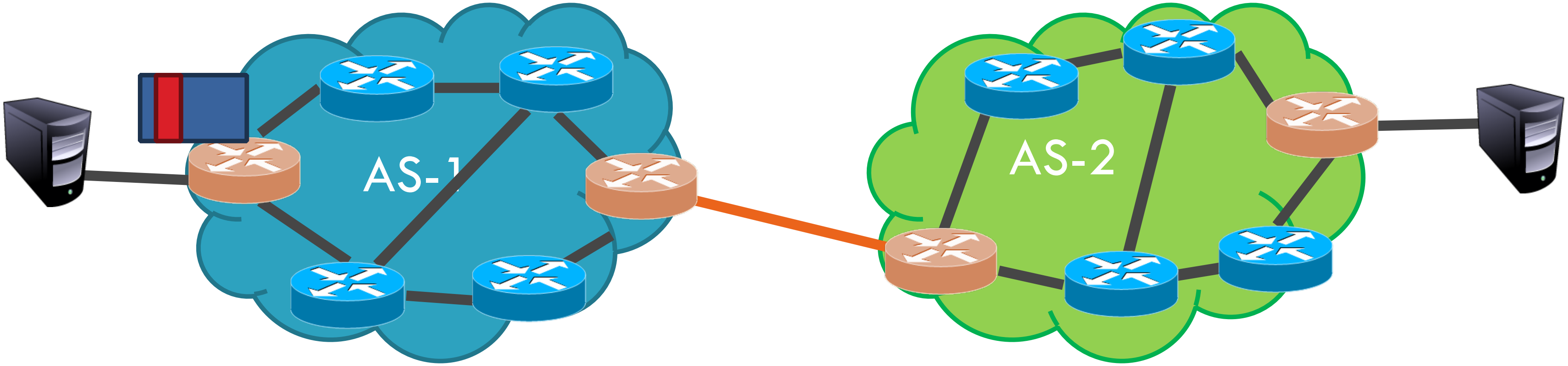
DiffServ at a High-Level



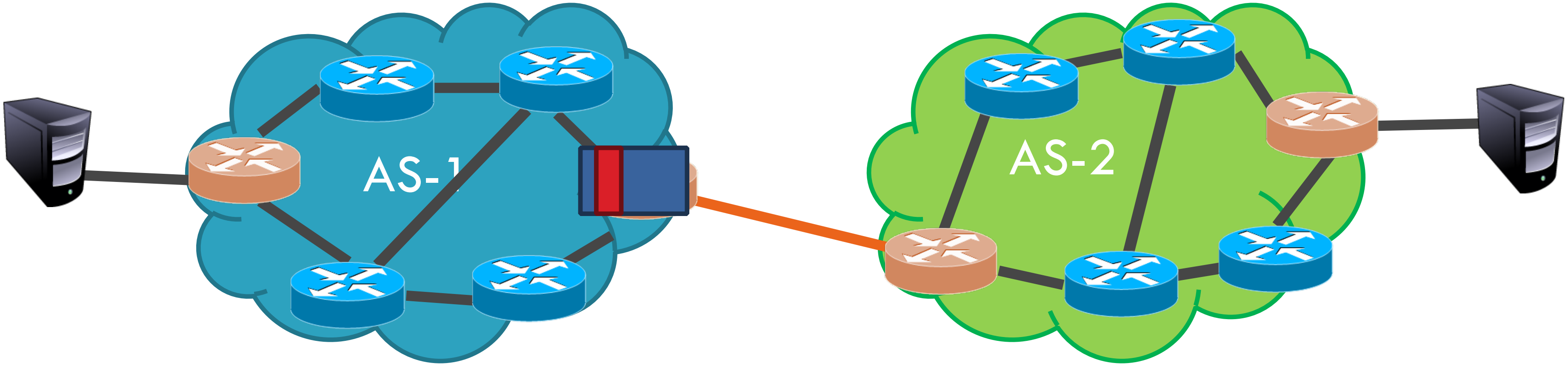
DiffServ at a High-Level



DiffServ at a High-Level

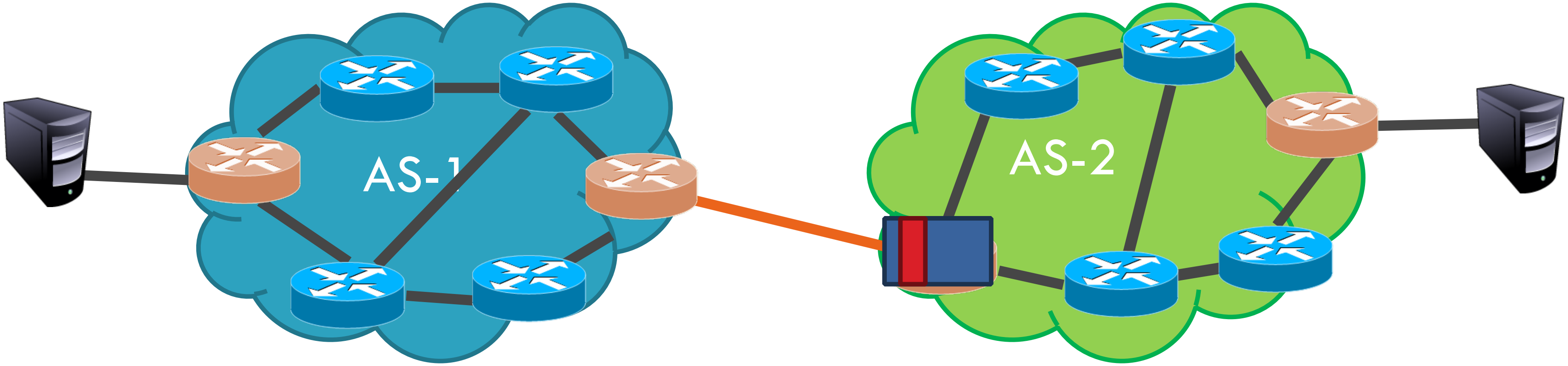


DiffServ at a High-Level



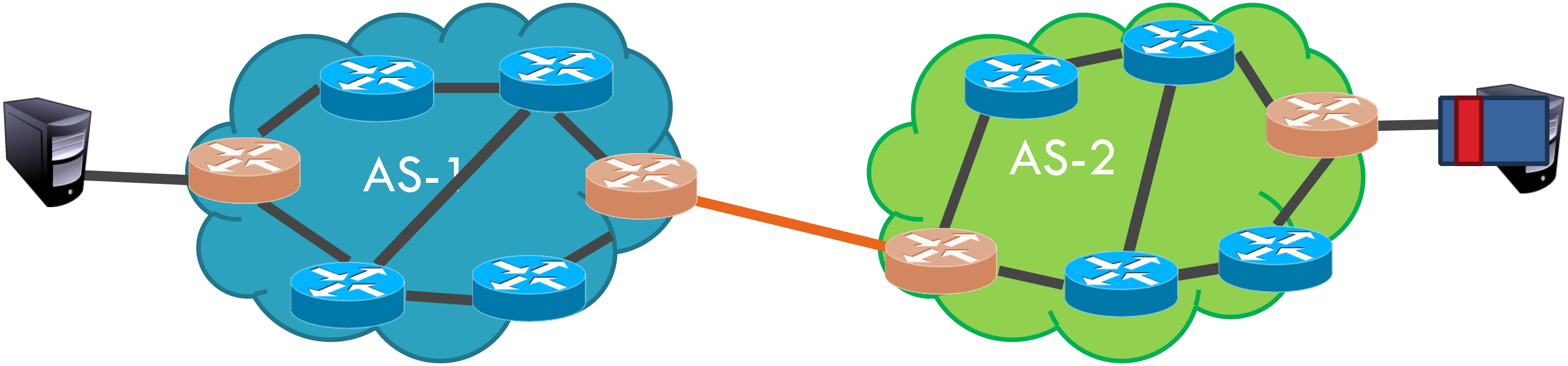
- Ingress/Egress routers assign class to each packet
  - Must analyze each packet, high overhead

DiffServ at a High-Level



- Ingress/Egress routers assign class to each packet
  - Must analyze each packet, high overhead
- Core routers use classes to do priority queuing

DiffServ at a High-Level



- Ingress/Egress routers assign class to each packet
  - Must analyze each packet, high overhead
- Core routers use classes to do priority queuing
- Classes may switch between AS boundaries

Per-Hop Behavior

- Traffic classes indicated by 6-bit DSCP in IP header
  - In practice, only 3 classes used

1. Default PHB: best-effort forwarding
   - Same as usual for the Internet

Per-Hop Behavior

- Traffic classes indicated by 6-bit DSCP in IP header
  - In practice, only 3 classes used

1. Default PHB: best-effort forwarding
   - Same as usual for the Internet
2. Expedited Forwarding (EF) PHB
   - Traffic requiring low delay, low loss, low jitter
   - Often given strict priority queuing above other classes
   - Admission control limits to 30% of capacity
     - Why?

Per-Hop Behavior

- Traffic classes indicated by 6-bit DSCP in IP header
  - In practice, only 3 classes used

1. Default PHB: best-effort forwarding
   - Same as usual for the Internet
2. Expedited Forwarding (EF) PHB
   - Traffic requiring low delay, low loss, low jitter
   - Often given strict priority queuing above other classes
   - Admission control limits to 30% of capacity
     - Why?
3. Assured Forwarding (AF) PHB
   - More general class with assurance of delivery
   - Only if traffic rate < subscribed rate

How do we Classify Packets?

- It depends.

How do we Classify Packets?

- It depends.
  - Based on ports
    - i.e. 80 (HTTP) takes precedence over 21 (FTP)

How do we Classify Packets?

- It depends.
  - Based on ports
    - i.e. 80 (HTTP) takes precedence over 21 (FTP)
  - Based on application
    - i.e. HTTP takes precedence over BitTorrent

How do we Classify Packets?

- It depends.
  - Based on ports
    - i.e. 80 (HTTP) takes precedence over 21 (FTP)
  - Based on application
    - i.e. HTTP takes precedence over BitTorrent
  - Based on location
    - i.e. home users get normal service…
    - While hospitals/policy/fire department get priority service

How do we Classify Packets?

- It depends.
  - Based on ports
    - i.e. 80 (HTTP) takes precedence over 21 (FTP)
  - Based on application
    - i.e. HTTP takes precedence over BitTorrent
  - Based on location
    - i.e. home users get normal service…
    - While hospitals/policy/fire department get priority service
  - Based on who pays more $$$
    - $100 for “premium” Internet vs. $25 “value” Internet

Traffic Policing/Shaping

- Purpose: need a mechanism to control packet flow
  - High vs. medium vs. low priority flows
  - Think of it like a toll booth

Traffic Policing/Shaping

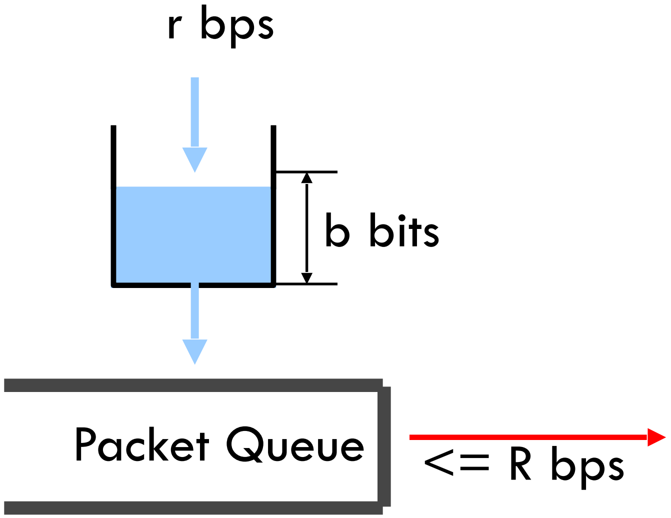
- Purpose: need a mechanism to control packet flow
  - High vs. medium vs. low priority flows
  - Think of it like a toll booth
- Token bucket (r, b)
  - r → rate the bucket fills
  - b → size of tokens in the bucket

Traffic Policing/Shaping

- Purpose: need a mechanism to control packet flow
  - High vs. medium vs. low priority flows
  - Think of it like a toll booth
- Token bucket (r, b)
  - r → rate the bucket fills
  - b → size of tokens in the bucket
- Police: if token is available, packet may pass
  - Otherwise, packet is queued or dropped
  - Queuing packets "shapes" the traffic flow

Leaky Buckets

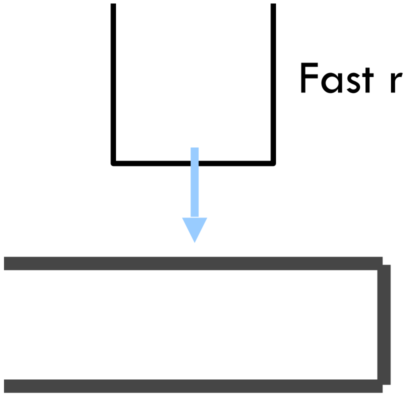
- Parameters
  - r –rate at which tokens fill the bucket
  - b – bucket depth
  - R – maximum link capacity or peak rate

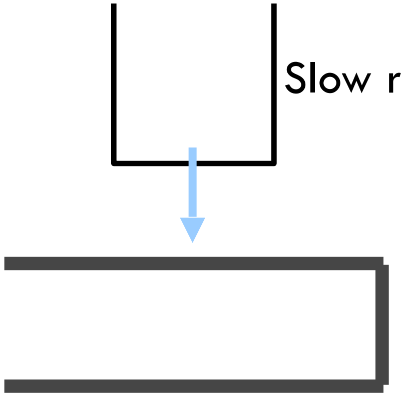


- Bits are only transmitted from a queue when there is a token of sufficient size available

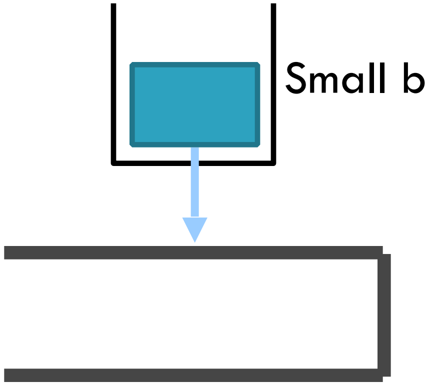
Bucket Parameters, Intuitively

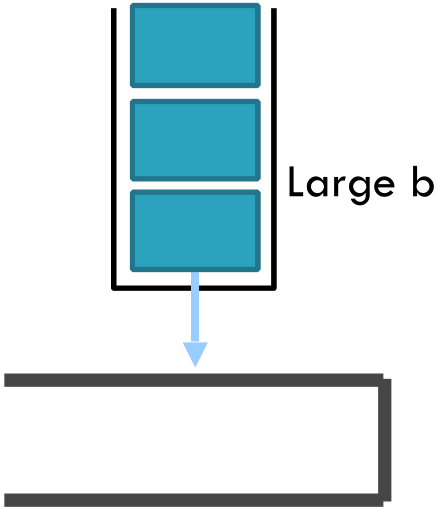
- r – speed packets can exit the queue





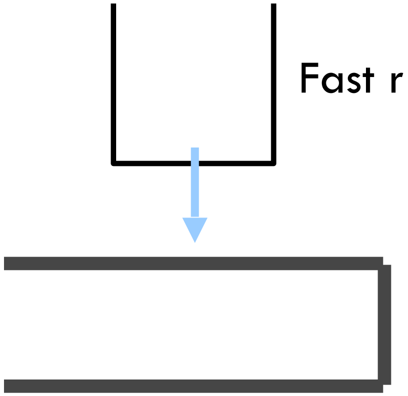
- b – burst tolerance

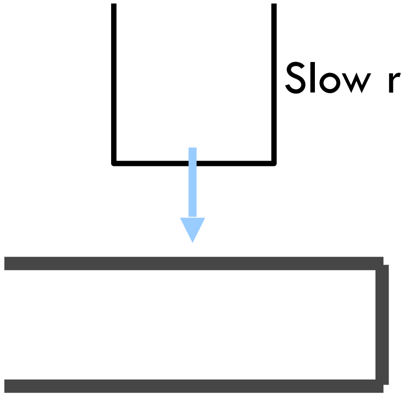




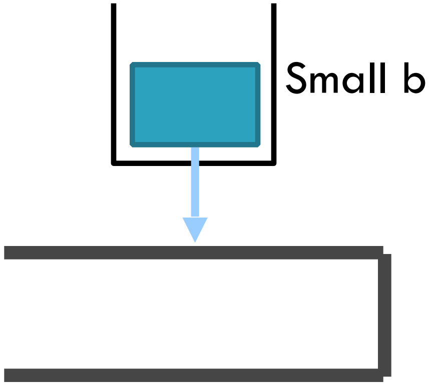
Bucket Parameters, Intuitively

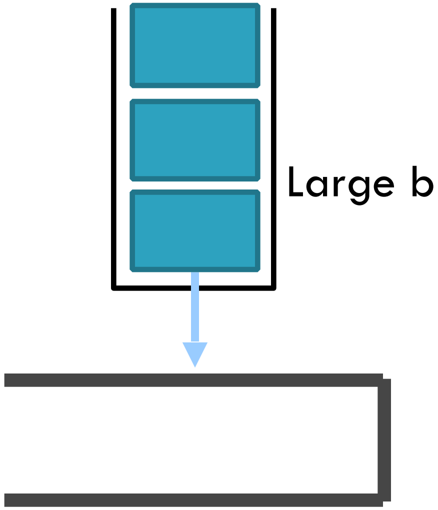
- □ r – speed packets can exit the queue





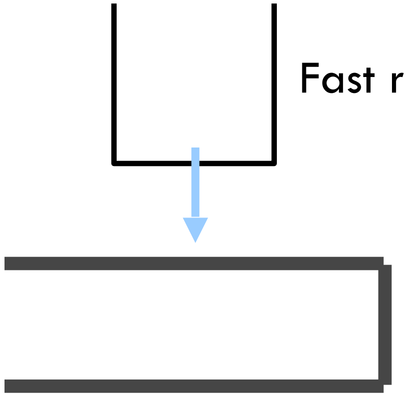
- □ b – burst tolerance

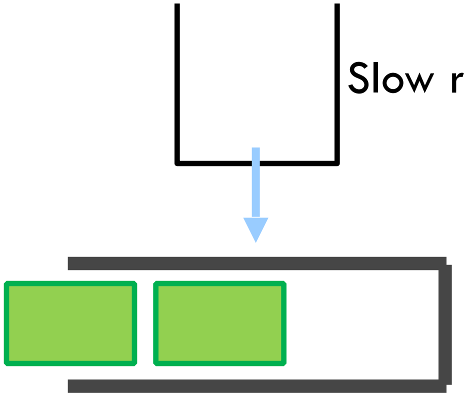




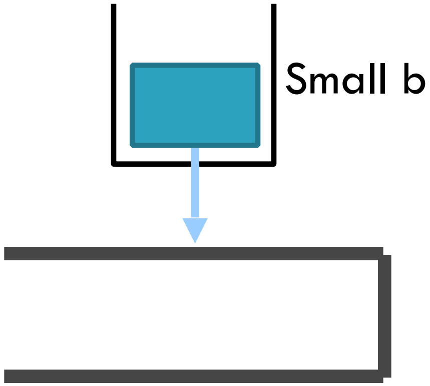
Bucket Parameters, Intuitively

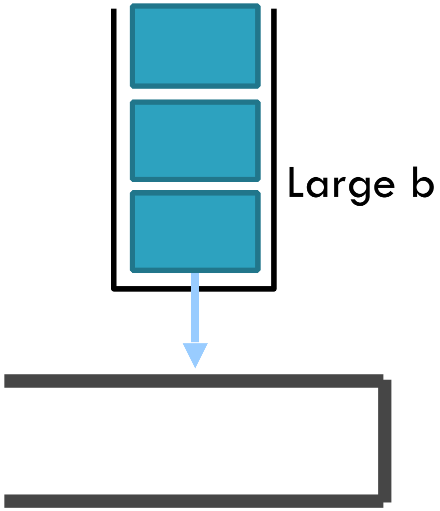
- r – speed packets can exit the queue





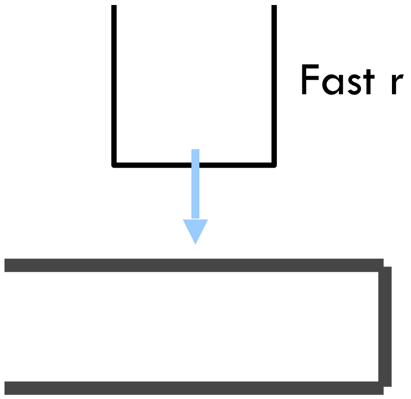
- b – burst tolerance

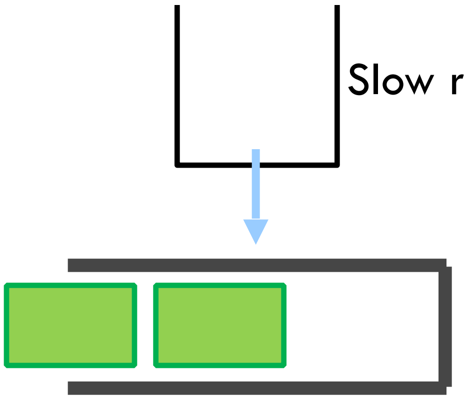




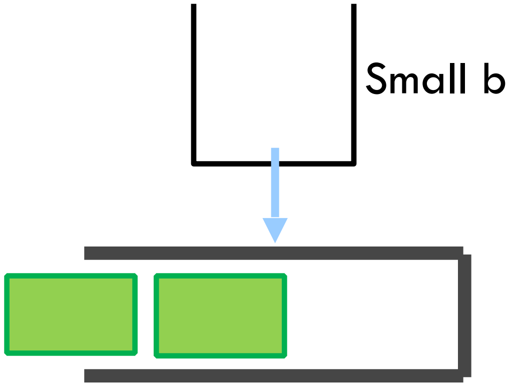
Bucket Parameters, Intuitively

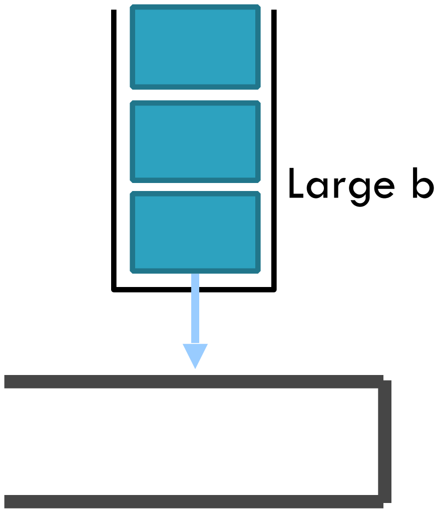
- □ r – speed packets can exit the queue





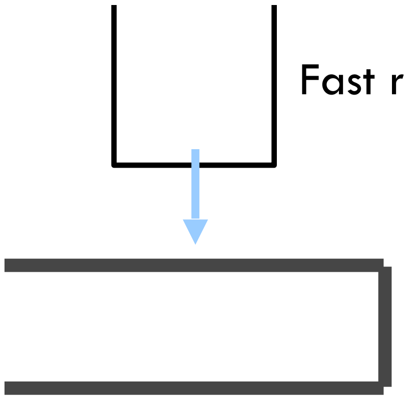
- □ b – burst tolerance

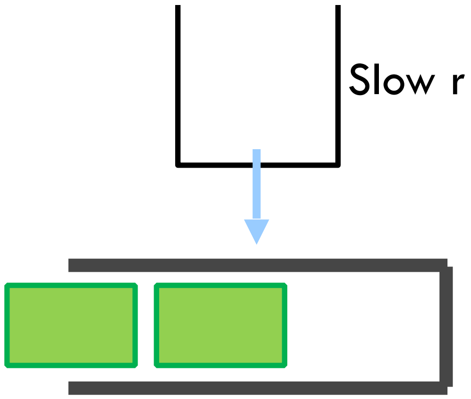




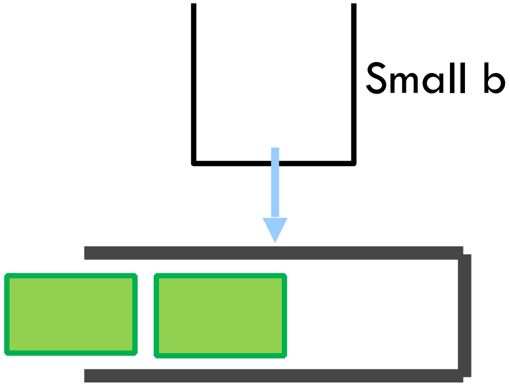
Bucket Parameters, Intuitively

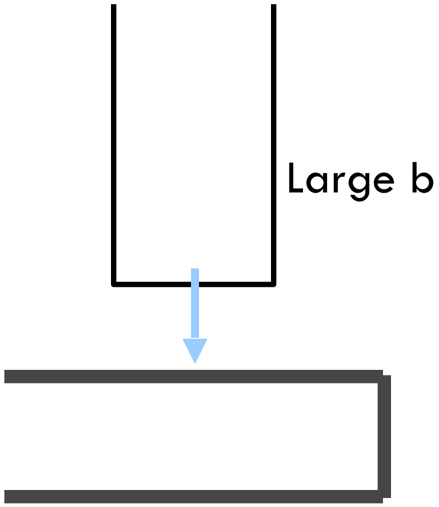
- r – speed packets can exit the queue





- b – burst tolerance





Advantages of DiffServ

- Giving priority does improve performance
  - … at the expense of reduced perf. for lower classes
- Relatively lightweight solution
  - Some overhead on ingress/egress routers
  - No per flow state, low overhead on core routers
- Easy to deploy
  - No hard reservations
  - No advanced setup of flows
  - No end-to-end negotiation

Disadvantages of DiffServ

- No performance guarantees
  - All gains are relative, not absolute
  - Classes are very coarse
    - i.e. all packets of a specific class get better performance
    - No per flow or per destination QoS
  - What if some ASs do not support DiffServ?
  - Impossible to predict end-to-end behavior
- Security
  - Any host can tag traffic as high priority
  - E.g. Win 2K tagged all traffic as high priority by default

Outline

- “Soft” QoS
  - Packet shaping/prioritization
  - DiffServ
- “Hard” QoS
  - IntServ

From Relative to Absolute Service

- Priority mechanisms can only deliver absolute assurances if total load is regulated

From Relative to Absolute Service

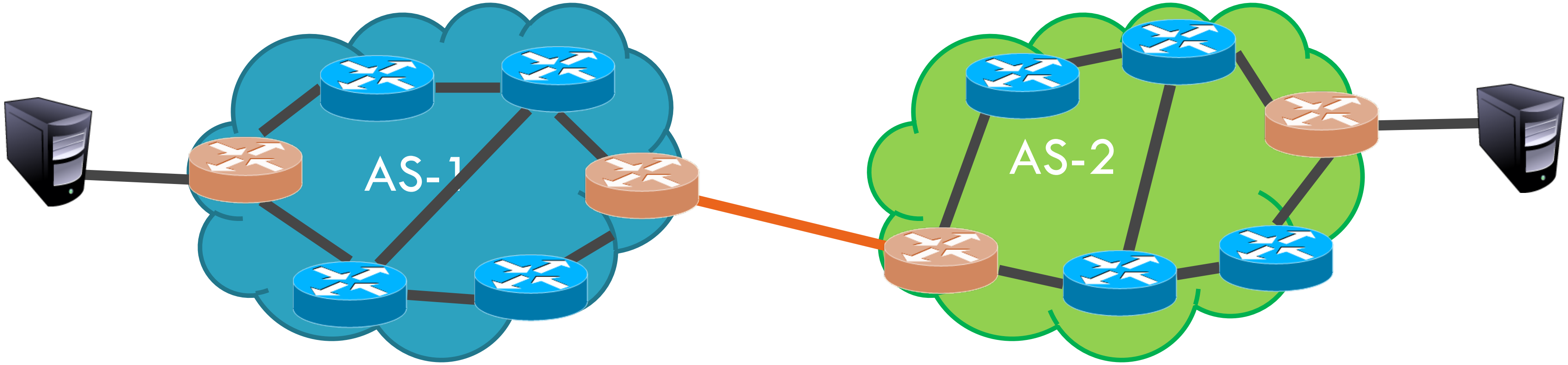
- Priority mechanisms can only deliver absolute assurances if total load is regulated
- Service Level Agreements (SLAs) specify:
  - Amount user (organization, etc.) can send
  - Level of service delivered to that traffic

From Relative to Absolute Service

- Priority mechanisms can only deliver absolute assurances if total load is regulated
- Service Level Agreements (SLAs) specify:
  - Amount user (organization, etc.) can send
  - Level of service delivered to that traffic
- DiffServ offers low (but unspecified) delay and no drops
  - Acceptance of proposed SLAs managed by "Bandwidth Broker"
  - Only over long time scales

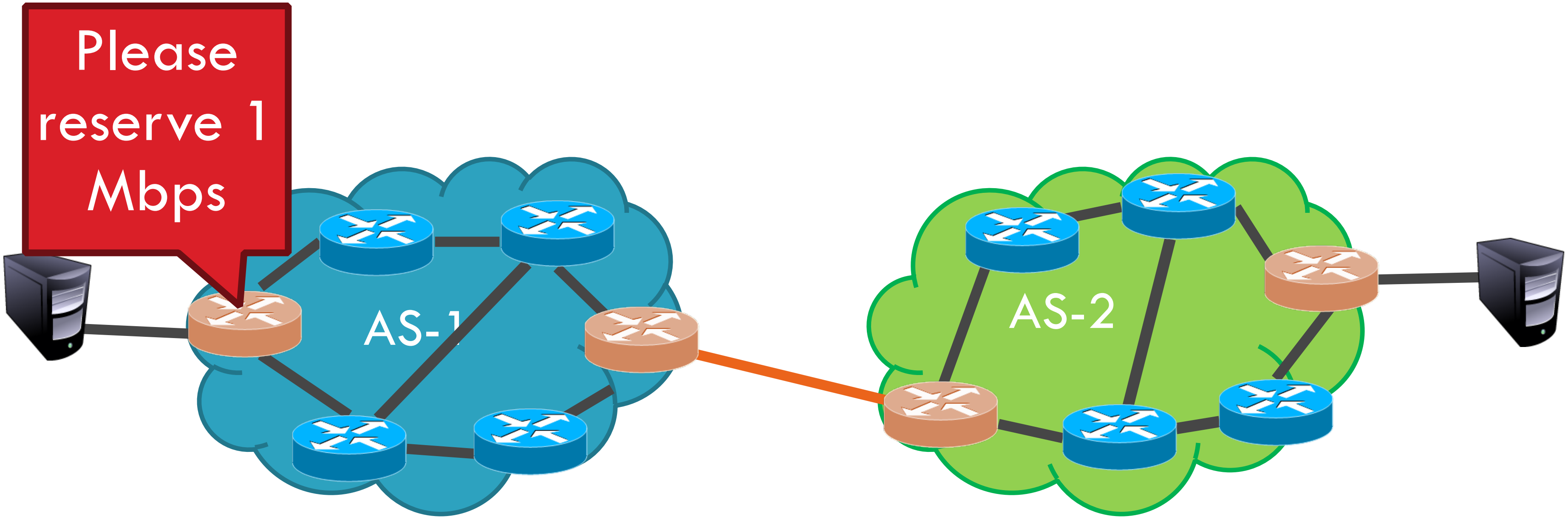
Inter-Domain Premium DiffServ

- Goal of IntServ: end-to-end bandwidth guarantees
- Mechanism: end-to-end bandwidth reservations
  - Like the telephone network, circuit reservations
  - End hosts ask for reserved capacity from the network



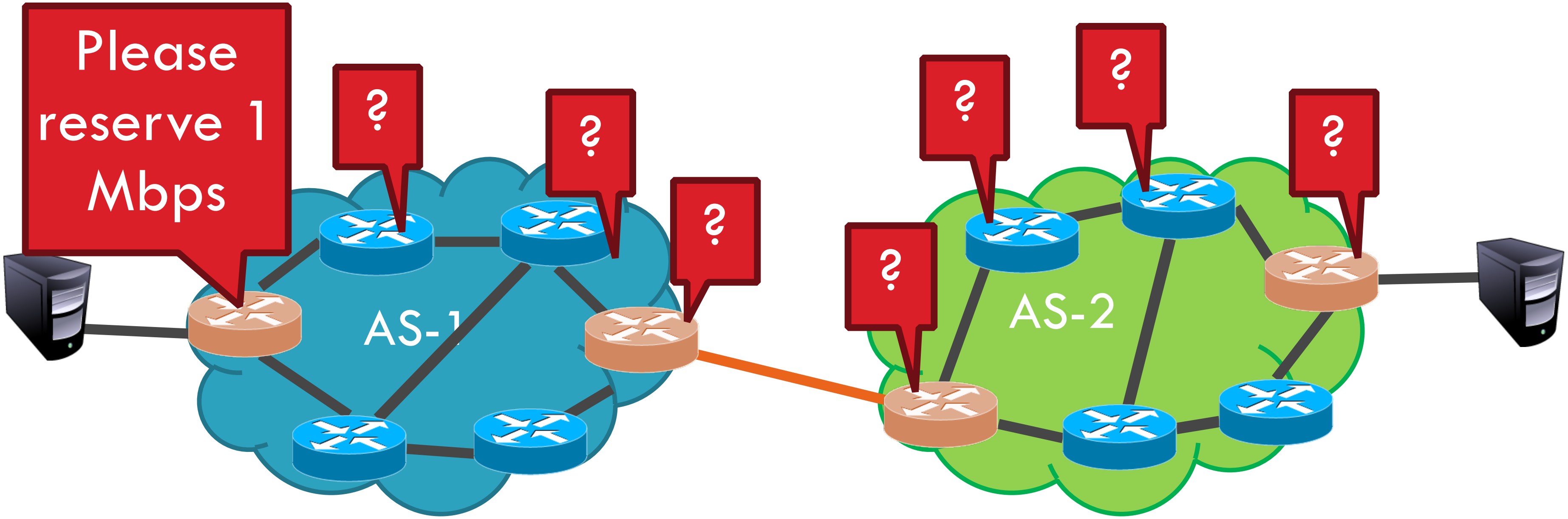
Inter-Domain Premium DiffServ

- Goal of IntServ: end-to-end bandwidth guarantees
- Mechanism: end-to-end bandwidth reservations
  - Like the telephone network, circuit reservations
  - End hosts ask for reserved capacity from the network



Inter-Domain Premium DiffServ

- Goal of IntServ: end-to-end bandwidth guarantees
- Mechanism: end-to-end bandwidth reservations
  - Like the telephone network, circuit reservations
  - End hosts ask for reserved capacity from the network



Philosophical Battle

- If your network allows reservations:
  - Then you must perform admission control
  - i.e. who can make reservations, and when?

Philosophical Battle

- If your network allows reservations:
  - Then you must perform admission control
  - i.e. who can make reservations, and when?
- Basic Question:
  - Should all flows be admitted (current Internet)
  - Or, should we refuse some flows to guarantee good service for reserved flows (IntServ Internet)

Philosophical Battle

- If your network allows reservations:
  - Then you must perform admission control
  - i.e. who can make reservations, and when?
- Basic Question:
  - Should all flows be admitted (current Internet)
  - Or, should we refuse some flows to guarantee good service for reserved flows (IntServ Internet)
- Which one is right?!?!

High-Level IntServ Design

- Reservations are made by endpoints
  - Applications know their own requirements
  - Applications run on end-hosts
  - Network does not need to guess about requirements

High-Level IntServ Design

- Reservations are made by endpoints
  - Applications know their own requirements
  - Applications run on end-hosts
  - Network does not need to guess about requirements
- Guarantees are end-to-end on a per-flow basis

High-Level IntServ Design

- Reservations are made by endpoints
  - Applications know their own requirements
  - Applications run on end-hosts
  - Network does not need to guess about requirements
- Guarantees are end-to-end on a per-flow basis
- Soft-state
  - State in routers constantly refreshed by endpoints

High-Level IntServ Design

- Reservations are made by endpoints
  - Applications know their own requirements
  - Applications run on end-hosts
  - Network does not need to guess about requirements
- Guarantees are end-to-end on a per-flow basis
- Soft-state
  - State in routers constantly refreshed by endpoints
- IntServ is multicast-oriented
  - Assumed that large broadcasts would drive multicast and IntServ deployment
  - This is why reservations are made by receivers

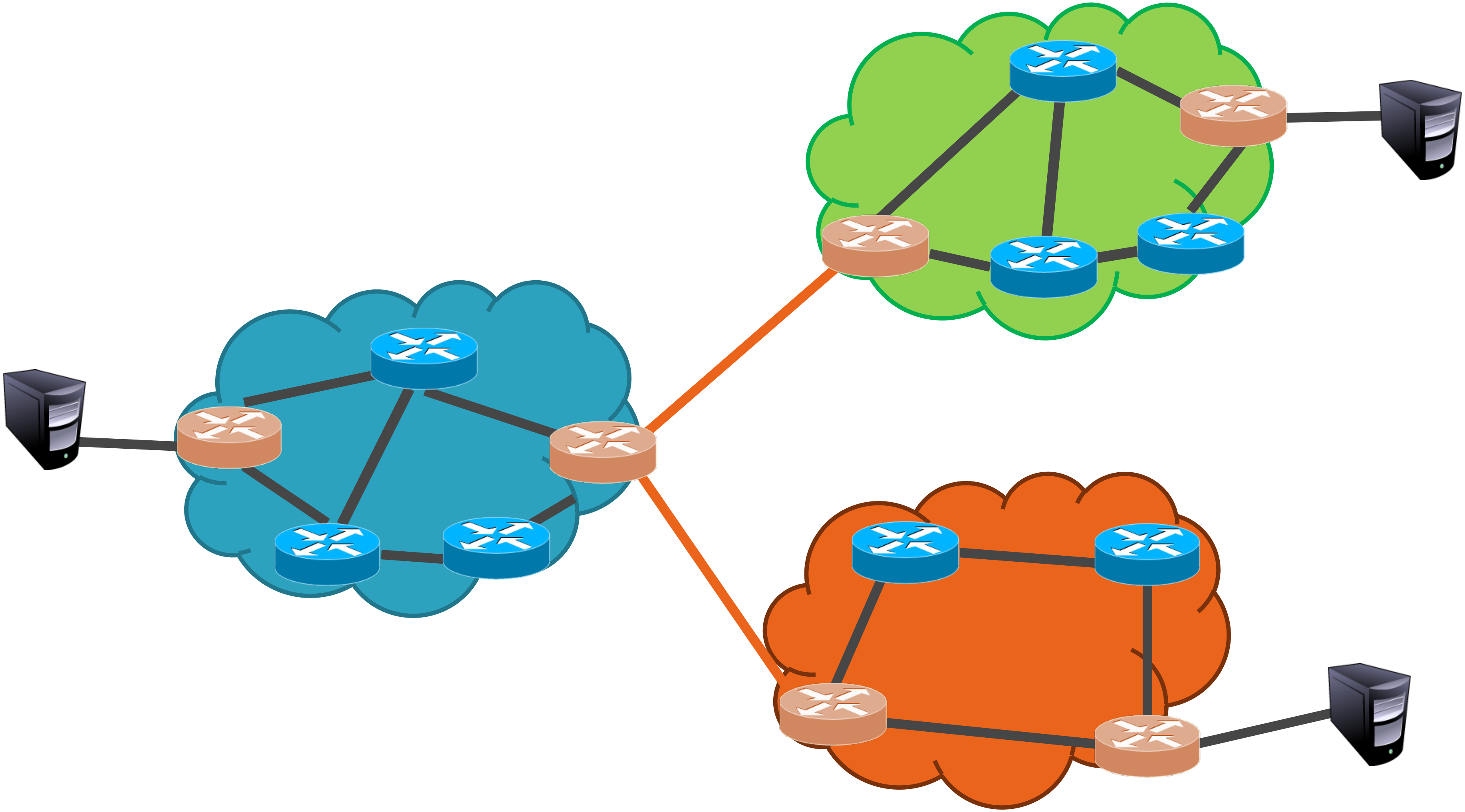
Requirements for IntServ

- Fixed, stable paths
  - Only routers on the path know about the reservation
  - Current Internet cannot guarantee this
- Routers maintain per-flow state
  - Very high overhead (even with soft-state)
- State is used to reserve bandwidth
  - Guarantees QoS for reserved flows
  - … but some flows may not be admitted
  - Security?

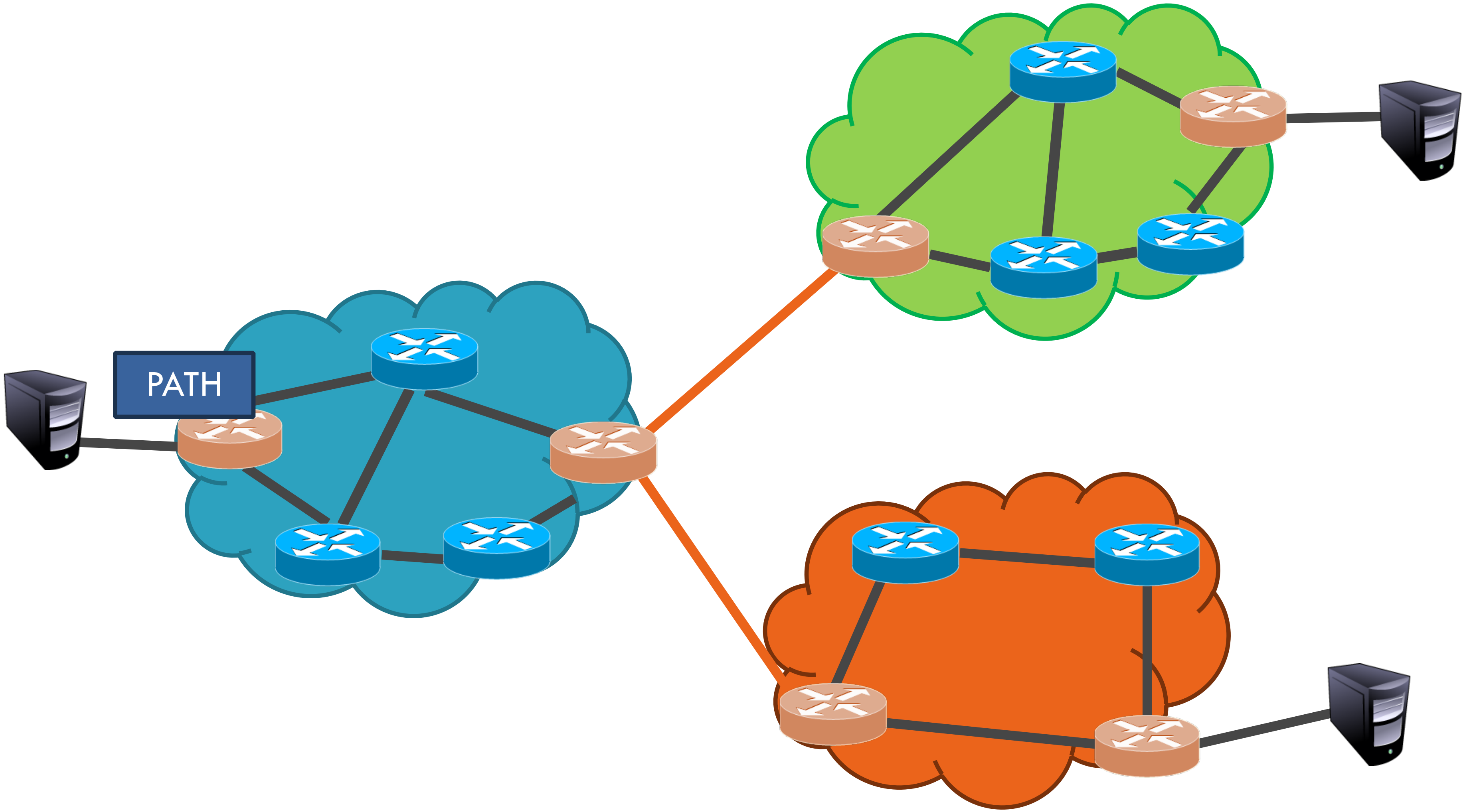
RSVP Reservation Protocol

- Performs signaling to set up reservation state
  - Initiated by the receiver
- Each reservation is a simplex data flow sent to a unicast or multicast address
  - <Destination IP, protocol # (TCP, UDP), port #>
- Multiple senders/receivers can be in the same session

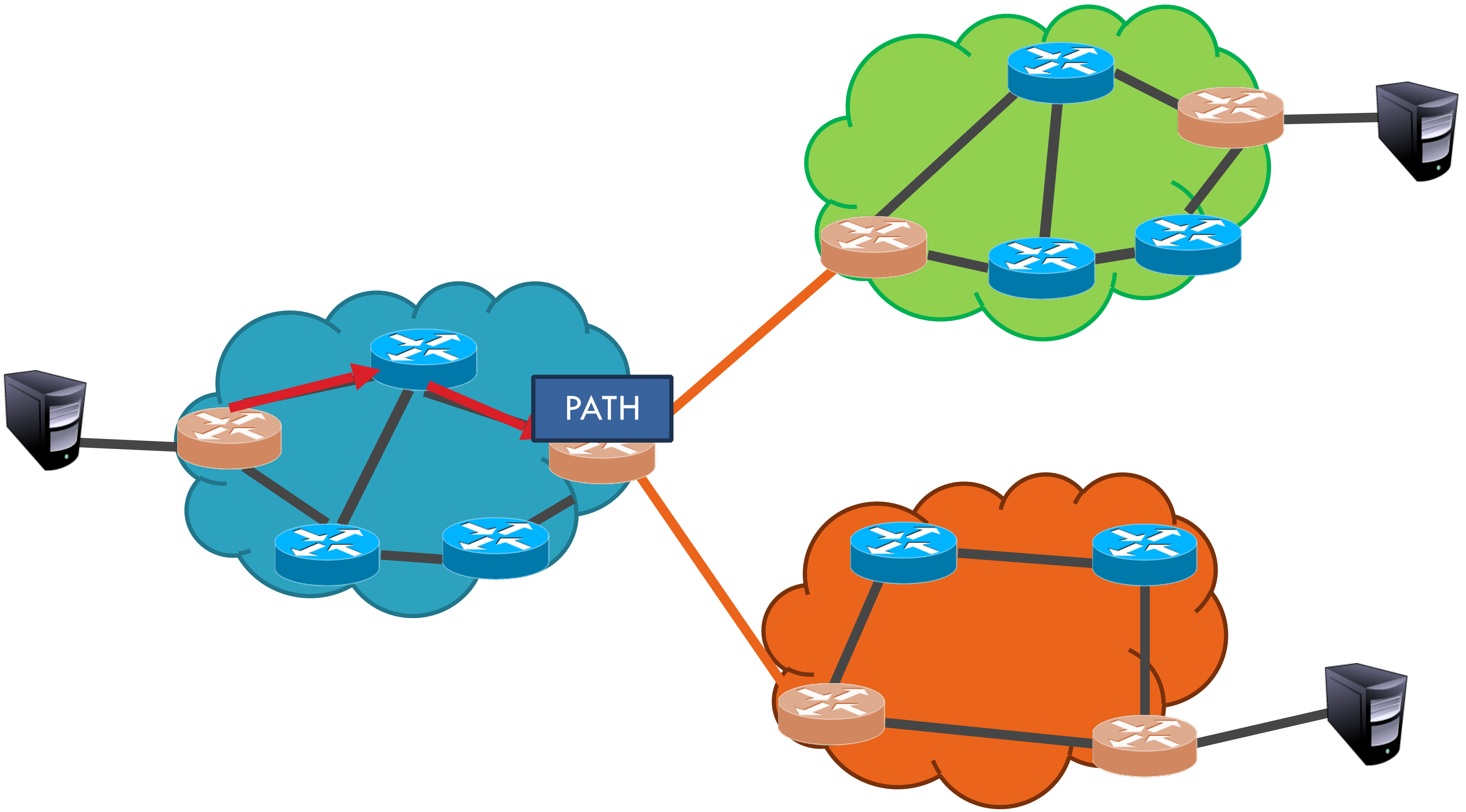
RSVP Example



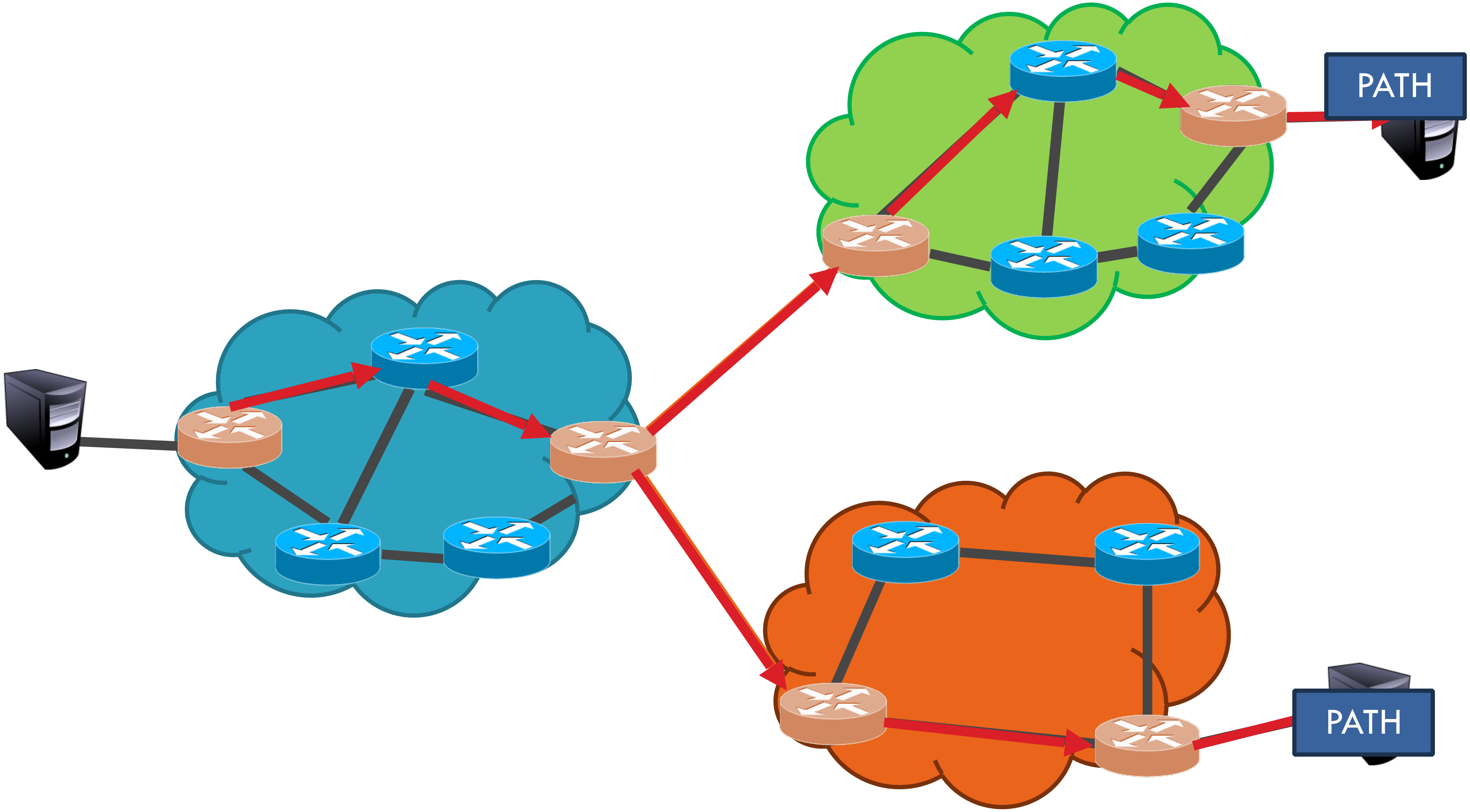
RSVP Example



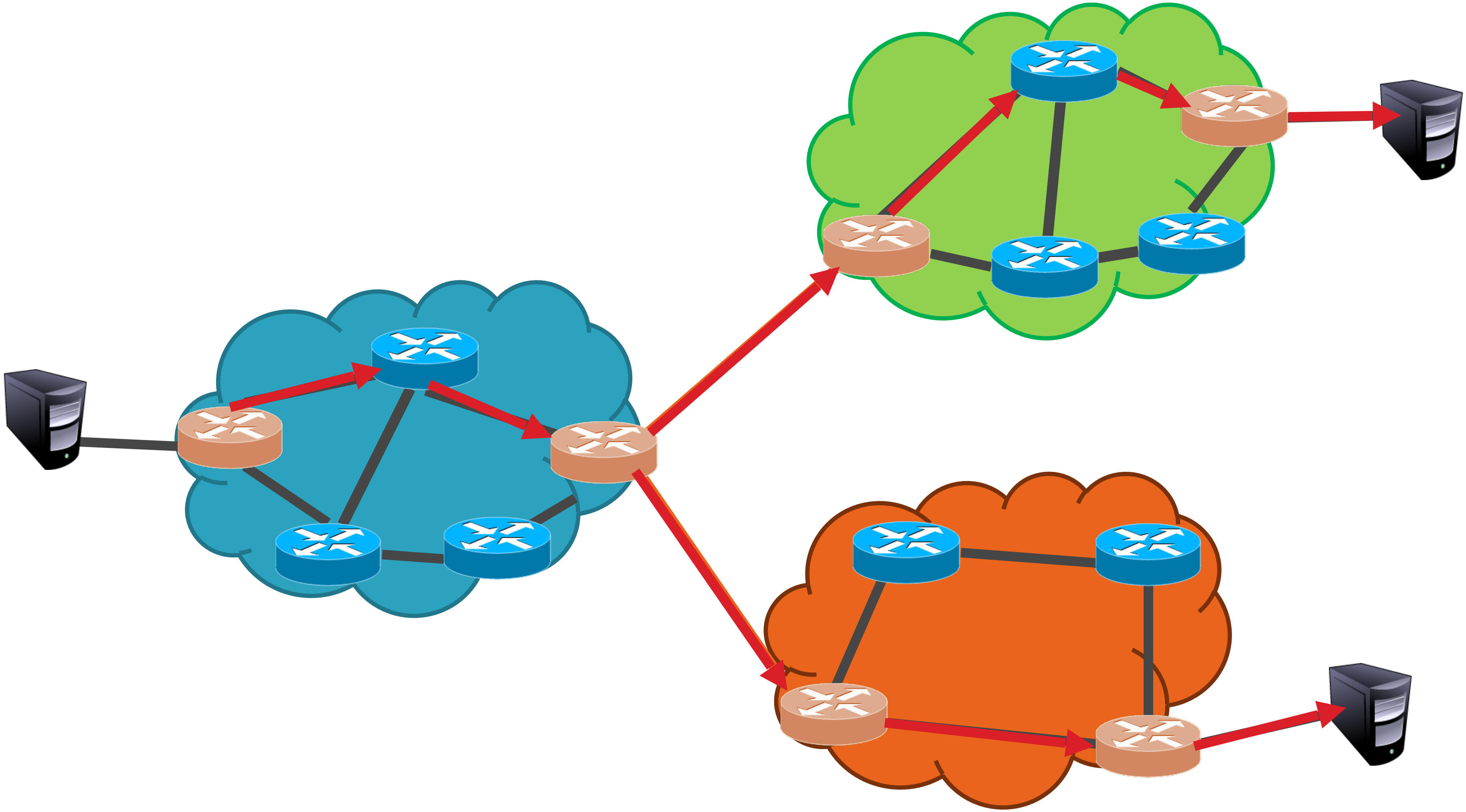
RSVP Example



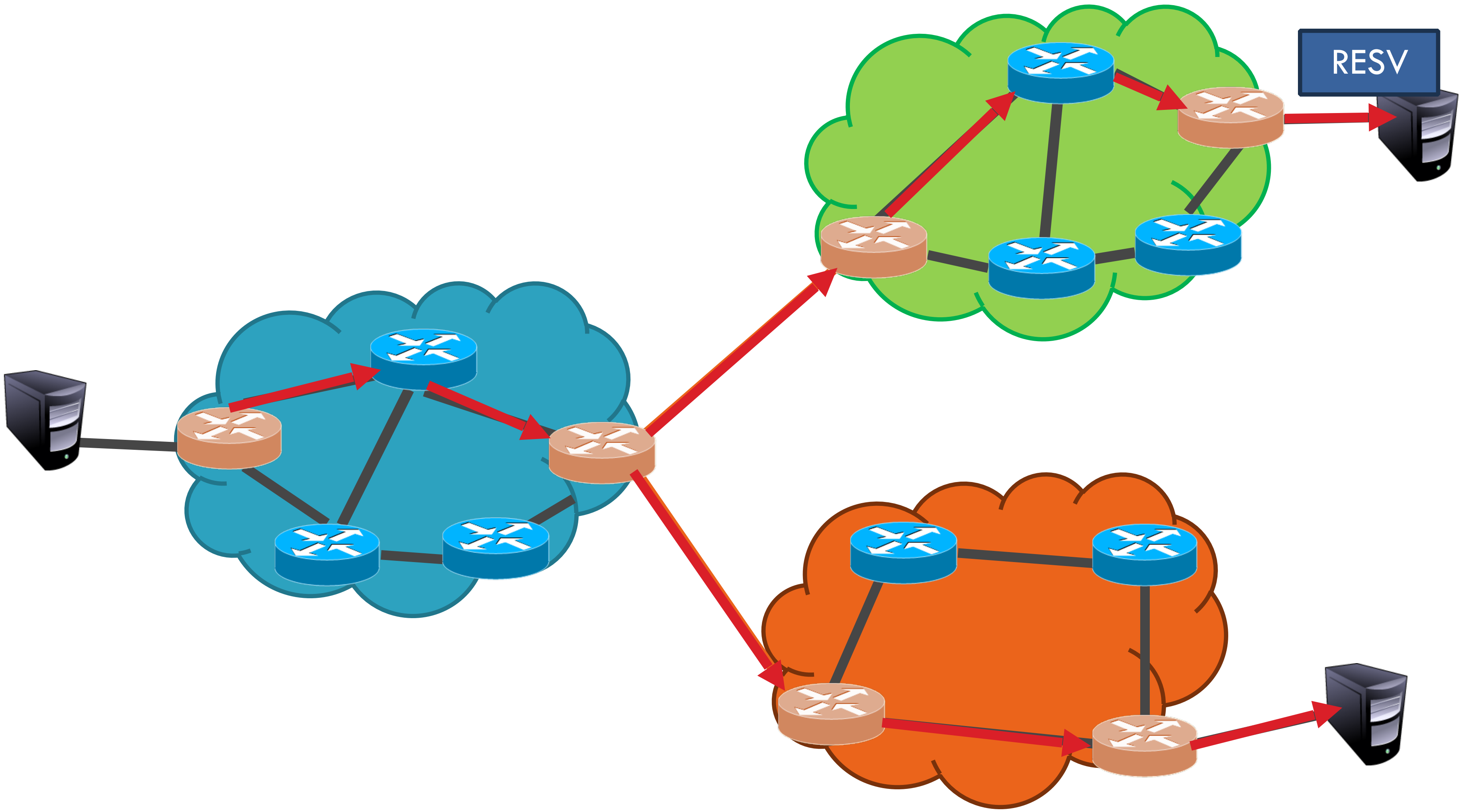
RSVP Example



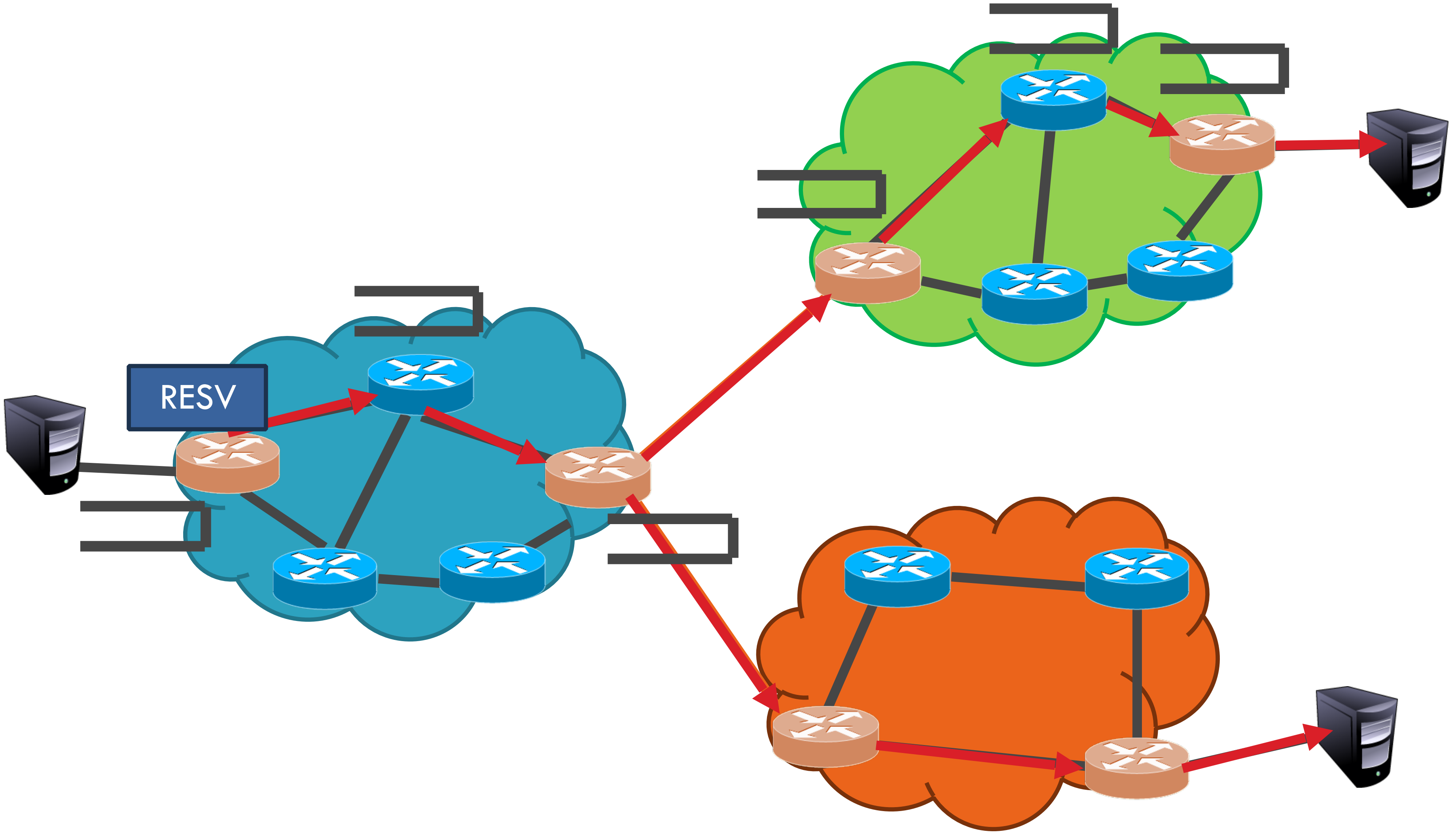
RSVP Example



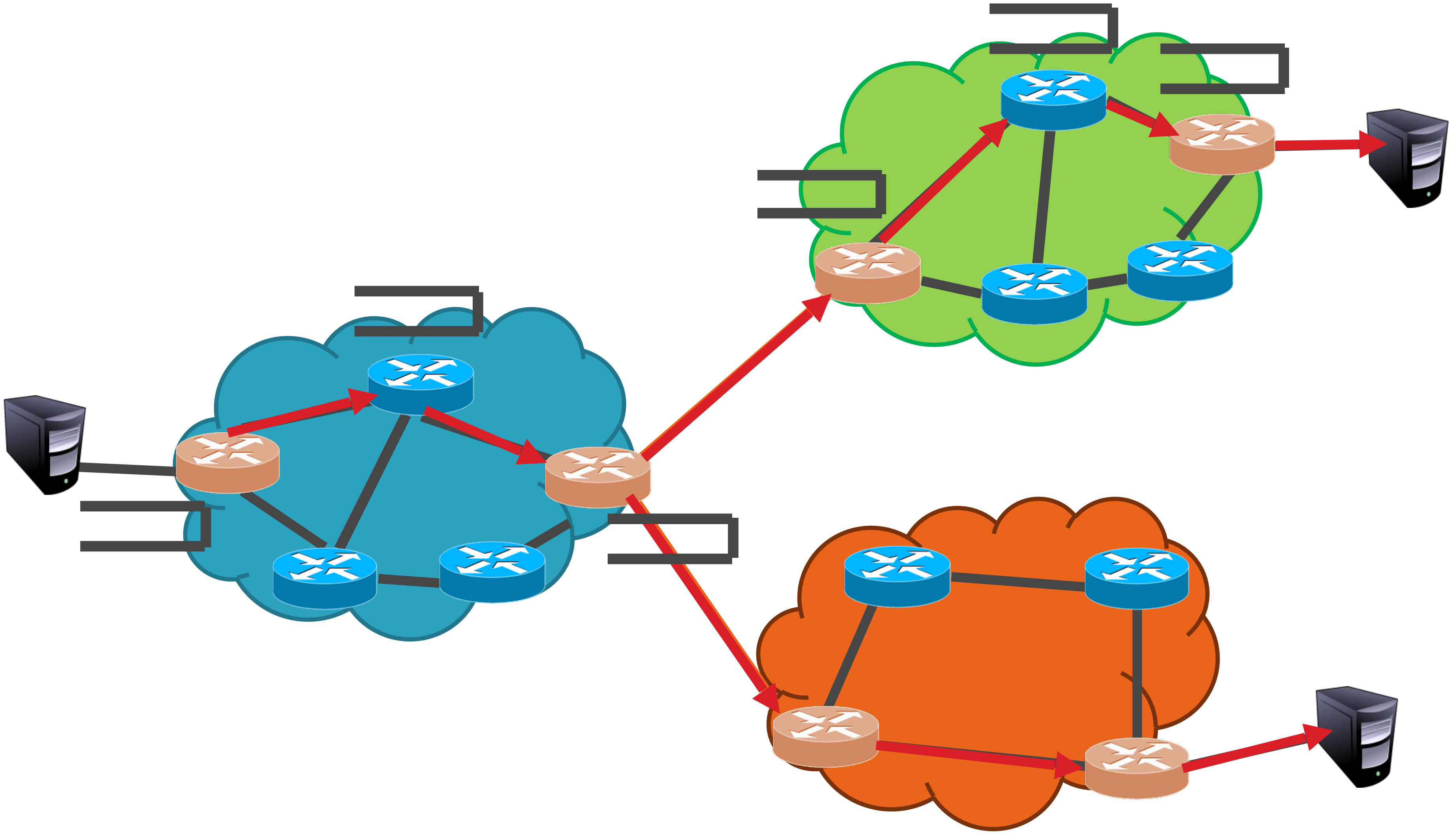
RSVP Example



RSVP Example

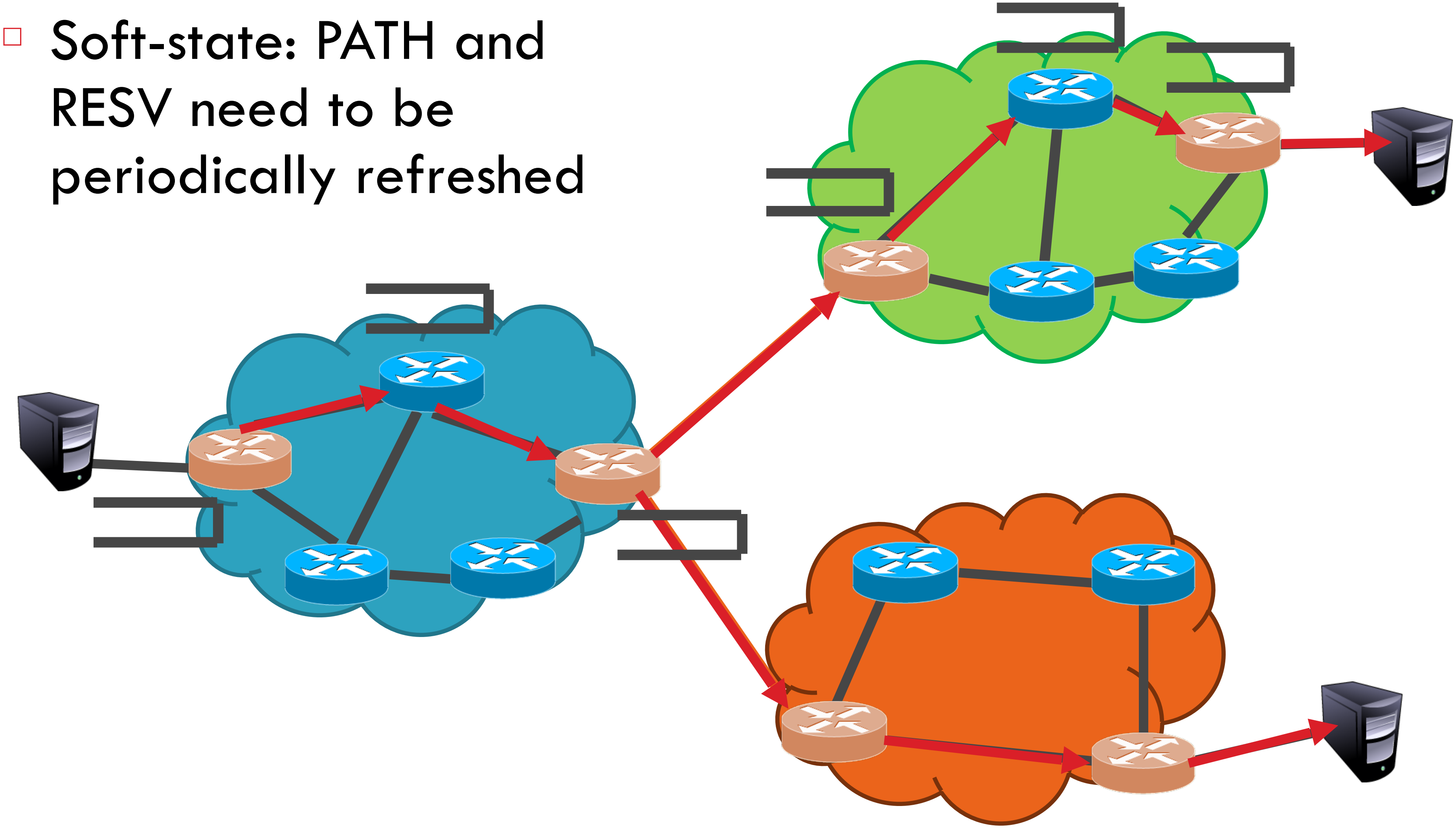


RSVP Example



RSVP Example

- Soft-state: PATH and RESV need to be periodically refreshed



IntServ Summary

- The good:
  - Reservations are guaranteed and precise
    - Reserved bandwidth is not shared with a class
    - Tight allocations for each flow
  - Soft-state slightly reduces overhead on routers

IntServ Summary

- The good:
  - Reservations are guaranteed and precise
    - Reserved bandwidth is not shared with a class
    - Tight allocations for each flow
  - Soft-state slightly reduces overhead on routers
- The bad:
  - IntServ is a whole Internet upgrade
  - Heavyweight mechanisms, per flow state
  - Security: end-hosts can DoS by reserving lots of bandwidth

QoS on the Internet Today

- QoS was huge in the ‘90s
  - DiffServ and IntServ are both IETF standards
  - … yet neither are widely deployed today

QoS on the Internet Today

- QoS was huge in the ‘90s
  - DiffServ and IntServ are both IETF standards
  - … yet neither are widely deployed today
- Internet capacity explosion
  - Packets only drop when capacity is reached
  - QoS is only useful if capacity is saturated
- After the 2000s Internet boom…
  - Huge glut of “dark” fiber capacity
  - Lots of spare capacity = little need for QoS

QoS on the Internet Today

- QoS was huge in the '90s
  - DiffServ and IntServ are both IETF standards
  - … yet neither are widely deployed today
- Internet capacity explosion
  - Packets only drop when capacity is reached
  - QoS is only useful if capacity is saturated
- After the 2000s Internet boom…
  - Huge glut of "dark" fiber capacity
  - Lots of spare capacity = little need for QoS
- Another technical solution killed by economics

QoS is Controversial

- Example: ISP sometimes drop or throttle BitTorrent
  - Net neutrality: is it okay to favor one kind of traffic over another?
  - Privacy: is it okay for ISPs to snoop on packets?

QoS is Controversial

- Example: ISP sometimes drop or throttle BitTorrent
  - Net neutrality: is it okay to favor one kind of traffic over another?
  - Privacy: is it okay for ISPs to snoop on packets?
- Counterargument: BitTorrent negatively impacts other customers
  - Is it okay for your neighbors BitTorrent traffic to make your Skype calls sound choppy?

QoS is Controversial

- Example: ISP sometimes drop or throttle BitTorrent
  - Net neutrality: is it okay to favor one kind of traffic over another?
  - Privacy: is it okay for ISPs to snoop on packets?
- Counterargument: BitTorrent negatively impacts other customers
  - Is it okay for your neighbors BitTorrent traffic to make your Skype calls sound choppy?
- Slippery slope:
  - Who decides which apps are favored?
  - Is it okay to ban apps entirely?
  - Is it okay to allow people to pay for higher priority?