# CpE654 / NIS654
# Design and Analysis of Network Systems

Performance Architecture

# Reading!

McCabe’s Book. - Chapter 8

Please **read the book** along with this presentation. **Chapter 4 onwards are important chapters that cover the core of the topic for this class**

# Network Performance

- Performance is a measure of capacity, delay, and RMA in the network.
- Network performance is defined through a Performance Architecture

# Performance Architecture

A network performance architecture describes how the performance requirements of the users, applications, devices are met within the network.

- Is the set of performance mechanisms to configure, operate, manage, and account for resources in the network
- describes where these mechanisms are applied within the network
- describes the sets of internal relationships within these mechanisms
- describes the sets of external relationships between this mechanisms and other network component architectures such as Network Management and security.

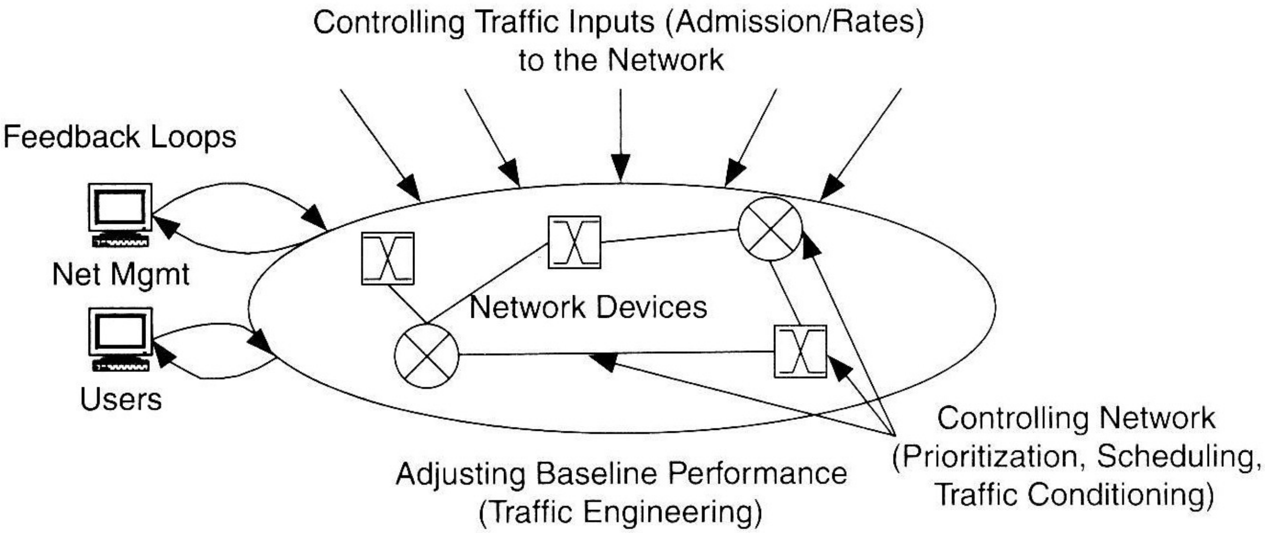
# Why Performance Architecture

- Do we need one?
  - Think of many multimedia applications with varied performance needs and supported by Internet
- What we are trying to solve?
  - For improving the overall performance of a network for a single-tier performance networks
  - Improving the performance for selected groups such as users, applications, and/or devices in multi-tier performance network
  - Merging multiple traffic types (voice, data, video) over a common network infrastructure, example such as for migrating voice onto the data network
  - Differentiating (and possibly charging) customers for multiple levels of service.
- These considerations and those identified during requirements analysis process set goals for the performance architecture

# Network Performance Architecture Process

- The first step of developing a performance architecture is defining the performance goals and goals for individual mechanisms.
- The process of developing goals begins during the requirements analysis process and is refined during the architecture process.
- Important inputs to the performance architecture process are:
  - flow specifications
  - flow maps

# Performance Architecture Mechanisms


Controlling Traffic Inputs (Admission/Rates) to the Network
Feedback Loops
Net Mgmt
Users
Network Devices
Controlling Network (Prioritization, Scheduling, Traffic Conditioning)
Adjusting Baseline Performance (Traffic Engineering)

Performance architecture mechanisms are:

•Service-level agreements (SLAs)

•Policies (Customer driven)

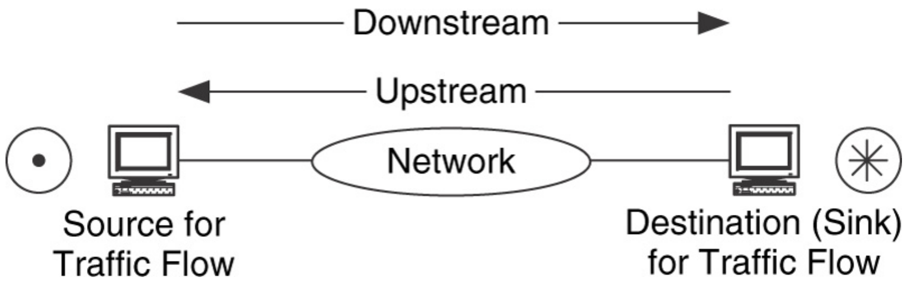
•QoS Mechanisms

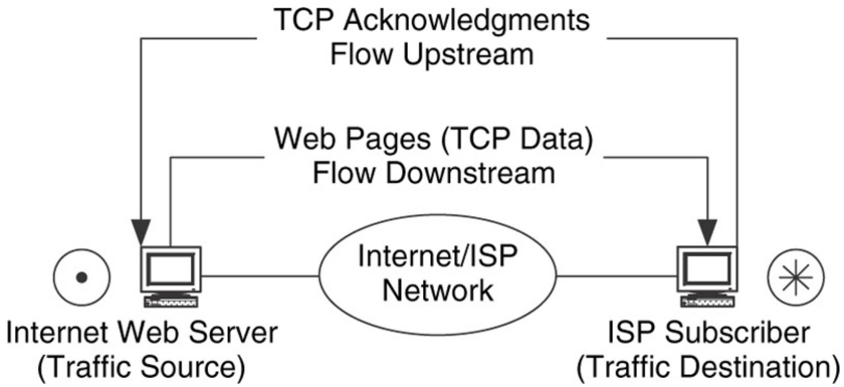
- Traffic Engineering
- Admission control of traffic
- Resource control (prioritization, traffic management, scheduling, and queuing
- Two primary mechanisms: IntServ and DiffServ

Subsets of performance mechanisms are usually used to performance requirements.

# Service-Level Agreement (SLA)

- SLAs are formal contracts between a provider and a user that define the terms of provider’s responsibility to the user and extent of accountability if those responsibilities are not met.
- SLAs can also be found in enterprise networks, e.g., between network administrator and network users
- The service provider has the responsibility of monitoring and managing service to ensure that the users are getting what they expect.
- SLA could be for the whole network or for each stream, e.g., upstream, downstream, or individual TCP streams.
- They can include delays, RMA, etc.



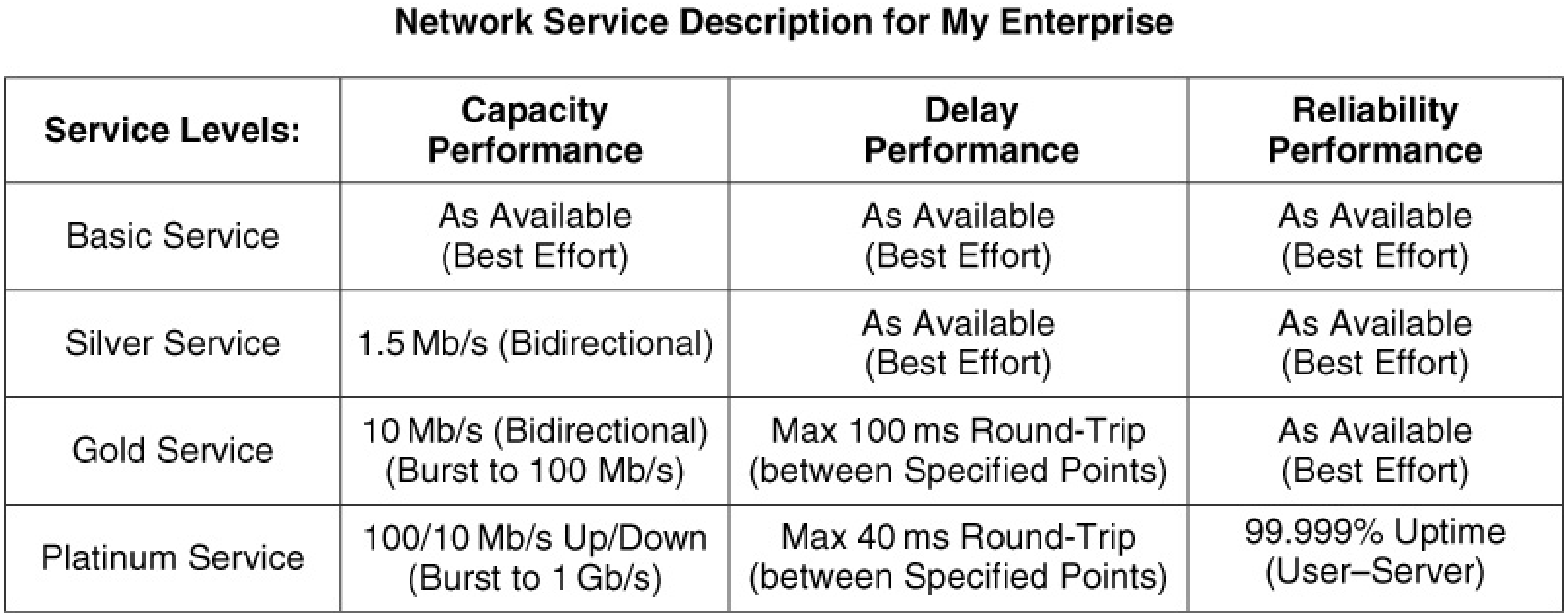

TCP Acknowledgments Flow Upstream
Web Pages (TCP Data) Flow Downstream
Internet/ISP Network
Internet Web Server (Traffic Source)
ISP Subscriber (Traffic Destination)

# SLAs

Examples of SLA Performance Elements

- Data rate (minimum, peak)
- Delay
- Burst Tolerance (Size, duration)
- Upstream/downstream
- RMA Metrics

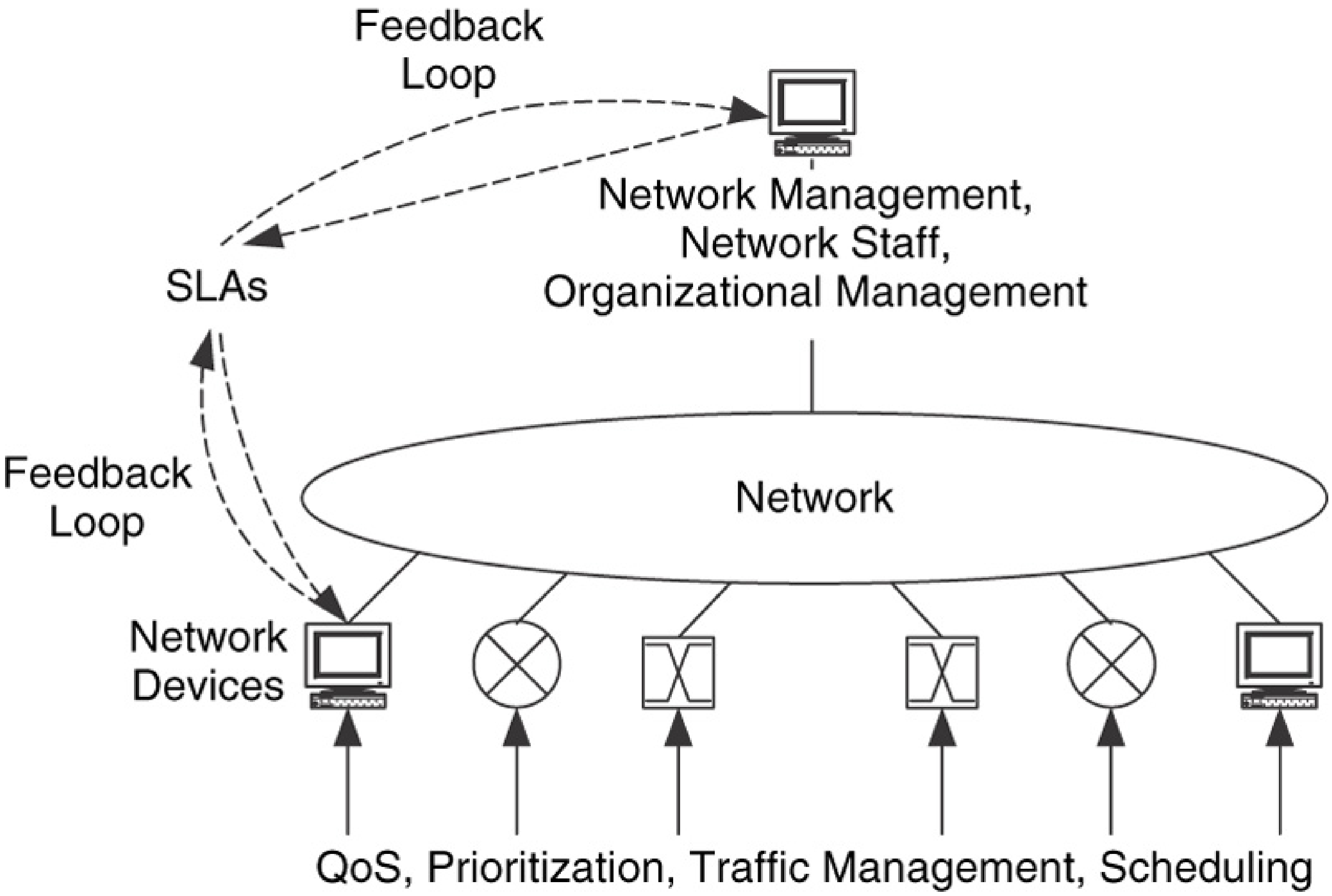
## Illustrative SLA categories



Network Service Description for My Enterprise

| Service Levels: | Capacity Performance | Delay Performance | Reliability Performance |
|---|---|---|---|
| Basic Service | As Available (Best Effort) | As Available (Best Effort) | As Available (Best Effort) |
| Silver Service | 1.5 Mb/s (Bidirectional) | As Available (Best Effort) | As Available (Best Effort) |
| Gold Service | 10 Mb/s (Bidirectional) (Burst to 100 Mb/s) | Max 100 ms Round-Trip (between Specified Points) | As Available (Best Effort) |
| Platinum Service | 100/10 Mb/s Up/Down (Burst to 1 Gb/s) | Max 40 ms Round-Trip (between Specified Points) | 99.999% Uptime (User–Server) |

# Performance Mechanisms with SLAs Added



# Policies

- Policies are developed during analysis phase and contract SLAs
- Policies are formal or informal sets of high- level statements and rules about how network resources (and, therefore, performance are to be allocated among users.
- Policies reside (stored) in a policy database kept in the network.
- Policy information is managed and implemented though this database
- Policies information is passed between databases and network devices using protocols, like:
    - ➢ Common Open Policy Services (COPS) (rfc4261)
    - ➢ Lightweight directory access protocol (LDAP) (rfc4511)

# Quality of Service (QoS)

QoS is determining, setting, and acting on priority levels for traffic flows.

Example mechanisms in different networks:

- IP QoS
  - DiffServ
  - Intserv
- QoS in Multiple Label Switching (MPLS) – through Label Switching Path (LSP)
- ATM Classes of Service (CoS) – Constant Bit Rate (CBR), rt- and nrt- (real time and non real time) Variable Bit Rate (VBR), Unspecified Bit Rate (UBR), and Available Bit Rate (ABR)
- Frame relay Committed Information rate (CIR), Peak Information Rate (PIR), Extended information rate (EIR)

# QoS for IP-based Traffic

As noted, there are two standard types of QoS for IP-based traffic:

- Differentiated services (DiffServ): approaches QoS from the perspective of aggregating traffic flows on a per-hop basis.
  - Type of Service (TOS) and DiffServ Code Points (DSCP) in IPv4; Traffic Class in IPv6
- Integrated services (IntServ): approaches QoS from the perspective of supporting traffic flows on an individual end-to-end basis.
  - RSVP protocol for reserving resources.

# Integrated Services

Used when a traffic flow needs to be tracked individually (e.g. flows with guaranteed requirements).

*IntServ* defines values and mechanisms for allocating resources to flows across the end-to-end path of the flow.

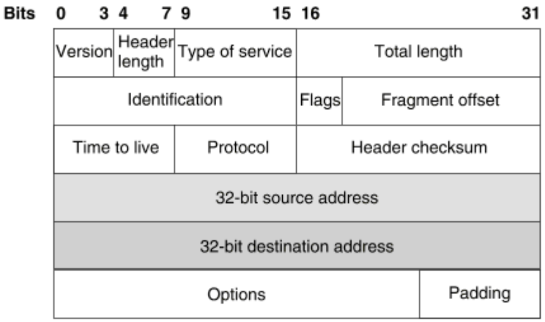
*IntServ* Requirements:

- Support on the network devices across which the flow travels, requiring resource (e.g., memory, processing, bandwidth) knowledge for each flow.
- A mechanism to communicate flow requirements, as well as the setup and teardown of resource allocations, across network devices in the end-to-end path of a flow.
- This is provided by the Resource Reservation Protocol (RSVP). Beside various messages in RSVP, Path Messages identifies the path in the forward direction and Resv Messages makes the resource reservation starting in the opposite direction.

IntServ requirements make it hard to scale in large networks.

It is best used for specific needs, like Traffic Engineering and in networks with guaranteed services/flows.

# Differentiated Services-DiffServ



- Set of routers for a Administrative Domain where a set of service classes are defines.

- IP packets in DiffServ (DS) are marked in *Type of Service* (ToS) 6 bits of the 8 bits in IPv4, and in traffic Class byte in IPv6.

- The 6 bits, the *DiffServ Code Points* (DSCP) is for indicating *Per Hop Behaviors* (PHB), and the upper 2 bits in IPv4 not used and in IPV6 are for *Explicit Congestion Notification* (ECN). see RFC 2474, e.g.

- DiffServ allows network controllers to define their own traffic classes. Most networks however use the following commonly defined Per Hop Behaviors (PHBs):

➢*Expedited forwarding* (EF): for traffic requiring low loss, low latency, ensured bandwidth (e.g. for voice and multimedia traffic sensitive to delays and jitter)

➢*Assured forwarding* (AF): delivers traffic with high assurance as long as the traffic does not exceed the profile. Some packets may get dropped. (can tolerate some delay, drop)

➢*Default:* typically for *Best effort* traffic

# Traffic Management or Traffic Conditioning

Traffic management is to enable specified performance for different types traffic. It consists of:

*Admission control:*

It is the ability to accept or refuse access to network resources, i.e., making network resources available based on traffic priority.

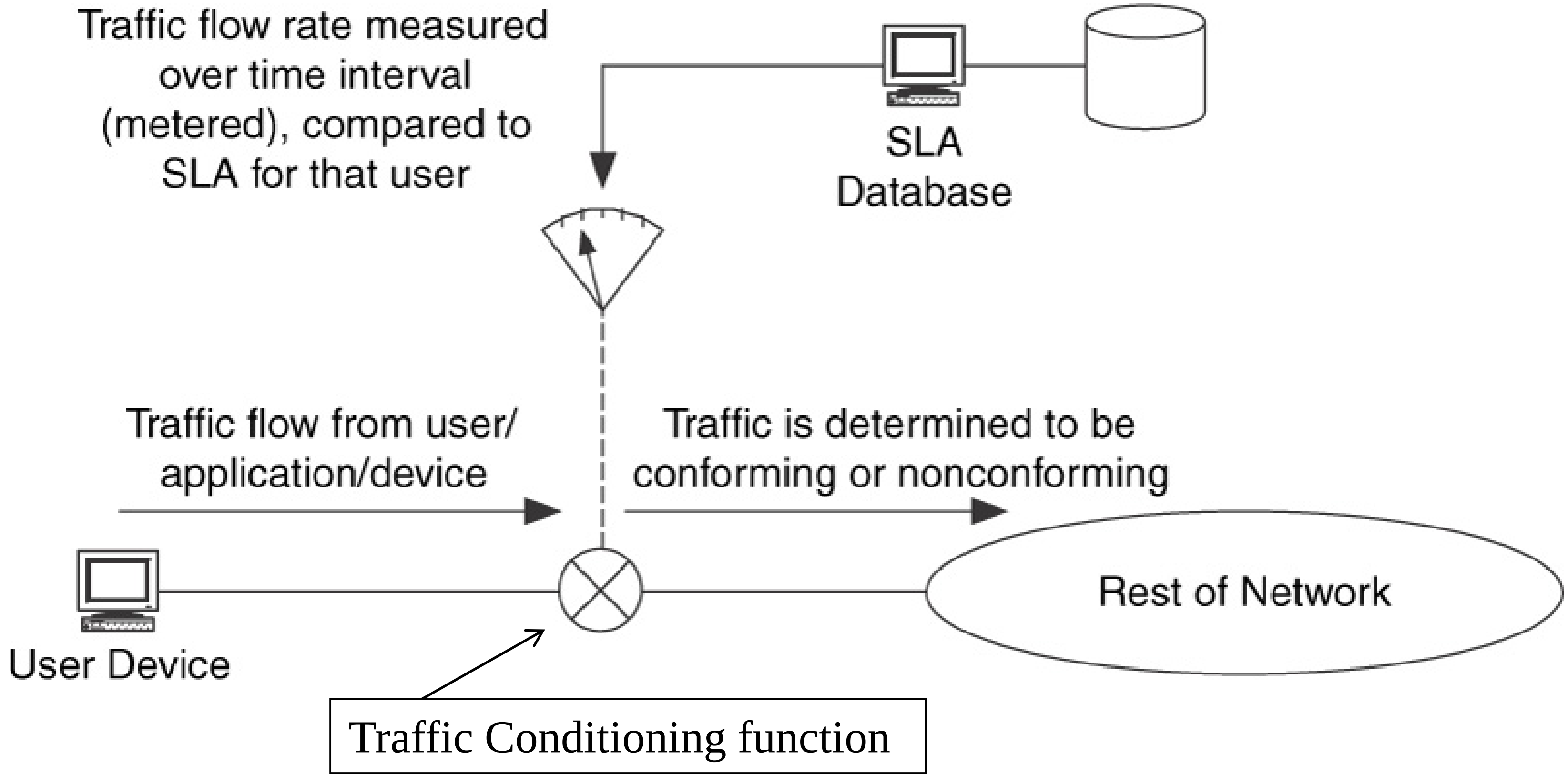
*Traffic conditioning implements DiffServ at a node and* consists of*:*

- *Classification* (ability to identify traffic flow),
- *Marking* :It may look at various parts of the packets, source, destination, port number, protocol identifier (e.g., SIP identifier for VOIP traffic, and tag them as in DiffServ by DiffServ Point Codes (DSPCs) for Expedited Forwarding, etc.
- *Metering*: measuring their performance levels, such as temporal characteristics, traffic rate and burst size and comparing with expected thresholds. These are often implemented at network devices such as switches and routers as a part of their performance implementation.
- *Shaping* (delaying non-conforming traffic, e.g., through leaky bucket algorithm)
- *Dropping* (discarding non-conforming traffic packets during congestion).

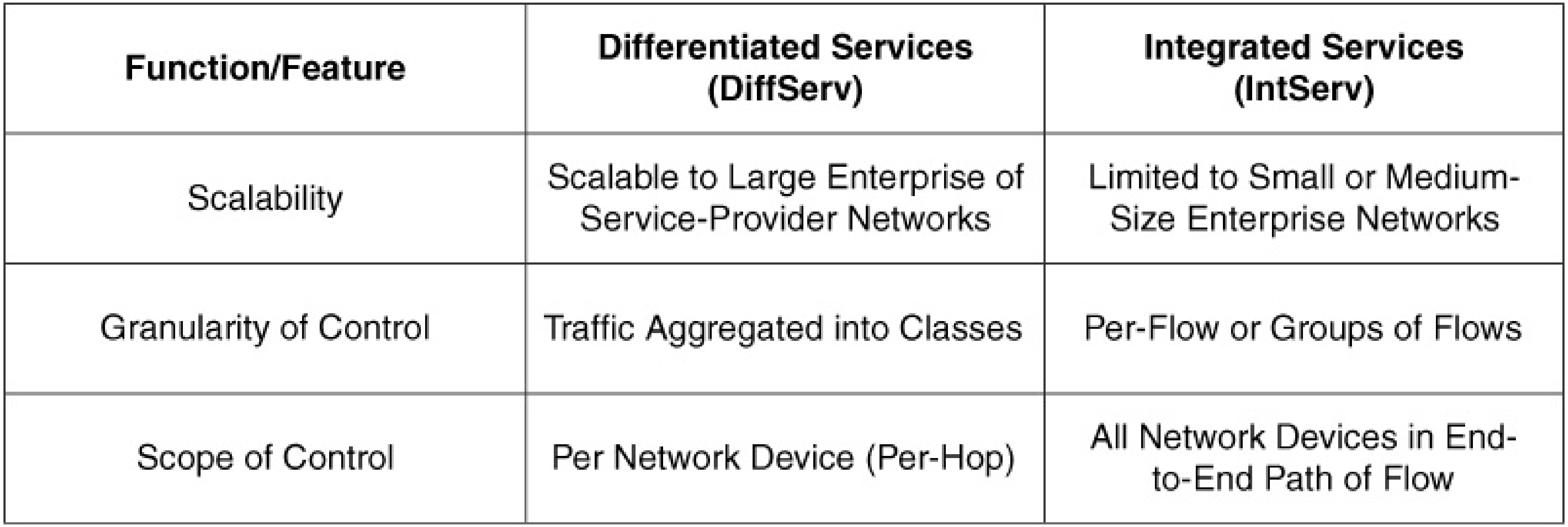
All these functions are used extensively at network devices in IP networks.

# Traffic Management and Conditioning

## Metering at a Switch Illustration


Traffic flow rate measured over time interval (metered), compared to SLA for that user
SLA Database
Traffic flow from user/ application/device
Traffic is determined to be conforming or nonconforming
User Device
Rest of Network
Traffic Conditioning function

# Comparison of DiffServ and IntServ

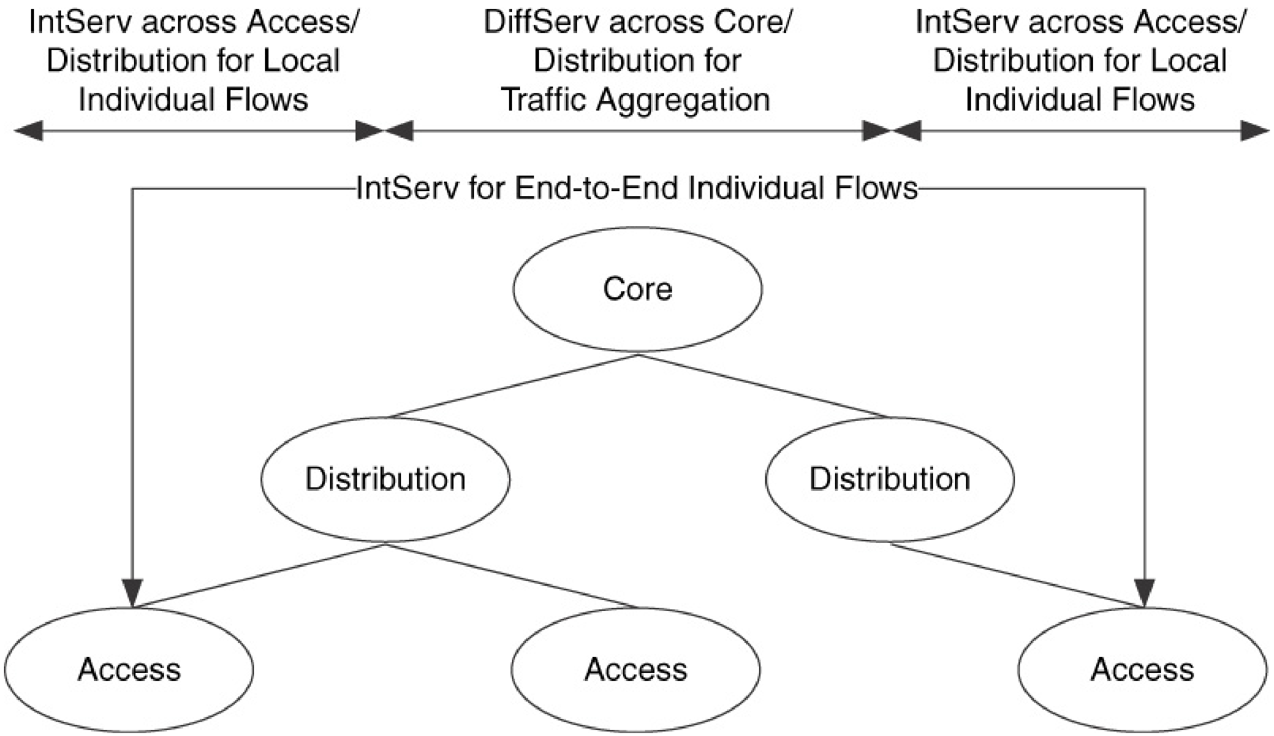


| Function/Feature | Differentiated Services (DiffServ) | Integrated Services (IntServ) |
|---|---|---|
| Scalability | Scalable to Large Enterprise of Service-Provider Networks | Limited to Small or Medium-Size Enterprise Networks |
| Granularity of Control | Traffic Aggregated into Classes | Per-Flow or Groups of Flows |
| Scope of Control | Per Network Device (Per-Hop) | All Network Devices in End-to-End Path of Flow |

# DiffServ and IntServ

DiffServ and IntServ can be applied

- Individually or together within a network
- Individually to different parts of the network


IntServ across Access/ Distribution for Local Individual Flows
DiffServ across Core/ Distribution for Traffic Aggregation
IntServ across Access/ Distribution for Local Individual Flows
IntServ for End-to-End Individual Flows
Core
Distribution
Distribution
Access
Access
Access

DiffServ/IntServ application on an access/distribution/core architecture

# QoS Mechanisms
## Prioritization, Traffic Management, Scheduling and Queuing

- A performance architecture may include one or more of these mechanisms in conjunction with SLAs and policies to provides desired performance.
- These mechanisms are implemented in network elements such as routers and switches, but could also be at edges to the network.
- These mechanisms along with appropriate sizing for capacity and management tools ensures desired QoS for individual services supported by the network.

# Prioritization

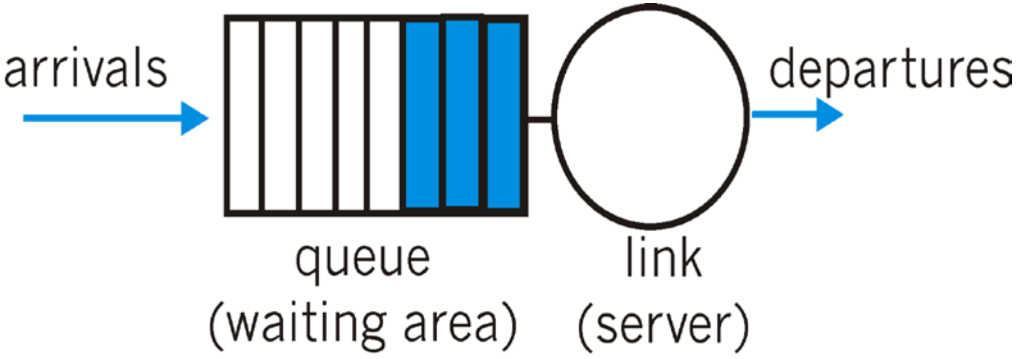
- Is the process of determining which users, applications, devices, flows, and connections get service ahead of others or get a higher level of service.
- Begins during the requirements and flow analysis.
- Whenever there are multiple levels of performance requirements in a network, there will be a need to prioritize these traffic flows (e.g., multi-tier performance requirements).
- Priority Levels could be based on:
  - The type of protocol (e.g. TCP Vs. UDP)
  - Port number
  - IP or MAC-layer address
  - DiffServ Service Code Point
  - Or any other information embedded within the traffic.
- Priority levels are used by network devices for admission control, scheduling of traffic flows, and conditioning of flows through the network.

# Flow Control Mechanisms
# Scheduling/Queuing Mechanisms

- Once traffic has been prioritized, it is forwarded to one or more output queues in the network devices for transmission onto the network.
- Scheduling/queuing is applied at network devices throughout a network.
- Scheduling/queuing determines the order (by priority level) in which traffic is processed for transmission. It is storing packets at a network device while they wait for processing. A number of queuing mechanisms are available in network devices. Some commonly used are:
- First In First Out (FIFO)
- Class-based queuing (CBQ)
- Weighted fair queuing (WFQ)
- Random early detect (RED)
- Weighted RED (WRED)

# Scheduling And Queuing Mechanisms

- Scheduling: choose next packet to send on link
- FIFO (first in first out) scheduling: send in order of arrival to queue
  - What are real-world example?
  - discard policy: if packet arrives to full queue: who to discard?
    - Tail drop: drop arriving packet
    - priority: drop/remove on priority basis
    - random: drop/remove randomly



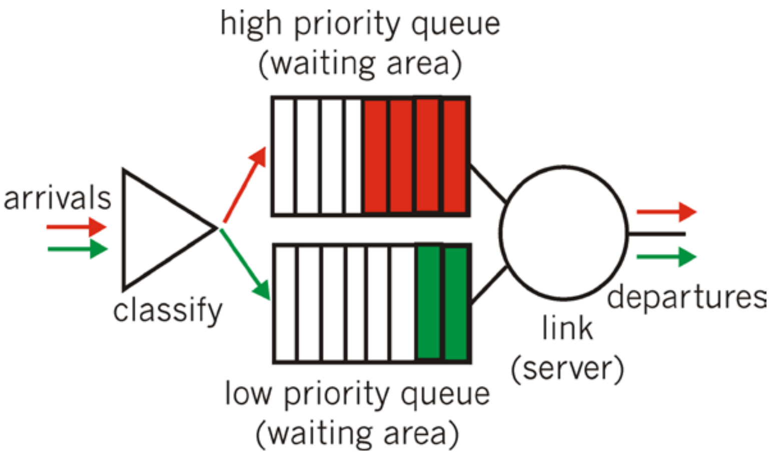
Source: Kurose and Ross

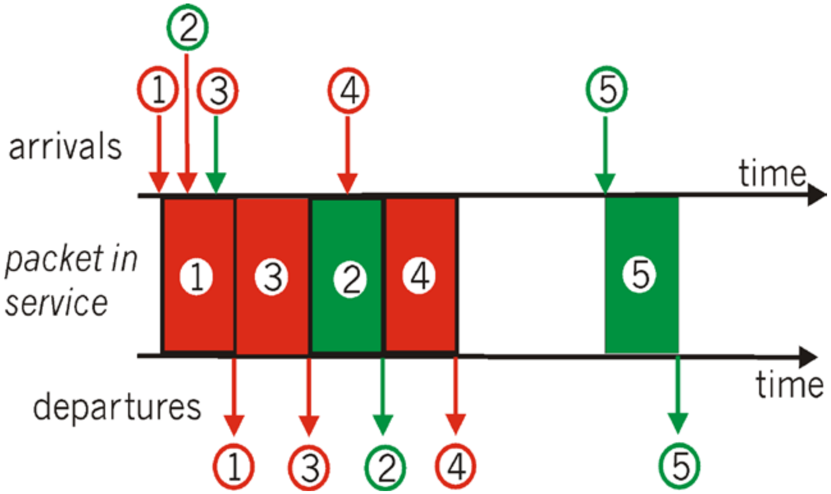
# Scheduling And Queuing Mechanisms

## Priority Class based Scheduling

: transmit highest priority queued packet

- multiple *classes*, with different priorities
  - class may depend on marking or other header info, e.g. IP source/dest, port numbers, etc..



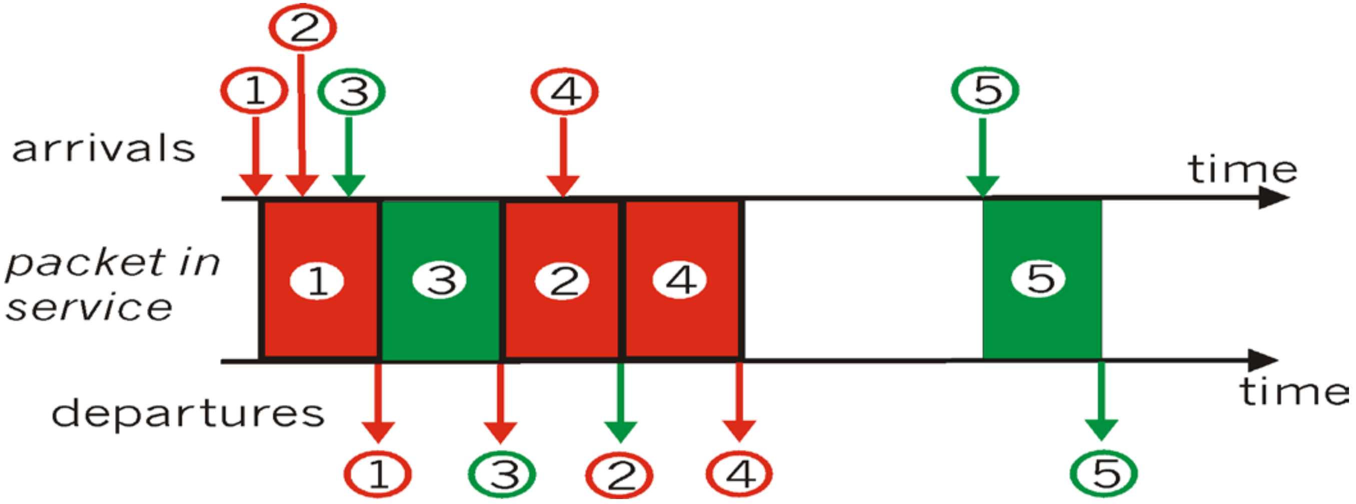


Source: Kurose and Ross

# Scheduling And Queuing Mechanisms

## Round Robin Scheduling

- multiple classes
- cyclically scan class queues, serving one from each class (if available)

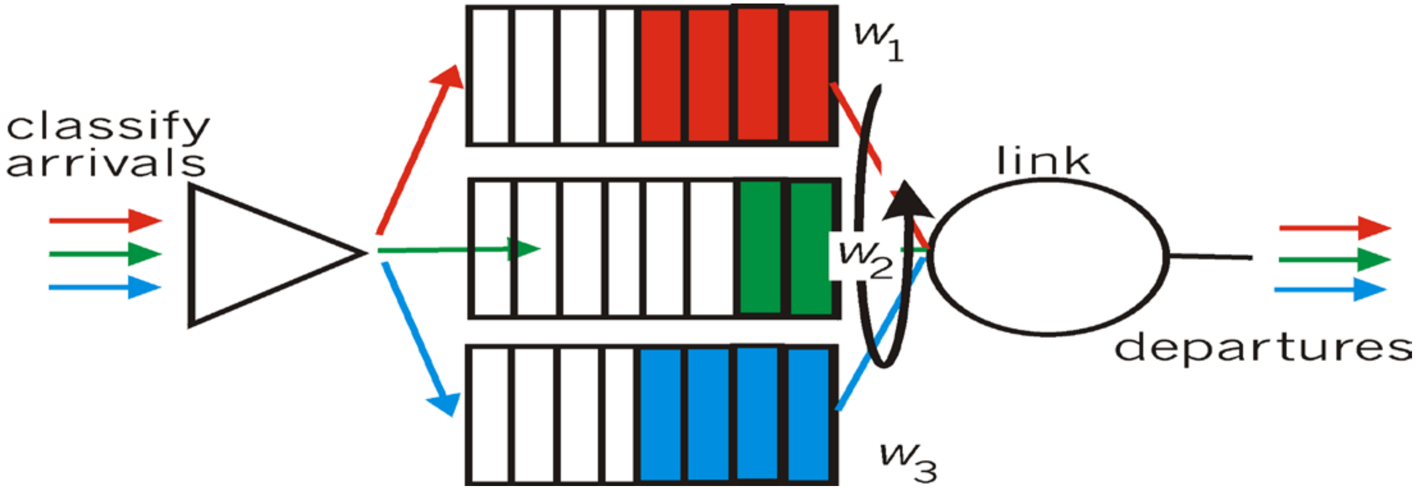


Source: Kurose and Ross

# Scheduling And Queuing Mechanisms

## Weighted Fair Queuing (WFQ)

- generalized Round Robin
- each class gets weighted amount of service in each cycle



Source: Kurose and Ross

# Policing Mechanisms

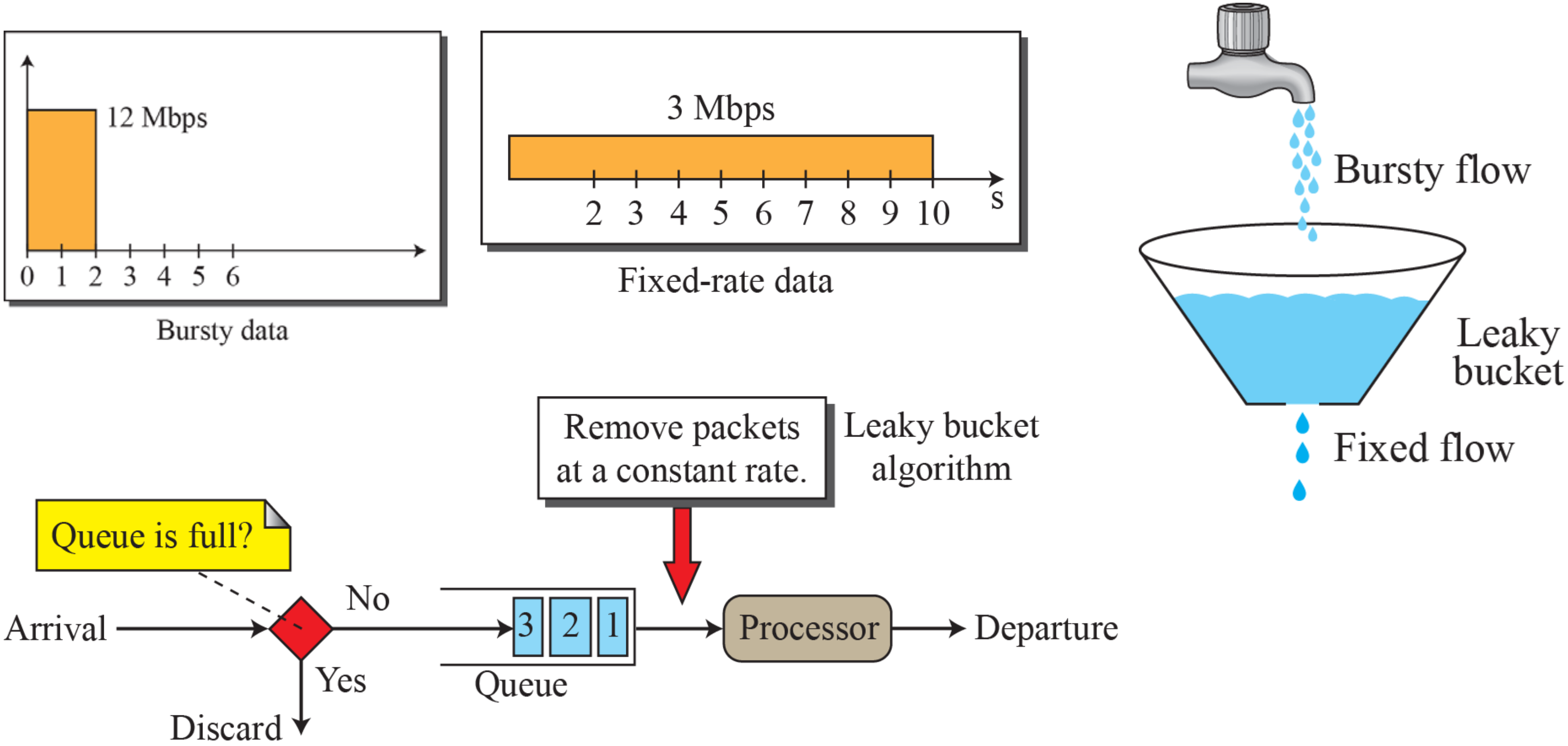
Goal: limit traffic to not exceed declared parameters

Three common-used criteria:

- *(Long term) Average Rate:* how many pkts can be sent per unit time (in the long run)
  - crucial question: what is the interval length: 100 packets per sec or 6000 packets per min have same average!
- *Peak Rate:* e.g., 6000 pkts per min. (ppm) avg.; 1500 ppm peak rate
- *(Max.) Burst Size:* max. number of pkts sent consecutively (with no intervening idle).

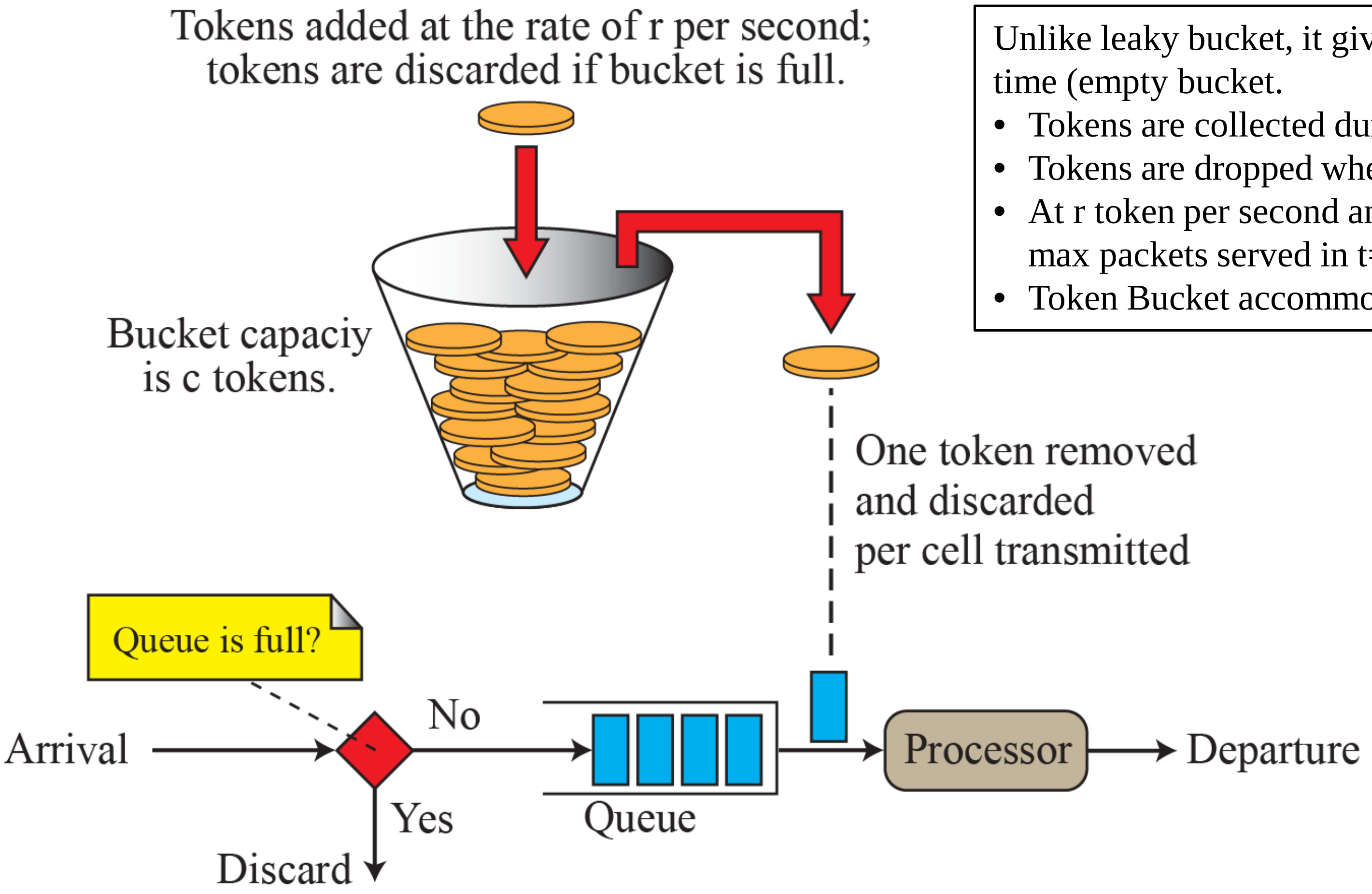
# Policing Mechanisms - Leaky bucket

Itsmooths the traffic but does not credit idle host



Source: Forouzan and Mosharraf”

# Policing Mechanisms - Token bucket



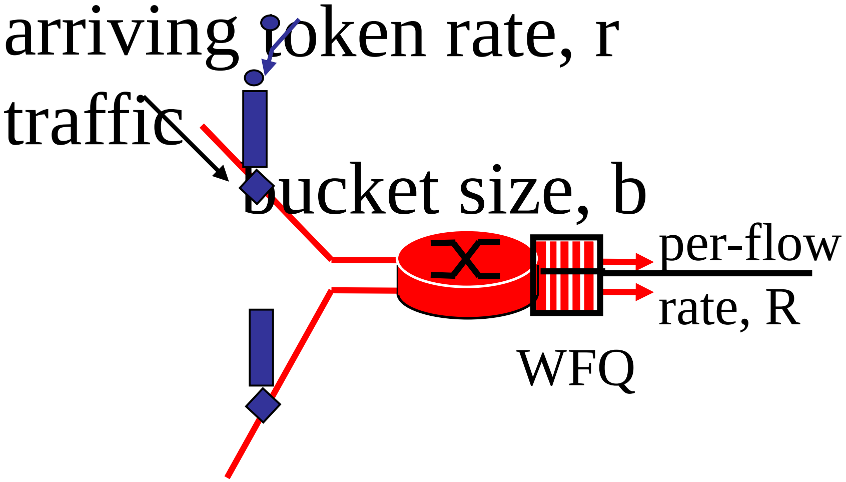
Unlike leaky bucket, it gives credit for idle time (empty bucket.

- Tokens are collected during idle time
- Tokens are dropped when bucket is full
- At r token per second and bucket size c, max packets served in t=r.t+c
- Token Bucket accommodates burstiness.

Source: Forouzan and Mosharraf”

# Policing Mechanisms

- token bucket, WFQ combine to provide guaranteed upper bound on delay, i.e., *QoS guarantee*!



arriving traffic

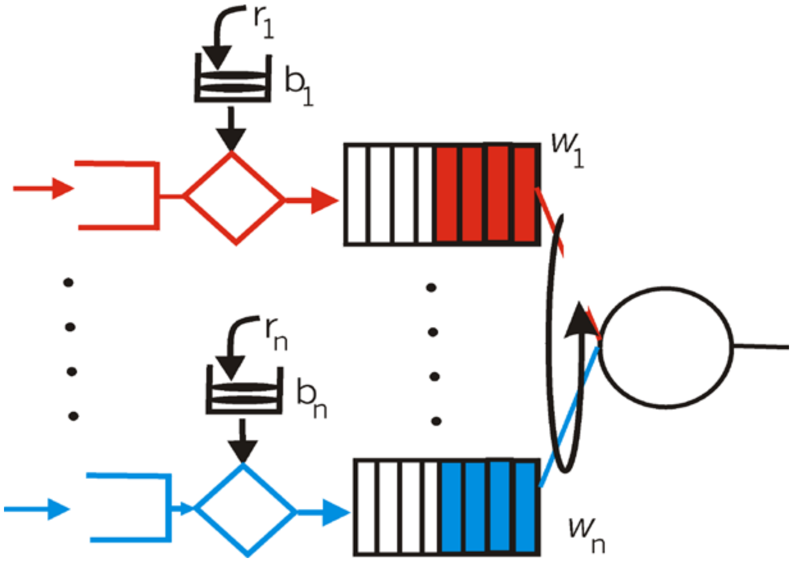
token rate, r

bucket size, b

per-flow rate, R

WFQ

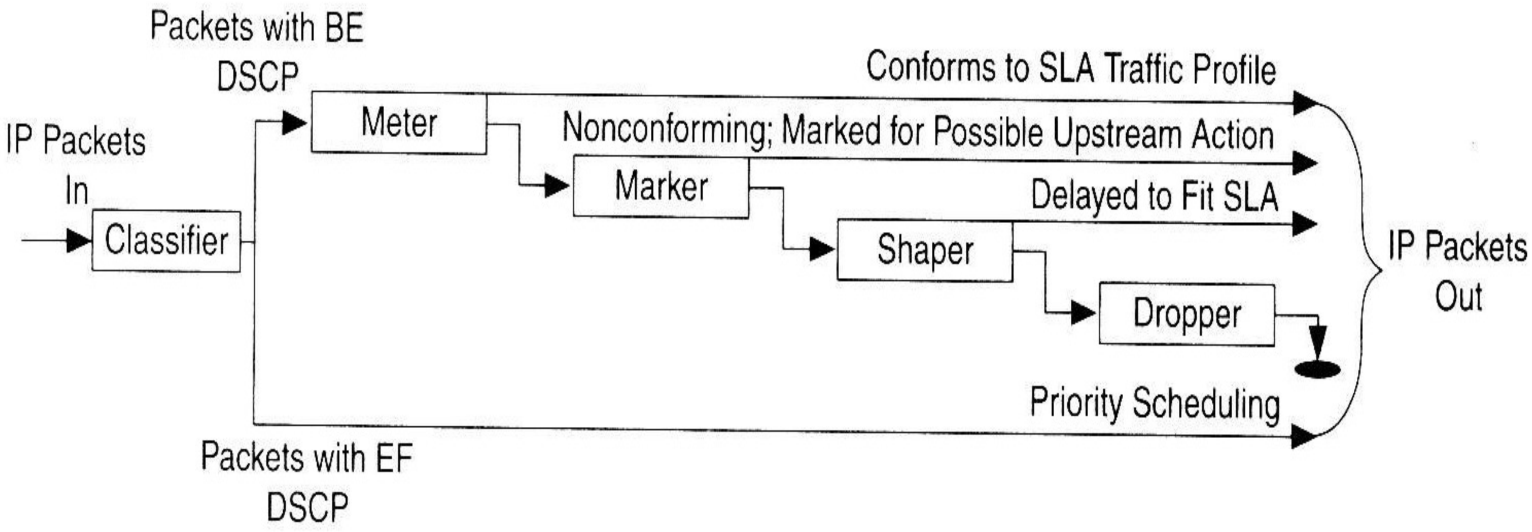
$D_{max} = b/R$



R= Transmission rate of the link

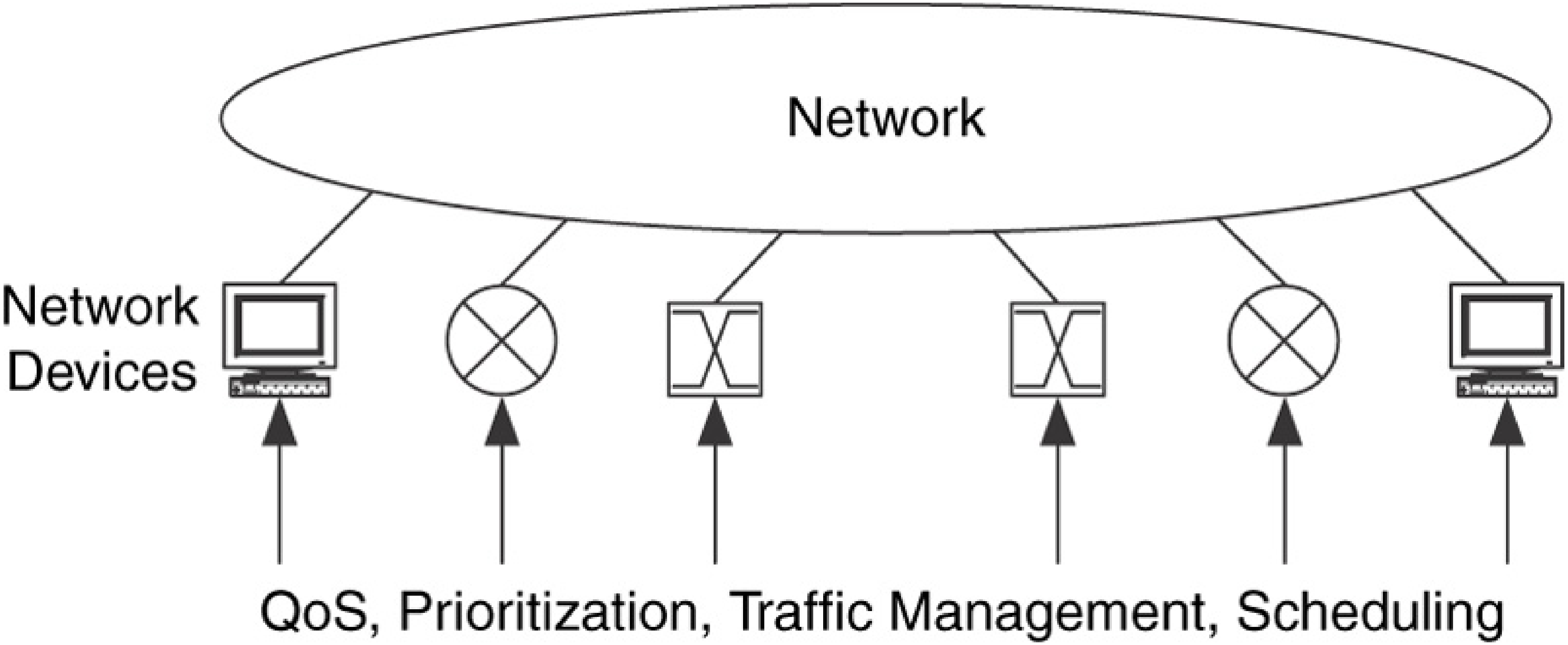
# Traffic Conditioning Functions in an DiffServ Enabled Network Element

At a DiffServ enabled router, traffic is classified (e.g., DSPC), metered for possible non-conformance, shaped to conform if necessary through delaying or by dropping packets.

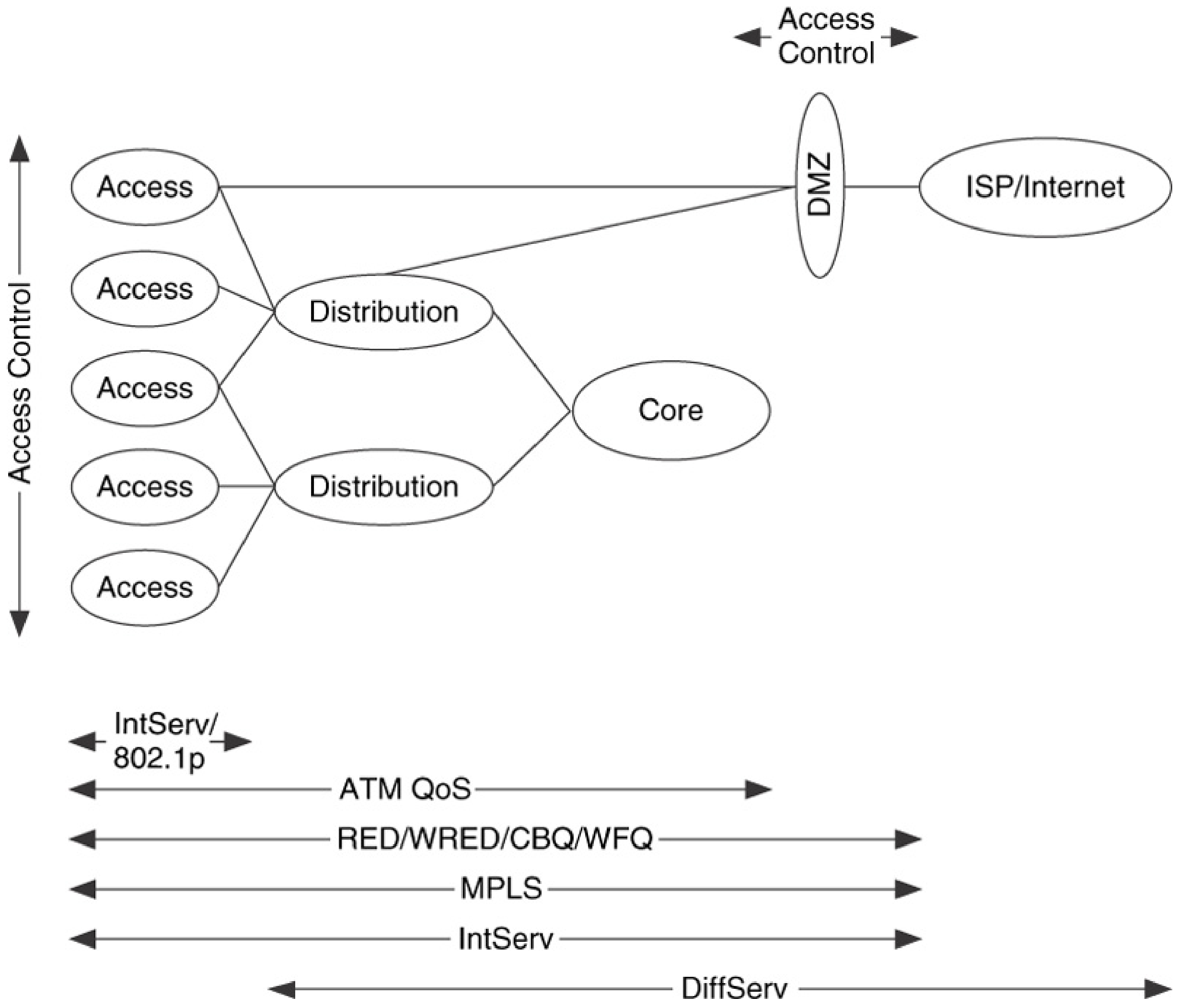


- Classifier-classifies
- Meter- clocks against negotiated profile
- Marker-Marks up or down as needed
- Shaper- Reshapes if not compliant

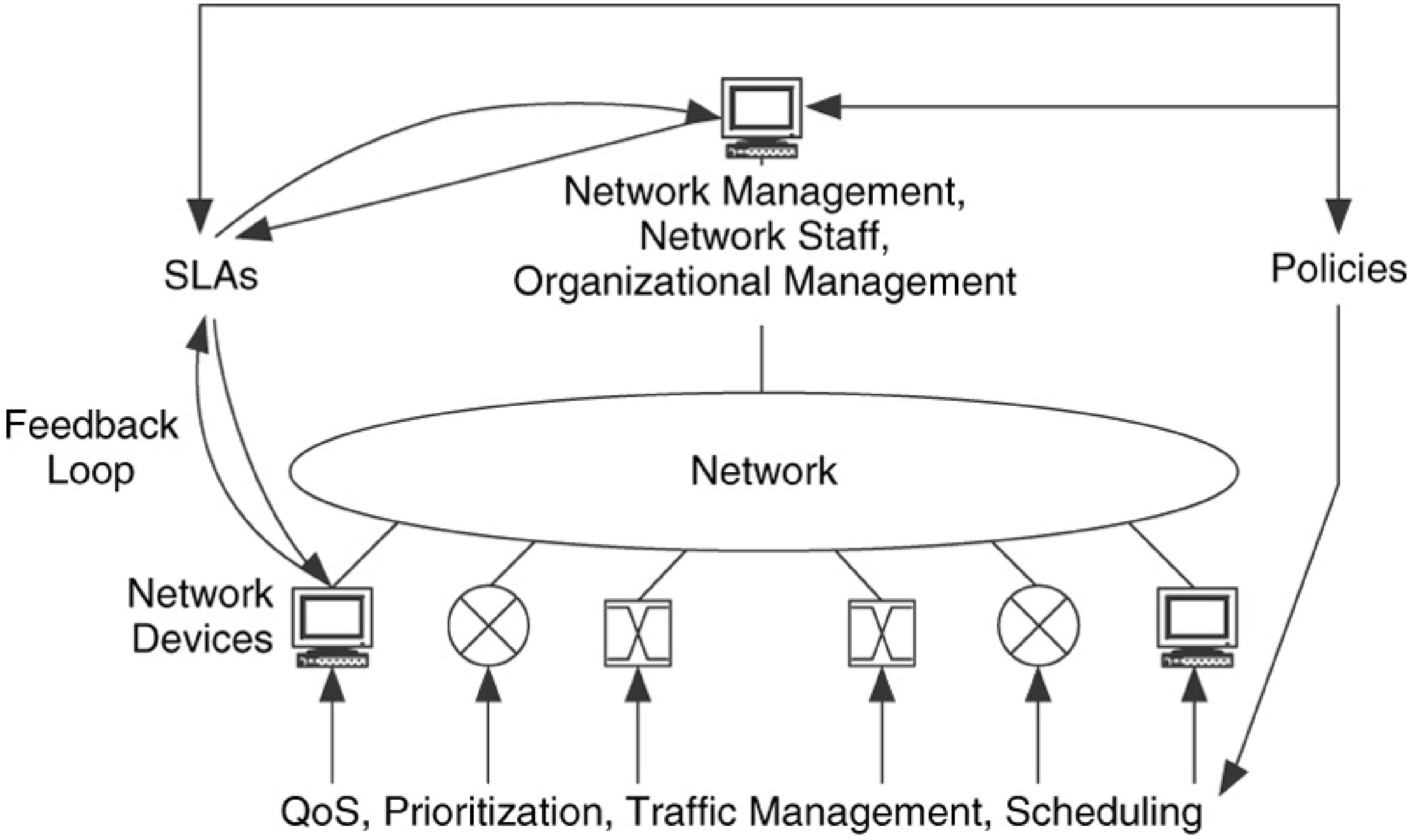
# Performance mechanisms Applied at Various Network Devices



# Performance Mechanisms Regions



# Performance Mechanisms with SLAs and Policies Added



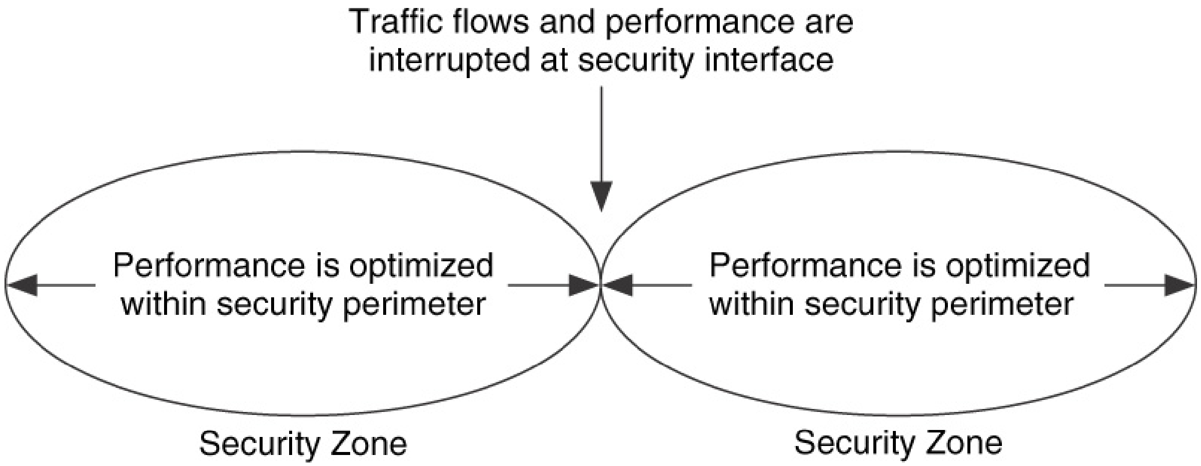
# Internal and External Relationships for Performance Architecture

Internal relationships may include trade-offs between:

- end-to-end and per-hop
- Scheduling and conditions, among others

External relationships may include interactions:

- Performance and addressing
- Performance and Network Management
- Performance and Security
- Service architecture such as CDN


Traffic flows and performance are interrupted at security interface
Performance is optimized within security perimeter
Performance is optimized within security perimeter
Security Zone
Security Zone