PERFORMANCE MEASUREMENTS WITH CISCO DEVICES

SESSION NMS-2042

Emmanuel TYCHON
Technical Leader
NMTG – Device Instrumentation

Based on a presentation developed by David MELTON.

Recuerde siempre:

- Apagar su teléfono móvil/pager, o usar el modo “silencioso”.

- Completar la evaluación de esta sesión y entregarla a los asistentes de sala.

- Ser puntual para asistir a todas las actividades de entrenamiento, almuerzos y eventos sociales para un desarrollo óptimo de la agenda.

- Completar la evaluación general incluida en su mochila y entregarla el miércoles 8 de Junio en los mostradores de registración. Al entregarla recibirá un regalo recordatorio del evento.

Objectives

- **This is about**
  - **The Process of Performance Management**
  - **Outline features inside Cisco Devices and the value they bring to performance management**
  - **Case studies for performance measures to be applied to a Service, Network and device**
  - **Overview management applications that support Cisco Device Instrumentation**
- **This is NOT about**
  - **In-depth description of the Cisco IOS Technologies used for performance measurements**

Agenda

- Process of Performance Management
- Technologies
- Performance Measurements—Case Studies
  - Availability
  - Monitoring Service (VoIP Quality)
  - Monitoring QoS
  - Element Utilization
- Different Scenarios

PROCESS OF PERFORMANCE MANAGEMENT

Reasons for Performance Management

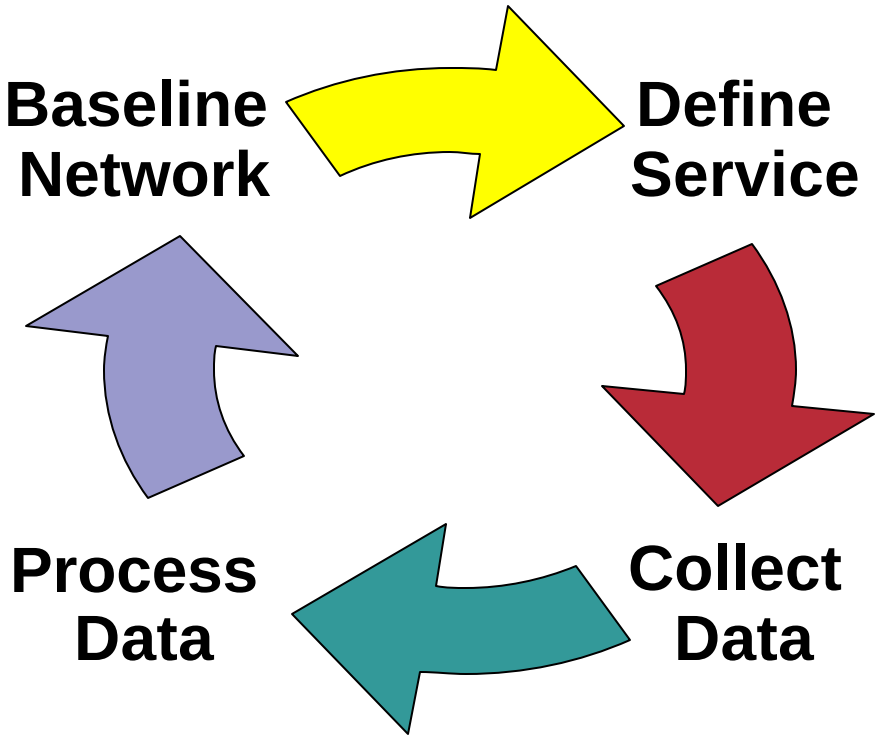
Good Performance Management Ensures:

- ✓ **Delivery of Customer SLAs**

  Users can be assured that applications and services will be there when they need them

- ✓ **Increased Network and Service Reliability**
- ✓ **A better understanding and control of the network**
- ✓ **Deployment of new application and services with complete confidence**

Performance Management Process



Measurement examples:

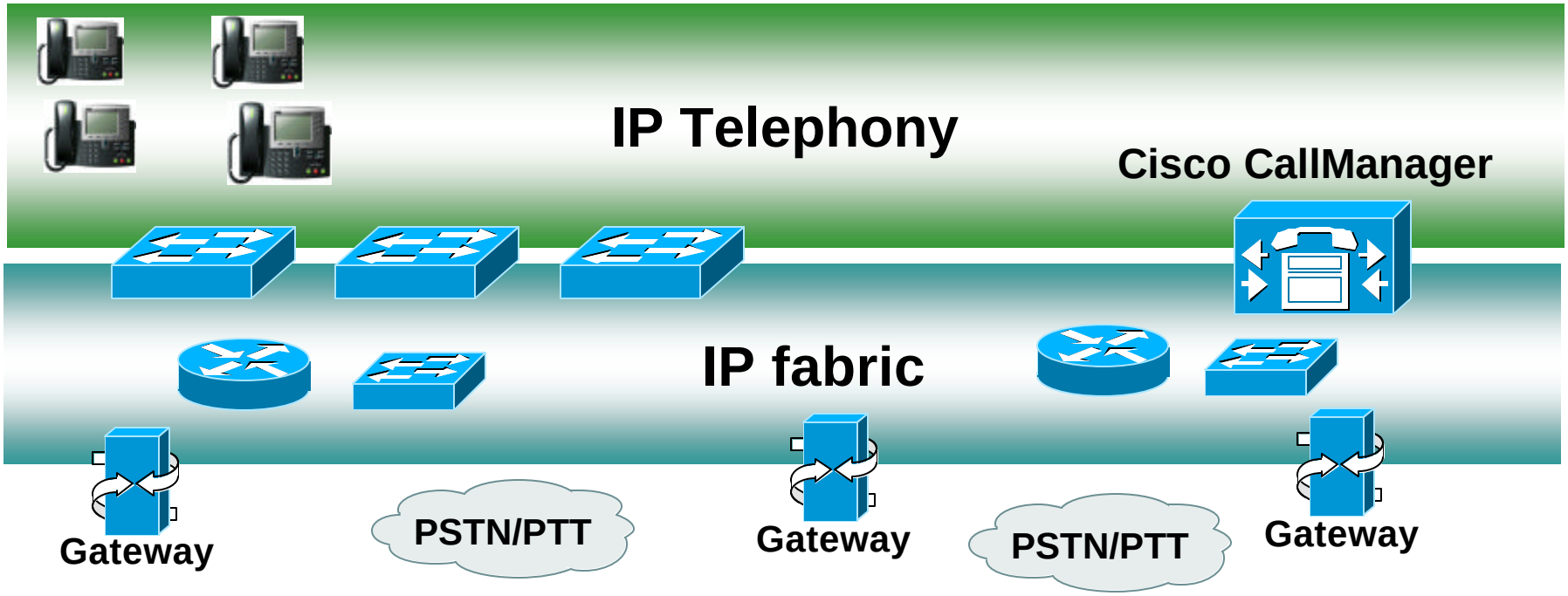
Service (Customer SLA) + Underlying Data network

Voice Quality, Component utilization, response times,…

Performance Monitoring: Start at Service Level

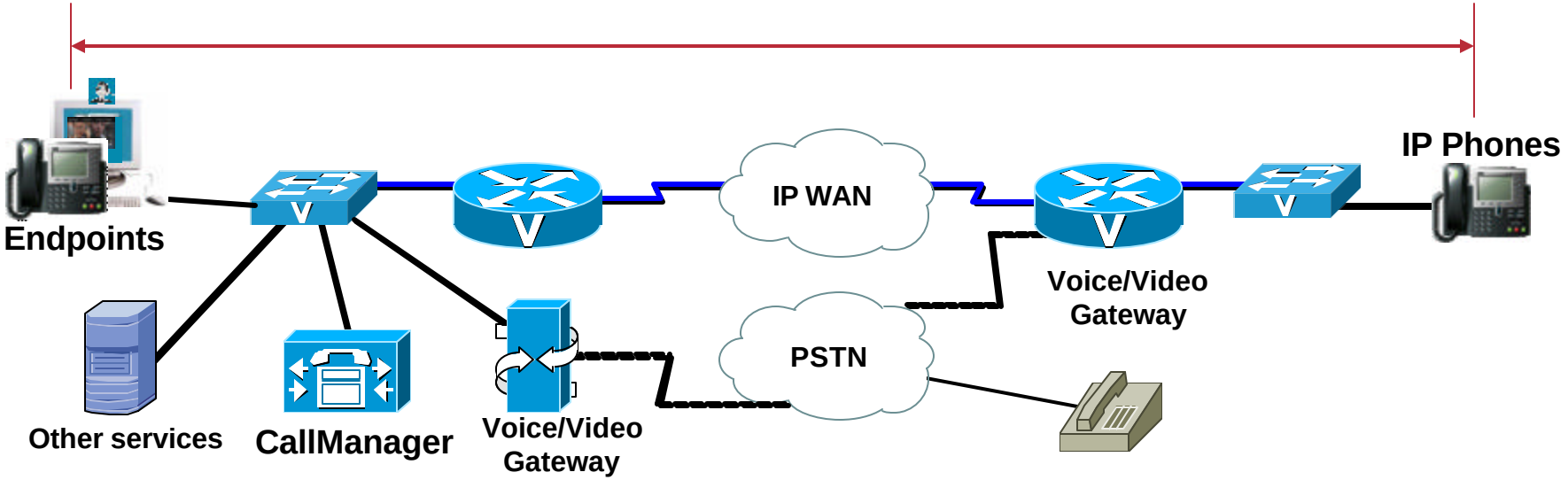
- **IP Telephony is a service carried over in IP infrastructure**

  We need to apply performance monitoring to both

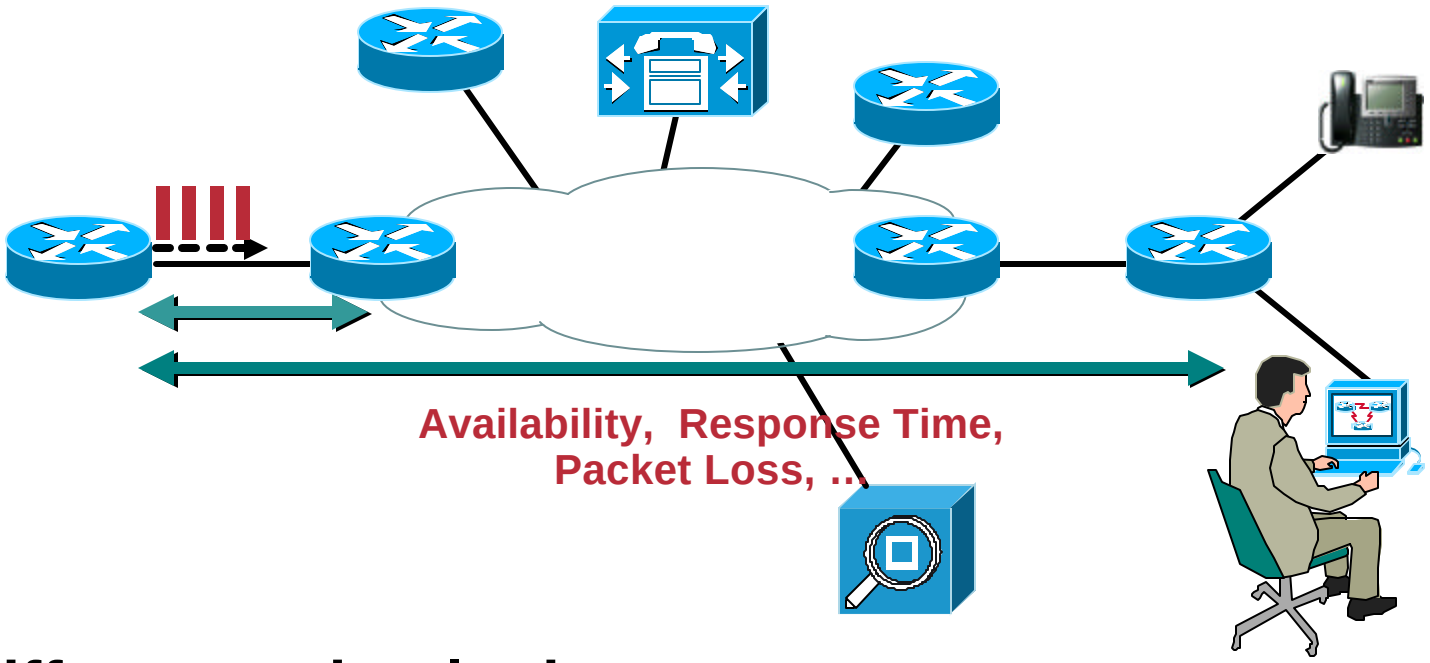


Monitoring the Service: The Starting Point

- **Monitoring service provides customer view of network performance**
  - **If Service measurements are outside of SLAs**
    - **We have performance issues somewhere in the network**
    - **We can trace along the path of service to identify performance issues**
- **Monitoring at the device level**
  - **May identify performance issues**
  - **Can't tell if these impact on customer service (SLAs)**



Cisco Performance Measurement Technologies



- **Different technologies**

  SNMP Polling, EEM, IP SLAs, RMON, ART MIB, NBAR, QoS, Corvil, and NetFlow

- **With different focus**

  Service Wide, Network Wide, Device Wide

TECHNOLOGIES

Performance Measurement Technologies

Cisco.com

SNMP MIBs and Embedded Event Management

MEASURES: CPU/Memory Utilization, Availability, QoS

Sampling:	Passive
Collection:	Embedded
Scope:	Device/Link
Perspective:	User/Network

NBAR/CORVIL

MEASURES: Response Time of Live Application Traffic to Server Device, QoS

Sampling:	Passive
Collection:	External Probe/Embedded
Scope:	Link/End-to-End
Perspective:	User/Network

Cisco IPSLA

MEASURES: Latency and Jitter Between Source Router and Specified Target

Sampling:	Active
Collection:	Embedded
Scope:	Link/End-to-End
Perspective:	User/Network

NetFlow

MEASURES: Device Interface Traffic Rate by S/D IP Address, Port Number or AS

Sampling:	Passive
Collection:	Embedded
Scope:	Link/End-to-End
Perspective:	Network

Call Manager

MEASURES: Voice Calls, Voice Quality, CallManager Performance

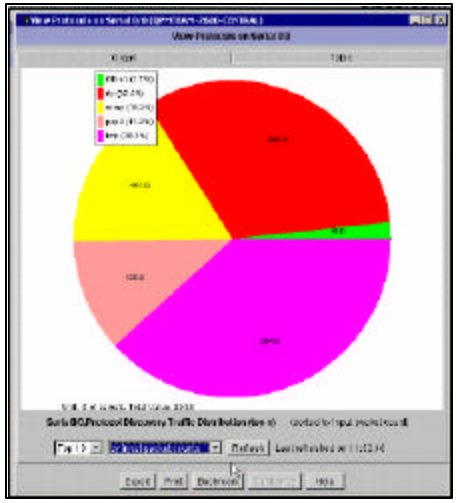
Sampling:	Passive
Collection:	Embedded
Scope:	Link/End-to-End
Perspective:	User/Network

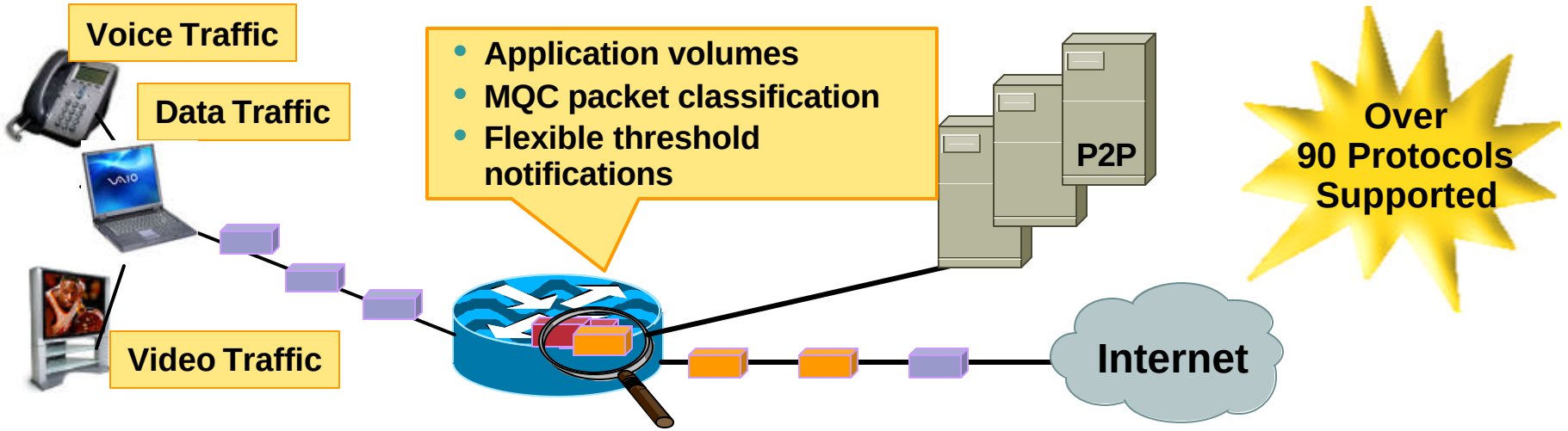
Measurement Technology: SNMP

- **Cisco-Voice-Dial-Control-MIB**
  - **RoundTripDelay, Planning Impairment Factor (ICPIF)**
- **Cisco-RTTMON-MIB**
  - **Collects latest IP SLA measurements and provides on device aggregation of IP SLA measurements.**
- **Cisco-Class-Based-QoS-MIB (CiscoCBQosMIB)**
  - **Primary Accounting Mechanism for QoS**
- **IF-MIB**
  - **Collects bandwidth or link utilisation and packet loss**
- **CISCO-PROCESS-MIB and ENTITY-MIB**
  - **Collect statistics on the CPU utilisation**
- **CISCO-MEMPOOL-MIB, CISCO-ENHANCED-MEMPOOL-MIB**
  - **Collect statistics on the memory utilization**

Measurement Technology:
Network-Based Application Recognition-NBAR

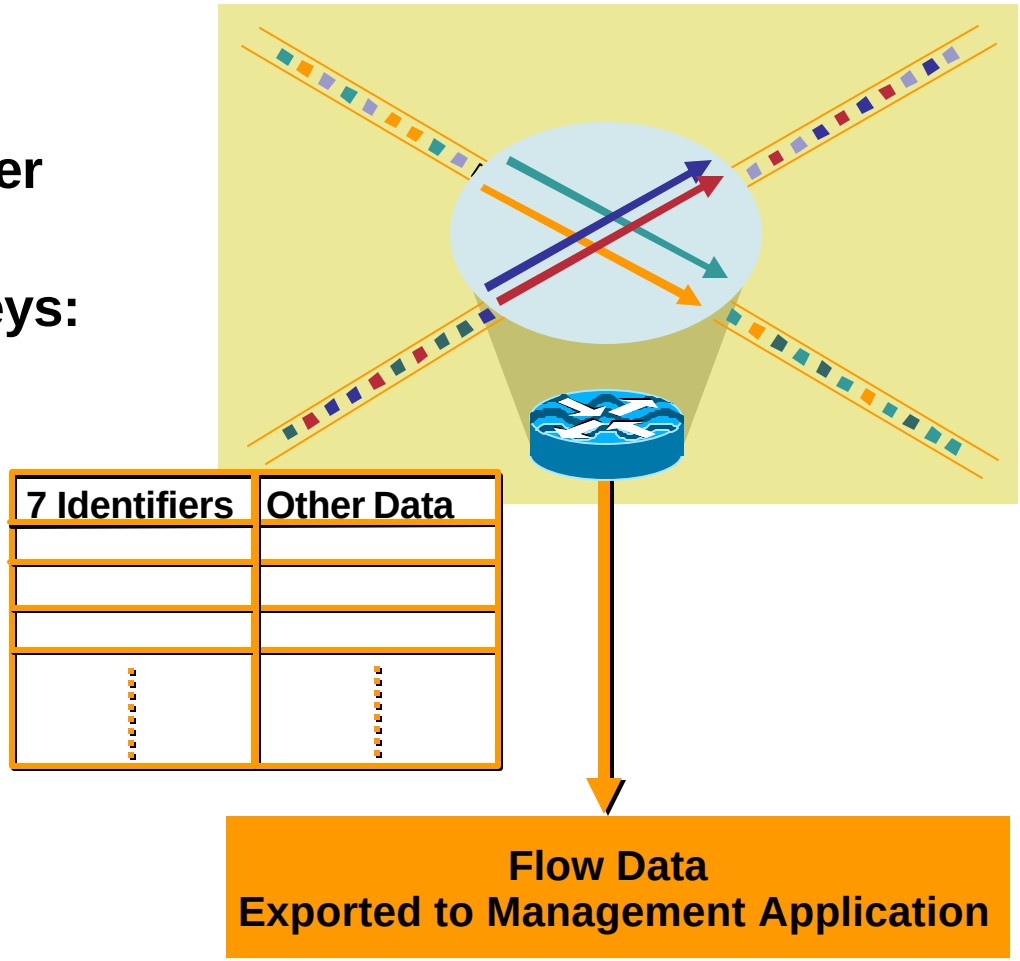
- Examines data from Layers 3 through 7
- Uses Layers 3 and 4 plus packet inspection for classification
- Stateful inspection of dynamic-port traffic
- Packet Description Language Modules (PDLMs) define applications recognizable by NBAR
- Customers can specify their own match criteria to identify TCP- or UDP-based applications



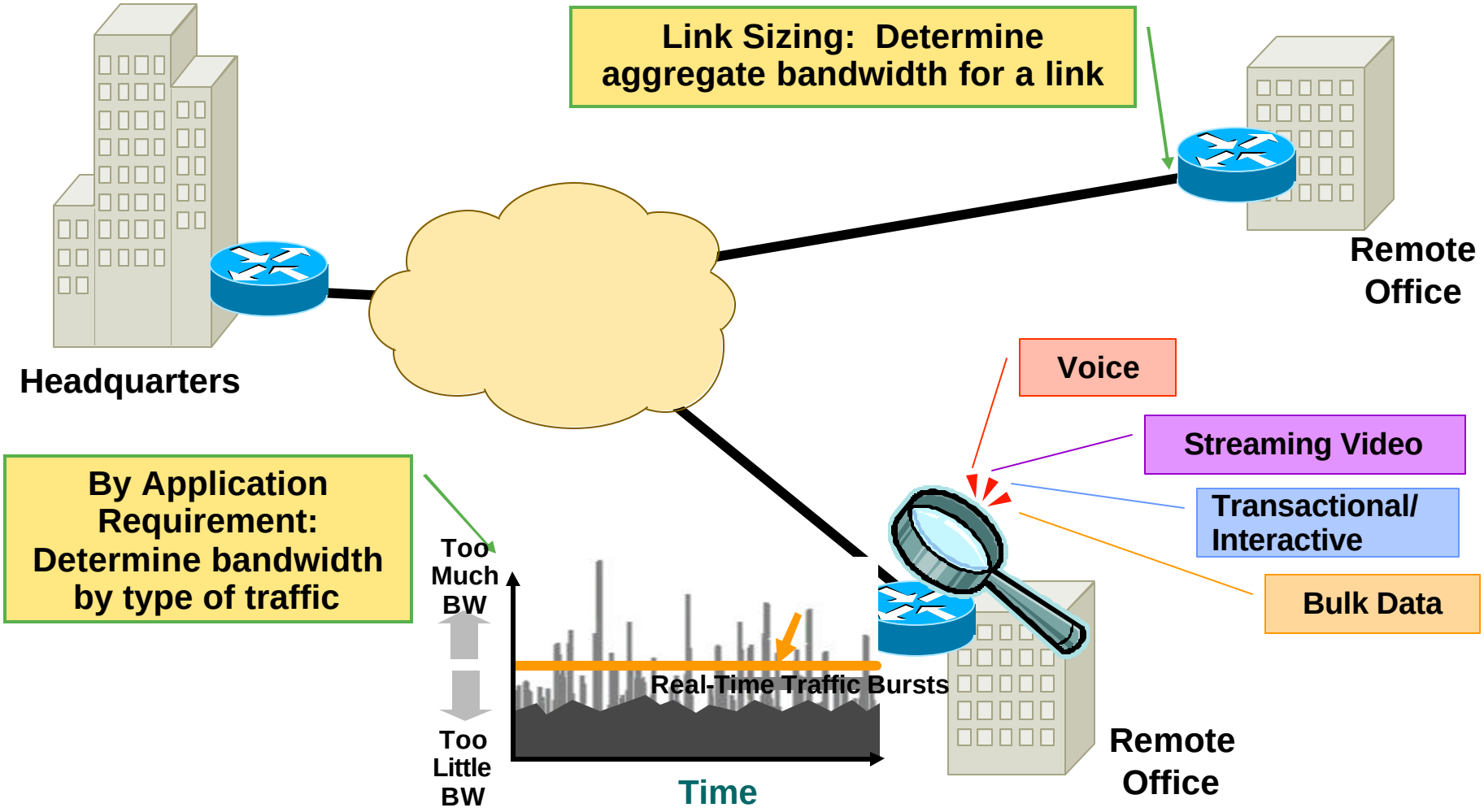


Measurement Technology: NetFlow

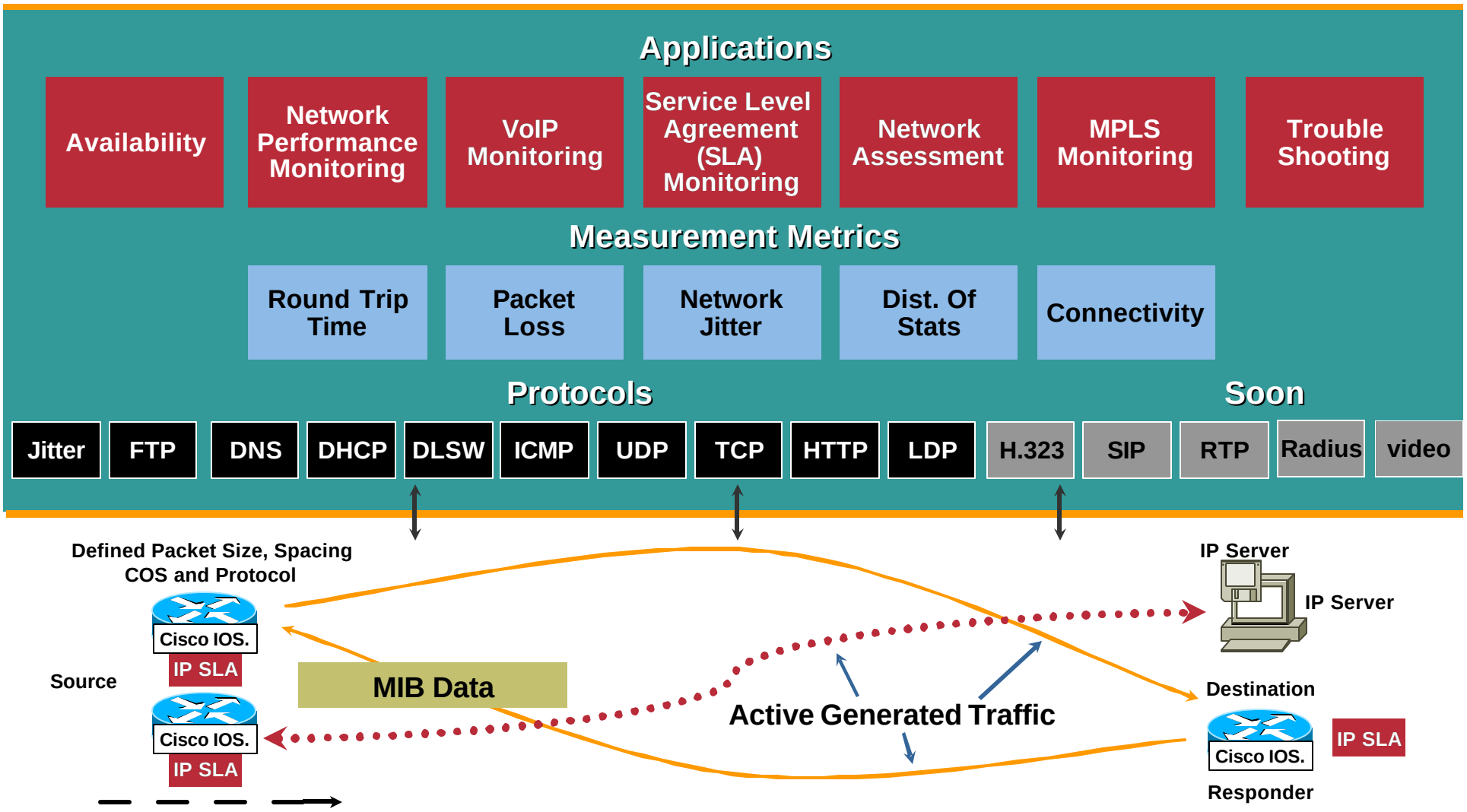
- Flows are unidirectional
- Flows are enabled on a per input-interface basis
- Flows are defined by 7 keys:
  - Source IP address
  - Destination IP address
  - Source port
  - Destination port
  - Layer 3 protocol
  - TOS byte (DSCP)
  - Input interface (ifIndex)



Measurement Technology: Corvil Bandwidth

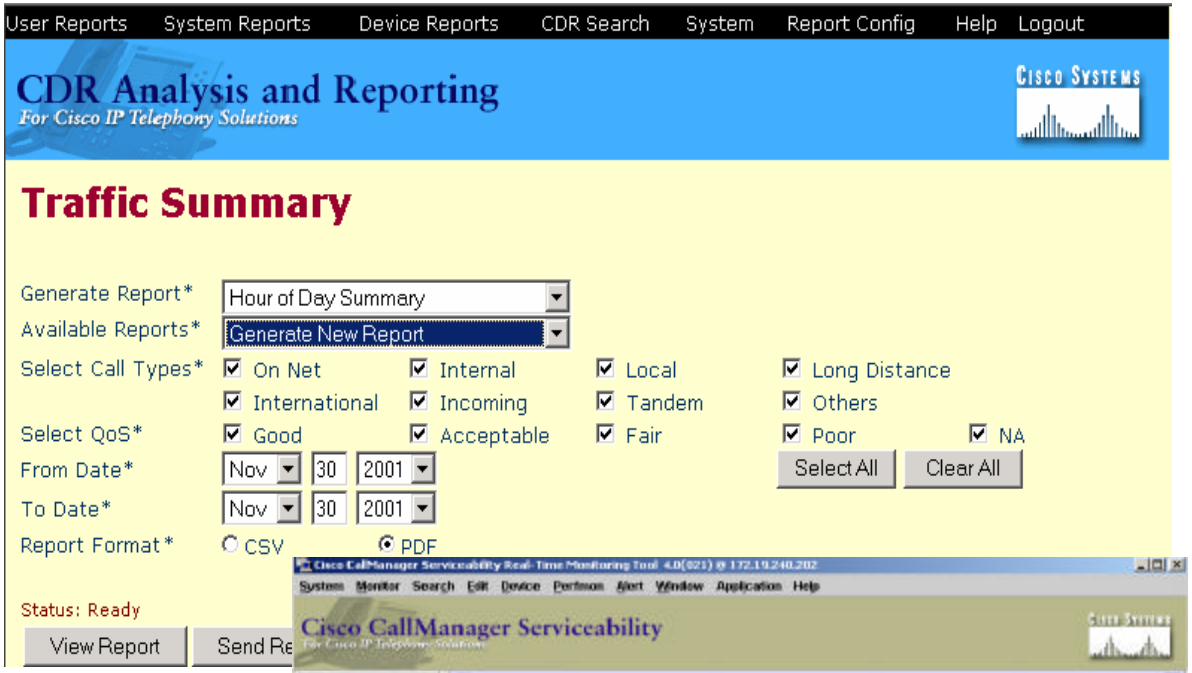


Measurement Technology: IP SLAs

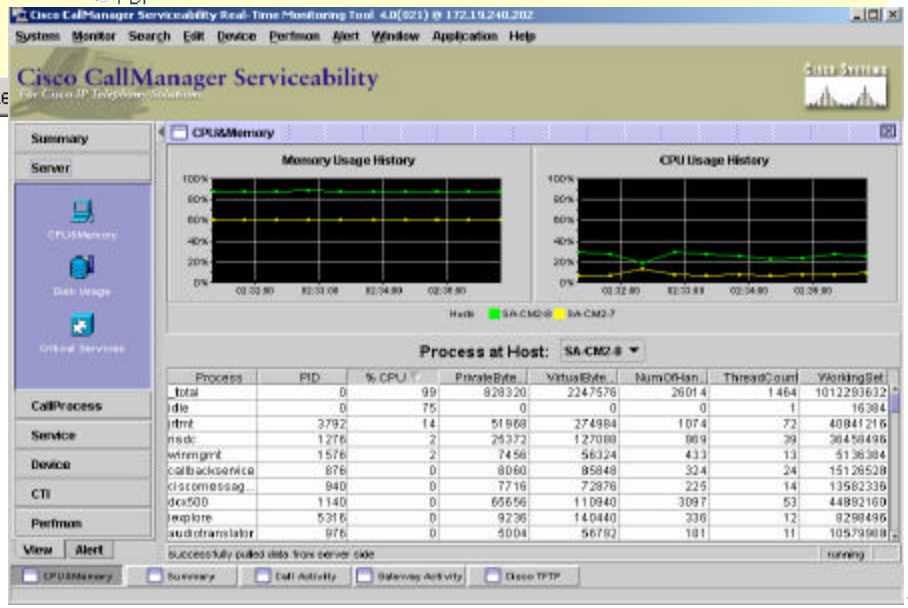


Measurement Technology: CallManager Serviceability

- CDR Analysis & Reporting (CAR)
- All Call Management Records (CMRs) have
  - ✓ packets/bytes sent
  - ✓ Packets lost
  - ✓ Jitter



- Real Time Monitoring Tool (RTMT)
  - ✓ Critical system performance CPU, Memory, Disk



PERFORMANCE MEASUREMENTS: CASE STUDIES

Performance Measurement Case Studies

Case Study/ Technology	COOL	CLI	SNMP	IPSLA	Call Manager	Corvil	NBAR	NetFlow
1 Availability	X	X	X					
2 Connectivity		X	X	X				
3 VoIP		X	X	X	X			
4 Services		X	X	X		X	X	X
5 QoS		X	X	X		X	X	X
6 Embedded Monitoring		X	X					

CASE STUDY 1:
Engineered Availability

Case Study 1: [1] Engineered Availability

- Engineered availability based on network design, component MTBF and MTTR
- MTBF = Mean Time between Failure

  Calculated by measuring the average time between failures on a device/component

  Vendors should provide MTBF for their network components
- MTTR = Mean Time To Repair

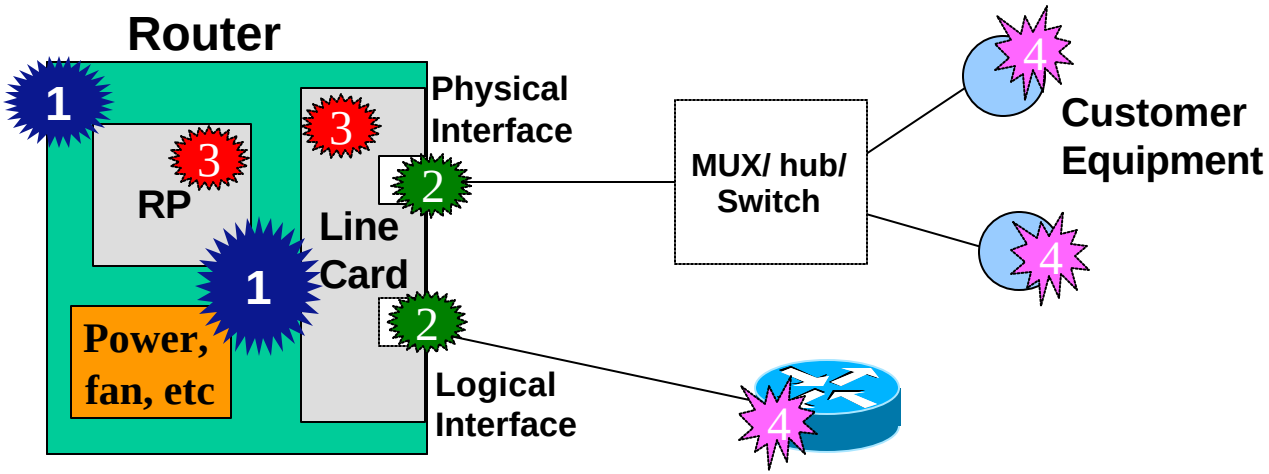
  The time between when the device/network broke and when it was brought back into service

  Dependant on:

  - ✓ Time taken for provider to identify and fix faults
  - ✓ Component stock and where it is located

Case Study 1: [2]
Engineered Availability: Possible Failures

1. Physical
2. Interface
3. Software
4. Remote



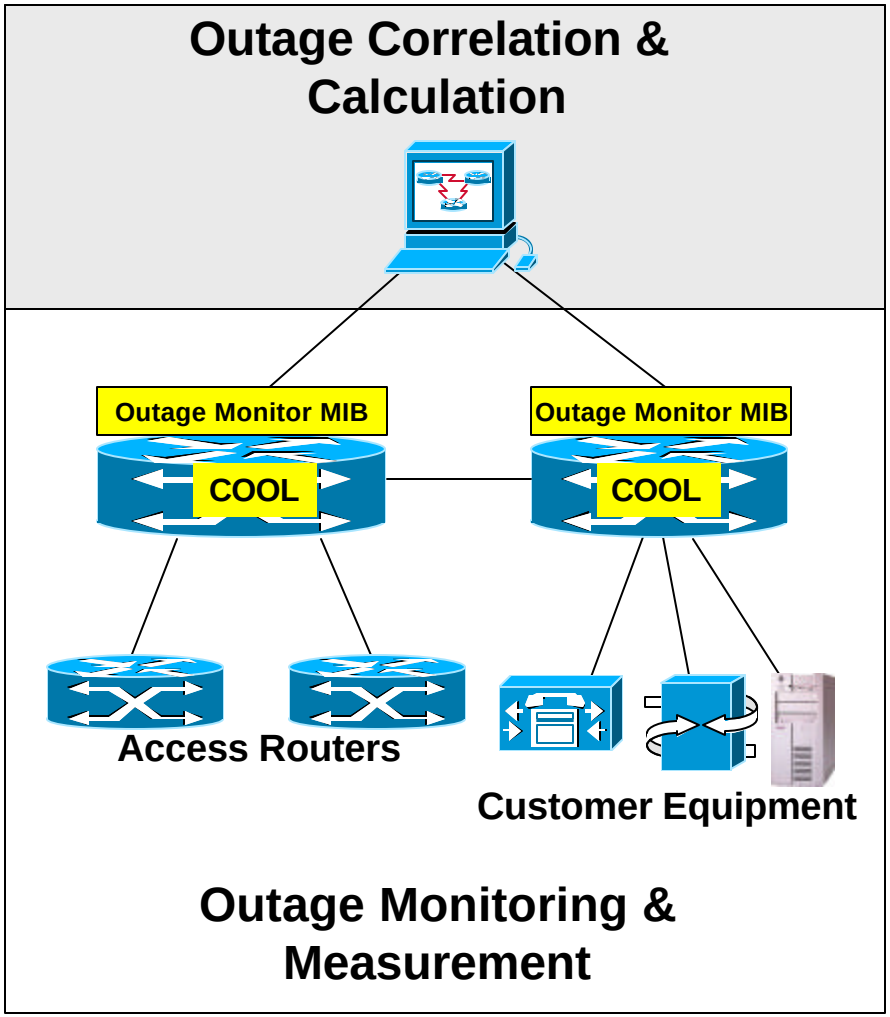
- Require MTBF for each component
  - Devices consist to multiple components each with an MTBF
- Single inline components reduce availability
- Parallel components improve availability
  - Examples: Dual PSU, Processors, links to remote devices, etc

Case Study 1: [3]
Engineered Availability: Component Outage On-Line

Component Outage On-Line (COOL)

On device MTBF and MTTR monitoring

- **Features**
  - ✓ On-line measurement embedded in router
  - ✓ Outage data stored in router
  - ✓ Maintains values over reload
  - ✓ Open access via CLI and CISCO-OUTAGE-MONITOR-MIB
  - ✓ Event filtering



Case Study 1: [4]
Engineered Availability: COOL

Cool Monitors:

- ✓ The device
- ✓ Interfaces Physical and Logical
- ✓ Remote devices
- ✓ Individual Processes for IOS-XR

Example Configuration

Not Engineered Availability but IP connectivity

Step	Command Line	Description
1	`(config)# cool group-interface Ethernet`	Monitor all Ethernet Interfaces
2	`(config)# cool group-interface Serial4`	Monitor all interfaces containing the string Serial4
3	`(config)# cool remote device 1 10.10.10.1 R1-2610 60 5`	Monitor a remote device R1-2610 Send 5x ICMP Ping to 10.10.10.1 once every 60s
4	`(config)# cool run`	Start Cool running

Case Study 1: [5]
Engineered Availability: COOL

1. Type
   - Failure code
2. Index
   - SNMP Index
3. Event
   - Event code 0=up
4. Time Stamp
   - Unix Time Stamp
5. Interval
   - Time since last event for that object
6. AOT
   - Accumulated Outage Time in seconds.
7. NAF
   - Number of Accumulated Failures

```
R4-7507#sh cool
 ****  COOL Event Table ****
type index event time-stamp  interval hist_id  object-name
1      3     0   1114509296   15          10    Ethernet1/0/0
1      3     1   1114509336   40          11    Ethernet1/0/0
1      3     0   1114509346   10          12    Ethernet1/0/0
4      1     5   1114509357   105         13    R1-2610

 ****  COOL Object Table ****
type index status last-change   AOT    NAF  object-name
1      3     1      1114509346   9358    5   Ethernet1/0/0
1      4     2      1114508564   15748   1   Ethernet1/0/1
1      7     1      1114508564   0       0   Serial4/0
2      1     1      1114508564   140     1   Router-Device
4      1     2      1114509357   183     1   R1-2610
```

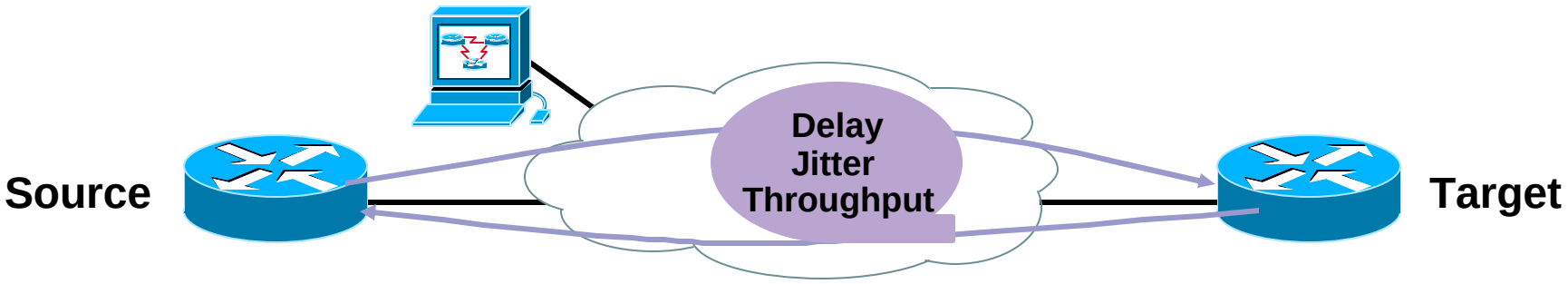
Remote device reloaded. 183s to restore IP connectivity

Interface not configured

Device reloaded once and took 140s to reboot

CASE STUDY 2: Connectivity

Case Study 2: Connectivity [1]



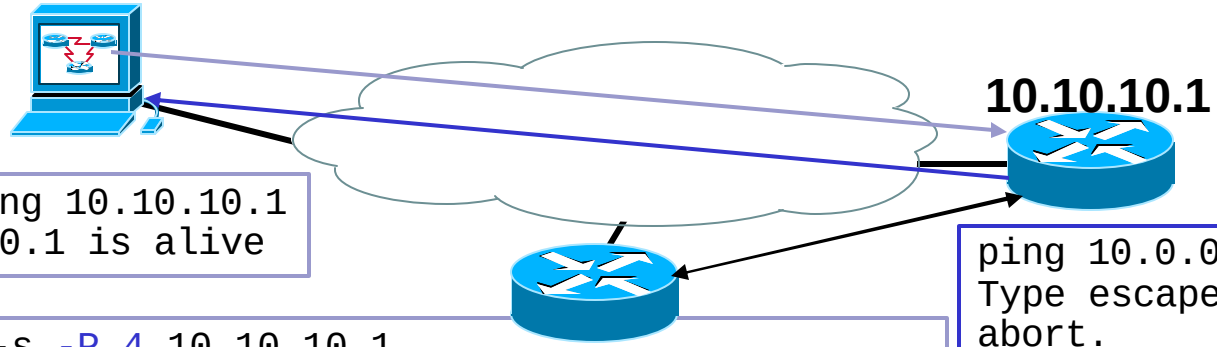
- **Availability is the metric used to determine uptime and downtime**

  Availability = (Uptime)/(Total Time)

- **In terms of a Network do I have connectivity or not**
- **Examples: COOL, CLI SNMP, PingMIB, IPSLA**
- **Adding bounded metrics provides some measurement of a service**

Case Study 2: Connectivity [2]
ICMP PING: CLI

- **Ping gives you availability details from:**
  - **Network management station to your devices.**
  - **Between Devices**
- **Ping can be QoS aware (TOS is configurable)**

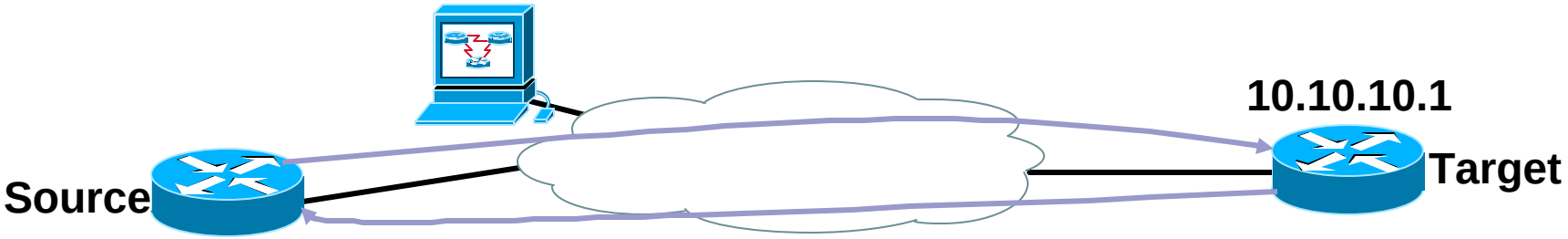



```
nms% ping 10.10.10.1
10.10.10.1 is alive
```

```
nms% ping -s -P 4 10.10.10.1
PING 10.10.10.1: 56 data bytes
64 bytes from 10.10.10.1: icmp_seq=0. time=66. ms
64 bytes from 10.10.10.1: icmp_seq=1. time=110. ms
64 bytes from 10.10.10.1: icmp_seq=2. time=71. ms
<snip>
^C
----10.10.10.1 PING Statistics----
6 packets transmitted, 6 packets received, 0% packet
loss
round-trip (ms)  min/avg/max = 32/63/110
```

```
ping 10.0.0.1
Type escape sequence to
abort.
Sending 5, 100-byte ICMP
Echos to 10.0.0.1, timeout
is 2 seconds:
!!!!!
Success rate is 100
percent (5/5), round-trip
min/avg/max = 76/97/156 ms
```

Case Study 2: Connectivity [3]
CISCO-Ping-MIB: SNMP



ftp://ftp.cisco.com/pub/mibs/v2/CISCO-PING-MIB.my

SNMP SET

```
1: ciscoPingEntryStatus.333 (integer) CreateAndWait(5)
2: ciscoPingProtocol.333 (integer) ip(1)
3: ciscoPingAddress.333 (octet string) 0A:0A:0A:01
4: ciscoPingEntryOwner.333 (octet string) dmelton
5: ciscoPingPacketCount.333 (integer) 100
6: ciscoPingEntryStatus.333 (integer) Active(1)
```

createAndWait(5) ping operation 333

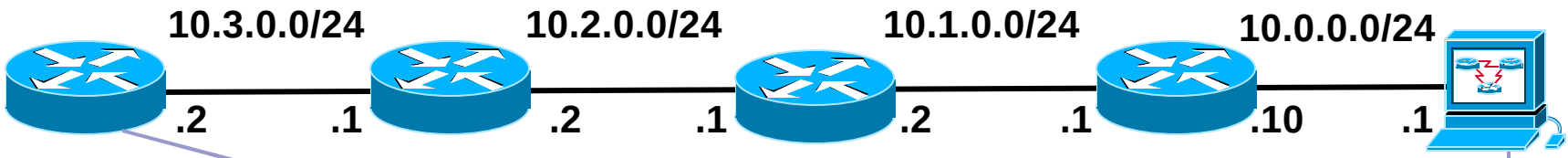
1st 4 SNMP sets required and must be set in order shown

Start probe
Runs once
Destroy(6) and repeat using CreateAndWait(5)

SNMP GET/WALK

```
ciscoPingSentPackets.333 (counter) 100
ciscoPingReceivedPackets.333 (counter) 100
ciscoPingMinRtt.333 (integer) 56
ciscoPingAvgRtt.333 (integer) 84
ciscoPingMaxRtt.333 (integer) 236
ciscoPingCompleted.333 (integer) true(1)
```

Case Study 2: Connectivity [4]
Traceroute: CLI

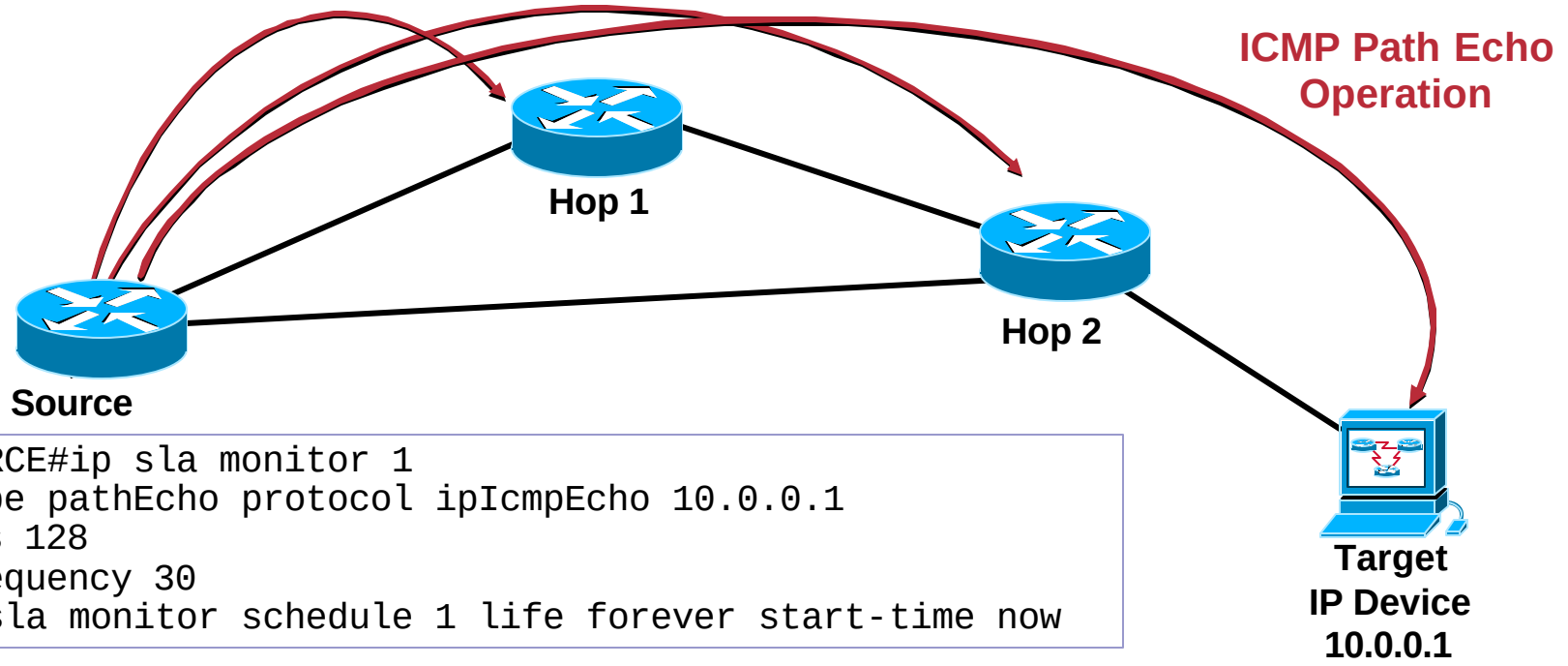


- **Indicates where failures occur**
- **Response at each hop**
- **Total network response time**
- **Traceroute can be QoS aware**

```
pag1#traceroute 10.0.0.1
Type escape sequence to abort.
Tracing the route to 10.0.0.1
  1 10.3.0.1 4 msec  0  msec 4   msec
  2 10.2.0.1 36 msec 77 msec 316 msec
  3 10.1.0.1 72 msec 80 msec 80  msec
  4 10.0.0.1 85 msec 72 msec 84  msec
```

```
traceroute: Warning: Multiple interfaces found; using
10.0.0.1 @ qfe0
traceroute to 10.10.10.1 (10.10.10.1), 30 hops max, 40
byte packets
 1  10.0.0.10 (10.0.0.10)  1.427  ms  0.737  ms 0.582  ms
 2  10.1.0.2  (10.1.0.2)   1.923  ms  2.096  ms 4.282  ms
 3  10.2.0.2  (10.2.0.2)   78.871 ms  80.063 ms 78.250 ms
 4  10.3.0.2  (10.3.0.2)   79.223 ms  *         37.262 ms
```

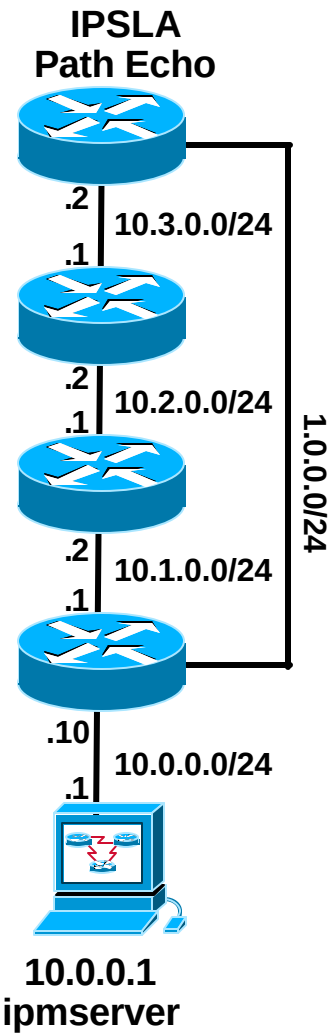
Case Study 2: Connectivity [5]
ICMP Path Echo: IPSLA

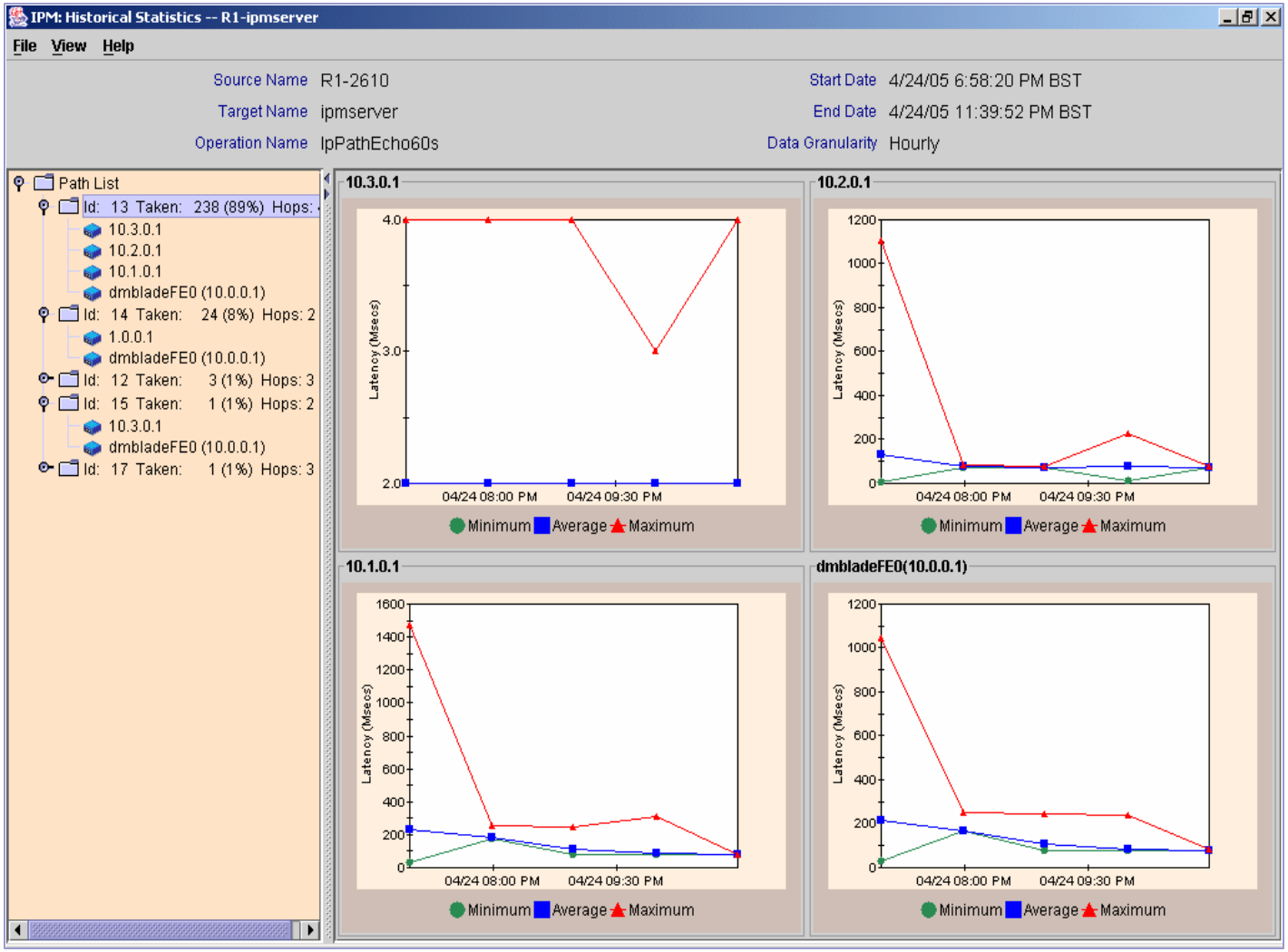



```
SOURCE#ip sla monitor 1
 type pathEcho protocol ipIcmpEcho 10.0.0.1
 tos 128
 frequency 30
ip sla monitor schedule 1 life forever start-time now
```

```
SOURCE#sh ip sla monitor distribution-statistics
<SNIP>
Entry StartT  Pth Hop Dst Comps OvrTh SumCmp TMax  TMin
1     4043292  1   1   1    1     0     72    72     72
1     4043292  1   2   1    1     0      6     6      6
1     4043292  1   3   1    1     0     74    74     74
1     4043292  2   1   1    6     0    440    74     73
1     4043292  2   2   1    6     0     36     6      6
```

Case Study 2: Connectivity [6]
ICMP Path Echo: IPM and IPSLA





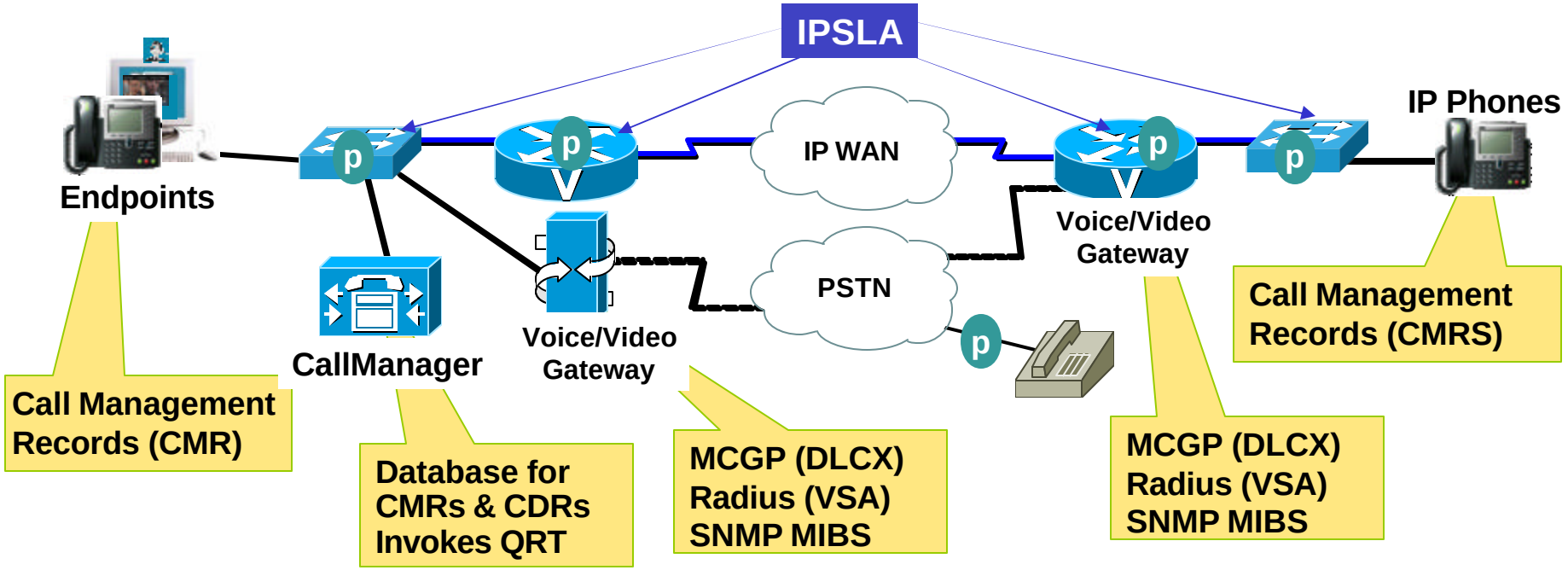
Case Study 2: Connectivity [7]
MPLS Network Availability

- **Basic IP Ping concepts have been extended to other technologies**
  - **IP Ping does not detect MPLS network failures**
- **Ping within a VRF**
  - **Used By Service Providers to check customer VPN connectivity from within the Service Provider network**
- **MPLS Aware Ping**
  - **LSP-PING detects MPLS failures and provides diagnostics**
  - **Also works for Traffic Engineering and Pseudowires**
  - **All devices along the Path must support LSP-ping**

CASE STUDY 3:
VoIP Quality

Case Study 3: VoIP Quality [1] Where do we Measure ?

- **Identify the points where Voice Quality Statistics is provided**
  - **As near to the end points (IP Phones) as possible**
  - **Embedded Passive Monitoring is preferred**
  - **Active Monitoring provides continuous monitoring**
  - **Embedded/External Probes for mid point analysis**



Case Study 3: VoIP Quality [2]
Basic Voice Quality Metrics

- **MOS**
  - Mean Opinion Score
  - Subjective voice quality measurement
- **R-factor**
  - Transmission Rating Factor
  - Estimates voice quality by calculating sum of impairments along voice path
- **ICPIF**
  - Calculated Planning Impairment Factor
  - Outdated by R-factor
  - Cisco voice gateways and IPSLA uses ICPIF

Quality	User Satisfaction	MOS	ICPIF	R-Factor
Best	Very Satisfied	5	0 - 3	-
High	Satisfied	4	4 - 13	100 - 80
Medium	Some users dissatisfied	3	14 – 23	59 – 79
Low	Many users dissatisfied	2	24 – 33	39 – 58
Poor	Nearly all users dissatisfied	1	34 – 43	0 - 38

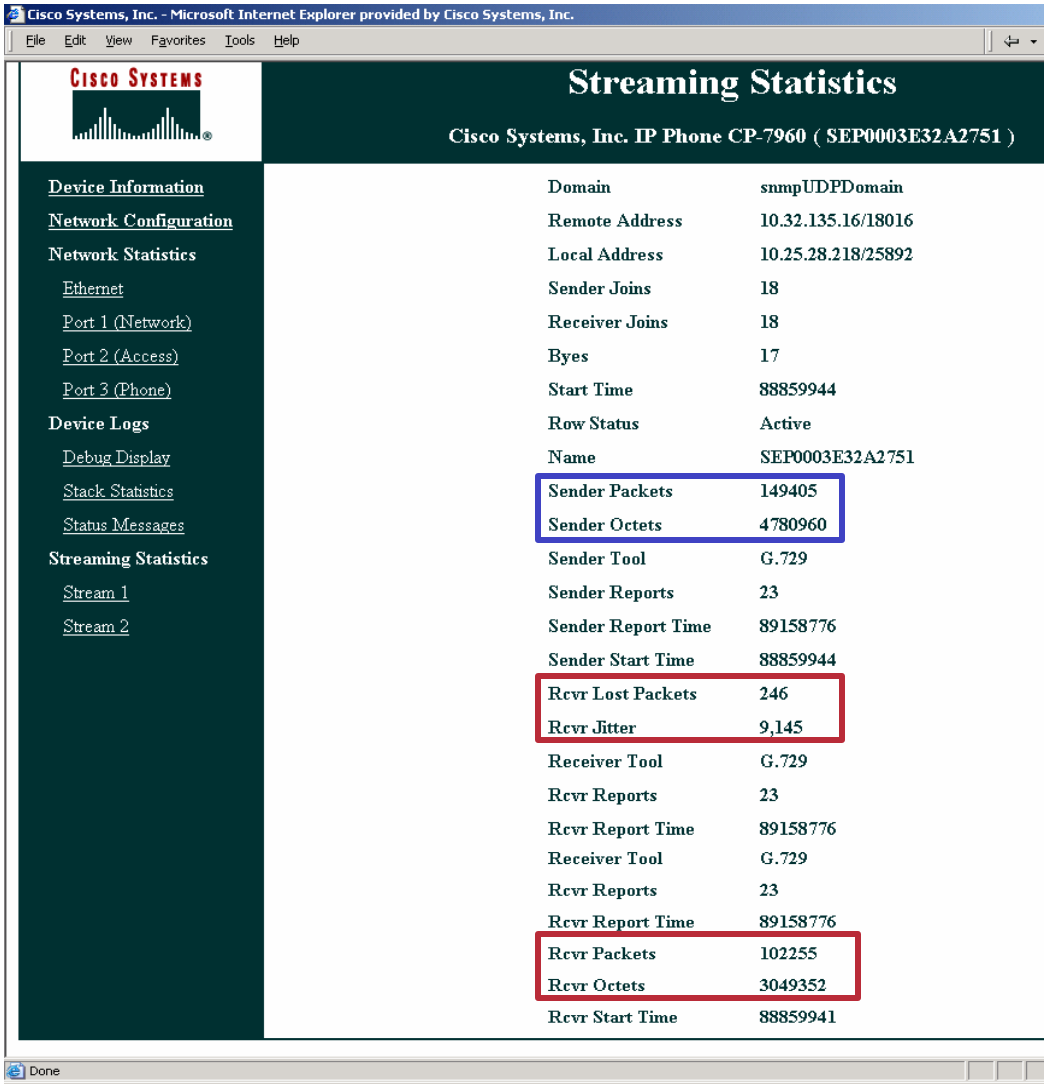
Note: R-Factor of 100 gives MOS 4.5

Case Study 3: VoIP Quality [3]
IP Phone Statistics

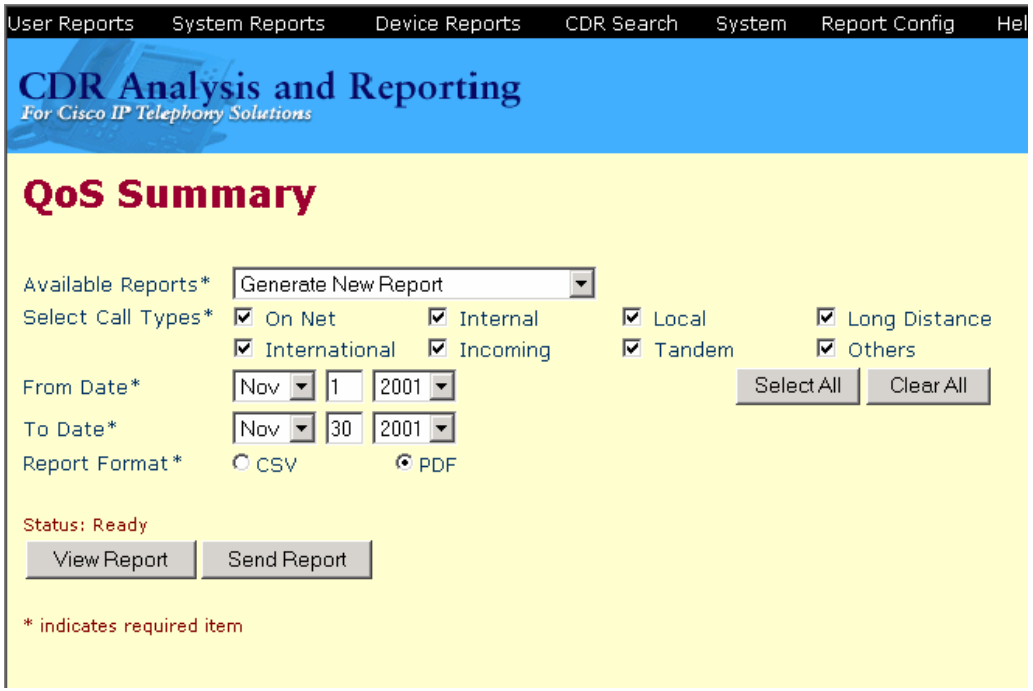
- IP Phone Call Statistics
  - ✓packets/bytes sent
  - ✓Packets lost
  - ✓Jitter
- Statistics for each end of the call

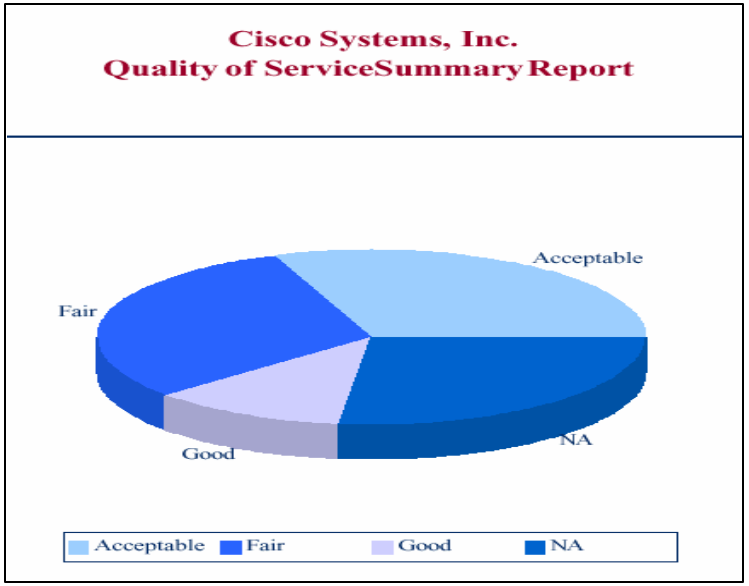
  End-to-end statistics requires an IP phone at both ends of the call
- Sent to Call Manager at end of call

  This can average out Voice Quality issue in the reports



Case Study 3: VoIP Quality [4]
Call Manager Statistics: CMR reporting





- **Call Management records CMR from IP phone at end of Call**
- **Administrator defines thresholds for Call Quality Categories**
- **Call Detail Records (CDR) generated at end of call.**

Case Study 3: VoIP Quality [5]
Call Manager Statistics: QRT

Quality Report Tool (QRT)

- **User Presses QRT key to log issues**

  9 Possible options (Echo, Choppy sound, Robotic sound, Long delays, Low volume, Can't hear other end).

  If Call is in process, QRT logs on to User IP phone (HTTP) and collects CMR statistics.

  QRT polls phone at 30-3600s (30s default).

  Data is recorded in log file not CMR reports.

  ccmQualityReport - This Notification is sent when a user reports a quality problem using the Quality Report Tool

Case Study 3: VoIP Quality [6]
Voice Gateway Statistics

Most useful Voice Gateway Statistics

- **PlayoutDelay (Minimum, Maximum, Average)**

  A measure of jitter on the packets received by the gateway.

- **Gap Fill with Prediction/Silence**

  Total duration of packets lost during the call.

- **Round Trip Delay (RTD)**

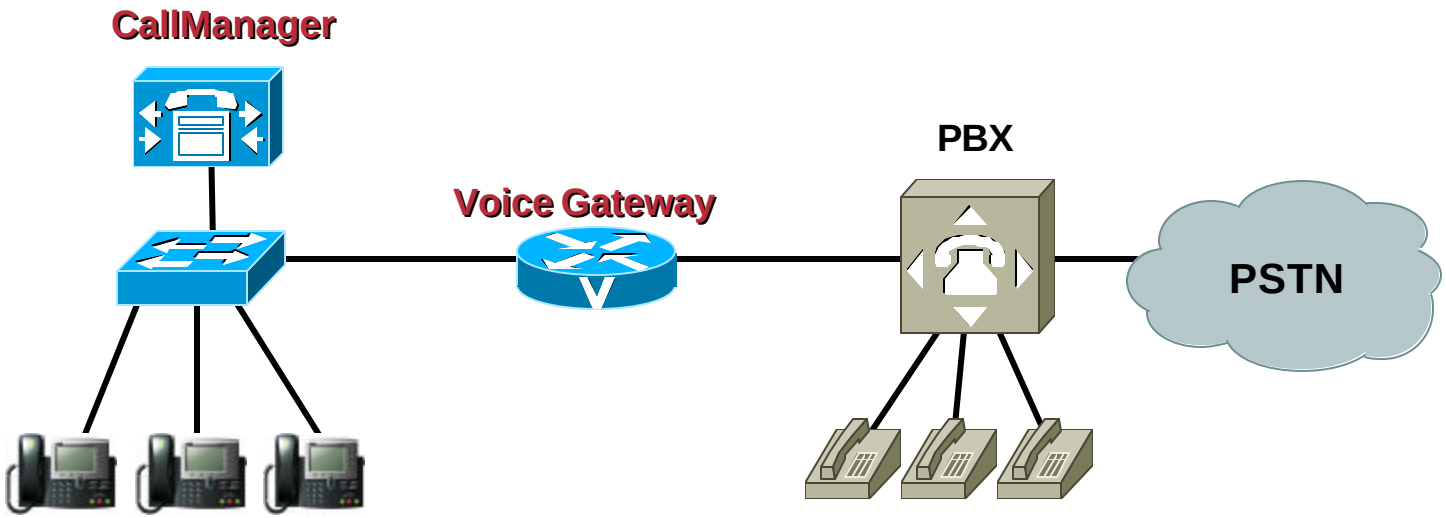
  The delay between the gateway and its peer.

  Uses RTCP so only works with a gateway on both ends of the RTP stream.

- **Calculated Planning Impairment factor (ICPIF)**

  Single metric for VoIP Quality. Mean Opinion Score (MOS) is more widely used

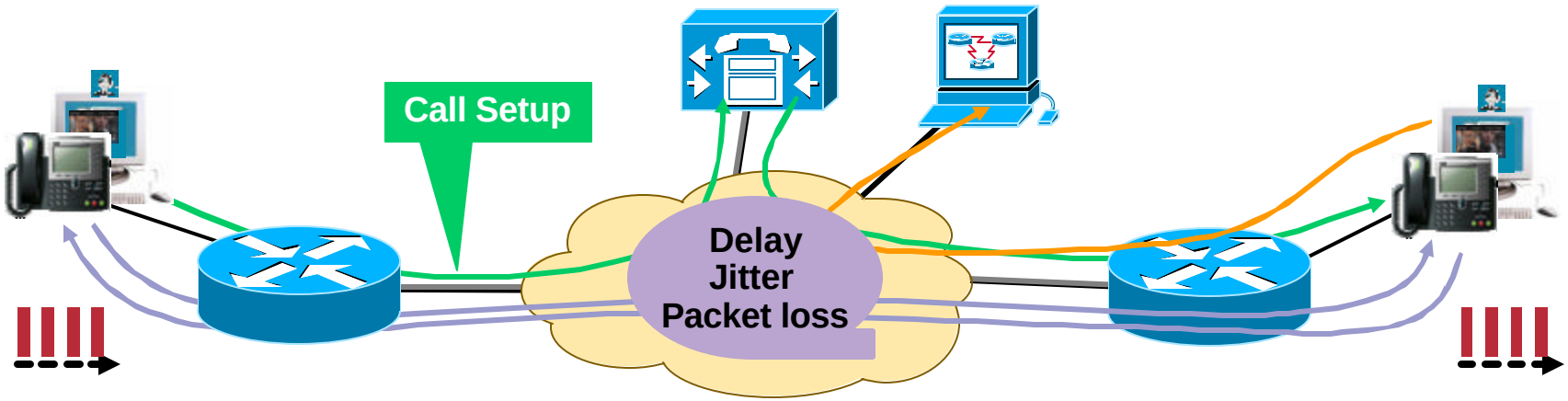
Case Study 3: VoIP Quality [7]
Voice Gateway Statistics



- **Voice Quality Information is exported from Gateway:**
  - **MGCP: Delete Connection (DLCX) message at the end of the call.**
  - **SIP, H.232: Radius server in the Vendor Specific Attributes (VSA), at the end of the call**
  - **SNMP Polled: Cisco-Voice-Dial-Control-MIB**

CASE STUDY 4:
Services

Case Study 4: Services [1]



- **Monitoring VoIP is not just about Monitoring VoIP Quality**
  - **Availability of Voice Services**
  - **Quality of VoIP over network infrastructure**
- **Networks carry many other services**
  - **May tolerate some degree of jitter, delay & packet loss**
  - **Examples; WEB services, file transfers**

Case Study 4: Services [6]
IPSLA Operations

	11.2	12.0(3)T	12.0(5)T 12.0(8)S	12.1(1)T	12.2(2)T	Eng2
ICMP Ping	X	X	X	X	X	12.2(11)T
ICMP Echo Path	X	X	X	X	X	12.2(11)T
SSCP (SNA)	X	X	X	X	X	
UDP Echo		X	X	X	X	12.2(11)T
TCP Connect		X	X	X	X	12.2(11)T
UDP Jitter			X	X	X	12.2(11)T
HTTP			X	X	X	12.2(11)T
DNS			X	X	X	12.2(11)T
DHCP			X	X	X	12.2(11)T
DLSw+			X	X	X	
One-Way Latency with UDP				X	X	12.2(11)T
FTP Get				X	X	12.2(11)T
MPLS VPN Aware					X	12.2(11)T
Frame Relay (CLI)					X	12.2(11)T
ICMP Path Jitter					X	12.2(11)T
VoIP UDP					X	12.3(4)T
VoIP Proactive Monitoring						12.3(7)T

All services

Case Study 3.2.2: Services [7]
IPSLA: VoIP

Standard jitter, Packet loss, latency measurements

```
(config)#type jitter dest-ipaddr 10.10.10.1
                 dest-port 5555 num-packets 100
```

Complex output difficult to estimate VoIP quality

Voice quality score measurements

```
(config)#type jitter dest-ipaddr
10.10.10.1 dest-port 5555 codec ?
       g711alaw  G.711 A Law 64000 bps
       g711ulaw  G.711 U Law 64000 bps
       g729a       G.729 8000 bps
```

MOS + ICPIF, based on delay and packet loss, provide direct measure of VoIP Quality

Call setup measurements

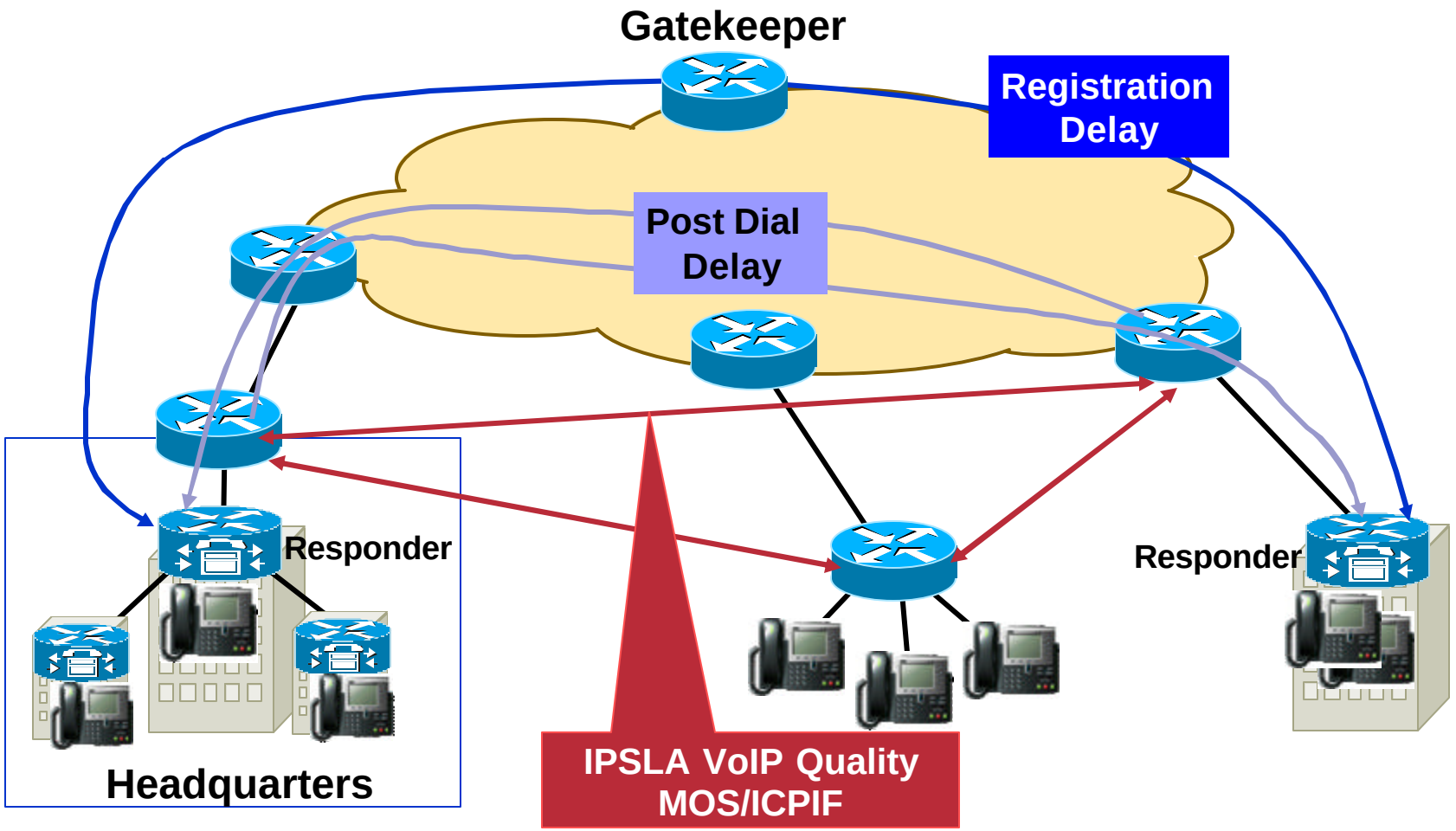
```
(config)#type voip delay gatekeeper registration
```

Enabled on Gatekeeper

```
(config#type voip delay post-dial
                        destination 449614
```

Response of test calls to voice end point

Case Study 4: Services [8]
IPSLA: VoIP 12.3(14)T



Case Study 4: Service [9] IPSLA: VoIP 12.3(14)T

- **Simulating G.711 A-Law codec (64 kbps transmission) VoIP Call**

```
(config)#
        ip sla monitor 10
         type jitter dest-ipaddr 10.10.10.4 dest-port 16384 codec
g711alaw advantage-factor 2
         owner admin
         tag jitter-with-voice-scores
        ip sla monitor schedule 10 start-time now
        ip sla monitor reaction-configuration 10 react mos
threshold-type immediate threshold-value 490 250 action-type
trapOnly
        ip sla monitor logging traps
        snmp-server host 10.10.10.10 version 2c public
        snmp-server enable traps syslog
```

set default values for: codec-numpackets, codec-size, & codec-interval

Translates IP SLA syslog in to Traps

connectionLoss, jitterAvg, jitterDSAvg, jitterSDAvg, Mos, PacketLossDS, PacketLossSD Rtt, Timeout, verifyError

0: Wire/land-line
5: Mobility in building
10: Mobility in Geographical area
20: Hard to reach location

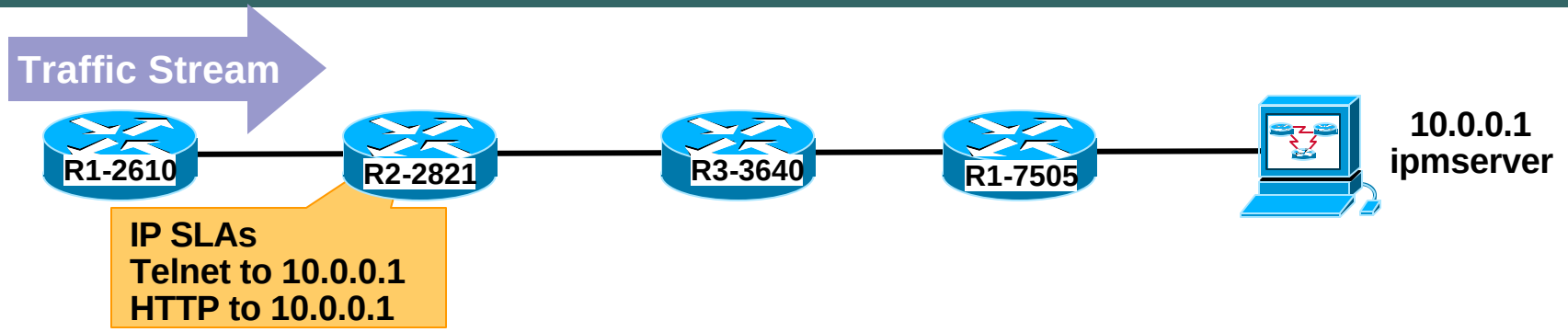
Note: Logging Commands Are Needed Only if You Want to Send the Syslog Message to a Logging Server

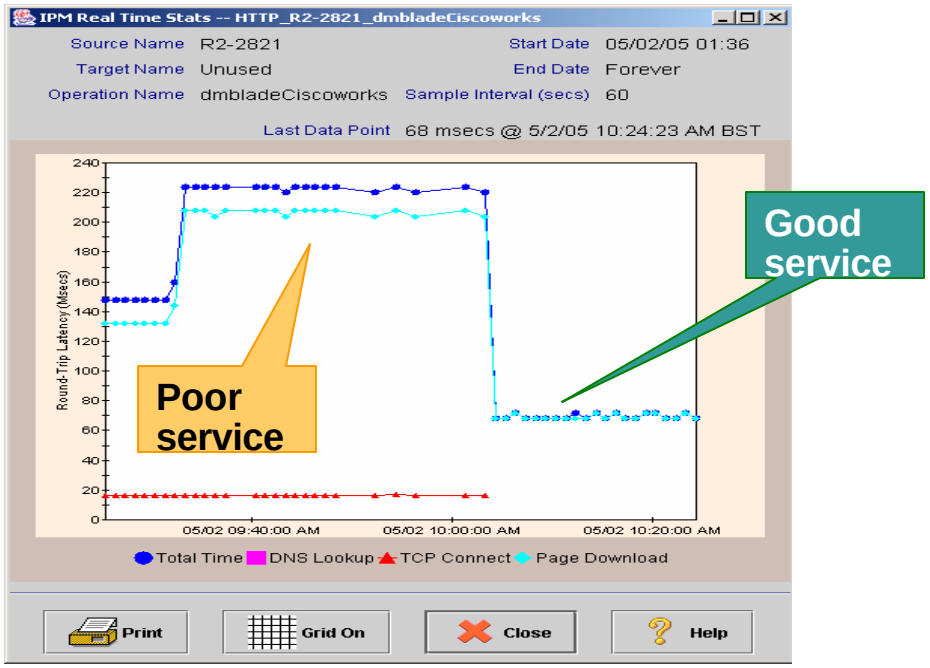
Case Study 4: Services [10]
IPSLA: VoIP

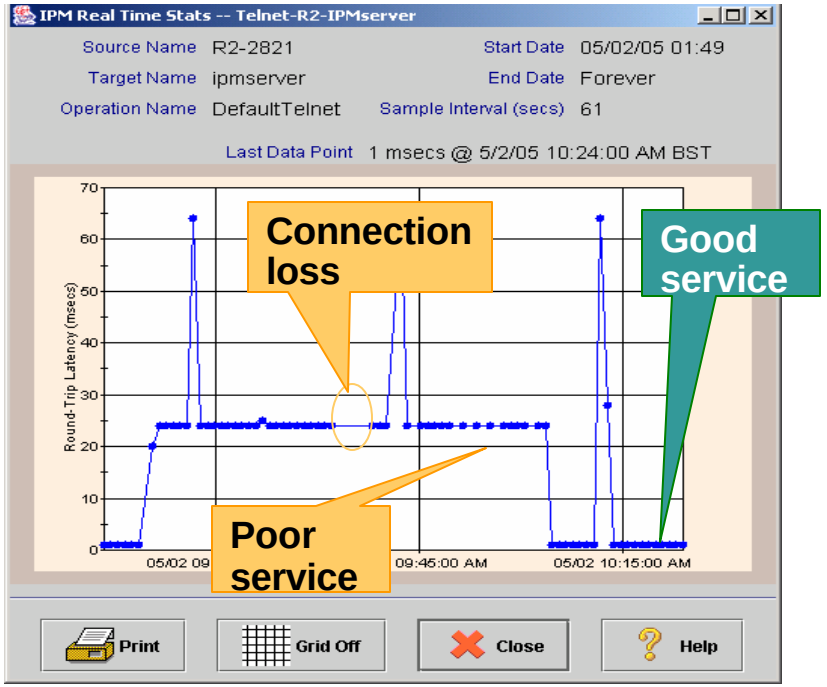

```
R2-2821#sh ip sla monitor operational-state 1
<SNIP>
Voice Scores:
ICPIF Value: 20 MOS score: 3.20
RTT Values:
NumOfRTT: 11    RTTAvg: 2583    RTTMin: 711     RTTMax: 4699
RTTSum: 28422   RTTSum2: 92644272
Packet Loss Values:
PacketLossSD: 0 PacketLossDS: 0
PacketOutOfSequence: 0  PacketMIA: 989  PacketLateArrival: 56
InternalError: 0        Busies: 0
Jitter Values:
MinOfPositivesSD: 1     MaxOfPositivesSD: 249
NumOfPositivesSD: 197   SumOfPositivesSD: 8792  Sum2PositivesSD: 794884
MinOfNegativesSD: 1     MaxOfNegativesSD: 158
NumOfNegativesSD: 761   SumOfNegativesSD: 8811  Sum2NegativesSD: 139299
MinOfPositivesDS: 1     MaxOfPositivesDS: 273
NumOfPositivesDS: 317   SumOfPositivesDS: 7544  Sum2PositivesDS: 581458
MinOfNegativesDS: 1     MaxOfNegativesDS: 183
NumOfNegativesDS: 603   SumOfNegativesDS: 6967  Sum2NegativesDS: 336135
Interarrival jitterout: 16      Interarrival jitterin: 35
One Way Values:
NumOfOW: 0
OWMinSD: 0      OWMaxSD: 0      OWSumSD: 0      OWSum2SD: 0
OWMinDS: 0      OWMaxDS: 0      OWSumDS: 0      OWSum2DS: 0
```

SD: Source to Destination
DS: Destination to Source
OW: One Way Delay

Case Study 4: Services [11]
IPSLA: Monitoring Other Services



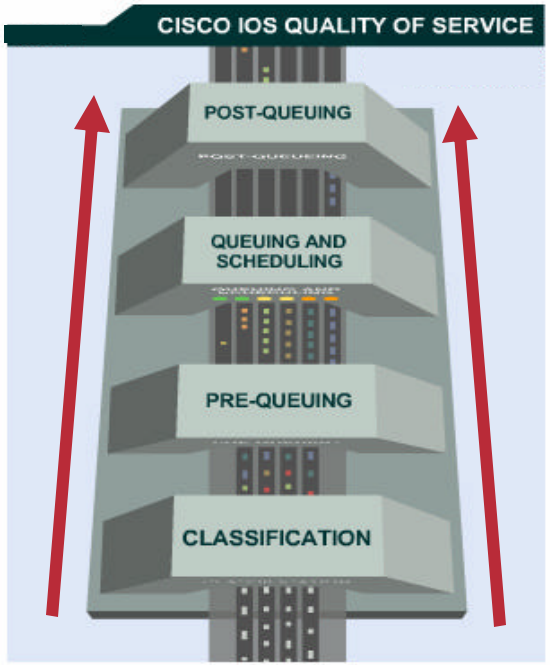




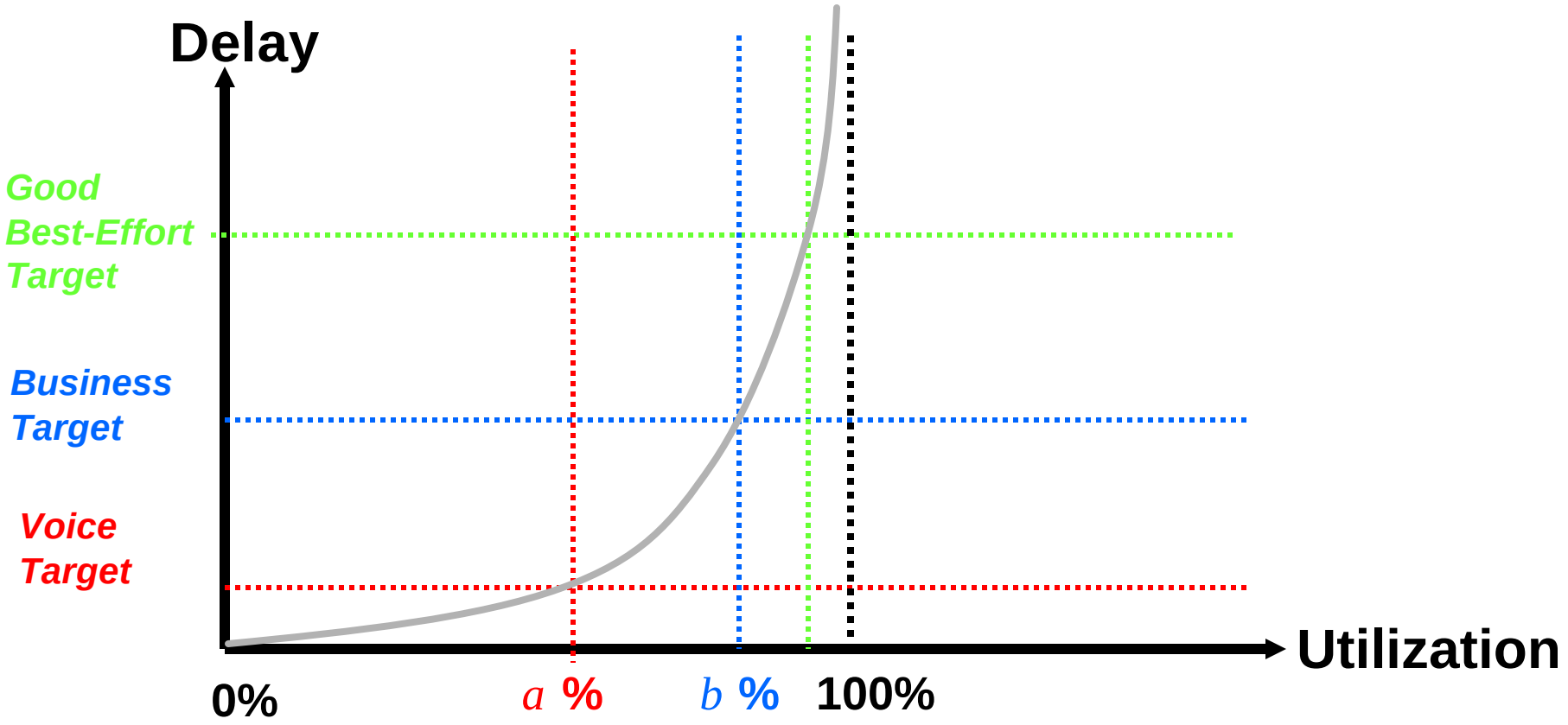
CASE STUDY 5: QoS

Case Study 5: QoS [1]
When is QoS Used?

- If bandwidth is abundantly, economically, and reliably available, there is no need for QoS
- Otherwise, QoS mechanisms are:
  - Essential for real-time applications
    - Two way, interactive: voice, videoconferencing
    - One way, streaming: movies, lectures
  - Important for high priority data applications
    - Response-time sensitive: interactive, transactional
  - Less important for "bulk" data applications – except when assigned to lower priority
    - "Elastic" applications: will wait and use data whenever packets arrive
  - Helpful to limit undesired traffic
    - Peer-to-peer file sharing in some networks

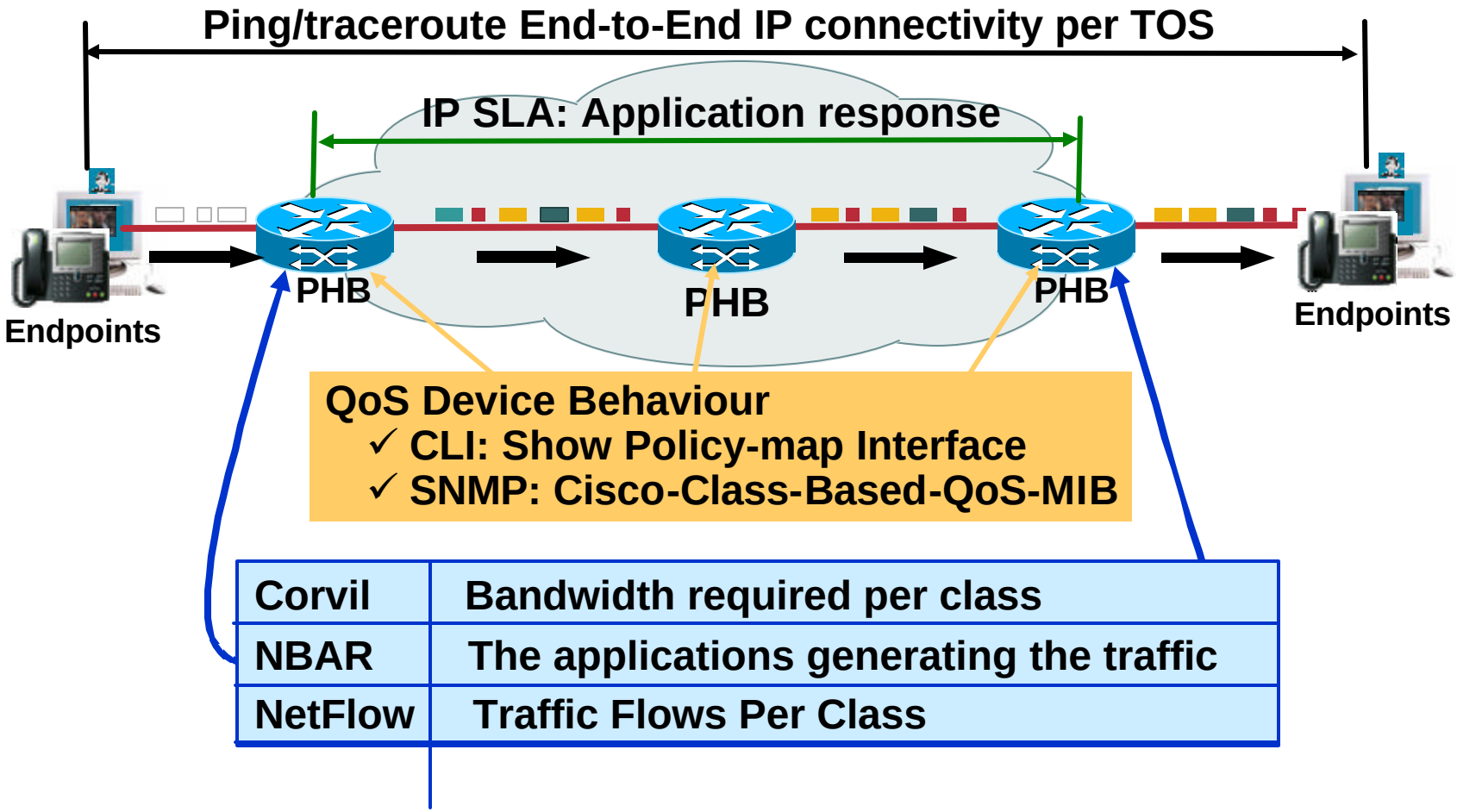


Case Study 5: QoS [2]
Managed Unfairness



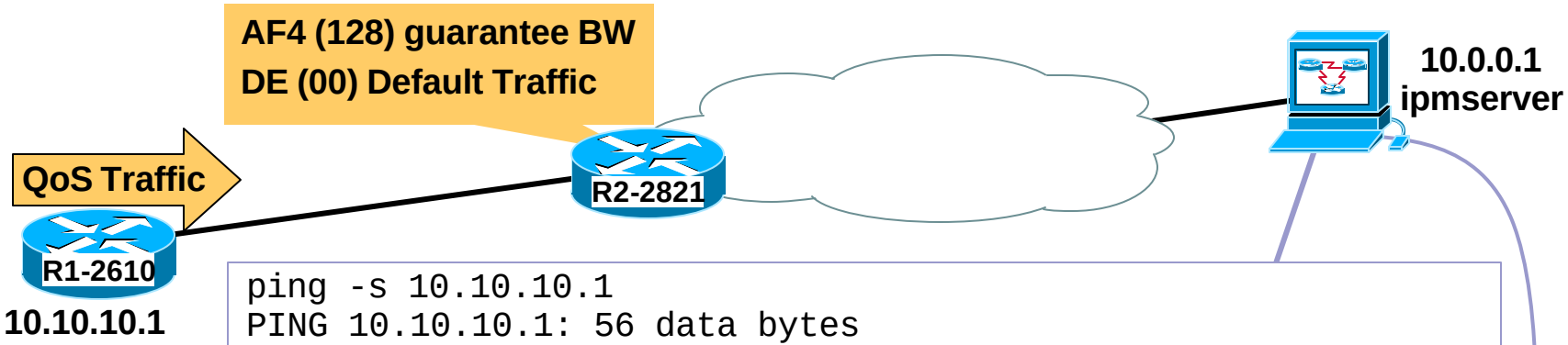
If I can keep Voice traffic < a % , I will keep Voice delay under *M1* ms
If I can keep Business traffic < b % , I will keep Business delay under *M2* ms

Case Study 5: QoS [3]
QoS Monitoring Tools



Note: PHB Per-hop behaviour

Case Study 5: QoS [4]
CLI Ping




```
ping -s 10.10.10.1
PING 10.10.10.1: 56 data bytes
64 bytes from 10.10.10.1: icmp_seq=0. time=159. ms
64 bytes from 10.10.10.1: icmp_seq=1. time=205. ms
<snip>
----10.10.10.1 PING Statistics----
7 packets transmitted, 7 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 40/92/205
```

Ping in Class AF4 improves response

Traceroute in class DE identifies hop causing increased delay

```
# ping -s -c 128 10.10.10.1
PING 10.10.10.1: 56 data bytes
64 bytes from 10.10.10.1: icmp_seq=0. time=19. ms
64 bytes from 10.10.10.1: icmp_seq=1. time=22. ms
<snip>
----10.10.10.1 PING Statistics----
6 packets transmitted, 6 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 19/47/183
```

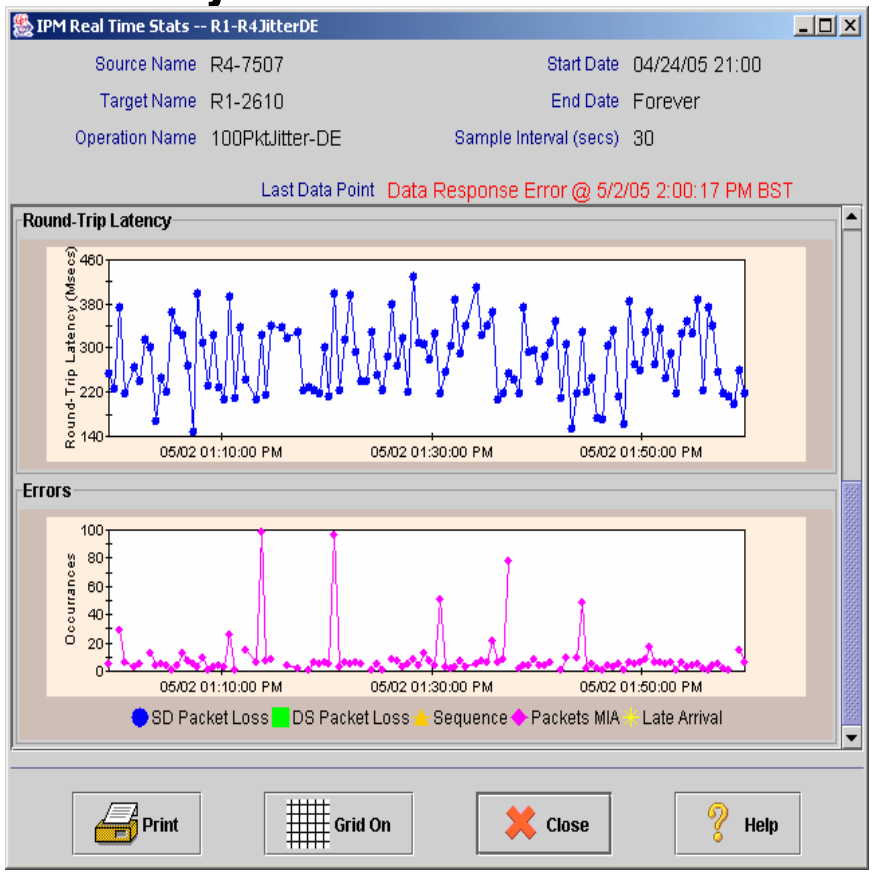
Case Study 5: QoS [5]
IPSLA and QoS

- If you have the right probe type to match the service use it
  - Examples: Telnet, HTTP, FTP, DNS
- Overall network quality should be measured using IP SLA jitter probes because:
  - A Sequence packets is sent per operation so more likely to be impacted by QoS
  - Provides best response measurements including:
    - ✓ Jitter: Source->Destination, Destination->Source
    - ✓ Packet Loss: Source->Destination, Destination->Source
    - ✓ Round trip and one-way delay.

```
R2-2821(config)#ip sla monitor 1
R2-2821(config-sla-monitor)#type jitter dest-ipaddr 10.10.10.4
                                       dest-port 4444
R2-2821(config-sla-monitor-jitter)#tos ?
               <0-255>  Type of Service Value
R2-2821(config-sla-monitor-jitter)#tos 128
```

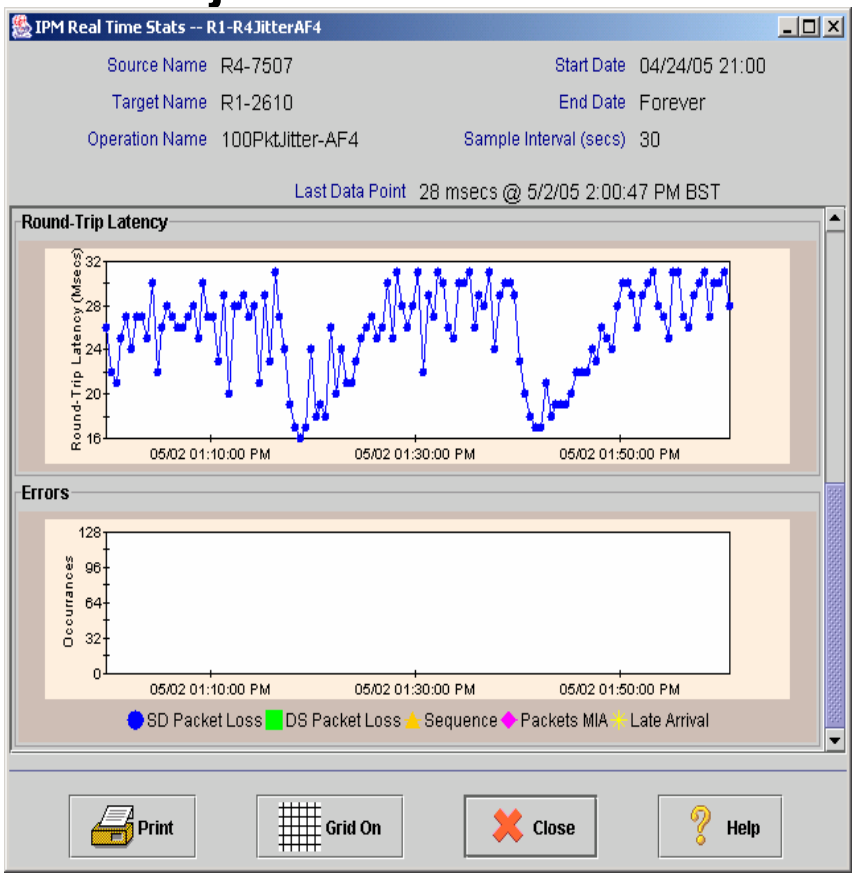
Case Study 5: QoS [6]
IPSLA & IPM

IP SLA jitter DE



Response 200-400ms
Excessive Packet Loss

IP SLA jitter AF4



Response 20-30ms
No Packet Loss

Case Study 5: QoS [7]
CLI: Show Policy-map

Show policy map-interface

- All QoS on the device
- There can be a lot of output


```
R2-2821#show policy-map interface serial 0/2/1
Serial0/2/1

Service-policy : inner_toPE
  Class-map: ef_toPE (match-any)
    67182 packets, 6559656 bytes
    30 second offered rate 0 bps, drop rate 0 bps
  Match: ip dscp ef (46)
    67182 packets, 6559656 bytes
    30 second rate 0 bps
    Queueing
      Strict Priority
      Output Queue: Conversation 24
      Bandwidth 50 (kbps) Burst 1250 (Bytes)
      (pkts matched/bytes matched) 28136/1352768
      (total drops/bytes drops) 294/18400
     police:
       cir 50000 bps, bc 1562 bytes
       conformed 57467 packets, 4829248 bytes;
       actions: transmit
       exceeded 8091 packets, 1645920 bytes;
       actions: drop
       conformed 0 bps, exceed 0 bps
```

```
Class-map: af4_toPE (match-any)
  716176 packets, 29235196 bytes
  30 second offered rate 8000 bps, drop rate 0 bps
  Match: ip dscp cs4 (32) af41 (34)
     715982 packets, 29219676 bytes
     30 second rate 8000 bps
   Match: ip dscp af42 (36) af43 (38)
      0 packets, 0 bytes
      30 second rate 0 bps
    Match: protocol ntp
       194 packets, 15520 bytes
       30 second rate 0 bps
     Queueing
       Output Queue: Conversation 25
       Bandwidth 100 (kbps) Max Threshold 64 (packets)
       (pkts matched/bytes matched) 495810/23671304
       (depth/total drops/no-buffer drops) 0/0/0

Class-map: class-default (match-any)
  24515193 packets, 654530939 bytes
  30 second offered rate 380000 bps, drop rate 190000 bps
  Match: any
```

Case Study 5: QoS [8]
CLI: Show Policy-map

- **How the device treats QoS**
  - **Statistics for all QoS enable objects**
- **clear counters interface**
  - **Resets QoS statistics**
- **Configure Interface load-interval to 30s**
  - **More responsive output**
  - **Check offered rates**

```
Class-map: af4_toPE (match-any)
  716176 packets, 29235196 bytes
  30 second offered rate 8000 bps, drop rate 0 bps
  Match: ip dscp cs4 (32) af41 (34)
    715982 packets, 29219676 bytes
    30 second rate 8000 bps
  Match: ip dscp af42 (36) af43 (38)
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol ntp
    194 packets, 15520 bytes
     30 second rate 0 bps
  Queueing
   Output Queue: Conversation 25
   Bandwidth 100 (kbps) Max Threshold 64 (packets)
   (pkts matched/bytes matched) 495810/23671304
   (depth/total drops/no-buffer drops) 0/0/0
```

Traffic in AF4 will have a good service

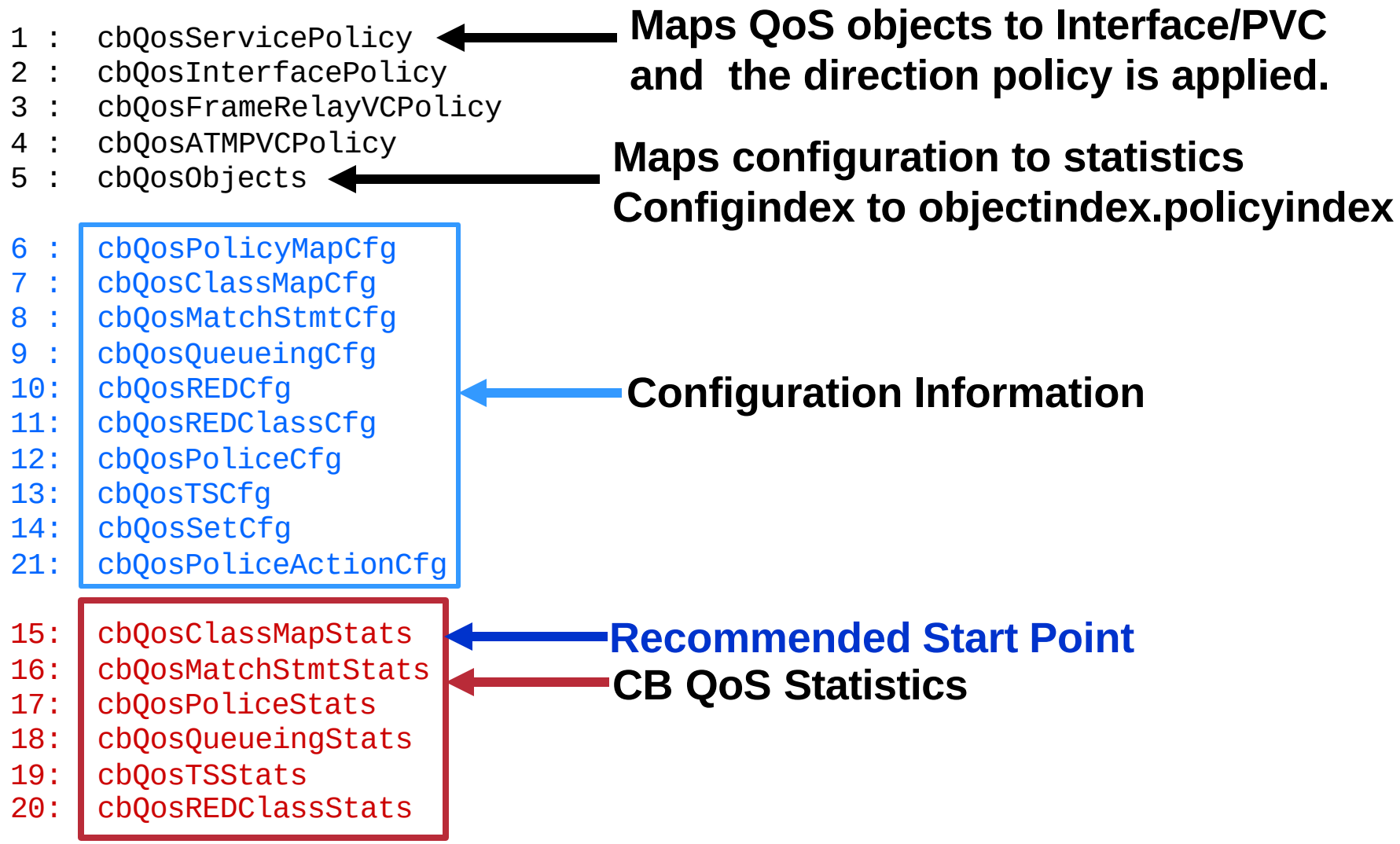
```
Class-map: class-default (match-any)
  24515193 packets, 654530939 bytes
  30 second offered rate 380000 bps, drop rate
                                     190000 bps
```

Traffic in DE will not have a good service

Case Study 5: QoS [9]
Cisco-Class-Based-QoS-MIB

- **Primary accounting mechanism for QoS:**
  - ✓ **Policing**
  - ✓ **Classification**
  - ✓ **Shaping**
  - ✓ **Queuing**
  - ✓ **Congestion Avoidance**
- **Long term QoS Monitoring**
  - **NMS Partners: Concord, InfoVista, Quallaby, Cisco Info Centre, Cisco Info Centre, Cisco QoS Policy Manager**
- **Only accounts on configured QoS behaviour**
  - **Does not inspect packets for TOS/DSCP**
- **Provides equivalent statistics to “Show policy-map interface”**
  - **Counters can not be reset.**
- **Navigation is Complex**

Case Study 5: QoS [10]
Cisco-Class-Based-QoS-MIB



Case Study 5: QoS [11]
CBQoSMIB: Quick Navigation

1. Show Service policy interface
2. Record: Service Policy, Class map & offered rate
3. SNMP Walk Classmap Table PrePolicyBitrate

```
cbQosCMPrePolicyBitRate.1103.1107  5000    EF
cbQosCMPrePolicyBitRate.1103.1117  8000    AF4
cbQosCMPrePolicyBitRate.1103.1131  386000  DE in
cbQosCMPrePolicyBitRate.1103.1137  401000  DE Out
```

4. Compare CBQos PrePolicybit rates to Show Policy Int offered rate
5. Assumes QoS traffic is applied to the device

```
Service-policy output: outer_toPE
 Class-map: class-default (match-
any)
 30 second offered rate 401000
bps,drop rate 201000 bps
```

```
Service-policy : inner_toPE
 Class-map: ef_toPE (match-any)
 18544 packets, 593408 bytes
 30 second offered rate 5000 bps,
 drop rate 0 bps
```

```
 Class-map: af4_toPE (match-any)
  79836 packets, 3248112 bytes
  30 second offered rate 8000 bps,
  drop rate 0 bps
```

```
 Class-map: class-default
  (match-any)
  4472619 packets, 114998532 bytes
  30 second offered rate 386000
bps, drop rate 201000 bps
```

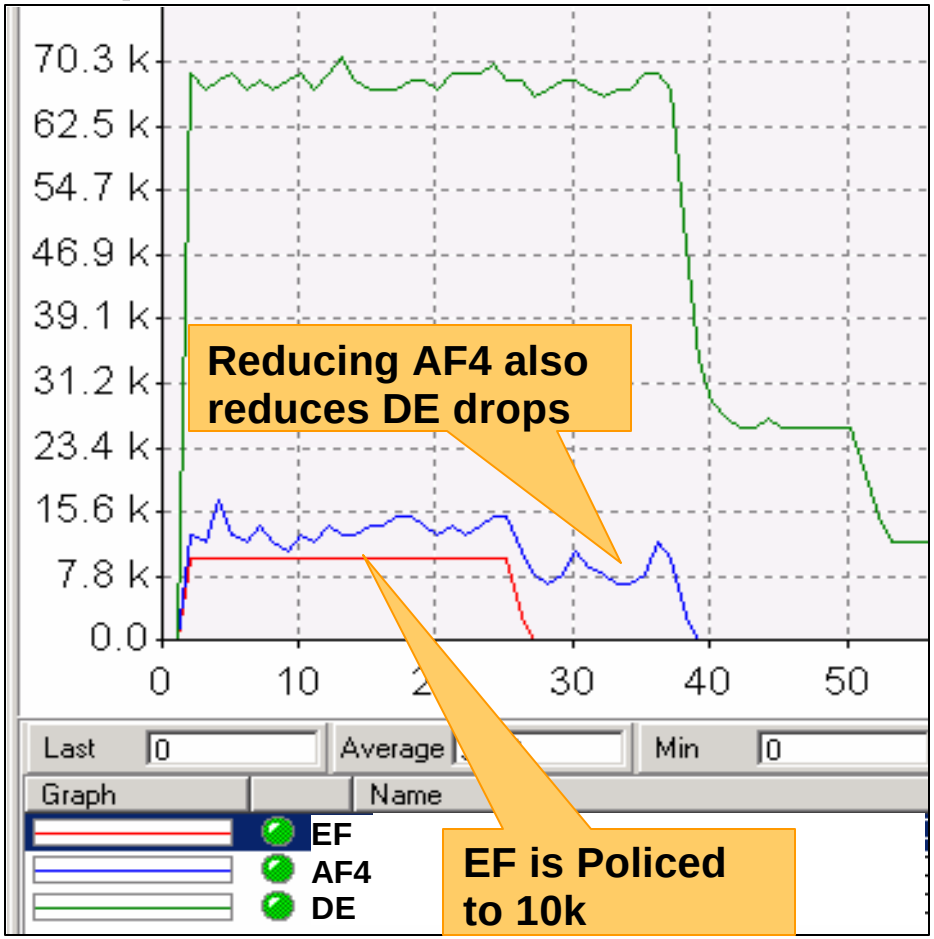
Case Study 5: QoS [12]
CBQoSMIB: Performance Monitoring

- **Historical Analysis**
  - **Pre QoS Policy Statistics: cbQosCMPrePolicyByte64**
  - **Post Policy Statistics: cbQosCMPostPolicyByte64**

Delta (Pre Policy Bytes - Post Policy Bytes) x8
Delta(sysuptime)

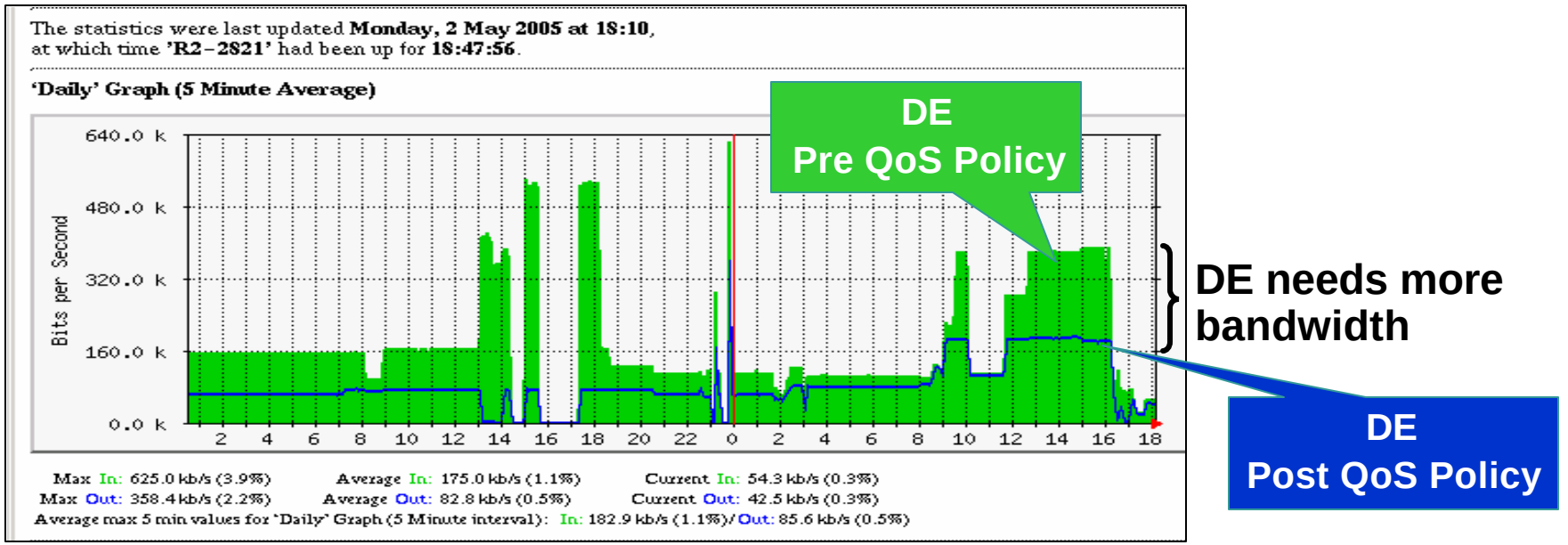
- **Troubleshooting**
  - **Dropped Bytes/Packets: cbQosCMDropBitRate**
- **Indexes not yet persistent over reboot**
- **Partners**
  - **Concord, InfoVista, Quallaby**
  - **Cisco QPM, MRTG**

DropBitRate for EF, AF4 & DE

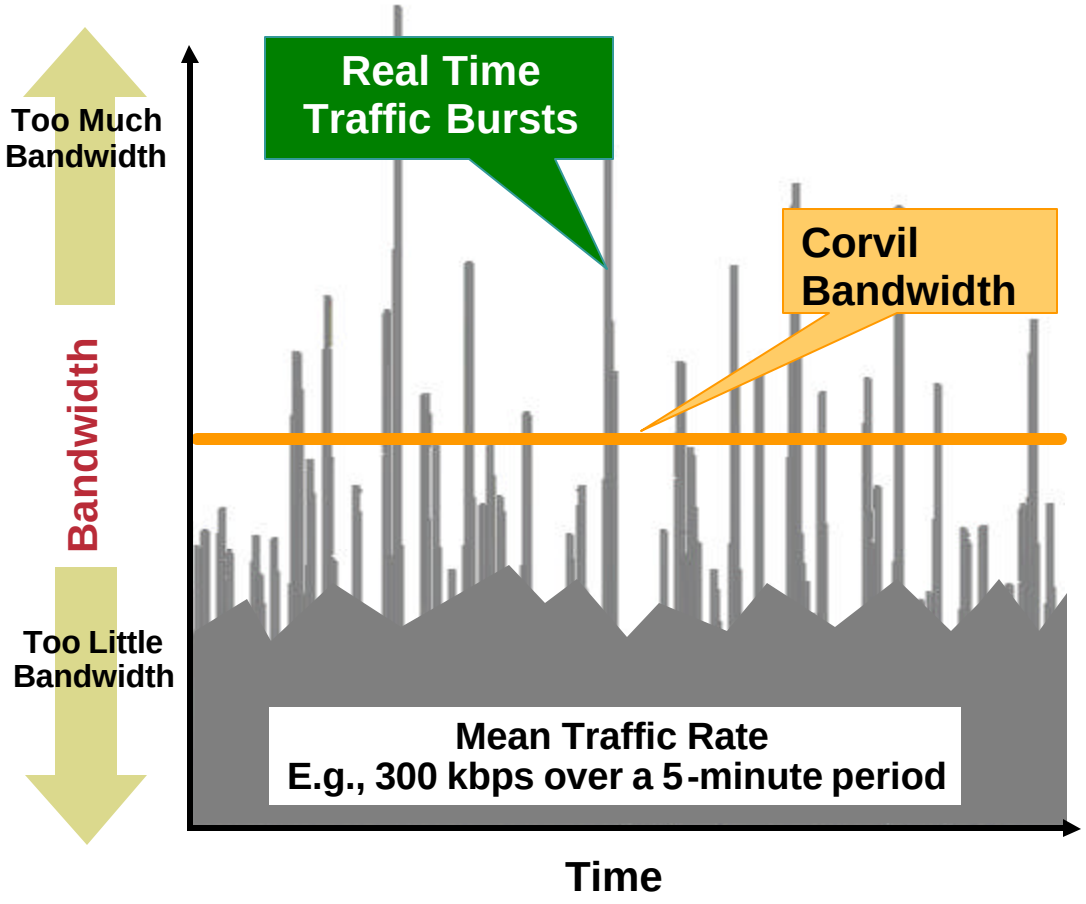


Case Study 5: QoS [13]
Corvil Bandwidth

- We know that we are dropping traffic
- How can we calculate how much bandwidth we need:
  - To deliver against a defined SLA
  - Per Class of Service



Case Study 5: QoS [14]
Corvil Bandwidth



- Adequate bandwidth is essential for application performance
  - Too little bandwidth can make business services unusable
  - Too much bandwidth can be very costly
- Corvil Bandwidth provides the minimum bandwidth required to meet a Quality of Service target
  - Based on 8ms measurement within a traffic class.

Case Study 5: QoS [15]
Corvil Bandwidth: Configuration

- **Commence observation for a traffic class**
  - **Use `estimate bandwidth` command in an MQC policy map**
    - **Specify tolerance for packet loss**
    - **Specify tolerance for delay duration**

```
Policy-map inner_toPE
   class class-default
      R2-2821(config-pmap-c)#estimate bandwidth drop-one-in 50
                             delay-one-in 50 milliseconds 100
```

  - ✓ **Allows 1 packet in 50 to be dropped packet loss <2.00%**
  - ✓ **Allows 1 packet in 50 to be delayed by 100ms**

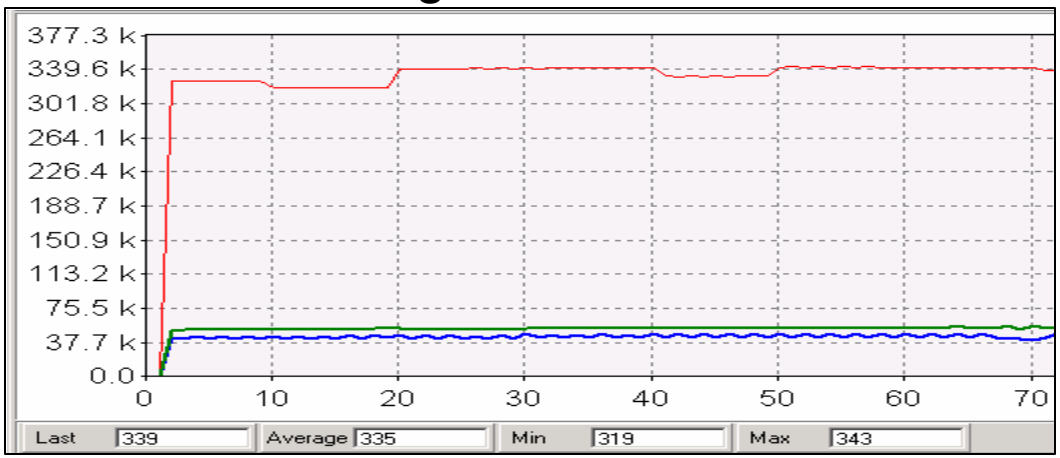
- **Can only be applied on outbound policies**
- **Monitoring is via CLI or SNMP**

Case Study 5: QoS [16]
Corvil Bandwidth: Monitoring

CLI: Show policy-map interface

```
Class-map: class-default (match-any)
     13970096 packets, 366991592 bytes
     30 second offered rate 57000 bps, drop rate 15000 bps
     Match: any
        Bandwidth Estimation:
         Quality-of-Service targets:
            drop no more than one packet in 50 (Packet loss < 2.00%)
            delay no more than one packet in 50 by 100 (or more) milliseconds
            (Confidence: 98.0000%)
            Corvil Bandwidth: 327 kbits/sec
```

SNMP: Monitoring Corvil Bandwidth Value

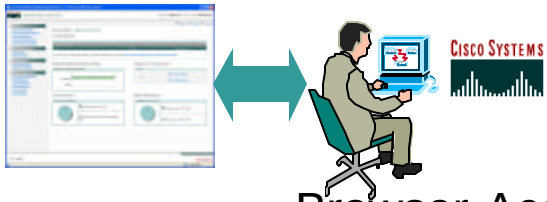


Corvil BW for DE

DE Pre Policy

DE Post Policy

Case Study 5: QoS [17]
Corvil Bandwidth: Monitoring Application



Browser Access

Cisco BQA Agent Software for IBM 336 Appliance

Cisco BQA Server Software for IBM 346 Appliance

Cisco CB Agent Software (Corvil Bandwidth feature license) in Cisco IOS (800 to 7200 routers)

Bandwidth Quality Analysis (BQA) Server

- **Required to get most benefit from Corvil BQA agents**
- **Corvil Traffic Descriptors (CTD) can only be decoded by the Server.**
- **Accounts for multiplexing across class**
- **Carries out what if scenarios**

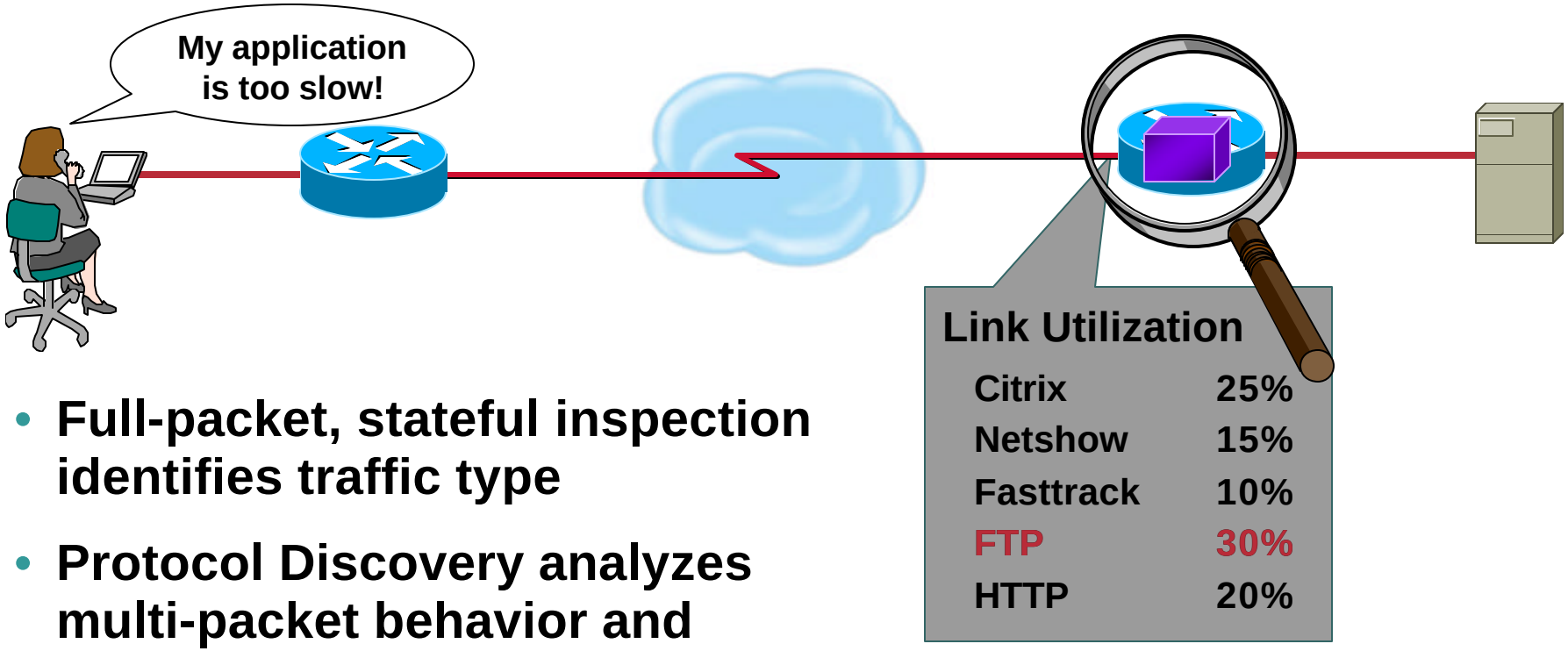
  CTD Library that represent traffic profiles of services

Case Study 5: QoS [18]
Network Based Application Recognition (NBAR)

Cisco.com

- QoS Case Studies so far
  - ✓ Network & Service response per Class of Service
    - Ping/IP SLA
  - ✓ Device behaviour to QoS
    - CLI / CBQoSMIB
  - ✓ Bandwidth required to meet defined SLA
    - Corvil
- We Still Don't Know
  - ✓ What applications are generating the traffic
  - ✓ Traffic classification as it enters the network
  - ✓ Traffic flows in the Network

Case Study 5: QoS [19]
NBAR



- **Full-packet, stateful inspection identifies traffic type**
- **Protocol Discovery analyzes multi-packet behavior and application signatures**
- **Enables application of QoS policies to traffic flows**

Mark Citrix as interactive traffic and police FTP

Assure bandwidth for Citrix!

© 2003, Cisco Systems, Inc. All rights reserved.

Case Study 4.5.2: QoS [20] NBAR Modes of Operation

- **Protocol Discovery Per Interface.**

```
R2-2821(config-if)#ip nbar protocol-discovery
```

per-interface, per-protocol, bi-directional statistics:

bit rate (bps)

packet counts

byte counts

- **Modular QoS Traffic Classification**

```
R2-2821(config)#class-map af4_toPE
R2-2821(config-cmap)#match protocol ?
  aarp              AppleTalk ARP
  appletalk         AppleTalk
  arp               IP ARP
  bgp               Border Gateway Protocol
  <snip>
  xwindows          X-Windows remote access
R2-2821(config-cmap)#match protocol Telnet
```

Lists all NBAR protocols

Case Study 5: QoS [21]
NBAR Protocol Discovery Configuration

Configuring Protocol Ports

```
(config)#ip nbar port-map bgp udp 179 180 181 182 183 184

(config)#ip nbar port-map custom-01 udp 1000 1100

(config)#ip nbar custom CiscoProtocol source udp
                        range  40000 41000 50000
```

- Redefine existing protocol ports
- Configure custom protocols
- Define protocol by name

Identifying Protocol Ports

```
#show ip nbar port-map
port-map CiscoProtocol  udp 40000 41000 50000
port-map bgp            udp 179 180 181
                            182 183 184
port-map bgp            tcp 179
<snip>
port-map telnet         tcp 23
port-map tftp           udp 69
port-map vdolive        tcp 7000
port-map winmx          tcp 6699
```

- List all protocols and ports
- Port-map and custom are maintained over reboot.

Case Study 5: QoS [22]
NBAR CLI

- **Top-N for all interfaces with Cisco NBAR Protocol Discovery enabled**
- **"show ip nbar protocol-discovery" lists all protocols including those with no applications.**

```
#sh ip nbar protocol-discovery top-n 5 Serial0/3/0

          Input                     Output
-----------------------------------------------------
Protocol  Packet Count              Packet Count
          Byte Count                Byte Count
          5min Bit Rate(bps)        5min Bit Rate(bps)
          5min Max Bit Rate(bps)    5min Max Bit Rate(bps)
-----------------------------------------------------
gre             52270               0
                1254480             0
                23000               0
                36000               0
egp             52269               0
                1254456             0
                23000               0
                36000               0
CiscoProtocol   7975                0
                255200              0
                18000               0
                18000               0
<snip>
Total           125679              4922
                3886478             857316
                73000               3000
                101000              4000
```

Case Study 4.5.3: QoS [23] NBAR SNMP

- **CISCO-NBAR-PROTOCOL-DISCOVERY-MIB**

  SNMP configures and collects protocol discovery statistics

  ```
  snmpwalk cnpdAllStatsEntry
  ```

  Lists all protocol entries including those with 0 counters. 99 protocols in this case

  Top-N statistics must be configured via SNMP

SNMP SET

```
1: cnpdTopNConfigStatus.5 (integer) createAndWait(5)
2: cnpdTopNConfigIfIndex.5 (integer) 5 [5]
3: cnpdTopNConfigRequestedSize.5 (gauge) 6
4: cnpdTopNConfigStatsSelect.5 (integer) bitRateIn(1)
5: cnpdTopNConfigSampleTime.5 (gauge) 30
6: cnpdTopNConfigStatus.5 (integer) createAndGo(4)
```

CreateAndWait for Interface ifindex 5

Create Top N protocol table For ifindex 5

Create Top 6 protocol table

Order by inbound interface bit rate

1st two SNMP sets required and in order shown

Bit rate over 30s interval

CreateAndGO top 6 protocol table for ifindex 5 (Serial 0/3/0)

Case Study 5: QoS [24]
NBAR SNMP

- Top N protocols are listed
  Based on counters
- Counters are:
  - Bit rate/s 30s interval
  - Inbound on Serial 0/3/0
  - As configured via SNMP
- NBAR Protocol Traps configured via SNMP

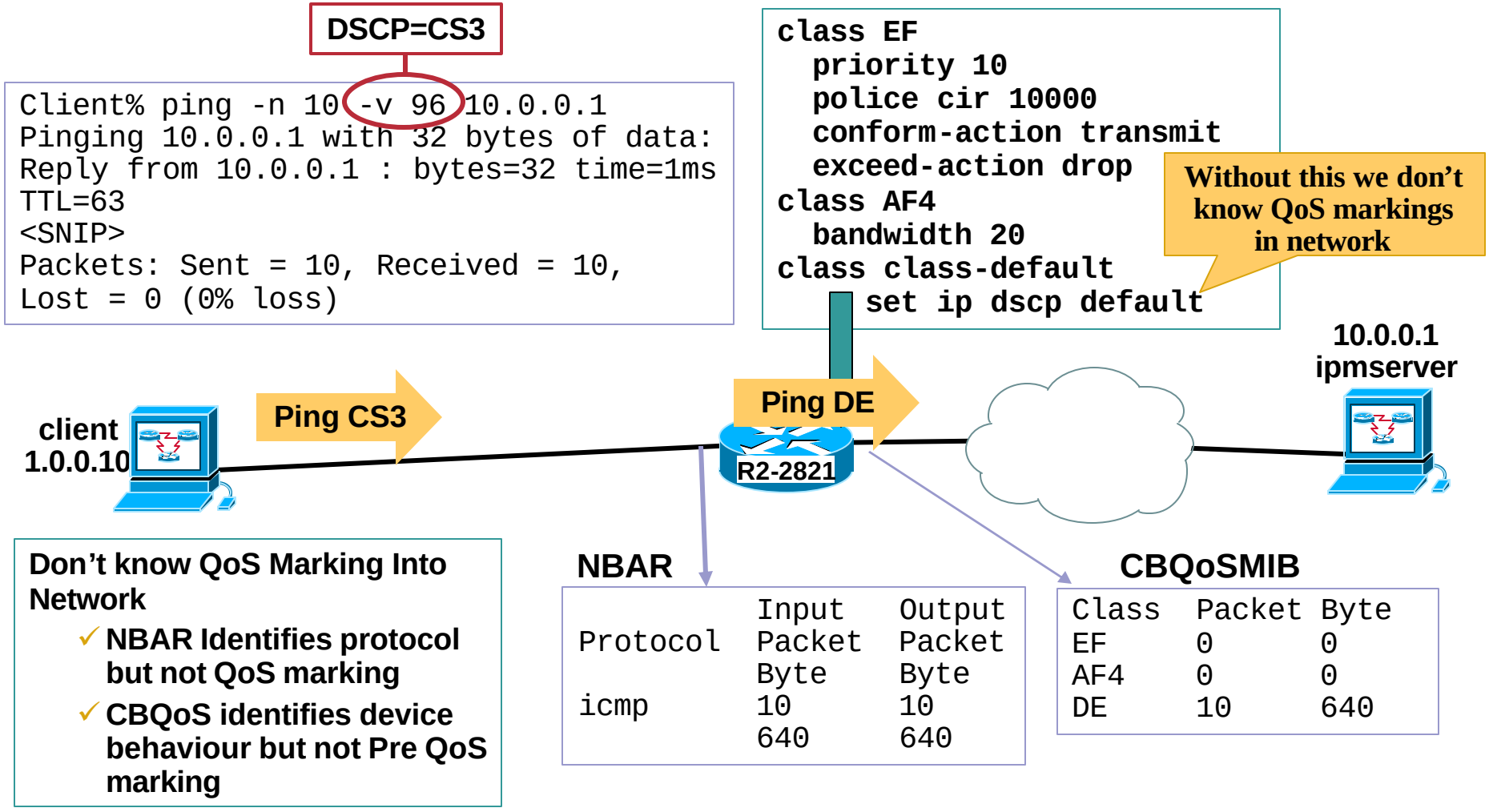
Based on

  - Any protocol
  - Defined protocol
  - Combination of threshold conditions

SNMP GET/WALK cnpdTopNStatsEntry

```
1: cnpdTopNStatsProtocolName.5.1  egp
2: cnpdTopNStatsProtocolName.5.2  gre
3: cnpdTopNStatsProtocolName.5.3  CiscoProtocol
4: cnpdTopNStatsProtocolName.5.4  l2tp
5: cnpdTopNStatsProtocolName.5.5  notes
6: cnpdTopNStatsProtocolName.5.6  rtp1
<snip>
1: cnpdTopNStatsHCRate.5.1 (counter64) 38000
2: cnpdTopNStatsHCRate.5.2 (counter64) 38000
3: cnpdTopNStatsHCRate.5.3 (counter64) 25000
4: cnpdTopNStatsHCRate.5.4 (counter64) 2000
5: cnpdTopNStatsHCRate.5.5 (counter64) 2000
6: cnpdTopNStatsHCRate.5.6 (counter64) 2000
```

Case Study 5: QoS [25]
NBAR + CBQoSMIB

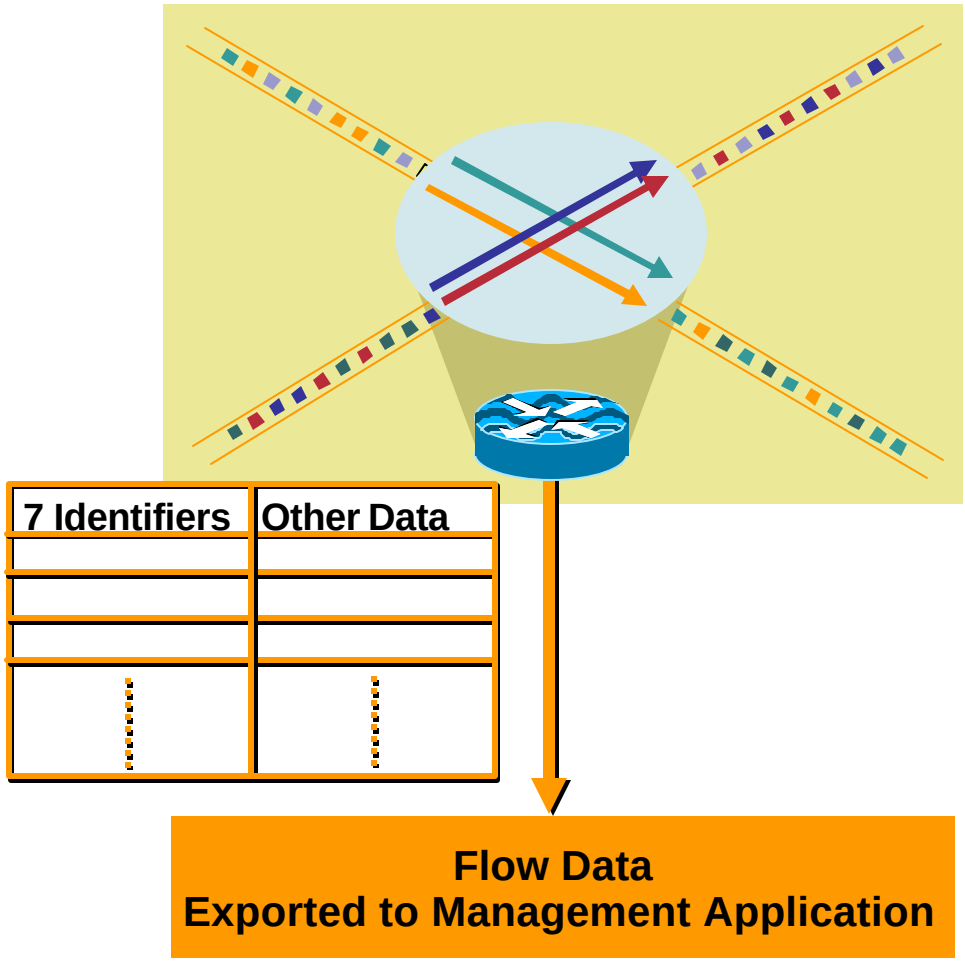


Case Study 5: QoS [26] NetFlow

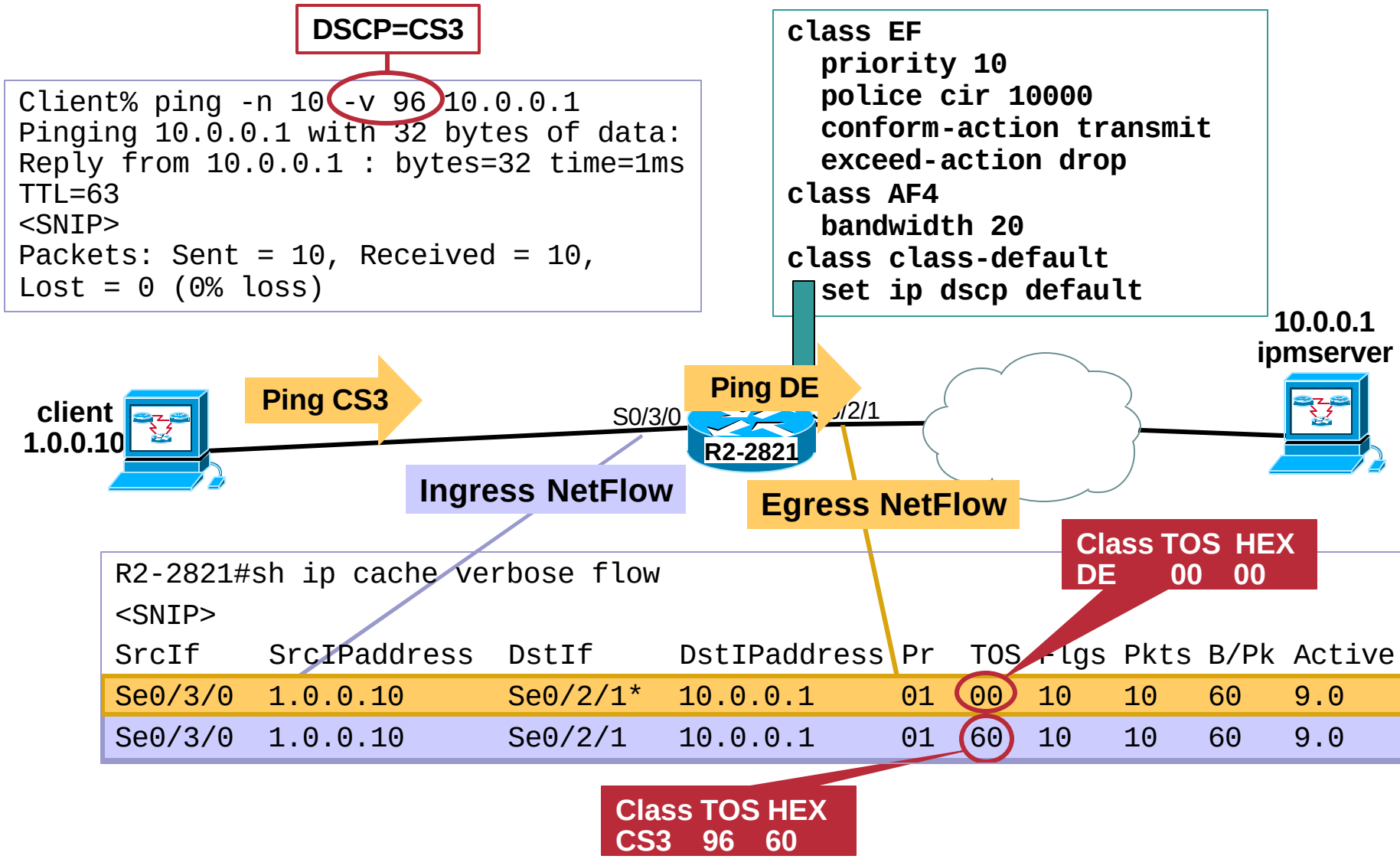
Cisco.com

NMS-3132 - Advanced NetFlow Usage

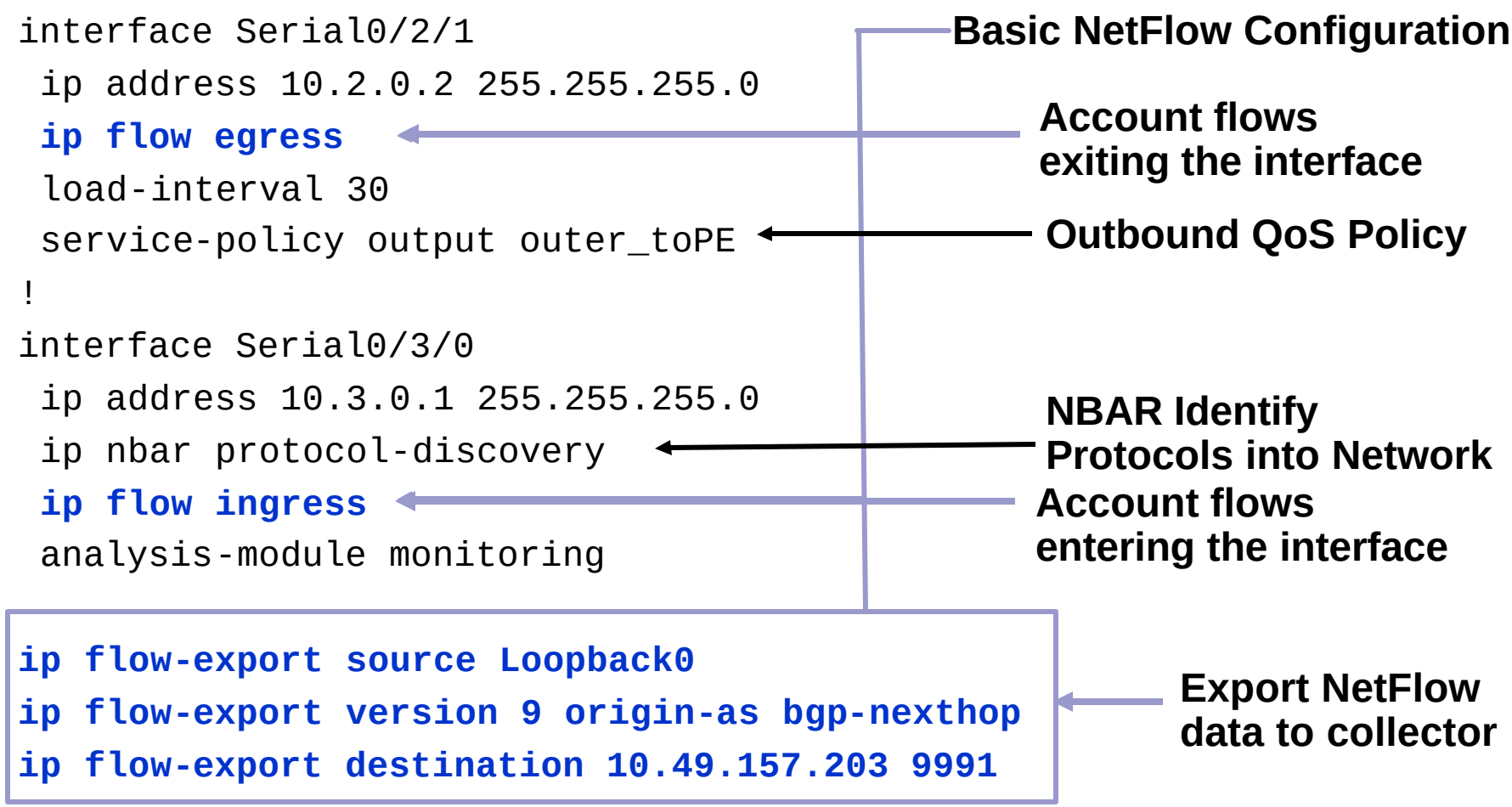
- **NetFlow**
  - ✓ **Inspects packets**
  - ✓ **Provides flow information Per Class of Service (TOS)**
  - ✓ **Provides pre and post policy QoS classification**
- **Flows are defined by 7 keys:**
  - **Source IP address**
  - **Destination IP address**
  - **Source port**
  - **Destination port**
  - **Layer 3 protocol**
  - **TOS byte (DSCP)**
  - **Input interface (ifIndex)**



Case Study 5: QoS [27]
NetFlow



Case Study 5: QoS[28] NetFlow: Configuration



Case Study 5: QoS [29]
NetFlow: Top Talkers

Configuration

```
ip flow-top-talkers
 top 20
 sort-by bytes
 cache-timeout 120
 match input-interface Serial0/3/0
```

Default Timeout = 5s

- **Increase to SNMP Polling Interval**
- **Increase to view via CLI**

```
nms% ping -n 10 -v 96 10.0.0.1
```

Pre and post QoS Classification

CLI

```
R2-2821#sh ip flow top-talkers verbose
SrcIf   SrcIPaddress  DstIf    DstIPaddress  Pr TOS Bytes Port Port B/Pk Active
Se0/3/0  10.10.10.1   Se0/2/1*  10.10.10.4   08 00  239K  0000 0000 20   119.7
Se0/3/0  10.10.10.1   Se0/2/1   10.10.10.4   08 00  239K  0000 0000 20   119.7
<SNIP>
Se0/3/0  1.0.0.10     Se0/2/1*  10.0.0.1     01 00  44K   0000 0800 60   746.0
Se0/3/0  1.0.0.10     Se0/2/1   10.0.0.1     01 60  44K   0000 0800 60   746.0
16 of 20 top talkers shown. 16 of 16 flows matched.
```

SNMP: NetFlow MIB

cnfTopFlowsSrcAddress	cnfTopFlowsDstAddress	cnfTopFlowsTOS	cnfTopFlowsBytes
.1 0A.0A.0A.01	.1 0A.0A.0A.04	.1 0	.1 364980
.2 0A.0A.0A.01	.2 0A.0A.0A.04	.2 0	.2 364980
.15 01.00.00.0A	.15 0A.00.00.01	.15 0	.15 48480
.16 01.00.00.0A	.16 0A.00.00.01	.16 96	.16 48480

Case Study 5: QoS [30]
NetFlow: Configuration Tips

- **When monitoring NetFlow via CLI**
  - **increase inactive timers**
- **When exporting NetFlow records to a NetFlow collector NFCv5)**
  - **Timers should not be longer than NFC collector aggregator interval**

Long Cache timers will skew NFC data

```
R2-2821#sh ip cache flow
<SNIP>
IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 1760 added
  141533 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
<SNIP>
```

- **Default Cache timers**
- **Easy to miss short flows via CLI**

- **Increases inactive flow timer**
- **All udp flows are inactive**

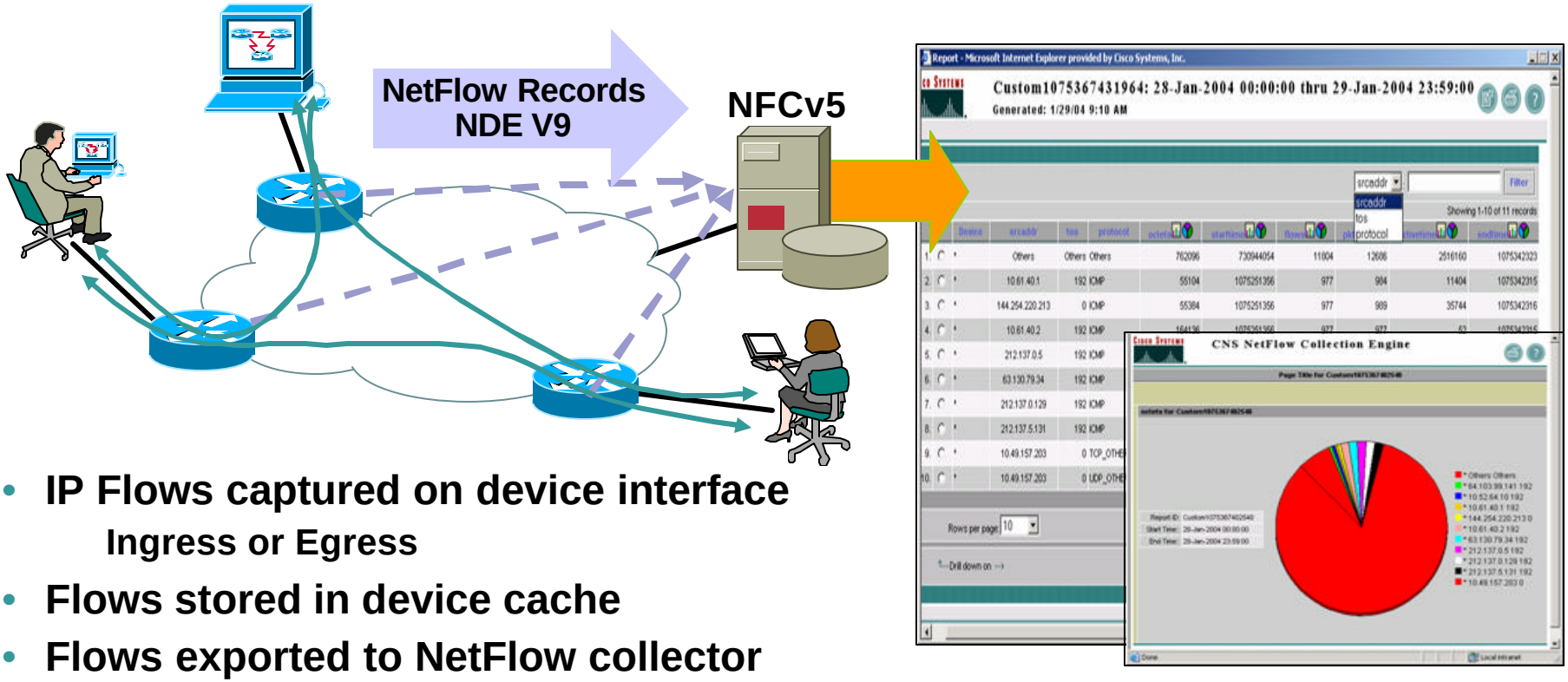
```
R2-2821(config)#ip flow-cache timeout inactive 120
```

Case Study 5: QoS [31]
NetFlow: TOS-Aggregations

- **NetFlow ToS-Aggregation schemes reduce**
  - **CLI: Show Cache flow output**
  - **NDE: data exported to the NefFlow Collector**
- **If Aggregation is used then**
  - **Only export aggregated data to the collector**
  - **Set timers in main cache to default**

```
R2-2821#sh ip cache flow aggregation ?
as-tos                   AS TOS aggregation cache
bgp-nexthop-tos          BGP nexthop TOS aggregation cache
destination-prefix-tos   Destination Prefix TOS aggregation cache
prefix-tos               Source/Destination Prefix TOS aggregation
                         cache
protocol-port-tos        Protocol, port, TOS aggregation cache
source-prefix-tos        Source Prefix TOS aggregation cache
```

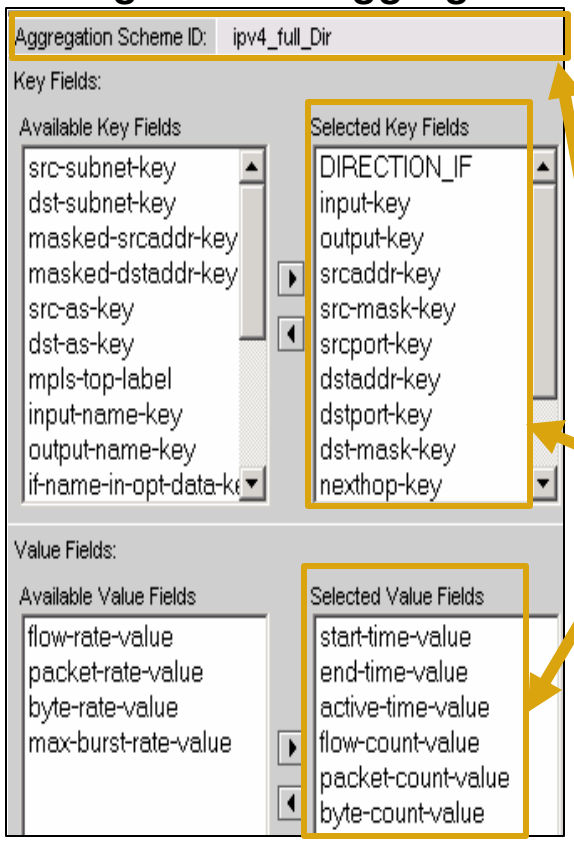
Case Study 5: QoS [32]
NetFlow: NetFlow Collector V5 (NFCv5)



- **IP Flows captured on device interface**
  - **Ingress or Egress**
- **Flows stored in device cache**
- **Flows exported to NetFlow collector**
  - **Cache timers default inactive=15s, active=30s**
- **NFCv5 Aggregation scheme**
  - **Listens on udp port**
  - **Aggregates flows over time interval & key flow fields**

Case Study 5: QoS [33]
NetFlow: NetFlow Collector (NFCv5)

Configuration->Aggregation Scheme

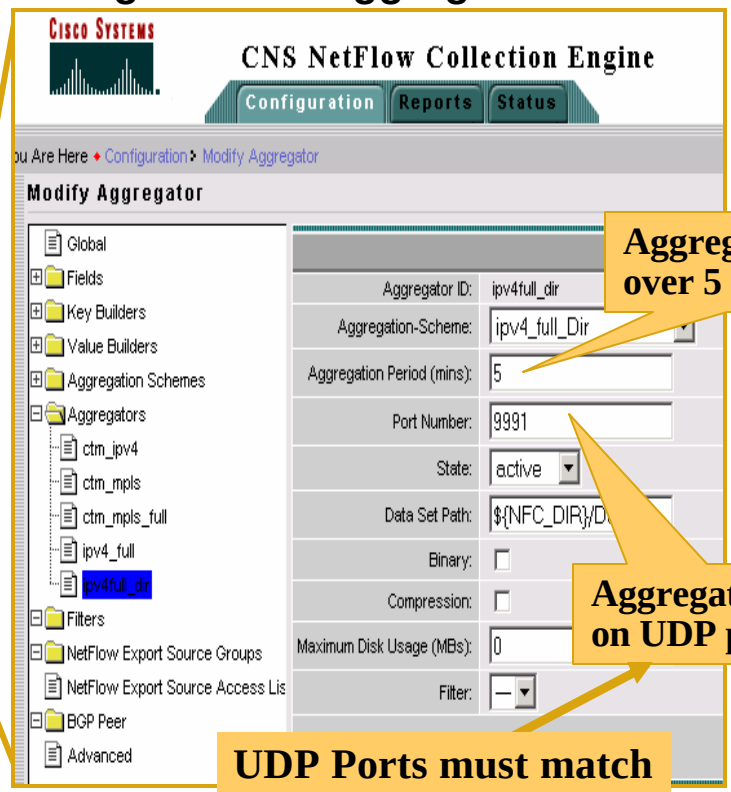


Aggregation Scheme

Key Flow Fields

Flow Value Fields

Configuration->Aggregator



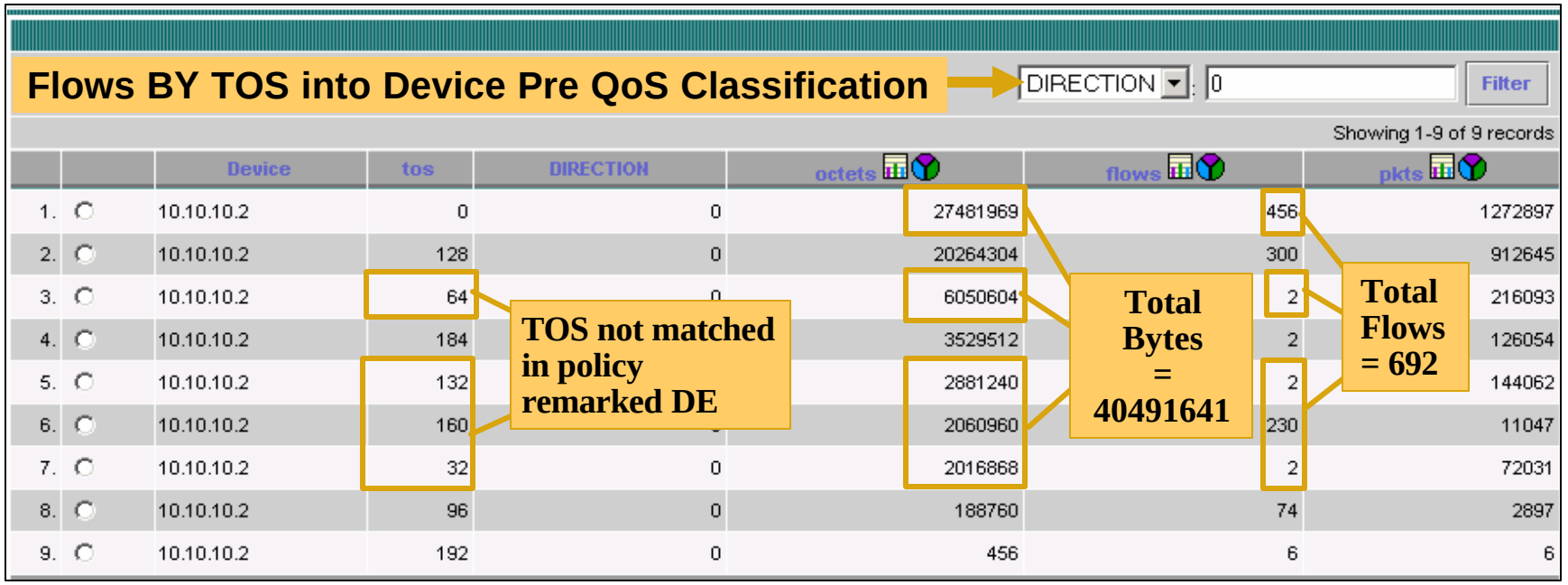
Aggregate flows over 5 minutes

Aggregate flows on UDP port 9991

UDP Ports must match

```
R2-2821(config)# ip flow-export destination 10.49.157.203 9991
```

Case Study 5: QoS [34]
NetFlow: NFCv5 QoS Reports



Flows BY TOS into Device Pre QoS Classification

DIRECTION : 0 Filter

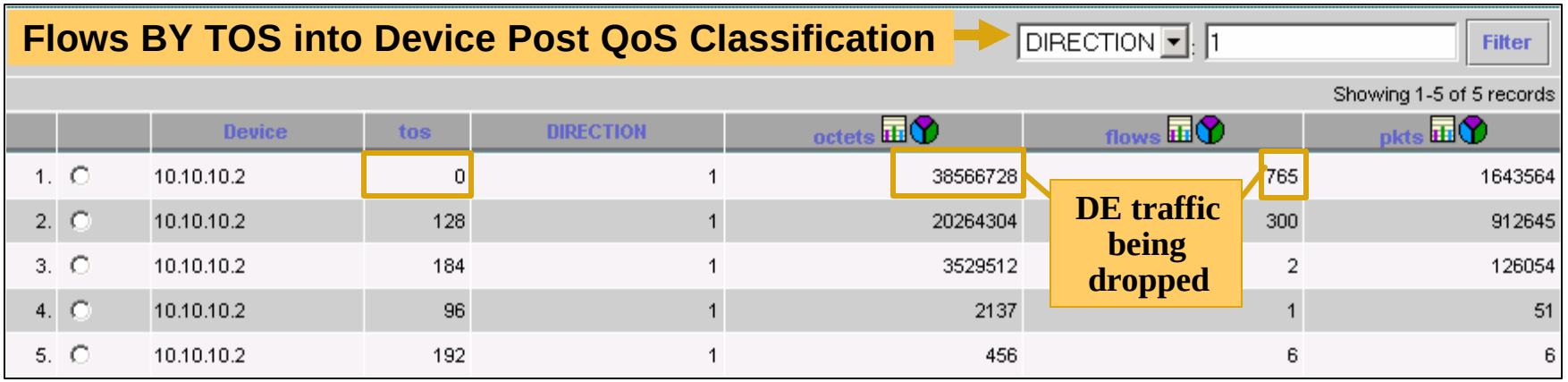
Showing 1-9 of 9 records

	Device	tos	DIRECTION	octets	flows	pkts
1.	10.10.10.2	0	0	27481969	456	1272897
2.	10.10.10.2	128	0	20264304	300	912645
3.	10.10.10.2	64	0	6050604	2	216093
4.	10.10.10.2	184	0	3529512	2	126054
5.	10.10.10.2	132	0	2881240	2	144062
6.	10.10.10.2	160	0	2060960	230	11047
7.	10.10.10.2	32	0	2016868	2	72031
8.	10.10.10.2	96	0	188760	74	2897
9.	10.10.10.2	192	0	456	6	6

TOS not matched in policy remarked DE

Total Bytes = 40491641

Total Flows = 692



Flows BY TOS into Device Post QoS Classification

DIRECTION : 1 Filter

Showing 1-5 of 5 records

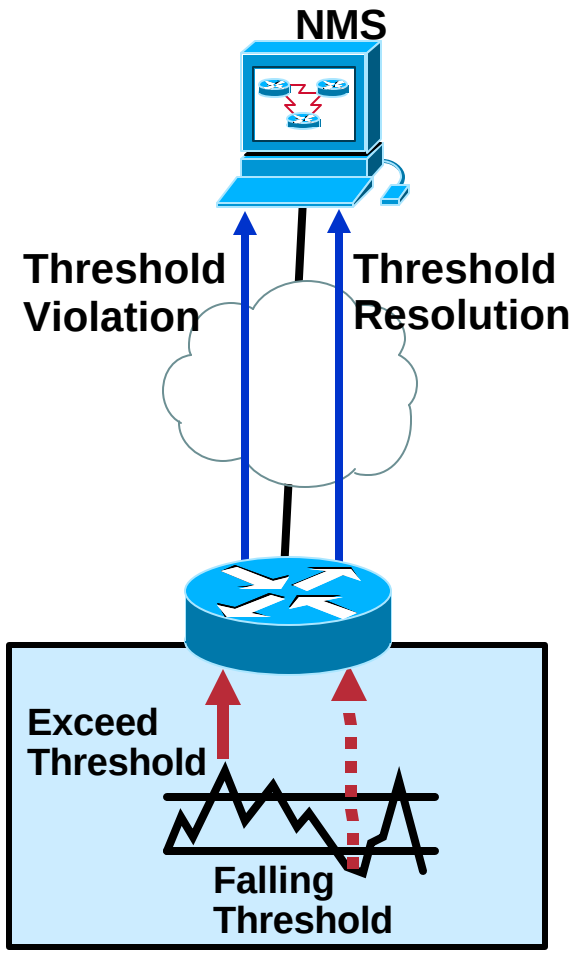
	Device	tos	DIRECTION	octets	flows	pkts
1.	10.10.10.2	0	1	38566728	765	1643564
2.	10.10.10.2	128	1	20264304	300	912645
3.	10.10.10.2	184	1	3529512	2	126054
4.	10.10.10.2	96	1	2137	1	51
5.	10.10.10.2	192	1	456	6	6

DE traffic being dropped

CASE STUDY 6:
Embedded Management

Case Study 6: Embedded Management [1]

- **What is Embedded Management ?**
  - ✓ On Device monitoring of Elements and network services
  - ✓ No external monitoring
  - ✓ Ability to send traps on the crossing of thresholds
  - ✓ Take action based on threshold crossed
- **Why Embedded Management is used ?**
  - ✓ Reduce network overhead
  - ✓ Scalable performance monitoring
  - ✓ High speed monitoring
- **Cisco's Embedded Management**
  - ➢ COOL, IPSLA (Service/Availability)
  - ➢ RMON, Embedded Event Manager (Any MIB object)



Case Study 6: Embedded Management [2] RMON: CPU Utilization

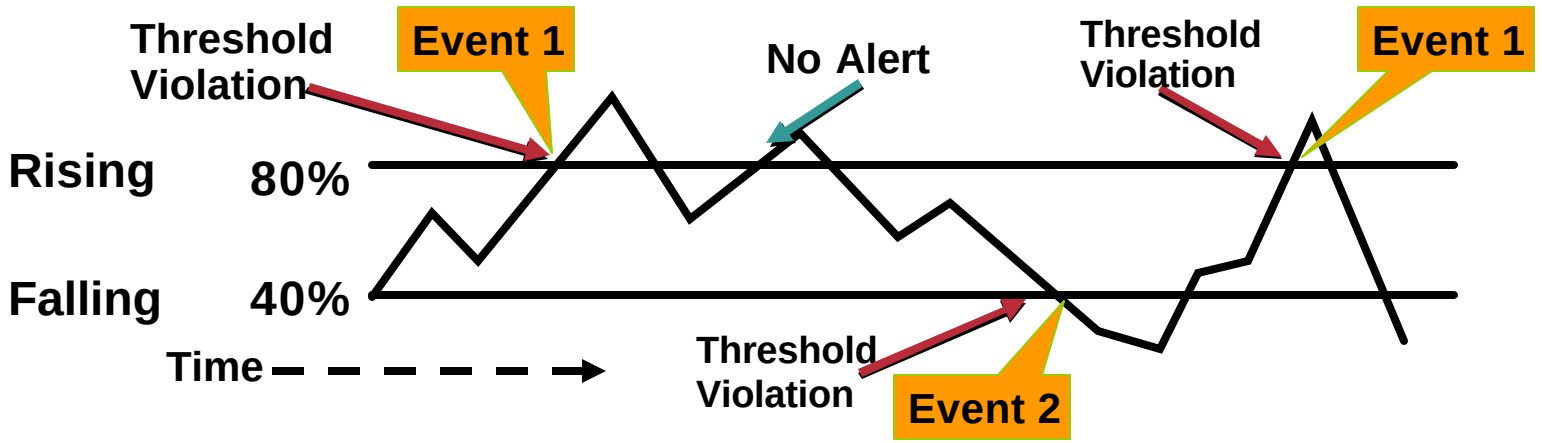
- **Configure RMON to generate a trap if CPU utilization reaches or exceeds 80%, and rearm the trap if utilization drops to 40% or less, sampling interval is 20 seconds**

Rising Condition **Falling Condition**

```
Router(config)#rmon alarm 1 cpmCPUTotalEntry.3.0 20 absolute
rising-threshold 80 1 falling-threshold 40 2 owner me

Router(config)#rmon event 1 log trap public description "cpu
busy" owner me

Router(config)#rmon event 2 log description "cpu not too busy"
```



Case Study 6: Embedded Management [3]
EEM: CPU Utilization

- **The following configuration example generates a syslog message to be logged when**
  - **CPU utilization goes above 90%**
  - **Event monitoring will resume when the CPU utilization falls below 80% or when 20 minutes have passed**

```
Event Manager applet totcpualert
event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.6.1 get-type
exact entry-op ge entry-val 90 exit-comb or exit-op le
exit-val 80 exit-time-sec 1200
action syslog priority alert msg *** Warning*** CPU
utilization is above 90%
```

- **More flexible than RMON event and alarm**

Summary and Conclusion

- **Starting point should be monitoring the Service**
  - **Identify what the service is.**
  - **Where Service Information is available.**
- **Quality of service is dependant on underlying infrastructure**
  - **QoS is key to delivery of multiple services over a network.**
- **Deployed device instrumentation is preferred**
  - **Deployed in all parts of the network & already managed.**
- **The performance monitoring cycle is an ongoing process**
  - **Define service, Baseline, Collect data Process.**
  - **Needs to be constantly tuned**

Complete Your Online Session Evaluation!

Cisco.com

Por favor, complete el formulario de evaluación.

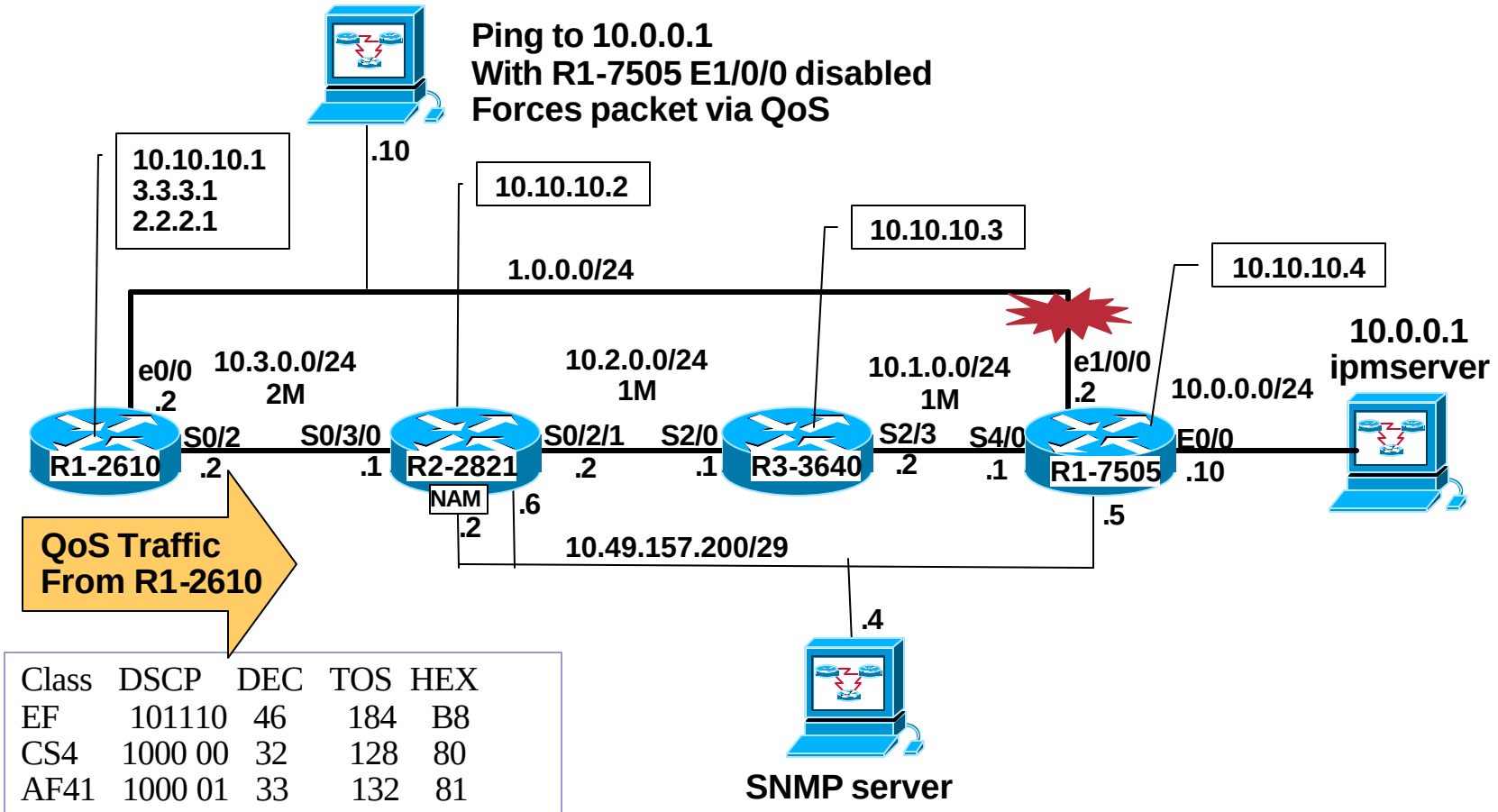
Muchas gracias.

Session ID: NMS-2042

Performance Management with Cisco Devices.

Questions?

Appendix A:
Reference Network for Case Studies



Class	DSCP	DEC	TOS	HEX
EF	101110	46	184	B8
CS4	1000 00	32	128	80
AF41	1000 01	33	132	81
AF3	0110 00	16	96	60
AF2	0100 00	8	64	40
DE	0000 00	0	00	00

Appendix B
CASE STUDY 7:
Device Elements

Case Study 7: Device Elements [1]
CPU Utilization: CLI Commands

- **Routers running constantly at high utilization level can affect the overall performance of forwarding and processing packets**

```
Router#sh proc cpu
CPU utilization for five seconds: 50%/20%; one minute:5%; five minutes:2%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0         1          0  0.00%  0.00%  0.00%   0 Chunk Mgr
   2         368    274108          1  0.00%  0.00%  0.00%   0 Load Meter
   3       32940    716632         45  0.00%  0.00%  0.00%   0 OSPF Hello
Router#sh proc cpu sorted ?
  1min  Sort based on 1 minute utilization
  5min  Sort based on 5 minutes utilization
  5sec  Sort based on 5 seconds utilization
  |     Output modifiers
```

Case Study 7: Device Elements [2]
CPU Utilization: SNMP

- CPU utilization using OLD-CISCO-CPU MIB
- Supported since 10.2
- As of 12.0 all OLD-CISCO-*-MIBs are “deprecated”

busyPer avgBusy1 avgBusy5

```
Router#sh proc cpu
CPU utilization for five seconds: 50%/20%; one minute:5%; five minutes: 2%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0         1          0  0.00%  0.00%  0.00%   0 Chunk Mgr
   2         368    274108          1  0.00%  0.00%  0.00%   0 Load Meter
   3       32940    716632         45  0.00%  0.00%  0.00%   0 OSPF Hello
```

- OLD-CISCO-CPU MIB only applies to RP CPU
- OLD-CISCO-CPU MIB doesn’t apply to CPU utilization for VIP cards (7500) or LC (GSR)

Case Study 7: Device Elements [3]
CPU Utilization: SNMP

- **CISCO-PROCESS-MIB: New MIB introduced in 12.0T train; provides information on CPU utilization and running processes**

```
Router#sh proc cpu
CPU utilization for five seconds: 50%/20%; one minute:5%; five minutes: 2%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0         1          0  0.00%  0.00%  0.00%   0 Chunk Mgr
   2         368    274108          1  0.00%  0.00%  0.00%   0 Load Meter
   3       32940    716632         45  0.00%  0.00%  0.00%   0 OSPF Hello
```

cpmCPUTotal5sec

cpmCPUTotal1min

cpmCPUTotal5min

- **Solution for VIP cards(7500) and LC(GSR):**

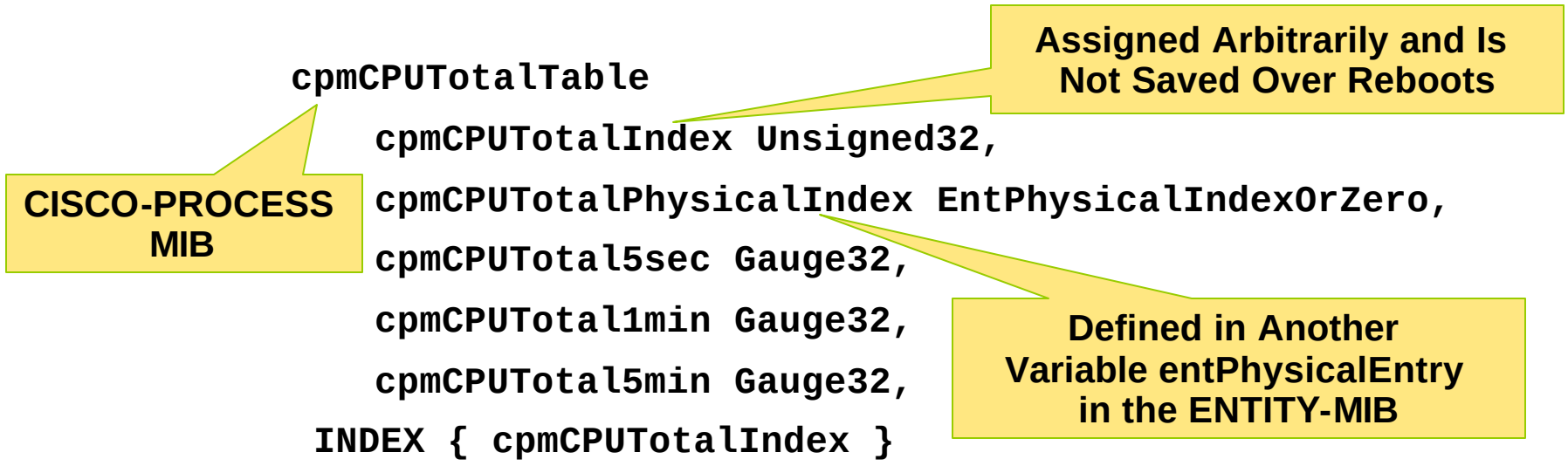
  CISCO-PROCESS-MIB + ENTITY-MIB

cpmProcessEntry

cpmProcessExtRevEntry

Case Study 7: Device Elements [4]
CPU Utilization: SNMP for VIP and LC

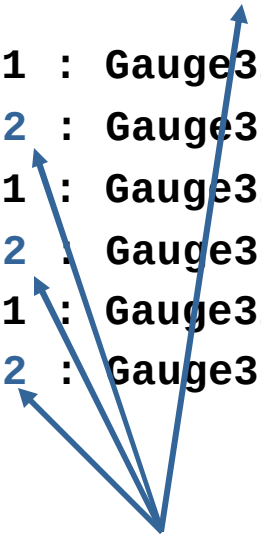
- The ENTITY-MIB provides an inventory of the chassis, cpu card(s), line cards, fans, power supplies etc.; this MIB is the industry-standard replacement to the OLD-CISCO-CHASSIS-MIB
- Which MIB variables to use for VIP and LC?



Case Study 7: Device Elements [5]: CPU Utilization: VIP & ENTITY & CISCO-PROCESS MIBs

- **CISCO-PROCESS-MIB:**

```
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotalPhysicalIndex.1 :
  INTEGER: 0
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotalPhysicalIndex.2 :
  INTEGER: 28
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal5sec.1 : Gauge32: 12
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal5sec.2 : Gauge32: 9
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal1min.1 : Gauge32: 10
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal1min.2 : Gauge32: 5
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal5min.1 : Gauge32: 8
cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPUTotal5min.2 : Gauge32: 4
```



- **There are 2 CPUs displayed**
- **The second CPU is a VIP identified by an index number of 2**

Case Study 7: Device Elements [7]
CPU Utilization: VIP & ENTITY & CISCO-PROCESS MIBs

- **ENTITY-MIB:**

```
entPhysicalDescr.28 : OCTET STRING- (ascii):Versatile Interface Processor (VIP2-50)
entPhysicalVendorType.28 : OBJECT IDENTIFIER:
        .iso.org.dod.internet.private.enterprises.cisco.ciscoModules.3.1.9.7.26
entPhysicalContainedIn.28 : INTEGER: 8
entPhysicalClass.28 : INTEGER: module
entPhysicalParentRelPos.28 : INTEGER: 6
entPhysicalName.28 : OCTET STRING- (ascii):
entPhysicalHardwareRev.28 : OCTET STRING- (ascii):   2.00
entPhysicalFirmwareRev.28 : OCTET STRING- (ascii):
entPhysicalSoftwareRev.28 : OCTET STRING- (ascii):   22.20
entPhysicalSerialNum.28 : OCTET STRING- (ascii):     13944617
entPhysicalMfgName.28 : OCTET STRING- (ascii):       CISCO
entPhysicalTable.entPhysicalEntry.entPhysicalModelName.28 : OCTET STRING- (ascii):VIP2-50
entPhysicalTable.entPhysicalEntry.entPhysicalAlias.28 : OCTET STRING- (ascii):
entPhysicalTable.entPhysicalEntry.entPhysicalAssetID.28 : OCTET STRING- (ascii):
entPhysicalTable.entPhysicalEntry.entPhysicalIsFRU.28 : INTEGER: true
```

Case Study 7: Device Elements [8] Memory Utilization: CLI Commands

- The amount of main memory left on a router's processor has significant impact on performance

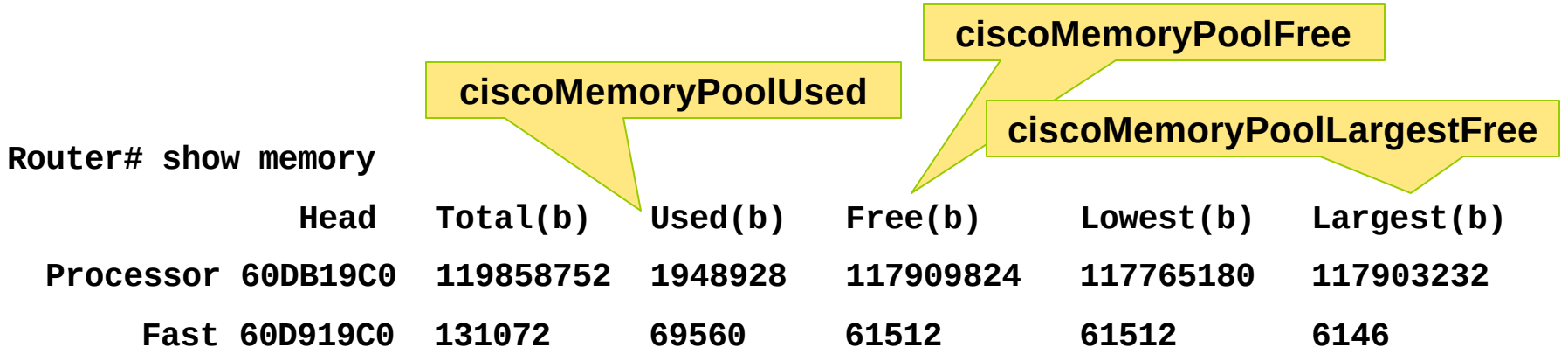
```
Router# show memory
                Head   Total(b)   Used(b)   Free(b)   Lowest(b)   Largest(b)
  Processor 60DB19C0   119858752  1948928   117909824  117765180   117903232
       Fast 60D919C0   131072     69560     61512      61512       6146 …
```

- Buffers are allocated from memory into different memory pools

```
Router# show buffers
  Buffer elements:
       499 in free list (500 max allowed)
       124485689 hits, 0 misses, 0 created
  Public buffer pools:
  Small buffers, 104 bytes (total 120, permanent 120:
  112 in free list (20 min, 250 max allowed)…
```

Case Study 7: Device Elements [9] Memory Utilization: SNMP

- Initially OLD-CISCO-MEMORY MIB
- CISCO-MEMPOOL-MIB only supports RP



- Solution for VIP cards(7500) and LC(GSR): CISCO-ENHANCED-MEMPOOL MIB
- CISCO-ENHANCED-MEMPOOL MIB is supported from 12.0(21)S for VIP cards and 12.0(20)ST for GSR LCs

Case Study 7: Device Elements [10] Memory Utilization in VIP and LC: SNMP

- **CISCO-ENHANCED-MEMPOOL MIB**

```
"A table of memory pool monitoring entries for all physical
  entities on a managed system.“
cempMemPoolTable
        cempMemPoolIndex            CempMemPoolIndex,
        cempMemPoolType             CempMemPoolTypes,
        cempMemPoolName             SnmpAdminString,
        cempMemPoolPlatformMemory   AutonomousType,
        cempMemPoolAlternate        CempMemPoolIndexOrNone,
        cempMemPoolValid            TruthValue,
        cempMemPoolUsed             Gauge32,
        cempMemPoolFree             Gauge32,
        cempMemPoolLargestFree      Gauge32,
        cempMemPoolLowestFree       Gauge32
 INDEX    { entPhysicalIndex, cempMemPoolIndex }
```

Case Study 7: Device Elements [11] Bandwidth Utilization: CLI Commands


```
Router# show interfaces serial 0/1
Serial0/1 is up, line protocol is up
Internet address is 192.1.1.105/30
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load
  1/255
    …
5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     345817 packets input, 27998005 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     1 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 1 abort
     277596 packets output, 20203129 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     3 carrier transitions
     RTS up, CTS up, DTR up, DCD up, DSR up
```

Default value

Case Study 7: Device Elements [12] Bandwidth Utilization—SNMP

- ifInOctets—Total number of octets received on the interface, including framing characters
- ifOutOctets—Total number of octets transmitted out of the interface, including framing characters
- ifSpeed—An estimate of the interface's current bandwidth in bits per second; for interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth

Input Util = [(delta (ifInOctets)) * 8 *100] / [(number of seconds in delta) * ifSpeed]

Output Util = [delta(ifOutOctets)) * 8 *100] / [(number of seconds in delta) * ifSpeed]

Case Study 7: Device Elements [13]
Device Packet Loss: SNMP

- **OLD-CISCO-INTERFACES-MIB**

  locIfInputQueueDrops—OID .1.3.6.1.4.1.9.2.2.1.1.26

  “The number of packets dropped because the input queue was full.”

- **IF-MIB**

  ifInDiscards—OID .1.3.6.1.2.1.2.2.1.13

  "The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol; one possible reason for discarding such a packet could be to free up buffer space

  Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime”

From “`show interface`” command

```
Output queue 0/40, 0 drops; input queue 0/75, 0 drops, …
```

locIfOutQueueDrops/ ifOutDiscards

locIfInQueueDrops/ ifInDiscards