

Teste07

Aula 17/12/2021

Acondicionamento do DiffServ

Per-Hop Behavior

Delay

Jitter

Configurar o DiffServ

Desvantagens do DiffServ

Aula 07/01/2022

Políticas de admissão

Metáfora Token Bucket

Filas de espera de ingresso

Filas de espera de egresso (que sai)

Imagem importante do QoS (pag 224)

Gestão de buffers de egresso

Como é que funciona?

Em suma

Aula 17/12/2021

- No protocolo IP existe um campo chamado **Type Of Service** para o DiffServ
- Para o IPv6 existe o **Traffic Class**
 - No IPv6 existe o *Flow Label* que permite apanhar um pacote e ver a que fluxo pertence

Acondicionamento do DiffServ

Per-Hop Behavior

- **Default**
 - É o único comportamento cuja implementação é obrigatória
 - Absorve todo o tráfego que não se enquadra noutra classe
 - Serviço *Best Effort*

- **Expedited Forwarding (EF)**

- Serve tráfego com requisitos de tempo real: voz e video
- Latência, perdas e jitter reduzidos

- **Assured Forwarding (AF)**

- A entrega é assegurada se não forem excedidos débitos subscritos
- Aqui podem haver 4 classes com tratamentos diferentes

Assured Forwarding (AF) Behavior Group

	Class 1	Class 2	Class 3	Class 4
Low Drop	AF11 (DSCP 10)	AF21 (DSCP 18)	AF31 (DSCP 26)	AF41 (DSCP 34)
Med Drop	AF12 (DSCP 12)	AF22 (DSCP 20)	AF32 (DSCP 28)	AF42 (DSCP 36)
High Drop	AF13 (DSCP 14)	AF23 (DSCP 22)	AF33 (DSCP 30)	AF43 (DSCP 38)

- **Class Selector**

- Solução que o IETF propoe para manter compatibilidade com o tratamento do campo de 3 bits que antes se denominava *Precedende*

- Em suma

Name	DSCP (Decimal)	DSCP (Bits)	
Default	0	000000	→ Default (Best effort)
AF11	10	001010	Assured Forwarding
AF12	12	001100	
AF13	14	001110	
AF21	18	010010	
AF22	20	010100	
AF23	22	010110	
AF31	26	011010	
AF32	28	011100	
AF33	30	011110	
AF41	34	100010	
AF42	36	100100	
AF43	38	100110	
EF	46	101110	→ Expedited Forwarding

- Mapeamento da marcação L2(VLAN PCP) <-> L3(DSCP)

- Podemos reescrever a tabela de marcação, tanto de L2 para L3 como ao contrário

CoS Value	DSCP Value		DSCP Values	CoS Value
0	0	Equivalente ao mapeamento directo CoS-(antigo)IP Precedence	0	0
1	8		8 and 10	1
2	16		16 and 18	2
3	24		24 and 26	3
4	32		32 and 34	4
5	40		40 and 46	5
6	48		48	6
7	56		56	7

Diagram illustrating the mapping between CoS (Class of Service) and DSCP (Differentiated Services Code Point) values. The left table shows the direct mapping from CoS (0-7) to DSCP (0-56). The right table shows the mapping from DSCP values back to CoS values. A note indicates that the DSCP values 48 and 56 are equivalent to the old IP Precedence CoS values.

Delay

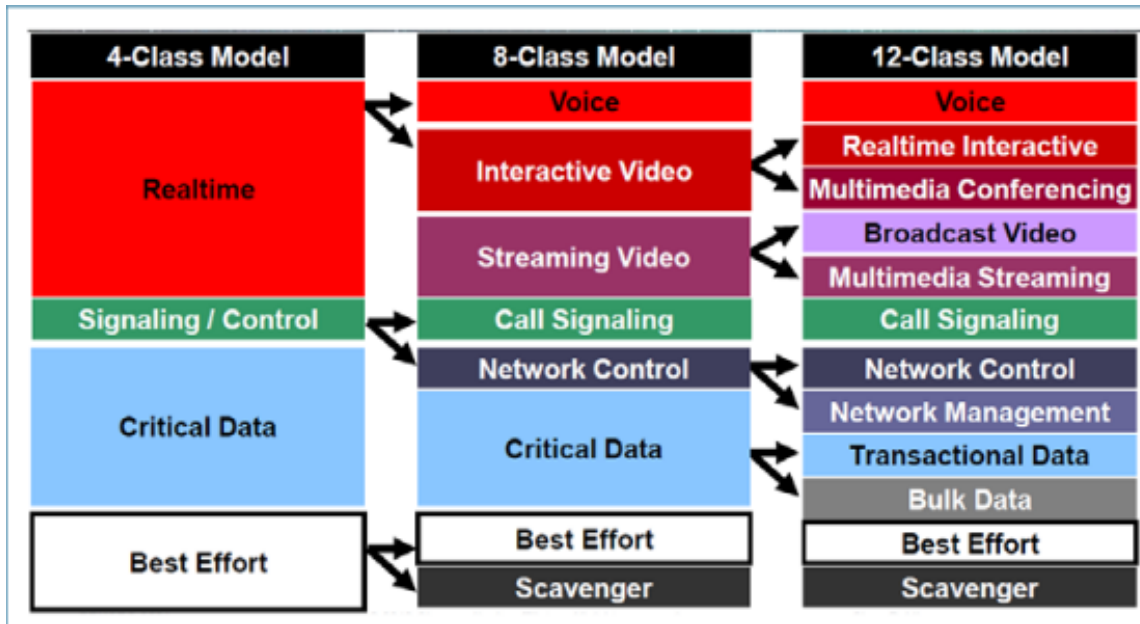
- Tempo perdido do envio até à transmissão
- O delay é das coisas mais complicadas de combater

Jitter

- Diferença de atraso que um pacote tem em relação ao seguinte

Configurar o DiffServ

- O melhor para configurar o DiffServ é começarmos com poucas classes e irmos testando a rede
- Depois, consoante os requisitos da rede, vamos aumentando a complexidade aumentando assim o nº de classes

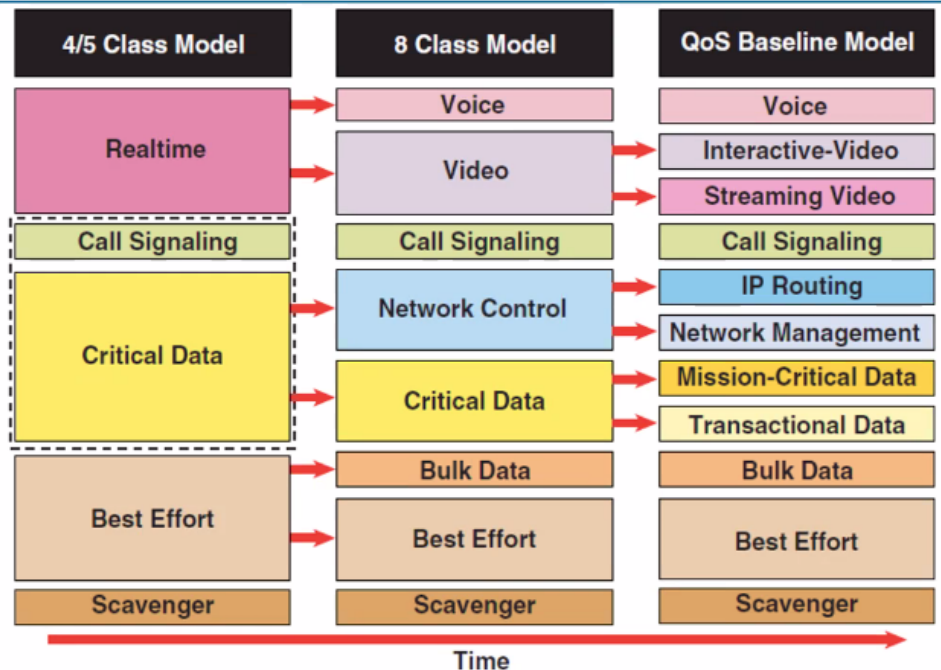


Desvantagens do DiffServ

- Não permite às aplicações saberem se estão a receber da rede a QoS desejável
- Modelo que funciona bem com transferências curtas (web browsing) mas que oferece um desempenho pobre em aplicações de tempo real
- Pensado para redes relativamente estáticas
- Difícil de redistribuir recursos pelas classes geridas
- **Garantias de QoS associadas a uma classe não se traduzem necessariamente em garantias dadas a um fluxo (os fluxos de uma mesma classe poderão interferir entre si)**

Aula 07/01/2022

- Aplicação incremental



- O DiffServ só funciona se o tráfego não for demais, ou seja, se limitarmos o tráfego que entra na nossa rede e isso é feito na camada de acesso
- Nos podemos trabalhar no trafego que entra para limitar a quantidade de trafego que nos entra numa determinada interface
- Mas aqui na cadeira temos de nos preocupar com o tráfego que sai e não o que entra no entanto é obvio que se deixamos muito tráfego entrar o caos instala-se

Políticas de admissão

- Quando o tráfego em excesso atinge o limite definido para o *burst* o tráfego é descartado ou remarcado

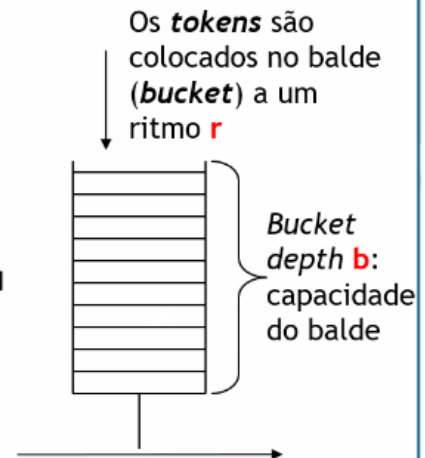
Metáfora Token Bucket

- Ha um contador na porta de entrada e esse contador esta sempre a entrar
 - Cada vez que entra trafego o contador incrementa
 - Cada vez que sai trafego o contador decrementa
 - Se ele passar o limite que nós definimos, o que entra é descartado
 - Ou seja, nós definimos o limite e definimos o nivel de escoamento

Políticas de admissão (*Policing*)

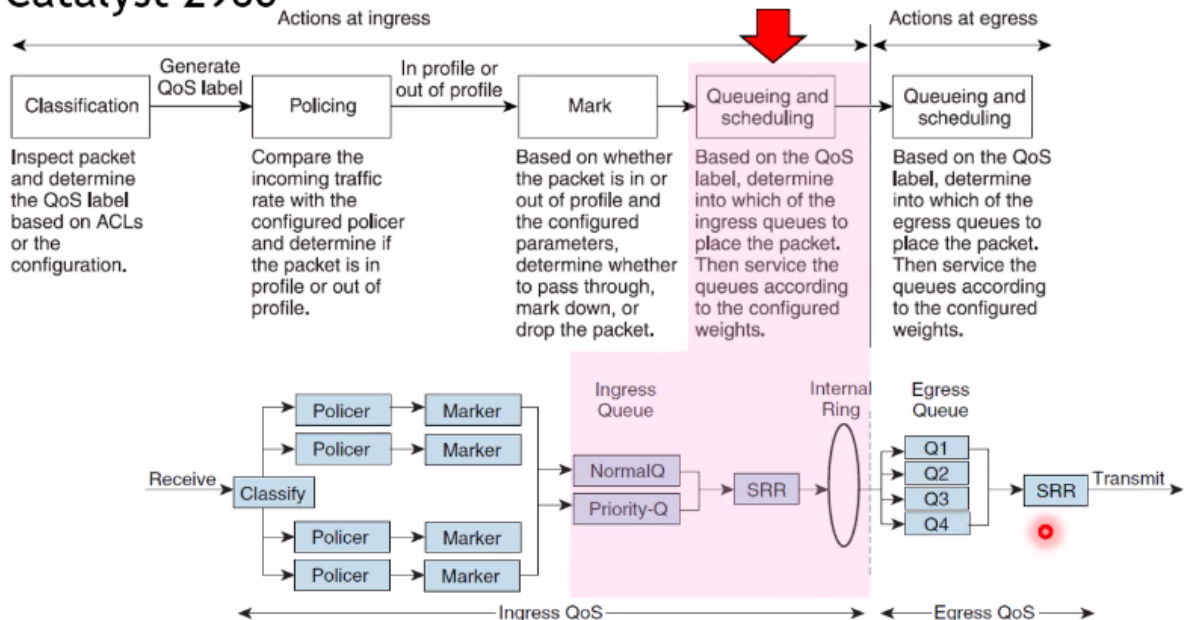
• Metáfora *Token Bucket*

- Se o balde ficar cheio, os créditos (*tokens*) que vão chegando transbordam (i.e. não são acumulados para benefício posterior).
- O envio de um pacote de P bytes pressupõe o consumo de $P \cdot 8$ *tokens*.
- Quando chega um pacote que necessita de mais *tokens* que os disponíveis toma-se uma de duas medidas:
 - Policing: o pacote é descartado ou marcado para tratamento diferenciado posteriormente
 - Shaping: o pacote é *buffered*
 - » Abordado mais tarde



Filas de espera de ingresso

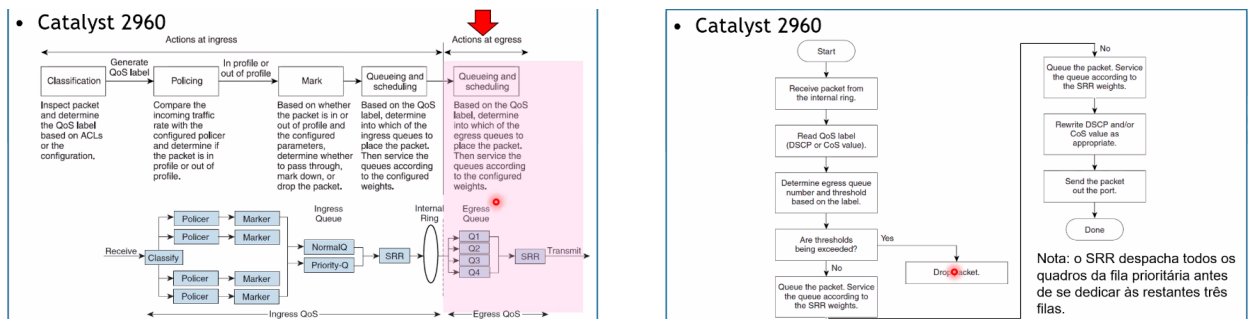
• Catalyst 2960



- A fila prioritária recebe uma fração **garantida** da largura de banda da *stack ring*, facto que assegura um atraso limitado na presença de congestão
- O resto da largura de banda é repartida por ambas as filas de espera na proporção configurada
- A politica usada para servir as filas de espera de ingresso é chamada de **Shaped Round Robin**
 - Sobre as filas de ingresso o escalonamento opera apenas em modo *shared*

Filas de espera de egresso (que sai)

- Q (Queue)
- T (Threshold)
- P (Priority)
- 4Q1T (4 Queue, 1 Threshold)
- 1P3Q1T (1 Priority, 3 Queue, 1 Threshold)



- Para eu ter mais formas diferentes de tratar o tráfego, posso criar limiares(threshold) para que tipos de tráfegos diferentes entrem para a mesma fila

- Escalonador de saída

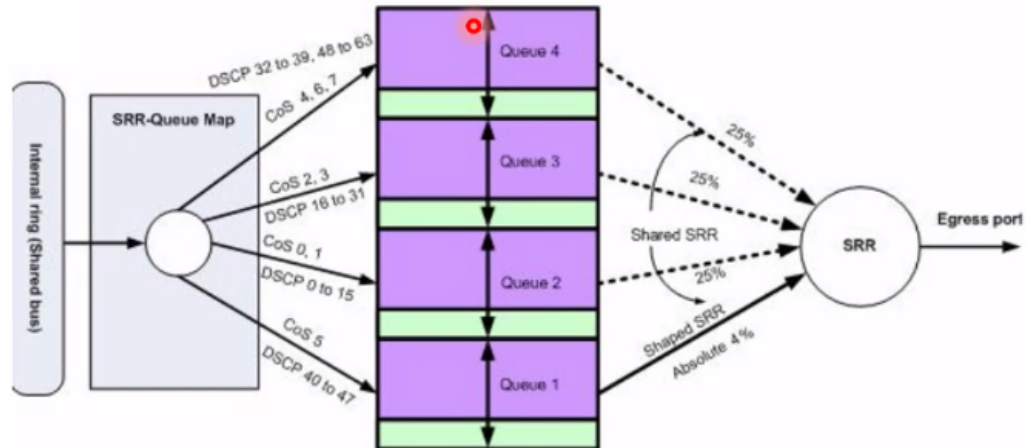


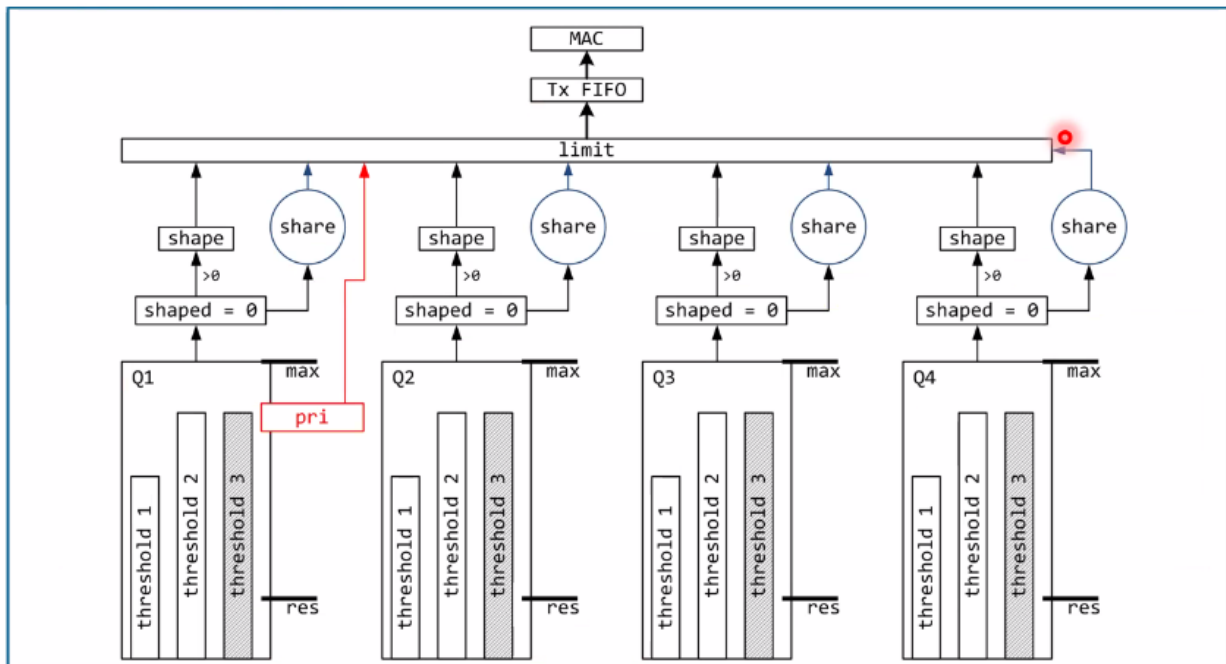
Imagem importante do QoS (pag 224)

- Vamos trabalhar no modelo 1P3Q3T
 - 1 fila prioritaria (é sempre a 1)

Filas de espera de egresso

Modelos de fila por interface:

- 4Q3T
- 1P3Q3T



- Podemos dizer que a fila 1 é prioritária e aqui estamos a trabalhar no **modelo 1P3Q3T**
 - Se dissermos isto, este tráfego passa sempre à frente de todos
- Se dissermos que estamos a trabalhar sem filas prioritária estamos a usar o **modelo 4Q3T**
- No shaped
 - Podemos configurar
- No share
 - Podemos configurar

Gestão de buffers de egresso

- O switch reserva um mínimo de buffers para cada fila para evitar que uma fila ou interface prive as restantes de espaço

Como é que funciona?

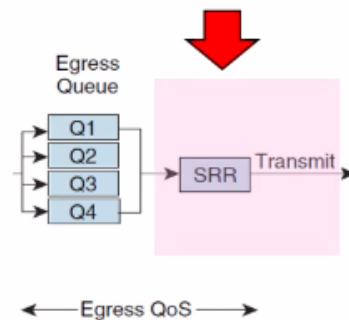
- As filas de egresso podem ser atendidas em modo *shared* ou em modo *shaped*

- Modo *shared* (disponível no ingresso e egresso)

- Se a fila mais prioritária estiver vazia as filas menos prioritárias recebem a largura de banda excedente.
- Optimiza a utilização da *bandwidth* disponível

- Modo *shaped* (apenas disponível no egresso)

- É reservada uma largura de banda estrita (10% a 90%) e apenas esta é dada mesmo quando está disponível um valor superior.
- Modo que estabelece um *hard limit* específico
- Suaviza o fluxo, evitando picos e vales cavados



- Catalyst 2960

- Interpretação dos valores por omissão das filas de egresso

```
srr-queue bandwidth shape 25 0 0 0
srr-queue bandwidth share 25 25 25 25
```

- *Shape* (configuração toma precedência)

- Q1: $1/25 = 4\%$ BW
- Q2, Q3, Q4: Partilham os 96% BW sobranes em modo *shared*

- *Share*

- Q2: $25/(25+25+25) = 33\%$ (96% BW) $\geq 32\%$ BW
- Q3: $25/(25+25+25) = 33\%$ (96% BW) $\geq 32\%$ BW
- Q4: $25/(25+25+25) = 33\%$ (96% BW) $\geq 32\%$ BW

“ \geq ” Porque é o modo *shared*

- Nota: esta interpretação assume que a fila Q1 não foi configurada como prioritária. Sendo prioritária o SRR ignora estes valores para a mesma e serve-a exclusivamente enquanto tiver tráfego.

Em suma

- O tráfego é marcado à entrada do switch
- Em função do possível *policing* o excedente é então descartado ou remarcado
- Em função das condições de congestão da *switch fabric* pode ou não passar pelas filas de ingresso da interface de entrada

- Uma réplica é então comutada para determinada interface de saída em função do valor e tipo de endereço
- Em função da classificação do quadro é escolhida a fila de egresso
- Em função da classificação do quadro é escolhido um dos três threshold limits e o quadro entra na fila ou é linearmente descartado
- Em função da parametrização das quatro filas o quadro é enviado para o meio físico em determinado instante