Università degli Studi di Bergamo

DIPARTIMENTO DI INGEGNERIA E SCIENZE APPLICATE

# RETI INTERNET MULTIMEDIALI

## Internet Quality of Service

*Il documento è adattato da materiale cortesemente messo a disposizione dai Prof. Stefano Paris, Antonio Capone, Flaminio Borgonovo, Paolo Giacomazzi e Vittorio Trecordi*

# IL PROBLEMA DELLA QoS

# Cosa è la QoS

- La QoS (Quality of Service) è un *indice di qualità* che misura il livello di servizio rispetto alle attese dell'utente
- La QoS è associata ai servizi di rete e ai rispettivi flussi informativi ed è influenzata da
  - Banda disponibile (compressione, rate di trasmissione)
  - Ritardi
  - Jitter dei ritardi
  - Packet dropping
  - Blocking probability
  - Set-up delay
  - …

# Cosa è la QoS

- La QoS è legata ma si distingue dalla QoE (Quality of Experience)
  - La QoE è un indice che misura in termini *soggettivi* il valore del servizio offerto all'utente
  - La QoS è invece una misura *oggettiva* della qualità offerta all'utente, sulla base di parametri misurabili

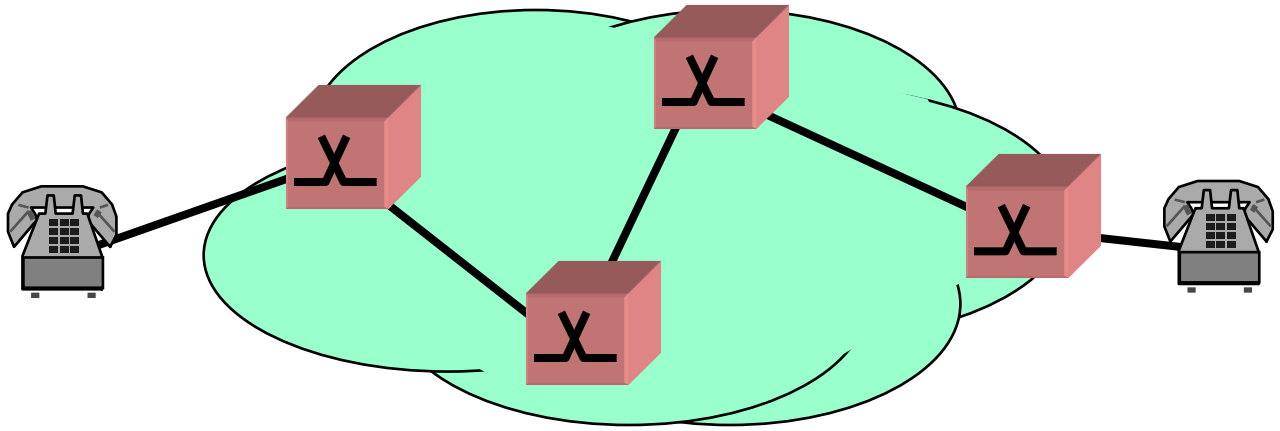
- E' possibile definire la QoS in modo
  - *Assoluto*: definizione di valori che devono essere rispettati da un insieme di parametri prestazionali
  - *Relativo*: definizione della modalità di trattamento di una classe di traffico rispetto alle altre

# QoS in Internet

- Il funzionamento di Internet e delle reti IP è basato sulla modalità **best-effort**
  - Nessuna garanzia di consegna, e quindi di qualità
- Il successo della modalità best-effort è dovuto alla sua semplicità
- Attualmente, però, la richiesta di qualità nelle reti IP sta crescendo a dismisura e sono state sviluppate nuove soluzioni per garantire la qualità
  - IP MPLS, IP diffserv, IP intserv, …

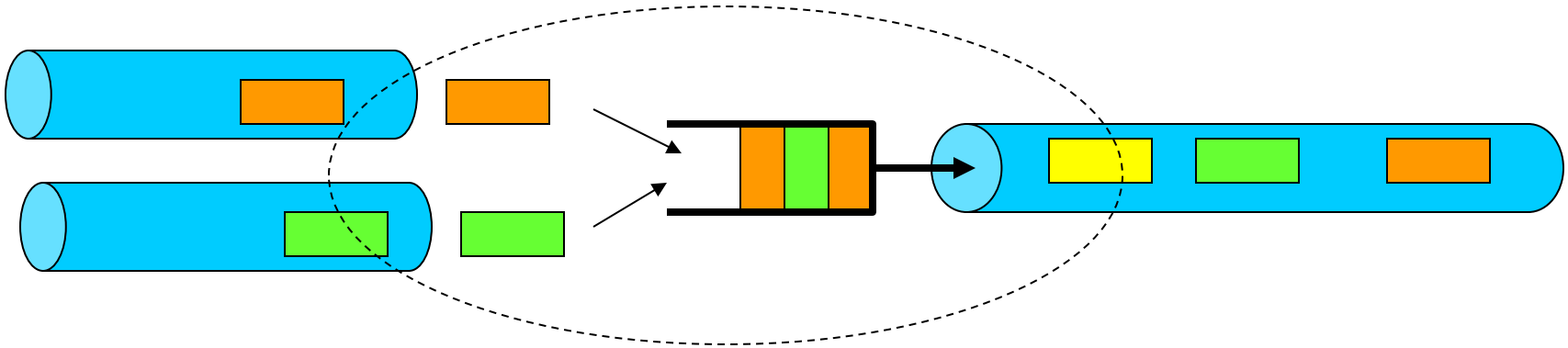
# La Commutazione di Circuito

- Ritaglia un circuito a banda costante (es. 64 kb/s) end-to-end
- Ogni bit è trasferito con la stessa velocità (servizio sincrono)
- Nessun jitter

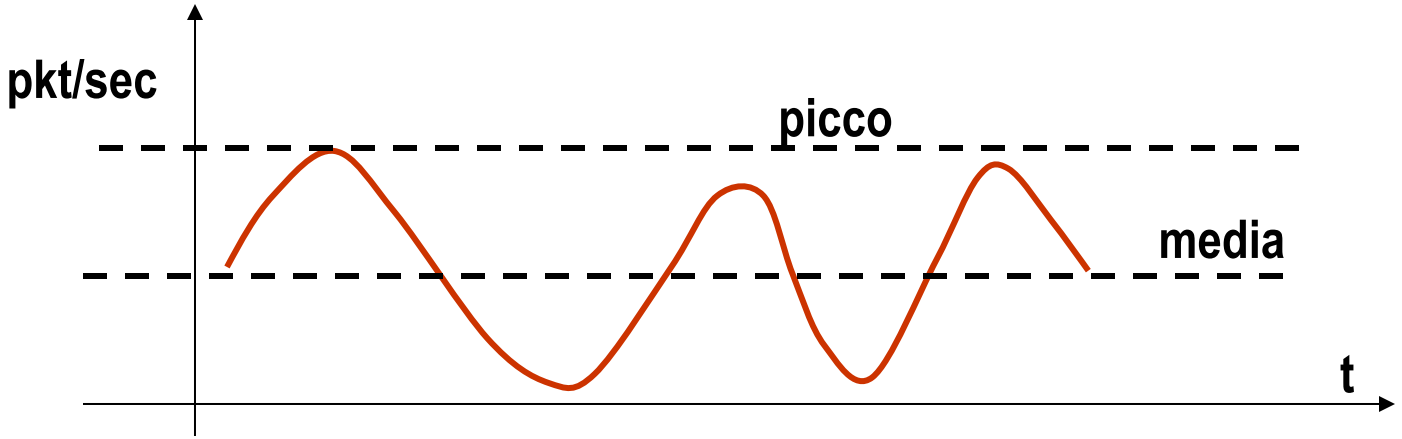


# La Commutazione di Pacchetto

- Trasferisce pacchetti o trame, ovvero unità informative composte da più bit
  - Il trasferimento è asincrono
- Le variazioni di traffico variano la **banda** disponibile sul canale
- I conflitti introducono **ritardi** variabili



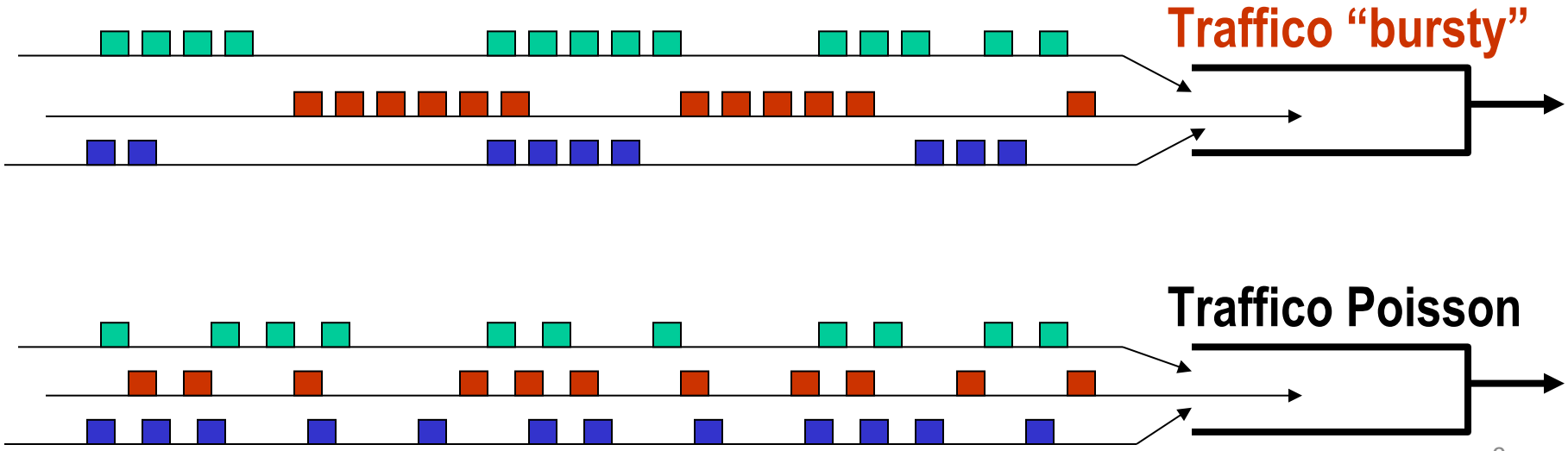
# Banda Variabile




- Si definiscono
  - Banda media
  - Banda di picco

# Burstiness

- Spesso si assume che i flussi di traffico in rete siano di tipo Poisson
  - Il numero di chiamate/pacchetti che arriva in rete segue una distribuzione di Poisson
  - Il tempo di inter-arrivo tra due chiamate/pacchetti segue una distribuzione esponenziale negativa (e i tempi di inter-arrivo sono indipendenti)
  - In questo modo è semplice calcolare i ritardi medi in rete
- In realtà, il traffico in rete è «bursty»
  - Bisogna tenere conto anche di questo aspetto quando si fanno valutazioni in termini di QoS



# Servizi real-time ed elastici

- E’ possibile definire (almeno) due categorie di tipologie di servizio
  - *Servizi real-time*: molto sensibili al ritardo ma meno alla perdita (es. VoIP)
  - *Servizi elastici*: molto sensibili alla perdita ma meno al ritardo, solitamente richiedono un ricontro (ACK) di corretta consegna (es. Web Browsing)
- Diverse tipologie di servizio richiedono, quindi, diversi requisiti in termini di QoS

# TCA e SLA

- Un provider di servizi IP a qualità garantita stabilisce con il cliente due contratti
  - Service Level Agreement (SLA)
  - Traffic Conditioning Agreement (TCA)
- Lo SLA specifica la QoS che il provider si impegna a garantire per uno specifico insieme di flussi di traffico, relativo a una specifica tipologia di servizio
  - L'impegno del provider sulla garanzia dello SLA non può prescindere dalla quantità di traffico generata dal cliente
- Il TCA specifica il profilo di traffico
  - Impone un limite su vari parametri caratterizzanti il traffico offerto dal cliente per cui il provider assicura l'impegno di onorare la SLA
  - Il traffico conforme al TCA si chiama *traffico IN* (o semplicemente traffico conforme)
  - Il traffico che eccede il TCA si chiama *traffico OUT* (o traffico non conforme)

# TCA e SLA

- Un requisito tipico delle reti di comunicazioni è la *disponibilità* (availability)
  - Percentuale di tempo in cui il provider eroga il servizio in modo corretto
  - In questo caso, lo SLA è specificato dal provider nel seguente modo: «Garantisco disponibilità al 99.99%»
  - Molto comune è la «five-nines availability» (99.999%)
- Gli SLA in generale sono definiti sulla base di vari parametri misurabili come
  - Ritardo end-to-end
  - Jitter
  - Throughput (volume di dati consegnati nell'unità di tempo)
  - Tasso di perdita

# Allocazione delle risorse

- La capacità del provider di garantire gli SLA per flussi o aggregati di flussi si fonda sulla riservazione di un adeguato ammontare di risorse di rete
  - La quantità adeguata di risorse dipende sia dai TCA che dagli SLA dei flussi che condividono ogni risorsa di rete
- L'accettazione in rete di traffico non conforme, ad allocazione di risorse avvenuta sulla base dell'insieme di TCA e SLA, è rischiosa
  - Il traffico non conforme potrebbe consumare risorse in misura tale da compromettere la capacità di garantire gli SLA
- L'accettazione in rete di traffico non conforme è operazione delicata
  - Si può marcare il traffico che viola il TCA per renderlo predisposto allo scarto in caso di sovraccarico delle risorse
  - Si può attribuire priorità inferiore al traffico non conforme (il traffico non conforme vene servito solo dopo il traffico conforme)

# Policing, Shaping e Marking

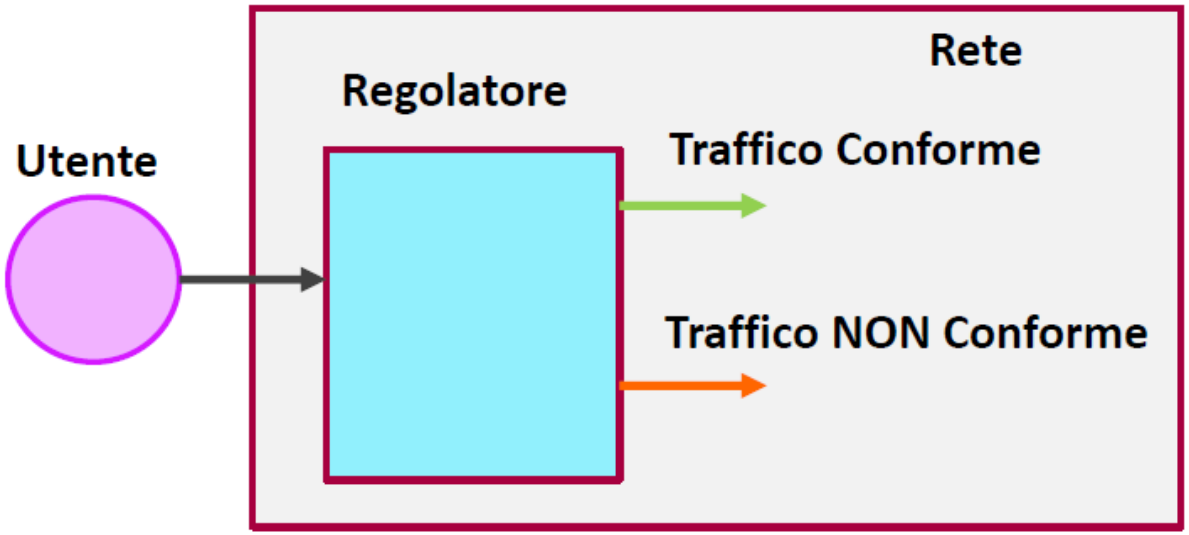
- Il provider deve garantire lo SLA solo per il traffico conforme, mentre possono essere effettuate diverse strategie di trattamento per il traffico non conforme
  - Il provider protegge gli SLA dei flussi conformi (di tutti gli utenti) dagli eccessi di traffico immessi da utenti che violano il loro TCA
  - Vengono evitate situazioni di sovraccarico e congestione che possono compromettere gli SLA
- Principali modalità di trattamento del traffico OUT
  - *Policing*: traffico OUT scartato
  - *Shaping*: traffico OUT ritardato in modo da attribuire un comportamento conforme al TCA
  - *Marking*: traffico OUT contrassegnato in modo tale da essere riconosciuto ed eliminato in caso di congestione

# Traffic Conditioning Agreement

- Il TCA può includere una varietà di parametri che caratterizzano il traffico conforme e non conforme
- Tali parametri includono usualmente
  - Velocità o Tasso di picco (Peak rate)
  - Velocità o Tasso medio (Average rate)
  - Massima lunghezza del burst: massimo numero di pacchetti consecutivi trasmessi alla velocità di picco del flusso
  - Lunghezza massima dei pacchetti
  - Lunghezza minima dei pacchetti
- Il TCA specifica quindi il profilo statistico del traffico conforme

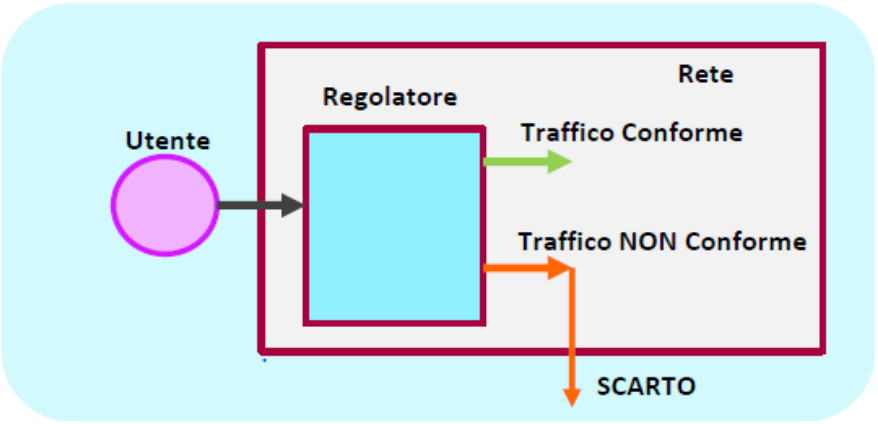
# Regolatore del traffico all'ingresso

- Stabilito per un flusso di traffico TCA e SLA, il traffico offerto alla rete viene esaminato da un Regolatore che distingue il traffico in almeno due flussi
  - Traffico conforme
  - Traffico non conforme
- Il trattamento del traffico non conforme distingue diversi tipi di regolatore

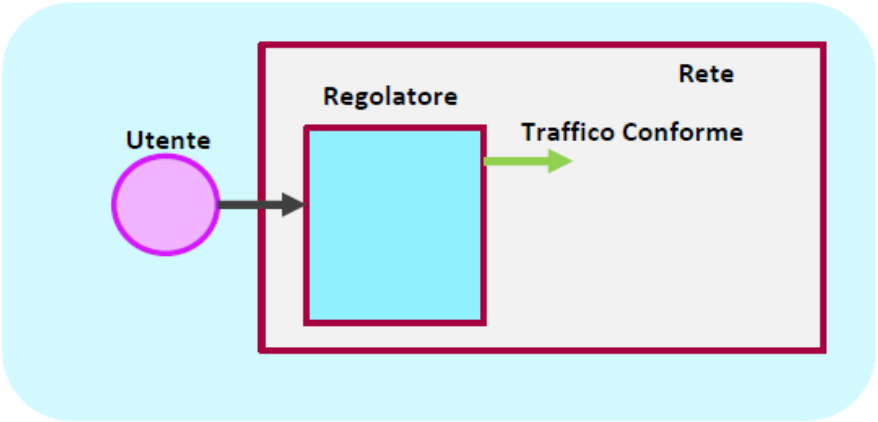


# Regolatore del traffico all'ingresso

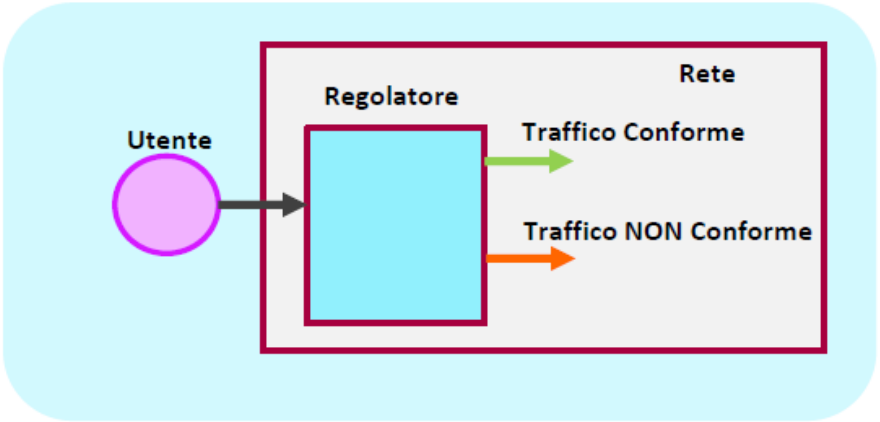
POLICER




SHAPER

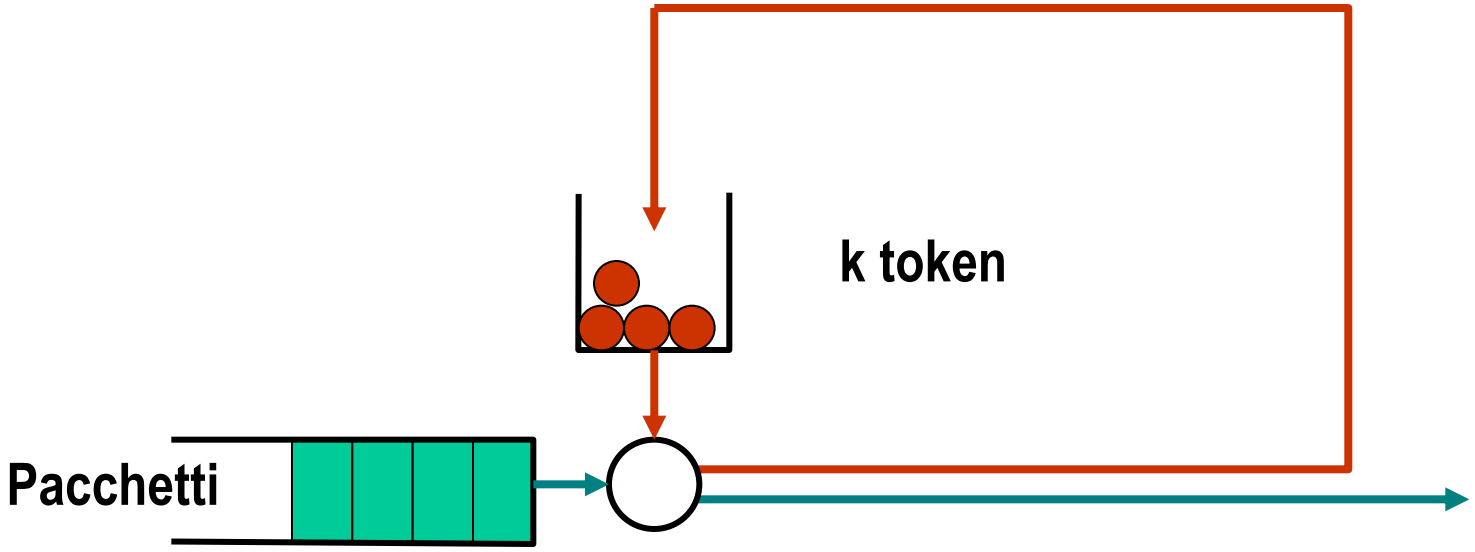



MARKER



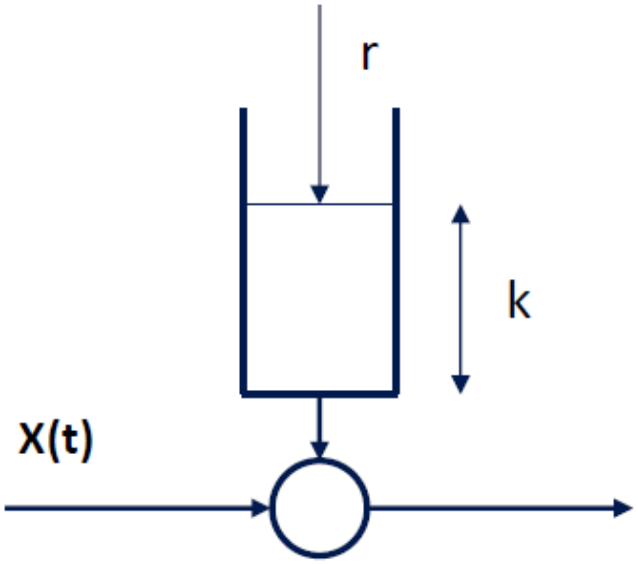
# Modello Token Bucket

- Token Bucket è un algoritmo utilizzato dal regolatore per controllare i principali parametri definiti nel TCA e discriminare il traffico conforme dal traffico non conforme
- Parametri controllati
  - Velocità di picco **p** [bit/s]
  - Velocità media **b** [bit/s]
  - Lunghezza del burst **L** [s]



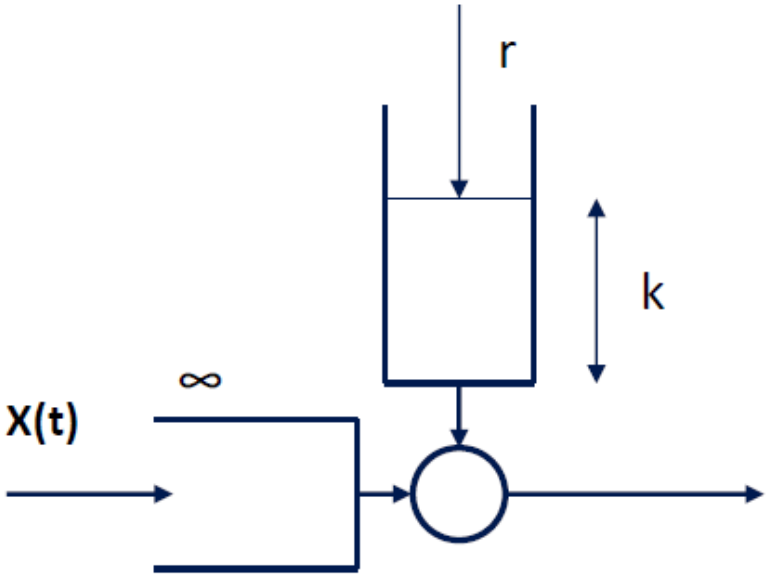
# Token Bucket - Policing

- Il token bucket policing regulator ha un serbatoio (bucket) di crediti (token) con dimensione massima di $k$ unità di traffico (token bucket size)
  - $k$ può essere misurato in bit, byte o pacchetti
- Il serbatoio dei crediti viene incrementato a ritmo costante ogni $1/r$, dove $r$ è il *token rate*
- Il dispositivo ammette il passaggio di una unità di traffico offerta (bit, byte o pacchetto) solo se il serbatoio dei crediti contiene almeno un token
  - In tal caso, il serbatoio viene decrementato di una unità
- Se il serbatoio non dispone di crediti, le unità di traffico vengono scartate



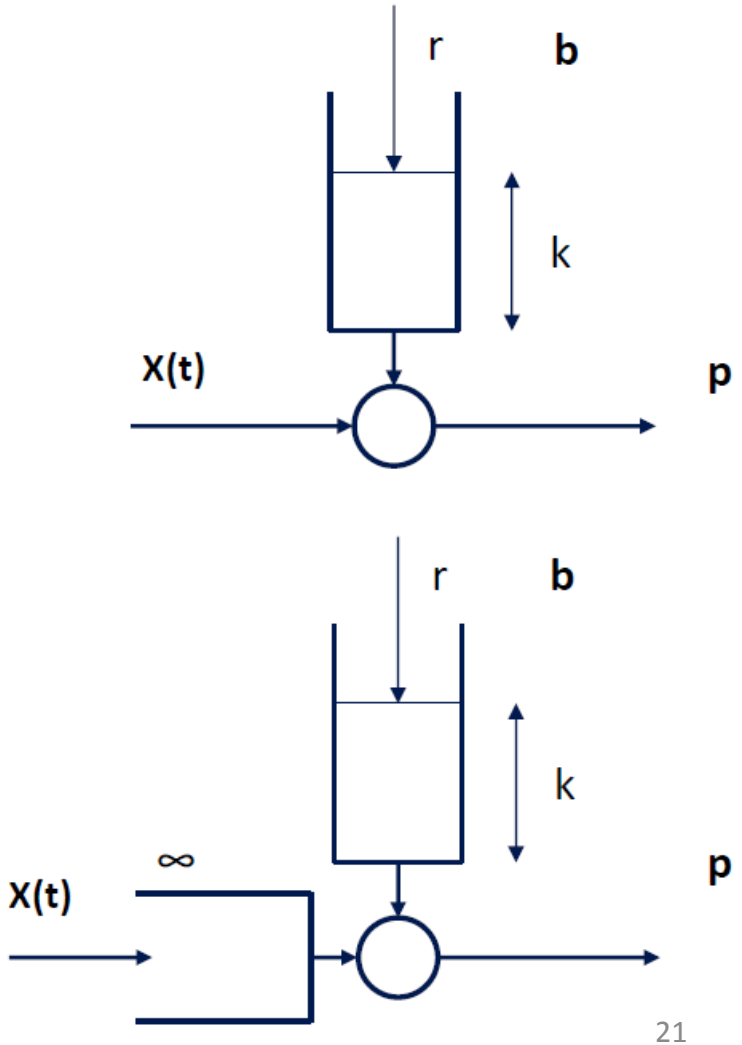
# Token Bucket - Shaping

- Il serbatoio dei crediti di uno shaping regulator opera come nel caso del policer
- Un'unità di traffico passa direttamente attraverso il regolatore se al suo arrivo il serbatoio ha capienza idonea e il buffer di ingresso (di dimensione infinita) è vuoto
- Se il buffer non è vuoto e/o il serbatoio di token non ha capienza idonea, l'unità di traffico viene trattenuta nel buffer di ingresso
- Quando il buffer di ingresso non è vuoto, una unità di traffico viene prelevata e inoltrata non appena si ha almeno un token disponibile nel serbatoio



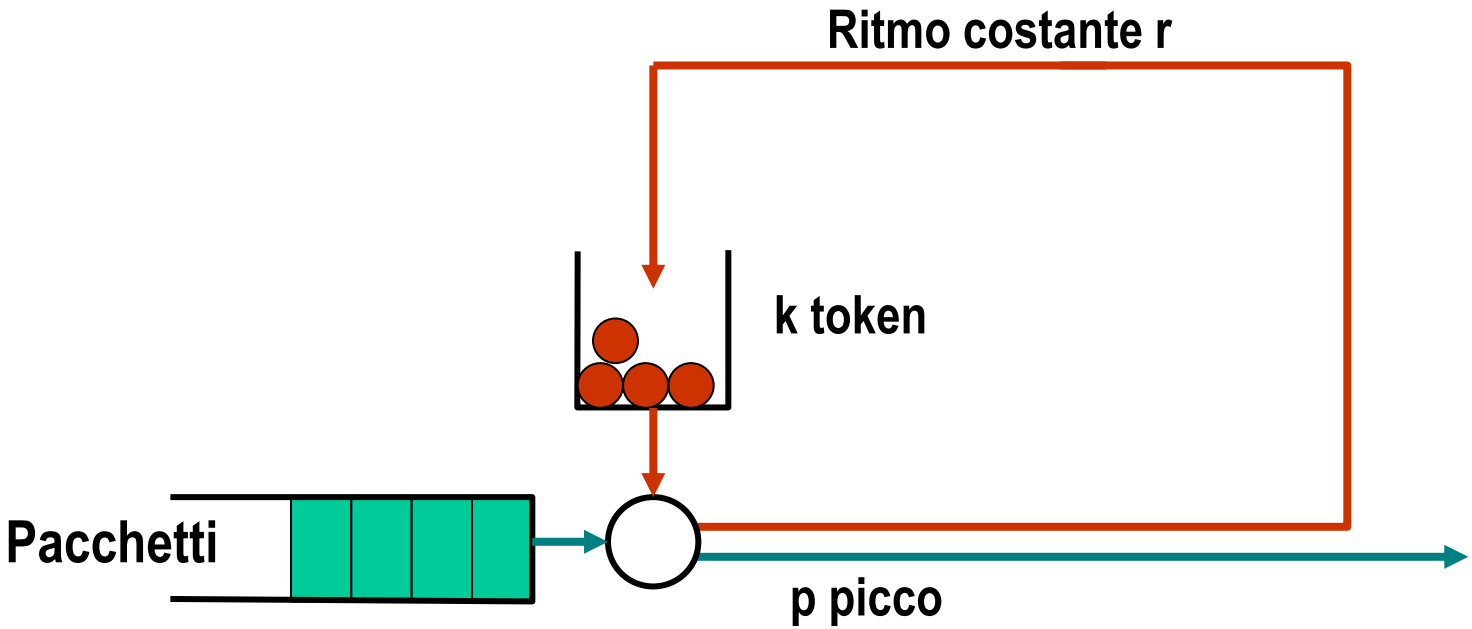
# Banda media e di picco

- Il tasso di arrivo dei token $r$ influenza il tasso medio del traffico offerto alla rete $\boldsymbol{b}$ (misurato in bit/s)
- La banda di picco $\boldsymbol{p}$ (misurata in bit/s) indica il tasso massimo di traffico offerto alla rete
  - Corrisponde alla velocità della linea
  - Ovviamente si ha che $\boldsymbol{b} < \boldsymbol{p}$



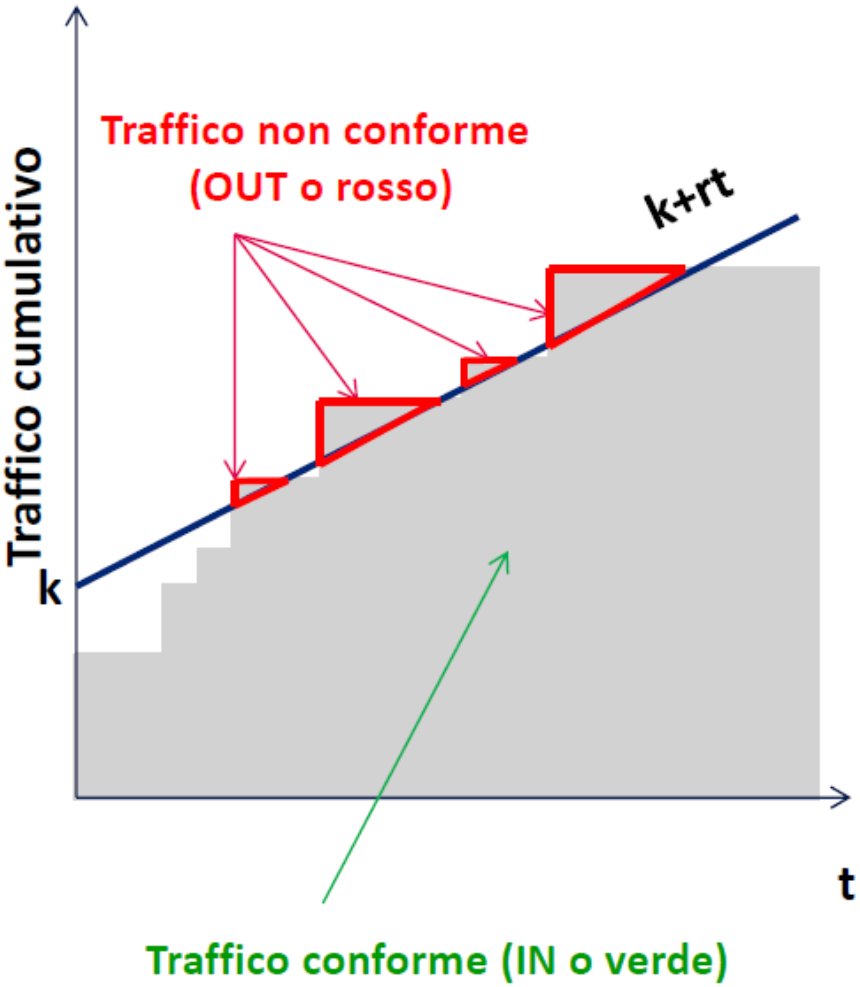
# Traffico «bursty»

- La presenza del serbatoio di crediti consente al traffico che viene generato dalla sorgente di essere ammesso in rete ad una velocità maggiore della banda media
- Il traffico offerto alla rete assume pertanto un profilo tipicamente discontinuo che si presenta con «raffiche» (o «burst» di traffico) alternate a periodi di inattività
- I regolatori controllano la banda media $b$, la banda di picco $p$ e la durata massima del burst $\boldsymbol{L}$
- E' possibile calcolare la durata massima del burst $\boldsymbol{L}$ nel seguente modo: $L = \frac{k}{p-r}$
- Token bucket preserva la «burstiness»



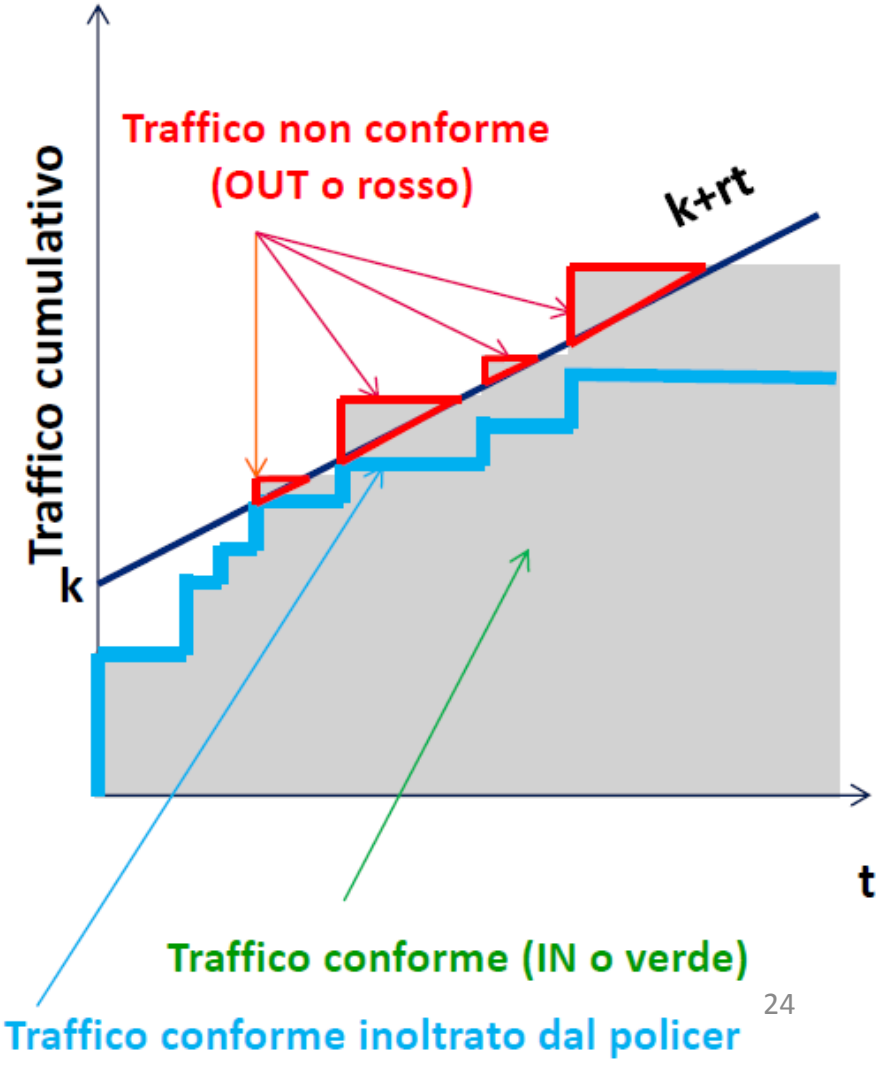
# Funzione vincolo

- I regolatori policer e shaper implementano una funzione vincolo
- La funzione vincolo è la retta $k + rt$ e rappresenta il numero massimo di unità di traffico corrispondenti al traffico conforme che il regolatore lascia passare in rete
- In un periodo di tempo di durata $\bar{t}$, il massimo traffico conforme che il regolatore lascia passare è uguale a $k + r\bar{t}$
- Il traffico in eccesso è non conforme ed è trattato in modo diverso da policer e shaper



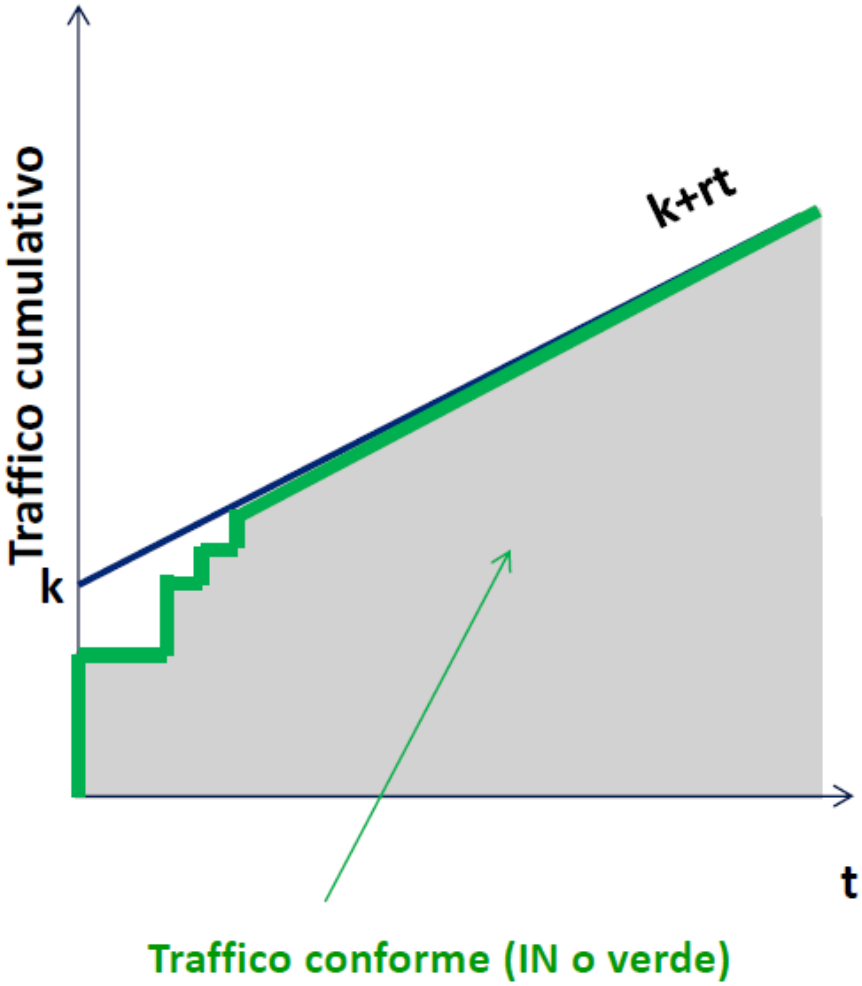
# Comportamento del policer

- Un policer scarta il traffico non conforme
- In questo modo, solo il traffico conforme entra in rete



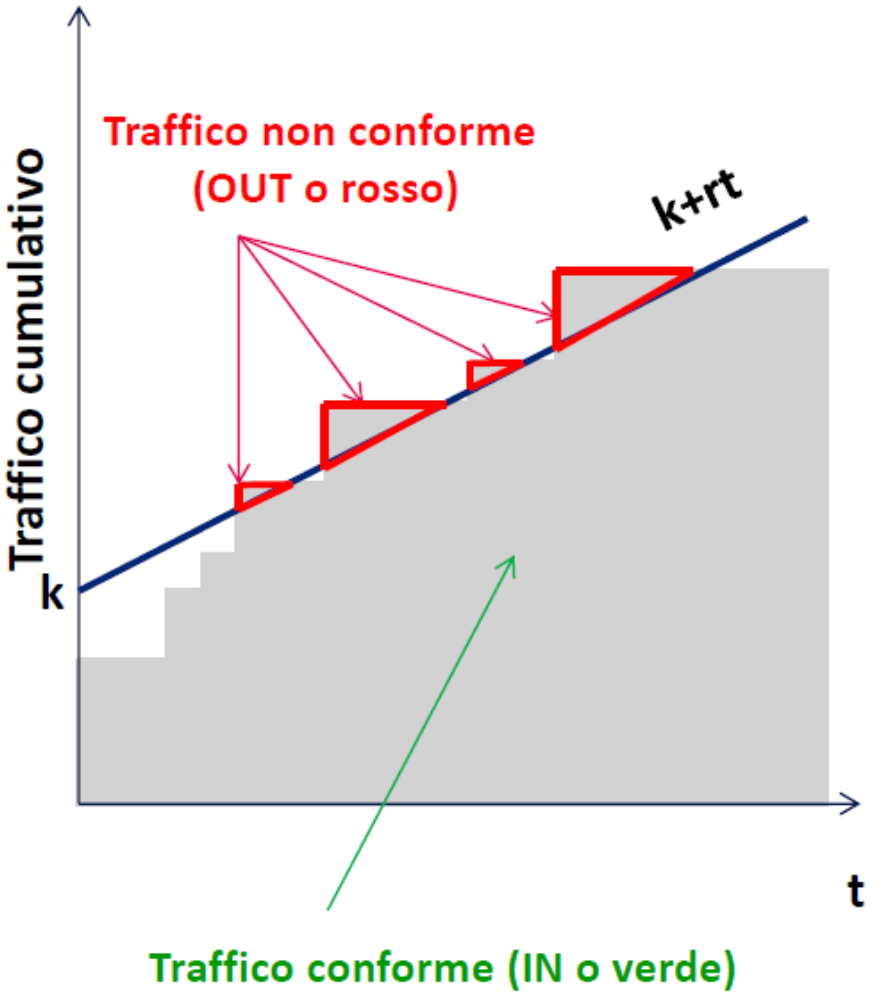
# Comportamento dello shaper

- Uno shaper trattiene il traffico non conforme nel buffer e lo serve appena sono disponibili token in conformità con il vincolo
- Lo shaper elimina la perdita di pacchetti all'ingresso ma aggiunge ritardo
- Il policer introduce un ritardo trascurabile ma causa perdita di pacchetti



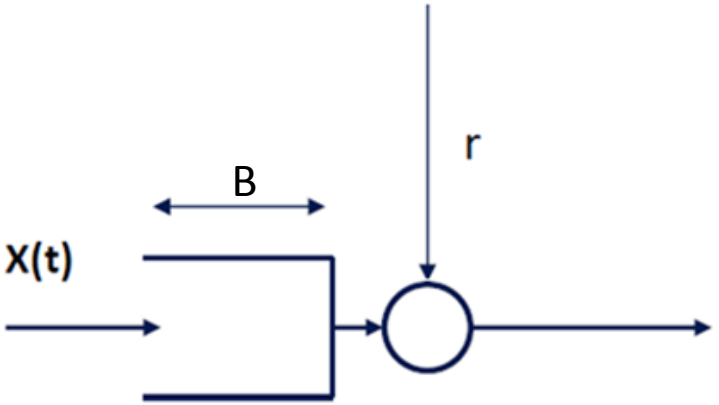
# Comportamento del marker

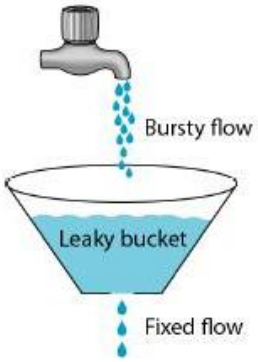
- Un token bucket marker lavora come un policer ma non scarta il traffico non conforme
- Il traffico non conforme viene marcato e inoltrato in rete
- In caso di congestione i pacchetti marcati possono essere scartati prima del traffico conforme

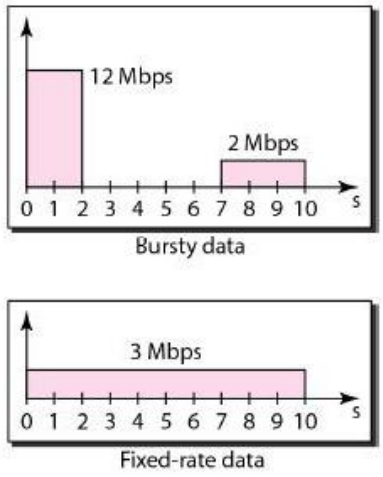


# Leaky bucket

- Algoritmo alternativo al Token Bucket
- A differenza dal Token bucket, il Leaky bucket non ha un serbatoio per i crediti
- Viene generato un credito ogni $1/r$
- Il buffer delle unità di traffico, invece ha dimensione $B$ (e non infinita)
  - Oltre $B$ il traffico viene scartato
- Non potendo essere accumulati i crediti, un eventuale traffico burst viene reso «smooth» (si perde l'informazione relativa a **L**)
  - Non è possibile avere una velocità in uscita superiore alla velocità $r$, che è quindi anche la velocità di picco ($r = p$)



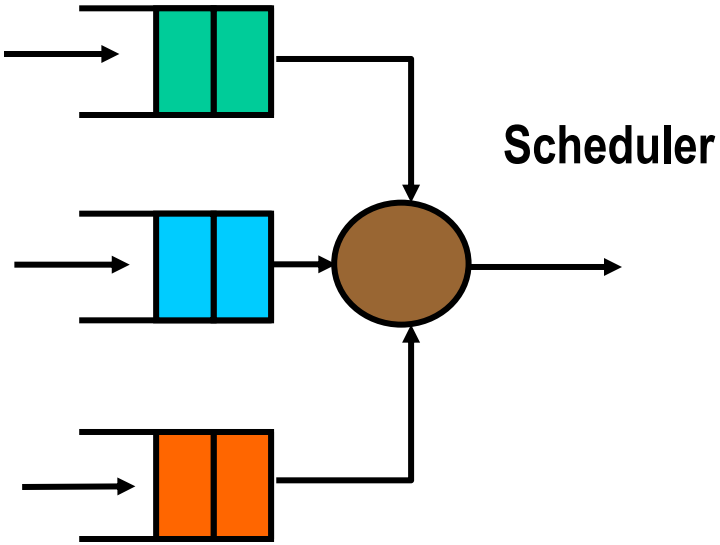
# STRUMENTI PER GARANTIRE LA QoS

# Strumenti per garantire la QoS

- Per garantire la *QoS* servono
  1. Strumenti  di identificazione dei flussi (label, virtual circuits, etc.)

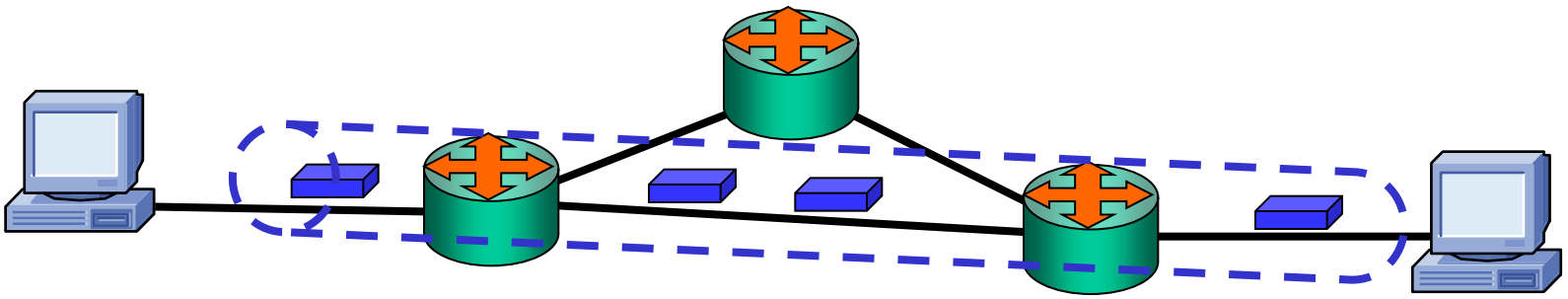
# Strumenti per garantire la QoS

2. Strumenti per suddividere la banda nei router (tecniche di ***scheduling***)



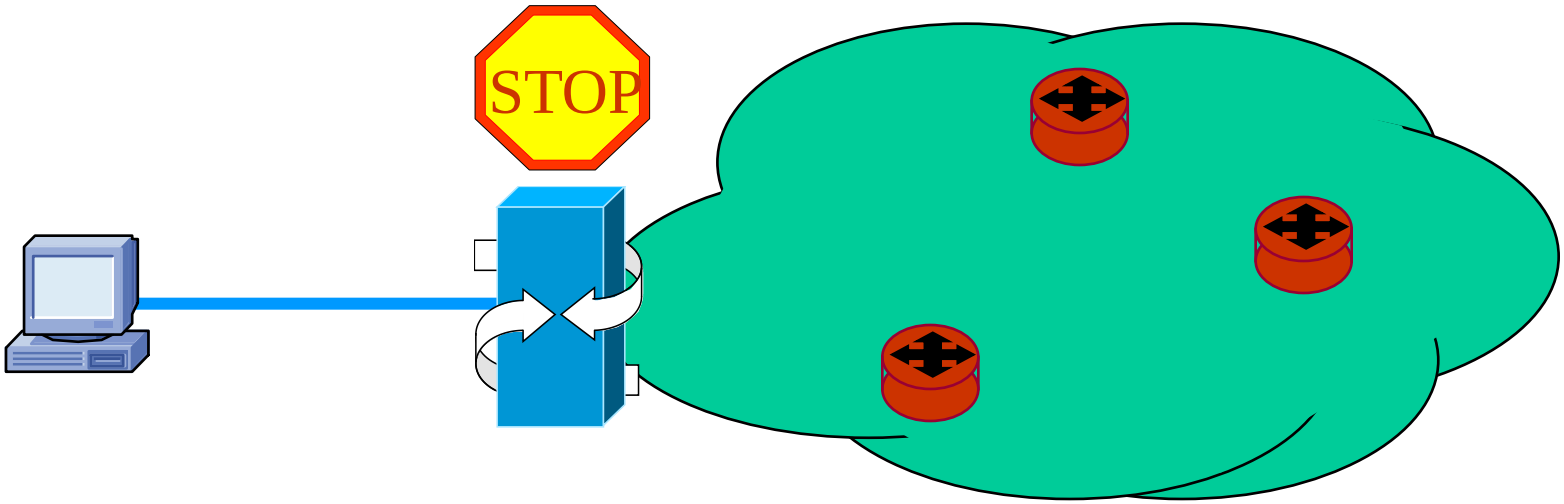
# Strumenti per garantire la Banda

- Per garantire la *banda* richiesta dalle SLA occorrono strumenti per fissare i cammini in rete (es. MPLS)



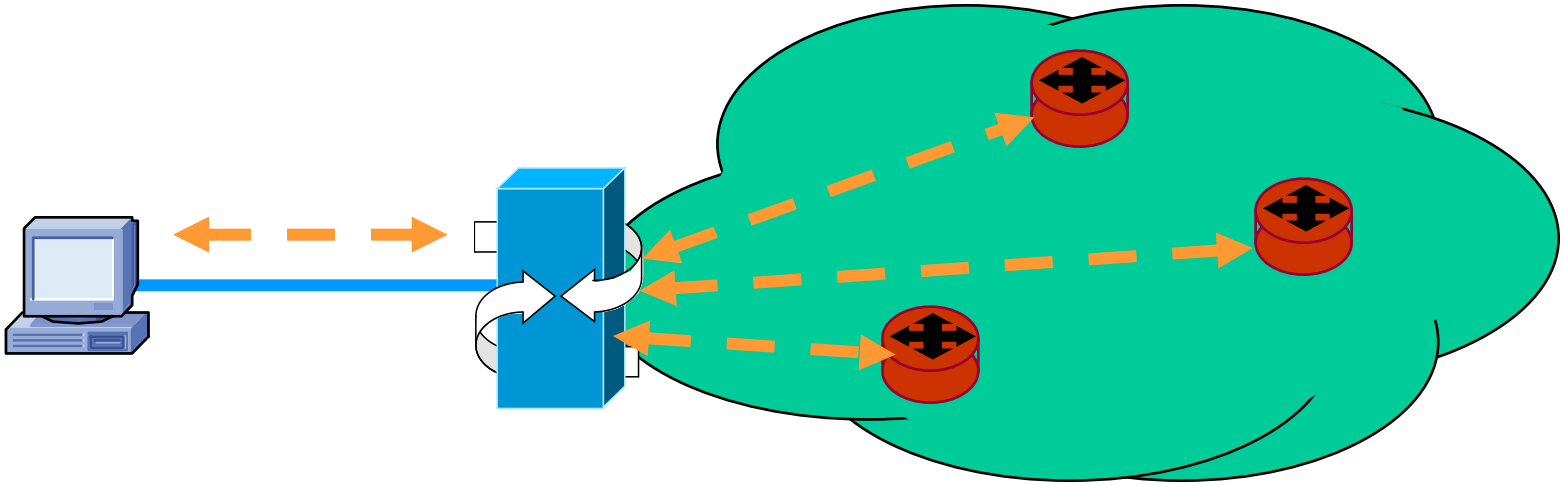
# Garantire la Banda in modo dinamico

- Occorrono
  1. Meccanismi di controllo delle risorse (CAC: Call Admission Control) che possano rifiutare la richiesta se la banda non è disponibile



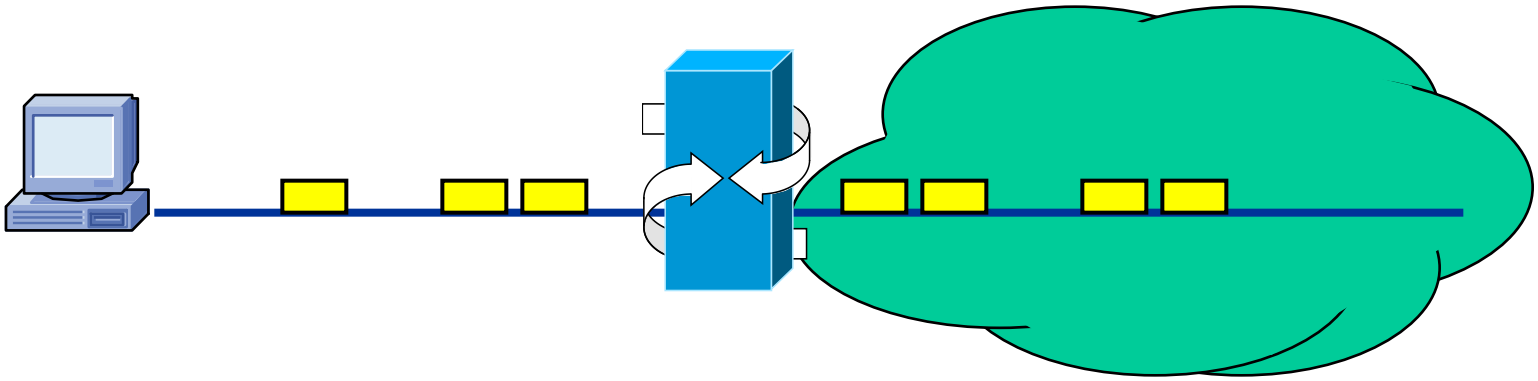
# Garantire la Banda in modo dinamico

2. Meccanismi di segnalazione delle risorse di rete (banda disponibile sui vari cammini)



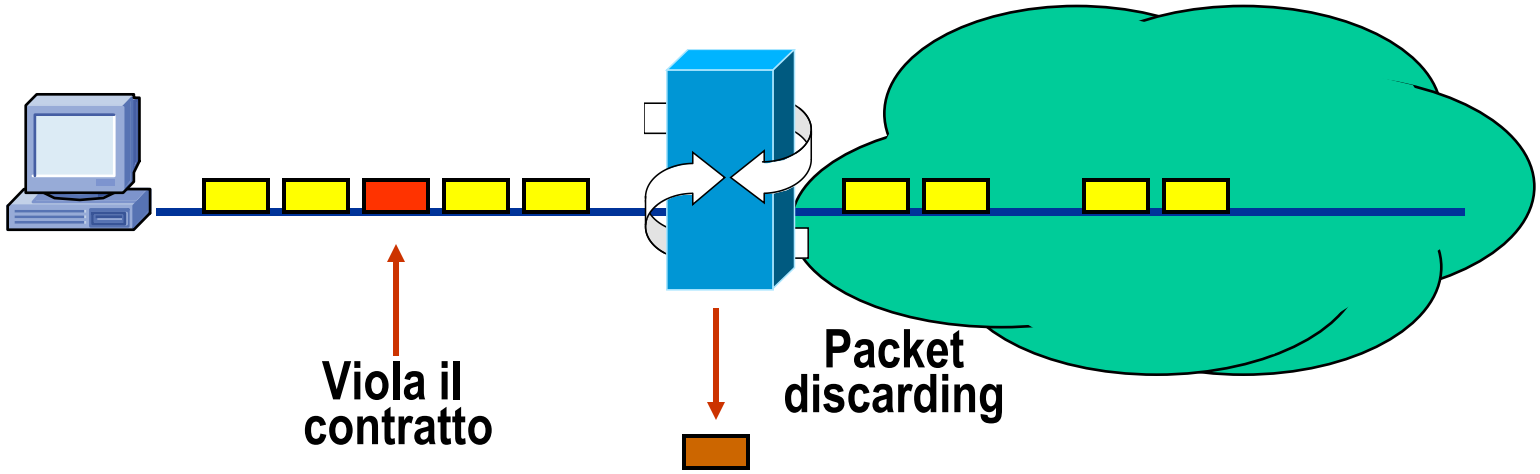
# Controllo del traffico immesso

- Il traffico immesso non è vincolato dal mezzo fisico
  - Spesso è variabile, con banda media $b$ minore di quella di linea $p$ (picco)
  - I contratti tengono conto della media e del picco (lunghezza del burst)



# Strumenti di regolazione del traffico

- Per evitare che la banda venga consumata oltre le aspettative occorrono strumenti di regolazione del traffico che
  - Controllino il traffico immesso
  - Verifichino la conformità alle risorse
  - Prendano provvedimenti in caso di violazione



# Over-provisioning

**Se la banda fosse infinita…**

**… la QoS non sarebbe un problema**

- La banda non è infinita (ha un costo)
- Le richieste fluttuano nel tempo
- Le richieste crescono col tempo
- Le richieste prima o poi generano conflitti, che risultano in perdita di QoS e trattamento iniquo dei flussi (unfairness)

# Traffic Engineering

## Con il CAC la QoS può essere garantita su qualunque rete

- Basta limitare il  traffico ammesso a un livello sufficientemente basso
  - Ma il traffico troppo basso scontenta i gestori
- Per questo in aggiunta si usano strumenti di *traffic engineering* che consentono di sfruttare meglio le risorse  e quindi di spostare il livello di traffico a cui far intervenire il CAC a valori più elevati

# ALLOCAZIONE DELLE RISORSE

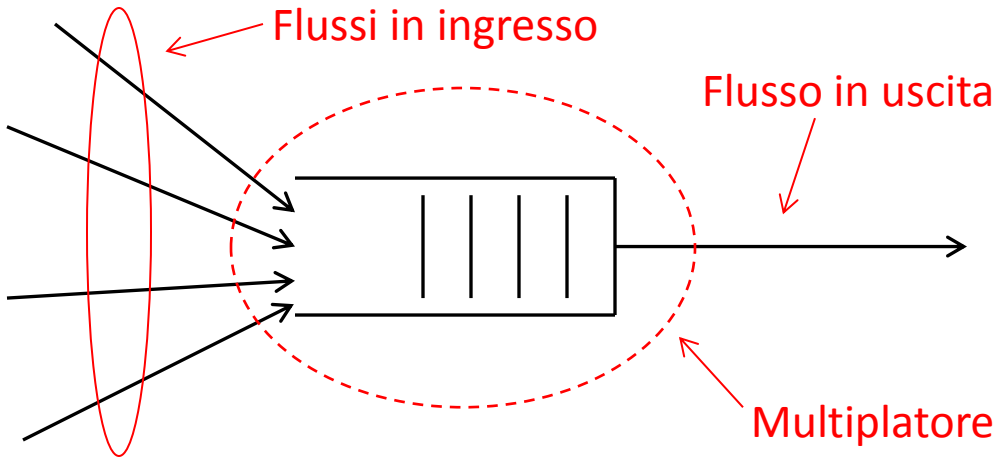
## Controllo del traffico immesso

# Tipologie di allocazione delle risorse

- Le tecniche di *traffic shaping/policing/marking* viste in precedenza permettono di definire dei *regolatori* che controllano il traffico immesso in rete
- Queste tecniche sono molto importanti perché, se utilizzate correttamente, permettono di migliorare l'utilizzo e l'allocazione delle risorse in rete

# Tipologie di allocazione delle risorse

- Quando diversi flussi di pacchetti IP vengono multiplati da un multiplatore per generare un unico flusso aggregato in uscita, è necessario che le risorse (in uscita) vengano suddivise (allocate) in modo intelligente

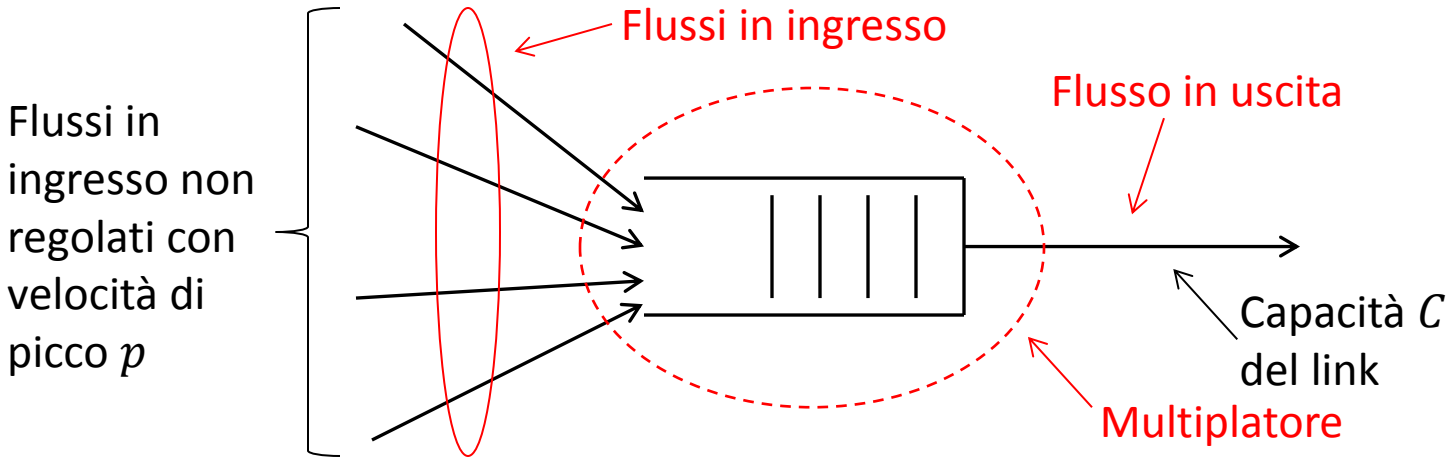


# Tipologie di allocazione delle risorse

- Esistono due diverse tipologie di allocazione delle risorse
  - *Allocazione deterministica*: le risorse sono suddivise staticamente tra i differenti flussi in ingresso
    - Non è prevista perdita per i pacchetti dei flussi
  - *Allocazione statistica*: le risorse sono suddivise in modo dinamico tra i flussi secondo criteri statistici
    - E' possibile avere perdita di pacchetti, che deve essere controllata
- Ci soffermiamo su due tecniche di allocazione deterministica
  - Allocazione al picco
  - Allocazione mediante algoritmo Dual Leaky Bucket (DLB)

# Allocazione al picco

- L'allocazione al picco deve essere utilizzata quando il traffico relativo ai flussi in ingresso non è stato regolato da alcun regolatore
  - La risorsa che deve essere suddivisa è la capacità $C$ [bit/s] del link in uscita dal multiplatore
- L'allocazione al picco assegna a ogni flusso una porzione di capacità in uscita pari alla velocità di picco (di linea) $p$ [bit/s]



# Allocazione al picco

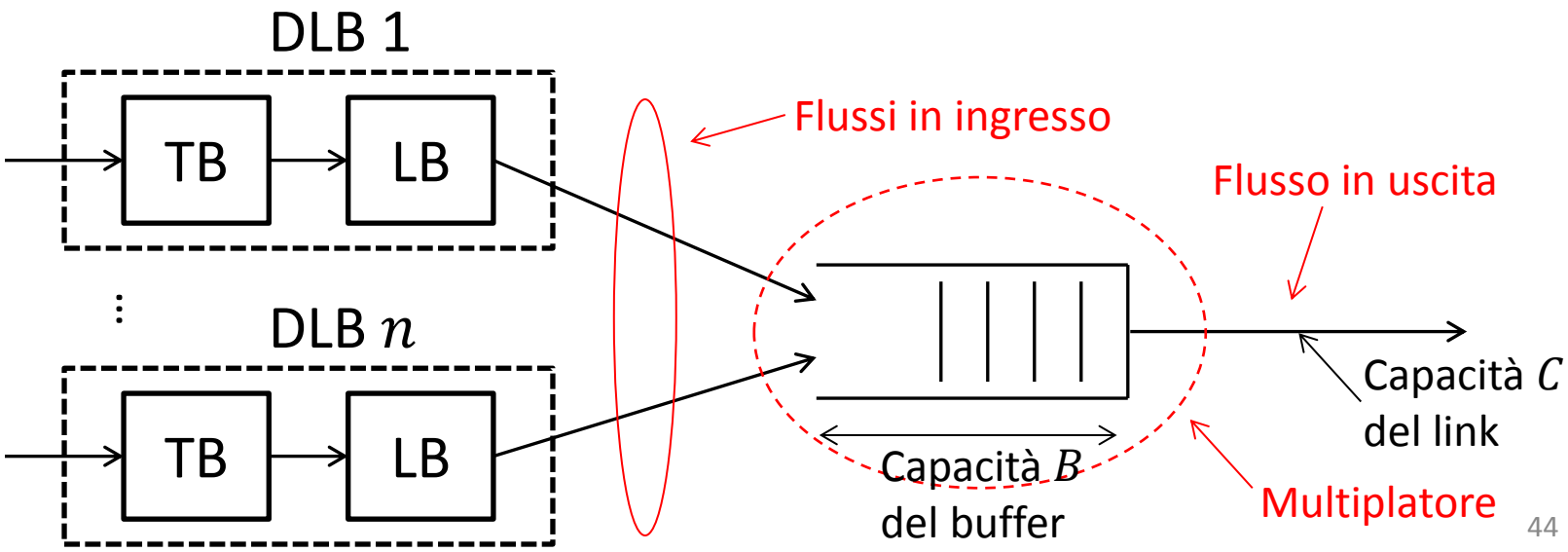
- Supponendo di avere flussi in ingresso omogenei, ovvero tutti con velocità di picco $p$, il numero massimo di flussi che possono essere multiplati senza perdita è pari a $\mathrm{N}_p = \frac{C}{p}$
- Se $b_m \leq p$ è la velocità media di ognuno dei flussi, possiamo definire *efficienza* della strategia di allocazione la seguente quantità:

$$\eta_p = \frac{N_p \cdot b_m}{C} = \frac{b_m}{p}$$

- L'allocazione al picco alloca le risorse sulla base del caso peggiore
  - Porta a una tanto più scarsa efficienza $\eta_p$ tanto maggiore è la differenza tra la velocità media e di picco dei flussi emessi dalle sorgenti, ma garantisce l'assenza di perdita di pacchetti in ogni condizione

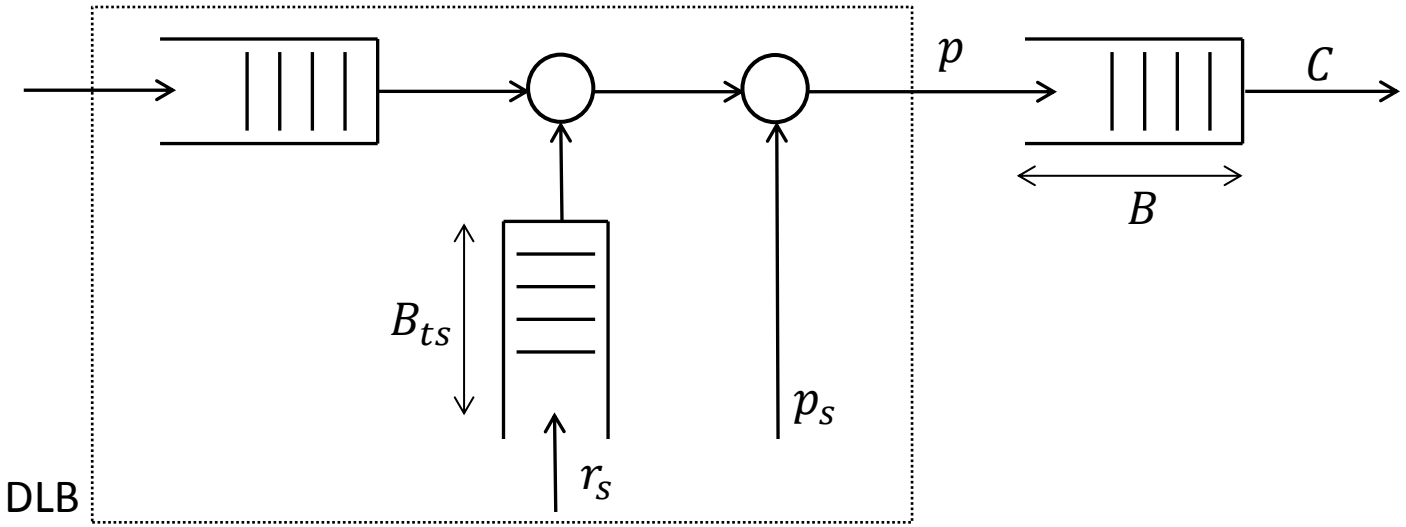
# Allocazione Dual Leaky Bucket

- L'allocazione DLB è possibile quando il traffico dei flussi in ingresso è stato regolato alla sorgente da una cascata token bucket/leaky bucket (TB+LB)



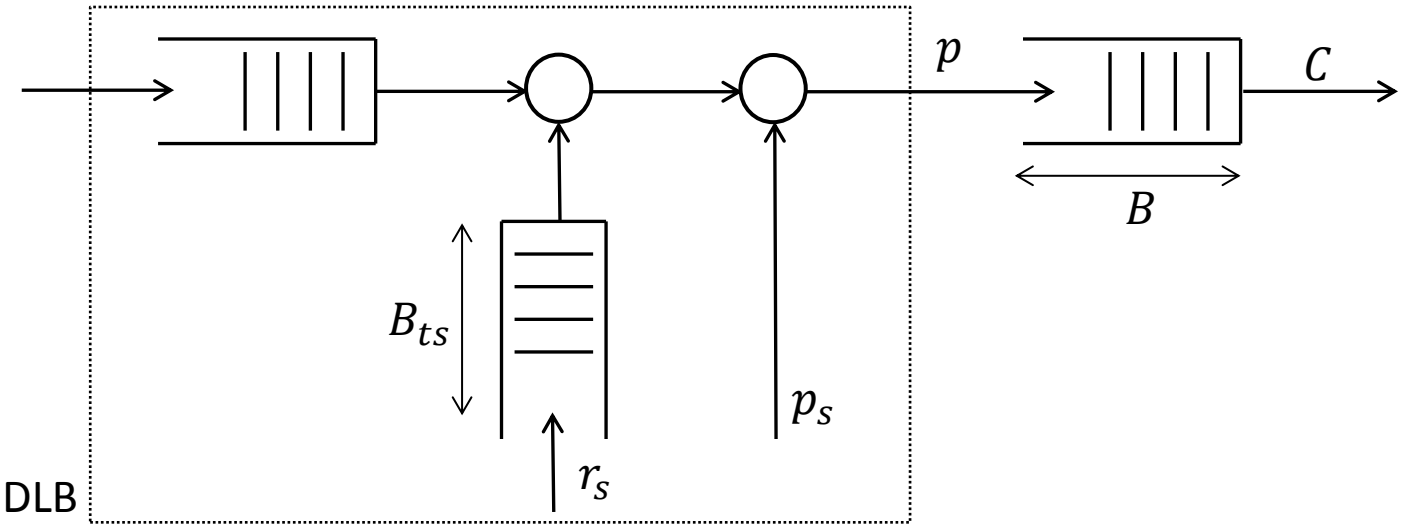
# Allocazione Dual Leaky Bucket

- In questo modo, una sorgente di un flusso può essere modellata da tre parametri ben definiti
  - $r_s$ [bit/s]: velocità della sorgente sostenibile
  - $B_{ts}$ [bit]: dimensione del buffer dei token
  - $p_s > r_s$ [bit/s]: ritmo di trasferimento di picco



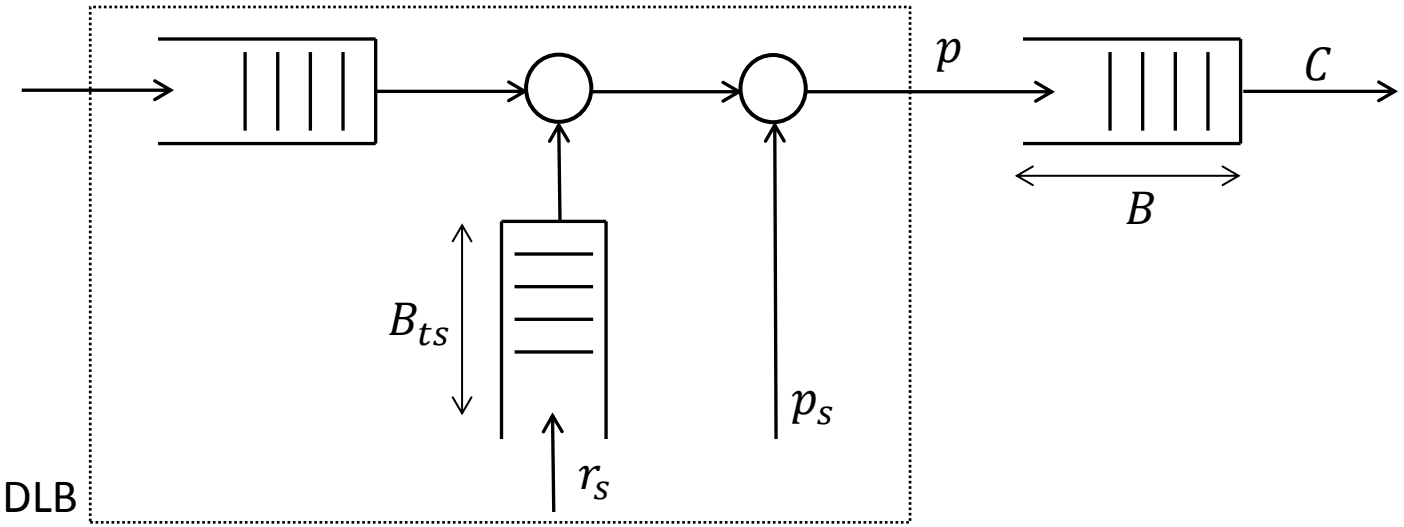
# Allocazione Dual Leaky Bucket

- Il DLB riduce la durata dei burst ad un valore massimo prefissato
- Permette di effettuare una allocazione deterministica (senza perdita) efficiente nel caso in cui vengano dimensionati correttamente i parametri $B$ e $C$
  - $B$ [bit]: dimensione del buffer del multiplatore
  - $C$[bit/s]: capacità di uscita del multiplatore



# Allocazione Dual Leaky Bucket

- Il primo elemento (token bucket) regola la velocità in uscita $r_s$, permettendo una certa tolleranza (dipendente dalla dimensione di $B_{ts}$)
- Il secondo elemento (leaky bucket) regola la velocità di picco $p_s$ (inferiore alla velocità di linea $p$)



# Allocazione Dual Leaky Bucket

- La presenza del DLB impone un limite al tempo massimo in cui è possibile inviare pacchetti alla velocità di picco $p_s$ (durata massima del burst)

$$T_{picco} = \frac{B_{ts}}{p_s - r_s} \text{ [s]}$$

- La massima dimensione di un burst vale quindi

$$W_p = p_s \cdot T_{picco} = \frac{B_{ts} \cdot p_s}{p_s - r_s} \text{ [bit]}$$

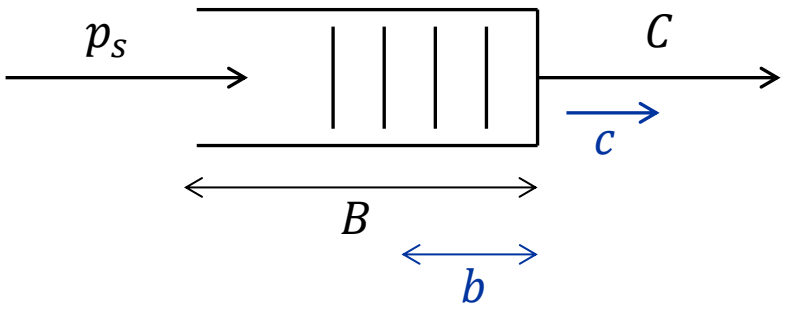
# Allocazione Dual Leaky Bucket

- L'obiettivo è quindi di allocare una porzione di banda/buffer del multiplatore $(c, b)$ per ogni flusso che sia una frazione di $(C, B)$ (*fair share*) rispettando
  1. Un vincolo di ritardo massimo imposto, $D_{max}$

$$\frac{B}{C} = \frac{b}{c} = D_{max} \text{ [s]}$$

  2. Un vincolo per assicurare una frazione di buffer $b$ per flusso tale da evitare perdite

$$b = (p_s - c)T_{picco} = \frac{B_{ts}}{p_s - r_s}(p_s - c) \text{ [bit]}$$



# Allocazione Dual Leaky Bucket

- Se definiamo $N_{DLB}$ il numero massimo di flussi nel sistema, il secondo vincolo può essere riscritto come

$$N_{DLB} b = N_{DLB} \ (p_s - c) \ T_{picco}$$

$$N_{DLB} b = (N_{DLB} p_s - N_{DLB} c) T_{picco}$$

$= B$ $= C$

$$B = (N_{DLB} p_s - C) T_{picco} = \frac{B_{ts}}{p_s - r_s} (N_{DLB} p_s - C)$$

# Allocazione Dual Leaky Bucket

- Impostando il sistema

$$\begin{cases} B = C \cdot D_{max} \\ B = \dfrac{B_{ts}}{p_s - r_s}(N_{DLB} p_s - C) \end{cases}$$

- Si può calcolare $N_{DLB}$

$$N_{DLB} = \frac{C}{p_s}\left(1 + \frac{D_{max}}{T_{picco}}\right)$$

- Variando opportunamente i parametri $p_s$, $r_s$ e $B_{ts}$ (e considerando un ritardo massimo nella coda del multiplatore $D_{max}$) delle sorgenti è possibile ottenere valori maggiori o minori di $N_{DLB}$
  - Si agisce a monte (alla sorgente)

# Confronto allocazione al picco/DLB

- Numero di flussi massimi con allocazione al picco

$$\mathrm{N}_p = \frac{C}{p}$$

- Numero di flussi massimi con allocazione DLB

$$N_{DLB} = \frac{C}{p_s}\left(1 + \frac{D_{max}}{T_{picco}}\right)$$

- Essendo $p \geq p_s$ si ha che

$$N_{DLB} \geq N_p$$

- Il rapporto $D_{max}/T_{picco}$, poi, permette di aumentare $N_{DLB}$ ulteriormente rispetto a $N_p$ (tale rapporto dipende anche da $B_{ts}$ e $r_s$)
- Di conseguenza, assumendo una velocità media $b_m$ costante per le due tipologie di flusso

$$\eta_p = \frac{N_p \cdot b_m}{C} \leq \eta_{DLB} = \frac{N_{DLB} \cdot b_m}{C}$$

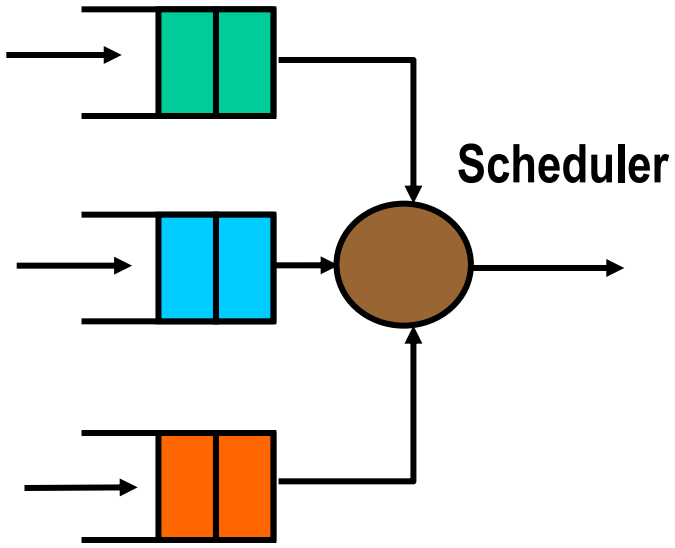
- Dual Leaky Bucket, quindi, permette di migliorare l'allocazione delle risorse

# ALLOCAZIONE DELLE RISORSE

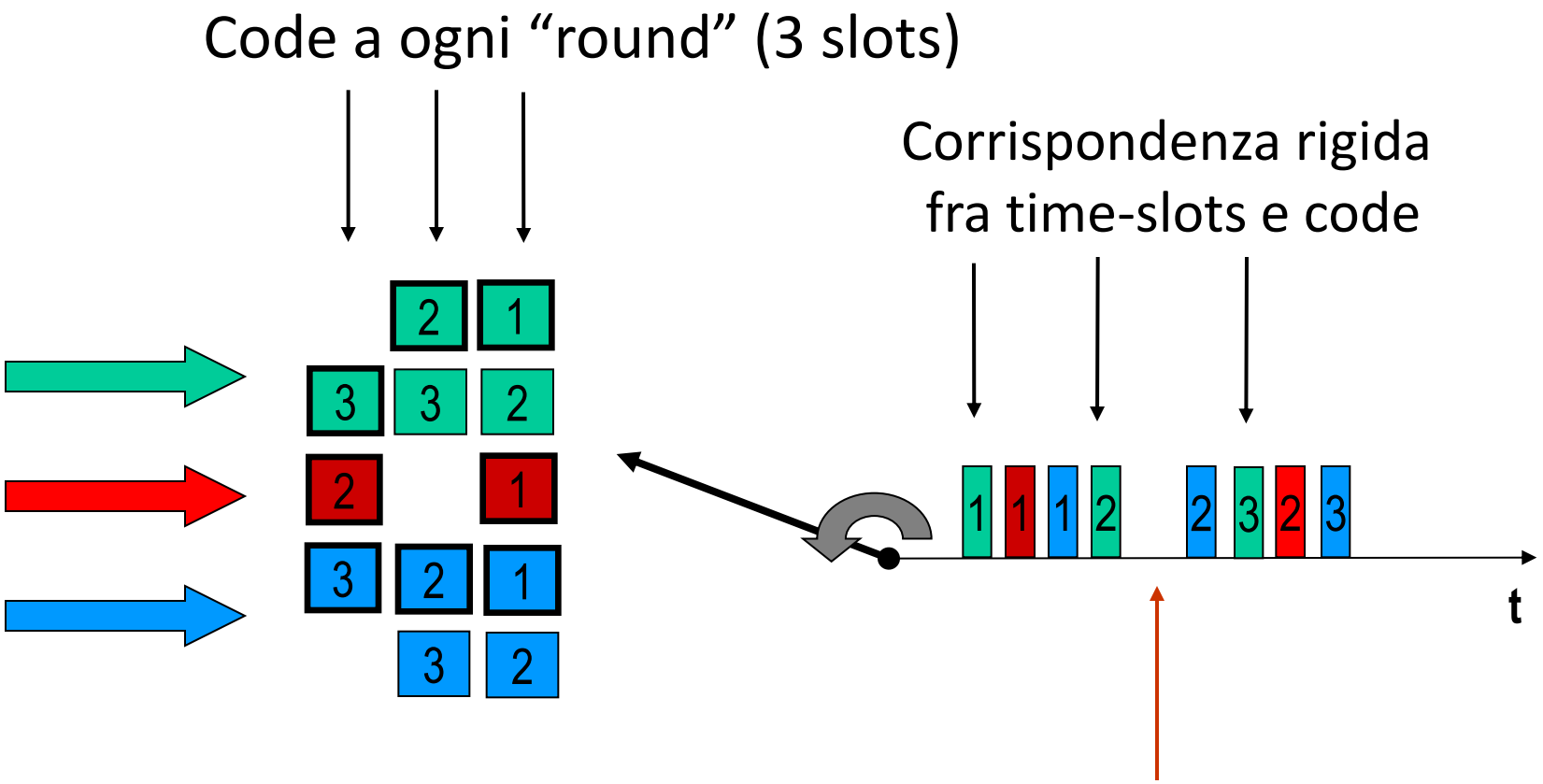
## Suddivisione della banda

# Scheduling

- Strumenti per suddividere la banda nei router (tecniche di scheduling)
- La banda viene condivisa tra pacchetti di diverse *code* in ingresso



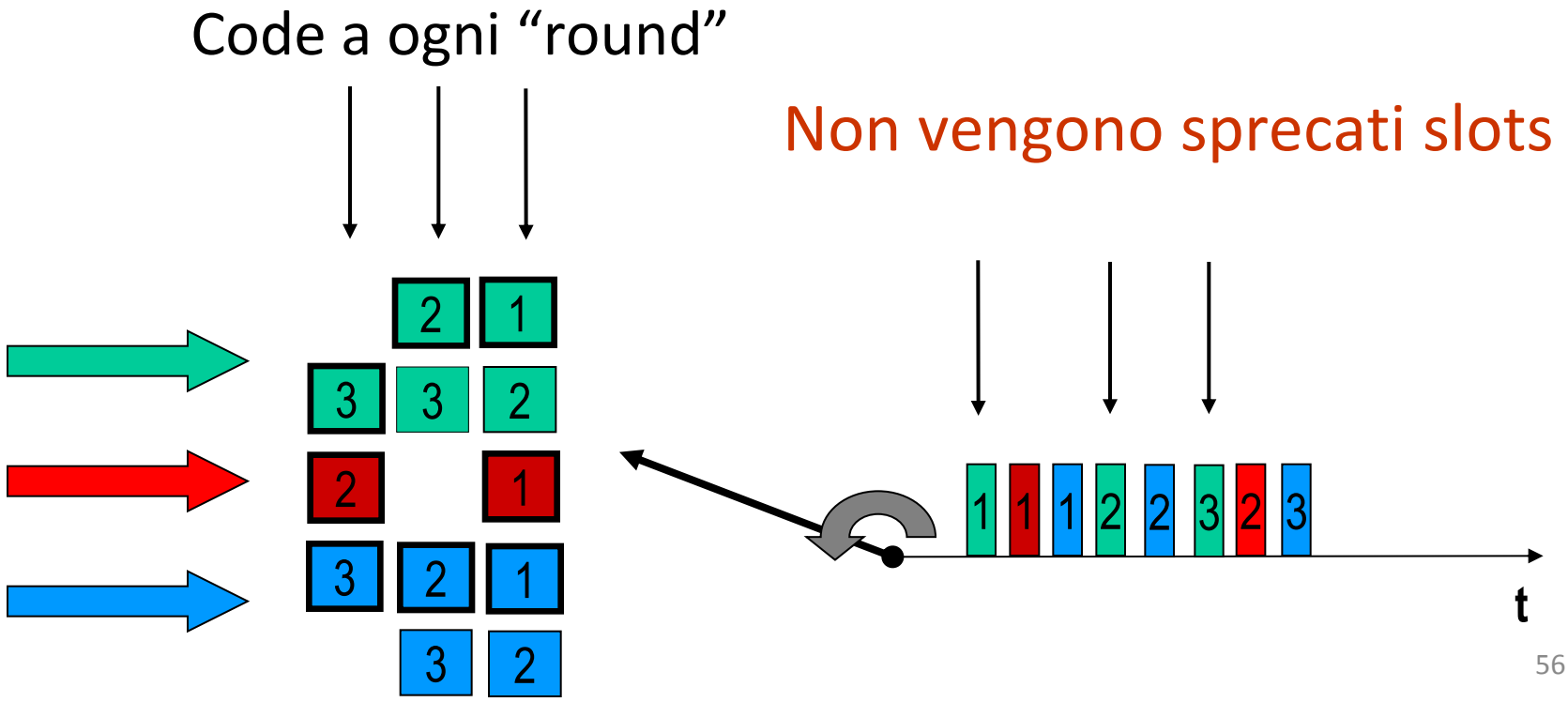
# Time Division Multiplexing




**Non consente il riutilizzo di risorse non usate da un flusso**

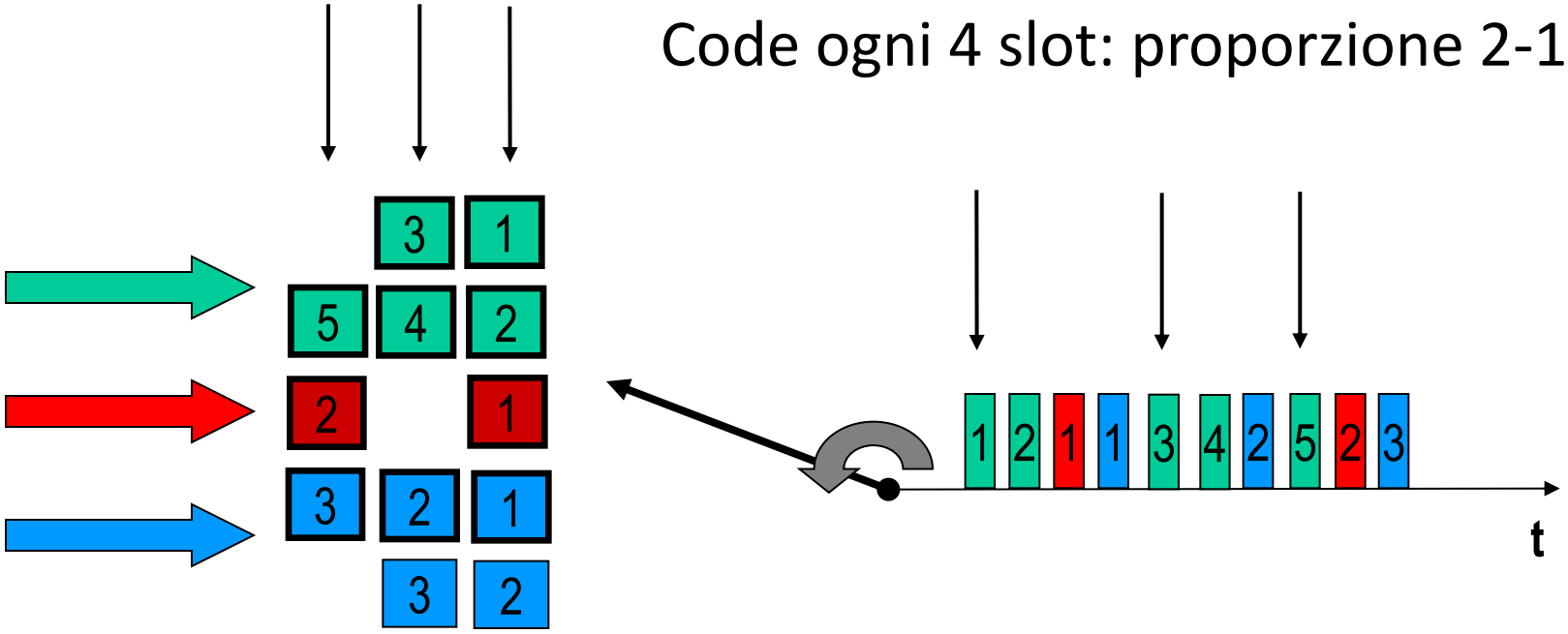
# Round Robin (Fair Queuing)

- Suddivisione dinamica della banda ottenuta trasmettendo un pacchetto per ogni flusso ciclicamente



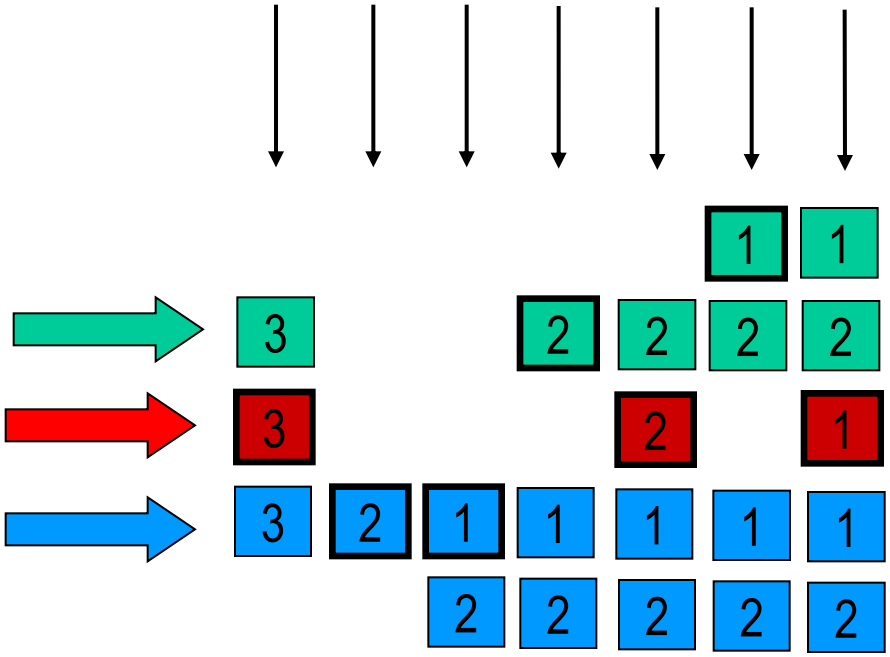
# Weighted Fair Queuing

- Suddivisione dinamica e proporzionale della banda ottenuta trasmettendo $K_i$ pacchetti per ogni flusso $i$ in modo pseudo-ciclico.

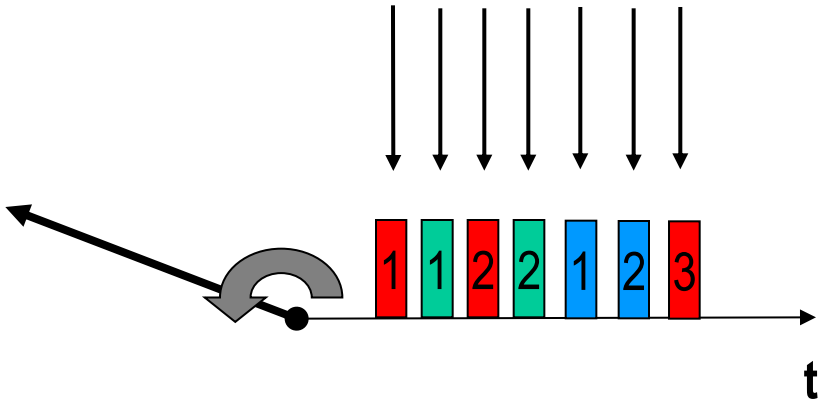


# Priorità di Servizio

- Suddivisione dinamica preferenziale della banda
- Le code con priorità basse possono sperimentare ritardi elevati

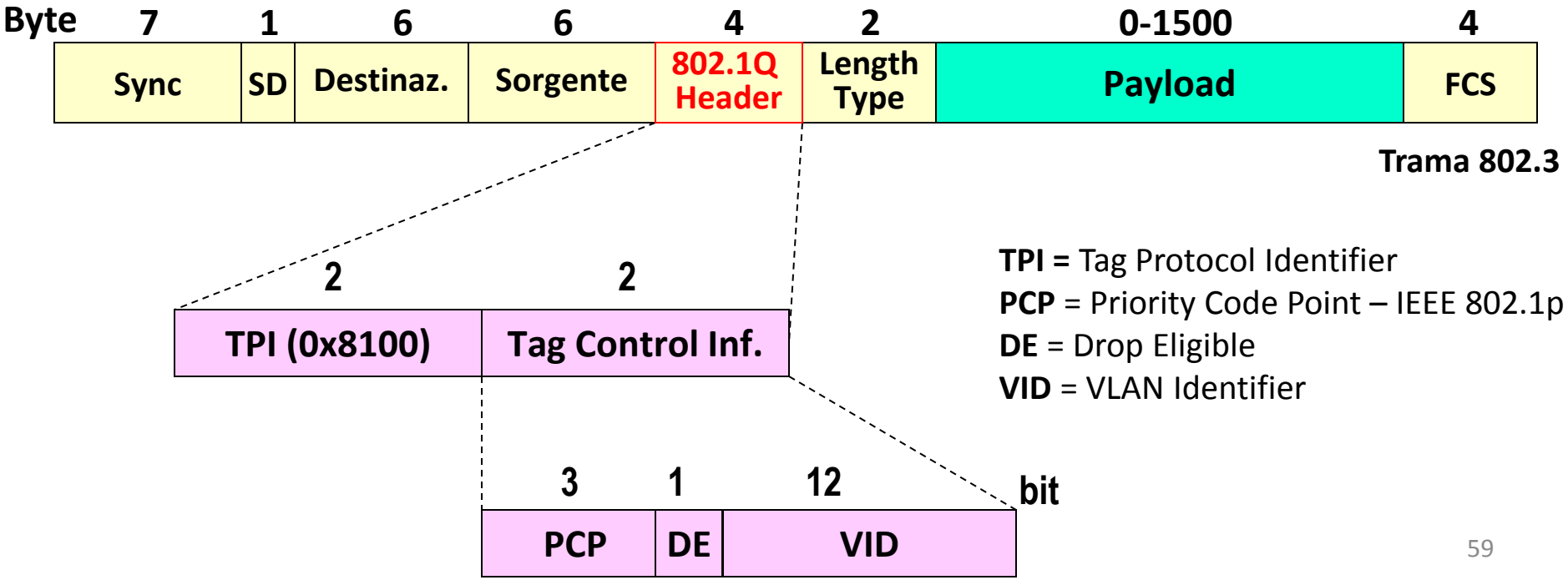



Servizio esaustivo ad ogni slot:
priorità alta, media, bassa



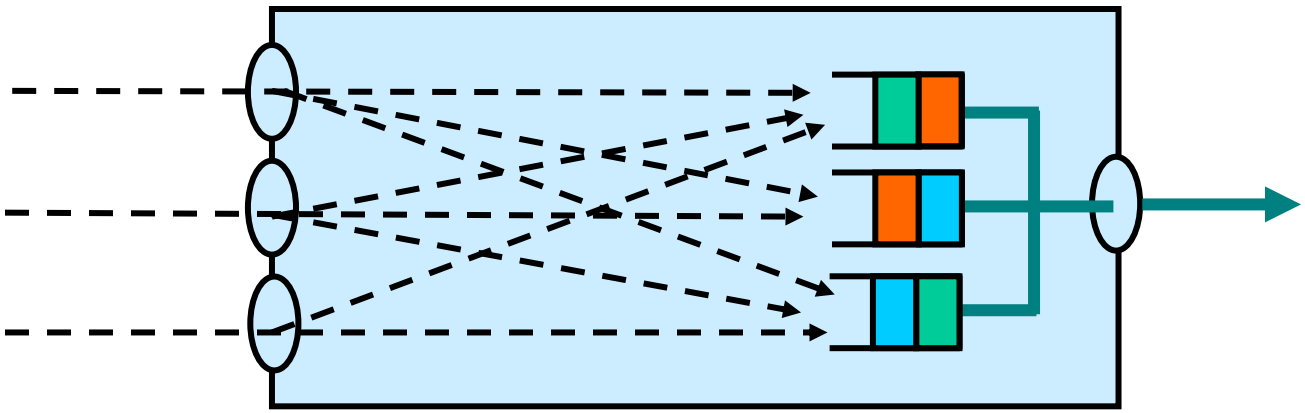
# Priorità nelle LAN

- Lo standard IEEE 802.1Q (VLAN) introduce campi di priorità anche per 802.3 (Ethernet) e 802.11 (WLAN)



# Priorità nelle LAN

- Fino a 8 priorità di servizio sono state standardizzate in 802.1Q
  - Mappano le priorità della trama MAC (si rifanno a 802.1p)
- Queste sono impostate al momento dell'incapsulamento dei livelli superiori in base
  - Alle priorità dei livelli superiori
  - Alla VLAN di appartenenza
  - Al tipo di traffico trasportato (livello 4)

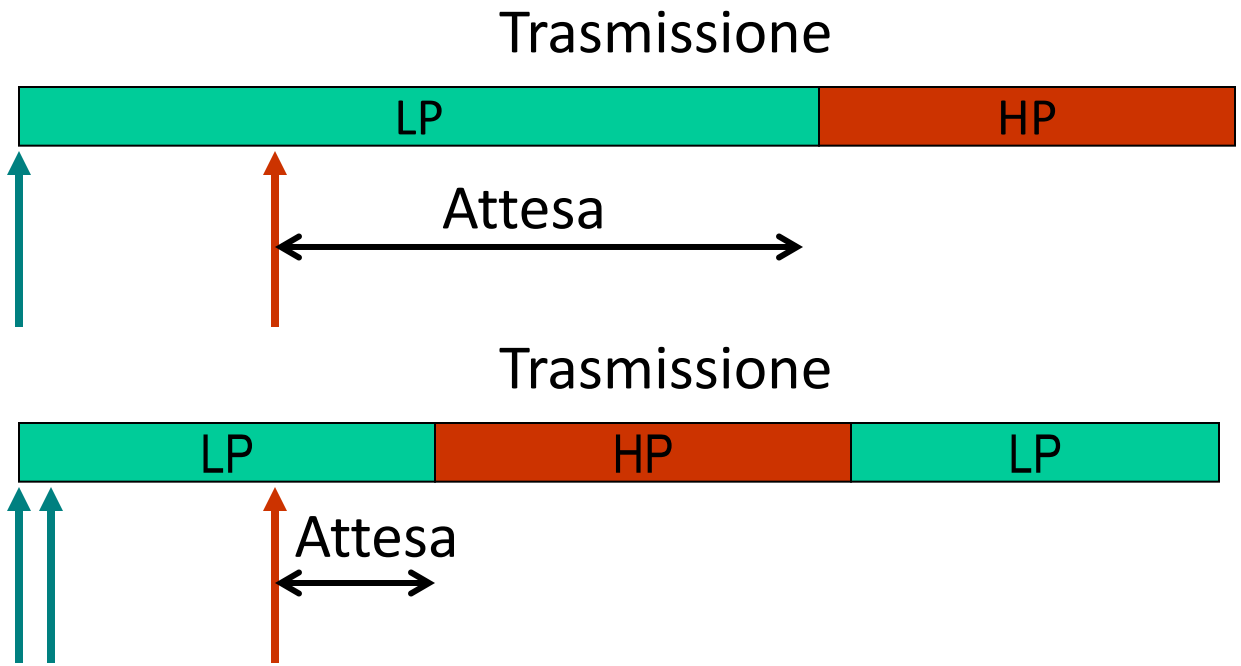


# Priorità e Scheduling: problematiche

- I pacchetti sono unità intere di lunghezza variabile
  - La suddivisione è quantizzata a pacchetti di lunghezza variabile
  - E' complicato suddividere esattamente la banda
- Priorità e scheduling vengono effettuate sulla base di pacchetti interi
  - Non si possono interrompere le trasmissioni
- Inoltre, se i pacchetti in trasmissione sono lunghi, gli altri ne subiscono il ritardo

# Priorità di Servizio: pacchetti lunghi

- Esempio
  - **Rosso** alta priorità
  - Effetto di un pacchetto **verde lungo** e di **due verdi di lunghezza metà**



# Trasmissione delle Trame

| | 1500 Byte (max Ethernet) | 40 Byte (voce) |
|---|---|---|
| **64 kb/s** | **187.5 ms** | **5 ms** |
| **128 kb/s** | **93.75 ms** | **2.5 ms** |
| **512 kb/s** | **23.4 ms** | **0.625 ms** |
| **2 Mb/s** | **6 ms** | **0.16 ms** |
| **10 Mb/s** | **1.2 ms** | **0.033 ms** |

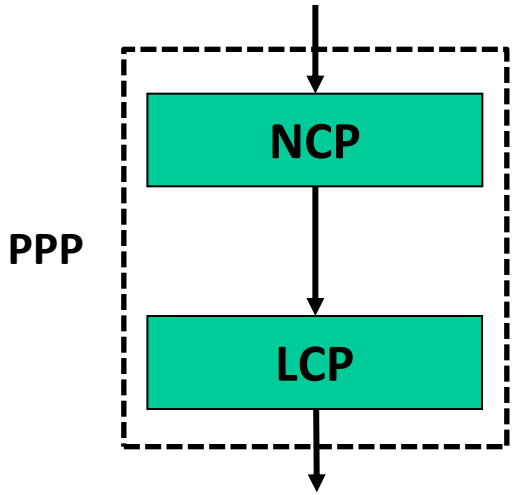
Necessità di frammentare su link poco veloci

# Modalità di Frammentazione

- Ridurre la lunghezza di tutte le trame IP porta ad enormi inefficienze
- Si potrebbero frammentare trame a livello IP solo su connessioni lente
  - In questo modo, però, IP è in grado di ricostruirle solo al destinatario
- Occorre un meccanismo a livello 2 sulle connessioni lente che possa frammentare le trame lunghe in trasmissione e che le ricomponga in ricezione
  - E' possibile utilizzare il protocollo Multilink PPP, che si basa sul protocollo PPP

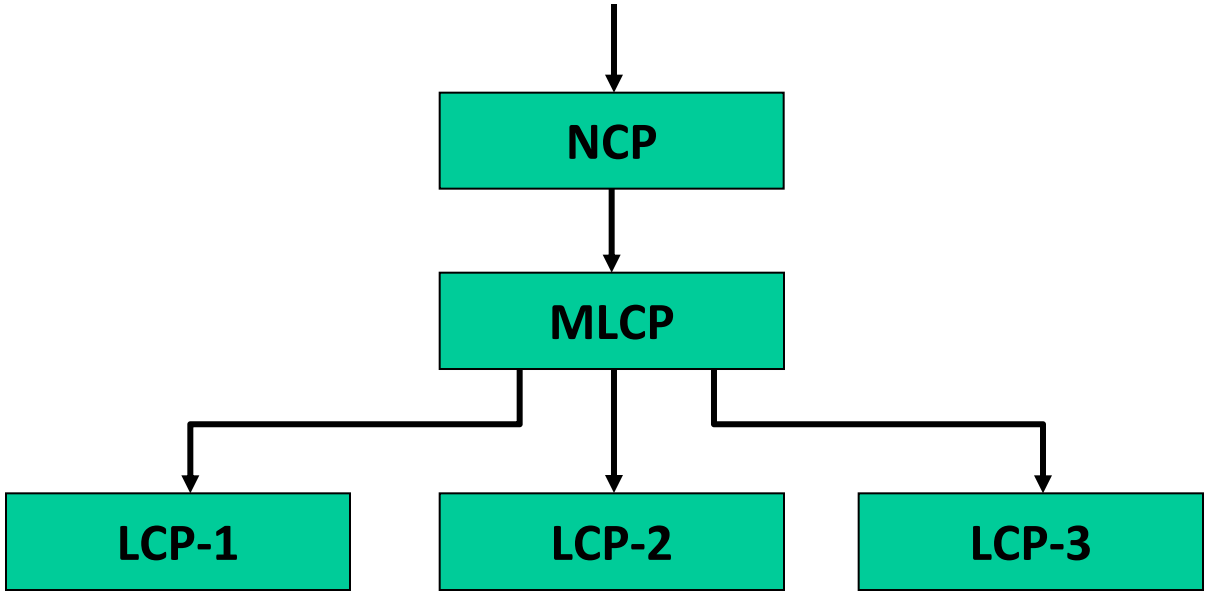
# Il protocollo PPP

- Protocollo di livello data-link per link di tipo punto-punto
  - Usato ad esempio per le connessioni tra utenti residenziali e ISP
- E' definito a sua volta da due protocolli differenti
  - *Network-Control Protocol (NCP)*: protocollo per negoziare i parametri di configurazione ottimi in base al livello di rete (consideriamo sempre IP)
  - *Link-Control Protocol (LCP)*: protocollo per inizializzare, configurare, verificare e chiudere il collegamento PPP



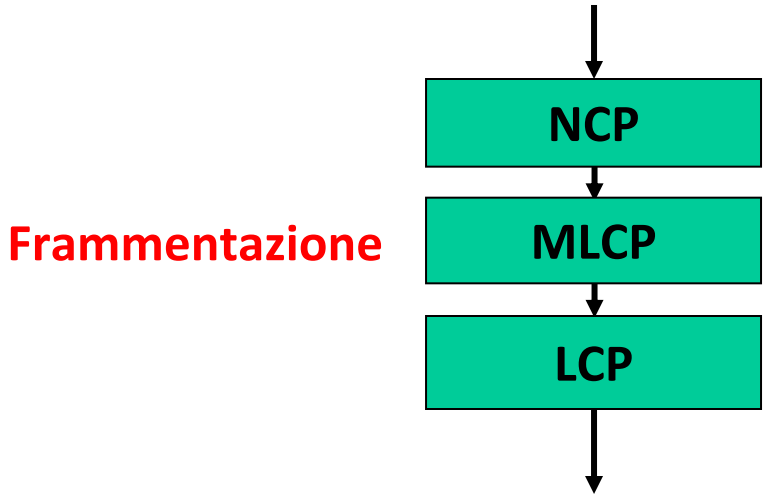
# Il protocollo Multilink PPP

- In generale, il protocollo Multilink PPP permette di «legare» multipli link PPP e fare in modo che si comportino come un singolo link PPP



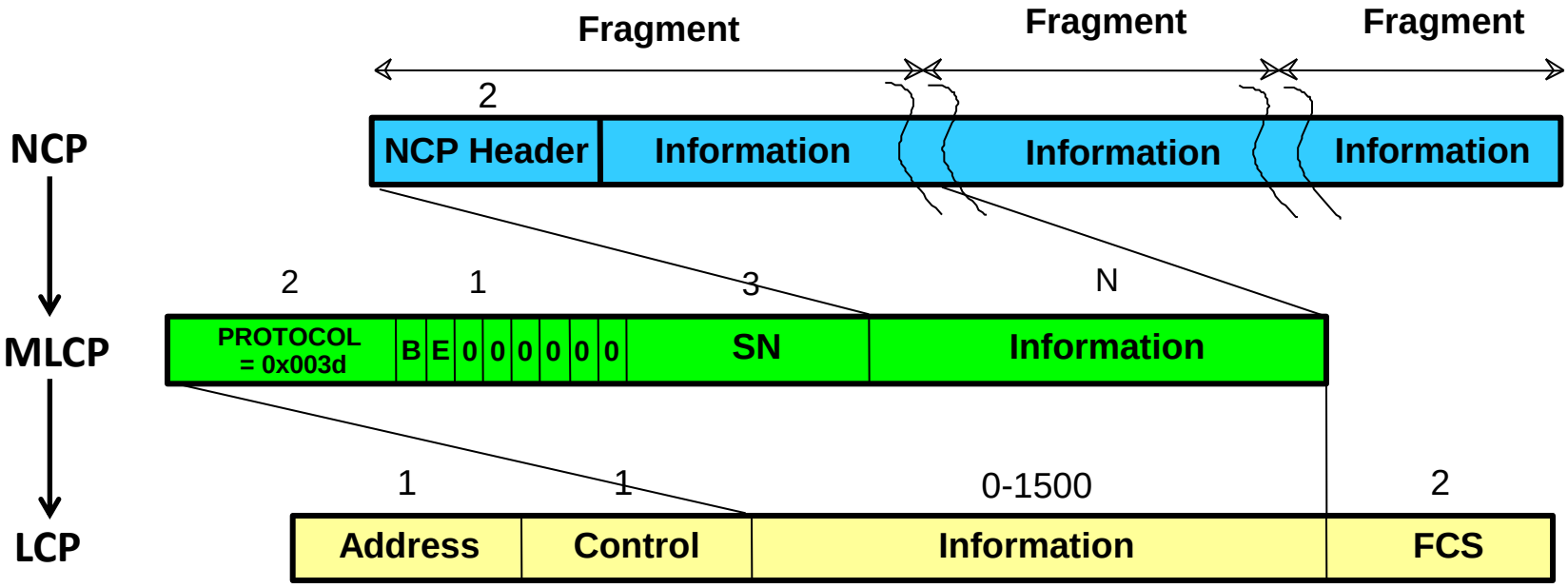
# Il protocollo Multilink PPP

- Multilink PPP prevede la frammentazione
- Al fine di frammentare e ricombinare lato ricezione del collegamento punto-punto i pacchetti, viene usato Multilink PPP
  - In questo caso Multilink PPP viene utilizzato non per «legare» multipli link PPP ma esclusivamente per la sua proprietà di frammentazione

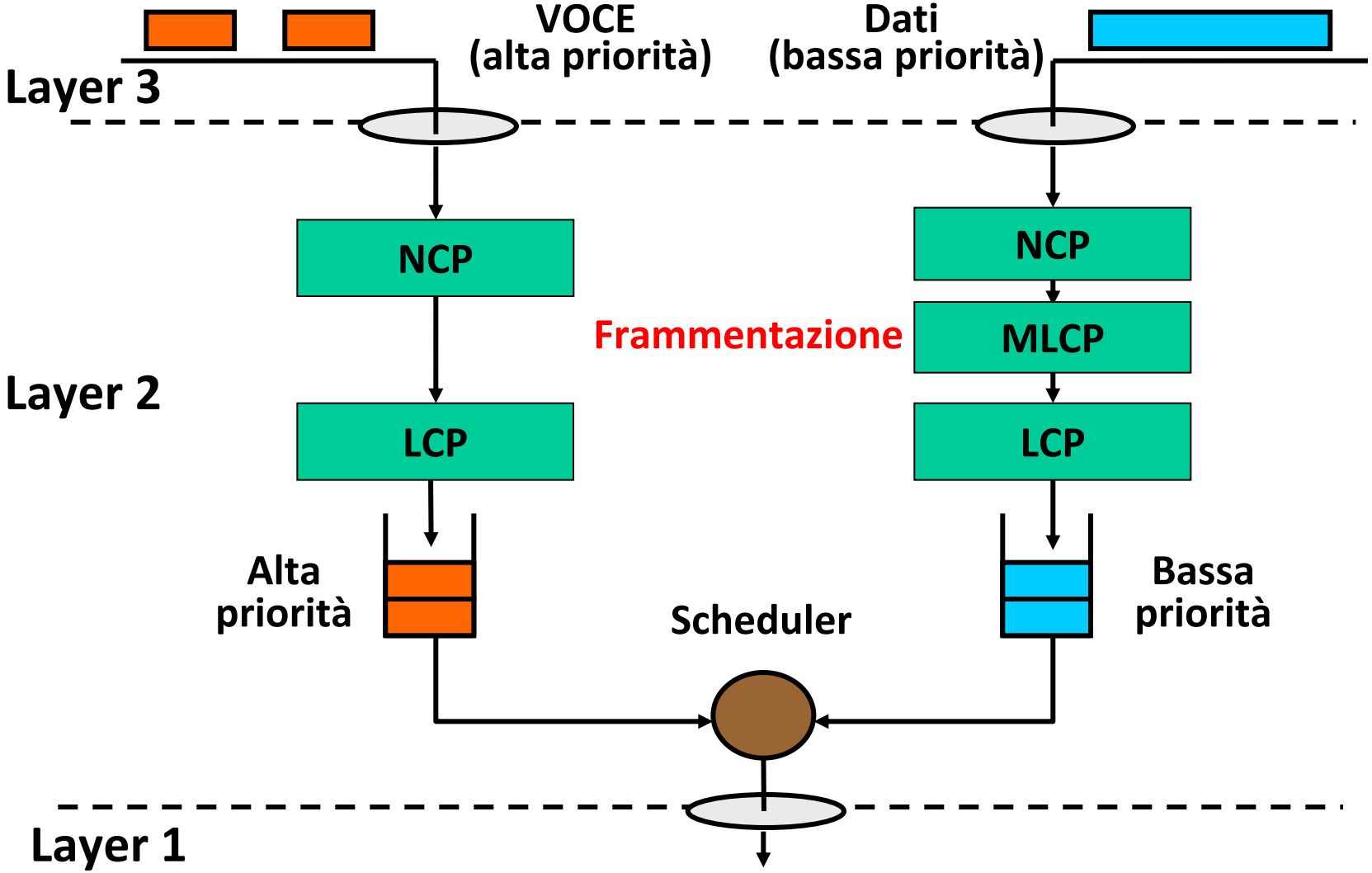


# Il protocollo Multilink PPP

- Gli strati LCP e MLCP incapsulano nella loro trama l'informazione proveniente dal sotto-strato superiore
- Lato ricezione del link PPP si effettua l'operazione di ricomposizione dell'informazione, ricombinando i frammenti



# Utilizzo dei protocolli PPP/Multilink PPP per QoS



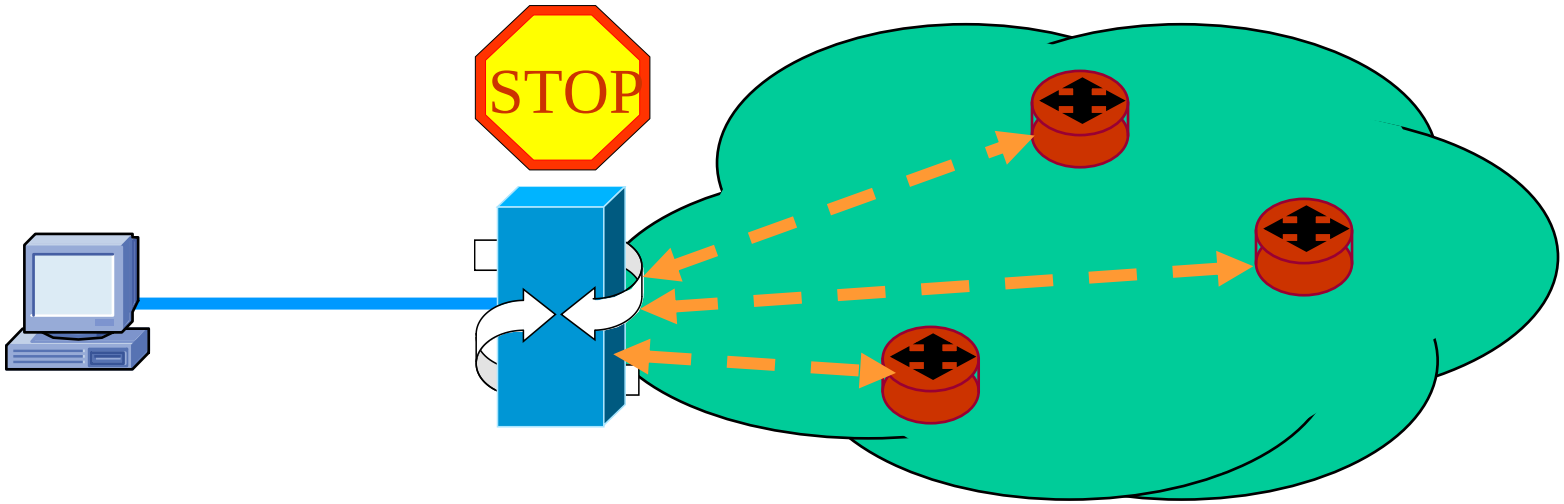
# IL CONTROLLO D'ACCESSO

# Call Admission Control

- Insieme delle azioni intraprese da una rete durante la fase di instaurazione o ri-negoziazione di una connessione, allo scopo di stabilire se la richiesta possa essere accettata
- Lo scopo della procedura di CAC è assicurare che sul cammino tra sorgente e destinazione seguito dal flusso in rete siano assegnati, su ogni link e nodo del cammino
    1. Una porzione della capacità del link (Bandwidth assignment)
    2. Una porzione del buffer del nodo (Buffer assignment)
- La quantità di banda e buffer allocata per un flusso dipende dai requisiti di QoS

# Call Admission Control

- La procedura di CAC deve conoscere
  - La quantità di risorse richiesta
  - La quantità di risorse usata su ogni link/nodo
- Quindi deve
  1. Verificare se esiste un cammino con tale disponibilità
  2. Prenotare le risorse in caso affermativo



# Modalità di calcolo e prenotazione

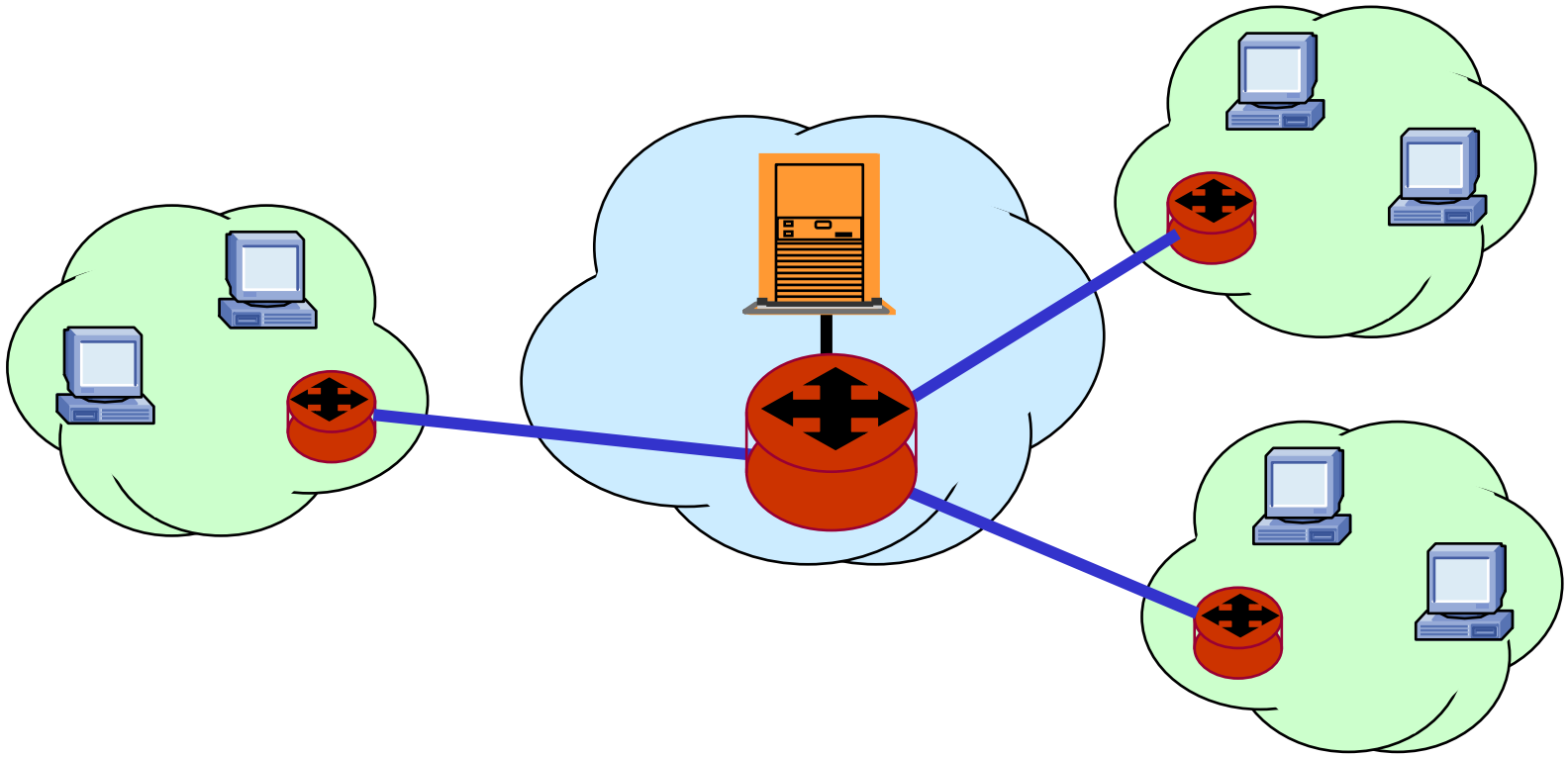
- Le modalità di calcolo e prenotazione possono essere suddivise in tre gruppi
  - **Centralizzata**: Il CAC viene effettuato da un server centrale
  - **Distribuita**: Ogni nodo concorre al CAC
  - **Mista**: Tecnica che consente il CAC solo ai nodi d'ingresso della rete

# Modalità Centralizzata

- Il CAC viene effettuato da un server centrale
  - Il server riceve tutte le richieste e conosce tutte le informazioni necessarie
  - La segnalazione verso i nodi è semplificata
  - Può essere determinato il cammino ottimale
- Svantaggi
  - Problemi di scalabilità, affidabilità e velocità
  - Non è ben tollerato da IP (se non per sistemi semplici)
    - Per il cammino ottimale occorre poter decidere e segnalare l'instradamento

# Modalità Centralizzata

- Adatta a piccoli sistemi di rete



# Modalità Distribuita

- Ogni nodo conosce le risorse occupate in rete a lui relative e concorre al CAC
- Caratteristiche
  - E’ un sistema robusto e affidabile, ma complesso
  - Occorre un sistema di segnalazione per
    - Raccogliere la disponibilità dei nodi
    - Prenotare le risorse, ad esempio mediante il protocollo Resource ReSerVation Protocol (RSVP, RFC 2210) o, nella telefonia, SS7

# Modalità Mista

- Il CAC è effettuato dai router posti all'ingresso della rete
  - E' un sistema misto, in cui tali router ottengono periodicamente informazioni sullo stato delle risorse in rete
- Anche in questo caso occorrono
  - Un sistema di segnalazione per raccogliere periodicamente lo stato di occupazione dei nodi
  - Un sistema di segnalazione per prenotare le risorse (RSVP)

# IL CONTROLLO DELLE RISORSE

## IntServ, DiffServ

# Meccanismi di controllo delle risorse

- Il controllo delle risorse per offrire QoS deve utilizzare meccanismi scalabili e distribuiti
- Esistono due approcci che si differenziano per le garanzie che offrono e la complessità che presentano
  - Integrated Services (**IntServ**)
  - Differentiated Services (**DiffServ**)

# INTEGRATED SERVICES

## IntServ

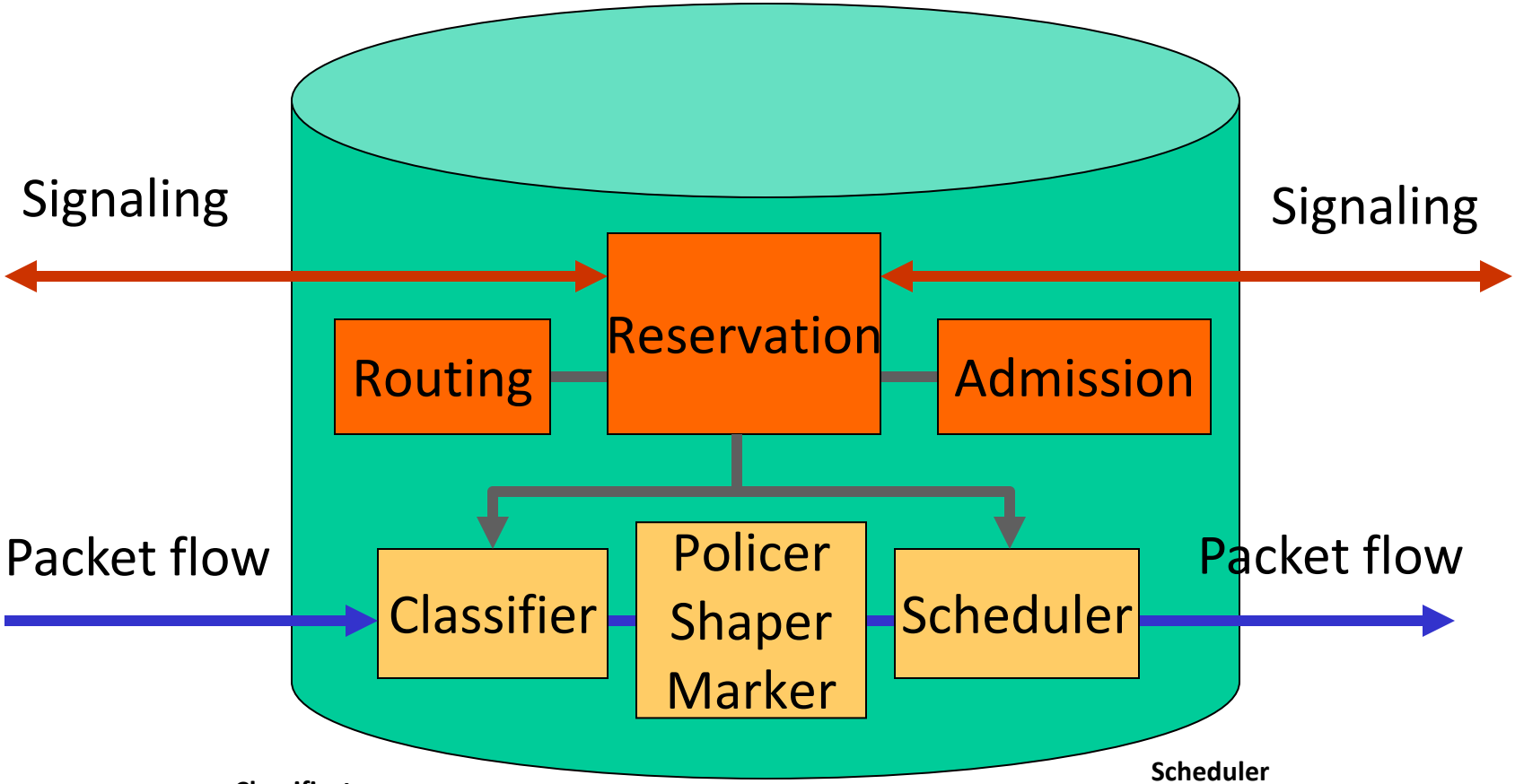
# Integrated Services

- Introducono in IP una modalità di controllo dinamica e assoluta (RFC 2215) di ciascun flusso d'utente
- Offrono due classi di servizio oltre alla classe Best-Effort
  - Guaranteed Service, GS (RFC 2212)
    - Emula il servizio a circuito con ritardi garantiti
  - Controlled Load Service, CLS (RFC 2211)
    - Emula la modalità Best-Effort ma in una rete non congestionata
- Viene utilizzato il protocollo RSVP per prenotare le risorse
- Svantaggi
  - Sono necessarie modifiche sostanziali all'architettura dei router
  - L'utilizzo di un protocollo di prenotazione delle risorse come RSVP lo rende un sistema complesso

# Modalità assoluta di controllo e assegnazione delle risorse

- E’ il paradigma da sempre utilizzato in telefonia (Call Admission Control)
  - Sistema di prenotazione flusso per flusso (fine-grained) a blocco
    1. Le risorse vengono richieste dall’utente
    2. Se esistono vengono prenotate
    3. Se non esistono la richiesta è rifiutata
- Miglioramenti introdotti con IP
  - Diverse classi di servizio (diversificazione delle risorse richieste)
  - Riutilizzo di risorse prenotate ma non usate

# IntServ: funzionalità dei router




**Classificatore**
Identifica il flusso a cui appartiene il pacchetto in arrivo

**Shaper**
Effettua la regolazione del traffico (policing/shaping/marking)

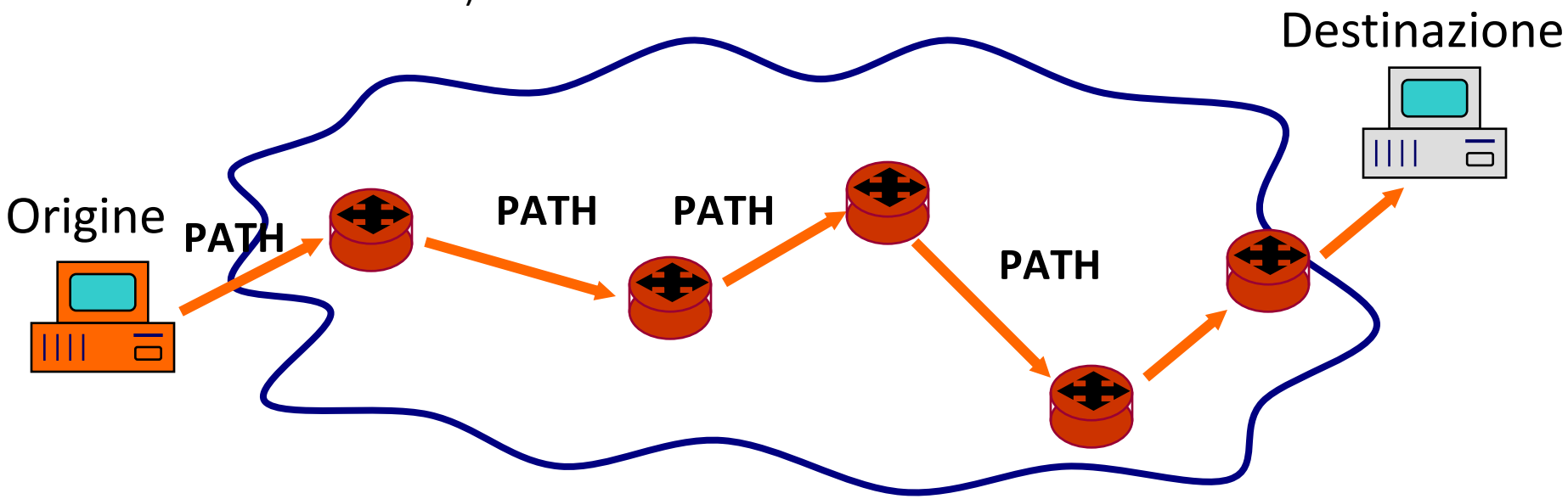
**Scheduler**
Seleziona il pacchetto che deve essere inviato per primo sul link di uscita

# IntServ: prenotazione risorse

- Viene utilizzato il protocollo RSVP
  - Trasmette ai router sul cammino informazioni di richiesta di risorse e QoS relative a ogni flusso
  - I router valutano se le richieste possono essere accettate
  - Se sì, memorizzano le richieste accettate in modo da poterle soddisfare all'arrivo dei flussi
- Lo stato nei router ha una validità limitata nel tempo (soft state)

# RSVP – PATH MESSAGE

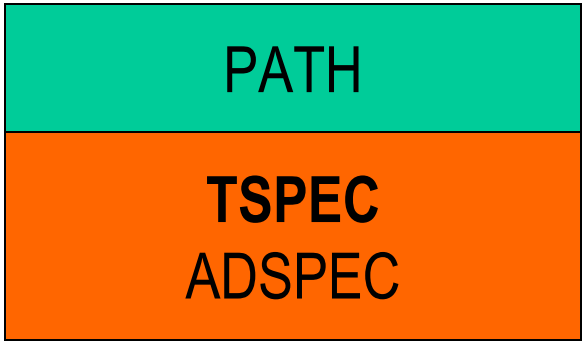
- Fissa il cammino su cui deve essere effettuata la riservazione (secondo le regole dell'instradamento IP)




- La sorgente invia un messaggio di PATH verso la destinazione
  - Messaggi PATH sono emessi periodicamente per consentire l'eventuale rivelazione di variazioni nell'instradamento
  - In tal caso è necessario effettuare nuovamente la riservazione delle risorse sul nuovo cammino
- Ciascun router che riceve un messaggio di PATH mantiene un «*path state*», contenente l'indirizzo del nodo precedente attraversato dal flusso, i parametri relativi alle caratteristiche del flusso a cui PATH si riferisce, l'interfaccia in ingresso e l'interfaccia in uscita

# RSVP – PATH MESSAGE

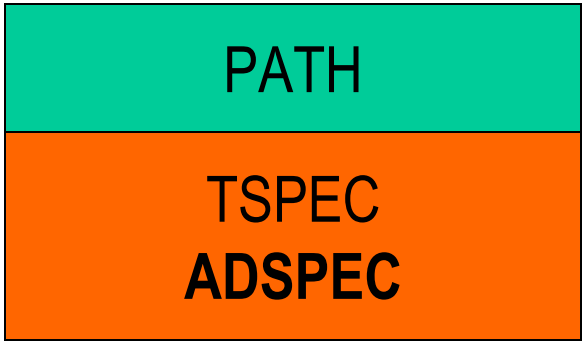
- Informa i router sulle caratteristiche del flusso




- TSPEC (Traffic SPECification) contiene la descrizione del traffico generato dalla sorgente (ad es. attraverso i parametri del token bucket)
- TSPEC non può essere modificato dai router

# RSVP – PATH MESSAGE

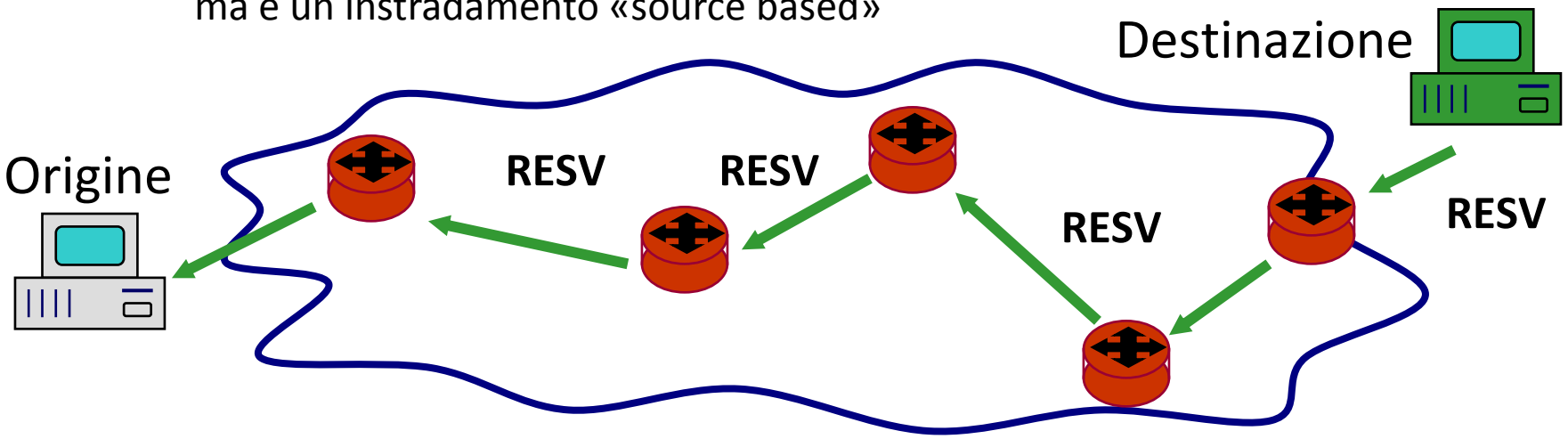
- ADSPEC (ADvertising SPECification) raccoglie informazioni preliminari riguardo la QoS sul cammino




- ADSPEC è esaminato e opportunamente modificato ad ogni hop
  - Sintetizza informazioni sulla QoS conseguibile sul cammino (es. se ci sono router non RSVP-compliant, se alcune classi di servizio non sono supportate etc.)

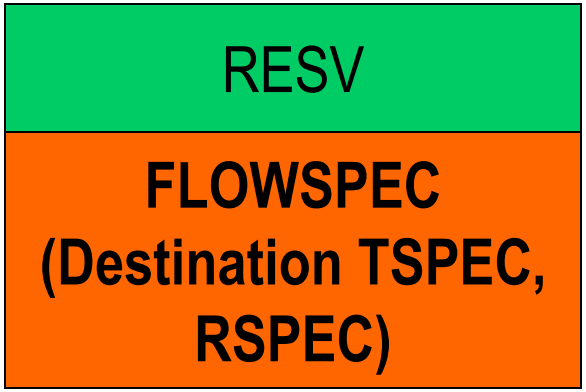
# RSVP – RESV MESSAGE

- La destinazione emette il messaggio RESV verso l'origine per effettuare le richieste di riservazione
- Il messaggio RESV segue in verso opposto lo stesso cammino seguito dal messaggio PATH («Reverse Path» è indicato nei messaggi PATH)
  - L'instradamento dei messaggi RESV non segue le regole dell'instradamento IP, ma è un instradamento «source based»




- Sulla base dei campi TSPEC e ADSPEC ricevuti nel messaggio PATH, la destinazione stabilisce che richiesta di riservazione di risorse effettuare (riportata nel campo FLOWSPEC)

# RSVP – RESV MESSAGE



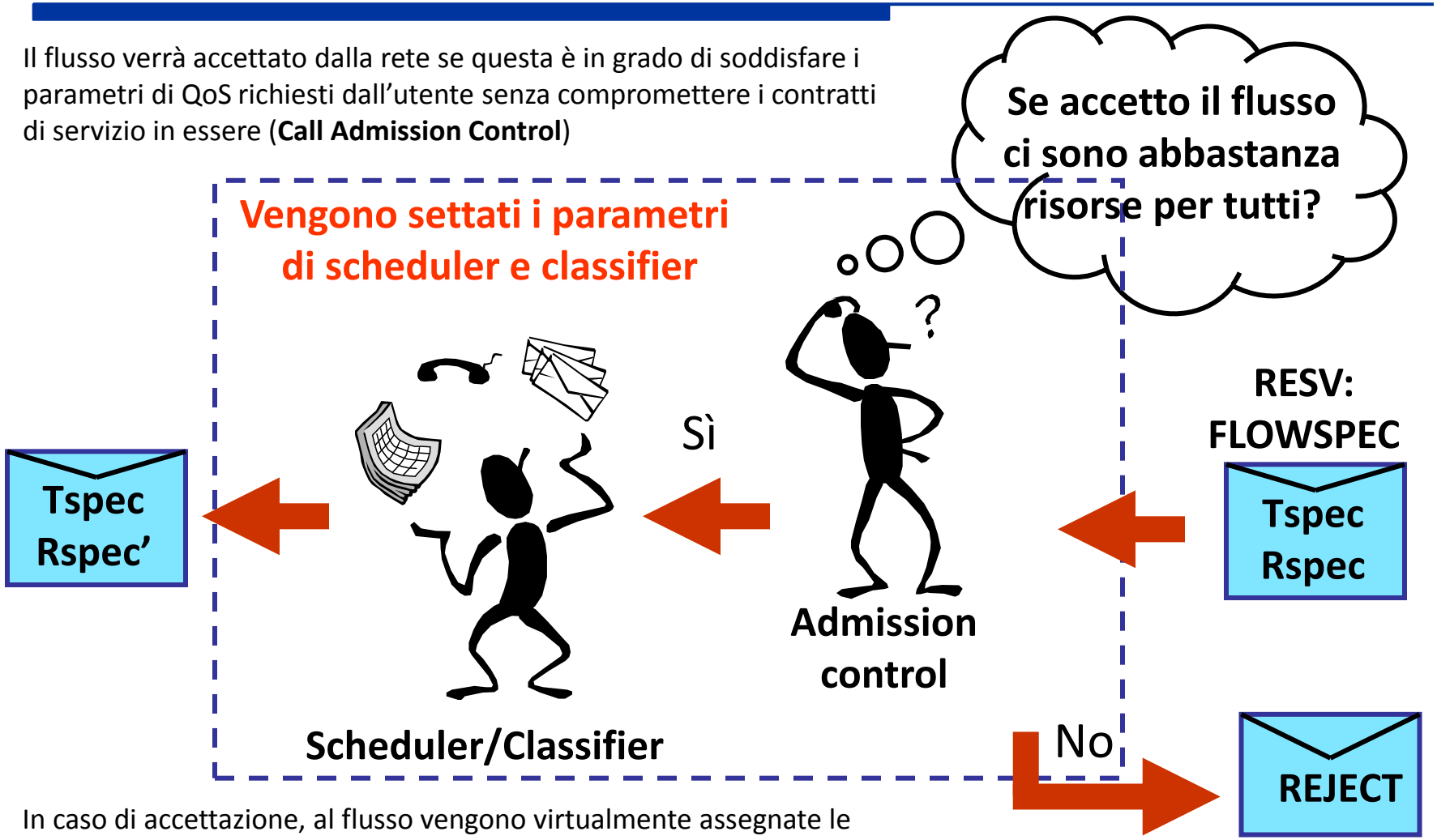

- FLOWSPEC (FLOW SPECification) stabilisce la richiesta di riservazione da effettuare, ed è composto dai campi TSPEC e RSPEC (Reservation SPECification)
  - In TSPEC sono contenute le descrizioni del traffico, eventualmente cambiate dal destinatario
  - In RSPEC è contenuta la descrizione del tipo di servizio desiderato, specificando le garanzie che devono essere offerte al traffico

# RSVP – RESV MESSAGE
# Call Admission

- Viene effettuata passo a passo dai router
  - Ricevono in input informazioni sul flusso
    - **RSPEC:** definisce la QoS desiderata
    - **TSPEC:** descrive il flusso dati
  - Conoscono l'occupazione delle risorse dovuta a chiamate già accettate
    - Parametri di traffico «a priori» delle chiamate accettate
    - Misurazioni della effettiva occupazione delle risorse
  - Determinano le risorse da assegnare alla nuova chiamata
    - Accetta il nuovo flusso se le risorse esistono
      - Determinano nuovi RSPEC da inviare a monte
    - Altrimenti rigetta la richiesta di riservazione delle risorse (e quindi il flusso)

# RSVP – RESV MESSAGE
# Decisione al router

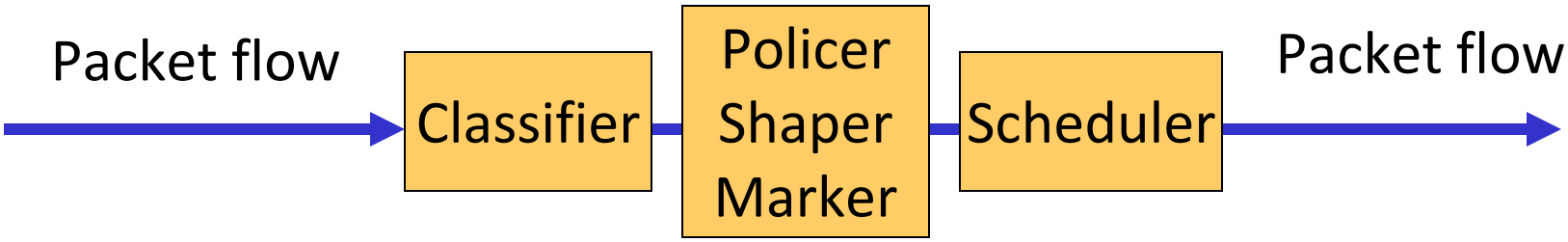
Il flusso verrà accettato dalla rete se questa è in grado di soddisfare i parametri di QoS richiesti dall'utente senza compromettere i contratti di servizio in essere (**Call Admission Control**)




In caso di accettazione, al flusso vengono virtualmente assegnate le risorse di rete (banda, buffer) necessarie al soddisfacimento dei requisiti di QoS (**Resource Allocation**)

# Controllo del traffico

- I *router di frontiera* devono effettuare regolazione del traffico (policing, shaping, marking) sui parametri dichiarati
- I *router interni* effettuano
  - Classificazione dei pacchetti di ciascun flusso
  - Regolazione del traffico
  - Scheduling basato sulla QoS richiesta

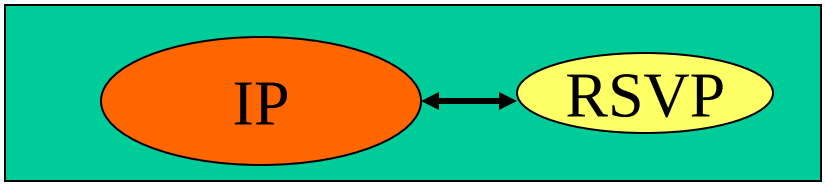


# Soft State

- Lo stato nei router ha una validità limitata nel tempo e controllata da un timer
- La sorgente e la destinazione inviano quindi periodicamente nuovi messaggi di PATH/RESV
- Inoltre, se l'instradamento cambia, il ricevente deve riservare esplicitamente le risorse sul nuovo cammino
- Vantaggi
  - Recupero da situazioni di errore
  - Tolleranza alla perdita di pacchetti di segnalazione
  - Buona adattabilità a cambiamenti del cammino tra sorgente e destinazione (nel caso di routing dinamico)
- Svantaggio: segnalazione pesante (aumento del traffico di segnalazione in rete)

# Il protocollo RSVP

- E’ un protocollo che si colloca a livello 3, anche se usa i servizi stessi di IP

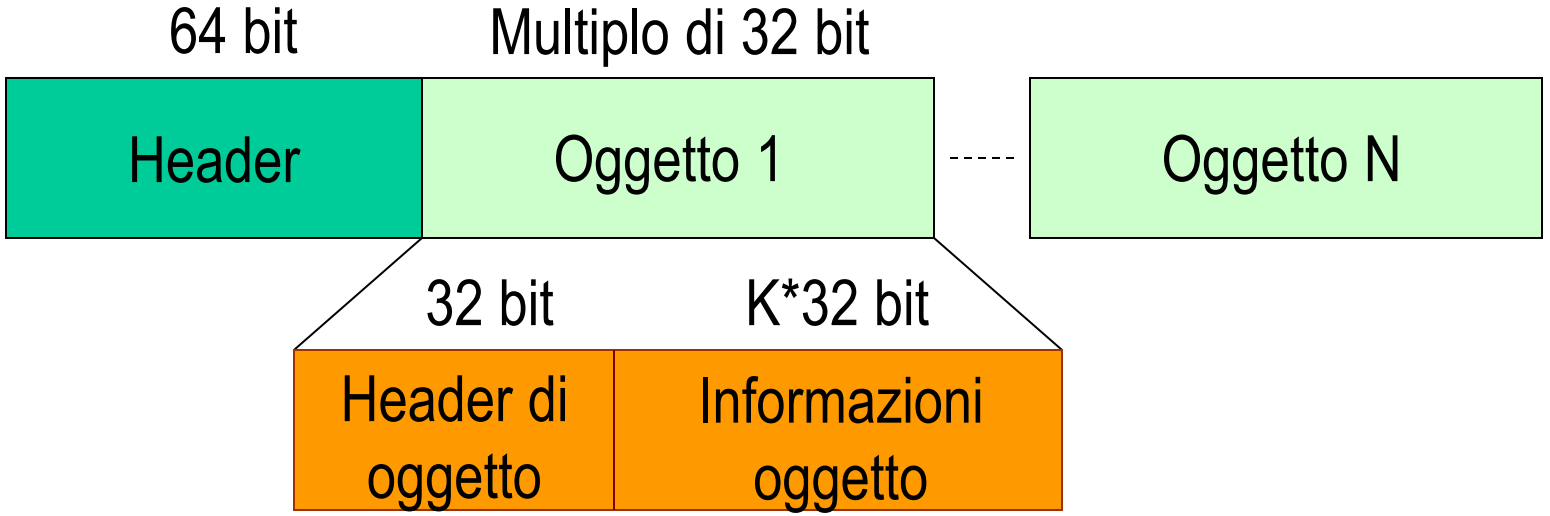



- I messaggi RSVP vengono incapsulati all’interno di un pacchetto IP con *ID Protocol* pari a 46
- E’ richiesto che venga limitata la congestione dei pacchetti di segnalazione
  - Assegnazione di una quantità di banda per la segnalazione
  - Oppure utilizzo di meccanismi a priorità

# Messaggi RSVP

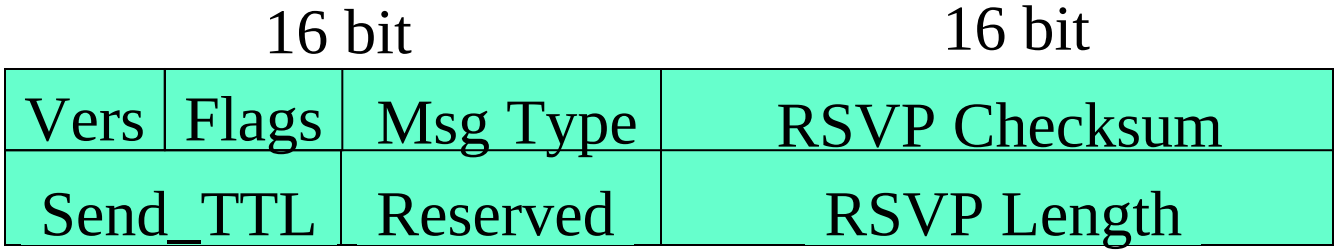
- PATH: messaggio iniziale sul cammino di andata
- RESV: messaggio di prenotazione (reservation) sul cammino di ritorno
- PATH ERR: comunica l'identificazione di errori durante l'elaborazione del messaggio di PATH
- RESV ERR: indica il fallimento della fase di prenotazione
- PATH TEAR: abbattimento dello stato instaurato nei router da un messaggio di PATH
- RESV TEAR: abbattimento dello stato instaurato nei router da un messaggio di RESV
- RESV CONF: messaggio inviato alla destinazione per confermare la riuscita della fase di prenotazione delle risorse

# Formato dei messaggi RSVP




- Ciascun oggetto raggruppa informazioni di carattere simile trasportate dal messaggio
  - Caratterizzazione del traffico
  - Dati di regolazione del traffico
  - …

# Header dei messaggi RSVP




- *Vers* indica la versione del protocollo (attualmente 1)
- *Flags* non è ancora specificato
- *Msg Type* contiene l'identificatore del tipo di messaggio (PATH, RESV, ecc)
- *RSVP Checksum* verifica l'integrità del contenuto del messaggio
- *Send TTL* (tramite il confronto con il TTL dell'header IP permette di individuare l'attraversamento di nodi non RSVP)
- *RSVP Length* indica la lunghezza del messaggio

# Formato degli oggetti RSVP


- *Length* indica la lunghezza dell'oggetto in byte
- *Class_NUM* identifica un tipo di oggetto (Es., ADSPEC, SENDER_TSPEC, FLOWSPEC, etc.)
- *C_Type* identifica univocamente il tipo di formato usato per un tipo di oggetto (1 ➔ IPv4, 2 ➔ IPv6)

# Assegnazione di risorse

- Il protocollo RSVP nell'architettura IntServ permette l'assegnazione e riservazione delle risorse mediante meccanismi di call admission control per due classi di servizio
  - Guaranteed Service (GS)
  - Controlled Load Service (CLS)

# Guaranteed Service (RFC 2212)

- Emula il servizio a circuito con ritardi garantiti
- Garantisce
  1. Perdita nulla nei buffer di trasmissione
  2. Un upper bound al ritardo

# Guaranteed Service (RFC 2212)

- TSPEC definisce le caratteristiche del traffico e contiene
  - I parametri del token bucket
    - Bucket size $k$ [bit]
    - Token rate $r$ [bit/s]
    - Peak rate $p > r$ [bit/s]
- ADSPEC contiene parametri che permettono lato ricezione di stimare il ritardo end-to-end in rete
- RSPEC comprende i seguenti parametri
  - Banda $B$ [bit/s] da riservare
  - Slack term (o termine di correzione) $S$ [$\mu s$]

# Guaranteed Service (RFC 2212)

- Il ricevitore sulla base di ADSPEC determina la banda $B_j$ che i router devono assegnare al flusso $j$ e lo Slack term $S$

  $S$ = ritardo richiesto - ritardo massimo con la riservazione di banda $B_j$

- Il ricevitore invia $B_j$ e $S$ come RSPEC
- Se $S$ è positivo, i router a monte possono ridurre $B_j$ e $S$, quindi aggiornare RSPEC, oppure non modificare tali valori
- Se la banda $B_j$ non è disponibile la chiamata viene rifiutata
- Costa in complessità negli apparati dei router (classifier, regulator, scheduler) che devono gestire flusso per flusso
- E' il più adatto per la telefonia e per l'emulazione di circuiti

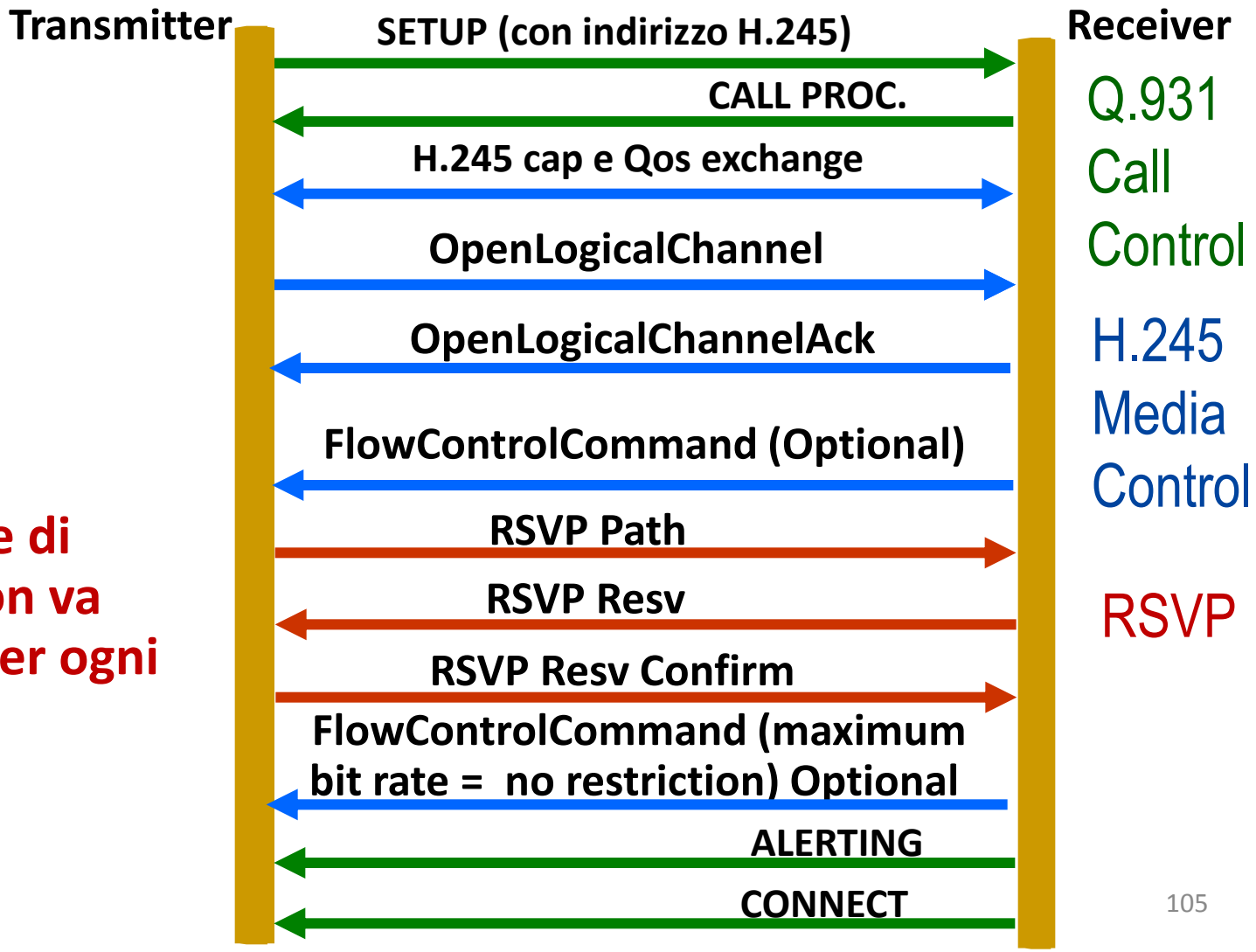
# Controlled Load Service (RFC 2211)

- Offre un servizio che emula il «best effort» in una rete  non congestionata
- Non garantisce ritardi minimi
- La gestione nei nodi è più semplice
- Lascia ampio spazio a varianti di implementazione

# Controlled Load Service (RFC 2211)

- TSPEC contiene la descrizione del token bucket
- ADSPEC è privo di parametri QoS
  - Tali parametri non sono definiti a priori
  - Il ritardo end-to-end non viene stimato
- FLOWSPEC non contiene RSPEC
- Ogni router decide la banda assegnata $B_j$ necessaria a garantire il servizio indipendentemente, in base a una politica predeterminata

# Riservazione delle risorse con RSVP in H.323




**NB: La fase di Reservation va eseguita per ogni canale**

# Integrated Services: problemi

- Richiedono di mantenere nei router uno stato per ciascun flusso
  - Sia appartenente alla classe di servizio Guaranteed Service che Controlled Load Service
- Scarsa scalabilità
- Segnalazione pesante
- Forte impatto sull'architettura di rete esistente
- Adatto per reti di piccole dimensioni

# DIFFERENTIATED SERVICES

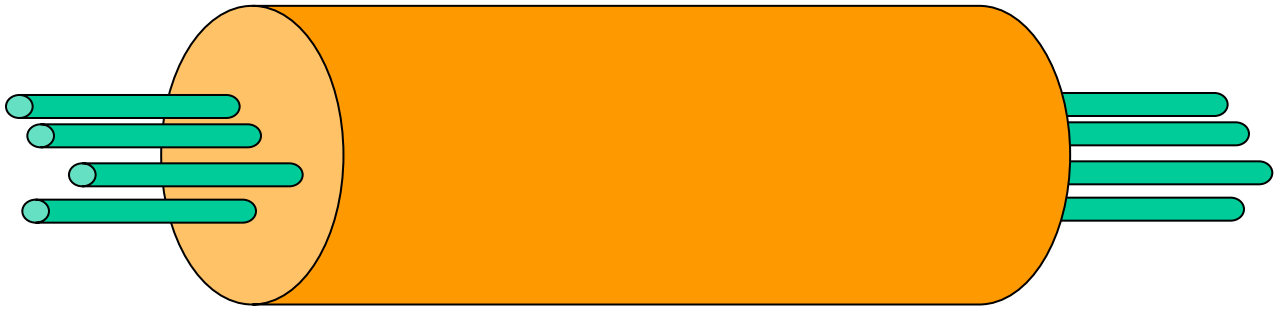
## DiffServ

# Differentiated Services

- Perseguono una soluzione più semplice, scalabile e di basso costo
- Si rinuncia al controllo stretto flusso per flusso nei router (tecnica coarse-grained)
- All'interno di ogni singolo router si dimensiona la classe di servizio collettivamente
- Se necessario, la riservazione di risorse può ancora essere effettuata per singoli flussi attraverso RSVP

# Microflussi senza controllo

- I microflussi non sono controllati singolarmente
- Eventuale traffico eccedente sottrae QoS a tutti i flussi appartenenti alla medesima classe di servizio



- A livello microflusso occorre una gestione esterna a DiffServ, che consenta di controllare il traffico immesso in rete

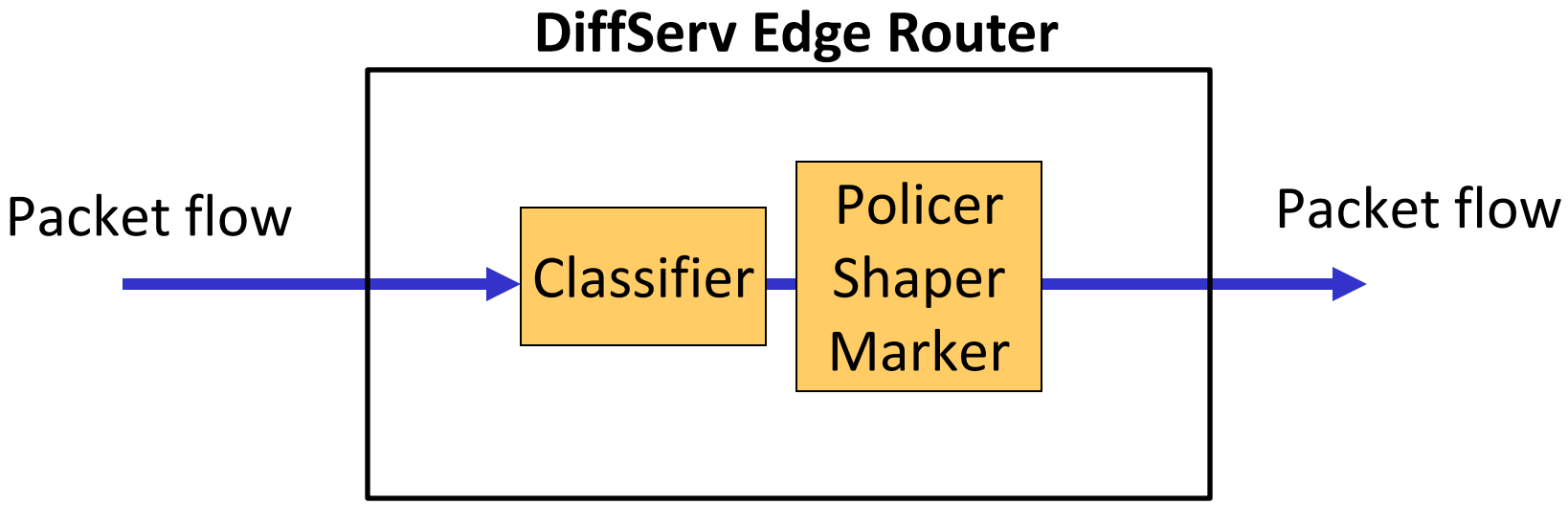
# Differentiated Services (RFC 2475)

- Le operazioni complesse e il controllo dei microflussi vengono effettuate esclusivamente ai bordi della rete
- I router interni alla rete devono essere solo in grado di applicare forwarding differenziati a poche differenti classi di traffico
  - Non richiedono modifiche significative, ed in particolare non necessitano di mantenere uno stato per flusso o gestire segnalazione
  - Sono necessarie code diverse con priorità diverse per la differenziazione del servizio

# Differenziazione dei flussi

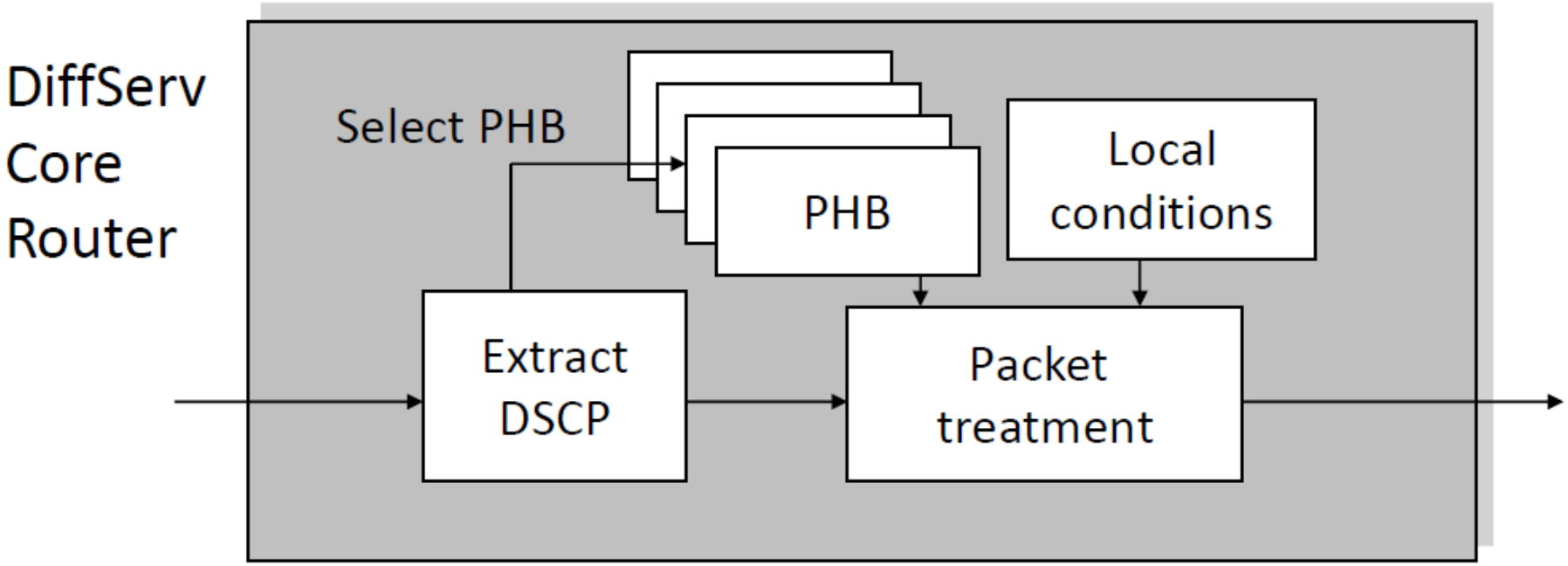
- Viene usato il campo (DS) dell'header IP
  - Corrisponde al TOS (*Type Of Service*) byte di IPv4 o al *Traffic Class* byte di IPv6
- 6 bit sono utilizzati per specificare il *Differentiated Service Code Point* (DSCP), mentre i rimanenti due bit non sono utilizzati
- Il DSCP viene interpretato dai router di core per selezionare il tipo di servizio (o Per-Hop-Behavior, PHB) da applicare (Es. 0X00 indica best effort forwarding)

# Diffserv in Edge Router




- Gli Edge Router (router di frontiera) verificano la coerenza con gli SLA di accesso
  - Si ha una regolazione del traffico a livello microflussi
  - Viene definito il DSCP del pacchetto IP

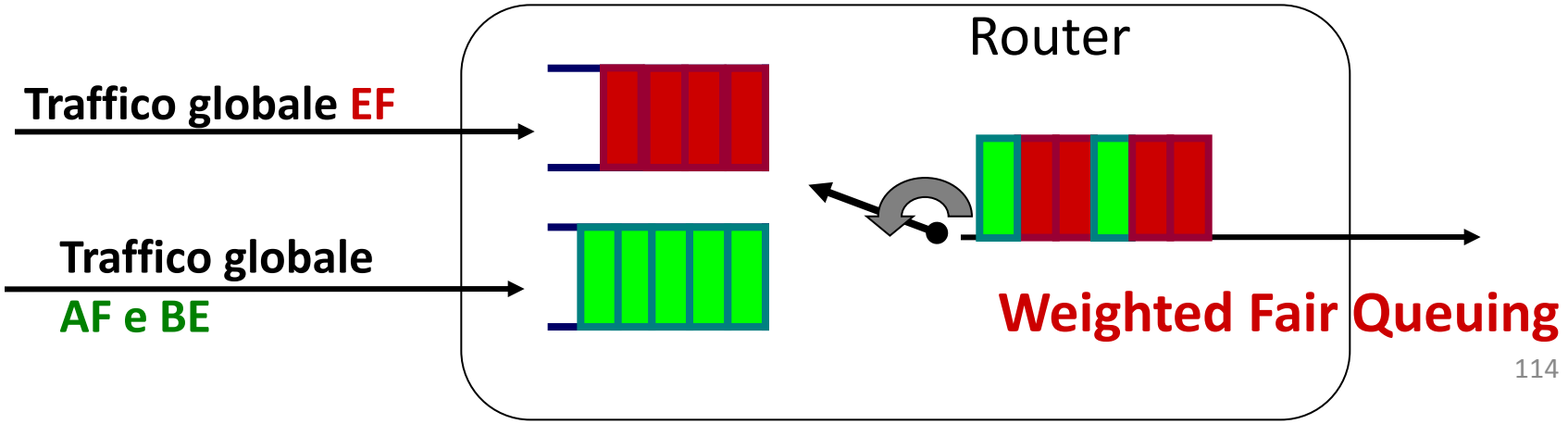
# Diffserv in Core Router




- I router di core gestiscono aggregati di traffico e trattano i flussi secondo i PHB selezionati

# Possibili PHB

- Alcuni tra i più importanti Per Hop Behavior (PHB) sono
  - **Expedited Forwarding (EF)** per applicazioni a basso ritardo, bassa latenza e basso jitter
  - **Assured Forwarding (AF)** service per applicazioni che richiedono affidabilità di consegna
  - **Best Effort (BE)** service, nessuna garanzia



# Expedited Forwarding (RFC 2598)

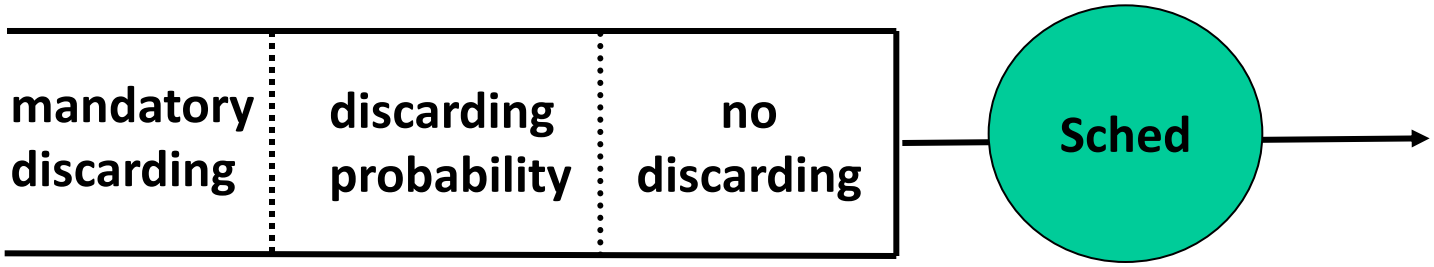
- Expedited Forwarding vuole emulare una linea dedicata
  - Gli utenti richiedono una quantità di banda per la comunicazione tra due punti
  - Se il traffico non eccede quanto stabilito (no violazione TCA), i ritardi e la percentuale di pacchetti persi devono essere molto bassi
  - Il traffico va condizionato e controllato, quello in eccesso rispetto a TCA non viene immesso in rete
  - DSCP: 101110

# Assured Forwarding (RFC 2597)

- Assured Forwarding fornisce 4 livelli differenti di priorità, ovvero di garanzia di forwarding (in termini di banda e spazio nelle code)
- Si propone di evitare situazioni di congestione mediante uno scarto preventivo e differenziato dei pacchetti
  - Si introducono 3 diversi livelli di probabilità di scarto per ognuna delle 4 classi di qualità definite in precedenza

# Assured Forwarding

- Lo scarto dei pacchetti deve avvenire in modo controllato
- Una possibilità è usare RED (con parametri diversi per ciascuno dei tre livelli di scarto)
- Si contrappone al «tail drop», ovvero lo scarto incontrollato dei pacchetti quando la coda è piena




**RED (Random Early Discarding)**

# Random Early Discarding