Università di Genova
Facoltà di Ingegneria

Internet Quality of Servises (IP-QoS): Differentiated Services

Prof. Raffaele Bolla

Reti integrate nei servizi

- Uno degli obiettivi delle reti di telecomunicazioni nell'ultimo decennio è stato l'integrazione dei servizi: creare una rete in grado di supportare ogni tipo di servizio (previsto e prevedibile)
- Un rete integrata permetterebbe una elevata efficienza e renderebbe appetibile l'offerta dnamica di nuovi servizi di tlc ma richiede una tecnologia molto flessibile

Commutazione di pacchetto

6.2

IP e ATM

- Inizialmente gli standard (ITU-T) e la ricerca hanno proposto ATM (Asynchronous Transfer Mode)
  - Commutazione pacchetto veloce
  - Orientato alla connessione
  - Pensato principalmente per servizi *real-time* (video-voce)
  - ATM non si trova molto "a suo agio" con il traffico dati, al contrario di IP che è invece stato pensato solo per i dati.
- Con Internet, il TCP/IP si è imposto sia come tecnologia che come "infrastruttura" di rete, quindi è sembrato "naturale" proporre **Internet** anche come **rete integrata universale**

6.3

QoS

- Per poter fornire supporto a servizi differenti, bisogna poter assicurare una **Qualità** ossia rispettare dei requisiti diversi in termini
  - Perdita massima ⇩
  - Ritardo massimo ⇩
  - Jitter ⇩
  - Throughtput ⇧
- In corrispondenza di flussi di traffico differenti per
  - Tasso di generazione
  - Statistiche di generazione (CBR, VBR,…)

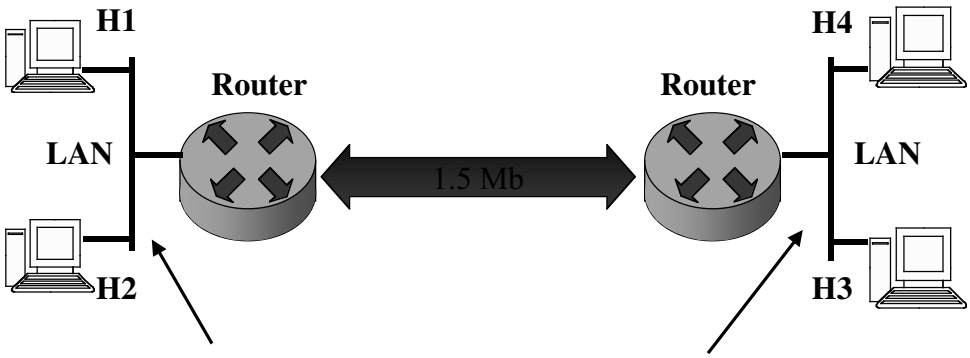
6.4

QoS su reti a pacchetto

- Internet o, in generale, l'architettura TCP/IP è nata ed è attualmente strutturata per fornire solo servizi ***best-effort***, ossia con
  - Perdita di pacchetto
  - Ritardi non superiormente limitati
  - Ritardi variabili (Jitter)
- Per poter utilizzare Internet come rete universale, bisogna poter assicurare un grado di QoS ad applicazioni o utenti che ne facciano richiesta, e quindi bisogna introdotti dei **nuovi componenti architetturali**.

6.5

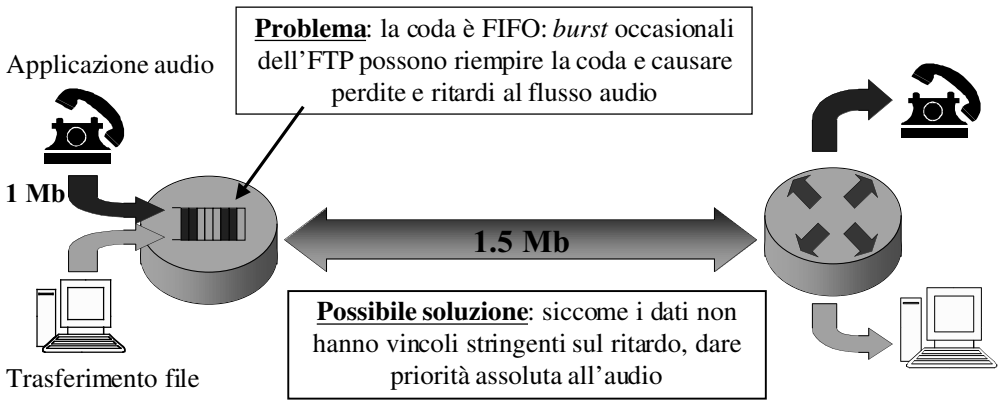
Scenario di riferimento



Si supponga che la LAN non sia un elemento critico (nel senso che ha una velocità molto più alta di quella presente fra i due *router*)

6.6

Scenario 1



Per poter servire differentemente i due traffici i *router* devono poter distinguere i pacchetti di classi diverse, ossia è necessario un "***packet marking***" per classe di traffico

6.7

Scenario 1

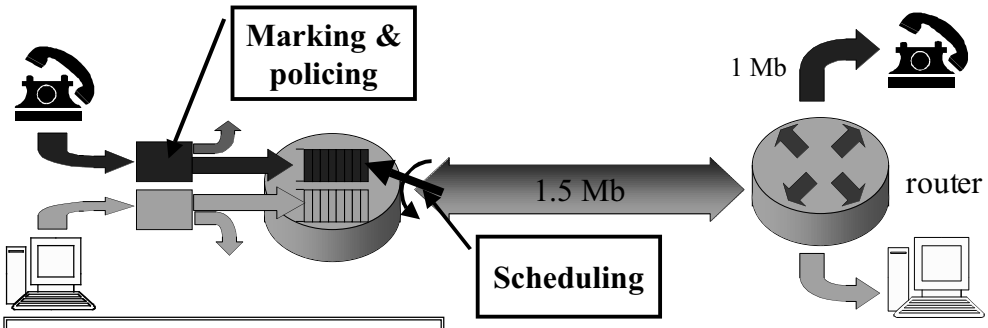
- Se l'utente del trasferimento file ha pagato molto per avere un servizio di "prima classe", mentre l'utente audio ha pagato poco per avere un servizio a basso prezzo, non è detto che la priorità dell'audio debba essere assoluta.
- Ci sono tre considerazioni da fare:
  - Il *Marking* esplicito serve solo a distinguere i pacchetti ma non determina necessariamente il tipo di servizio che riceverà un pacchetto.
  - Il modo con cui il *router* classifica (*classification*) i pacchetti può basarsi su diversi criteri (sorgente, destinazione, stato della linea) di cui l'identificazione esplicita è solo uno degli elementi
  - Il modo in cui i diversi pacchetti vengono classificati e poi trattati dipende dalla "politica" di servizio.

Non basta fare il marking per assicurare una QoS
Occorrono anche altri meccanismi

6.8

Scenario 2

L'applicazione audio (volontariamente o per errore) comincia a generare più di 1 Mbps; se arriva a generare ≥ 1.5 Mbps, l'FTP non passa più



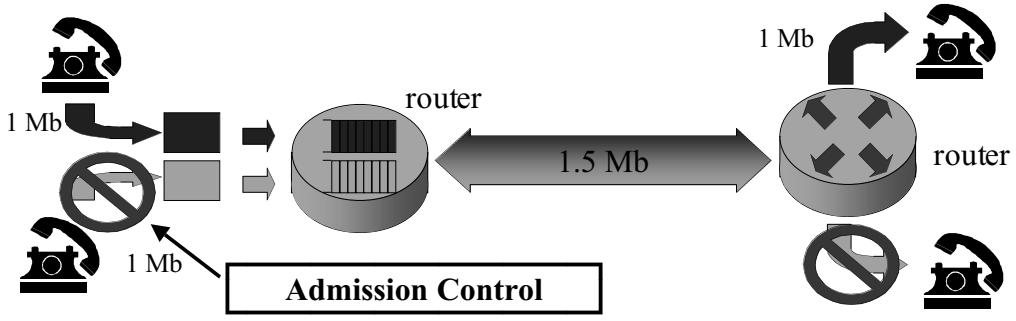
Bisogna poter fornire un **certo grado di indipendenza fra i flussi**, per evitare che flussi "sbagliati" influenzino negativamente le prestazioni degli altri

Si osservi però che l'isolamento fra flussi deve essere ottenuto mantenendo il **massimo grado di efficienza** nell'uso delle risorse (banda e buffer)

6.9

Scenario 3

Se si attivano due applicazioni audio da 1 Mbps, qualunque *classification* o *policing* o partizione (*scheduling*) non permette di mantenere la QoS



Bisogna che le applicazioni dichiarino i propri flussi e la richiesta di QoS e che subiscano un "**Controllo di Accesso**"

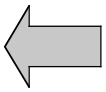
6.10

Elementi

- In sostanza gli elementi che devono essere considerati per fornire una QoS su IP sono:
  - **Classificazione dei pacchetti (*Packet Classification*)**
  - **Isolamento dei flussi: *Scheduling* e *Policing***
  - **Efficace utilizzazione delle risorse (Allocazione dinamica delle risorse)**
  - **Controllo di ammissione (*Call Admission*)**

6.11

IP-QoS

- L'IETF ha proposto due approcci diversi (ma non necessariamente mutualmente esclusivi):
  - **Integrated Services** 
  - **Differentiated Services**

6.12

Integrated Services (IntServ)

- In sostanza, porta la filosofia ATM in IP:
  - Tramite cinque parametri (indirizzo destinazione, ind. sorgente, protocollo, porta sorgente, porta destinazione) i pacchetti vengono identificati "su base connessione" *(marking).*
  - Quando una connessione chiede di essere attivata, un protocollo (RSVP) segnala ai router lungo il percorso le caratteristiche del flusso e la QoS richiesta e riceve da questi l'eventuale permesso di attivazione *(Admission control)*

6.13

Integrated Services (IntServ)

- I router devono assicurare banda e QoS a ciascun flusso (tramite uno *scheduling* fra i flussi)
- All'ingresso della rete sul flusso di ogni connessione è applicato un "*policing*" ed eventualmente uno "*shaping*" per mantenere il flusso in ingresso sempre conforme al dichiarato.

6.14

Integrated services

- Il protocollo principale legato a questo approccio è il ReSerVation Protocol (RSVP).
- Si tratta, in sostanza, di un protocollo di segnalazione che permette ai router (ed eventualmente agli host) di interagire per attivare e configurare flussi con QoS assicurata.
- Non definisce meccanismi da applicare per assicurare banda e ritardo, e neppure come calcolare quanta banda riservare ai diversi flussi e come decidere se accettare o meno un nuovo flusso.

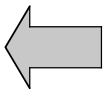
6.15

Integrated services

- Il distinguere ciascun flusso ha il vantaggio di permettere una precisa allocazione delle risorse ma ha il grosso limite di essere **poco scalabile**
- I router delle dorsali, che commutano molti flussi, potrebbero non riuscire a gestire una QoS per flusso in modo efficace.
- Per cui, questo approccio appare realistico all'interno di "aree" di rete e/o applicato insieme a delle tecnologie di supporto per la QoS (ATM o MLSP)

6.16

IP-QoS

- L'IETF ha proposto due approcci diversi (ma non necessariamente mutualmente esclusivi):
  - **Integrated Services**
  - **Differentiated Services** 

6.17

Differentiated Services

- Il vincolo principale a cui ha cercato di sottostare questo approccio è permettere una agevole scalabilità
- Con questo proposito si è introdotto (rispetto agli *IntServ*)
  - L'aggregazioni dei flussi in classi
  - La differenziazione delle funzionalità dei *router*
  - La riduzione del traffico di segnalazione
  - La riduzione delle variabili di stato nei *router*

6.18

Differentiated Services

- Il campo ToS (*Type of Service*) dell'IP viene rinominato **DS *field*** ed usato per identificare delle "classi di servizio".
- Ad ogni classe viene legato un
  Per Hop Behaviour (PHB)
  che stabilisce il tipo di trattamento che ad ogni *router* deve subire una certa classe

6.19

PHB

- ***Expedited Forwarding*** (EF)
  - E' pensato per costruire servizi *end-to-end* a
    - » bassa perdita
    - » bassa latenza
    - » basso *Jitter*
    - » banda assicurata
      (VLAN, Videoconferenza, voice over IP, …)
  - Ai due estremi appare come una "linea dedicata virtuale"
  - Richiede in sostanza un servizio prioritario ed una allocazione indipendente dalle altri classi

6.20

PHB

- ***Assured Forwarding*** (AF)
  - Ha l'obiettivo di permettere un servizio accettabile anche in condizioni di congestione della rete
  - Al suo interno si possono distinguere più sotto classi (AF1-AF4)
  - I pacchetti possono essere "marcati" con tre diversi colori (verde, giallo e rosso) che implicano diversi livelli di precedenza in caso di scarto (*drop*)

6.21

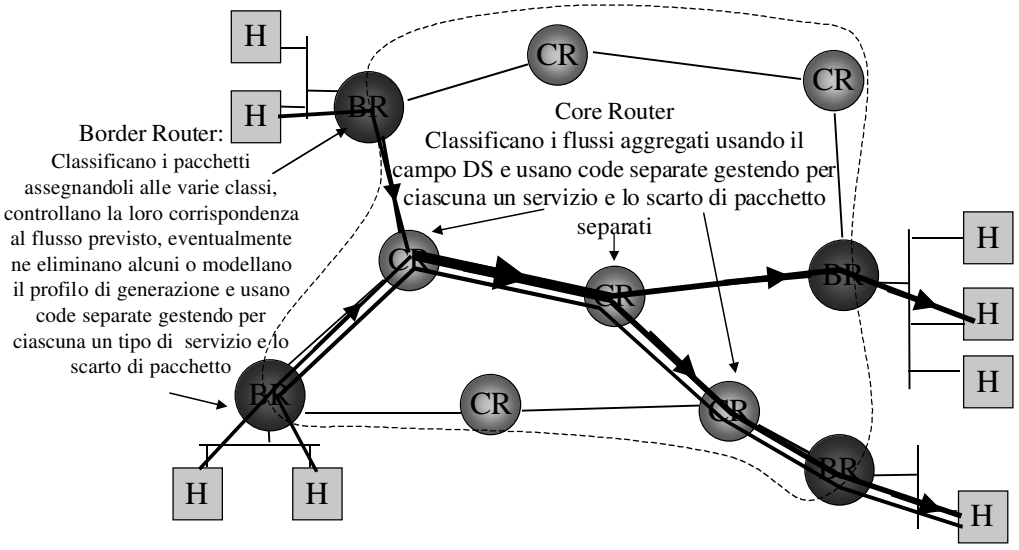
PHB

- ***Default Forwarding*** (DF)
  - Sono i pacchetti non esplicitamente marcati, ossia l'attuale traffico *best effort*
  - Questa classe, i n linea di principio, potrebbe venir trattata come un AF con la priorità più bassa in assoluto o considerata a parte.

6.22

Differentiated Services

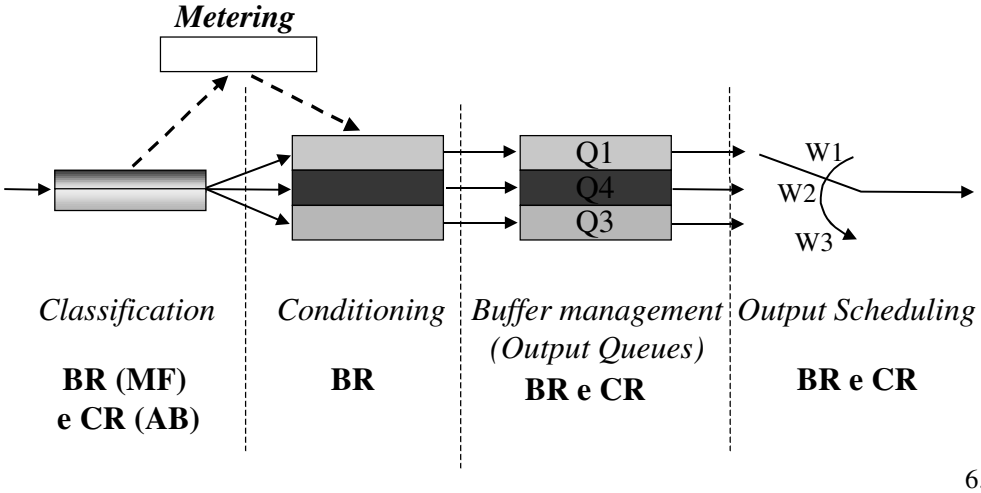
Schema di principio



6.23

Router Components

- Componenti dei *router Diffserv*



6.24

Classification

- La classificazione serve a decidere come trattare i pacchetti.
- Tipi di classificazioni
  - ***Behaviour Aggregate*** (BA): il *router* usa solo campo DS per decidere come trattare il pacchetto.
  - ***Multi- Field*** (MF): il *router* usa cinque campi dell'intestazione IP/TCP (indirizzo sorgente/destinazione, porta sorgente/destinazione e IP *protocol number)*
  - Ci possono essere anche altri tipi di classificazione (basate, per esempio, sugli indirizzi MAC).

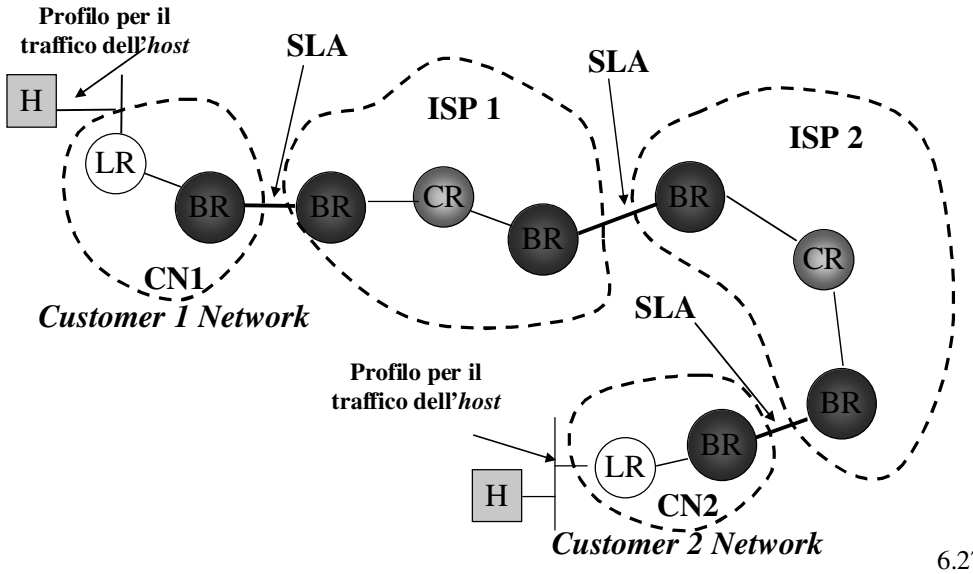
6.25

Service Level Agreement (SLA)

- In corrispondenza dell'attivazione di servizi differenziati, deve venir stabilito un "profilo di traffico" a cui il flusso generato deve sottostare.
- Questi profili sono definiti tramite ***Service Level Agreement*** (SLA), che definiscono le classi supportate e il livello di traffico con le sue caratteristiche.
- Ogni area di confine, fra *host* e "*Leaf Router*" (LR), fra rete utente e ISP, fra domini diversi e diversi ISP, devono essere definiti dei Profili di Traffico o degli SLA che stabiliscano, per ogni classe, le caratteristiche del flusso.

6.26

SLA



6.27

SLA

- Gli SLA possono essere
  - statici (definiti per lunghe durate, mesi/anni).
  - dinamici, cioè attivati su richiesta, all'occorrenza.
- Per poter gestire gli SLA dinamici occorre un protocollo di segnalazione (RSVP?).
- Gli SLA dinamici devono ovviamente essere soggetti ad un controllo di accesso.

6.28

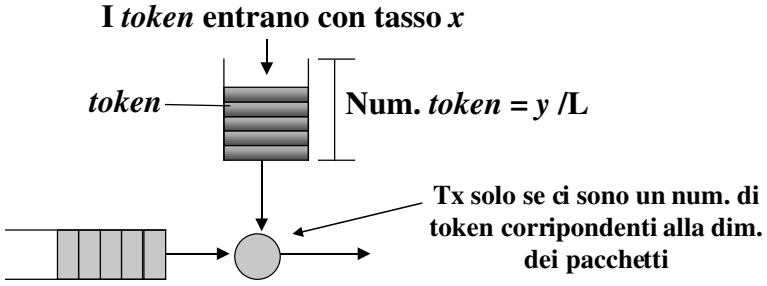
Metering e Conditioning

- Il *Metering* serve a verificare che i flussi in ingresso sia conformi agli SLA o ai profili definiti.
  - Agisce effettuando delle misure sui tassi medi e di picco
- Il *Conditioning* serve a intervenire sul traffico quando questo non risulta conforme al previsto
- Può operare le seguenti azioni
  - Riclassificare i pacchetti non conformi cambiando il valore del DS all'interno della classe o cambiandoli classe (ad es. portandoli alla classe *best-effort*)
  - Applicare degli *shaper*, ad esempio un *Leaky Buket,* per rimodellare il flusso
  - Eventualmente scartare dei pacchetti

6.29

Shaping

- *Linear bound arrival process*:
  (num. bit trasmessi in t) ≤ xt+y
  x = tasso a lungo termine
  y = lungh. massima *burst* (deviazione massima)
- Implementazione: ***Leaky Bucket***



6.30

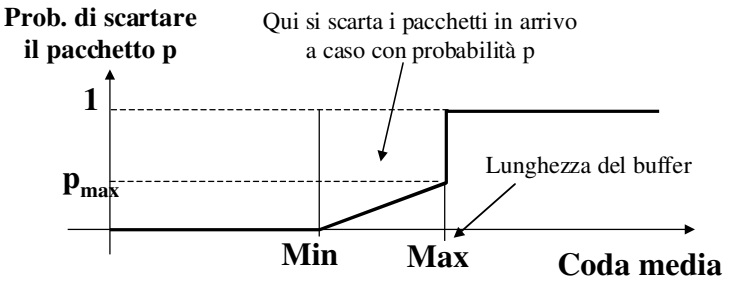
Buffer managenment

- In un caso semplice, la perdita di pacchetti, avviene in modo "naturale" in corrispondenza di un sovraccarico temporaneo, per riempimento della coda: i pacchetti che arrivano quando la coda è piena vengono scartati (*dopped*).
- Questo modo di procedere non è, generalmente, il più efficace perché da luogo a "*burst*" di pacchetti consecutivi persi che hanno due effetti:
  - Peggiorano la qualità di flussi non controllati
  - Tendono a sincronizzare i controlli di flusso del TCP dando luogo a comportamenti oscillatori poco efficienti

6.31

Random Early Detection (RED)

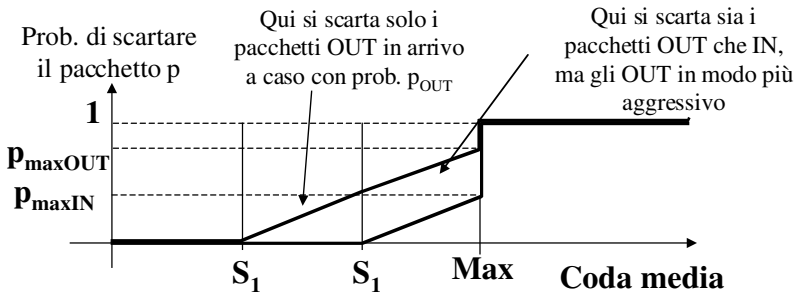
- Un modo per porre rimedio a questo problema è cercare di rendere le perdite "casuali".
- Un metodo per farlo prende il nome RED (*Random Early Detection*); al di sopra di una certa soglia sulla coda media (media esponenziale) introduce una probabilità di perdita che varia linearmente con il valore della coda media



6.32

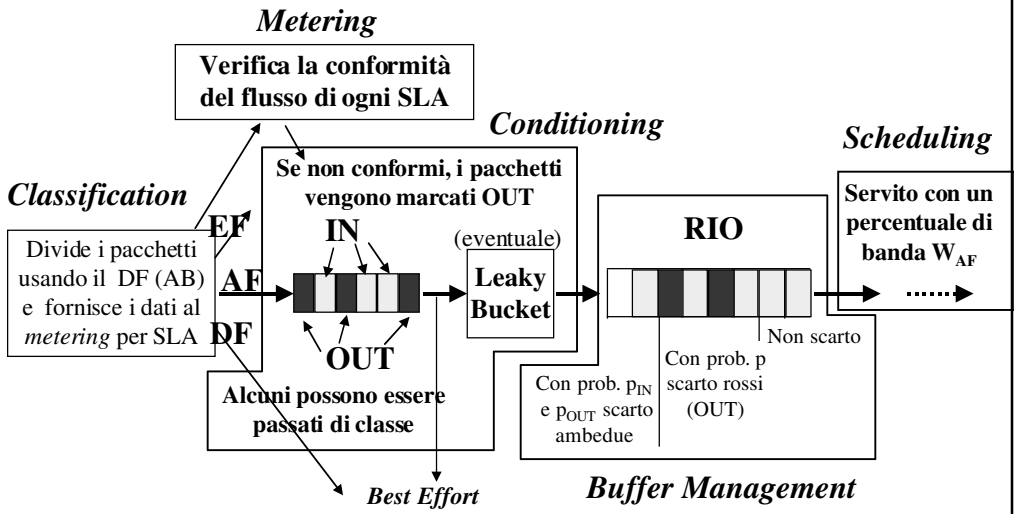
RIO

- Una seconda tecnica più sofisticata si chiama *Random early detection with In and Out* (RIO).
  - In questo caso si suppone la presenza di due classi di pacchetti: IN e OUT (gialli e rossi).
  - Si applicano due RED separati per ciascuna classe:



6.33

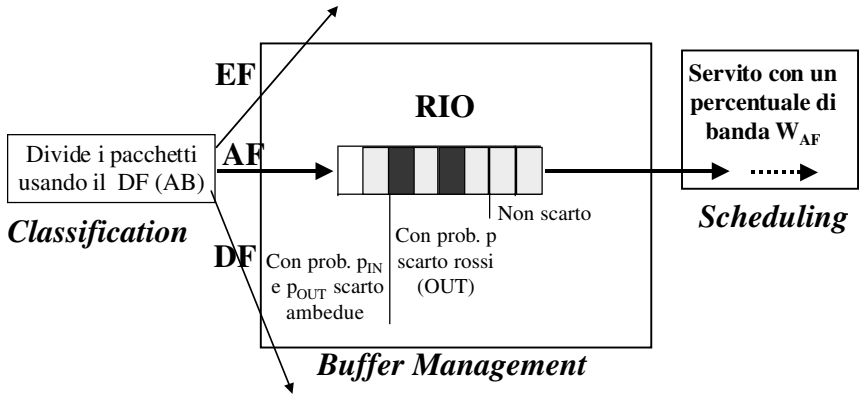
Border router (AF)



Complesso ma non deve servire troppo traffico aggregato ne fare instradamenti molto complessi

6.34

Core Router (AF)



Deve gestire flussi aggregati ma le sue funzionalità sono molto più semplici

Il *Leaf Router* è simile al BR, ma deve fare una classificazione MF ed eventualmente può essere meno restrittivo nel *conditioning*.

6.35

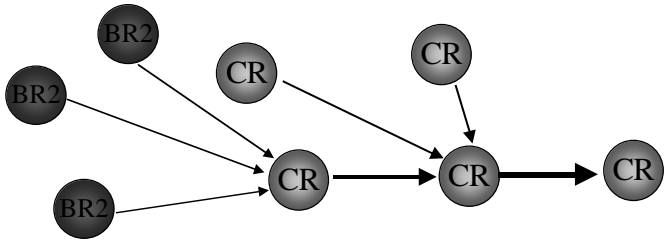
Expedited Forwarding (EF)

- Per i EF il discorso è simile al AF con alcune differenze:
  - Il traffico è sottoposto a *shaping* sia ai LR che ai BR
  - Il traffico in eccesso (sul tasso di picco) non è cambiato di classe ma scartato
  - Lo *scheduler* serve con priorità elevata il traffico EF, indipendentemente dal carico generato dagli altri traffici.
  - Il traffico EF è soggetto a controllo di accesso (gli SLA possono essere sia statici che dinamici).
  - La quantità di traffico EF ammesso nelle rete deve essere una percentuale ridotta (10%) della capacità totale del traffico in rete.

6.36

Expedited Forwarding (EF)

- Con questi presupposti, dato che in genere tutti i *link* in un *router* sono *full-duplex* e quindi la capacità in ingresso è uguale a quella d'uscita, se il flusso EF è minore del 10% ed è prioritario subisce sempre pochissima perdita e ritardo.
- In realtà non necessariamente detto che questo sia vero (perché sia così bisogna usare degli algoritmi di *routing* per la QoS o fare un management specifico)



6.37

Allocazione di Banda - BB

- Sebbene gli elementi architettura siano stati definiti, in realtà non è stato definito l'elemento che determina l'allocazione delle risorse ed eventualmente effettua il Controllo d'accesso.
- Tale elemento viene in genere indicato col nome di

  Bandwidth Broker

- Si tratta di una entità logica che risiede in ogni dominio (di utente o di ISP).
- Nella rete dell'utente, il BB interagisce con l'host all'attivazione del sevizio e configura il LR e tutti i *router* intermedi per permettere la realizzazione del servizio

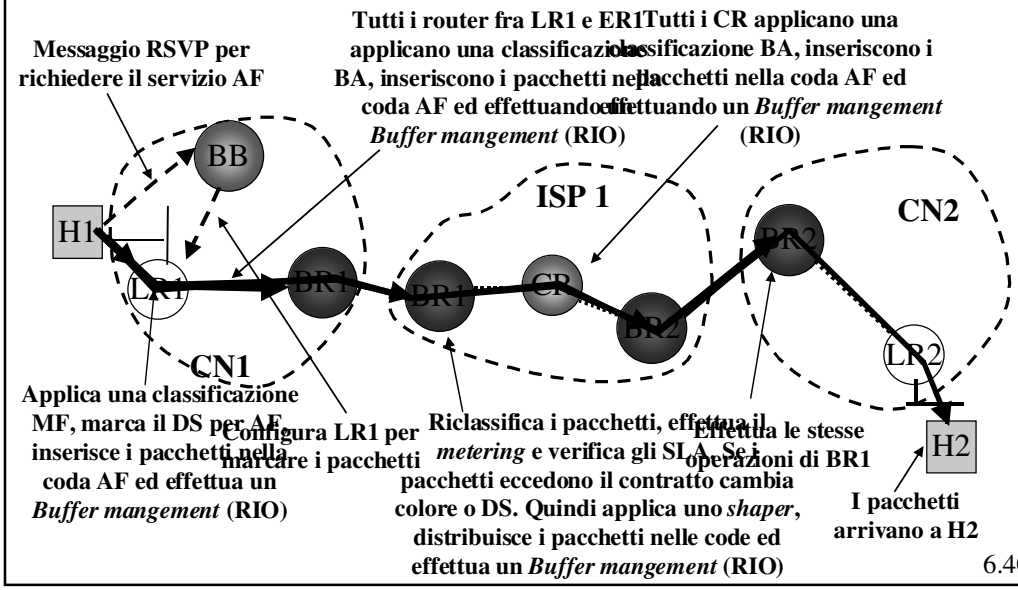
6.38

Allocazione di Banda - BB

- All'interno degli ISP, se gli SLA sono statici, non è strettamente necessario un BB, in quanto le risorse possono essere allocate via *management*
- Se ci sono SLA dinamici, i BB sono necessari per configurare i *router* (BR e CR) e devono essere in grado di colloquiare fra loro anche fra domini diversi.
- Sebbene definiti architetturalmente, non sono definite (e neppure banali) le politiche attraverso le quali le risorse devono essere allocate.

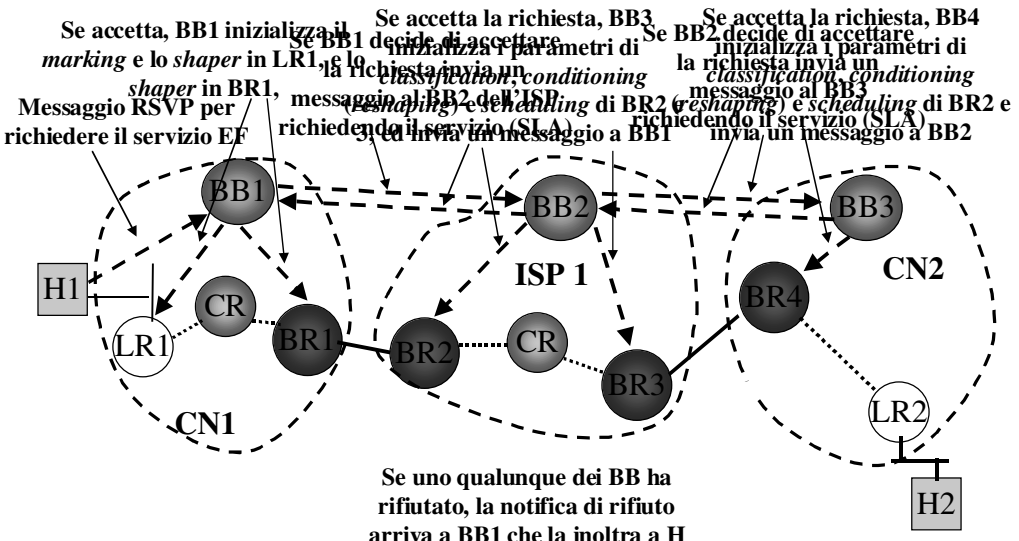
6.39

AF - Un esempio



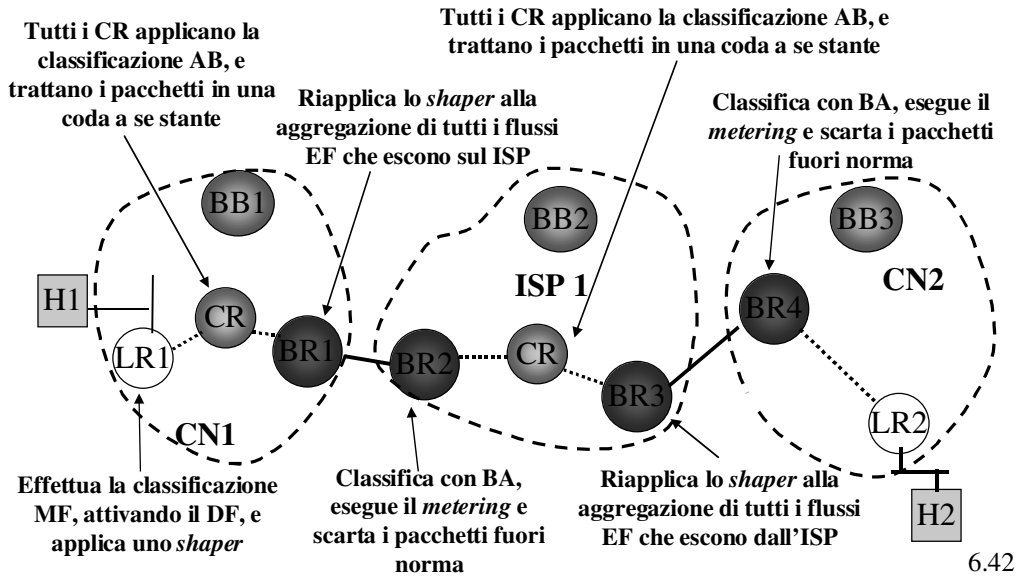
6.40

EF - Un esempio (segnalazione)



6.41

EF - Un esempio (fase dati)



6.42

Altri elementi importanti

- *MultiProtocol Label Switching* (MPLS)
- QoS *Routing* (*Constraint based routing*)
- *Traffic Engineering*
- ATM

6.43