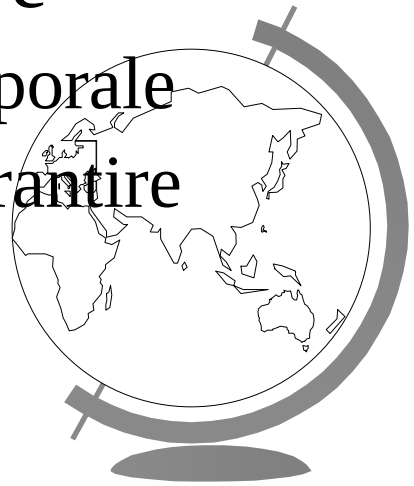
LA QUALITÀ DI SERVIZIO NELLE RETI

Prof. Ing. Maurizio Casoni

Dipartimento di Ingegneria “Enzo Ferrari”
Università degli Studi di Modena e Reggio Emilia

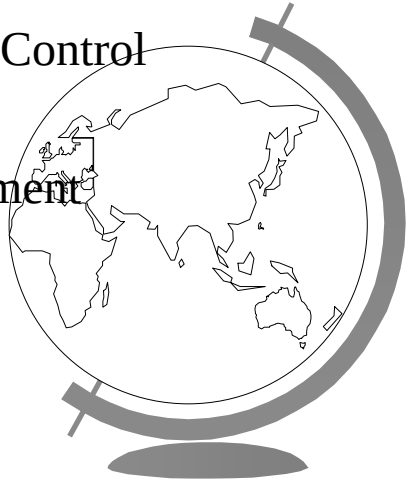
Reti integrate

- Reti che consentono il trasporto di diversi tipi di servizio (dati, voce video)
- Applicazioni *non-real-time* o loss sensitive
  - dati: richiedono trasparenza semantica
  - ritrasmissione end-to-end
- Applicazioni *real-time* o delay sensitive
  - voce e video: richiedono trasparenza temporale
  - la rete stessa deve operare in modo da garantire bassi tempi di latenza



Modi per controllare la QoS

- Quality of Service (QoS) è un termine che si può ricondurre ad un certo valore di throughput garantito
- Per massimizzare l'utilizzazione delle risorse di rete, banda e memorie, soddisfacendo al contempo le richieste di QoS per ogni singolo utente, diversi meccanismi di controllo della QoS devono essere impiegati per gestire i diversi flussi di utente nel modo opportuno
- Soddisfare una certa classe di servizio significa operare in modo coordinato:
  1. un controllo degli accessi o Admission Control
  2. un controllo del traffico in ingresso alla rete o Traffic Access Control
  3. un Packet Scheduling
  4. una gestione delle memorie di trasmissione o Buffer Management
  5. un controllo della congestione o di flusso
  6. un QoS Routing



Controllo dell’ Accesso

- Limita il carico offerto alla rete, e ai relativi sistemi a coda, determinando se una nuova richiesta può essere soddisfatta in modo soddisfacente senza penalizzare le garanzie di qualità degli utenti già in servizio
- Call Admission Control (CAC) viene eseguito per decidere se accettare o rifutare la nuova richiesta di connessione
- In ATM, utente fornisce il traffic descriptor, i requisiti di QoS e i modi per mantenere un profilo “corretto”: la rete valuta quindi se ha sufficienti risorse in termini di banda e memorie per soddisfare la QoS richiesta

Traffic Access Control

- Operazione che serve per sagomare il flusso dati in ingresso alla rete e in particolare punti della rete
- Serve per smussare eventuali picchi di traffico che potrebbero creare problemi alla rete e alla connessione stessa di utente perchè potrebbero venire meno le richieste in termini di QoS
- Esiste poi la possibilità di imporre al flusso di utente di rispettare il profilo di traffico contrattato al momento dell' ammissione, mediante funzioni di **policing** che possono scartare o marcare il traffico in eccesso emesso dalla sorgente

Il modello Best Effort

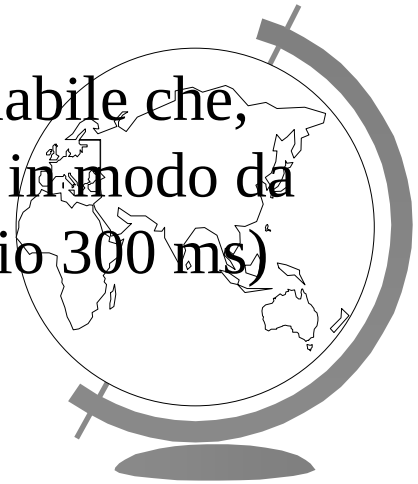
- E’ la modalità su cui si basa il funzionamento di Internet
- La rete ha l’obiettivo di consegnare i pacchetti e lascia ai capi della rete i compiti di recupero di situazioni di errore
- Non è sufficiente per le applicazioni real-time
- Occorre un nuovo modello di servizio di rete che consenta di garantire il livello di servizio richiesto dalle applicazioni

Esempio: applicazione audio

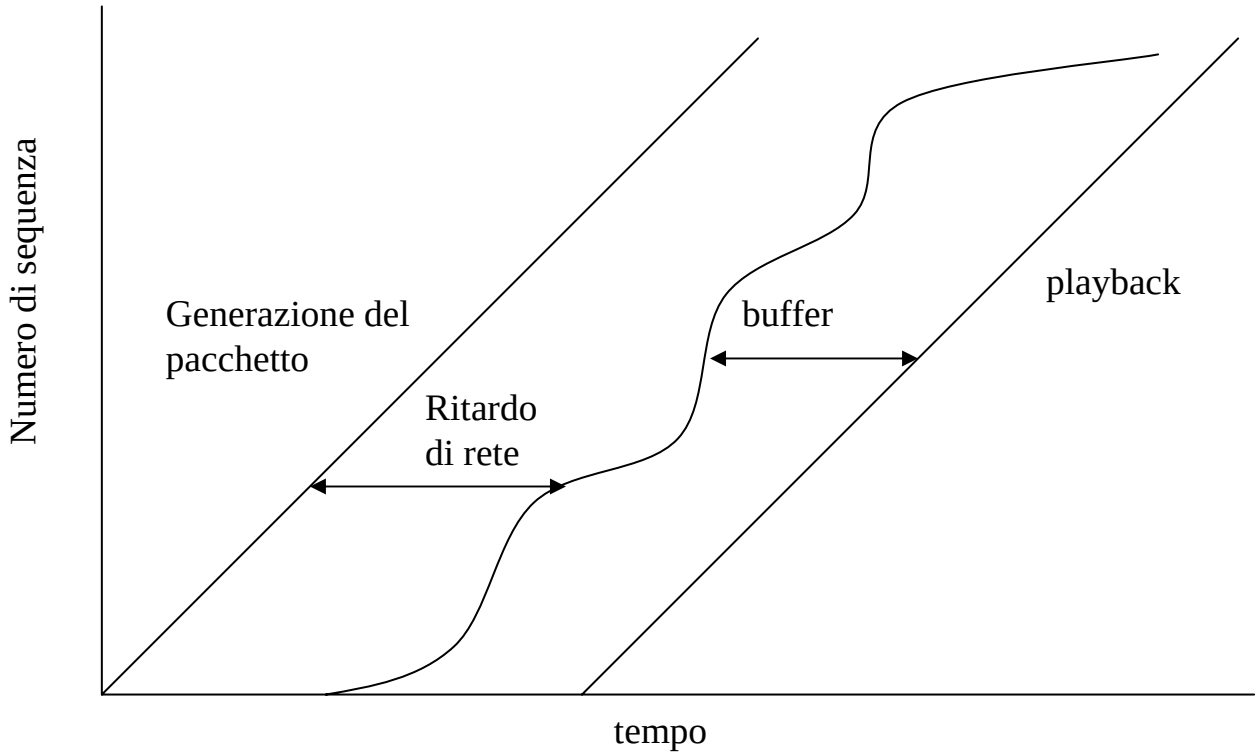
- Applicazione audio genera informazione per conversione A/D e i dati vengono immessi in pacchetti e trasmessi attraverso la rete
- Al ricevitore i dati devono essere consegnati con la tempistica appropriata per la conversione D/A
- Playback time: istante in cui il pacchetto è richiesto al ricevitore
  - ogni campione vocale ha un tempo di playback che è 125 microsecondi successivo a quello del campione che lo precede

Playback buffer

- Per far funzionare l’ applicazione vocale dovremmo garantire che tutti i campioni attraversino la rete con lo stesso ritardo
  - questo è molto difficile da garantire in una rete a pacchetto
- Al ricevitore si memorizzano alcuni campioni in un buffer in modo da garantire la consegna con la corretta tempistica
  - ad ogni campione viene aggiunto un ritardo variabile che, aggiunto al ritardo di rete dà un ritardo costante in modo da rispettare il playback time (per applicazioni audio 300 ms)



Playback buffer



Oltre il best effort

- Occorre introdurre dei meccanismi nuovi su Internet che realizzino le seguenti funzioni
  - Processo di ammissione dei flussi
  - Classificazione dei pacchetti come appartenenti a diverse classi di servizio
  - Monitoraggio dei flussi (policing)
  - Gestione differenziata dell'accodamento (scheduling)
  - Utilizzo efficiente delle risorse

Classificazione

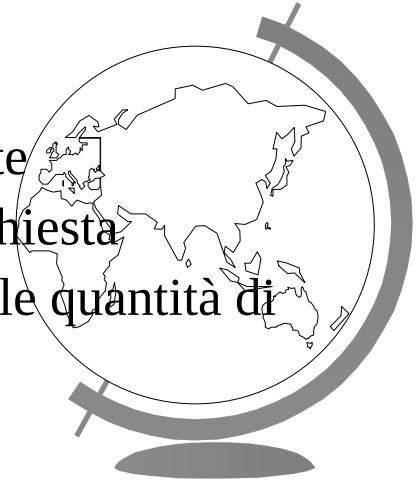
- Viene fatta analizzando fino a cinque campi nel pacchetto:
  - indirizzo sorgente
  - indirizzo destinazione
  - identificatore del protocollo
  - porta sorgente
  - porta destinazione
- In IPv6 si può usare il campo FlowLabel
- Si ottiene una corrispondenza tra le informazioni specifiche del flusso e le classi di servizio trattate nel router

Packet Scheduling

- E’ la modalità con cui i pacchetti nelle code vengono selezionati per la trasmissione
- Esistono diverse modalità che hanno influenza sul livello di servizio
- La più semplice politica di scheduling è la FIFO:
  - i pacchetti che trovano la linea occupata vengono messi in coda e serviti rispettando l’ordine di arrivo
  - Occorre definire la politica di scarto quando la coda è piena
  - Il pacchetto viene rimosso dalla coda quando è stato completamente trasmesso

Bandwidth Allocation

- In assenza di risorse sufficienti a soddisfare tutte le richieste di traffico, la banda deve essere assegnata in modo equo a tutti i flussi che la richiedono
- La politica di equa-condivisione (***fair-sharing***) più largamente diffusa è quella nota come ***max-min fair sharing***
- Max-min fair sharing cerca di massimizzare la minima porzione di banda di un flusso dati la cui richiesta non viene completamente soddisfatta
- I principi di funzionamento sono i seguenti
  - La banda viene assegnata in ordine di richieste crescente
  - A nessun flusso viene assegnata più banda di quella richiesta
  - I flussi con richieste non soddisfatte ottengono un uguale quantità di risorse



Max-min fair sharing

Può essere calcolato nel modo seguente:

1. Calcolare il valore iniziale di *equa condivisione = capacità totale / numero totale dei flussi*
2. Per tutti i flussi con richieste uguali o inferiori all'equa condivisione, si assegna la banda richiesta
3. Escludere i flussi soddisfatti e sottrarre le relative porzioni di banda allocate alla capacità disponibile
4. Ripetere i passi 2 e 3 per i flussi rimanenti con *equa condivisione corrente = capacità residua / numero dei flussi residui*, finchè tutte le richieste residue sono maggiori della equa condivisione corrente. Alla fine tutte le richieste residue otterranno l' *equa condivisione corrente*

☞ Con Max-min fair sharing tutti i flussi con richieste inferiori all' *equa condivisione* verranno soddisfatti, mentre le risorse non usate da essi verranno assegnate in modo equo per i flussi con richieste superiori all' *equa condivisione*

Generalized Processor Sharing

- GPS è un algoritmo di fair queueing ideale che fornisce *l'ideale max-min weighted fair sharing*
- In GPS un flusso si dice in *backlogged* quando ha dati in coda in attesa di essere trasmessi
- Supponiamo vi siano N flussi serviti da servitore di capacità R e che l'i-esimo flusso abbia assegnato un peso w_i; sia inoltre S(i, τ,t) la quantità di dati serviti del flusso i nell'intervallo (τ,t)
- In GPS per ogni flusso in backlog i e j si ha:

$$\frac{S(i,\tau,t)}{S(j,\tau,t)} \geq \frac{w_i}{w_j}$$

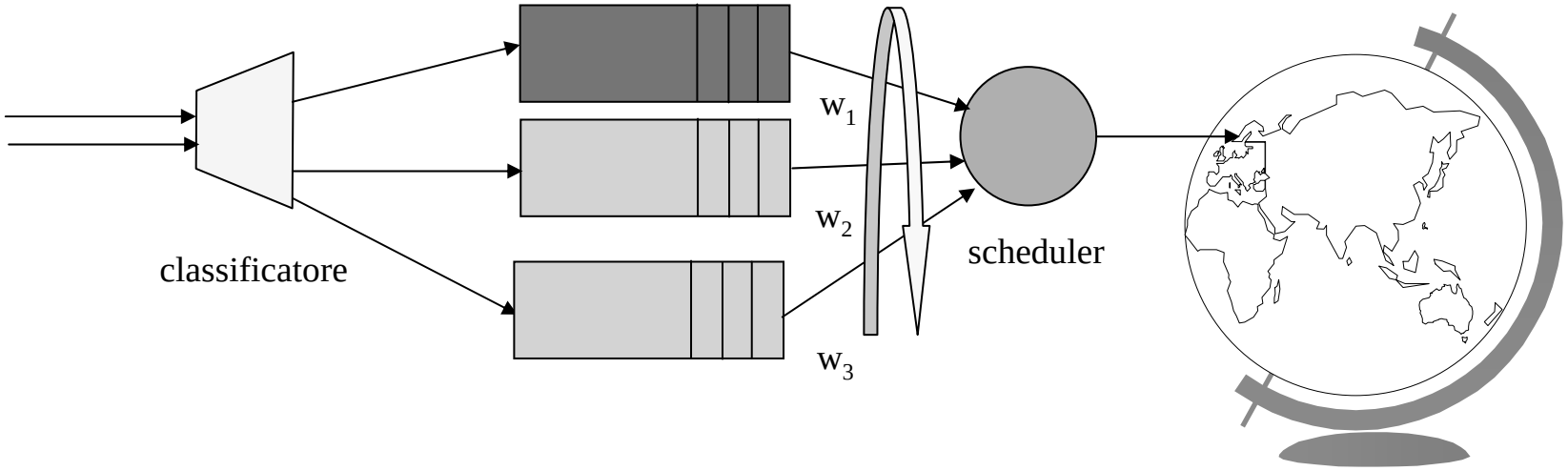
- Nell'intervallo (τ,t) il flusso i riceve una minima condivisione proporzionale al suo peso, $w_i/\Sigma w_i$ R, dove la sommatoria è estesa ai flussi in backlog
- In sostanza, tutti i flussi non backlogged (che hanno richieste inferiori all'equa porzione di banda) sono serviti senza fare coda; la banda residua viene condivisa tra i flussi backlogged in proporzione ai loro pesi
- Quindi, un minimum fair share è garantito a tutti i flussi e la banda in eccesso viene distribuita in modo proporzionale tra i flussi backlogged.
- GPS fornisce la perfetta equità nell'allocazione della banda

Accodamento equo pesato

- La denominazione è la traduzione di *Weighted Fair Queuing (WFQ)*
- È l'approssimazione pacchetizzata del GPS
- I pacchetti vengono smistati in classi
- Si basa sulla modalità di servizio round robin
  - Serve ciclicamente le diverse classi
  - Si dice *work conserving* se non lascia mai inattiva la linea di trasmissione in presenza di pacchetti da trasmettere
- La modalità WFQ è in grado di assegnare diverse quantità di servizio alle classi attraverso l'introduzione dei pesi, numeri reali, w_i

Differenziazione in WFQ

- A ciascuna classe è attribuito un peso, numero reale, w_i
- Alla classe i è garantita una frazione di servizio pari a $w_i/\Sigma w_i$ calcolata relativamente alle classi che hanno pacchetti da trasmettere
- Per una linea di capacità R alla classe i-esima è garantito un throughput pari a $R\ w_i/\Sigma w_i$
- In realtà i pacchetti hanno dimensione discreta ed inoltre possono o meno essere interrotti per iniziare la trasmissione di pacchetti di priorità più elevata (modalità pre-emptive)

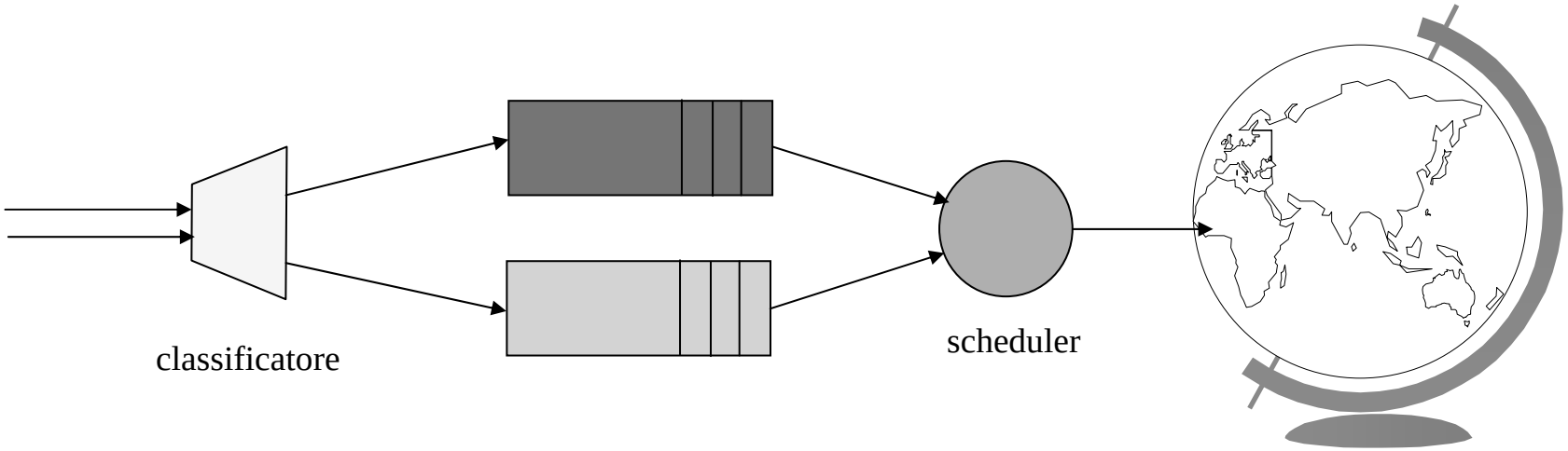


Scheduler Round-Robin

- Modo semplice per implementare il GPS
- Esamina ogni coda in ordine ciclico e serve un pacchetto da ogni coda non vuota incontrata
- Un utente o coda che cresce più delle altre non le influenza: round-robin fornisce così *protezione* da comportamenti eccessivi
- è un primo tentativo di fornire equità tra tutti gli utenti nell' utilizzo della capacità di uscita delle linee
- Si comporta bene quando tutti gli utenti sono uguali ed i pacchetti sono di uguali dimensioni, altrimenti non si ha equità
- Quando gli utenti hanno peso diverso si usa una variante: weighted round-robin (WRR):
  - es. 2 utenti, A e B con A 30% e B 70%, WRR può operare secondo una sequenza ABBABBABBB che si ripete periodicamente

Accodamento prioritario

- I pacchetti in arrivo vengono classificati secondo due o più classi di priorità
- Ciascuna classe ha la sua coda
- Politica di trasmissione: viene trasmesso il pacchetto a priorità più alta la cui coda sia non vuota
- La scelta tra i pacchetti di una stessa classe di priorità avviene con politica FIFO



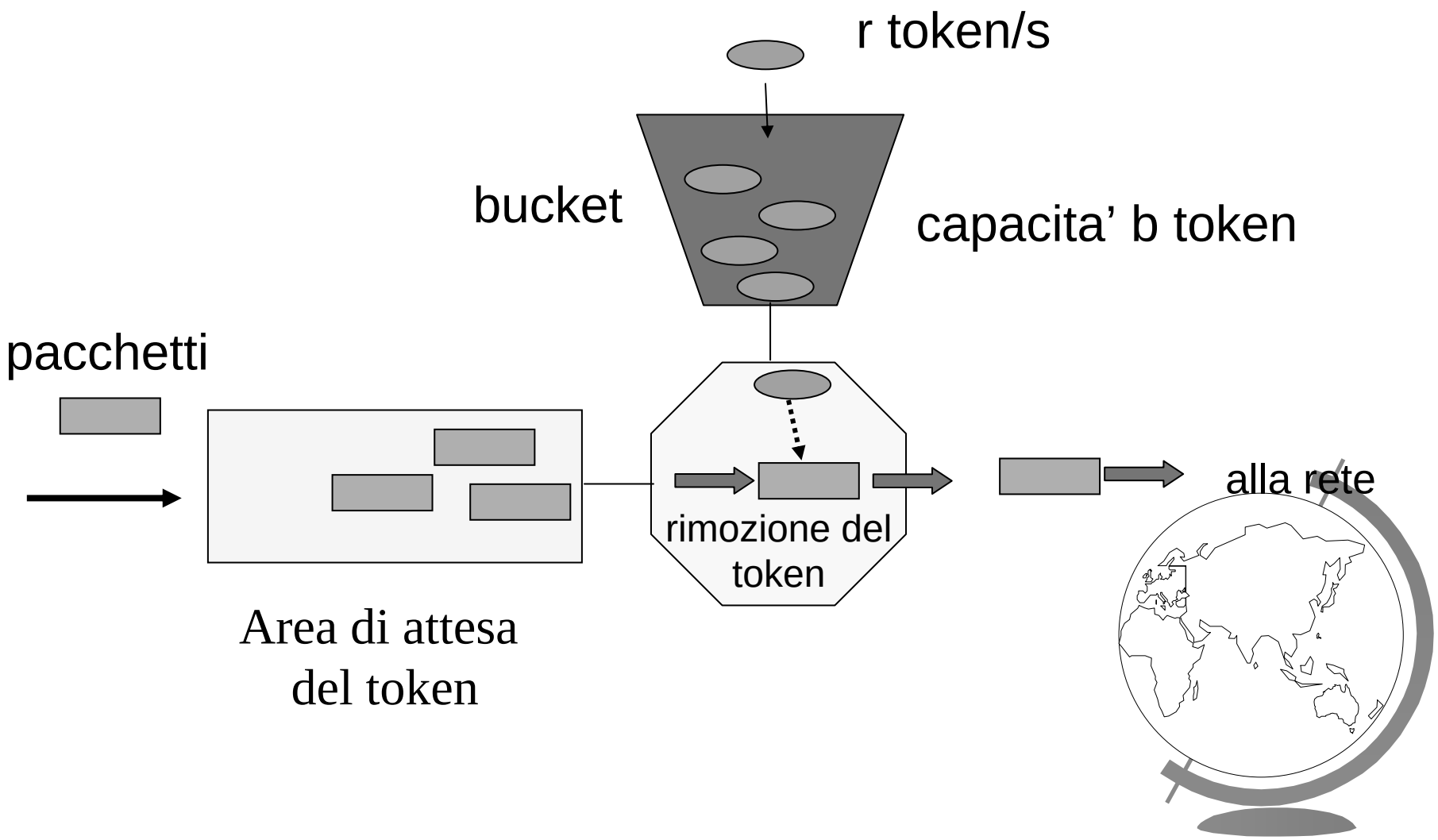
Policing

- E’ una *funzione di sorveglianza*
- Parametri da controllare
  - Velocità media (lungo periodo)
  - Velocità di picco (breve periodo)
  - Dimensione del burst (numero massimo di pacchetti trasmessi alla velocità della linea
- Meccanismi leacky bucket e token bucket
  - Utilizzano entrambi un contenitore di capacità prefissata che viene riempito e svuotato secondo alcuni criteri in relazione alla grandezza che si vuole controllare

Token bucket

- Il contenitore ha una capacità di b gettoni
- I nuovi token che vengono immessi nel contenitore vengono generati a una velocità di r token al secondo
- Quando il contenitore è pieno il token generato viene ignorato
- Un pacchetto, prima di essere immesso nella rete deve prelevare un token dal contenitore
  - Se il contenitore è vuoto il pacchetto aspetta
- Impatto sul traffico
  - Poiché nel contenitore ci possono essere al massimo b token, b è il numero massimo di pacchetti che possono costituire il burst
  - Il numero massimo di pacchetti che possono entrare nella rete in un intervallo generico t è rt + b

Schema del token bucket



Token bucket + WFQ

- Uscita di un router con n flussi ciascuno regolato da un token bucket di parametri b_i e r_i
- Ogni flusso ha una banda $Rw_i/\Sigma w_i$ garantita
- Il massimo ritardo per i pacchetti è dato dal ritardo dell'ultimo pacchetto del burst relativo al flusso i-esimo e cioè b_i $/Rw_i/\Sigma w_i$

Buffer Management

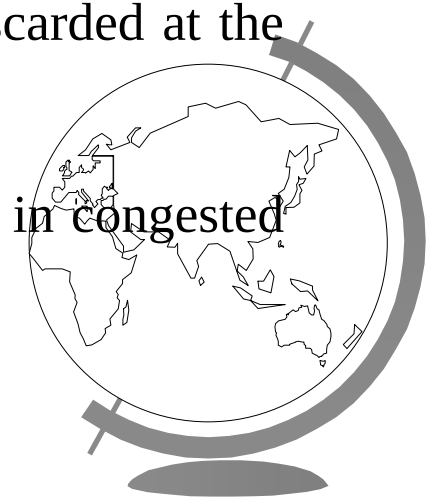
- La gestione delle memorie è fondamentale in quanto queste sono condivise tra un certo numero di flussi dati
- Occorre definire politiche di gestione per decidere quali pacchetti scartare in caso di sovraccarico delle memorie
- Queste politiche sono quindi molto importanti perché possono influire in modo significativo sulle prestazioni

TODAY APPLICATIONS

- ✓ Many CBR and VBR applications have small peak rates relative to the link rate and modest peak-to-average rate ratios while many data applications have big
- ✓ Many applications (audio, video, low speed data) are quite predictable while others (images retrieval) are not
- ✓ Unpredictability of the human user and high peak rates for acceptable interactive QoS imply
  - **Trade-off: low network utilization vs. overload period**

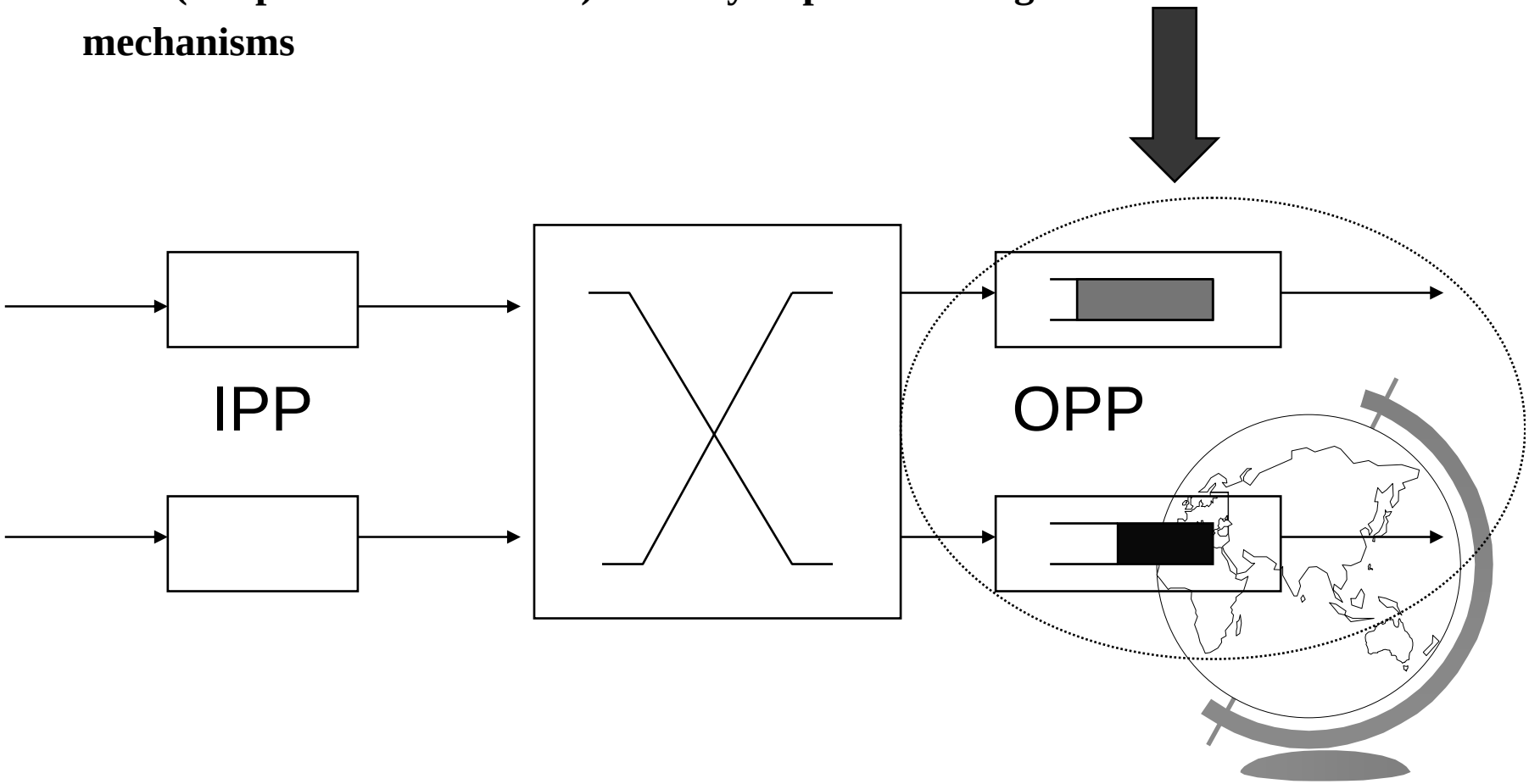
WHY DISCARDING PACKETS ?

- ✓ End-to-end protocols (e.g. TCP) fragment data in packets for fitting L2 frames
- ✓ Single lost frame leads to the retransmission of an entire transport level packet
- ✓ Congestion collapse
  - **Throughput drops to zero as the offered load increases**
- ✓ Important to avoid transmitting frames which will be discarded at the destination
- ✓ Packet discard mechanisms aim at improving throughput in congested switch/routers by selectively dropping incoming frames



GENERIC SWITCHING SYSTEM

- ✓ **IPP (Input Port Processor)**
- ✓ **OPP (Output Port Processor): may implement congestion control mechanisms**

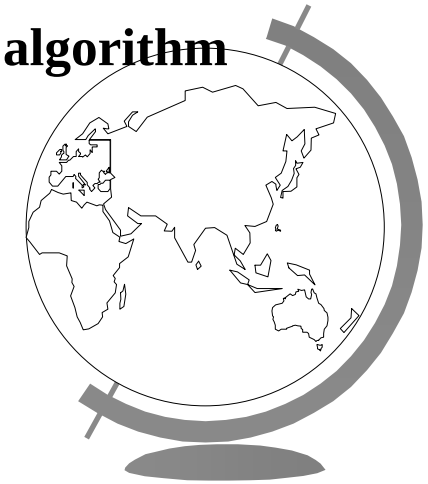


SWITCH/ROUTER

How a switch/router must treat different Class of Service packets ?

Two main approaches:

- **Packet or Threshold dropping**
  - ✓ **one FIFO shared buffer per output link for packets for all classes**
  - ✓ **it provides service differentiation through thresholds**
- **Priority scheduling**
  - ✓ **one separate logical queue for each class and some algorithm drives the scheduler for packet forwarding**



QoS Routing

- Attuali protocolli di instradamento in reti IP sono insensibili a requisiti di QoS che diversi pacchetti o flussi possono richiedere
- Conseguenza: decisioni di routing sono prese senza alcuna considerazione delle disponibilità delle risorse di rete e dei requisiti di QoS
- Flussi dati potrebbero quindi essere instradati su percorsi che alla fine possono rivelarsi incapaci di soddisfare la loro QoS mentre invece percorsi alternativi avrebbero potuto essere più adeguati
- QoS routing significa far uso di algoritmi che possono identificare percorsi con sufficienti capacità residue di risorse per soddisfare un certo flusso dati

Modelli a supporto della QoS

- Possono essere classificati in due grandi categorie:
  - *fine-grained*: controllano la qualità di servizio per la singola applicazione o il singolo flusso. Appartiene a questa categoria l'architettura Integrated Services definita da IETF e tipicamente associata a RSVP;
  - *coarse grained*: forniscono qualità di servizio per classi di dati o traffico aggregato. Appartiene a questa categoria l'architettura Differentiated Services di IETF.

Integrated Services

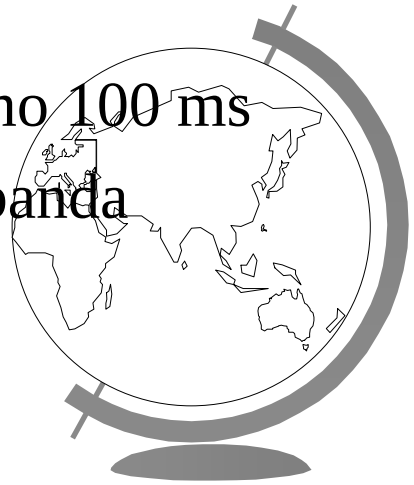
- Integrated services è un gruppo di lavoro di IETF che ha operato in IETF principalmente negli anni 1995-97
- Ha definito alcune classi di servizio progettate per le necessità dei tipi di applicazione classificate in precedenza
- Ha definito come RSVP può essere utilizzato per riservare risorse in rete in relazione alle classi di servizio

Classi di servizio

- ***Guaranteed services****:* è definita per le applicazioni intolleranti per le quali un pacchetto non può mai arrivare in ritardo.
  - *La rete garantisce un valore massimo del ritardo in relazione al quale viene definito il tempo di playback*
- ***Controlled load***: è definita per applicazioni in grado di adattarsi allo stato della rete
  - *Si definisce qualitativamente che una percentuale sufficientemente elevata dei pacchetti riceverà una qualità di servizio molto prossima a quella che riceverebbe da una rete scarica (RFC 2211)*
  - *si attua con meccanismi di accodamento insieme a procedure di controllo di ammissione in modo da limitare il carico sui link*

Meccanismi per realizzare le classi di servizio

- Occorre comunicare alla rete il servizio richiesto
  - l' insieme delle informazioni che vengono fornite alla rete è detto *flowspec:* un *flusso* è l'insieme di pacchetti associati ad una applicazione con i medesimi requisiti
- Esempi di specifica:
  - tipo di servizio es: *controlled load*
  - valore dei parametri di servizio es: ritardo massimo 100 ms
  - caratteristiche del traffico es: informazioni sulla banda



Altre componenti per la gestione delle classi di servizio

- Controllo di ammissione
  - In relazione ad una richiesta di servizio la rete deve stabilire se può offrire quel servizio
- Riservazione o segnalazione
  - sono i meccanismi con cui i vari componenti della rete si scambiano le informazioni
- Scheduling
  - è l' insieme dei meccanismi che consentono di soddisfare i requisiti di un flusso ammesso

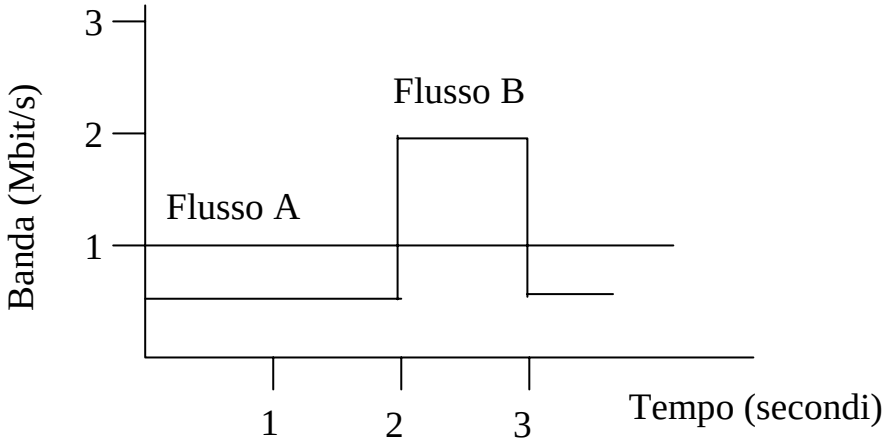
Flowspec

- Consiste di due componenti:
  - ***Tspec***: descrive le caratteristiche del traffico: necessita di meccanismi per la definizione di requisiti di banda
  - ***Rspec:*** descrive il servizio richiesto alla rete: è una specifica abbastanza semplice che si limita a dire se si tratta di un servizio guaranteed o controlled load e, nel primo caso fornisce anche informazioni quantitative sul ritardo massimo

Descrizione della banda

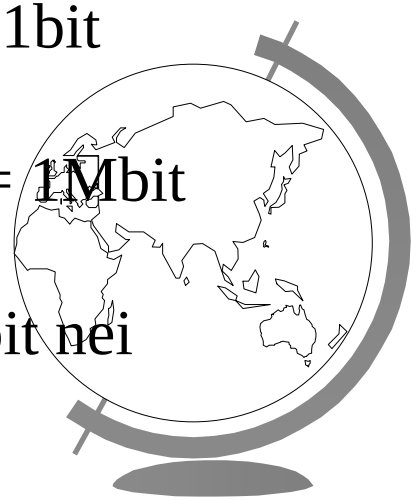
- Si attua attraverso la specifica del filtro *token bucket*
- Il filtro token bucket è descritto dai due parametri
  - r: velocità del token
  - b: profondità del recipiente
- Funzionamento
  - Per poter trasmettere un byte devo avere un token
  - I token si accumulano alla velocità di r token al secondo
  - Non si possono accumulare più di b token
  - Si possono inviare b byte alla velocità che si vuole purché su un periodo di tempo sufficientemente lungo non si trasmettano più di r byte al secondo

Esempio



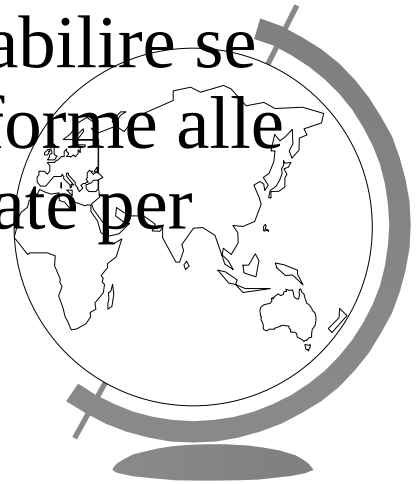
- Il flusso A può essere descritto da r=1 Mbit/s e b=1bit
- Il flusso B può essere descritto da r= 1Mbit/s e b = 1Mbit

  in questo modo accumula 0.5 Mbit/s x 2s= 1Mbit nei due secondi in cui trasmette a 0.5 Mbit/s



Controllo di ammissione

- E’ la procedura che consente di accettare un nuovo flusso
- Si basa sull’analisi di *Tspec* e *Rspec* per decidere se il livello di servizio richiesto per il profilo di traffico specificato può essere garantito in relazione allo stato corrente della rete
- Il *policing* si applica su base pacchetto per stabilire se un flusso ammesso si comporta in modo conforme alle specifiche di traffico *TSpec* che sono state usate per l’ammissione



Il protocollo di riservazione

- Il protocollo connection less usato in Internet è molto robusto nei confronti di router o linee fuori servizio
  - quando una linea o un router va fuori servizio la connettività end-to-end viene mantenuta poiché il routing utilizza percorsi diversi
- Occorre che il protocollo di riservazione per flusso si avvicini il più possibile a queste caratteristiche di robustezza
- Il più diffuso protocollo di riservazione usato in Internet è ***RSVP (ReSource reserVation Protocol)***

Caratteristiche principali di RSVP

- ***Soft state***:
  - le informazioni memorizzate nei router per la gestione dei flussi non necessitano di esplicita cancellazione ma vengono mantenute per un tempo limitato a meno che non vengano “rinfrescate” periodicamente
- ***Receiver-oriented***:
  - i riceventi si curano di aggiornare le loro esigenze in relazione ai diversi trasmittenti da cui desiderano ricevere
  - orientato alla gestione del multicast (molti più riceventi che trasmittenti)

Funzionamento di RSVP

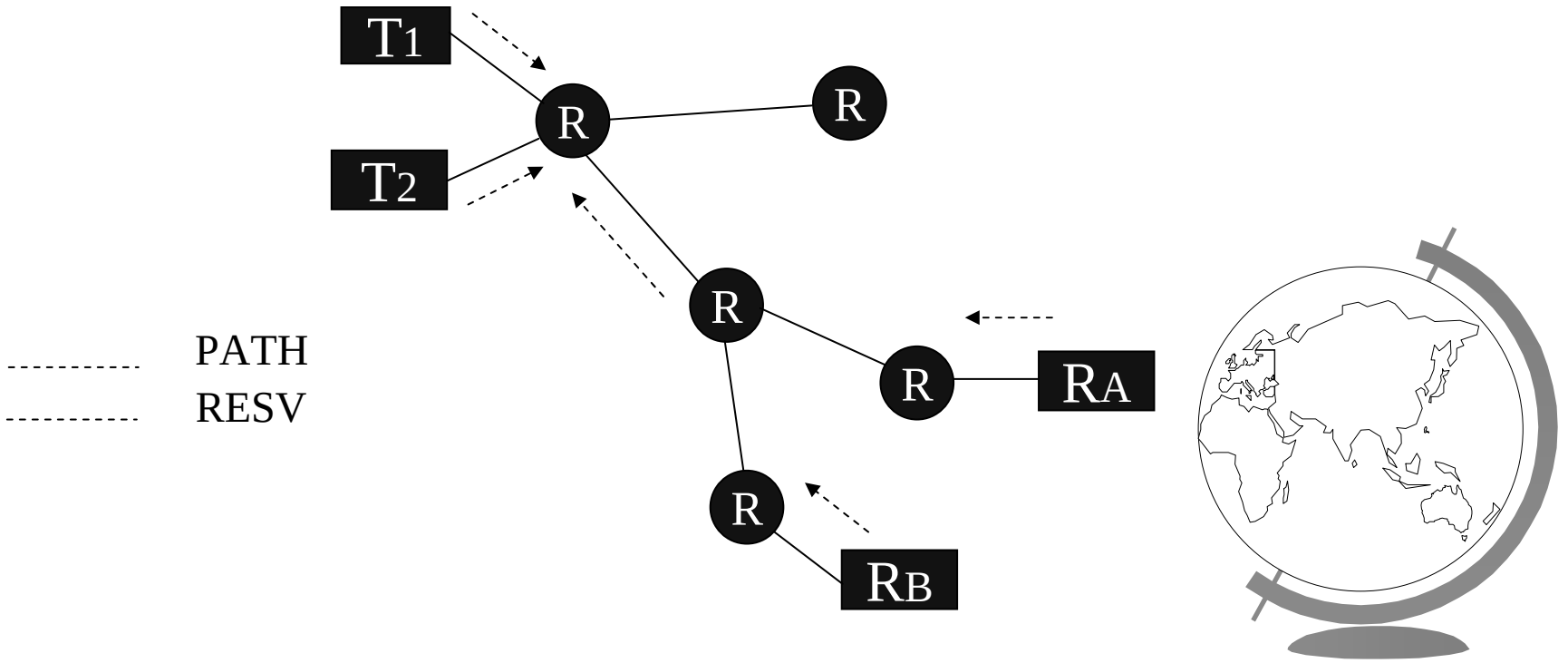
- Il trasmettitore invia il messaggio *PATH* che consente al ricevitore di conoscere
  - il tipo di traffico che viene generato al trasmettitore (Tspec)
  - il percorso dal trasmettitore al ricevitore in modo da effettuare la riservazione di risorse su ciascun router del percorso
- Dopo avere ricevuto il *PATH* il ricevitore effettua la riservazione mediante il messaggio *RESV*
  - contiene TSpec del trasmettitore e RSpec del ricevitore
  - ogni router del percorso vede la richiesta e cerca di allocare le risorse necessarie
  - Se la richiesta può essere soddisfatta il router invia il messaggio *RESV* al router successivo altrimenti invia un messaggio di errore al ricevitore;
  - i messaggi *RESV* vengono ripetuti ogni 30 s

Guasti

- Quando un router o un link è fuori servizio i protocolli di routing provvedono a realizzare un nuovo link tra trasmettitore e ricevitore
- i messaggi PATH sono normalmente inviati ogni 30s ma vengono inviati anche non appena il router rivela una modifica nella tabella di inoltro sul nuovo percorso
- i messaggi RESV seguenti seguono il nuovo percorso ed effettuano la nuova riservazione
- Intanto i router sul vecchio percorso, non ricevendo più messaggi RESV rilasciano le risorse allocate

Multicast

- E’ la situazione in cui esistono trasmittenti multipli e riceventi multipli



Ricevitori multipli e trasmettitore singolo

- Quando un messaggio percorre a ritroso l' albero multicast è probabile che trovi un tratto in cui è già stata fatta riservazione per la stessa applicazione
  - Es: il ricevitore A ha richiesto un servizio guaranteed con ritardo inferiore a 100 ms e la nuova richiesta di B è per un ritardo inferiore a 200 ms: non occorre ulteriore riservazione.
  - Se la richiesta fosse per una garanzia di ritardo inferiore a 50 m si innesca la procedura di riservazione . Se va a buon fine quando A invia i messaggi RESV per 100 ms non occorre inoltrare la richiesta a monte del punto di confluenza delle richieste

Trasmettitori multipli

- In questo caso i ricevitori devono fare riservazione in relazione a tutti i TSpec di tutti i trasmettitori
- Non necessariamente i TSpec si sommano
  - la modalità di combinazione dei TSpec dipende dalla applicazione
- RSVP prevede differenti stili di riservazione
  - riservazione per tutti i trasmettitori
  - riservazione per n trasmettitori qualsiasi
  - riservazione per alcuni specifici trasmettitori

Funzioni del router

Una volta effettuata la riservazione, ogni router del percorso deve:

- associare ogni pacchetto alla riservazione appropriata
  - *classificazione*
- Gestire i pacchetti nelle code in modo che ottengano il servizio richiesto
  - *scheduling*

Problemi di scalabilità

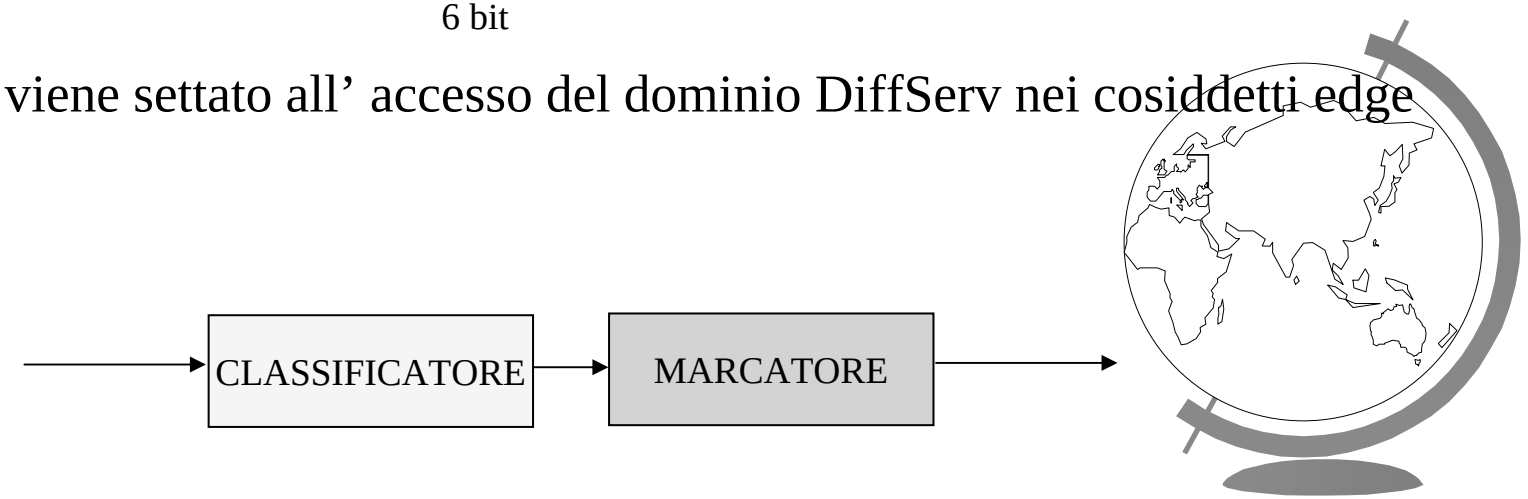
- Nel modello best effort i router non memorizzano alcuna informazione sui flussi che li attraversano
- Nel modello ***Integrated Services*** ogni flusso che passa attraverso un router potrebbe avere una sua specifica riservazione
- Al crescere delle dimensioni della rete il numero di informazioni memorizzate in un nodo potrebbe diventare molto grande
  - Es: link a 2.5 Gbit/s con flussi audio a 64 Kbit/s: numero totale di flussi 39000 ciascuno dei quali richiede riservazione che deve essere rinfrescata periodicamente

Modello a servizi differenziati (RFC2475)

- Le risorse vengono allocate a un piccolo gruppo di classi di traffico anziché a un grande numero di flussi
  - Tali classi sono denominate BA (behavior aggregate)
- Fornisce i componenti funzionali con cui i servizi possono essere costruiti
- I pacchetti vengono identificati dal router come appartenenti all' una o all' altra classe per mezzo di un byte contenuto nella intestazione IP
  - Campo DS (8 bit) della intestazione IPv4 o IPv6

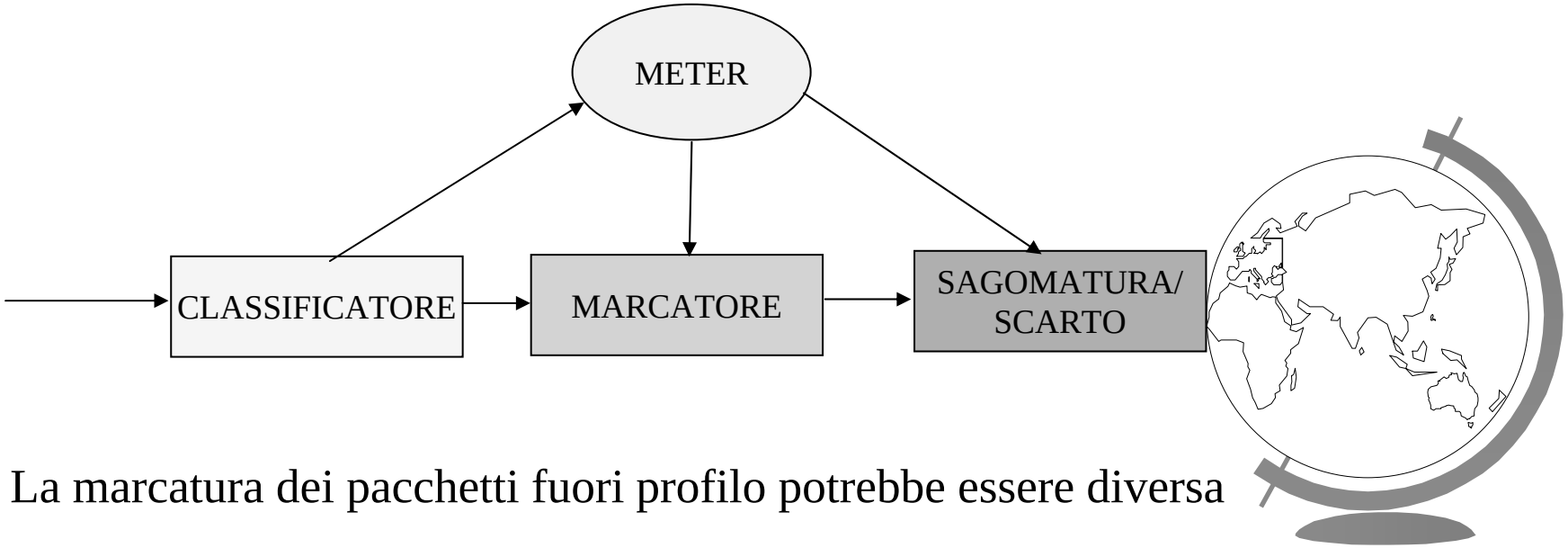
6 bit

  - Il byte viene settato all' accesso del dominio DiffServ nei cosiddetti edge router



Modello funzionale dell'edge router

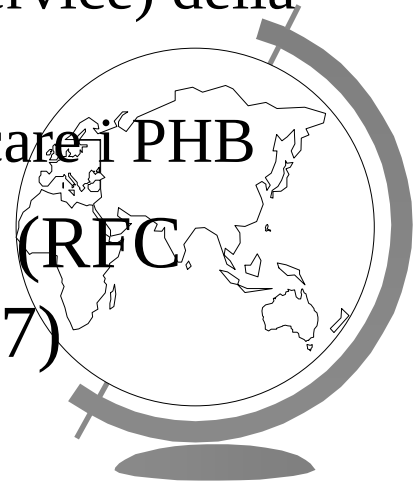
- L'utente stipula un contratto con un ISP per ottenere un servizio prioritario
- Al tempo stesso deve impegnarsi a rispettare alcuni vincoli sulla trasmissione (per es sulla velocità massima)
  - Definizione del profilo di traffico
- Occorre una funzione di misura per determinare le caratteristiche del traffico trasmesso
- Occorre una funzione di sagomatura per far rientrare il profilo nelle specifiche



- La marcatura dei pacchetti fuori profilo potrebbe essere diversa

IETF Differentiated Services

- IETF ha standardizzato le modalità di comportamento dei router del dominio Diffserv
  - PHB: Per Hop Behaviour: definisce il comportamento dei singoli router e non dei servizi end-to-end
- Sono stati definiti diversi PHB per cui occorre più di 1 bit per codificarli
  - IETF ha deciso di usare il byte TOS (Type of Service) della intestazione IP: 6 bit sono allocati come DSCP (Differentiated Service Code Point) per identificare i PHB
- PHB principali: EF (Expedited Forwarding) (RFC 2598) e AF(Assured Forwarding) (RFC 2597)



Expedited Forwarding (RFC 2598)

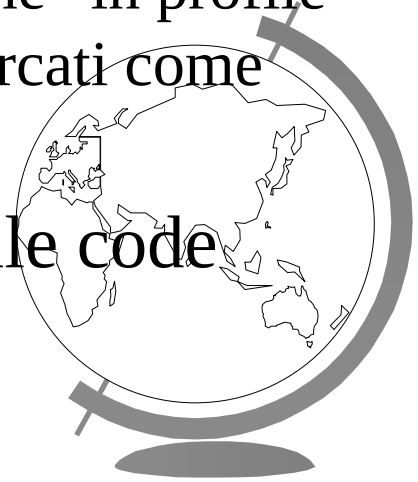
- I pacchetti marcati come EF devono attraversare il router con perdita e ritardo piccoli
- Occorre che la velocità di arrivo di pacchetti EF al router sia non superiore a quella di trasferimento del router.
- La limitazione di velocità sui pacchetti EF avviene configurando gli edge router in modo da consentire una velocità massima di ingresso al dominio per i pacchetti EF
  - strategia conservativa: somma delle velocità dei pacchetti EF che entrano nel domini inferiore alla capacità del link più lento del dominio

Realizzazioni di EF

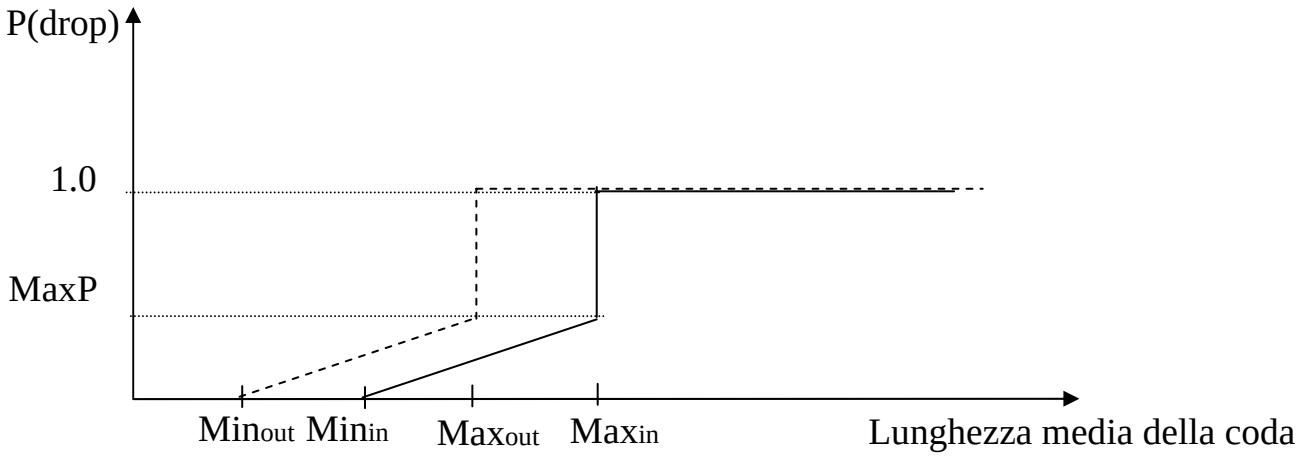
- Si può assegnare ai pacchetti EF una priorita’ stretta su tutti gli altri
- Si possono applicare meccanismi WFQ, con peso per EF selezionato opportunamente per consentire un passaggio veloce dei pacchetti EF consentendo comunque trasferimento anche di pacchetti non EF (per esempio pacchetti di routing)

Assured Forwarding

- E’ un servizio simile al controlled load
- Si garantisce una banda minima e perdita limitata
- Il profilo di traffico per un servizio AF puo’ essere del tipo: “utente X è abilitato a inviare fino a y Mbit/s di traffico assured”
  - se ne invia di meno i pacchetti sono marcati come “in profile” se ne invia di più i pacchetti in eccesso sono marcati come “out profile”
- Si avvale di alcuni algoritimi di gestione delle code
  - RIO, WRED, WFQ



Funzionamento del meccanismo RIO (RED with IN and OUT)



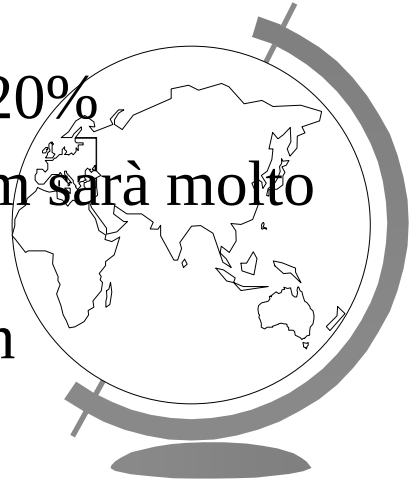
- Ogni classe di traffico ha una diversa curva di dropping
  - la curva *out* ha una soglia minima più bassa della curva *in* e quindi se la congestione è bassa vengono scartati solo pacchetti *out*
  - solo se la lunghezza media della curva supera Min_{in} vengono scartati anche pacchetti *in*
  - i pacchetti *in* da soli devono raramente portare *in* prossimità della congestione

Applicazione dell’ algoritmo RIO

- Si può generalizzare a un maggior numero di curve di dropping
- In questo caso il valore del DSCP indica quale curva di dropping scegliere in modo da gestire diverse classi di servizio
- Questa è l’idea su cui si basa l’ algoritmo WRED (Weighted RED)

Utilizzo di WFQ

- Si utilizza il DSCP per indicare la coda WFQ in cui inserire il pacchetto
- Nel caso più semplice si utilizza un DSCP per il traffico best effort e un altro per il traffico premium
- Il peso da assegnare alla coda premium dipende dal traffico premium offerto
  - Se si assegna peso 4 al best effort e 1 al premium si ottiene una banda per il traffico premium data da $W_{premium}/(W_{premium} + W_{BE})$, in questo caso 20%
  - se il traffico premium è 10%, il servizio premium sarà molto buono con ritardi bassi
  - se il traffico premium è 30%, il servizio premium sperimenterà congestione



Assured Forwarding in IETF (RFC 2597)

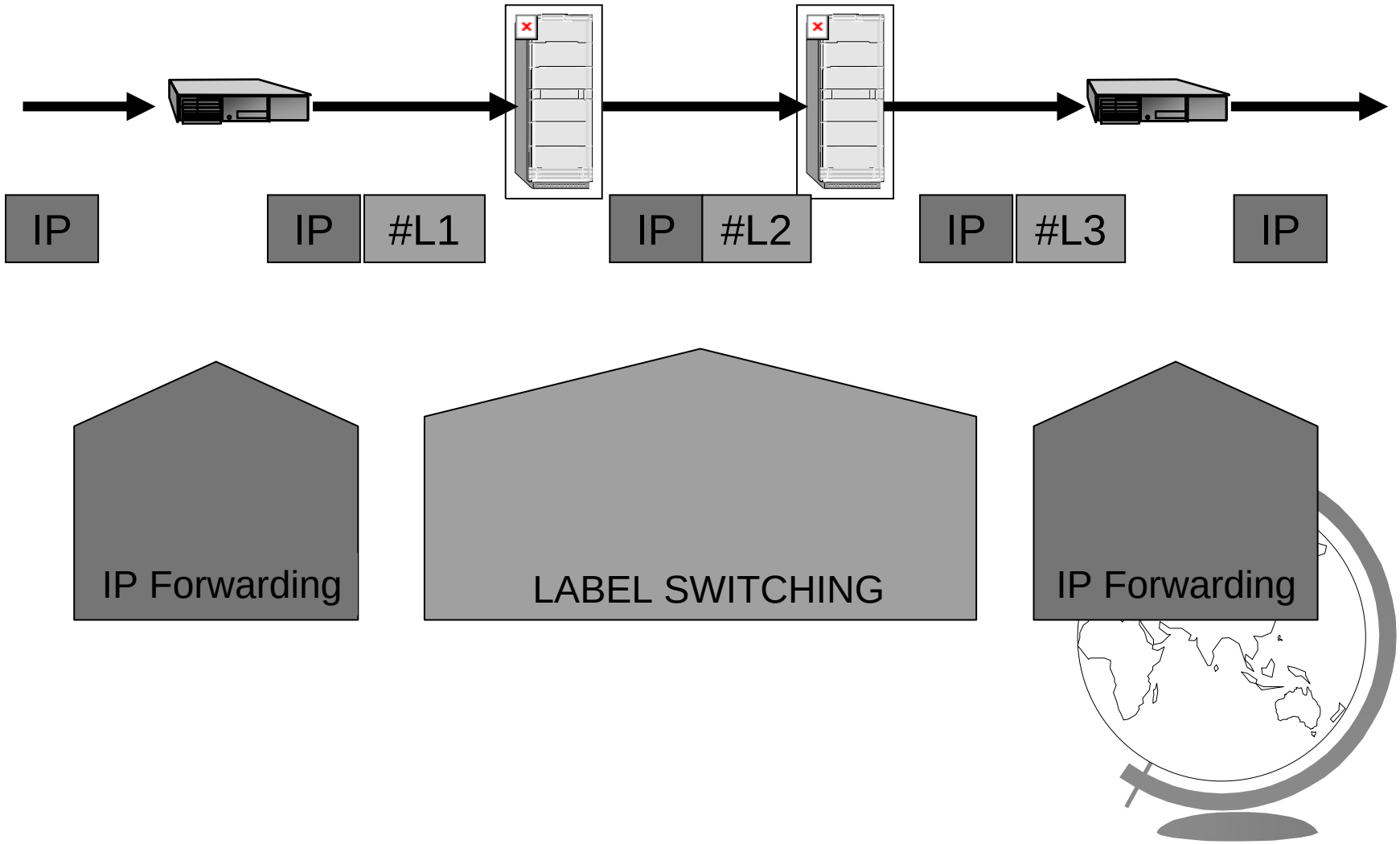
- Combina l'idea della selezione di una coda con peso con la scelta di una curva di dropping
- Si utilizzano 12 diversi valori di DSCP per definire
  - 4 code con pesi diversi
  - 3 livelli di dropping per ciascuna coda

	Classe 1	Classe 2	Classe 3	Classe 4
Basso	001010	010010	011010	100010
Medio	001100	010100	011100	100100
Alto	001110	010110	011110	100110

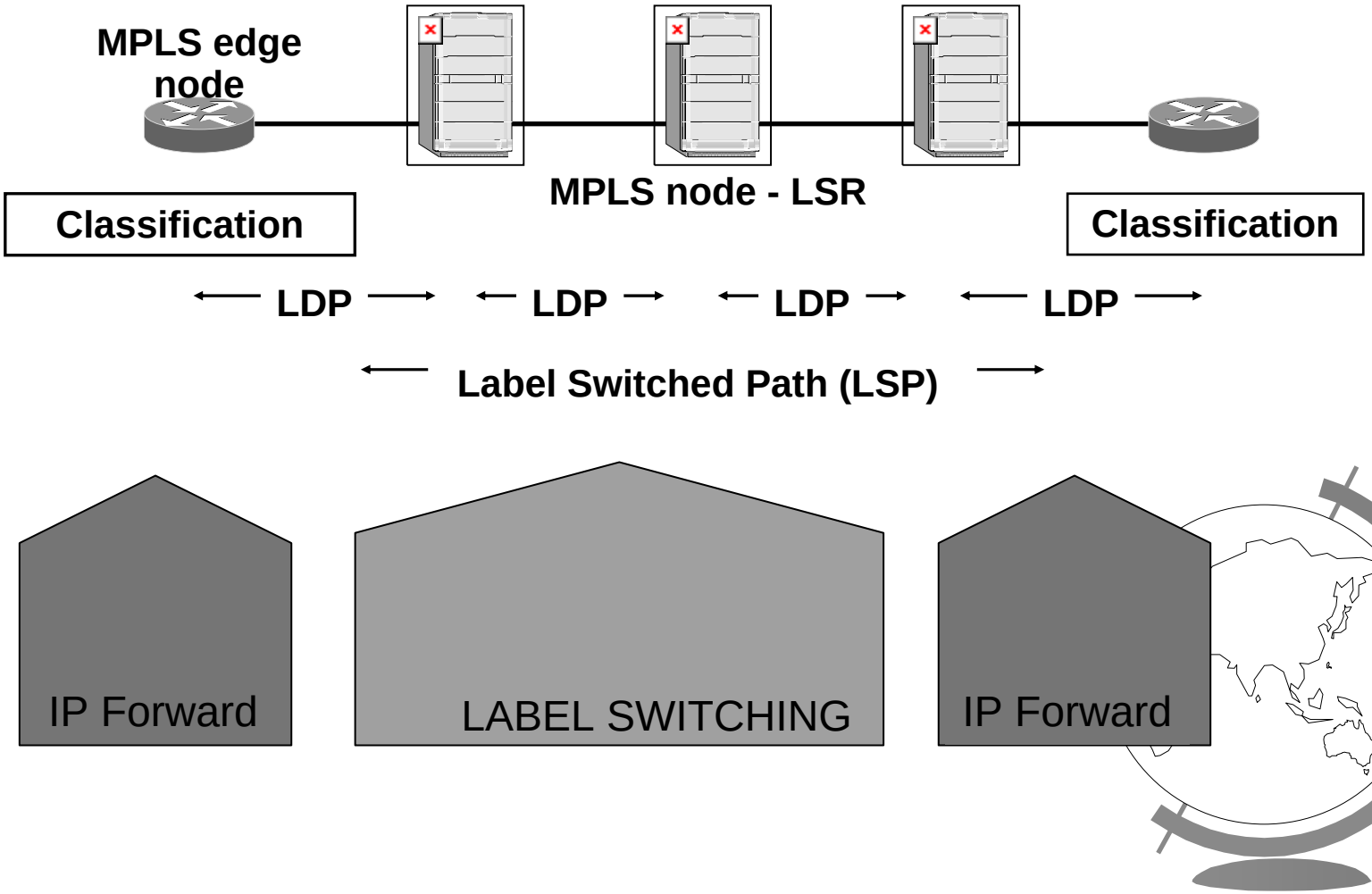
MultiProtocol Label Switching

- ✓ IP *Traffic Engineering* (= gestire il traffico in funzione delle capacità delle linee)
  - ✓ Constraint-based Routing
    - Quali “constraint” considerare? QoS,…
  - ✓ Si può fare a meno di ATM ?
- ✓ *Network Engineering* (= dimensionare le capacità in funzione del traffico)
- ✓ Virtual Private Networks
  - ✓ Soluzioni di tunneling controllato
- ✓ Sinergie IP & ATM (oggi)
- ✓ MPλS (Reti ottiche) (medio-lungo termine: 10 anni?)

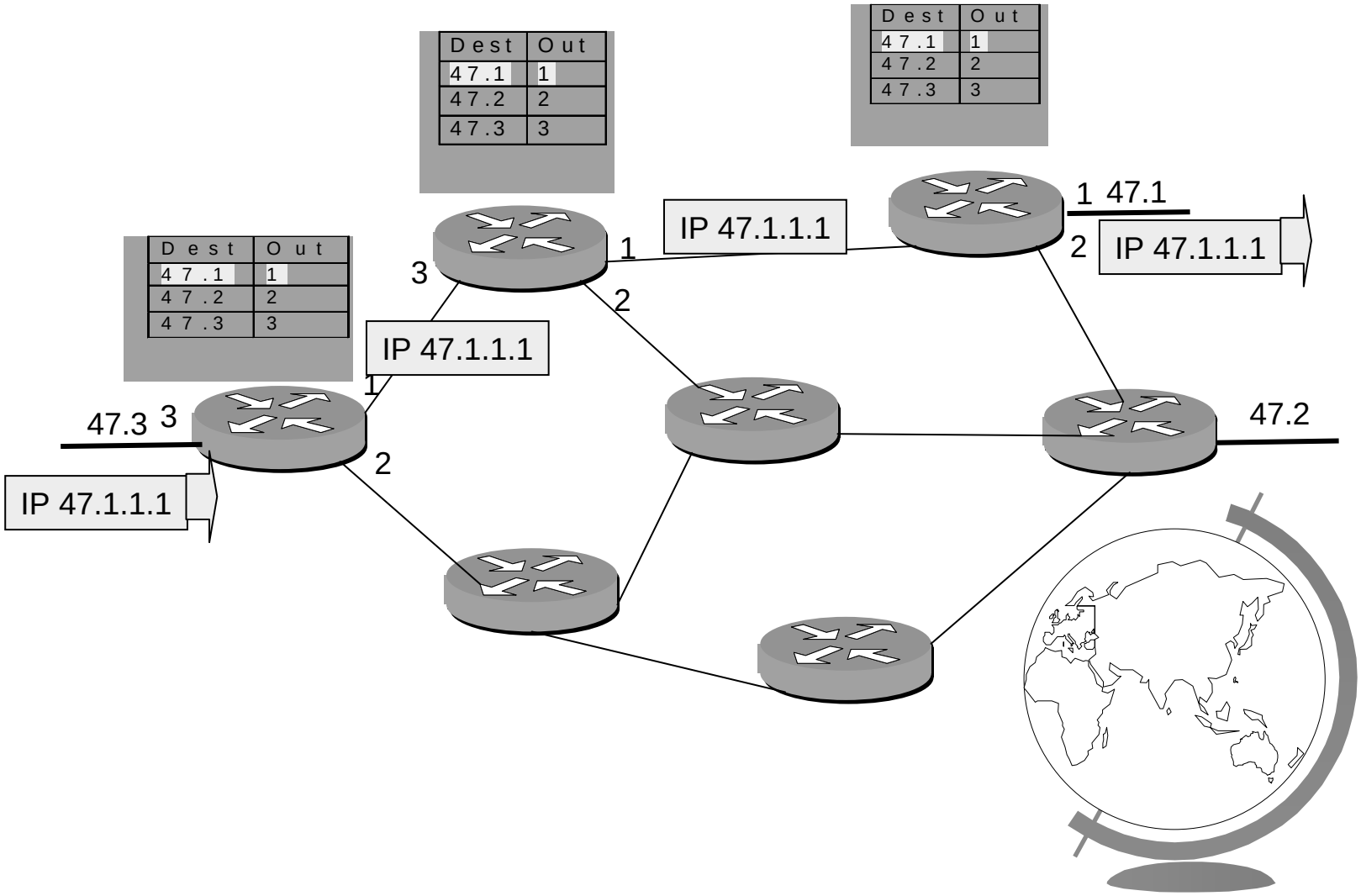
MPLS


IP
IP
#L1
IP
#L2
IP
#L3
IP
IP Forwarding
LABEL SWITCHING
IP Forwarding

MPLS: TERMINOLOGIA



IP FORWARDING

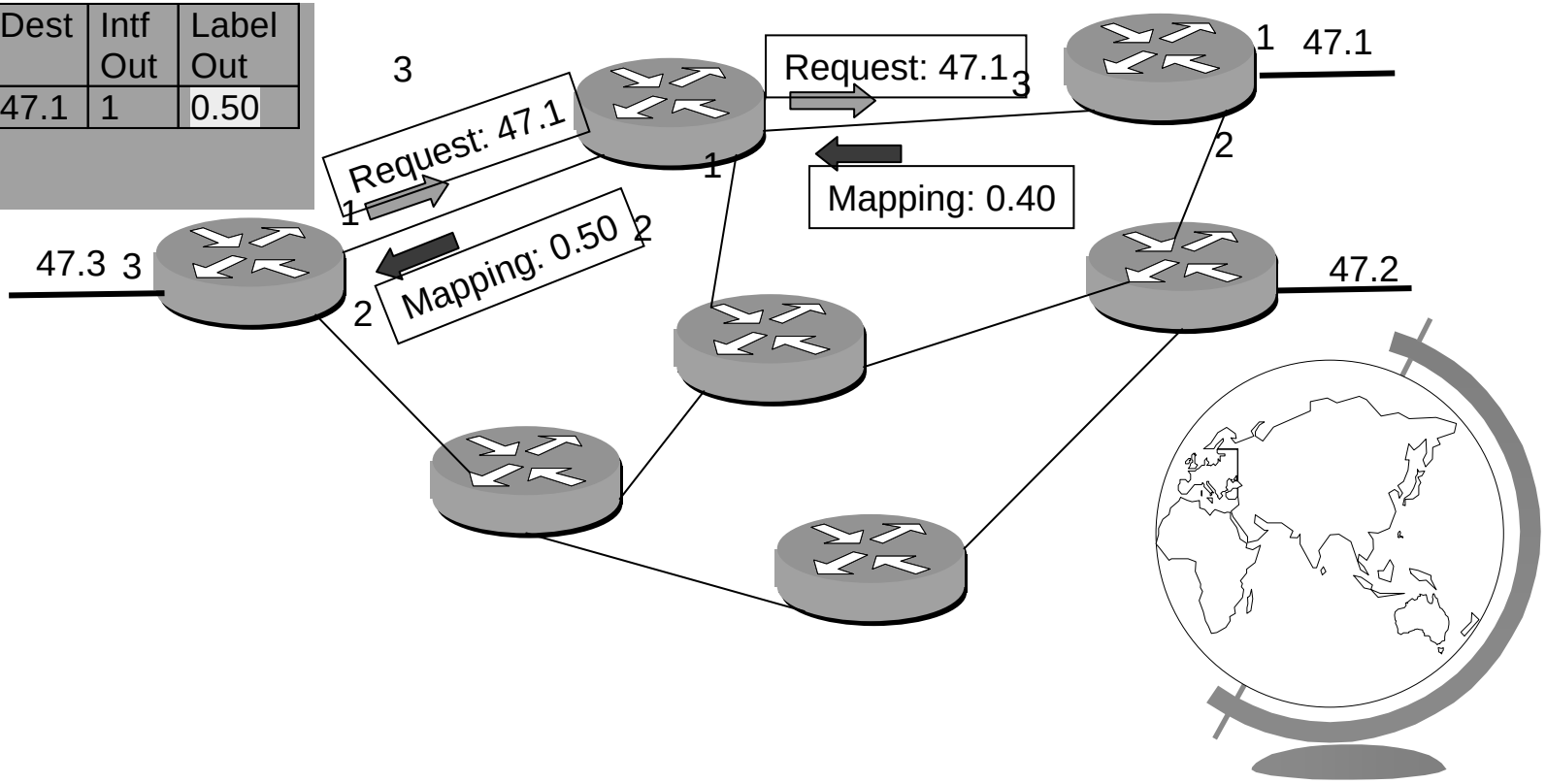


MPLS Label Distribution

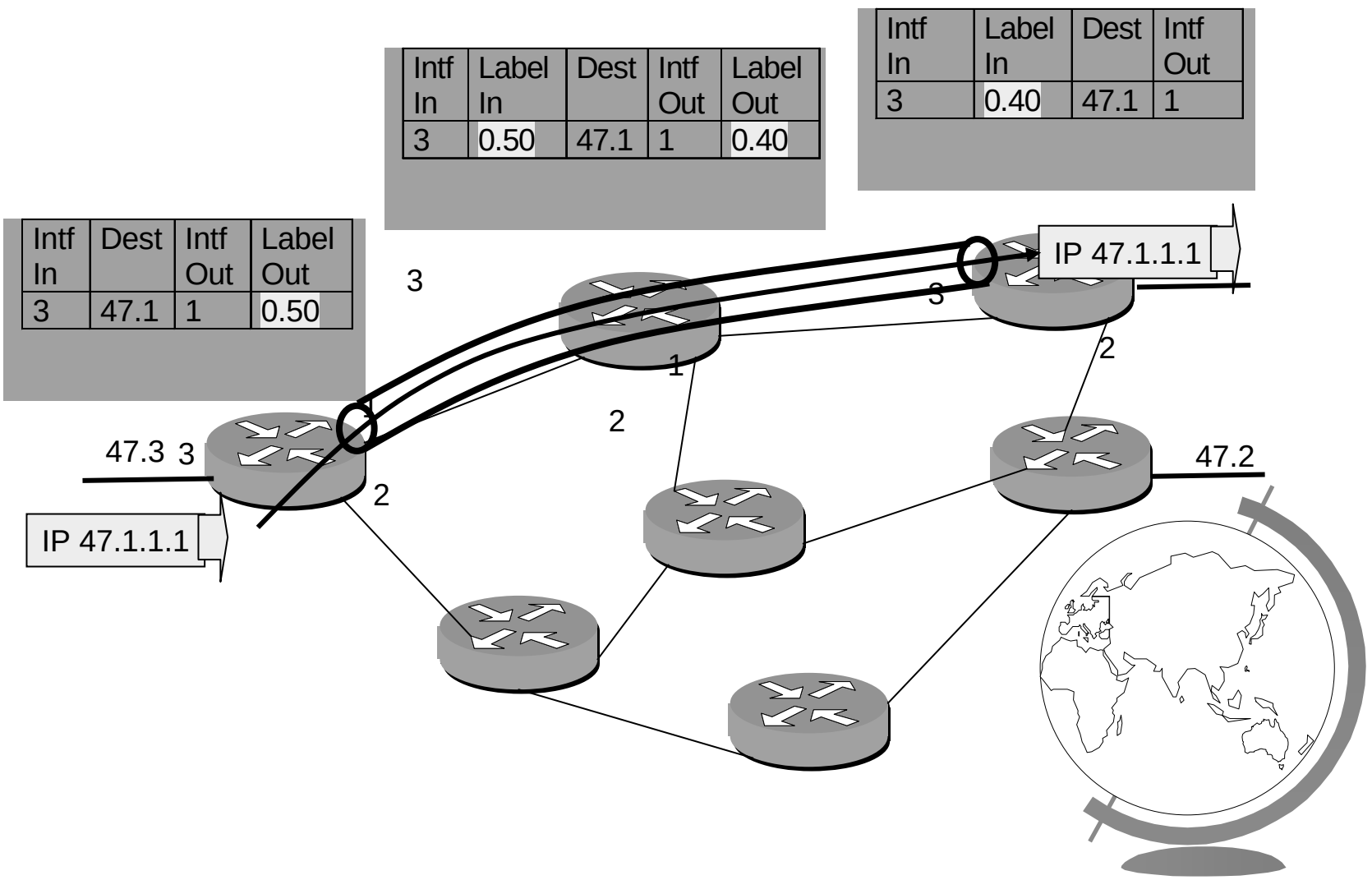
Intf In	Label In	Dest	Intf Out	Label Out
3	0.50	47.1	1	0.40

Intf In	Label In	Dest	Intf Out
3	0.40	47.1	1

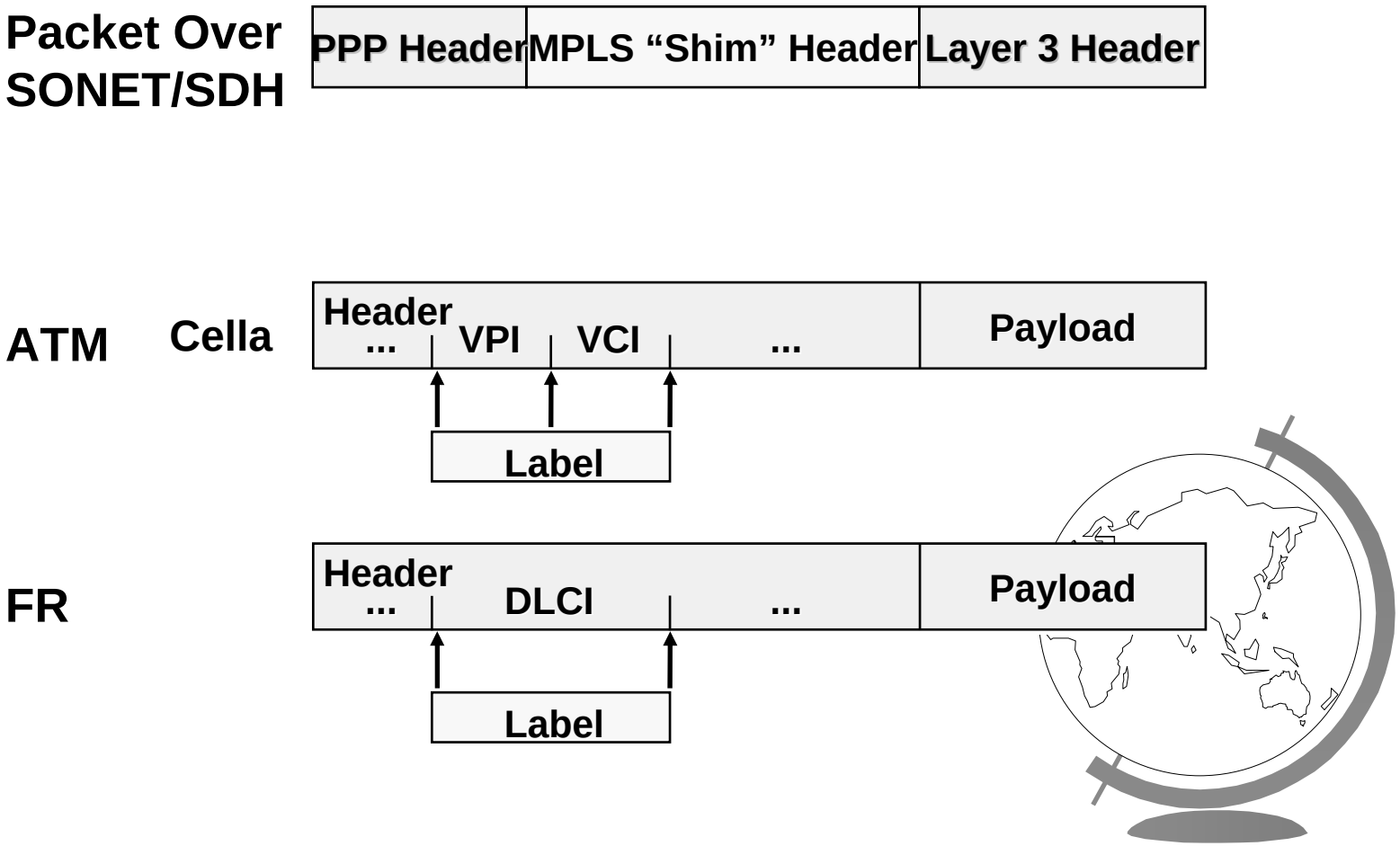
Intf In	Dest	Intf Out	Label Out
3	47.1	1	0.50



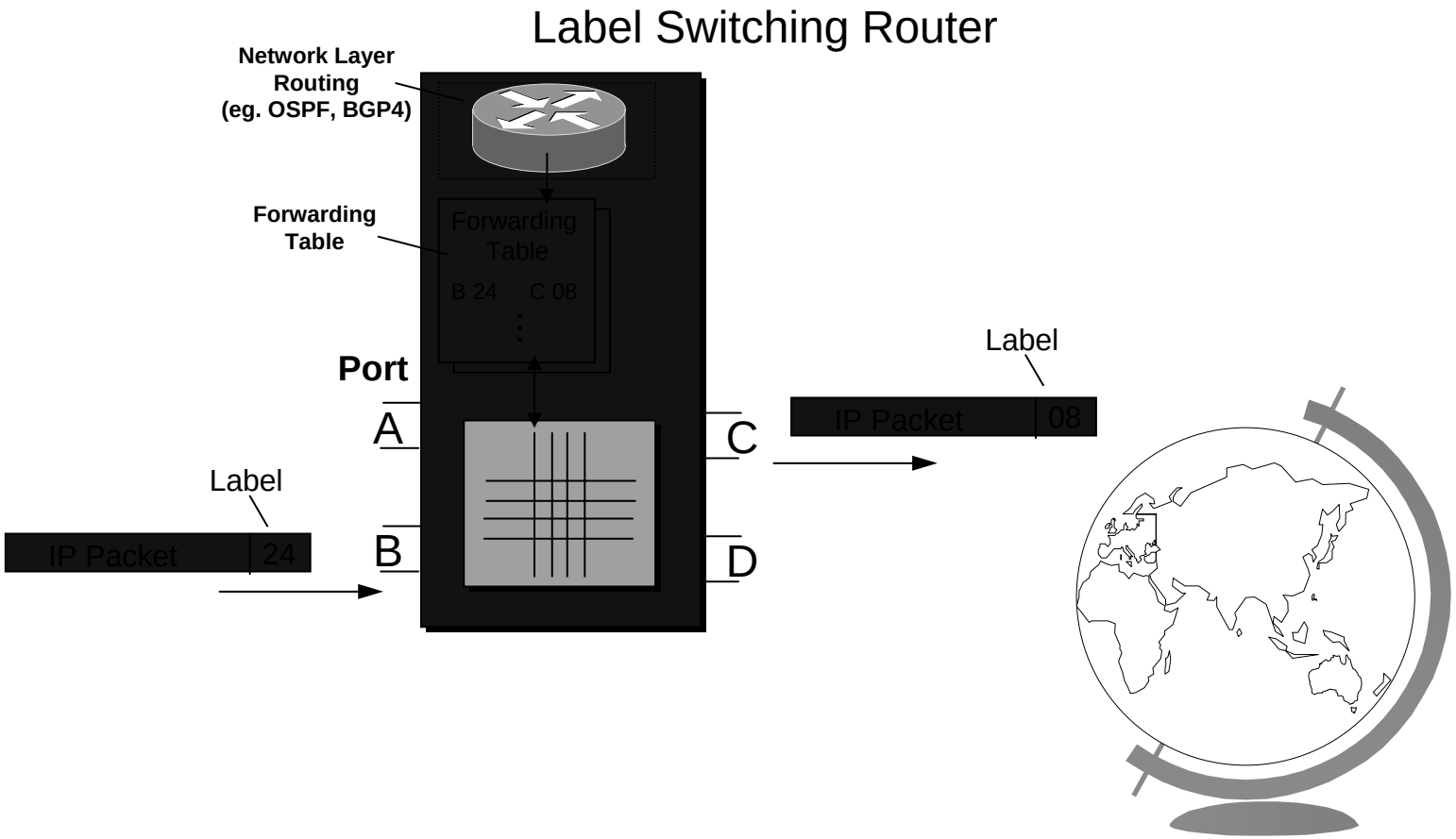
Label Switched Path (LSP)



MPLS label encapsulation



LSR/ATM


Label Switching Router
Network Layer
Routing
(eg. OSPF, BGP4)
Forwarding
Table
Forwarding
Table
B 24 C 08
Port
A
B
C
D
Label
IP Packet 24
Label
IP Packet 08