

Università di Genova  
Facoltà di Ingegneria

---

Telematica  
**? Instradamento multicast**

Prof. Raffaele Bolla



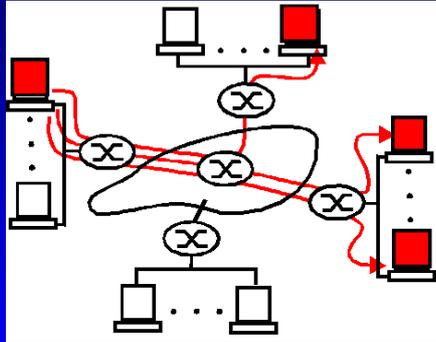
**Instradamento  
Multicast**

---

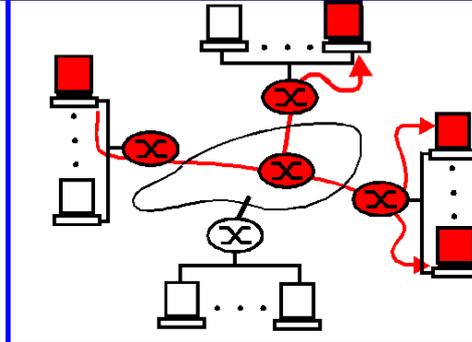
- Si tratta di dare supporto alle trasmissioni in cui un singolo pacchetto debba venir consegnato a molte destinazioni.
- Esempi di casi in cui è richiesto un **multicast** sono molti e coinvolgono
  - Trasferimenti dati (ad es. aggiornamenti di software)
  - *Broadcast* media (audio, video, testo)
  - Applicazioni condivise (*whiteboard*, teleconferenza, ...)
  - Aggiornamento di dati (quotazioni di borsa)
  - Giochi interattivi
  - Localizzazione di risorse (Server, stampanti, ...).

6.2

### Instradamento Multicast



*Multicast realizzato tramite invio di più pacchetti unicast*

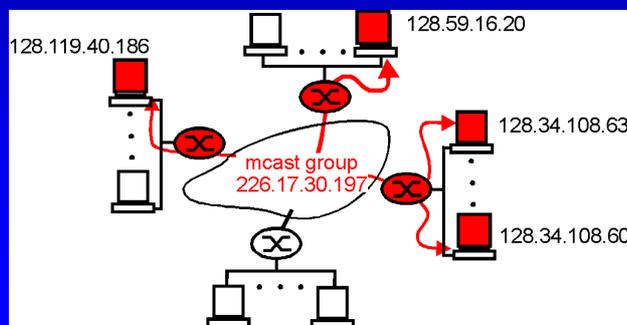


*Multicast realizzato con invio di un singolo pacchetto duplicato nei router*

6.3

### Instradamento Multicast

- Se si usa un singolo pacchetto duplicato all'occorrenza dai router, bisogna che tale pacchetto trasporti l'indirizzo di tutte le destinazioni.
- In alternativa si può astrarre l'indirizzo dalle destinazioni e creare un indirizzo del gruppo (o indirizzo *multicast*)



6.4

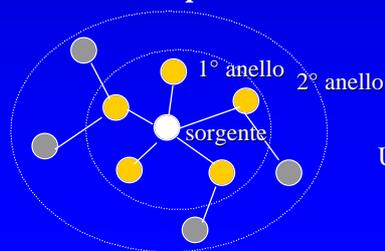
## Instradamento Multicast

- Elemento base del *multicast* è quindi il concetto di **gruppo**
  - Rappresenta l'associazione fra un insieme di trasmettitori ed un insieme di ricevitori. Ogni ricevitore del gruppo riceve i pacchetti inviati ad qualunque trasmettitore del gruppo stesso.
  - Il gruppo concettualmente esiste indipendentemente dalla presenza di elementi componenti.
  - Nella pratica un gruppo nasce nel momento in cui il primo elemento si aggrega e termina la sua esistenza quando tutti gli elementi si sono dissociati.
- Due sono quindi gli aspetti significativi:
  - **La gestione dei gruppi**
  - **La disseminazione dell'informazione (routing vero e proprio.)**

6.5

## Instradamento Multicast

- Si osservi che la potenza del *multicast* è anche legata proprio al disaccoppiamento fra trasmettitori e ricevitori, che permette ad un trasmettitore di localizzare il ricevitore senza conoscerne l'indirizzo specifico.
- Per esempio, un calcolatore che cerca un *print server*, può richiederlo tramite un “*well-know*” indirizzo multicast e utilizzare un meccanismo di tipo “*expanding ring*” per identificare il server più vicino.



Usando il *Time To Live* (TTL) si ricerca in anelli di dimensioni crescenti

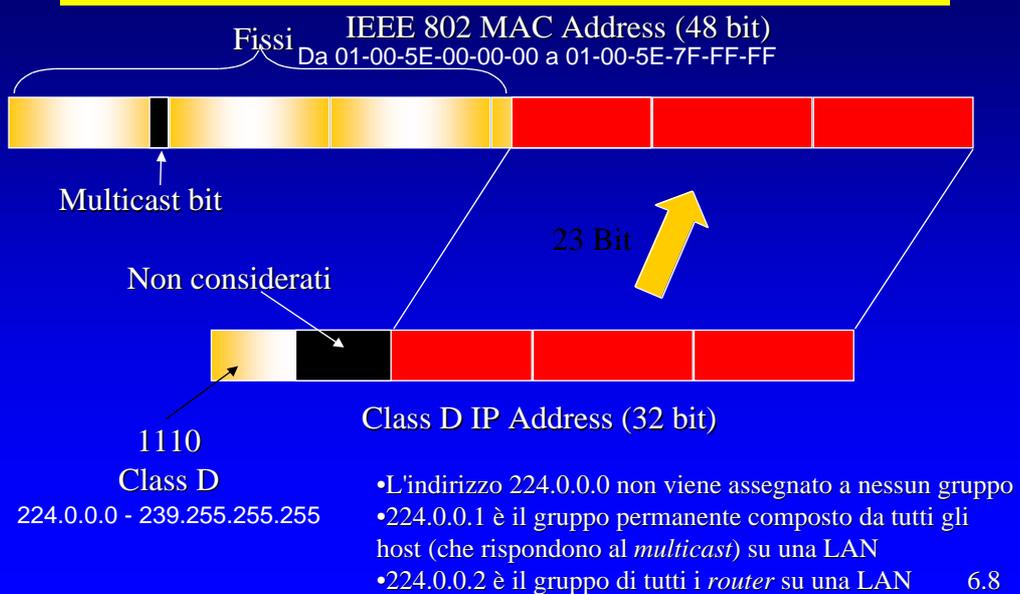
6.6

**Instradamento Multicast**

- Per quanto concerne la realizzazione del *multicast*, si possono distinguere due ambiti:
  - Broadcast LAN (quindi senza attraversare *router*)
  - WAN (fra *router*)
- Le LAN IEEE 802 prevedono un indirizzo di *multicast* che viene utilizzato in modo diretto nel dominio di collisione e, anche tramite VLAN, nel dominio di *broadcast*.
- La gestione della registrazione degli utenti può avvenire a livello di VLAN tramite **GMRP** (*Multicast Registration Protocol*)

6.7

**Instradamento Multicast**



### Instradamento Multicast - IGMP

---

- **Internet Group Management Protocol (IGMP)** è usato dai *Multicast router (Mrouter)* per la gestione dei gruppi *multicast di host* sulle B-LAN.
- La versione attuale dell'IGMP è la numero 2
  - esistono installazioni della 1
  - la numero 0 è obsoleta
- I messaggi IGMP vengono spediti in *multicast* (indirizzo 224.0.0.1 e TTL = 1) per tutti gli *host* sulla LAN e sono incapsulati in un *datagram IP* con campo *protocol* uguale a 2

6.9

### Instradamento Multicast - IGMP

---

- Un *Mrouter* designato sulla LAN invia periodicamente (1 al minuto, al massimo) dei pacchetti di “*query*” a cui ogni *host* interessato risponde in *broadcast* con un elenco dei gruppi a cui ha aderito o vuole aderire.
- Gli *host* rispondono alle *query* generando dei *report*, con cui segnalano all' *Mrouter* tutti gli *host group* a cui appartengono
- In particolare, inviano un *report* per ciascun *host group* a cui sono iscritti

6.10

### Instradamento Multicast - IGMP

- Si osservi che all' *Mrouter* non interessa in numero totale di elementi che aderiscono ad un gruppo, ma solo se ce ne è almeno uno.
- Per evitare troppo traffico e collisioni, quando un *host* A riceve una *query*, ritarda la propria trasmissione di un tempo casuale. Se prima che abbia trasmesso A, un altro nodo B trasmette segnalando la propria adesione agli stessi gruppi di interesse di A, A non trasmette più.
- Per aderire ad un gruppo, un *host* deve
  - configurare la propria interfaccia di rete per ricevere un dato indirizzo *multicast*.
  - Se un altro *host* ha già richiesto di aderire non deve fare altro, altrimenti deve attendere una *query*.

6.11

### Instradamento Multicast - IGMP

Type	Max Resp Time	Checksum
Group Address		

- **Type**
  - 0x11 = *Host Membership Query*: inviate dall' *Mrouter* verso gli *host*, per tenere aggiornata la lista degli *host* group attivi sulla LAN
  - 0x16 = *Host Membership Report*: inviate dall'*host* in risposta alle *query* del *router*
  - 0x17 = *Leave Group*: inviato (opzionalmente) agli *Mrouter* da un *host* per annunciare l'abbandono di un gruppo quando sia l'unico membro

6.12

## Instradamento Multicast - IGMP

---

### ● *Max Resp Time*

- Usato per *Membership Query*: Massimo tempo entro cui deve essere inviata la risposta, se contiene un
  - » Valore piccolo: i *router* sono aggiornati più velocemente sullo stato dei gruppi
  - » Valore grande: i *report* sono più sparsi nel tempo, minore *burstiness*

### ● *Group Address*

- Viene impostato a zero nelle *query* generali per scoprire quali gruppi operano sulla LAN
- Nei *report* e nelle *query* specifiche contiene l'indirizzo dell'*host group* a cui appartiene un *host*

- La versione 2 del protocollo prevede un meccanismo di designazione per il *router multicast*, la versione 3 prevede anche la possibilità di selezionare la sorgente.

6.13

## Instradamento Multicast

---

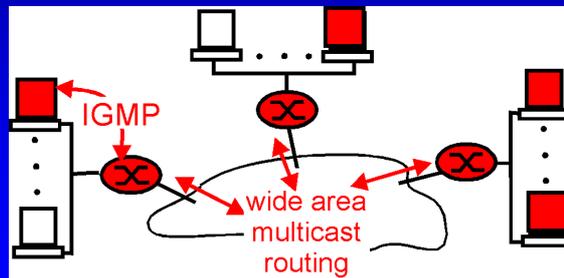
### ● Alcune osservazioni

- La creazione di un gruppo in Internet è una operazione "**receiver-driven**"
- La sorgente non partecipa alla formazione del gruppo e quindi non ne può neppure controllare la composizione
- Allo stesso modo non c'è controllo su chi invia al gruppo:
  - » Sovrapposizione di invii allo stesso gruppo
  - » Uso dello stesso indirizzo con sovrapposizioni di invii fra gruppi diversi
  - » Invii di trasmissioni di disturbo volute
- Dal punto di vista della sicurezza il problema va affrontato a livello di applicazione

6.14

### Instradamento Multicast - WAN

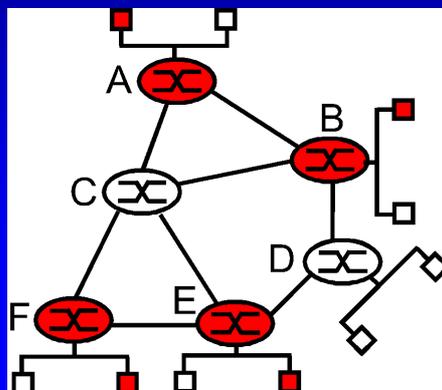
- Una volta che i *router* sono conosciuti la presenza di host ad essi connessi appartenenti a gruppi e sono in grado di inviargli e/o ricevere l'informazione *multicast*, il problema si sposta nel gestire il *routing multicast* fra i *router* (WAN)



6.15

### Instradamento Multicast - WAN

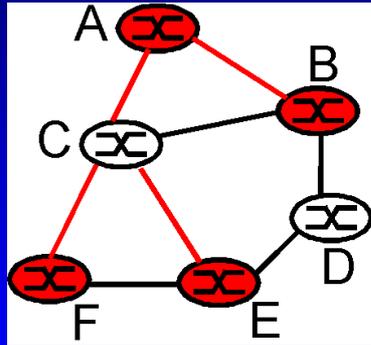
- In linea di principio il *routing multicast* si realizza individuando uno *spanning tree* che comprenda tutti i nodi interessati ad un gruppo ossia un sottoinsieme di tutti quelli della rete.



6.16

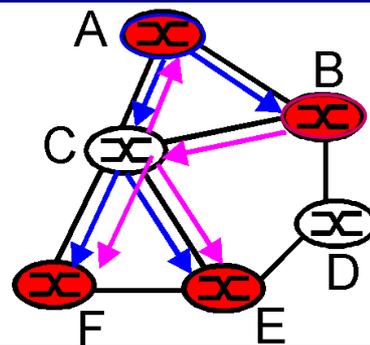
**Multicast WAN**

- Si hanno due possibilità principali:



Group-shared tree

Ossia un unico *spanning tree* per tutti i nodi del gruppo.



Source-based tree

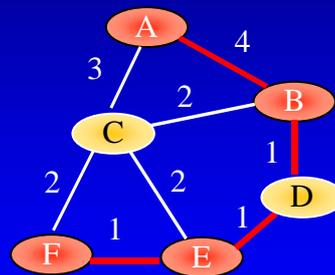
Ossia uno *spanning tree* diverso per ciascuna sorgente.

6.17

**Multicast WAN - Group-shared tree**

- Group-shared tree

- Si tratta di trovare lo *spanning tree* la cui somma dei costi su i *link* che lo compongono sia minima



Il problema di trovare questo albero è noto come "*Steiner Tree Problem*" ed un problema NP-completo.

- Esistono però euristiche che permettono di trovare buone approssimazioni

6.18

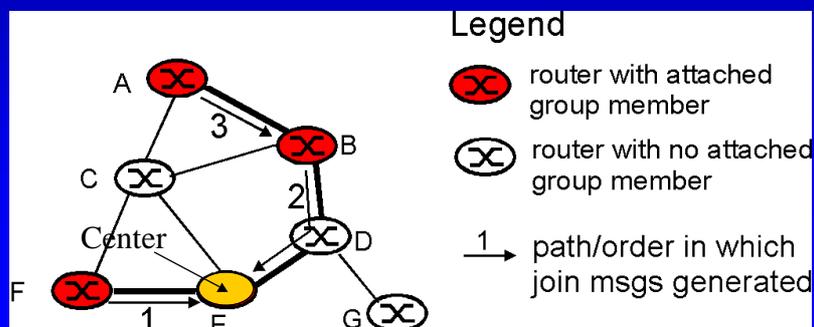
### Multicast WAN - Group-shared tree

- Avendo un unico albero e potenzialmente più sorgenti, non si può ottimizzare rispetto una specifica sorgente.
- Di conseguenza l'unica ottimizzazione fattibile è quella che cerca il **Minimum weight Spanning Tree (MST)**
- Nonostante la presenza di metodi approssimati efficaci per calcolare il *MST*, nessun algoritmo su Internet usa questo approccio.
- Questo perché:
  - Bisogna conoscere il costo di ogni *link* sulla rete
  - Si deve ripetere il calcolo ad ogni cambio di costo
  - Non riesce ad usare facilmente le tabelle di *routing* già calcolate per l'*unicast*
  - Le prestazioni hanno comunque dei limiti perché il costo medio e quello massimo per coppia sorgente-destinazione del gruppo sono elevati.

6.19

### Multicast WAN - Group-shared tree

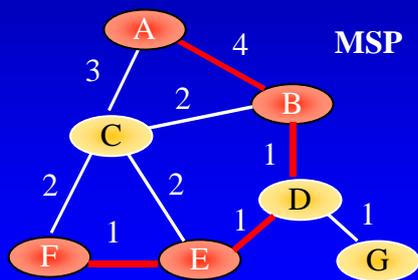
- Un approccio alternativo in questa categoria è la "Center Based Tree", che fa uso di un nodo di riferimento (*center node* o *rendezvous point* o *core*)
- Tutti i *router* con un *host* che aderisce ad un gruppo, inviano un messaggio di *join* lungo il percorso *unicast* verso il nodo di centro
- Fino a che il messaggio o raggiunge il centro o incontra un *router* già parte del gruppo, crea un percorso dell'albero



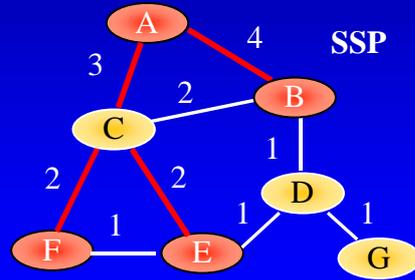
6.20

**Multicast**  
**WAN - Source-based tree**

- Gli algoritmi LS (Dijkstra), ricavano, in sostanza, una *short-path spanning tree* per ogni nodo.
- Depurando l'informazione della rete dei *router* non interessati al gruppo, ogni *router* può calcolare lo *short-path spanning tree* del gruppo per qualunque sorgente.



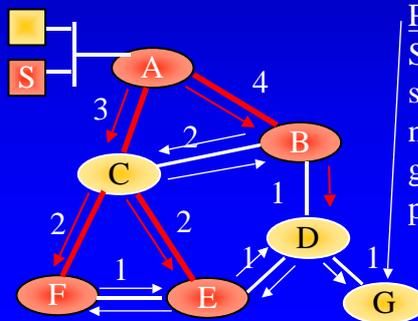
Costo = 1+1+1+4 = 7  
Dist. Media = 5.67 (max=7)



Costo = 3+2+4+2 = 11  
Dist. Media = 4.67 (max = 5) 6.21

**Multicast**  
**WAN - Source-based tree**

- Un modo elegante e più semplice per realizzare l'instradamento *multicast* è utilizzare il *Reverse Path Forwarding (RPF)*
- Un pacchetto proveniente dalla sorgente *S* viene inviato su tutte le uscite (tranne quella da cui è arrivato) solo se arriva dalla interfaccia che corrisponde al percorso più corto verso *S* (il *next hop* con destinazione *S* della RT).



**PROBLEMA**

Se dietro a questo nodo ci sono altri *router*, anche se nessuno di essi fa parte del gruppo, riceveranno tutti il pacchetto

6.22

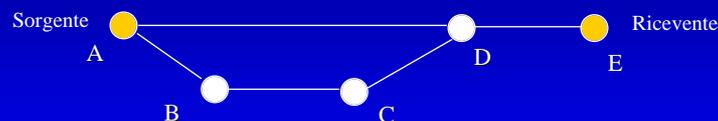
### Multicast WAN - Source-based tree

- Un modo per migliorare l' RPF è dato dall'utilizzo della tecnica di *pruning* (potatura), ossia l'esclusione dalla disseminazione dell'informazione dei nodi "foglia" (*leaf*) o terminali non interessati ad un gruppo
- Per far ciò bisogna
  - Identificare le foglie;
    - » Usando un ERPF si può verificare se si è sul percorso a lunghezza minima fra la sorgente e i nodi direttamente connessi; i nodi che non sono su percorsi sono nodi foglie.
  - Comunicare l'assenza di partecipanti al gruppo;
    - » Inviando messaggi di *pruning* per quel gruppo (sorg.).
- L'eventuale ri-inserimento può avvenire tramite richiesta esplicita (*graft*) o automaticamente legando il *pruning* ad un *timeout*.

6.23

### Instradamento Multicast - WAN

- Si osservi, però, che l'RPF non evita la presenza di copie multiple dello stesso pacchetto e quindi non corrisponde a utilizzare uno ST.
- Un miglioramento (**Extended RPF**) lo si ottiene imponendo che un nodo C invii il pacchetto verso D solo se il percorso più corto da A (sorgente) a D include C stesso.



- Questo implica però che ogni nodo sappia di essere transito per quella sorgente:
  - *Split horizon* invia infinito ai nodi "next hop"
  - L'informazione può essere inviata esplicitamente con i DV (serve un solo bit associato ai vettori verso destinazioni per cui il nodo ricevente è transito).

6.24

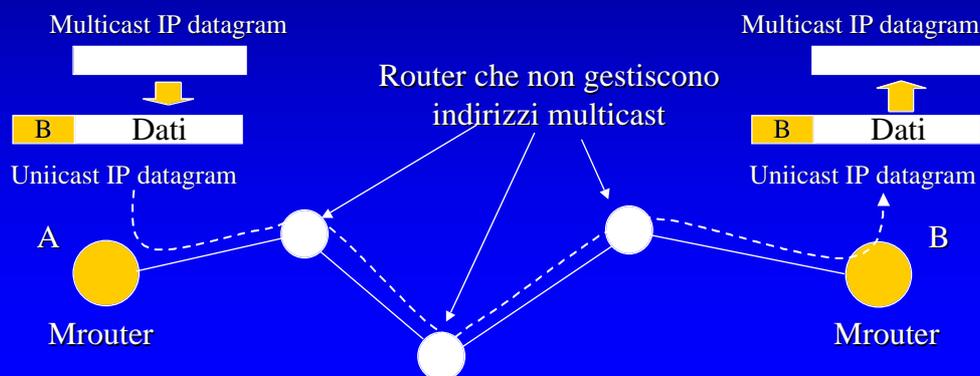
## IP Multicast Mbone

- Le funzionalità relative al *multicast* non sono presenti in tutti i *router*
- Per fornire servizi multicast su Internet è stata realizzata una rete virtuale che interconnette tutti gli *Mrouter* che è stata chiamata:  
*Multicast Backbone (Mbone)*
- Mbone nasce nel 1992 in forma sperimentale e permette a chi ci si connette di realizzare servizi *multicast* su WAN.

6.25

## IP Multicast Mbone

- Se due *Mrouter* non sono direttamente connessi, viene creata fra loro una connessione attraverso un *tunneling*:



6.26

### IP Multicast

#### Mbone

---

- Ogni *Tunnel* è definito da
  - *Local end-point* (fisso)
  - *Remote end-point* (fisso)
  - *Metric* (dinamico)
    - » Costo del tunneling
  - *Threshold* (dinamico)
    - » Valore minimo del TTL perché il pacchetto possa essere instradato nel tunnel (ogni *Mrouter* decrementa TTL di 1 nel pacchetto *multicast*).

6.27

### IP Multicast

#### Protocolli

---

- All'interno della rete virtuale di Mbone vengono usati i protocolli di *routing multicast*
- Per quanto concerne gli IGP *multicast*:
  - Protocolli *flood and prune*
    - » **Distance Vector Multicast Routing Protocol** (DVMRP)
    - » **Protocol Independent Multicast Dense Mode**(PIM-DM)
  - **Multicast OSPF** (MOSPF)
  - Protocolli *Center Based Tree* (CBT)
    - » **Core Based Tree** (CBT)
    - » **PIN - Sparse Mode** (PIN-SM)

6.28

## IP Multicast

**Protocolli *flood and prune* - DVMRP**

---

- ***Distance Vector Multicast Routing Protocol (DVMRP)***
- E' un protocollo DV definito nel RFC 1075
- Ignora le informazioni relative ad altri protocolli (*unicast*) e realizza un proprio DV classico con metrica in numero di *hop* effettuati sulla rete virtuale MBone.
- Usando la propria RT applica un *Reverse Path Forwarding* usando messaggi espliciti per le procedure di *prune*, ossia il traffico fluisce ovunque ed i nodi coinvolti si "potano" esplicitamente.

6.29

## IP Multicast

**Protocolli *flood and prune* - PIM-DM**

---

- ***Protocol Independent Multicast Dense Mode (PIM-DM)***
- E' molto simile al DVMRP ma, a differenza di questi, utilizza la RT dell'instradamento *unicast*.
- Sia il DVMRP che il PIM-DM non sono adatti a operare in modo globale, infatti costringono tutti i nodi non interessati raggiunti a "potarsi".
- Quando l'utenza è distribuita in modo "denso" (per esempio all'interno di una organizzazione) sono molto efficaci.

6.30

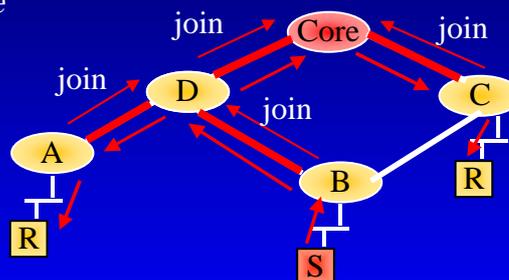
### IP Multicast MOSPF

- E' un LS che estende le funzionalità dell'OSPF per la gestione del *multicast*.
- E' definito dall'RFC 1584
- Estende il DataBase dei LS per memorizzare anche i diversi gruppi attivi presso gli altri *Mrouter*.
- Usando il Database esteso ogni *Mrouter* calcola gli ST troncati in modo autonomo.
- E' adatto per gruppi *multicast* a bassa densità, ma è poco scalabile perché richiede, in ogni nodo, della informazione esplicita sui partecipanti ai gruppi.

6.31

### IP Multicast Protocolli *Center Based Tree* - CBT

- Il **Core Base Tree** è cronologicamente il primo dei protocolli *Center Based Tree*



- Il CBT costruisce un albero bidirezionale perché i pacchetti possono viaggiare sia in direzione core che nella direzione opposta, a seconda della posizione della sorgente

6.32

## IP Multicast

**Protocolli Center Based Tree - CBT**

- La sorgente non deve necessariamente appartenere al *Tree*, in ogni caso il pacchetto viene inviato verso il *Core*, il primo nodo dell'albero che raggiunge, viene propagato sull'albero stesso.
- Può esserci più di un *Core*
- I limiti sono
  - posizionare il *Core* opportunamente è difficile e, se il *Core* non è ben posizionato, l'albero è inefficiente
  - Non si ha un metodo consolidato per legare l'indirizzo del *Core* e quello del gruppo
- Pregi
  - Efficiente per quanto concerne lo stato da mantenere nei *router*, solo informazione sulle porte di *forwarding* per il gruppo e nessuna informazione sulle sorgenti
  - Si scala meglio dei *flood and prune* su gruppi sparsi

6.33

## IP Multicast

**Protocolli Center Based Tree - PIM Sparse Mode**

- Nel *PIM Sparse Mode* il nodo di riferimento si chiama *Rendezvous Point* invece che *Core*, ma ha le stesse funzioni.
- Un *receiver* che voglia aggregarsi ad un gruppo manda un messaggio di *join* al RP. I *router* che il messaggio incontra registrano la presenza del percorso *multicast* creando un albero ma unidirezionale, ossia dal RP verso i ricevitori
- Una sorgente invece invia il pacchetto al *router*, che lo incapsula in un altro pacchetto *unicast* e lo invia al RP. Il RP lo estrae e lo invia sull'albero del gruppo corrispondente

6.34

## IP Multicast

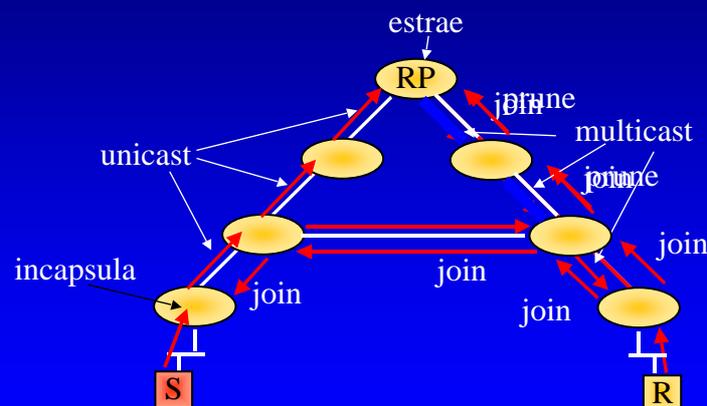
### Protocolli Center Based Tree - PIM Sparse Mode

- Questo significa che la distribuzione dei pacchetti inizia sempre dal RP e che l'albero di distribuzione è unidirezionale.
- La destinazione ha però la possibilità di cambiare l'albero di distribuzione:
  - Quando riceve i primi pacchetti dalla sorgente, invia alla sorgente stessa un messaggio di *join* diretto, che ovviamente raggiunge la sorgente lungo lo *Shortest path*
  - ogni *router* attraversato dal *join* attiva il *forwarding* per quel gruppo-sorgente.
  - Quando la destinazione comincia a ricevere i pacchetti dal nuovo percorso, invia un *prune* sull'albero principale
  - Se tutte le destinazioni fanno la stessa operazione, per quella sorgente si crea uno *Shortest Path Spanning Tree* di distribuzione per il gruppo

6.35

## IP Multicast

### Protocolli Center Based Tree - PIM Sparse Mode



6.36

## IP Multicast

**Protocolli Center Based Tree - PIM Sparse Mode**

---

- L'eventuale inefficienza dell'albero creato con il RP, viene corretta creando alberi esplicitamente.
- D'altro canto sorgenti che generano flussi ridotti non aumentano eccessivamente il carico di informazioni di stato perché possono usare l'albero del RT.
- Esiste un meccanismo per la scelta degli RT di riferimento che riduce la scalabilità dell'algoritmo che resta adatto anche per domini ampi ma non enormi.

6.37

## IP Multicast

**IP Multicast - EGP**

---

- Tutte le tecniche viste hanno dei limiti di scalabilità, che non le rende adatte ad essere applicate in ambito multi dominio.
- Inoltre fra domini diversi spesso ci sono *router* che non supportano il *multicast*.
- Al momento, in ambito EGP, esiste una soluzione che vede l'uso di due protocolli:
  - **Multiprocol Extension for BGP4 (MBGP)**
  - **Multicast Source Discovery Protocol (MSDP)**

6.38

## IP Multicast

**IP Multicast - MBGP**

---

- E' una estensione del BGP che permette di costruire ed aggiornare tabelle di *routing* multiple.
- Questa caratteristica permette di mantenere una tabella separata che costruisca una connettività per i *router* con capacità di *multicast*
- Tale tabella può essere sfruttata da algoritmi PIM per inviare messaggi di *join*.

6.39

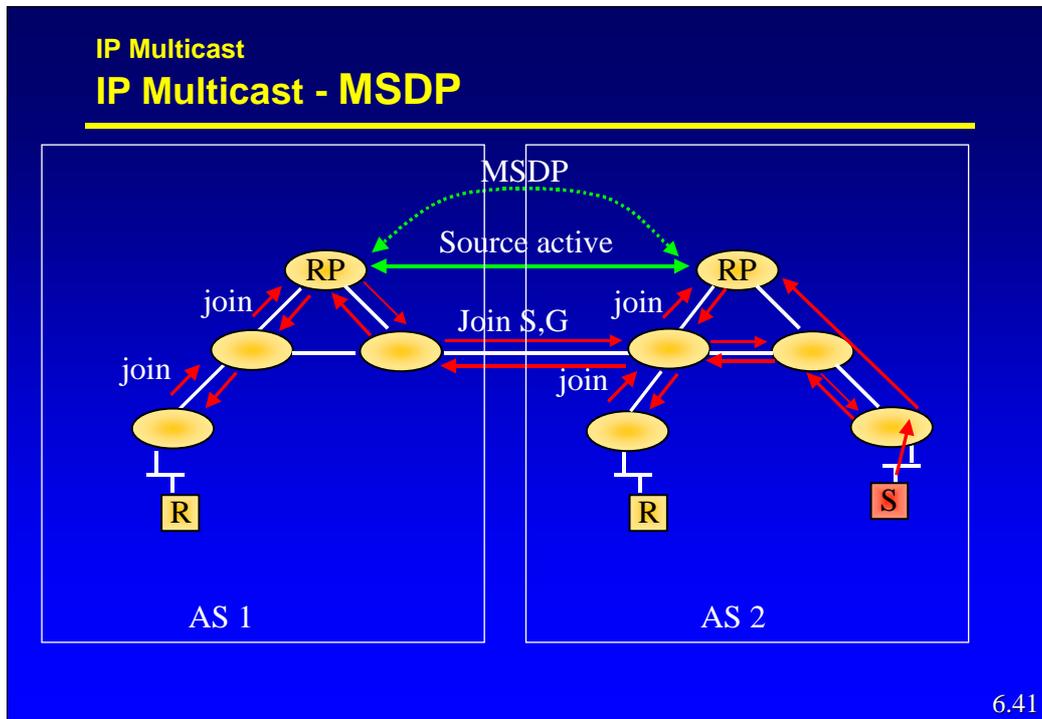
## IP Multicast

**IP Multicast - MSDP**

---

- Anche usando il PIM-SM, domini diversi in genere non vogliono dipendere da RP che non siano al loro interno.
- Questo protocollo permette, usando anche il MBGP, di far dialogare RP in domini diversi per creare alberi misti.

6.40



**IP Multicast**  
**IP Multicast - Scope**

- Il *multicast* consuma molte risorse, quindi è importante limitarne la dispersione (“*scope*”) sia per ragioni di prestazioni sia per sicurezza (mandare molti multicast può essere un modo per bloccare una rete); inoltre permette il riutilizzo di indirizzi
- Due modalità
  - TTL, si fissa una soglia al di sotto della quale il pacchetto multicast non viene propagato dei Border Router in Europa la soglia è 64)
  - Amministrativa: difficile da gestire specialmente in presenza di aree sovrapposte.

6.42