Communications de groupe dans l'Internet

Olivier Togni

Université de Bourgogne

IEM/LIB

o.togni.u-bourgogne.fr

olivier.togni@u-bourgogne.fr

modifié le 24/09/2020

Plan + références

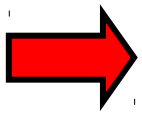
- *Problématique*
- *Protocoles de routage*
- *Applications multicast*
- *Protocoles pour le Multimédia*

- *IP Multicast, Volume II: Advanced Multicast Concepts and Large-Scale Multicast Design, First Edition, Cisco Press, 2018*
- *Introduction to Data Multicasting, IP Multicast Streaming for Audio and Video Media Distribution. L. Harte, ALTHOS, 2008*
- *Multimedia Multicast on the Internet. A. Benslimane, ISTE, 2007*

Multicast: définition et buts

Envoyer les mêmes données à N récepteurs

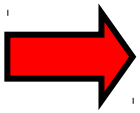
- En minimisant les ressources réseau:
  - en minimisant le nombre de copies d'un paquet
  - en évitant les boucles
- Si possible en optimisant la livraison (délais,…)



Un paquet devrait parcourir un arbre dont les feuilles sont les récepteurs

Arbre Multicast

Pour un réseau avec un ensemble X de noeuds, $S \subset X$ de sources de diffusion et $T \subset X$ de récepteurs (pas forcement disjoint de S) pour un groupe G donné, construire un arbre reliant tous les noeuds de $S \cup T$ (mais qui peut contenir d'autres noeuds de X)



Arbre qui minimise le coût (nombre de liens utilisés) = arbre de Steiner

Applications utilisant le multicast

- De un vers plusieurs (une seule source)
  - Télé-séminaire
  - Diffusion vidéo, radio,…
  - Protocoles de gestion (ex: diffusion des LSA dans OSPF)
- De plusieurs vers plusieurs
  - Téléconférence
  - Jeux en réseau

Hypothèses (Deering, 91)

- Réseau de type IP
  - Gestion et fonctionnement décentralisé, distribué
  - Aspect dynamique du groupe
- Paquets IP multicast même format qu'unicast
- Adr multicast identifie un ensemble de récepteurs (classe D en IPv4) 224.0.0.0 -> 239.255.255.255
- Groupe = Adr multicast
- Pas de connaissance explicite des membres du groupe
  - Les sources ne se déclarent pas

Modèle de Deering, suite

- Multipoint à multipoint si les sources sont aussi réceptrices
  - Possibilités d'un seul arbre pour N sources (la source est identifiée dans le paquet)
- Séparation des signalisations IGMP (adhésions/retraits) et du routage

Adressage Multicast

Mapping des adresses multicast sur le LAN (pas de ARP!): utilise un bloc d'adresses IEEE réservées pour multicast
01:00:5E:00:00:00 -> 01:00:5E:7F:FF:FF
les trois derniers octets de l'adresse MAC sont les 23 derniers bits de l'adresse IP.

Ex: 224.5.0.17 -> 01:00:5E:05:00:11

IGMP Internet Group Management Protocol

- Gère les adhésions, retraits aux groupes
- Chaque routeur désigné (DR) sur un LAN doit savoir pour chaque groupe, s'il y a au moins un récepteur local
- 3 types de message:
  - *join (report)*: émis par une station qui veut adhérer à un groupe, adressé à un routeur sur le LAN
  - *leave*: émis par une station pour avertir le routeur désigné de son retrait du groupe
  - *query*: émis par le DR pour demander s'il y a encore des récepteurs présent pour tel groupe

Routage multicast

- Permet l'acheminement d'un paquet multicast de la source vers les récepteurs
- Arbre construit par signalisation entre les routeurs
  - Semi-explicite (inondation/élagage)
  - Explicite (adhésion/retrait)
- Distinction entre
  - Mode dense/mode épars
  - Intra ou inter domaine

Routage en mode dense

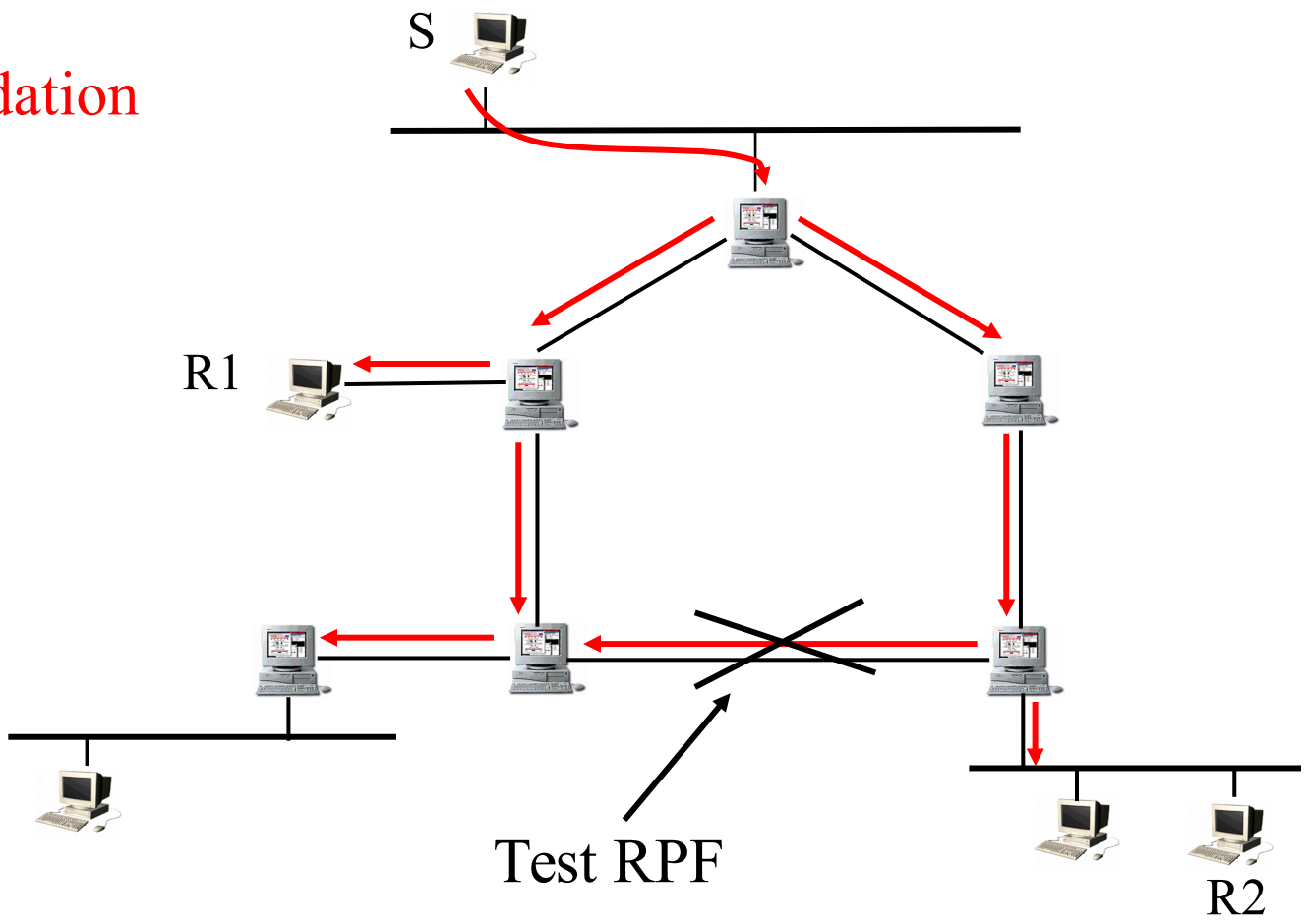
- Inondation du réseau (flood) et élimination des branches inutiles (prune)
  - DVMRP utilise son propre routage unicast (à la RIP)
  - PIM-DM utilise le routage unicast présent
- Un arbre par source (pas scalable)
  - O(S*G) entrées par routeur
- Repose sur le RPF Check

RPF Check

- B accepte un paquet multicast venant de A s'il arrive par la route unicast de B vers A
  - Permet d'éviter les boucles
  - La route unicast est connue de B
  - Problèmes si routes asymétriques

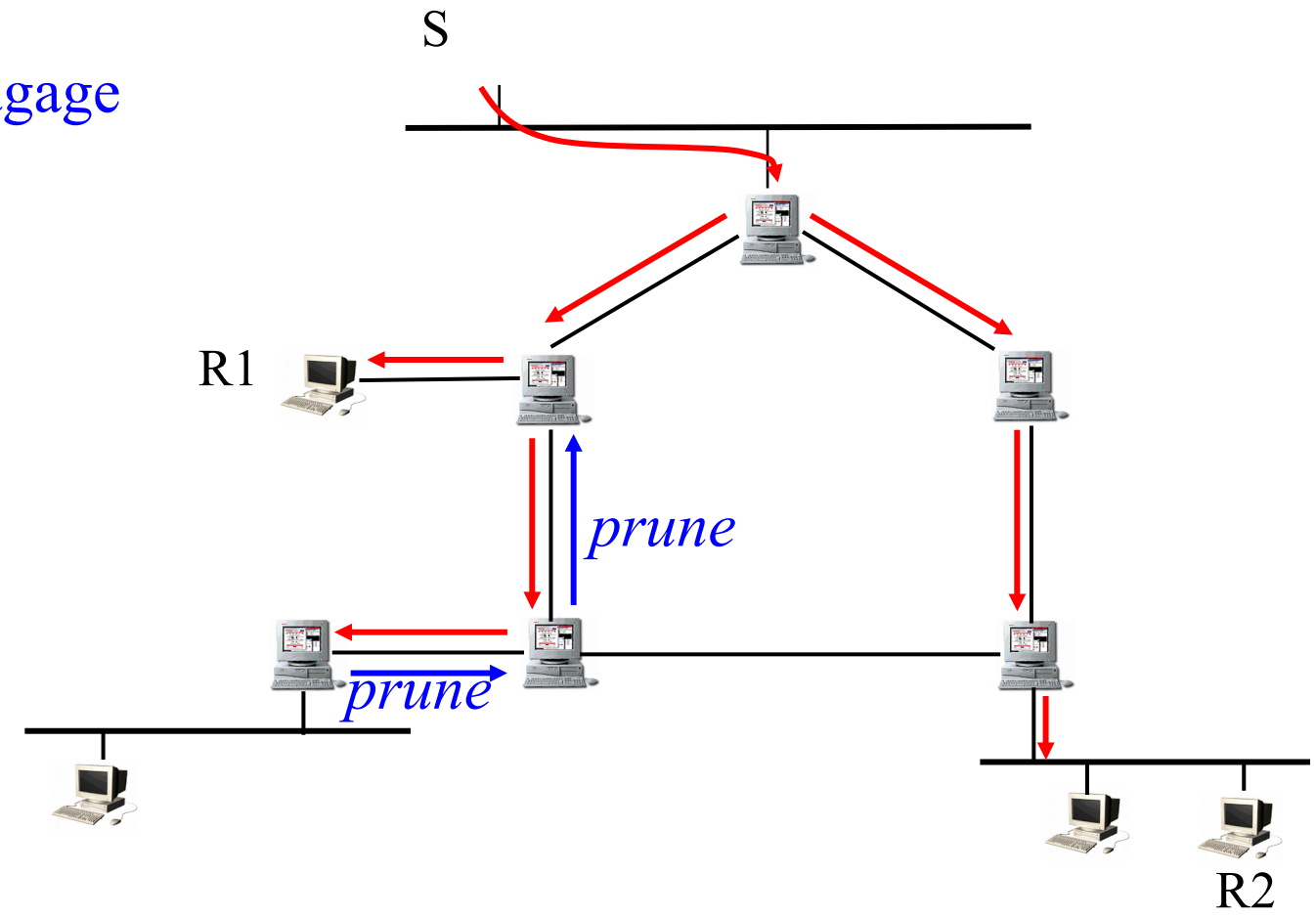
PIM-DM Protocol Independent Multicast, Dense mode

Inondation

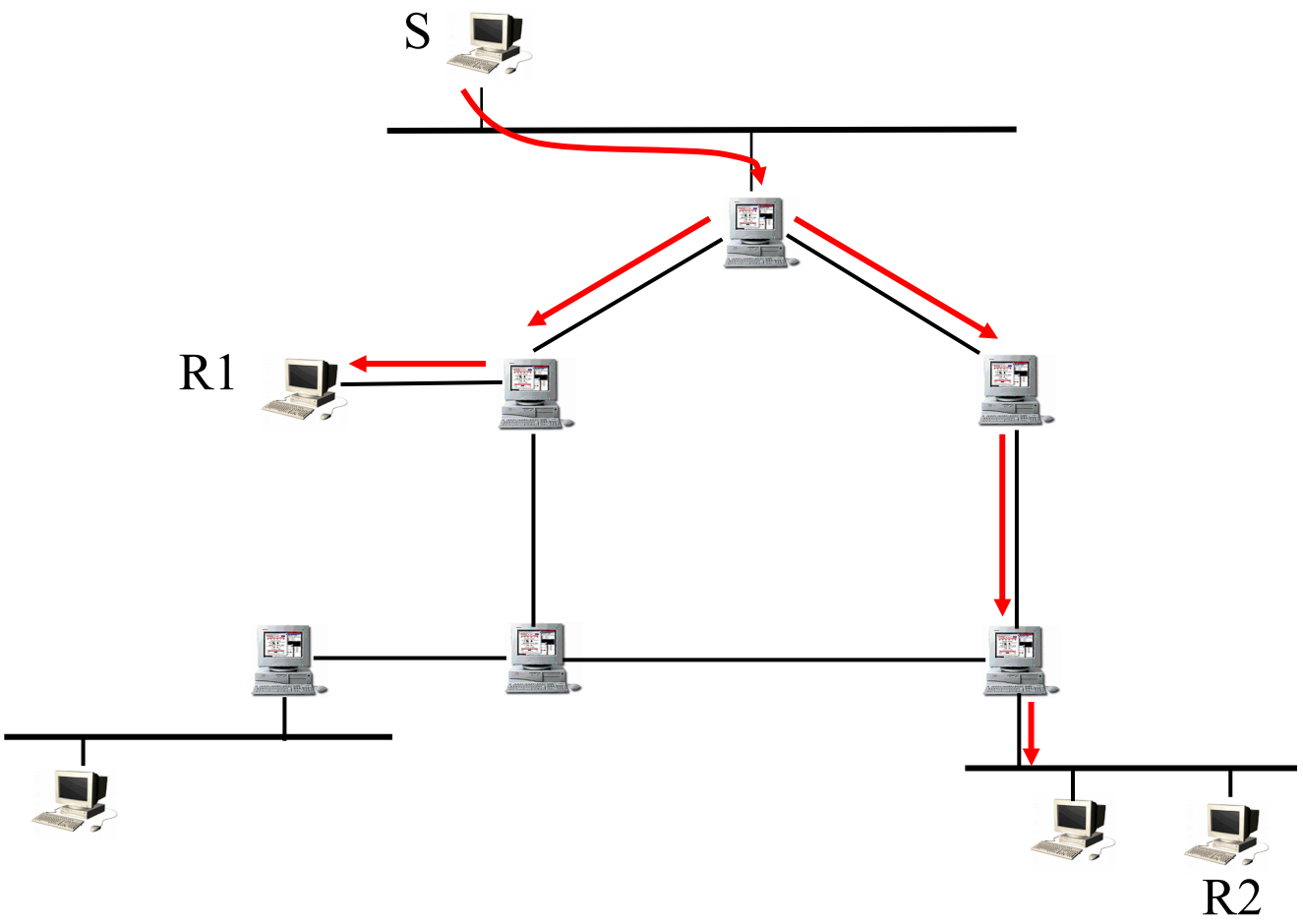


PIM-DM

Élagage

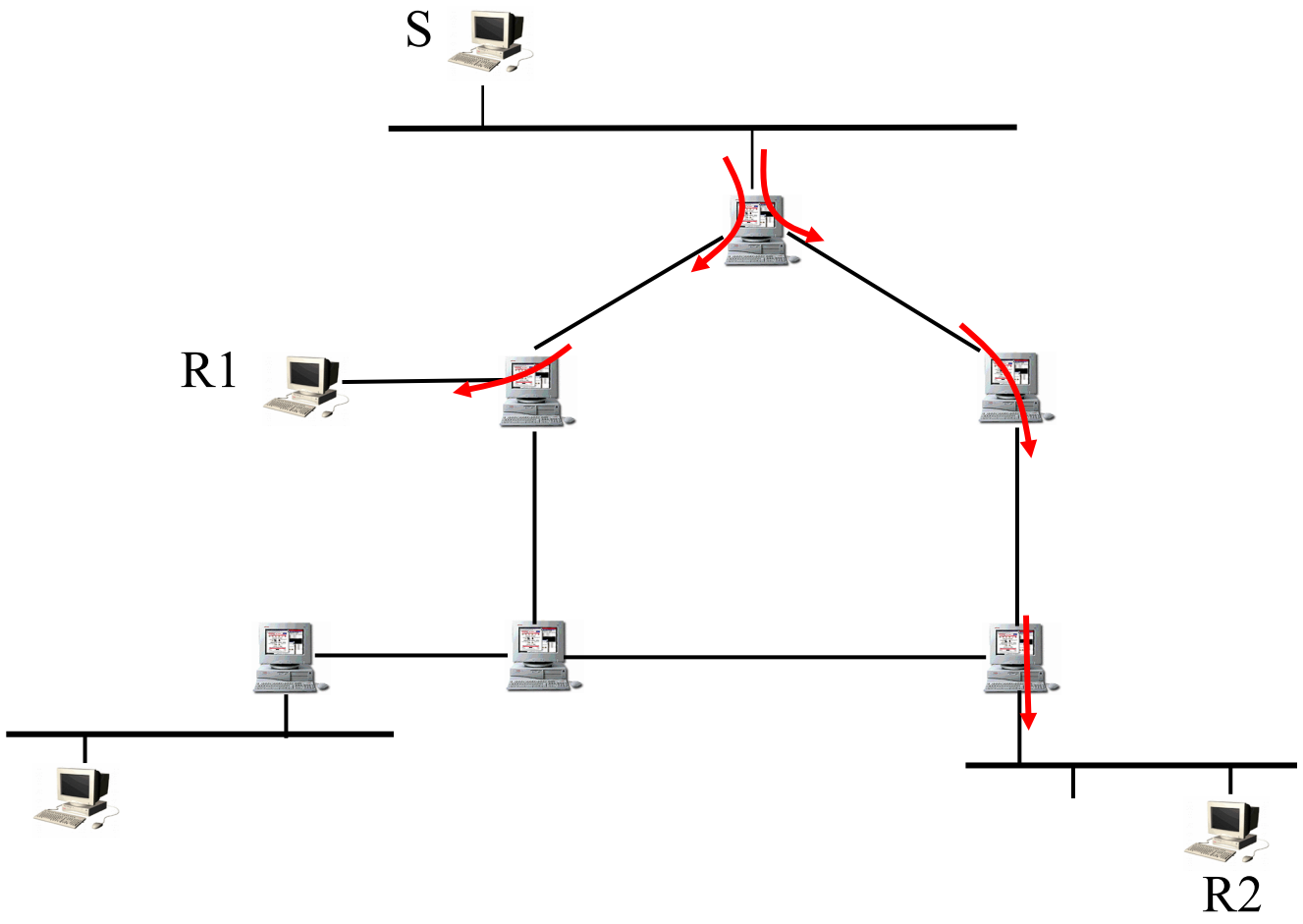


PIM-DM



PIM-DM

États des routeurs



MOSPF Multicast Open Shortest Path First

- Extension de OSPF
  - Chaque routeur connaît les liens du réseau (routage par état des liens)
- Nécessité de connaître l'existence des membres de chaque groupe sur chaque lien
- Calcul des arbres multicast à la volée (coûteux)
  - Chaque routeur calcule l'arbre de S vers tous les membres (Dijkstra)
  - Suis-je sur cet arbre et avec quel successeur?
  - Résultat mis en cache

MRIB: table de routage multicast

- Chaque nœud doit savoir quoi faire
- Nécessité de connaître l'existence des membres de chaque groupe sur chaque lien
- Ajout d'une route multicast :

ip mroute prefix mask @dest

Mode dense: conclusion

- Adapté aux petits domaines si la majorité des réseaux contiennent des membres
- Coûteux en signalisation (prune) si groupe épars
- Coûteux en états dans tous les routeurs du domaine

Protocoles en mode épars

- Un arbre partagé par toutes les sources
- Construit autour d'un point central (Core, RP)
- Signalisation explicite (*join*) des récepteurs vers le centre
- Arbre en général construit à l'envers «chemin inverse»
- Deux propositions majeures:
  CBT (SIGCOMM 93), PIM (SIGCOMM 94)

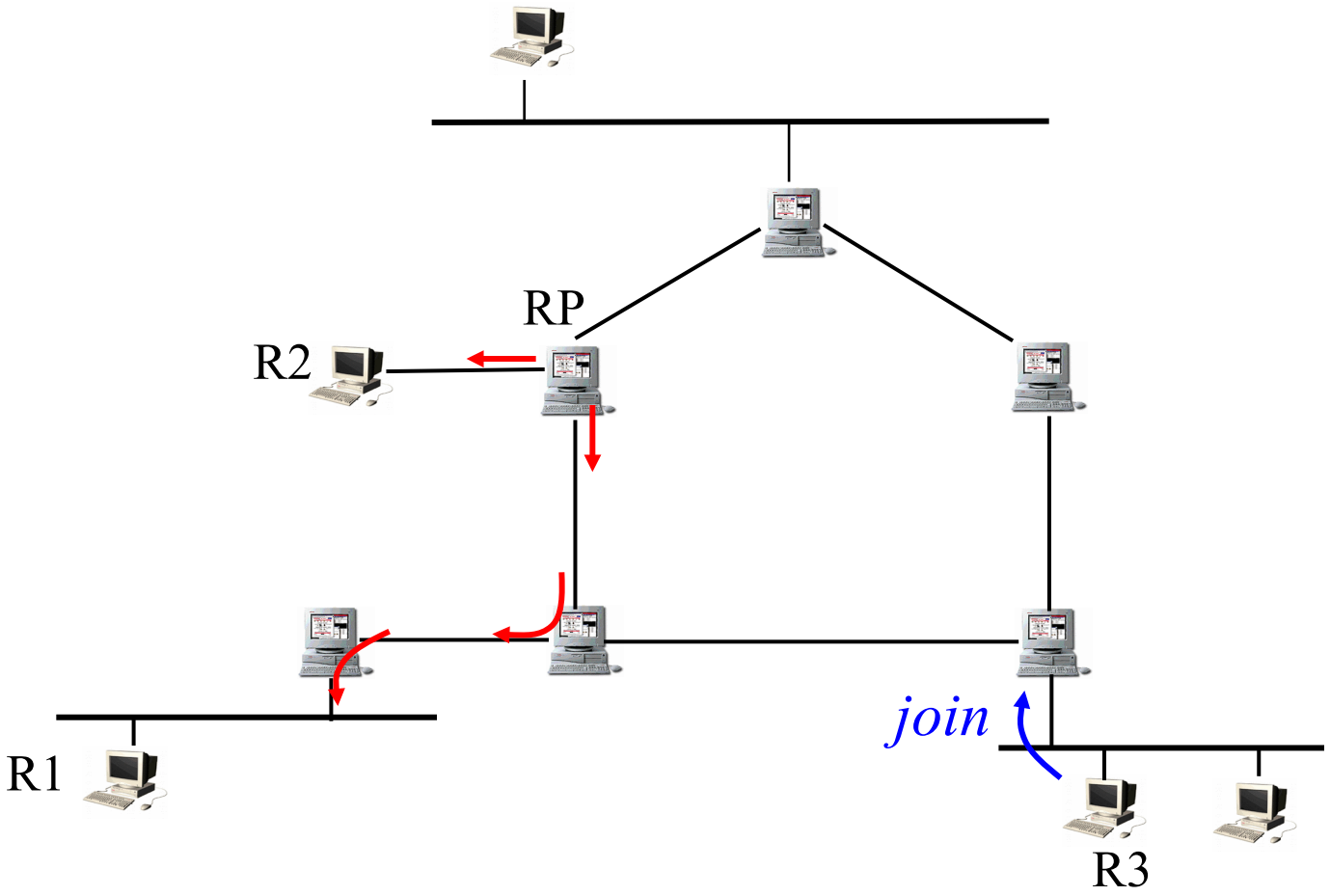
Construction de l'arbre PIM-SM

- Le routeur d'attachement du nouveau récepteur, à réception d'un *join* IGMP
  - Calcule l'adresse du RP
  - Transmet le *join* au routeur voisin vers le RP
- Le *join* remonte vers le RP jusqu'à rencontrer un routeur sur l'arbre

  => ajout d'une branche du récepteur vers l'arbre
- Acquittements dans l'autre sens
- Messages de retrait (prune) analogues

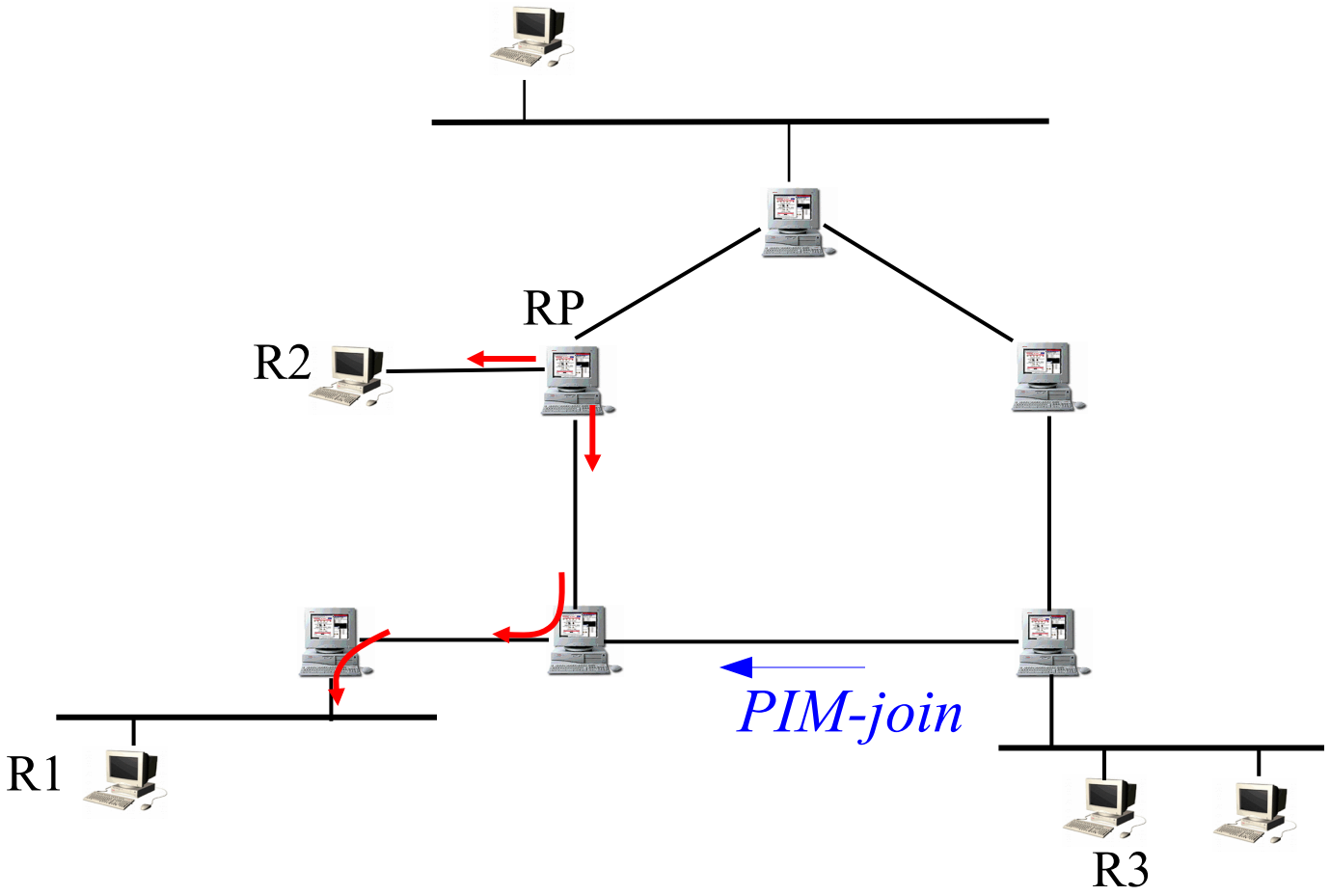
PIM-SM : exemple d'adhésion

Adhésion de R3



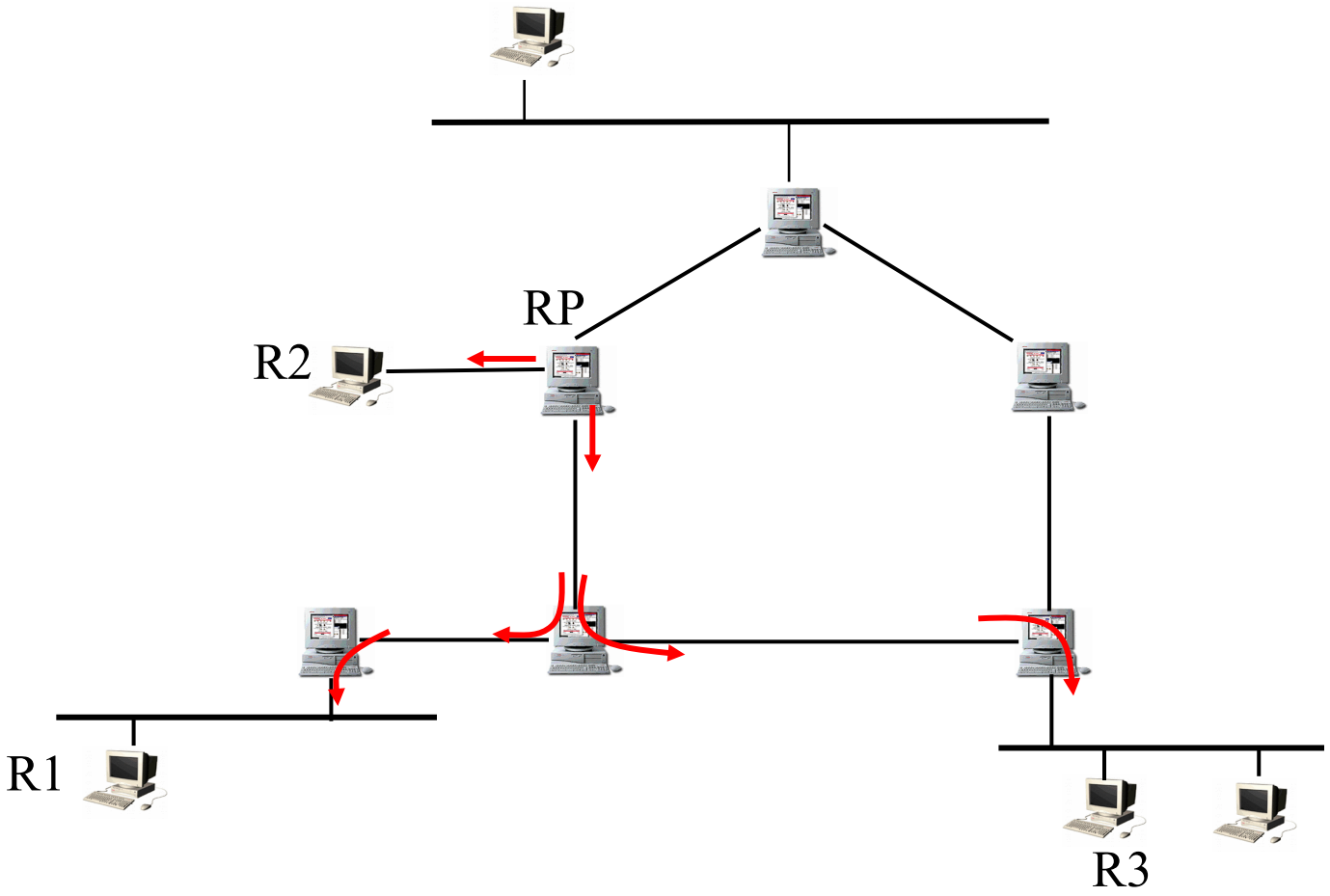
Exemple d'adhésion

Adhésion de R3



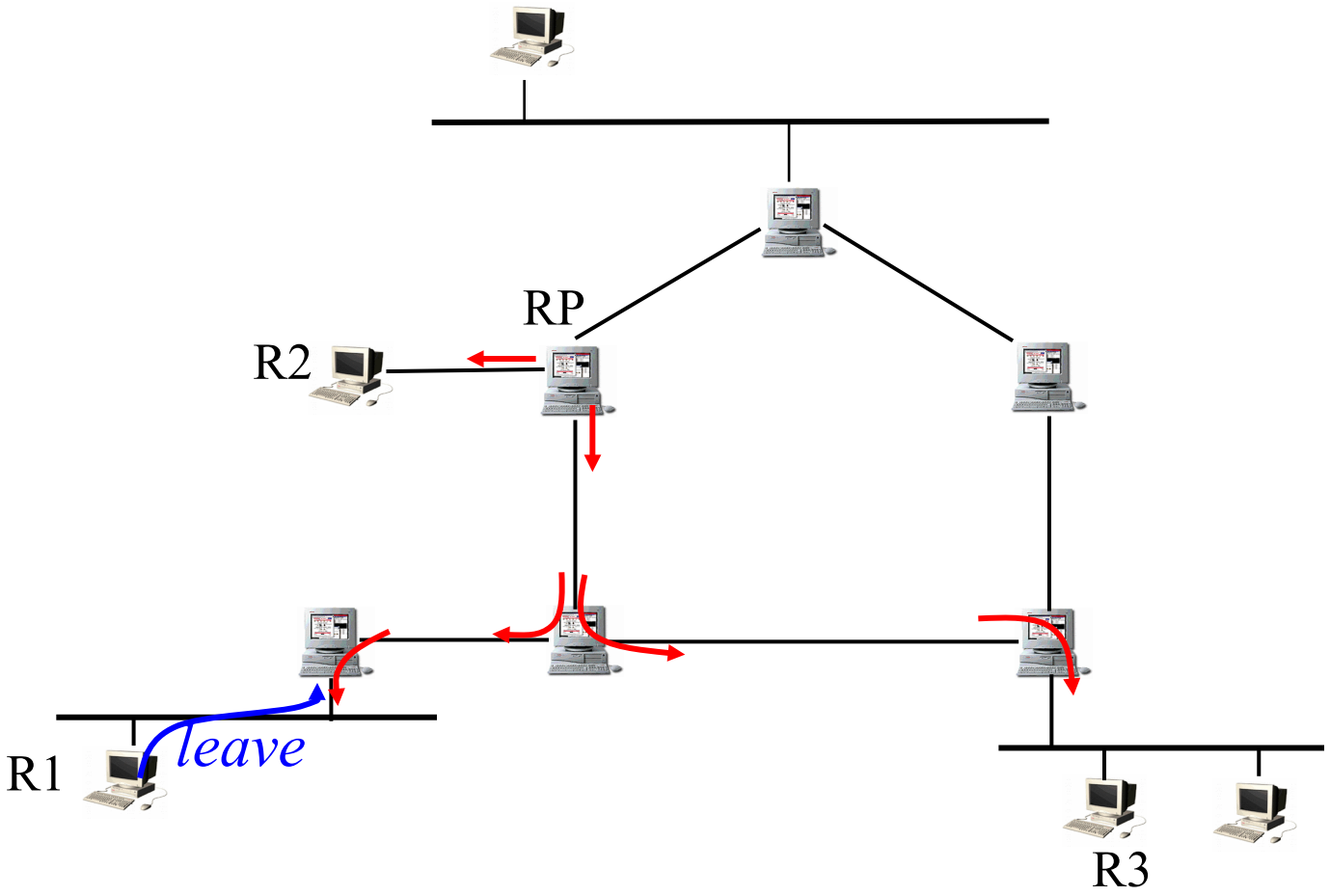
Exemple d’adhésion

Adhésion de R3



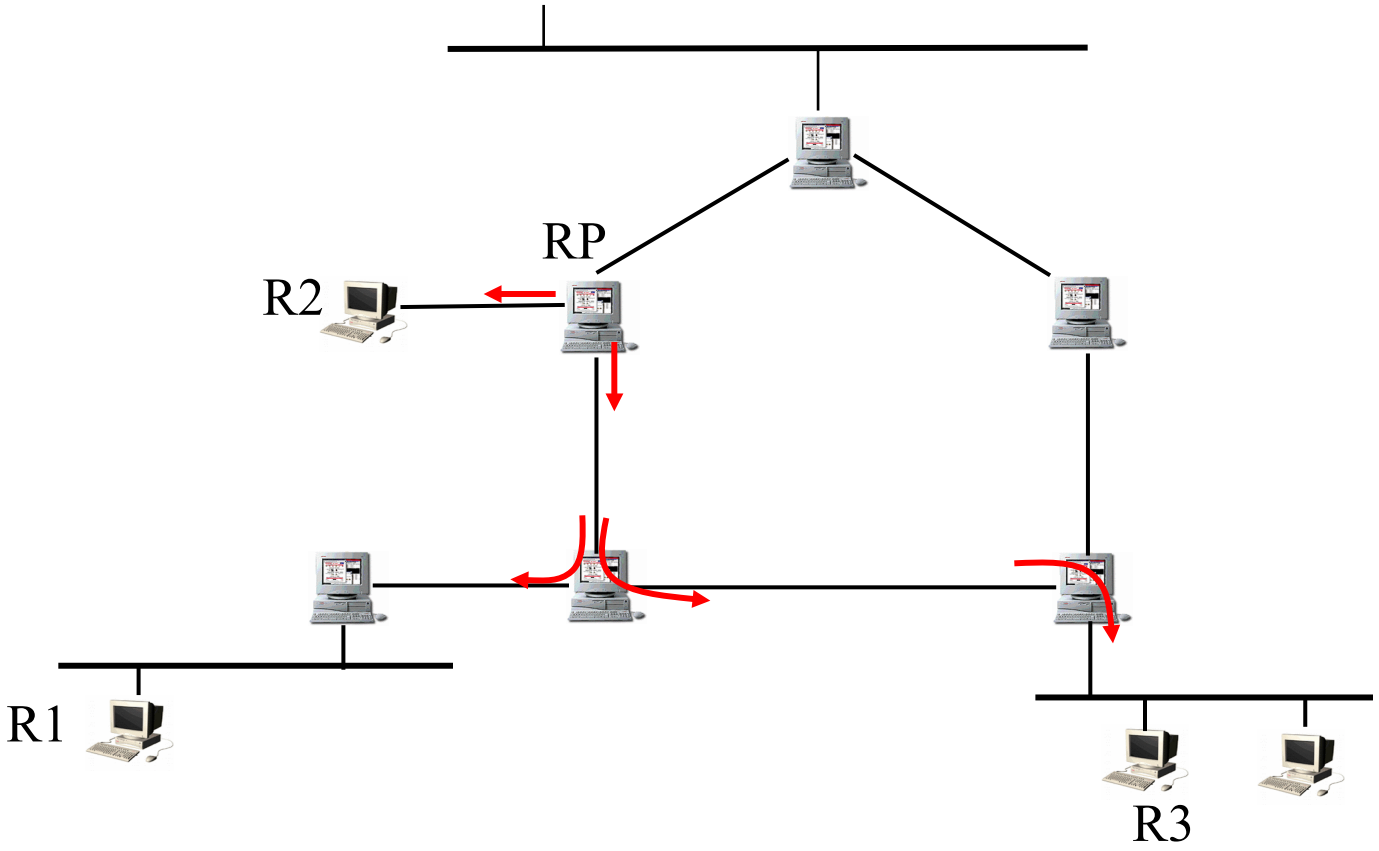
Exemple de retrait

Retrait de R1



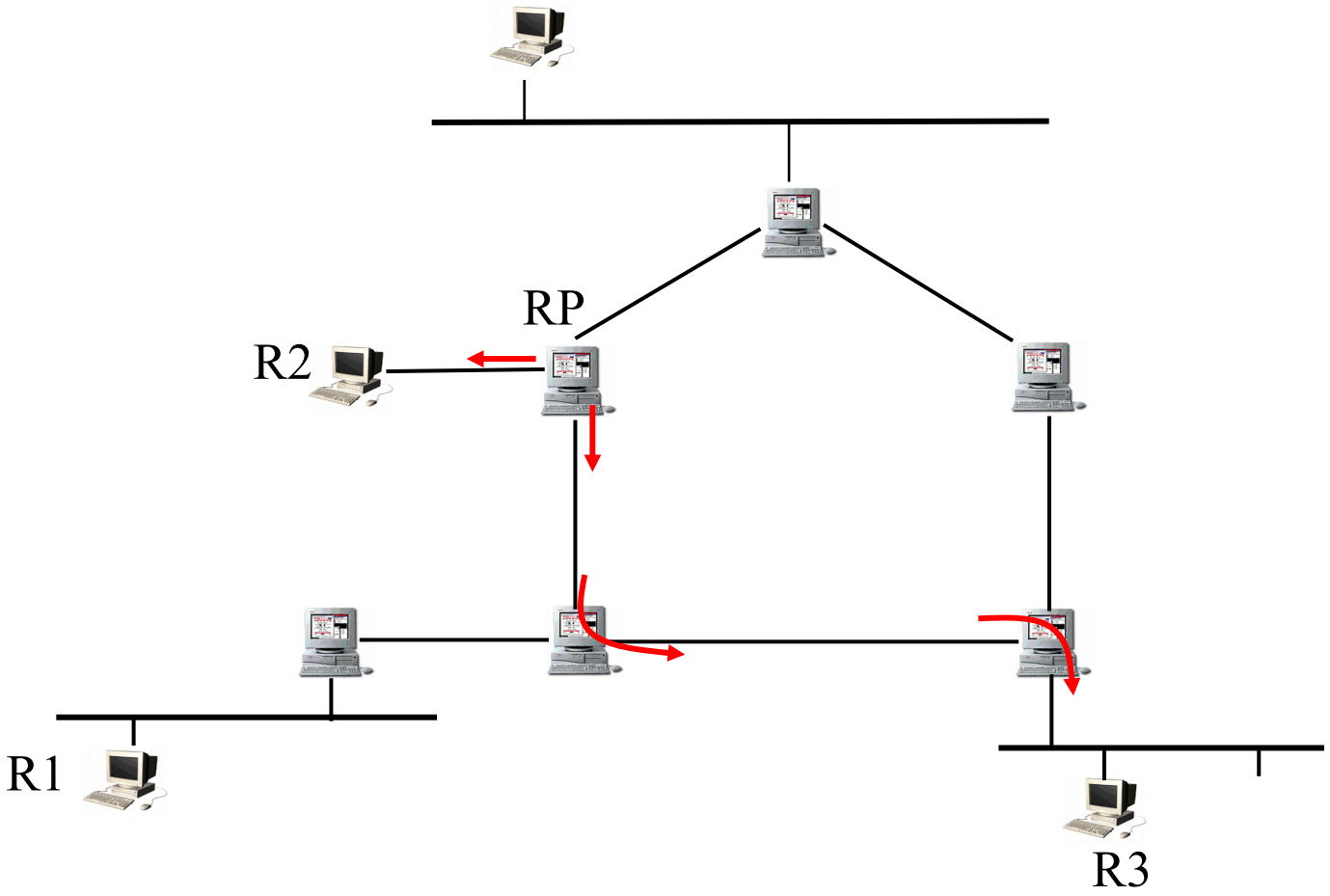
Exemple de retrait

Retrait de R1



Exemple de retrait

Retrait de R1

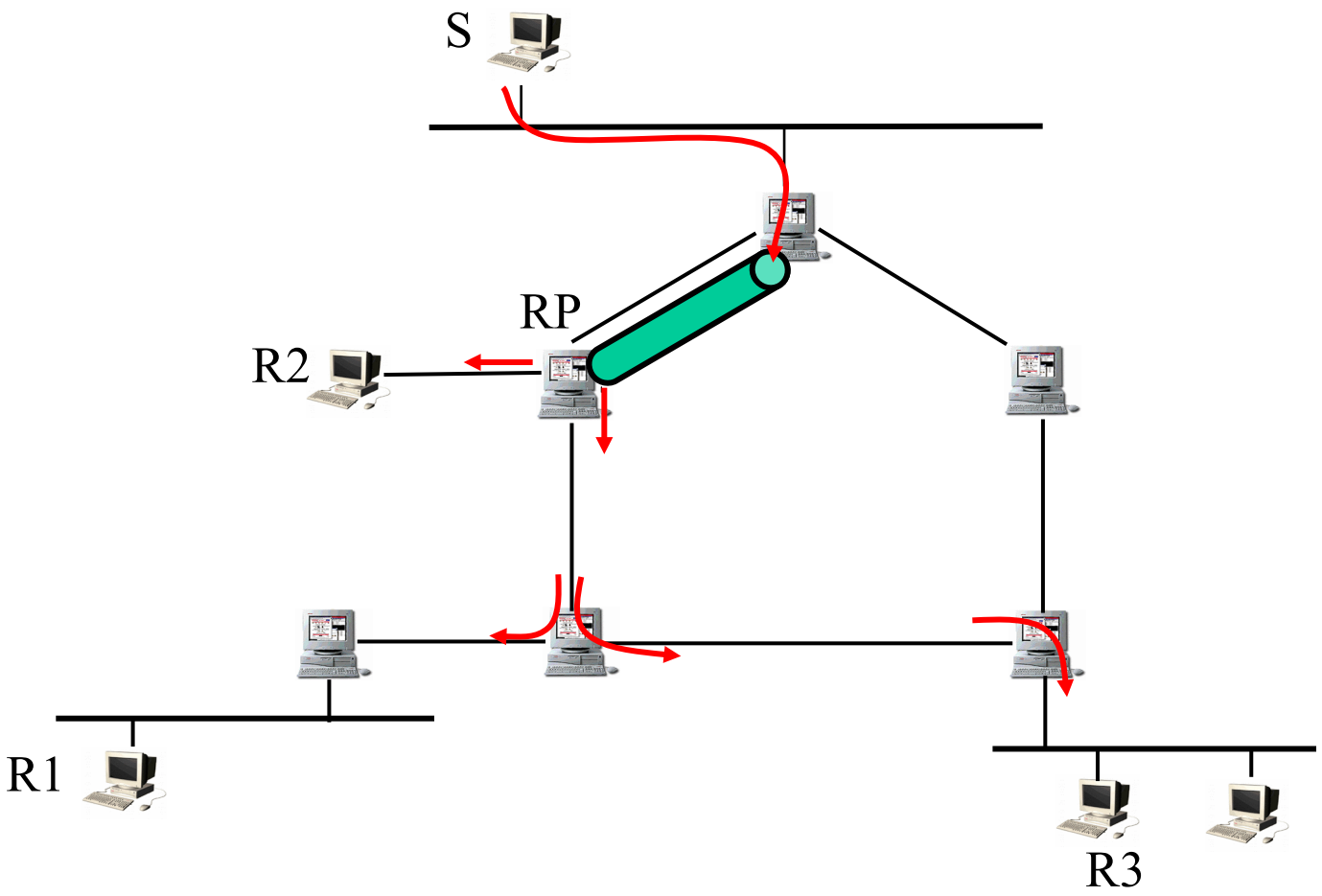


Émission de données PIM-SM

- La source envoie son paquet
- Le routeur d'attachement désigné
  - Détermine le RP associé
  - Encapsule le paquet et l'envoie au RP
- Le RP décapsule et achemine le paquet dans l'arbre (unidirectionnel)
- Possibilité de passer à un arbre par source (sur critères de débits)

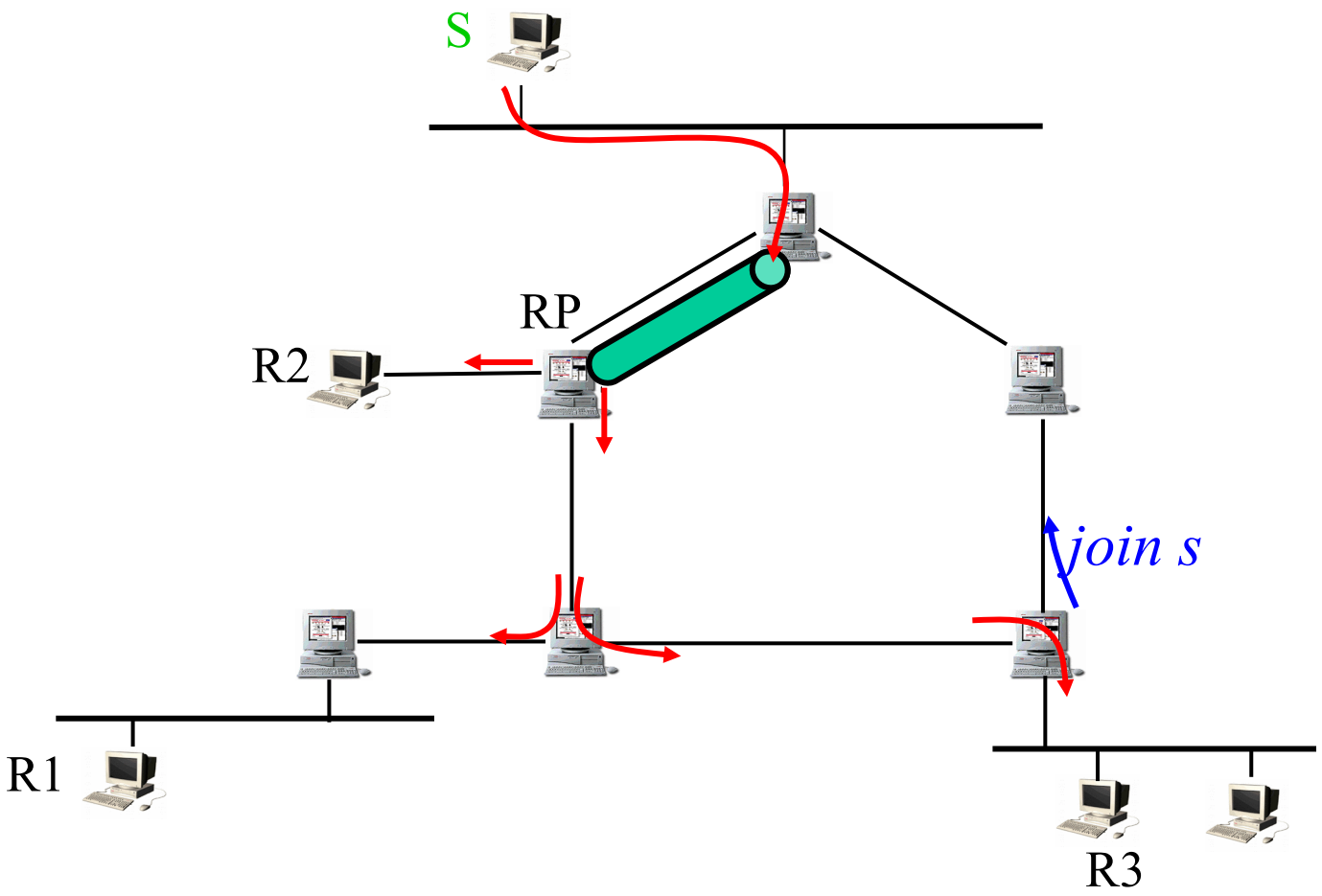
Exemple d'émission

S émet



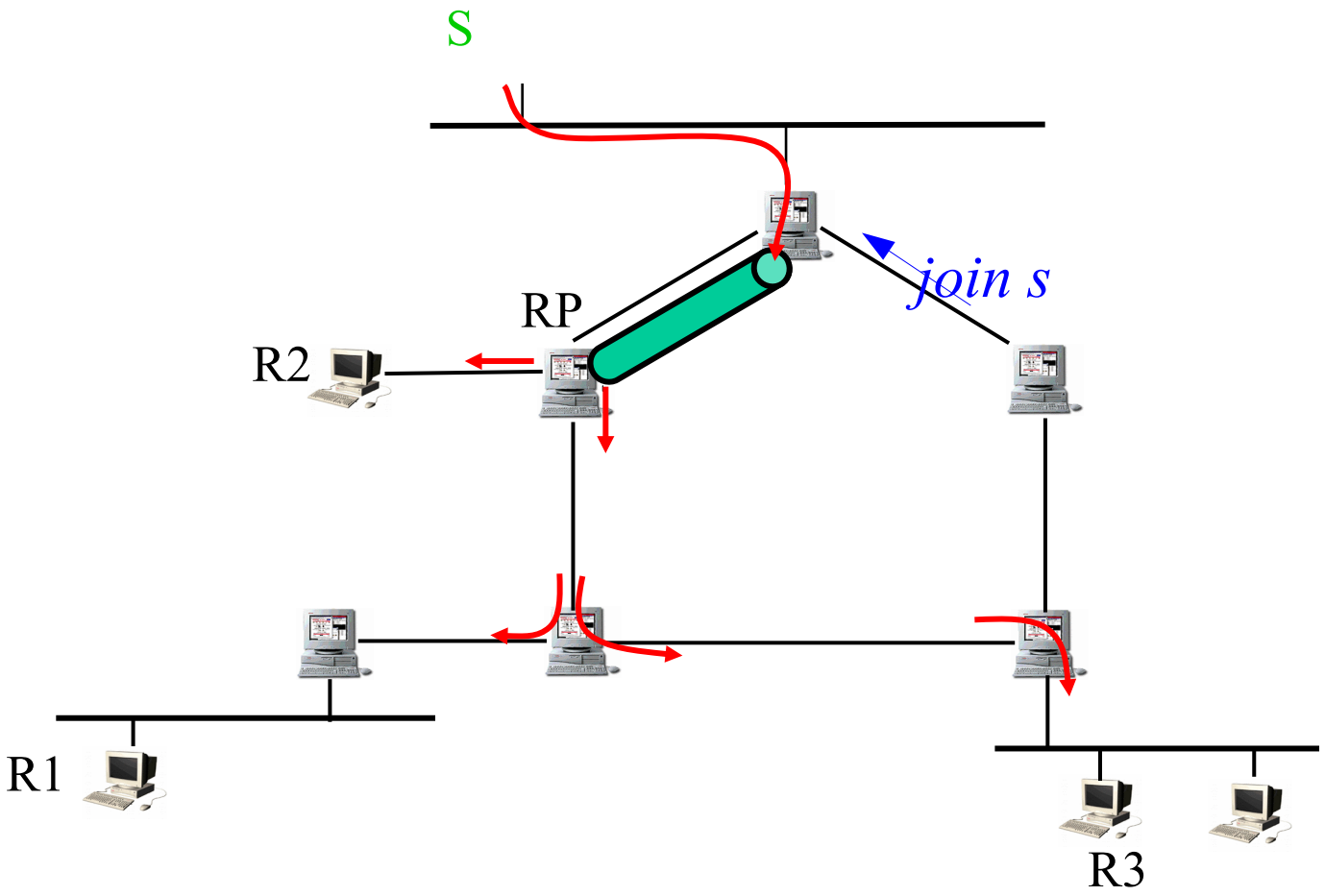
Passage arbre par source

R3 passe à un arbre par source : msg PIM-join s transmis entre routeurs



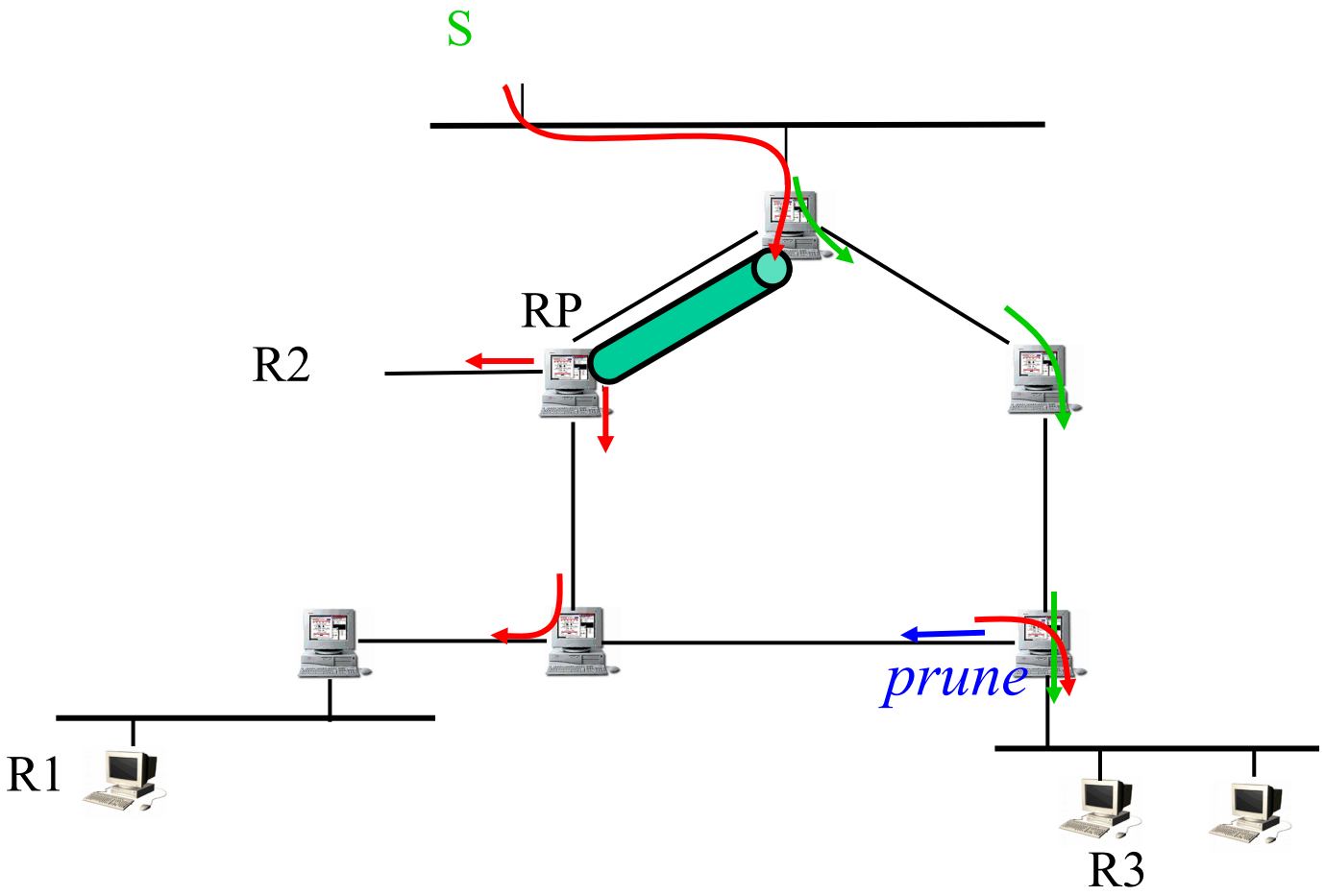
Passage arbre par source

R3 passe à un arbre par source



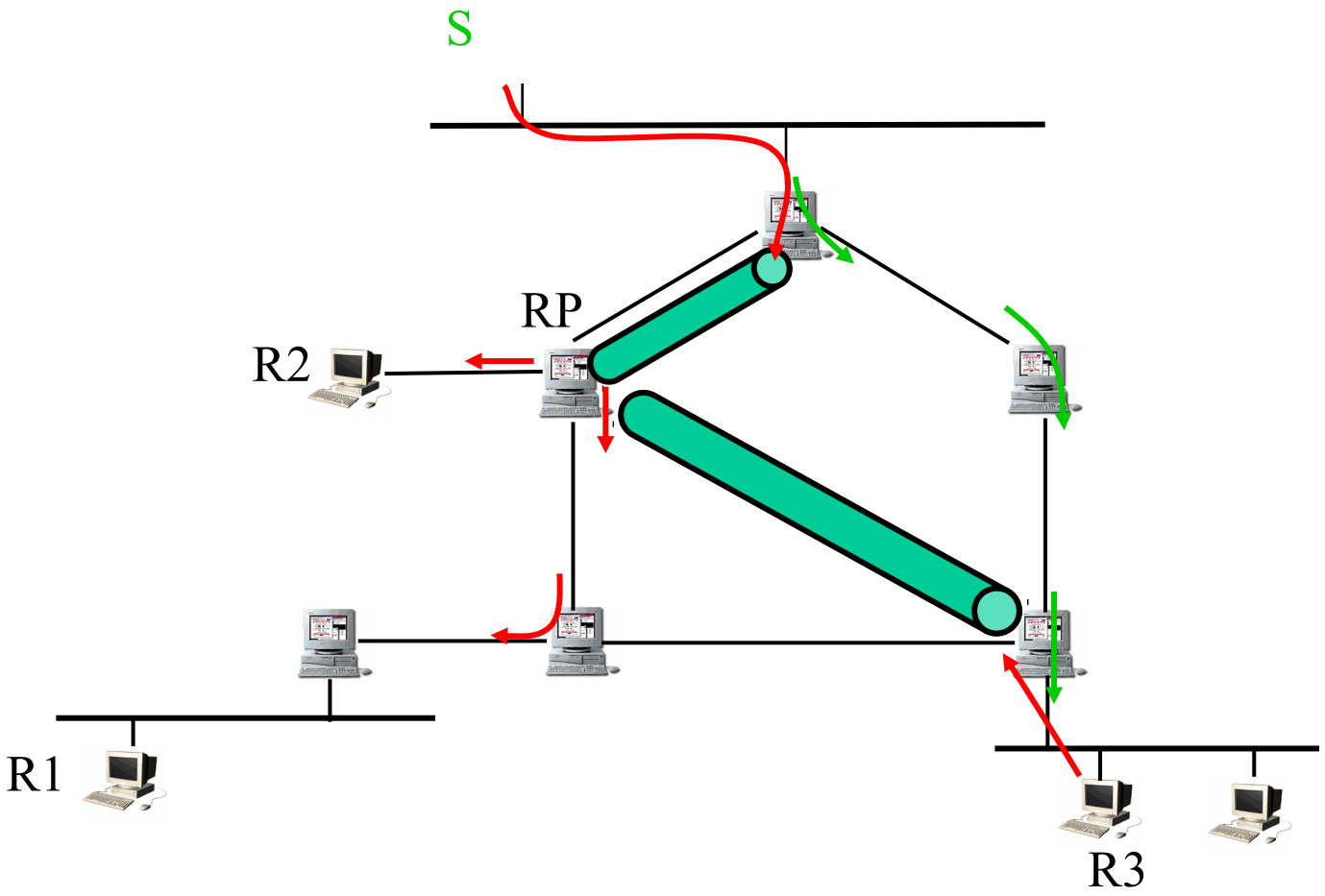
Passage arbre par source

R3 passe à un arbre par source : msg PIM-prune vers RP



Passage arbre par source

R3 émet des données



CBT Core Based Tree

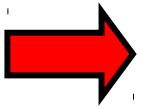
Assez semblable à PIM-SM mais

- Arbre bidirectionnel
- Données encapsulées vers le centre (core) dans le cas de sources non membres du groupe

Choix du centre (PIM et CBT)

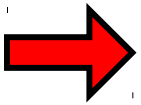
- Algorithme du bootstrap
  - Bootstrap router (BSR) élu parmi les candidats BSR
  - Le BSR diffuse une liste de points de rendez-vous (RP) valides (parmi les candidats RP)
  - Une fonction de hachage associe une adresse de groupe à un RP parmi la liste

Tout routeur peut calculer le RP associé à un groupe (une adresse)



Commentaires

- Le mécanisme de sélection du Core/RP n'est pas scalable:
  - La liste de tous les RP doit être connue de tous les routeurs (diffusion de cette liste)
  - Placement non optimal dans un grand réseau
  - Panne du Core/RP=reconstruction de tout l'arbre



Limités à l'intra-domaine

Déploiement actuel

- Implantations de DVMRP, MOSPF, PIM, CBT…
- Les opérateurs ne routent (en général) pas le multicast (pas de solution viable à grande échelle)
- Réseau expérimental Mbone (Fmbone)
  - Tunnels entre routeurs multicast à travers des routeurs non multicast
  - Allocation d'adresse manuelle

Problèmes de l'Inter-domaine (1/2)

- Extensibilité
  - En nombre de groupes
    - Potentiels, existants, existants à un endroit
  - Taille des groupes
  - États à mémoriser, signalisation
- Allocation des adresses
  - Unicité à l'échelle d'Internet
  - Dynamicité

Problèmes de l'Inter-domaine (2/2)

- Interactions protocoles de routage intra-inter
- Respect des politiques de routage
  - Contrôle du flux multicast en transit
- Fiabilité du routage
  - Liée à la taille
  - Importance si applications commerciales

Passage à l'inter-domaine

Problème du centre:

- Hiérarchisation (HPIM, HDVMRP, OCBT, HIP, BGMP)
- Adressage du centre (SIMPLE, EXPRESS)
  - Adr du groupe = adresse du centre + #port

BGMP Border Gateway Multicast Protocol

Utilisé en inter-domaine

- Un proto intra-domaine (MIGP) dans chaque domaine (PIM, CBT, MOSPF,…)
- L'arbre BGMP relie les domaines
  - Arbre enraciné dans le domaine possédant l'adresse du groupe (alloué par MASC et publié par BGP)
  - Bi-directionnel
- Fait transiter données et *join* d'un domaine à un autre

BGMP : Construction de l'arbre

- *join* du membre vers un routeur multicast frontière(BR) du domaine via protocole intra-domaine (nécessite que les BR appartiennent à tous les groupes: modification des protocoles épars)
- Le BR
  - Détermine le root domain associé à l'adresse multicast
  - Propage le *join*
    - Soit vers le routage intra du domaine suivant
    - Soit vers un voisin BGMP externe

Problèmes

- Complexité
- Scalabilité (annonce des adresse multicast)
- Vulnérabilité du root domaine
- Pas de contrôle d’accès
- Routes inverses
- Modifications des protocoles intra-domaines

Simple Multicast (crowcroft, Perlman et al.)

Groupe identifié par un couple (@core, ID)

- Plus besoin de déterminer le core
- Pas de problème d'allocation d'adresse multicast
- Arbre bidirectionnel à la CBT mais inter-domaine
- Nécessite un niveau d'adressage en plus
  - Nouveau protocole au dessus d'IP

Simple Multicast : problèmes

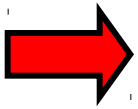
- Dépendance à la racine (fiabilité)
- Chemins inverses
- Contrôle
- Overhead
- Changement des interfaces hôtes

EXPRESS (Holbrook et Cheriton)

- Notion de canal (channel) unidirectionnel
  - 1 source vers N récepteurs
  - Adressé par couple (@source,ID)
  - Pas de problème d'allocation des adresses
- Prévu pour des applications de type canal télé
- Join envoyé du récepteur vers la source
  - Pas de problème de localisation du centre

EXPRESS (suite)

- Contrôle d'accès
  - Possibilité de clé d'authentification
    - Par canal
    - Stockée dans les routeurs de l'arbre du canal
- Arbre unidirectionnel
  - Facilite l'accounting (la source paye)



Protocole assez simple mais pas extensible aux grand groupes, vulnérable à la panne de la source

Autres propositions

- YAM (Yet another Multicast Protocol)
  - Arbre partagé, QoS
  - Adhésion par connection au nœud de l'arbre le plus proche
- Reunite
  - Pas d'adresse multicast
  - Arbre composé récursivement de branches unicast
- Multicast explicite
  - Transporter liste des récepteurs
  - Petits groupes

Conclusions

- Problèmes avec modèle de Deering:
  - L'existence d'un groupe n'est pas définie
  - Les émetteurs ne sont pas membre
- Choix et modification dynamique du centre de l'arbre
- Le multicast doit être présent en standard dans IPv6, mais quel(s) protocole(s)?
  - PIM-SM est l'un des plus complet, mais trop complexe? => PIM-SSM

PIM-SSM (Source Specific Multicast)

- Un arbre par source => pour applications de un vers plusieurs
- Sous-ensemble de PIM-SSM (pas de RP)
- Utilise une plage d'adresses réservée:

  232.0.0.0/8 en v4

  FF3X::/96 en v6
- Dépend totalement de IGMPv3 en IPv4 et MLDv2 en IPv6

MLDv2

- Messages de requête (query, type=130, émis par routeur) :
  - Général : connaître tous les groupes → FF02::1
  - Pour une adr multicast → @groupe
  - Pour une adr mcast et une source → @groupe

- Message de rapport (join, type=143, envoyés à FF02:16) :
  - Contient une liste d'enregistrements d'adresses
  - Chaque EA a un champs type d'enregistrement (réponse query, chgt mode filtrage, chgt liste sources) et une liste de champs d'adresses (@mcast, @src1,…, @srck)

Les applications multicast usuelles

- Audioconférence
  - vat (Visual Audio Tool) : audio-conférence
  - rat (Robust Audio Tool) : audio-conférence
- Vidéoconférence
  - ivs (Inria Videoconferencing System) : vidéo/audio-conférence
  - nv (Network Video) : vidéo-conférence
  - vic (Video Internet Conferencing) : vidéo-conférence
  - telesia : vidéo/audio-conférence

Les applications multicast usuelles

- Tableau blanc
  - wb (White-Board) : tableau blanc partagé
  - wbimport : contrôle du tableau blanc par un modérateur qui pilote les pages
- Catalogues des sessions
  - sd (Session Directory) : annonce les conférences en cours ou à venir
  - lancement automatique des applications
  - sdr (Session Directory) : annonceur nouvelle génération utilisant le transport RTPv2

API IP Multicast

- Multicast uniquement possible en UDP (pourquoi?)
- Des API propres à chaque langage proposent chacune des primitives pour la gestion du groupe (adhésion/retrait)
  - En C
    - RFC 3678 Socket Interface Extensions for Multicast Source Filters)
  - En Java (java.net)

Initialisation des sockets en C

```
#define GROUP "239.137.194.222"

#define PORT 55501

sdr = socket(PF_INET, SOCK_DGRAM, 0);

memset(&sock_r, 0, sizeof(sock_r));

    sock_r.sin_family = AF_INET;

    sock_r.sin_port = htons(PORT);

    sock_r.sin_addr.s_addr = htonl(INADDR_ANY);

sdw = socket(PF_INET, SOCK_DGRAM, 0);

memset(&sock_w, 0, sizeof(sock_w));

    sock_w.sin_family = AF_INET;

    sock_w.sin_port = htons(PORT);

    sock_w.sin_addr.s_addr = inet_addr(GROUP);
```

Réception d'un paquet multicast

Joindre le groupe:

```
imr.imr_multiaddr.s_addr = inet_addr(GROUP);

imr.imr_interface.s_addr = htonl(INADDR_ANY);

setsockopt(sdr, IPPROTO_IP, IP_ADD_MEMBERSHIP, (void *) &imr, sizeof(struct ip_mreq)
```

Associer la socket au descripteur:

```
bind(sdr, (struct sockaddr *)&sock_r, sizeof(sock_r)
```

Recevoir les données:

```
while (1) { cnt = recvfrom(sdr, buf, sizeof(buf), 0, (struct sockaddr *)&sock_r, &len_r);}
```

Emission de paquets multicast

Par défaut, paquets multicast sont envoyés avec TTL = 1 => portée = le lien

Modifier le TTL:

```
unsigned char ttl = 5;

setsockopt(sdw, IPPROTO_IP, IP_MULTICAST_TTL, &ttl, sizeof(ttl)
```

Emission de données:

```
cnt = sendto(sdw, buf, strlen(buf), 0, (struct sockaddr *)&sock_w, len_w)
```

Quitter le groupe:

```
setsockopt(sdr, IPPROTO_IP, IP_DROP_MEMBERSHIP, (void *) &imr, sizeof(struct
    ip_mreq)
```

API multicast JAVA

Joindre un groupe:

```
int group, port;

InetAddress group InetAddress.getByName(group);

MulticastSocket s = new MulticastSocket(port);

s.joinGroup(group);
```

Emission de datagrame:

```
byte[] msg = "Hello World!";

DatagramPacket hi = new DatagramPacket(msg, msg.length, group, port);

s.send(hi);
```

Fixer le TTL: s.setTTL(ttl);

API multicast JAVA

Réception de datagramme:

```
byte[] buf = new byte[512];

DatagramPacket recv = new DatagramPacket(buf, buf.length);

s.receive(recv);
```

Quitter un groupe:

```
s.leaveGroup(group);
```

Protocoles pour le multimédia

- Applications interactives en temps-réel
  - RTP (Real Time Protocol, RFC 3550)
  - RTCP (Real Time Control Protocol, RFC 3550)
  - SCTP (Stream Control Transfert Protocol, RFC 4960)
- Streaming
  - RTSP (Real Time Streaming Protocol, RFC 2326)
- Voix sur IP
  - SIP (Session Initiation Protocol, RFC 3261)
  - H.323 (ITU-T), utilise RTP
- QdS
  - RSVP (Ressource Reservation Protocol, RFC 2205)

RTP/RTCP

- RTP (Realtime Transport Protocol) et son compagnon RTCP (Realtime Transport Control Protocol) permettent respectivement de transporter et de contrôler des flots de données qui ont des propriétés temps-réel.
- Situés au niveau application, utilisent TCP ou UDP (plus souvent UDP) et les modes unicast ou multicast
- Utilisent une paire de ports: RTP utilise le port pair et RTCP le port impair immédiatement supérieur.

RTP (Real Time Protocol)

Gestion du temps réel et de la session multipoint

- 2 Intermédiaires: les mixeurs (regrouper plusieurs flots de plusieurs applications en un seul flot conservant le même format) et les translateurs (changer le format du codage, ex: de MPEG vers H.261)
- La gestion du « temps réel » est effectuée par RTCP (car les paquets RTP ne transportent que les données des utilisateurs, pas les informations de gestion)

RTP

Il permet de:

- identifier le type de l'information transportée pour compenser en cas de perte par exemple
- reconstituer la base de temps des différents flux multimédia (audio, vidéo...): synchronisation effectuée par des labels
- séquencer les paquets par ajout d'un numéro => détection des pertes de paquets

Il ne permet pas de:

- réserver des ressources dans le réseau;
- apporter une fiabilité dans le réseau;
- garantir le délai de livraison;

RTP: format du paquet

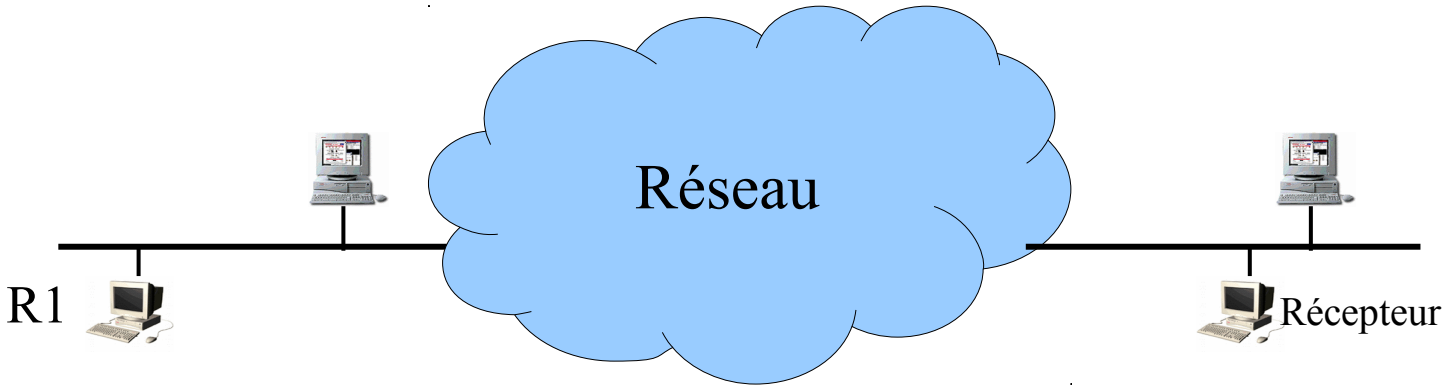
Sur 12 octets + liste d'identificateurs optionnelle. Dans l'ordre :

- Version (2 bits), indique la version du protocole (V=2)
- Padding (1 bit), si P=1, le paquet contient des octets additionnels de bourrage (padding) pour finir le dernier paquet.
- Extension (1 bit), si X=1 l'en-tête est suivie d'un paquet d'extension
- CSRC count (4 bits), contient le nombre de CSRC qui suivent l'entête
- Marker (1 bit), son interprétation est définie par un profil d'application (profile)
- Payload type (7 bits), ce champ identifie le type des données (audio, vidéo, image, texte, html, etc.),
  ex : type 26 = jpeg, horloge = 90000 Hz
  voir http://www.iana.org/assignments/rtp-parameters

RTP : timestamp et sequence

Ex pour Vidéo

- Horloge du timestamp : 90000Hz
- Un paquet RTP contient au plus une frame, transporté par un paquet UDP débit paquets RTP : 30Hz (parfois 25)
- Incrémentation du timestamp : 1/30/1/90000 = 9000/30 = 3000
- Sequence number incrémenté de 1 à chaque paquet



RTP: format du paquet (fin)

- Sequence number (16 bits), sa valeur initiale est aléatoire et il s'incrémente de 1 à chaque paquet envoyé, il peut servir à détecter des paquets perdus
- Timestamp (32 bits), reflète l'instant où le premier octet du paquet RTP à été échantillonné. Cet instant doit être dérivé d'une horloge qui augmente de façon monotone et linéaire dans le temps pour permettre la synchronisation et le calcul de la gigue à la destination
- SSRC (32 bits), identifie de manière unique la source (valeur aléatoire choisie par l'application).
- CSRC (32 bits), identifie les sources contribuantes (quand mixées).

RTCP

- Protocole de contrôle associé à RTP, mesure les performances mais pas de garantie.
- C'est un "feedback" pour l'émetteur sur la qualité de transmission et d'autres informations.
- Basé sur la transmission périodique de paquets de contrôle à tous les participants dans une session.
- Utilise le même mécanisme de distribution que les paquets de données.
- Si besoin de QoS, il faut employer un protocole de réservation du type RSVP ou bien s'assurer que les liens de communications utilisés sont correctement dimensionnés.

RTCP: quatre fonctions

- Fournir des informations sur la qualité de la session: information en retour pour une source (feedback), permet à une source de changer de politique et met en évidence des défauts de distribution individuels, collectifs
- Garder une trace de tous les participants à une session

  CNAME (Canonical Name) : identifiant unique et permanent pour un participant; SSRC (Synchronisation Source Identifier)
- Contrôler le débit auquel les participants à une session RTP transmettent leurs paquets RTCP. Le trafic RTCP doit représenter moins de 5% du trafic de la session.
- Transmettre des informations de contrôle sur la session (optionnel) exemple : identifier un participant sur les écrans des participants

RTCP

5 types de paquets RTCP pour transporter des informations de contrôle :

- SR : Sender Report, transmission de statistiques des participants actifs en émission
- RR : Receiver Report, transmission de statistiques des participants passifs
- SDES : Source Description items (CNAME, NAME, EMAIL, PHONE,...)
- BYE : Fin de participation
- APP : fonctions spécifiques à l'application

Implantations de RTP/RTCP

- Dans la couche Application ou comme sous-couche de la couche Transport
- Linux:
  - librtp du projet Gphone,
  - Vovida Networks
  - VLC media player: lecteur/serveur multimédia
- Java: Classes RTP dans Java™ Media Framework

WebRTC

- De W3C et IETF (RFC 7478, 7742, 7874, 9999)
- communication (interactive et multimedia) pour le Web
- API Javascript et canevas logiciel open source
- Intégré à la plupart des navigateurs
- Basé sur DTLS/SRTP, STUN, ICE