

# Réseaux

## Routage

Tuyêt Trâm DANG NGOC  
<dntt@u-cergy.fr>

Université de Cergy-Pontoise

2009–2010



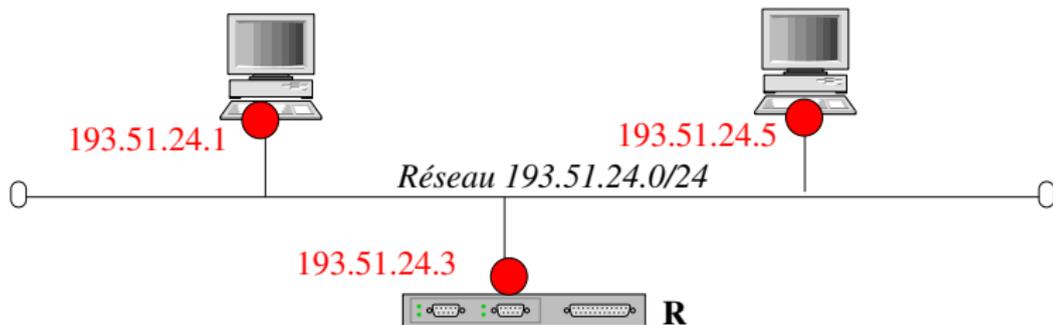
- 1 Routage statique
  - Table de routage
  - Routage par défaut
  - CIDR
- 2 Principes généraux du routage dynamique
  - Protocoles de routage
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF
  - Aire de routage
  - Messages OSPF
  - Protocole OSPF
  - Algorithme Shortest Path First (SPF)
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

- 1 Routage statique
  - Table de routage
  - Routage par défaut
  - CIDR
- 2 Principes généraux du routage dynamique
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

# Adressage IP et interface

Une adresse IP est associée à une interface.

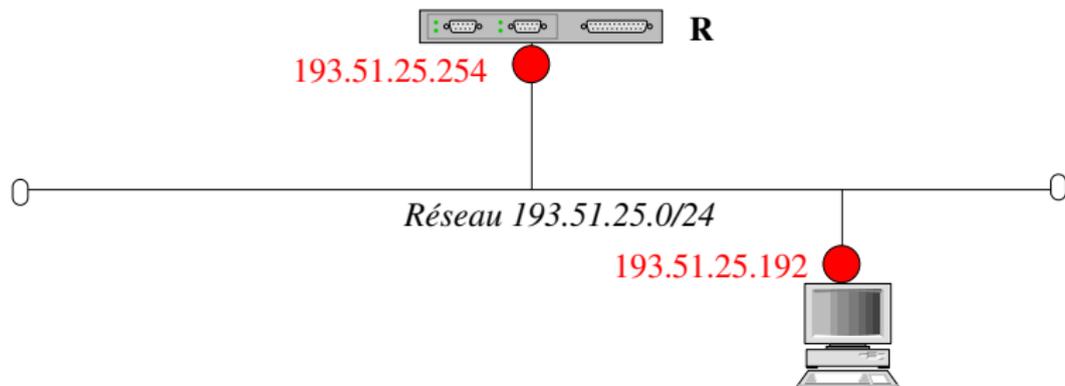
Exemple : le routeur R a deux interfaces, il a donc deux adresses :  
193.51.25.254 et 193.51.24.3



# Adressage IP et interface

Une adresse IP est associée à une interface.

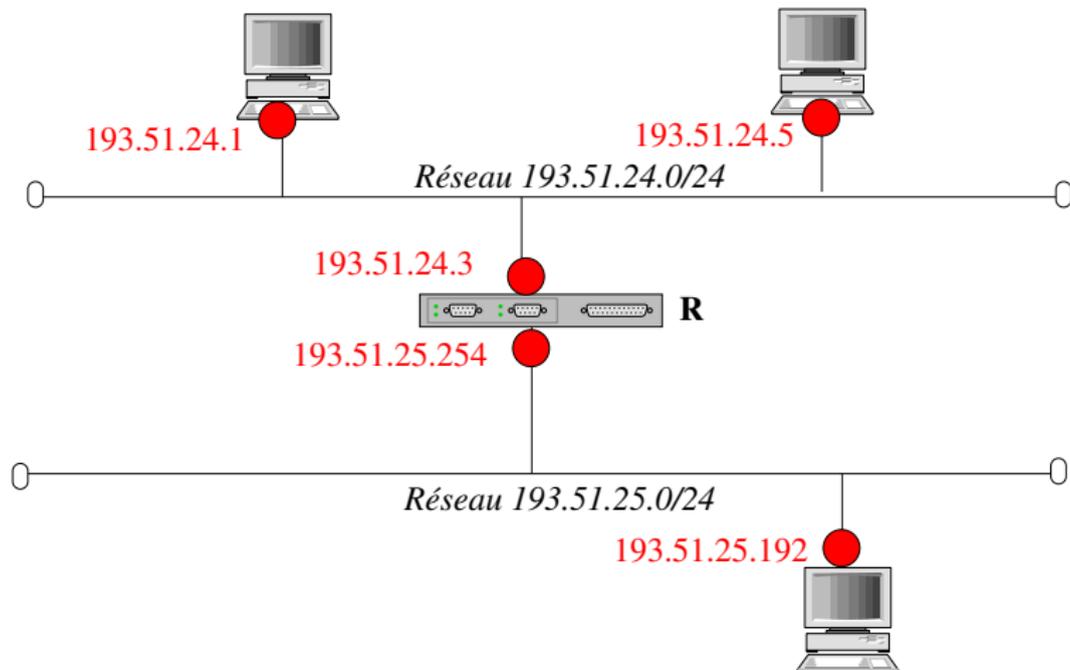
Exemple : le routeur R a deux interfaces, il a donc deux adresses :  
193.51.25.254 et 193.51.24.3



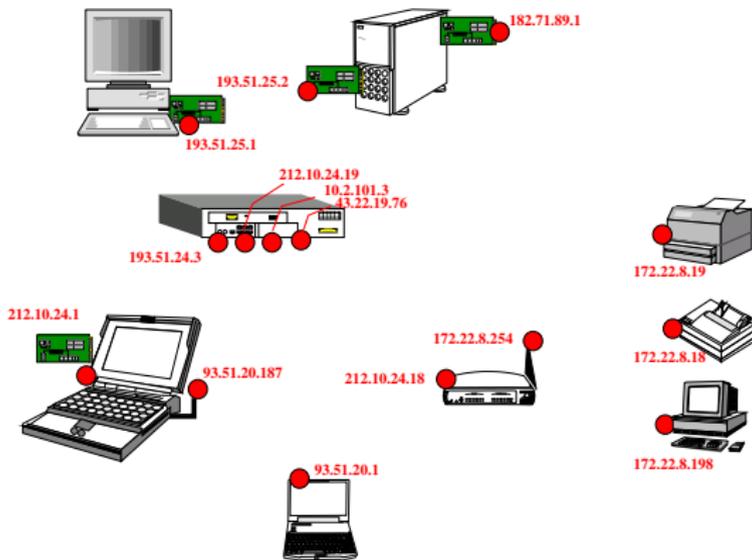
# Adressage IP et interface

Une adresse IP est associée à une interface.

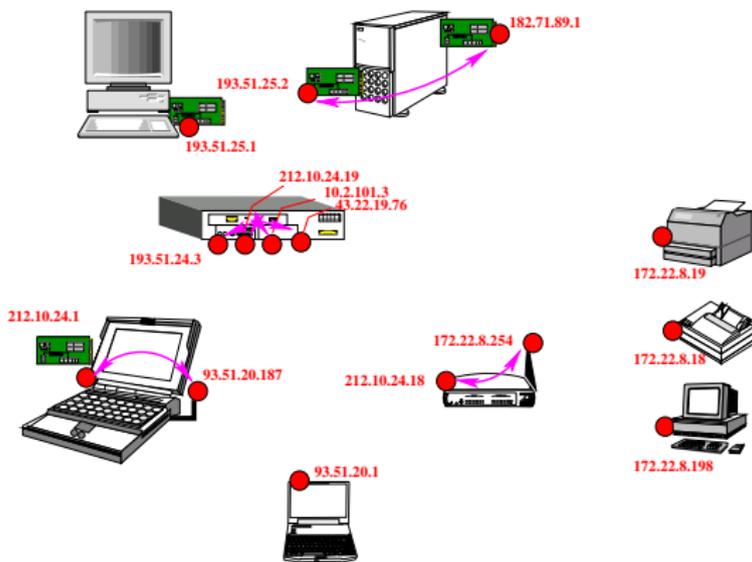
Exemple : le routeur R a deux interfaces, il a donc deux adresses :  
193.51.25.254 et 193.51.24.3



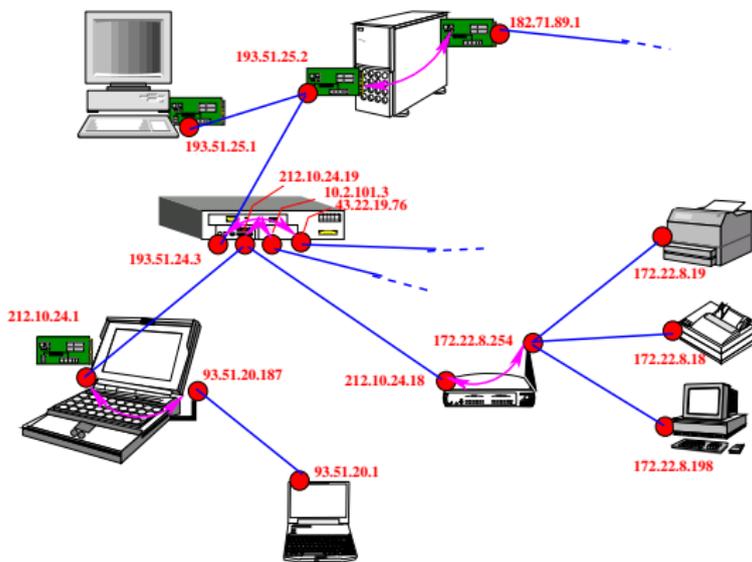
# Adresse IP et interface



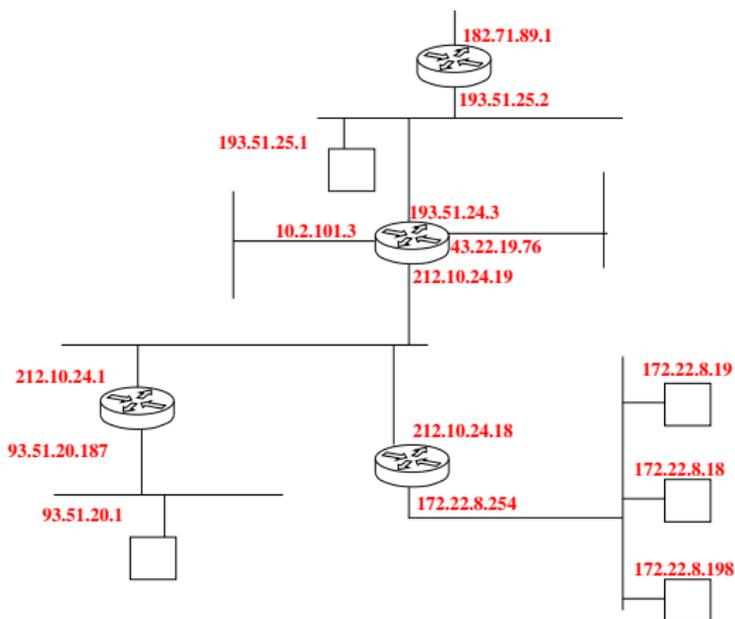
# Adresse IP et interface



# Adresse IP et interface

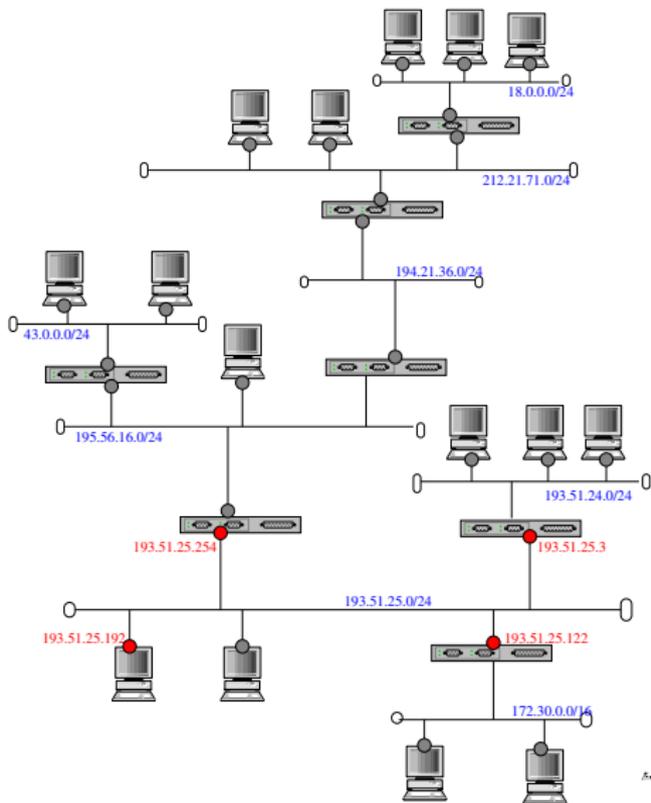


# Adresse IP et interface



# Table de routage

Pour aller sur le réseau	Je dois passer par
172.30.0.0/16	193.51.25.122
193.51.24.0/24	193.51.25.3
18.0.0.0/24	193.51.25.254
212.21.71.0/24	193.51.25.254
43.0.0.0/24	193.51.25.254
195.56.16.0/24	193.51.25.254
194.21.36.0/24	193.51.25.254

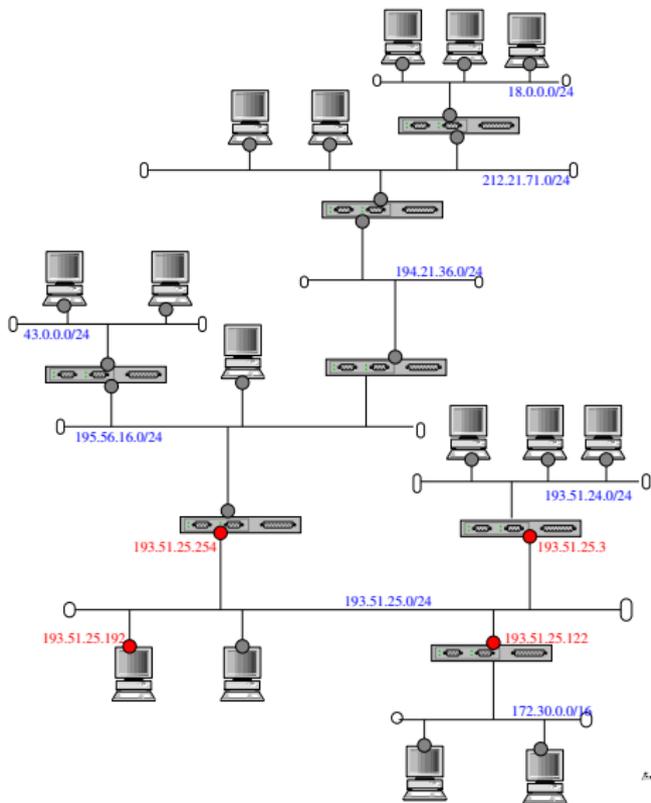


# Simplification des tables de routage

- Si il faut répertorier tous les réseaux de l'Internet dans chaque table de routage
- $\Rightarrow$  Explosion des tables de routage
- route par défaut
- agrégation de réseaux : CIDR

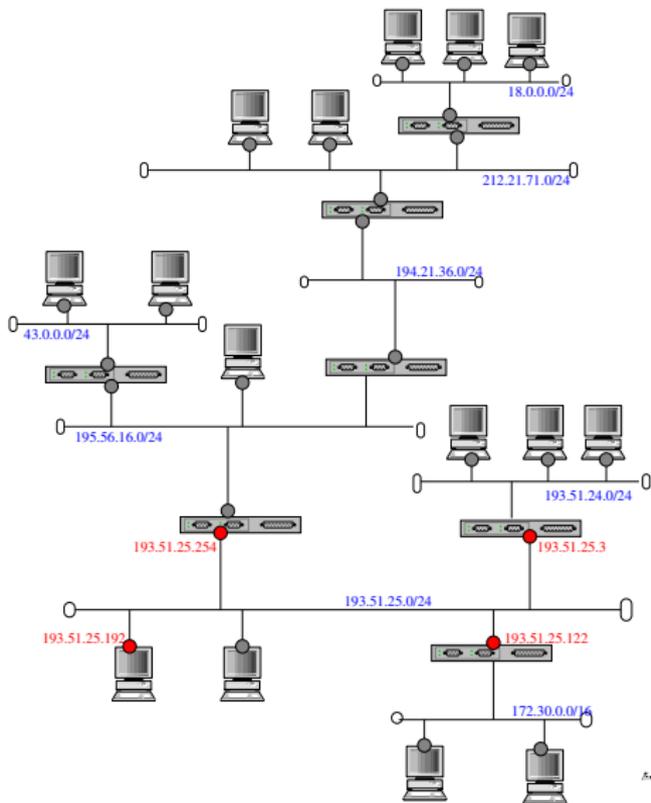
# Route par défaut

Pour aller sur le réseau	Je dois passer par
172.30.0.0/16	193.51.25.122
193.51.24.0/24	193.51.25.3
18.0.0.0/24	193.51.25.254
212.21.71.0/24	193.51.25.254
43.0.0.0/24	193.51.25.254
195.56.16.0/24	193.51.25.254



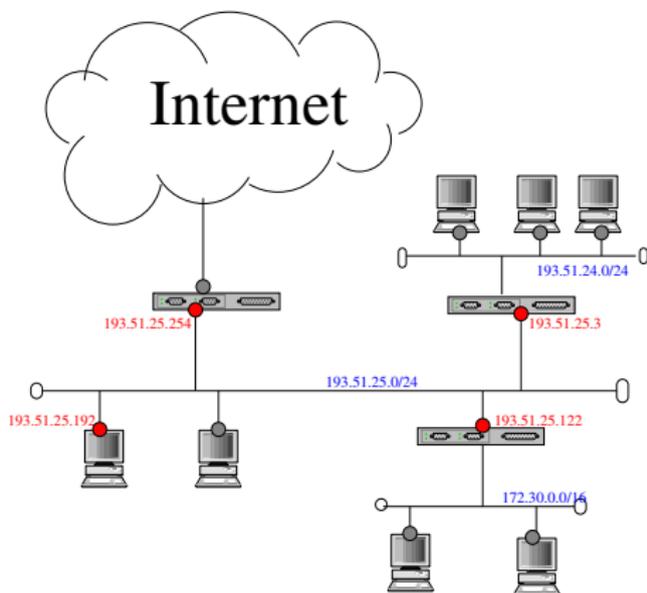
# Route par défaut

Pour aller sur le réseau	Je dois passer par
172.30.0.0/16	193.51.25.122
193.51.24.0/24	193.51.25.3
default	193.51.25.254



# Route par défaut

Pour aller sur le réseau	Je dois passer par
172.30.0.0/16	193.51.25.122
193.51.24.0/24	193.51.25.3
default	193.51.25.254



# Aggrégation de routes : CIDR (Classless Inter-Domain Routing)

Pour agréger les tables de routage :

- allouer aux "utilisateurs" des réseaux de classe C contigus des réseaux contigus ont les mêmes bits de poids fort :  
⇒ ils ont même préfixe
- grouper les préfixes par région, prestataires ...
- router les préfixes des supernets (ou agrégats) une seule entrée par agrégat dans la table de routage suffit

Exemple 1 :

Les deux réseaux :

193.51.32.0 de masque 255.255.255.0 (noté 193.51.32.0 / 24)

193.51.33.0 de masque 255.255.255.0 (noté 193.51.33.0 / 24)

sont agrégés en 193.51.32.0 255.255.254.0 (193.51.32.0 / 23)

5.

# Aggrégation de routes : CIDR (Classless Inter-Domain Routing)

Exemple 2 :

Les huit réseaux :

201.18.168.0/24

201.18.169.0/24

201.18.170.0/24

201.18.171.0/24

201.18.172.0/24

201.18.173.0/24

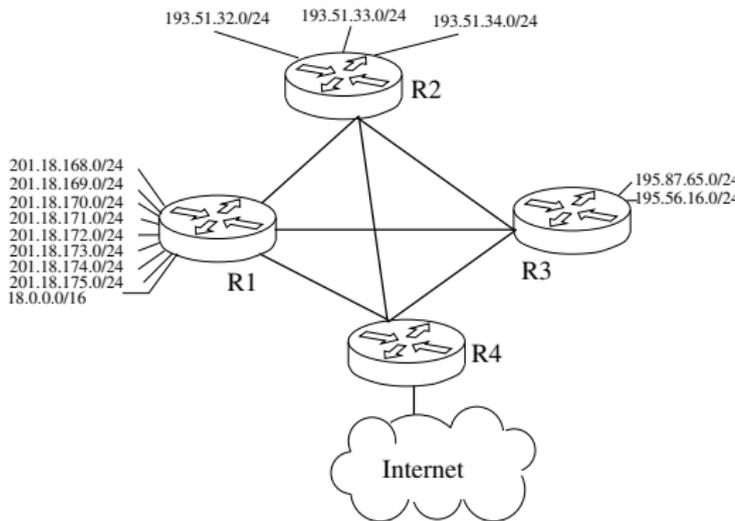
201.18.174.0/24

201.18.175.0/24

sont agrégés en 201.18.168.0/21

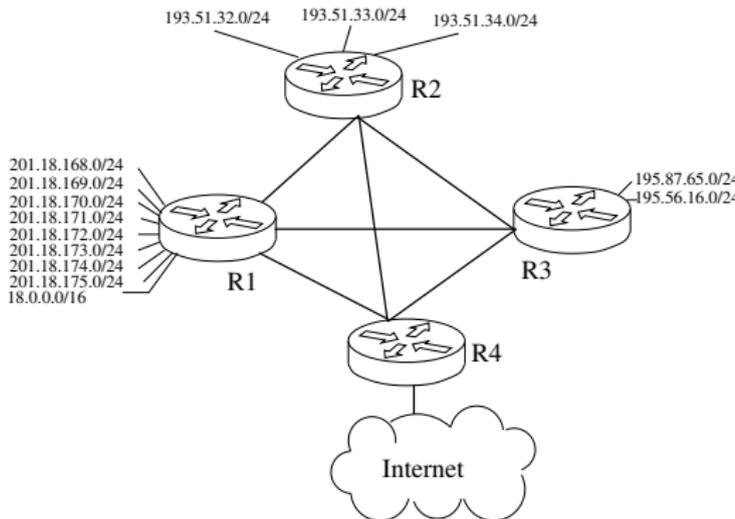
# Agrégation des tables de routage

Réseau destination	Routeur
201.18.168.0/24	R1
201.18.169.0/24	R1
201.18.170.0/24	R1
201.18.171.0/24	R1
201.18.172.0/24	R1
201.18.173.0/24	R1
201.18.174.0/24	R1
201.18.175.0/24	R1
18.0.0.0/16	R1
193.51.32.0/24	R2
193.51.33.0/24	R2
193.51.34.0/24	R2
195.87.65.0/24	R3
195.56.16.0/24	R3
Tous les réseaux de l'Internet	R4
1.0.0.0/8	R4
2.0.0.0/8	R4
...	R4
128.0.0.0/16	R4
128.1.0.0/16	R4
...	R4
223.255.254.0/24	R4
223.255.255.0/24	R4



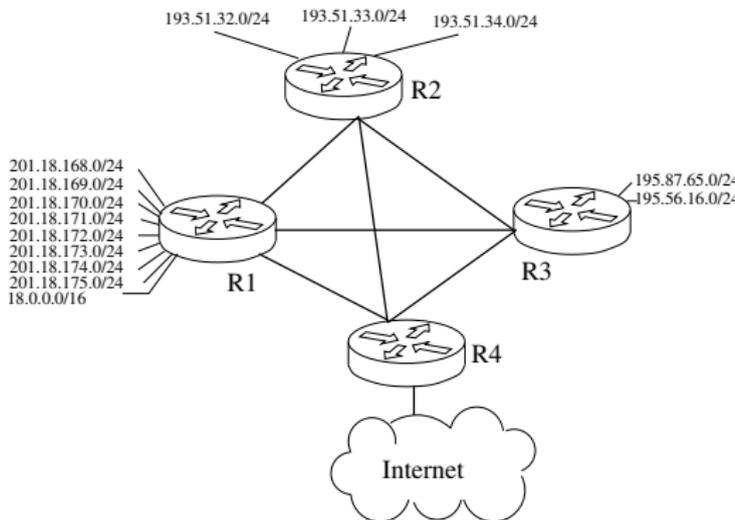
# Agrégation des tables de routage

Réseau destination	Routeur
201.18.168.0/24	R1
201.18.169.0/24	R1
201.18.170.0/24	R1
201.18.171.0/24	R1
201.18.172.0/24	R1
201.18.173.0/24	R1
201.18.174.0/24	R1
201.18.175.0/24	R1
18.0.0.0/16	R1
193.51.32.0/24	R2
193.51.33.0/24	R2
193.51.34.0/24	R2
195.87.65.0/24	R3
195.56.16.0/24	R3
default	R4

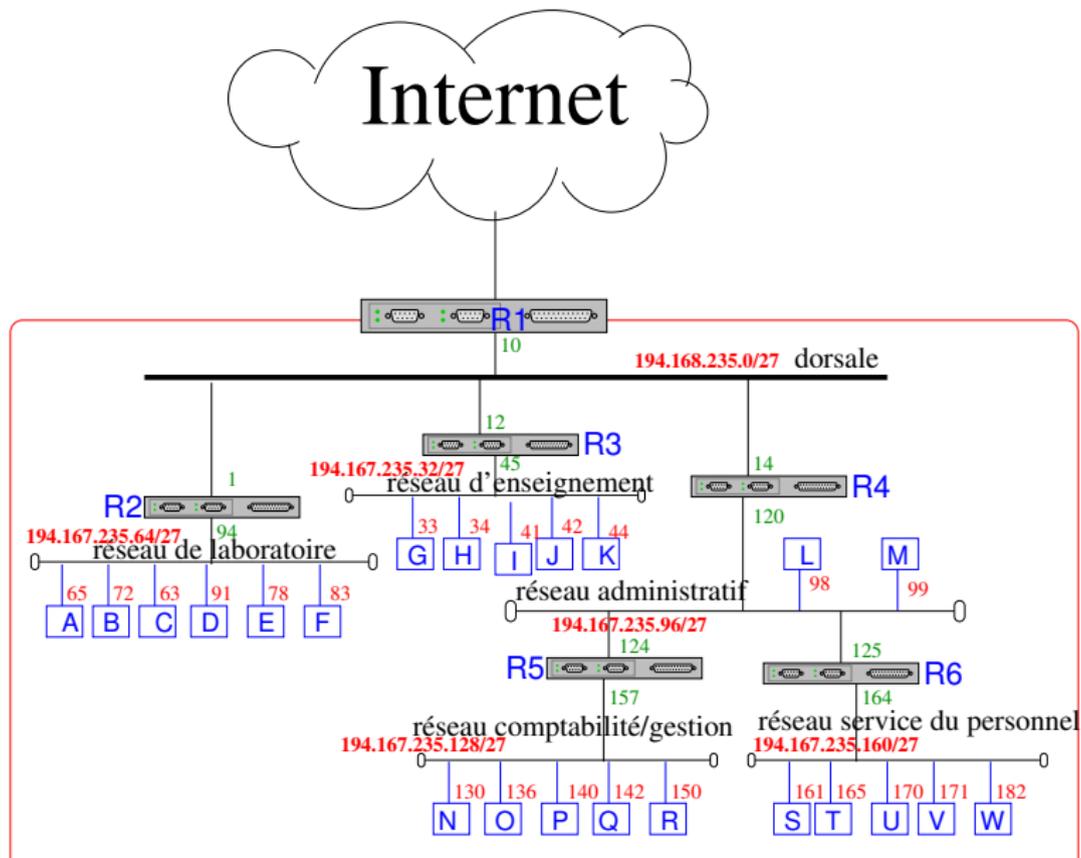


# Agrégation des tables de routage

Réseau destination	Routeur
201.18.168.0/21	R1
18.0.0.0/16	R1
193.51.32.0/23	R2
193.51.34.0/24	R2
195.87.65.0/24	R3
195.56.16.0/24	R3
default	R4



# Exemple de réseau



# Tables de routage (1/2)

R1

Réseau Destination	Routeur	Interface
default	<i>R_FAI</i>	ext
194.168.235.0/27 (dorsale)	lien local	int
194.168.235.32/27 (enseignement)	194.168.235.12	int
194.168.235.64/27 (laboratoire)	194.168.235.1	int
194.168.235.64/27 (administratif)	194.168.235.14	int
194.168.235.128/27 (compta/gestion)	194.168.235.14	int
194.168.235.160/27 (personnel)	194.168.235.14	int

R2

Réseau Destination	Routeur	Interface
default	194.168.235.10	ext
194.168.235.0/27 (dorsale)	lien local	ext
194.168.235.32/27 (enseignement)	194.168.235.12	ext
194.168.235.64/27 (laboratoire)	lien local	int
194.168.235.64/27 (administratif)	194.168.235.14	ext
194.168.235.128/27 (compta/gestion)	194.168.235.14	ext
194.168.235.160/27 (personnel)	194.168.235.14	ext

R3

Réseau Destination	Routeur	Interface
default	194.168.235.10	ext
194.168.235.0/27 (dorsale)	lien local	ext
194.168.235.32/27 (enseignement)	lien local	int
194.168.235.64/27 (laboratoire)	194.168.235.1	ext
194.168.235.64/27 (administratif)	194.168.235.14	ext
194.168.235.128/27 (compta/gestion)	194.168.235.14	ext
194.168.235.160/27 (personnel)	194.168.235.14	ext

# Tables de routage (2/2)

R4

Réseau Destination	Routeur	Interface
default	194.168.235.10	ext
194.168.235.0/27 (dorsale)	lien local	ext
194.168.235.32/27 (enseignement)	194.168.235.12	ext
194.168.235.64/27 (laboratoire)	194.168.235.1	ext
194.168.235.64/27 (administratif)	lien local	int
194.168.235.128/27 (compta/gestion)	194.168.235.124	int
194.168.235.160/27 (personnel)	194.168.235.125	int

R5

Réseau Destination	Routeur	Interface
default	194.168.235.120	ext
194.168.235.64/27 (administratif)	lien local	ext
194.168.235.128/27 (compta/gestion)	lien local	int
194.168.235.160/27 (personnel)	194.168.235.125	ext

R6

Réseau Destination	Routeur	Interface
default	194.168.235.120	ext
194.168.235.64/27 (administratif)	lien local	ext
194.168.235.128/27 (compta/gestion)	194.168.235.124	ext
194.168.235.160/27 (personnel)	lien local	int

# Et si on rajoutait ou enlevait des routeurs ?

- Configurer les tables de routages des routeurs supplémentaire
- Modifier les tables de routage de chacun des routeurs déjà présents
- ... et surtout, ne pas se tromper !!!

⇒ Dans un environnement complexe, la mise en oeuvre du routage statique est souvent difficile à maintenir.

- 1 Routage statique
- 2 Principes généraux du routage dynamique
  - Protocoles de routage
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

# Routage dynamique

- Dans un environnement complexe, la mise en œuvre du routage statique est souvent difficile à maintenir
- La mise en place d'un mécanisme de routage dynamique permet de faciliter les mises à jour
- Chaque routeur diffuse la liste des réseaux sur lesquels il est connecté
- Chaque routeur met à jour sa table de routage à partir des informations reçues depuis les autres
- Démons de routage : **routed**, **gated**, **ripd**, **ospfd**

# Routage dynamique

- Chemin de coût le moins élevé
- Echange d'information entre routeurs
- Algorithmes
  - à vecteur de distance (*distance vector*) : RIP
  - à état de liaison (*link state*) : OSPF

# Protocoles de routage

Le protocole de routage consiste à définir comment sont échangées les informations de routage, et donc à :

- découvrir les autres routeurs du réseau
- construire les tables de routage
- maintenir les tables de routage à jour

Attention : protocole de routage  $\neq$  politique de routage (décision)

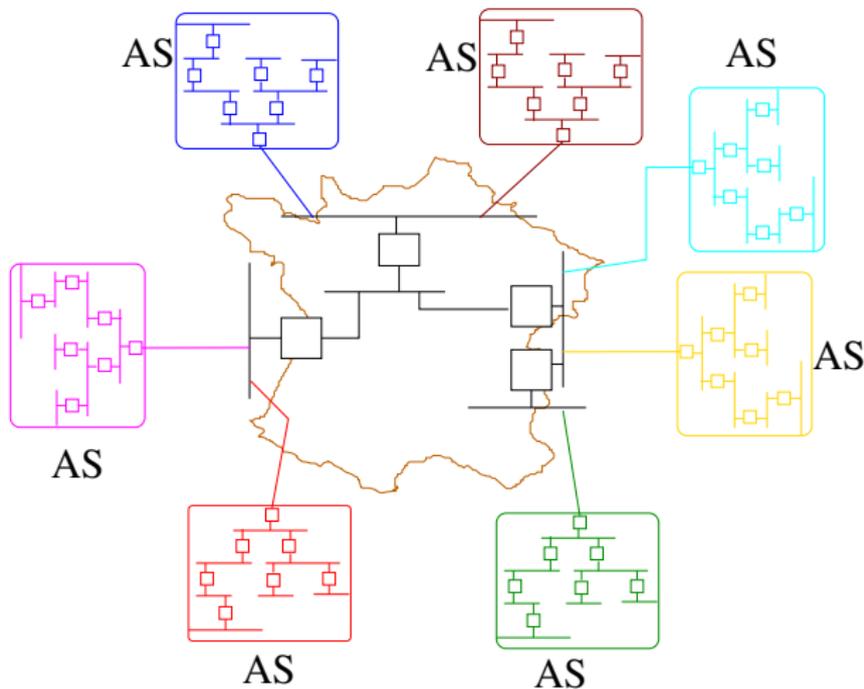
# Routage entre réseaux

Interconnexion de réseaux de différents opérateurs  $\Rightarrow$

- chaque opérateur se débrouille pour router ses propres informations en interne.
- protocole commun d'information de routage entre les réseaux des différents opérateurs.

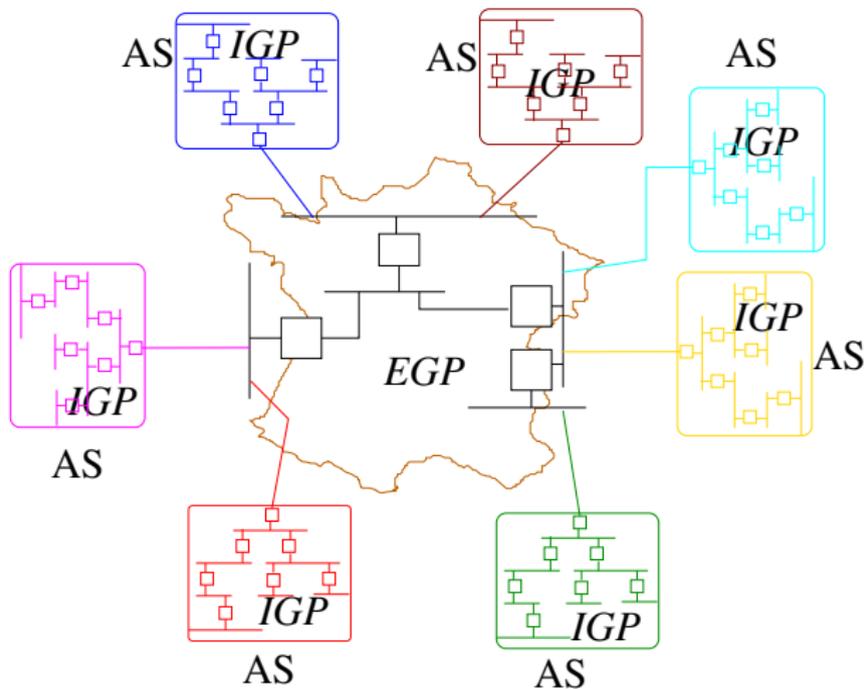
# AS (Autonomous System)

AS : ensemble de réseaux contrôlés par une seule autorité.



# AS (Autonomous System)

AS : ensemble de réseaux contrôlés par une seule autorité.



# AS (Autonomous System)

Les ressources d'adressage et de routage de l'internet -constituées par les adresses IP et les numéros AS- ont été réparties par l'IANA auprès de RIR ((Regional Address Registry). Les RIR répartissent ensuite ces ressources auprès de Local Internet Registries (LIR = Registres Internet Locaux) qui attribuent les adresses aux utilisateurs finaux.

- ARIN pour les zones Amérique du Nord
- AfriNIC pour l'Afrique
- APNIC pour les zones Asie-Pacifique
- LACNIC pour les zones Amériques du Sud - Caraïbes
- RIPE NCC pour la zone Europe étendue.

Les numéros d'AS sont des entiers stockés sur 16 bits.  $\Rightarrow$  Il ne peut donc y en avoir que 65535 au niveau mondial (très peu).

# IGP (*Interior Gateway Protocol*)

IGP : Protocole de routage utilisé dans les réseaux sous même entité administrative

- qu'à l'intérieur d'une entité (entreprise, association, etc)
- décisions (suppression/ajout d'une ligne) peuvent être prises par un service unique
- but : trouver la route la plus efficace, en faisant confiance aux autres routeurs.
- ex : RIP, OSPF

# EGP (*Exterior Gateway Protocol*)

EGP : Protocole de routage adapté à la redistribution de préfixes vers des réseaux extérieurs, ayant une entité administrative différente

- s'utilise entre entités distinctes (souvent concurrentes).
- Impossibilité de prendre une décision qui s'imposera à tous.
- On n'est pas prévenu de ce que vont faire les autres.
- Idée de méfiance : le but n'est pas de trouver la meilleure route mais au contraire d'empêcher les routeurs de choisir une route dont on ne voudrait pas.
- Pas d'information de **routage** mais d'**accessibilité**
- ex : BGP

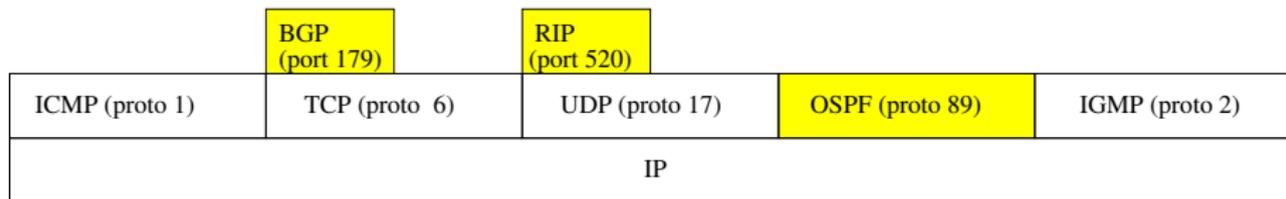
# Protocoles de routage

## IGP

- RIP (Routing Information Protocol) v1, v2 : protocole à vecteur de distance (Distance Vector)
- OSPF (Open Shortest Path First) : protocole de routage à état de lien (Link-state)
- IGRP/EIGRP protocole propriétaire CISCO

## EGP

- BGP (Border Gateway Protocol) : protocole à vecteur de chemin. C'est le protocole standard de l'Internet pour les interconnexions entre opérateurs.



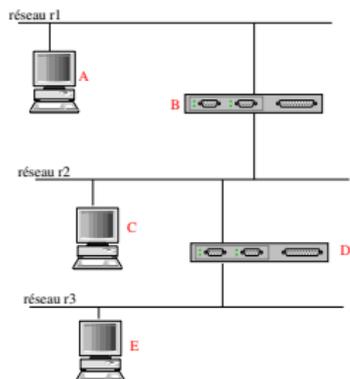
- 1 Routage statique
- 2 Principes généraux du routage dynamique
- 3 Protocoles de routage interne (IGP) : RIP**
- 4 Protocoles de routage interne (IGP) : OSPF
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

# RIP : Principe général

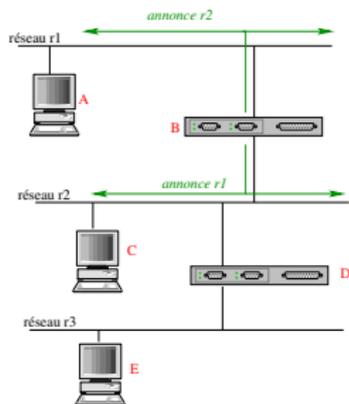
## Principe :

- Chaque routeur annonce périodiquement (30s) tous ses réseaux et le nombre de saut pour y aller
- Chaque machine écoute les annonces des passerelles et actualise sa table de routage
- Si au bout d'un certain temps (3mn=180s), un réseau n'est plus annoncé, il est supprimé de la table de routage.
- Il n'y a pas d'accusé de réception de message

# Principe général

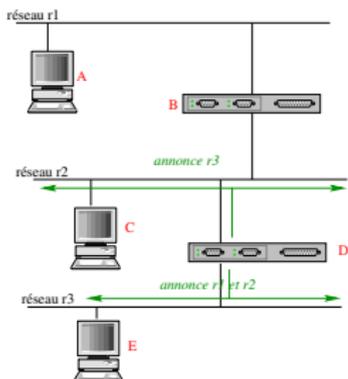


# Principe général



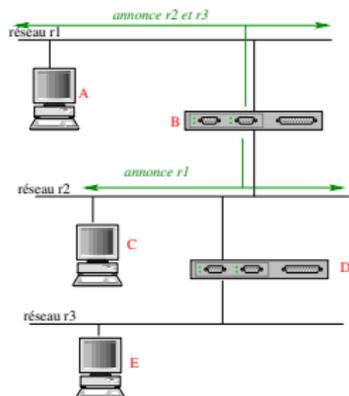
Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B

# Principe général



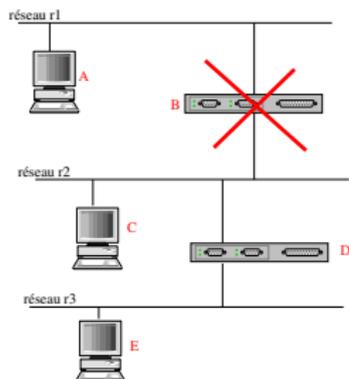
Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B
t2	D	r1 et r2	r3	E	r1 via D
					r2 via D
	D	r3	r2	B	r3 via D
	D			C	r3 via D

# Principe général



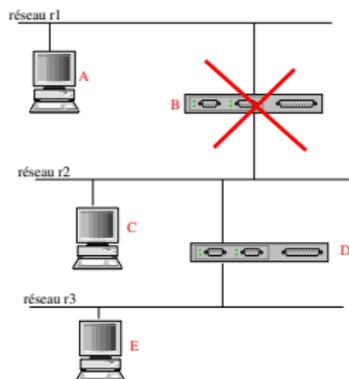
Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B
t2	D	r1 et r2	r3	E	r1 via D
					r2 via D
	D	r3	r2	B	r3 via D
				C	r3 via D
t3	B	r2 et r3	r1	A	r3 via B
	B	r1	r2		

# Principe général



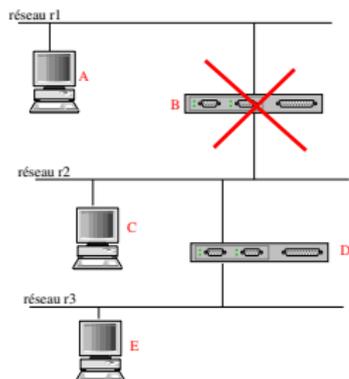
Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B
t2	D	r1 et r2	r3	E	r1 via D
					r2 via D
	D	r3	r2	B	r3 via D
	D			C	r3 via D
t3	B	r2 et r3	r1	A	r3 via B
	B	r1	r2		
B tombe en panne					

# Principe général



Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B
t2	D	r1 et r2	r3	E	r1 via D
					r2 via D
	D	r3	r2	B	r3 via D
				C	r3 via D
t3	B	r2 et r3	r1	A	r3 via B
		B	r1		
B tombe en panne					
t3 + 180s				A	r2 non routé
					r3 non routé
				C	r1 non routé
				D	r1 non routé

# Principe général



Temps	Routeur	Annonce	Sur	Machine	Route
t1	B	r2	r1	A	r2 via B
		r1	r2	C	r1 via B
				D	r1 via B
t2	D	r1 et r2	r3	E	r1 via D
					r2 via D
	D	r3	r2	B	r3 via D
				C	r3 via D
t3	B	r2 et r3	r1	A	r3 via B
		B	r1		
B tombe en panne					
t3 + 180s				A	r2 non routé
					r3 non routé
				C	r1 non routé
				D	r1 non routé
t3 + 360s				E	r1 non routé

# Routage à vecteur de distance (Algorithme de Bellman-Ford)

- Périodiquement un routeur envoie une copie de sa table de routage à tous les routeurs directement accessibles.
- Lorsque que J transmet un rapport au routeur K, K examine l'ensemble des destinations annoncées et leur distance. K modifie son entrée vers une destination si :
  - J connaît un plus court chemin
  - ou si J annonce une destination que K ne possède pas
  - ou si une destination via J a changé
- l'entrée de la table de K mise à jour signale la distance  $n + 1$  (avec  $n$  la distance annoncée par J pour la destination)

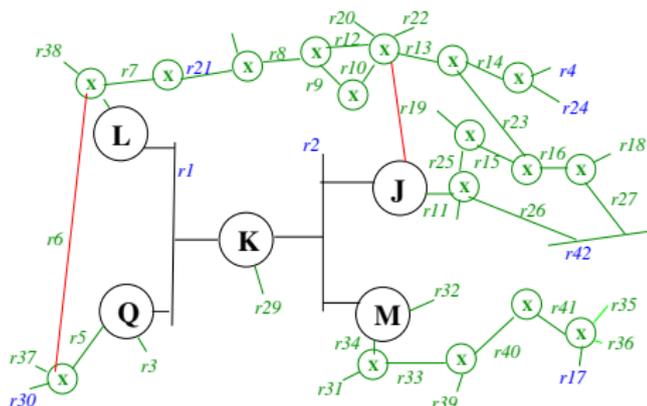
# Routage à vecteur de distance (Bellman-Ford)

Table de routage du  
routeur K

Destination	Dist.	Route
Réseau 1	0	directe
Réseau 2	0	directe
Réseau 4	8	Routeur L
Réseau 17	5	Routeur M
Réseau 24	6	Routeur J
Réseau 30	2	Routeur Q
Réseau 42	2	Routeur J

Message de mise à jour  
du routage issu du  
routeur J

Destination	Dist.
Réseau 1	2
Réseau 4	3
Réseau 17	6
Réseau 21	4
Réseau 24	5
Réseau 30	10
Réseau 42	4



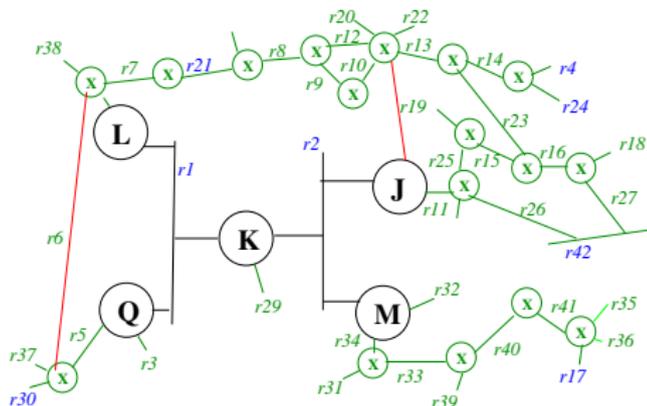
# Routage à vecteur de distance (Bellman-Ford)

## Table de routage du routeur K

Destination	Dist.	Route
Réseau 1	0	directe
Réseau 2	0	directe
<b>Réseau 4</b>	4	Routeur J
Réseau 17	5	Routeur M
<b>Réseau 21</b>	5	Routeur J
Réseau 24	6	Routeur J
Réseau 30	2	Routeur Q
<b>Réseau 42</b>	3	Routeur J

## Message de mise à jour du routage issu du routeur J

Destination	Dist.
Réseau 1	2
<b>Réseau 4</b>	3
Réseau 17	6
<b>Réseau 21</b>	4
Réseau 24	5
Réseau 30	10
<b>Réseau 42</b>	4



# RIP (Routing Information Protocol)

Principe :

- Chaque routeur annonce (par diffusion) périodiquement (30s) tous ses réseaux et le nombre de saut pour y aller
- Chaque machine écoute les annonces des passerelles et actualise sa table de routage
- Si au bout d'un certain temps (3mn=180s), un réseau n'est plus annoncé, il est supprimé de la table de routage.
- Il n'y a pas d'accusé de réception de message
- Protocole sur UDP, port 520

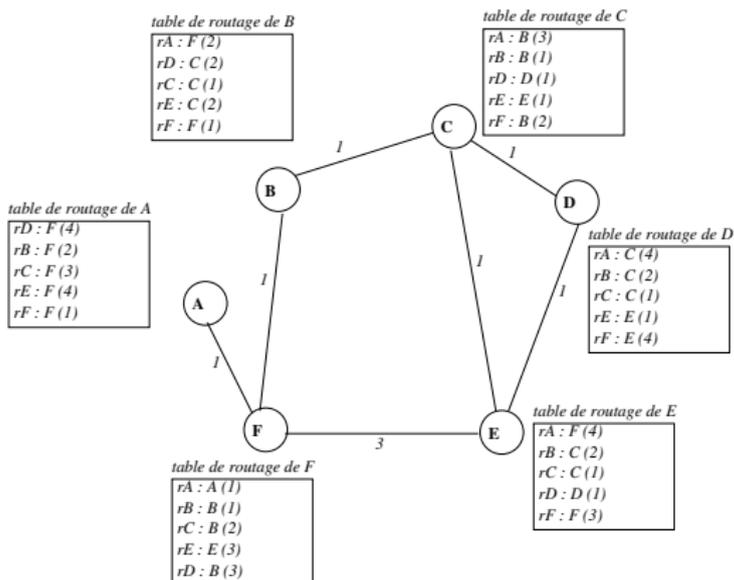
2 types de routeurs :

- **Routeur actif** : diffuse ses informations de routage vers les autres noeuds.
- **Routeur passif** : écoute ces informations et met à jour sa table de routage.

RIPv1 diffuse (broadcast) et RIPv2 multicast toute leur table de routage toutes les 30 secondes.

# RIP (Routing Information Protocol)

- Routage à vecteur de distance.
- chaque noeud n'a d'information que sur le prochain saut (next hop)
- pas de décisions globales



# Problèmes de RIP v1

- limite de 16 sauts (16 : inaccessible)  $\Rightarrow$  ne peut pas aller plus loin que 15 routeurs (hops)
- converge lentement (si route changent souvent, peut ne pas se stabiliser)
- informations circulent lentement
- trafic important
- boucles possibles
- ne se base que sur une seule métrique : le hop  $\Rightarrow$  peut choisir des routes lentes.
- pas de gestion de masque  $\Rightarrow$  pas de routage de sous-réseaux
- pas d'authentification
- 25 entrées maximum dans la table de routage (car taille du message = 512 o)

# RIP v2

2 algorithmes de plus :

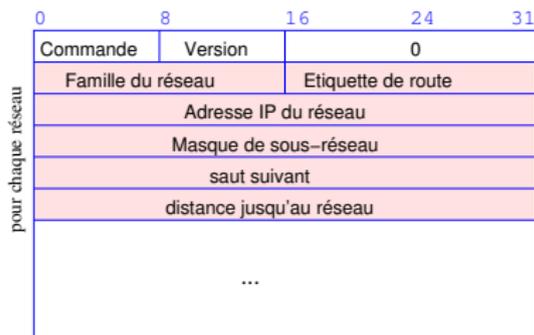
- *split horizon* : les données ne sont pas renvoyées vers le noeud d'où on les a appris
- *hold down* : Le routeur ignore les informations relatives à un réseau pendant une période fixe après réception d'un message qui en spécifie l'inaccessibilité.
- *poison reverse* : si on détecte une route coupée et qu'on reçoit un message avec un coût très supérieur au coût initial, on ignore l'information (considérée revenue par une boucle).

Plus les améliorations suivantes :

- masque de sous-réseau : sous-réseaux possibles + aggrégation des routes
- authentification (mot de passe en clair ou chiffré sur 16 octets)
- utilisation de domaines logiques (on ignore les messages d'un autre domaine)

# RIPv2 : Format

- commande :
  - 1 : demande d'information de routage
  - 2 : réponse contenant les info de la table de routage de l'expéditeur
  - 9 : demande de mise à jour (avec circuit de commande)
  - 10 : réponse de mise à jour (avec circuit de commande)
  - 11 : accusé de réception de mise à jour



# RIPv2 : clivage d'horizon (split horizon)

Contre le problème de convergence lente :

Un routeur ne transmet pas les informations relatives à une route vers la même interface que celle qui l'a initialement annoncé.

⇒ Les bonnes nouvelles vont vite, les mauvaises lentement.

XXX

# RIPv2 : mécanisme de gel (hold down)

Le routeur ignore les informations relatives à un réseau pendant une période fixe (60s) après réception d'un message qui en spécifie l'inaccessibilité.

XXX

# RIPv2 : antidote (poison reverse)

Après la disparition d'une connexion, le routeur qui l'a annoncé conserve l'entrée pendant plusieurs cycles de mise à jour en incluant un coût infini dans ses messages de diffusion.

XXX

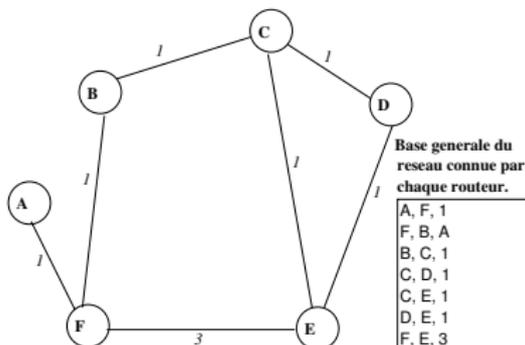
- RIP v2 corrige certains problèmes de RIP v1
- Encore des problèmes :
  - métrique : sauts uniquement
  - portée maximum de 15 sauts
  - taille de la table de 25 entrées maximum.
- ⇒ RIPv2 ne peut s'appliquer qu'aux petits et moyens réseaux.

- 1 Routage statique
- 2 Principes généraux du routage dynamique
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF**
  - Aire de routage
  - Messages OSPF
  - Protocole OSPF
  - Algorithme Shortest Path First (SPF)
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

# Routage à état de lien

## Principe :

- Envoyer à tous les noeuds les informations au sujet des voisins.
- Les noeuds ont une copie complète de la carte du réseau
- Chaque noeud exécute Dijkstra (plus court chemin et pas de cycles).



```

procedure DIJKSTRA_SP(V, E, w, s)
begin
  VT := {s};
  for all v ∈ (V - VT) do
    if (s, v) exists set l[v] := w(s, v)
    else set l[v] := ∞;
  while VT ≠ V do
    begin
      find a vertex u such that
      l[u] := min{l[v] | v ∈ (V - VT)};
      VT := VT ∪ u;
      for all v ∈ (V - VT) do
        l[v] := min{l[v], l[u] + w[u, v]};
      endwhile
    endwhile
end DIJKSTRA_SP
  
```

# OSPF ( Open Shortest Path First)

Routage à état de lien (*Link-State*) : permettre au routeur d'avoir une vision globale du réseau et de sa topologie

- une base de données sur chaque noeud représentant la topologie totale du réseau
  - détection de boucle
  - calcul de la route la plus courte par l'algorithme de Dijkstra
- configuration pour chaque interface
- métrique par type de coût (longueur de la file d'attente, débit, distance en saut, etc)
- ne diffuser que les modifications détectées dans la topologie (accessibilité et coût)
- routage par type de service (champ TOS du datagramme)
- notion d'aire de routage

# Aire de routage

Un réseau OSPF est divisé en plusieurs aires (*Area*) qui se connectent à une aire centrale de distribution appelée dorsale (*backbone*).

Chaque aire est désignée par un identifiant de 32 bits mis sous la forme X.Y.Z.T. Cet identifiant ne correspond pas forcément à l'adresse réseau (même si par commodité, on le choisit souvent ainsi).

Pas plus d'une cinquantaine de routeurs maximum par aire.

Réduction du nombre de routeur par zone de diffusion

⇒ trafic de gestion limité

⇒ échange entre routeurs plus nombreux

# Aire dorsale (*area backbone*)

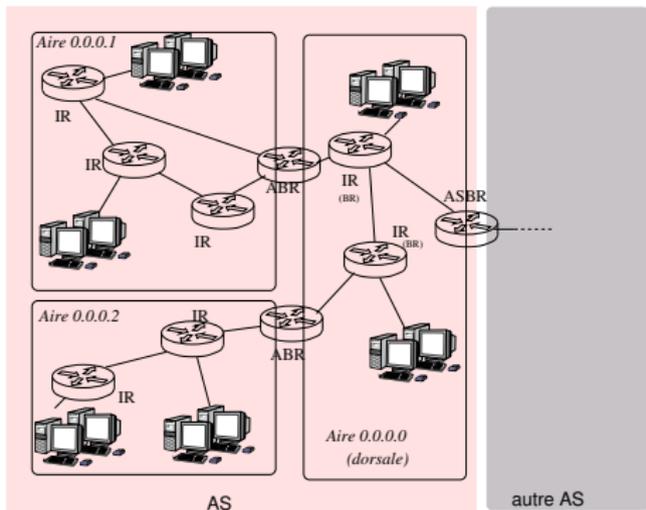
L'aire dorsale :

- a pour identifiant 0.0.0.0 obligatoirement
- sert pour l'acheminement inter-aire
- est obligatoire  $\Rightarrow$  si le réseau n'a pas été découpé en aire, il y en a qu'une seule et c'est la dorsale d'id 0.0.0.0.

# Routeurs

On distingue 3 types de routeurs dans OSPF :

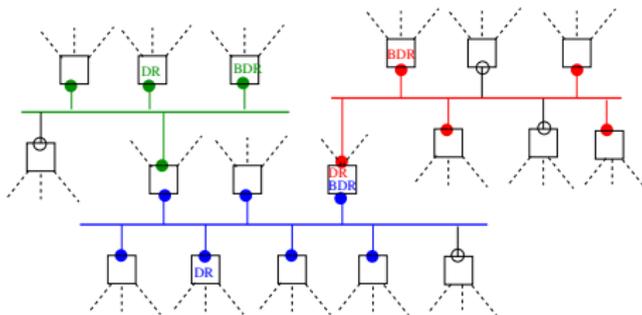
- routeur interne (*Internal Router - IR*) : qui annoncent les routes internes à leur aire
- routeur de la dorsale (*Backbone Router - BR*) : qui annoncent les routes internes à la dorsale. (En fait ce sont des IR de l'aire "dorsale")
- routeur frontière (*Area Boundary Router - ABR*) : qui assurent la connexion à la dorsale
- routeur frontière de systèmes autonomes (*Autonomous*



# Relation de voisinage et relation d'adjacence

Deux routeurs sont **voisins** s'ils appartiennent à une même zone et sont reliés par un même média (lien de diffusion (broadcast domain) ou à chaque extrémité d'un lien point-à-point).

Deux routeurs sont **adjacents** si ils sont voisins et synchronisés, c'est-à-dire s'ils échangent des informations sur la topologie du réseau pour s'assurer du bon fonctionnement l'un de l'autre.

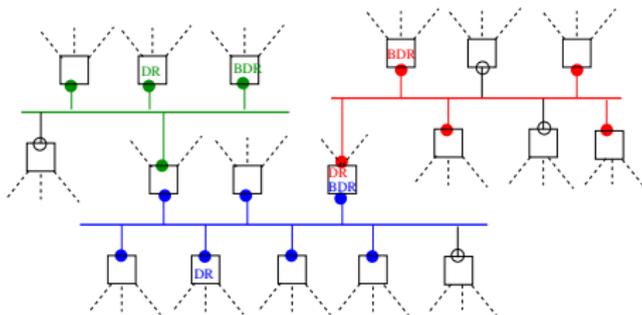


# Routeur Désigné (*Designated Router - DR*)

Un seul routeur parmi les routeurs voisins est responsable.

- le DR (et le BDR) assure la diffusion des messages vers les routeurs de la zone
- évite d'établir  $n^2$  relations entre routeurs voisins et de dupliquer la même information

Le DR (designated router) sert de point central d'échange. Le BDR (backup designated router - DR de secours) surveille le DR et prend sa place s'il ne répond plus.



# Election des DR/BDR

D'abord élection d'un BDR et puis, en l'absence d'un DR, le BDR quitte son statut pour devenir DR.

Election à deux tours :

- 1<sup>er</sup> tour : priorité la plus élevée sur les interfaces du réseau partagé (de 0 : disqualification automatique à 255 : qualification automatique)
  - priorité définie manuellement par interface ou par défaut (= 1)
  - une interface d'un routeur déjà DR est inéligible
- 2<sup>eme</sup> tour : cas de routeurs ex aequo. le routeur de plus haut ID OSPF qui remporte l'élection.
  - ID OSPF (32 bits) adresse IP la plus élevée parmi toutes les interfaces du routeur.

Le vainqueur de l'élection devient BDR.

Les routeurs placent leurs interfaces dans un état en conséquence : DR, BDR ou DROTHER.

Si un DR n'existe plus ou pas du tout, le BDR devient DR et une élection a lieu pour désigner le nouveau BDR.

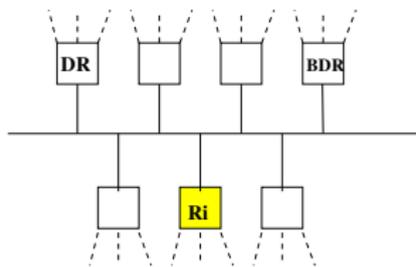
# Rôle du DR

- Le DR maintient la base topologique du réseau
- Relation maître-esclave entre le DR et les routeurs de la zone.
- Les routeurs de la zone ne sont adjacents qu'avec le DR et le BDR. Par contre, ils ne sont pas adjacents entre eux mais peuvent être voisins.
- En cas de panne du DR, le routeur de secours (BDR) maintiendra également la base de donnée et prendra le relais du DR en devenant lui-même DR (et un autre BDR sera élu).
- À chaque fois qu'un routeur envoie une mise à jour, il l'envoie aux DR/BDR (via une adresse dite multicast) et c'est le DR qui rediffuse cette information à tous les routeurs.
- Les routeurs n'ont pas à constamment se mettre à jour entre eux et reçoivent l'ensemble des mises à jours d'une seule et même source.
- L'utilisation du multicast permet de réduire la charge réseau.

# Envoi par inondation (flooding)

Envoi récursif :

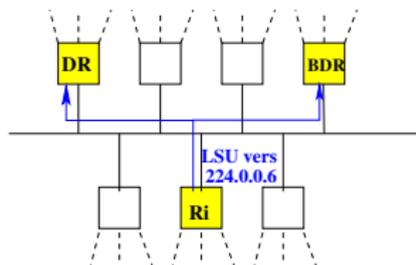
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

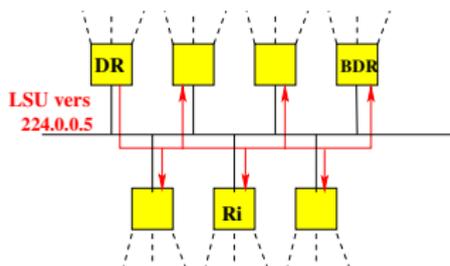
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

## Envoi récursif :

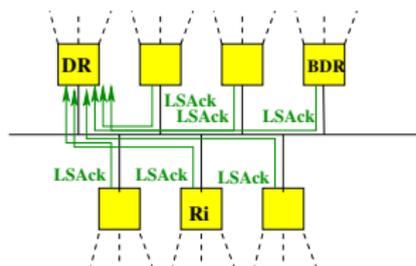
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- **Le DR fait passer le LSU aux autres routeurs (224.0.0.5)**
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

## Envoi récursif :

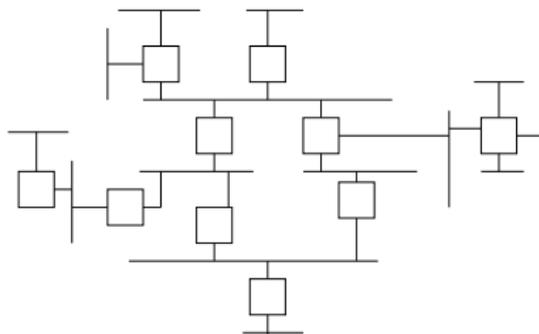
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- **les autres routeurs acquittent avec un LSAck**
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

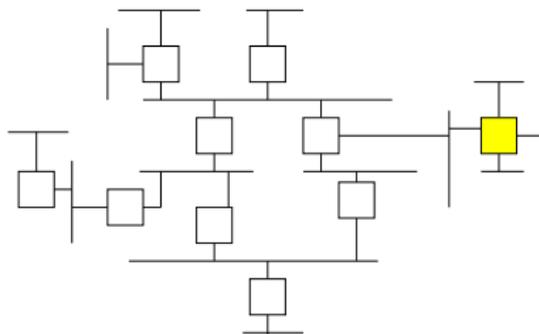
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

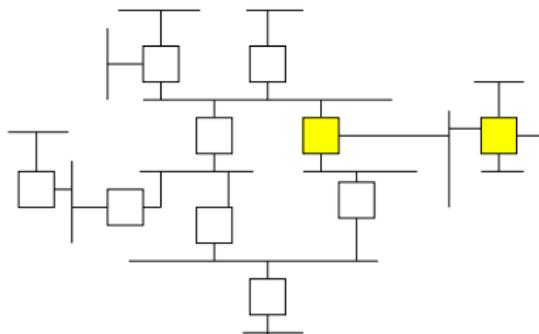
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

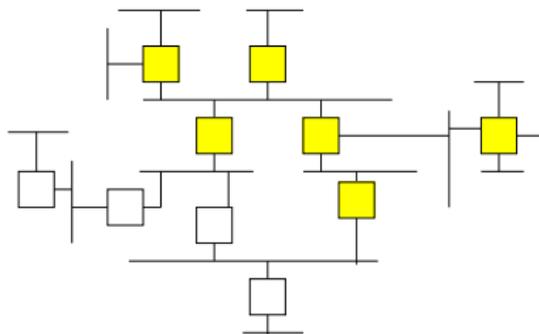
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

## Envoi récursif :

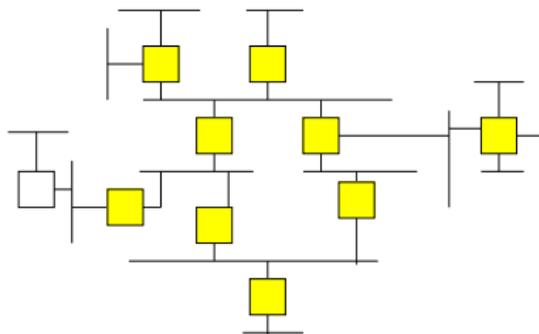
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

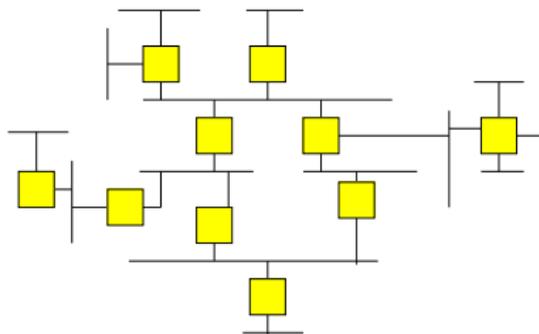
- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)



# Envoi par inondation (flooding)

Envoi récursif :

- Le routeur envoie un LSU contenant l'info du nouvel état de lien au DR et BDR (224.0.0.6)
- Le DR fait passer le LSU aux autres routeurs (224.0.0.5)
- les autres routeurs acquittent avec un LSAck
- Si un routeur se trouve connecté aussi à un autre réseau, il envoie le LSU au DR/BDR de cet autre réseau (224.0.0.6) (inondation récursive)

La coordination par les DR permettent d'éviter de renvoyer deux fois le même LSU et d'éviter les boucles.

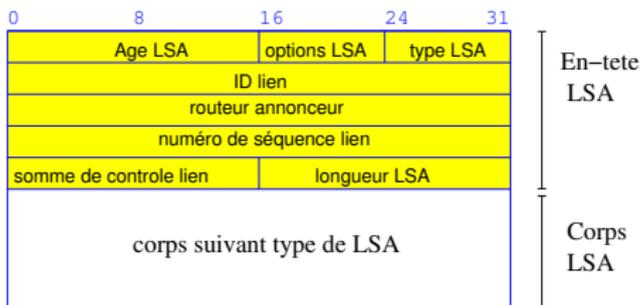
# Bases de données OSPF

Trois bases de données sur chaque routeur :

- **Base de données d'adjacence - Adjacencies database** : Liste de tous les routeurs adjacents avec lesquels est établie une communication bidirectionnelle.  
⇒ Unique pour chaque routeur (Liste composée d'un DR et d'un BDR par interface).
- **Base de données topologique - Link-state database (LSDB)** : BD topologique contenant la liste des informations sur tous les routeurs du réseau. Elle montre la topologie du réseau (graphe).  
⇒ Maintenu identique sur chaque routeur OSPF par inondation périodique des mises à jours  
⇒ Échangé entre le DR et le BDR
- **Table de routage - Forwarding database** : Liste des routes générées par l'algorithme de Dijkstra sur la BD topologique. ⇒ Unique pour chaque routeur  
⇒ Calculé par chaque routeur

# Types de lien

Type de lien	Description
1	LSA Routeur (Router-LSA)
2	LSA Réseau (Network-LSA)
3	LSA Résumé (BR) (Summary-LSA)
4	LSA Résumé (ASBR) (Summary-LSA)
5	LSA AS externe (AS-external-LSA)



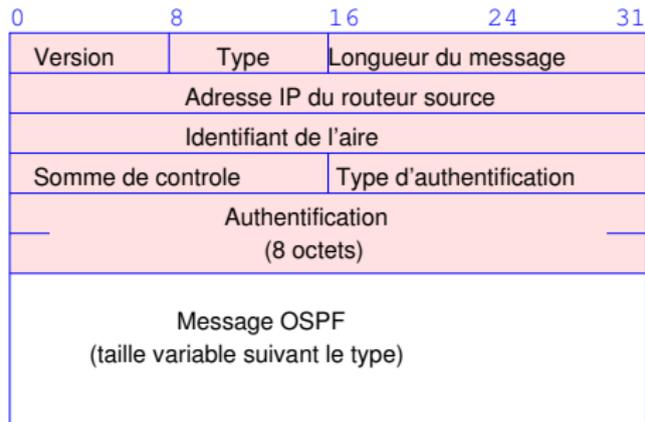
# Types de lien

- **1 - Router-LSA** : chaque routeur d'une aire génère un router-LSA décrivant l'état et le coût de chacun de ses liens (interfaces) vers l'aire.
- **2 - Network-LSA** : chaque DR génère un Network-LSA pour chaque réseau de l'aire supportant plus de 2 routeurs. Ce LSA décrit tous les routeurs attachés à ce réseau, y compris le DR.
- **3/4 - Summary-LSA** : Chaque routeur frontière (BR ou ASBR) génère des Summary-LSA décrivant les destination inter-aires.
- **5 - AS-external-LSA** : Chaque routeur frontière de l'AS (ASBR) génère un AS-external-LSA décrivant les destinations externes à l'AS.

# Message OSPF

## Types de message

- 1 **Hello (type 1)** : établit et maintient les informations d'adjacence des routeurs voisins
- 2 **DBD Database Description Packet (type 2)** : description du contenu de la LSDB d'un routeur
- 3 **LSR - Link State Request (type 3)** : demande de certains états de liens à la LSDB d'un voisin
- 4 **LSU - Link State Update (type 4)** : mise à jour d'états de liens. Transporte les annonce d'état de lien LSA (link-state advertisements) aux routeurs voisins
- 5 **LSAck - Link-State acknowledgement (type 5)** : accusé de réception des LSA



# Message OSPF HELLO (type 1)

OSPF établit et vérifie l'accessibilité des routeurs voisins en envoyant régulièrement des messages HELLO sur chaque lien.

- masque de sous-réseaux.
- période hello : nombre de secondes entre lesquels ce routeur envoie ses messages HELLO
- Options supportées par le routeur
- Priorité du routeur (pour l'élection)
- temporisation de panne : nombre de secondes avant qu'un routeur silencieux soit considéré comme Down.
- Adresse du DR (0 si il n'y en a pas)
- adresse du BDR (0 si il n'y en a pas)
- ID routeurs voisins : adresses IP de chaque routeur dont on a reçu récemment les HELLO messages.

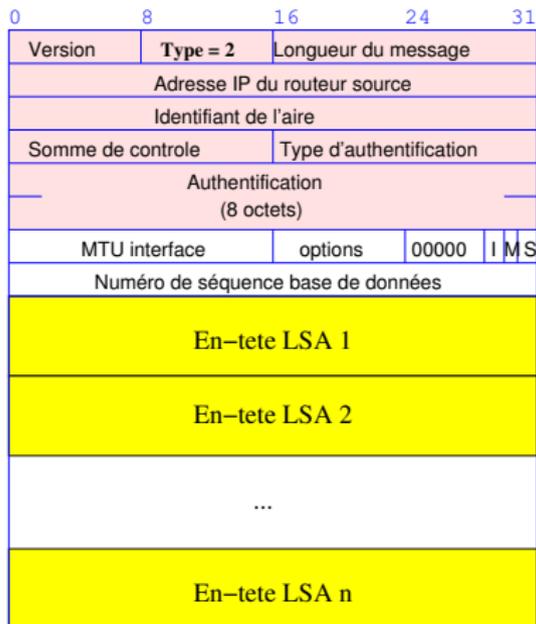
0	8	16	24	31
Version	Type = 1	Longueur du message		
Adresse IP du routeur source				
Identifiant de l'aire				
Somme de contrôle		Type d'authentification		
Authentification (8 octets)				
Masque réseau				
période HELLO		options	prio routeur	
temporisation de panne				
routeur désigné				
routeur désigné de secours				
ID routeur voisin 1				
ID routeur voisin 2				
...				
ID routeur voisin n				

# Message OSPF description de base de données (type 2)

## DBD

Les routeurs s'échangent des messages OSPF description de base de données pour initialiser leur base de données de topologie réseau.

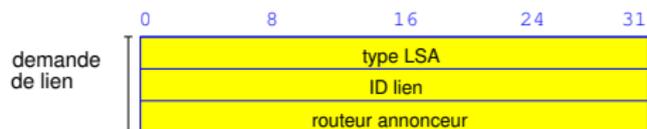
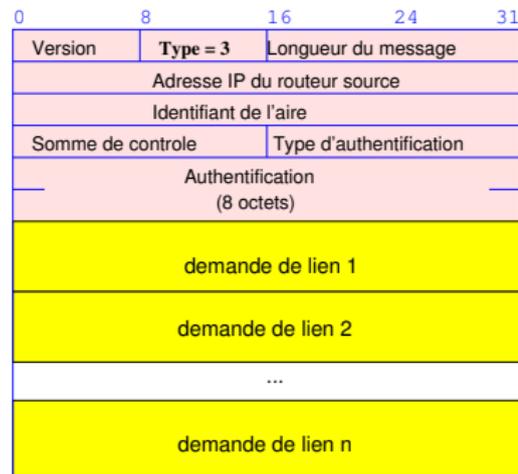
- MTU int. : taille maximum des paquets IP que l'interface du routeur peut envoyer sans fragmentation.
- Options supportées par le routeur
- 5 bits à zero
- Bit I (Init bit) : I=1 si premier packet de la sequence
- Bit M (More bit) : M=1 si d'autres paquets doivent suivre
- Bit MS (Master/slave bit) : MS=1 si le routeur est le maitre durant le processus d'échange de bases.
- DD Sequence Number : séquence utilisée pour numéroter les paquets pour pouvoir les reconstituer dans l'ordre (incrémementation).
- description des liens par des LSA (en-tête seulement)



# Message OSPF demande d'états de lien (type 3) LSR

Demande aux voisins des informations de mise à jour pour les liens qui semblent obsolètes. Dans cette demande sont transmises les informations les plus récentes qu'il possède à propos de ces liens.

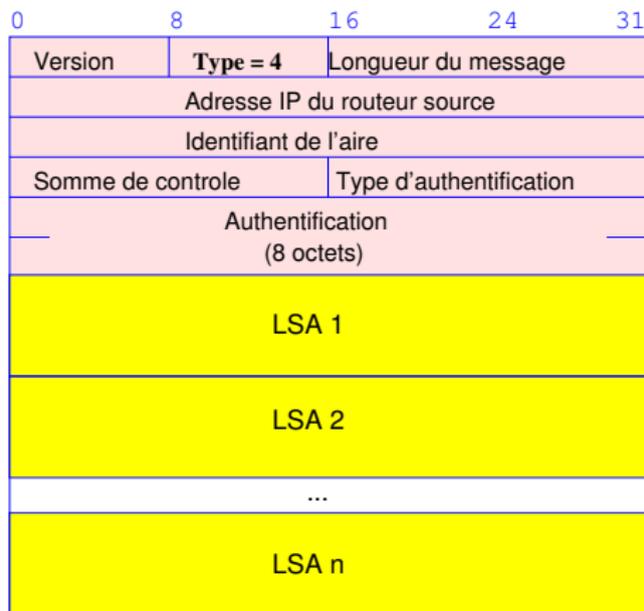
- LS Type : type de LSA recherché
- Link State ID : L'identifiant du LSA (généralement l'IP)
- L'ID du routeur (annonceur) qui a créé les LSA et dont l'update est recherchée.



# Message OSPF mise à jour d'état de lien (type 4) LSU

Les routeurs diffusent l'état des liens (LSA - link State Advertisement) en transmettant des messages de mise à jour d'état de lien.

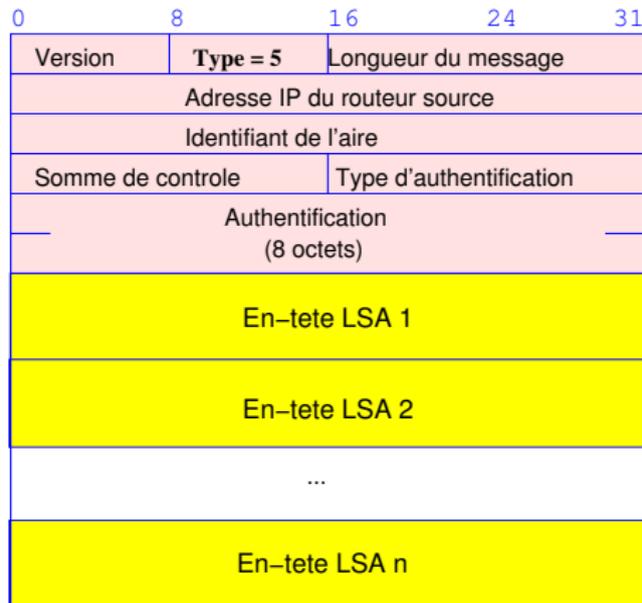
- description des liens par des LSA (en-tête et corps)



A.

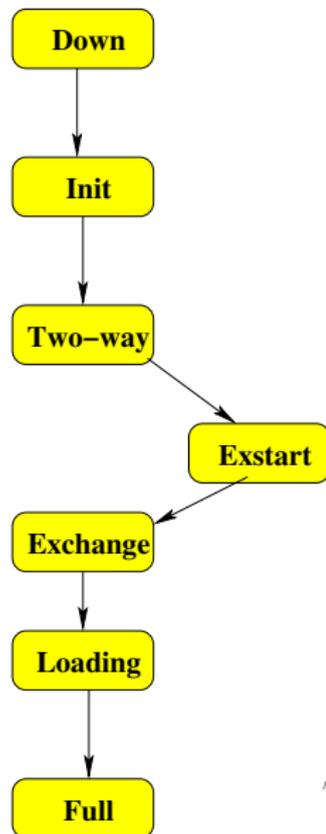
# Message OSPF accusé de réception d'états de liens (type 5) LSAck

- les en-tête LSA permettent d'accuser réception pour chaque LSA envoyé.



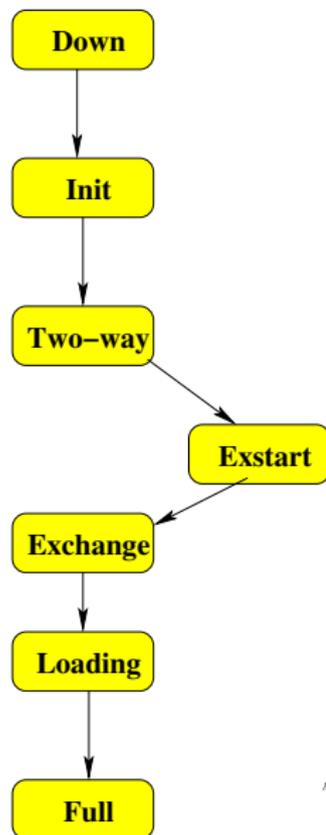
# Les 7 états OSPF

- **Down** Connectivité non assurée
  - Envoi de messages Hello annonçant son ID
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- Exstart
- Exchange Echanges de DBD coordonnées par le DR.
- Loading
- Full état terminal, routeurs en complète adjacence.



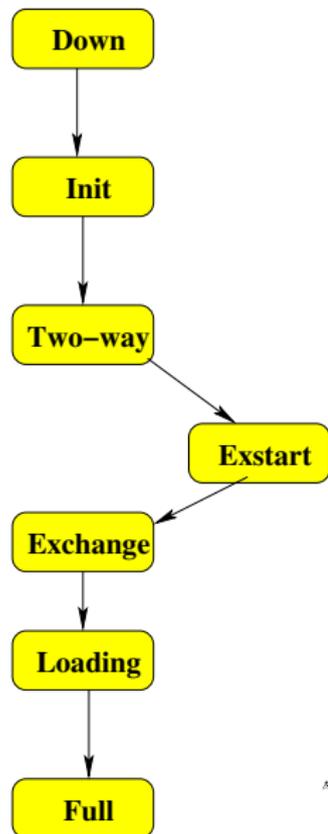
# Les 7 états OSPF

- Down Connectivité non assurée
- **Init A reçu son premier Hello (mais ne contenant pas son ID)**
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- Exstart
- Exchange Echanges de DBD coordonnées par le DR.
- Loading
- Full état terminal, routeurs en complète adjacence.



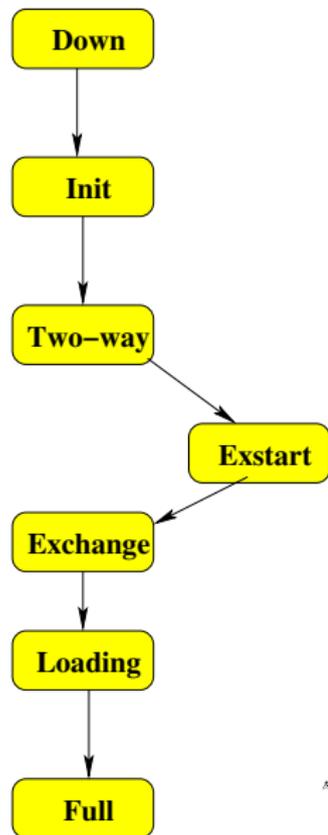
# Les 7 états OSPF

- Down Connectivité non assurée
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- **Two-way A reçu son premier Hello contenant son ID ⇒ connectivité dans les deux sens**
- Exstart
- Exchange Echanges de DBD coordonnées par le DR.
- Loading
- Full état terminal, routeurs en complète adjacence.



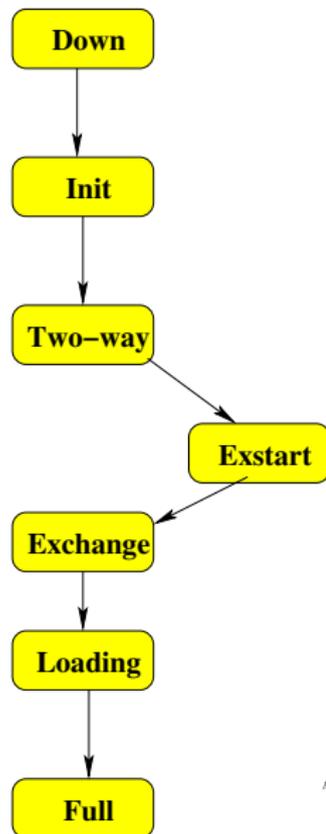
# Les 7 états OSPF

- Down Connectivité non assurée
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- **Exstart**
  - Si nécessaire, élection du DR (et du BDR) à l'aide de paquets HELLO.
- Exchange Echanges de DBD coordonnées par le DR.
- Loading
- Full état terminal, routeurs en complète adjacence.



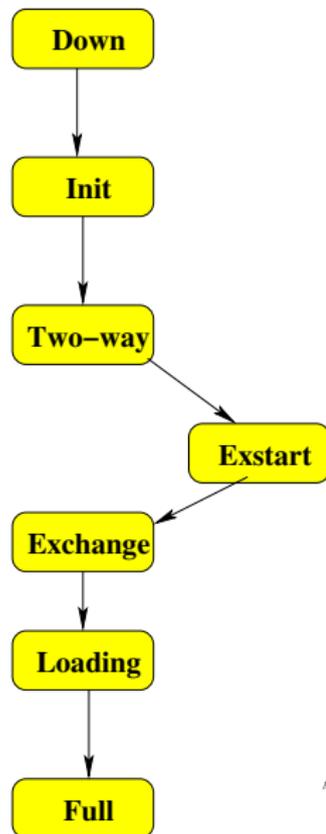
# Les 7 états OSPF

- Down Connectivité non assurée
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- Exstart
- **Exchange Echanges de DBD coordonnées par le DR.**
  - Les routeurs envoient des LSAck
  - Comparaison des DBD reçus avec leur DBD locale, si nouvelle route, passage en état "Loading" en envoyant un LSR.
- Loading
- Full état terminal, routeurs en complète adjacence.



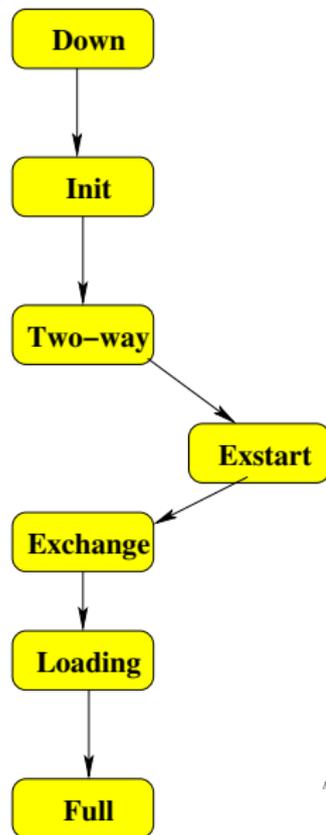
# Les 7 états OSPF

- Down Connectivité non assurée
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- Exstart
- Exchange Echanges de DBD coordonnées par le DR.
- **Loading**
  - le LSR a été envoyé.
  - Attente du LSU (contenant le LSA).
  - Acquiescement avec un LSAck
- Full état terminal, routeurs en complète adjacence.



# Les 7 états OSPF

- Down Connectivité non assurée
- Init A reçu son premier Hello (mais ne contenant pas son ID)
- Two-way A reçu son premier Hello contenant son ID  $\Rightarrow$  connectivité dans les deux sens
- Exstart
- Exchange Echanges de DBD coordonnées par le DR.
- Loading
- **Full état terminal, routeurs en complète adjacence.**
  - Les routeurs connaissent tous leurs routeurs voisins et l'état de toutes les liaisons du réseau.
  - Création de la table de routage (algorithme SPF).



# Étapes OSPF

- 1 **Etablir l'adjacence des routeurs (Hello) :**
- 2 **Election du DR et du BDR (si nécessaire) :** champ de priorité (0-255) dans paquet HELLO (et ID si égalité).
- 3 **Découvrir les routes :** échange de DBD : Type d'état de lien, les annonces d'adresses, le coût du lien, un nombre de séquence.  
Comparaisons des DBD reçus avec leur propres DBD. LSR+LSA dans LSU
- 4 **Selectionner les bonnes routes :**
- 5 **Maintenir les informations de routage :** Quand un changement survient, les routeurs utilisent le processus d'inondation (flooding) pour avertir leurs voisins sur le réseau.  
Si une ligne est down, le routeur envoie le nouvel état au DR ou BDR qui fera suivre aux autres routeurs (LSU+LSAck).

# Adresses multicast

224.0.0.1	tous les systèmes du réseau local
224.0.0.2	tous les routeurs du réseau local
224.0.0.5	tous les routeurs OSPF
224.0.0.6	tous les routeurs OSPF désignés (DR et BDR)
224.0.0.24	échange des descriptions de bases de données compatibles TE durant la synchronisation des BD (OSPF-IGP-TE) - expérimental.

# Protocole d'inondation (flooding)

- À chaque changement d'état d'un lien, le routeur qui en a la charge émet un LSU (contenant le LSA) vers les DR/BDR (224.0.0.6)
- le DR attribue un numéro de séquence et émet le LSU vers tous les routeurs (224.0.0.5).
- Chaque routeur recevant le LSA cherche l'entrée du LSA dans sa base par le numéro de séquence.
- Si LSA non présent ou si annonce plus récente
  - met à jour la base
  - retransmet le LSA sur toutes ses interfaces sauf celle par laquelle il a reçu l'annonce
  - acquitte le message (OSPF type 5)

# Maintien des informations de routage

Détection d'un changement sur un lien :

- disparition d'un lien : silence de la ligne. Toutes les 10 secondes par défaut, les routeurs envoient un HELLO. Si silence pendant 40 secondes, la ligne est considérée comme "down".
- (r)établissement d'un lien : un HELLO est reçu par le routeur sur ce lien.

Un routeur détecte un changement :

- un LSU est envoyé par flooding
- les routeurs recevant le LSU mettent à jour leur base de données d'état de lien (LSDB)
- les routeurs recalculent leur table de routage par l'algorithme SPF.

Si aucun changement d'état n'intervient dans le réseau, les infos seront quand même mises à jour périodiquement.

Chaque LSA reçu à une période d'existence (30 min par défaut chez Cisco).

# Algorithme Shortest Path First (SPF)

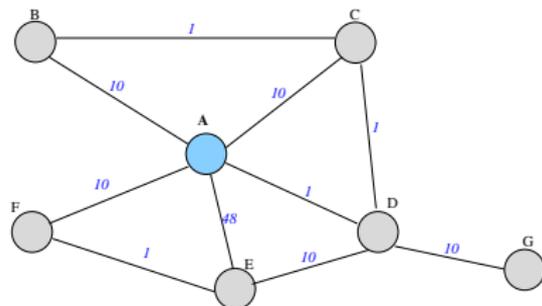
- Calcul des plus courts chemin dans un graphe.
- Algorithme de Dijkstra (plus court chemin et pas de cycles) pour déterminer les routes les moins couteuse.
- Chaque routeur se voit comme la racine d'un arbre contenant les meilleures routes.

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



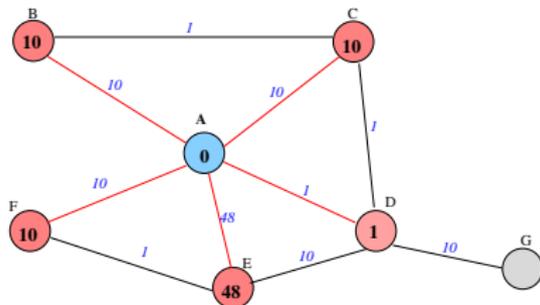
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



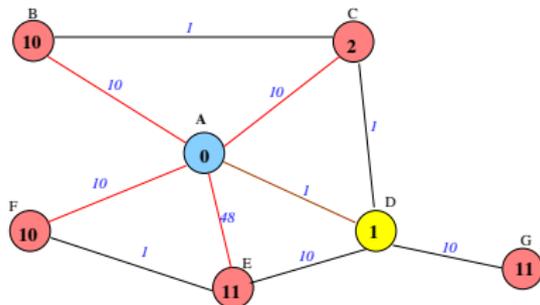
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



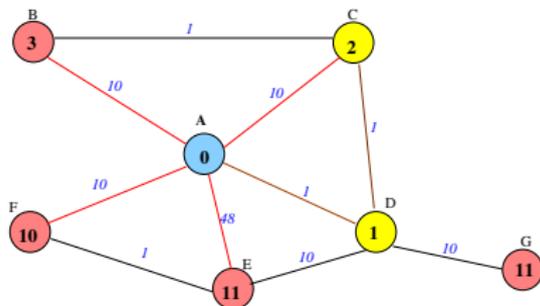
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



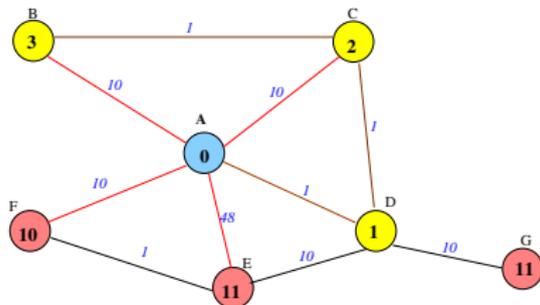
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



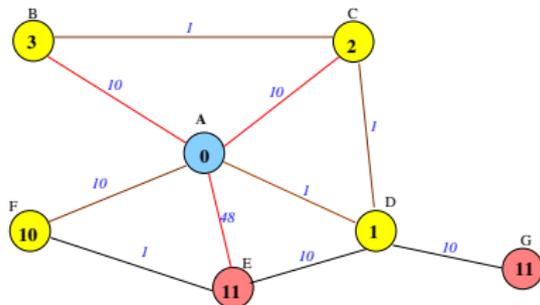
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



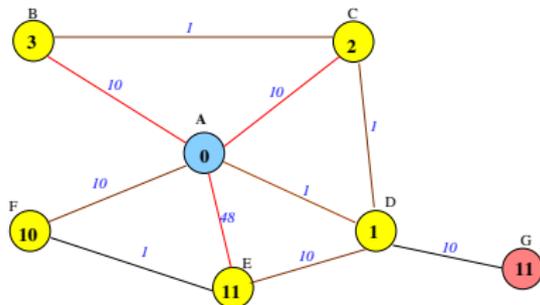
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



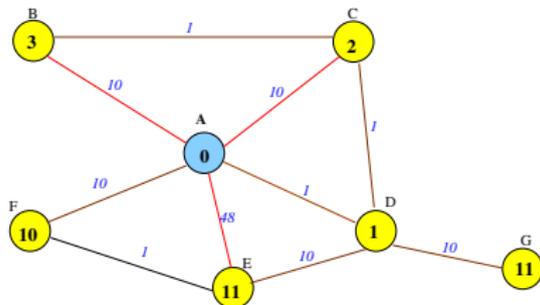
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



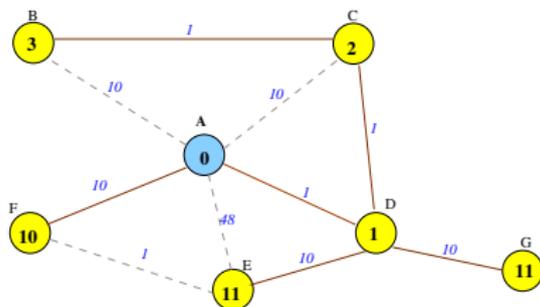
L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .



L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

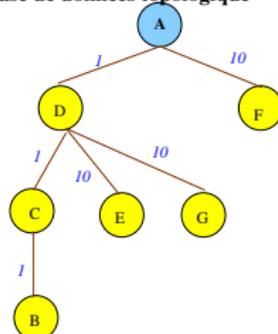
# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .

Base de données topologique



L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

# Algorithme Shortest Path First (SPF)

Init :  $P = r$ , et la distance de  $r$  à lui-même vaut zéro

A chaque étape :

- 1 Identifier toutes les arêtes  $a_i = (s_{i1}, s_{i2})$  dans  $P \times G$  tel que  $s_{i1}$  est dans  $P$  et  $s_{i2}$  est dans  $G$  ;
- 2 Choisir l'arête  $a_j = (s_{j1}, s_{j2})$  dans  $P \times G$  qui donne la distance minimum depuis  $r$  à  $s_{j2}$  en passant tous les chemins créés menant à  $s_{j2}$ . La placer dans  $P$ .

L'algorithme se termine soit quand  $P$  devient un arbre couvrant de  $G$

Base de données topologique

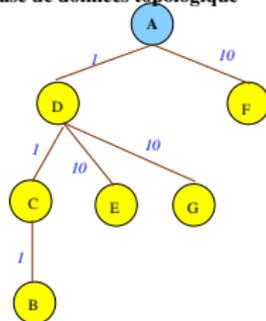


Table de routage

Pour aller vers le réseau	Passer par le routeur
réseau(x) attaché(s) à D	D
réseau(x) attaché(s) à C	D
réseau(x) attaché(s) à E	D
réseau(x) attaché(s) à G	D
réseau(x) attaché(s) à B	D
réseau(x) attaché(s) à F	F

# Coût

Par défaut, coûts utilisés en fonction de la bande passante du lien :

Type de réseau	Coût par défaut
FDDI, FastEthernet	1
Ethernet 10 Mbps	10
E1 (2,048 Mbps)	48
T1 (1,544 Mbps)	65
64 Kbps	1562
56 Kbps	1758
19.2 Kbps	5208

Suivant la formule :

$$cout = \frac{bande\_passante\_de\_reference\_en\_bps}{bande\_passante\_du\_lien\_en\_bps}$$

(par défaut,  $bande\_passante\_de\_reference\_en\_bps = 100Mbps$ )

# Synthèse OSPF

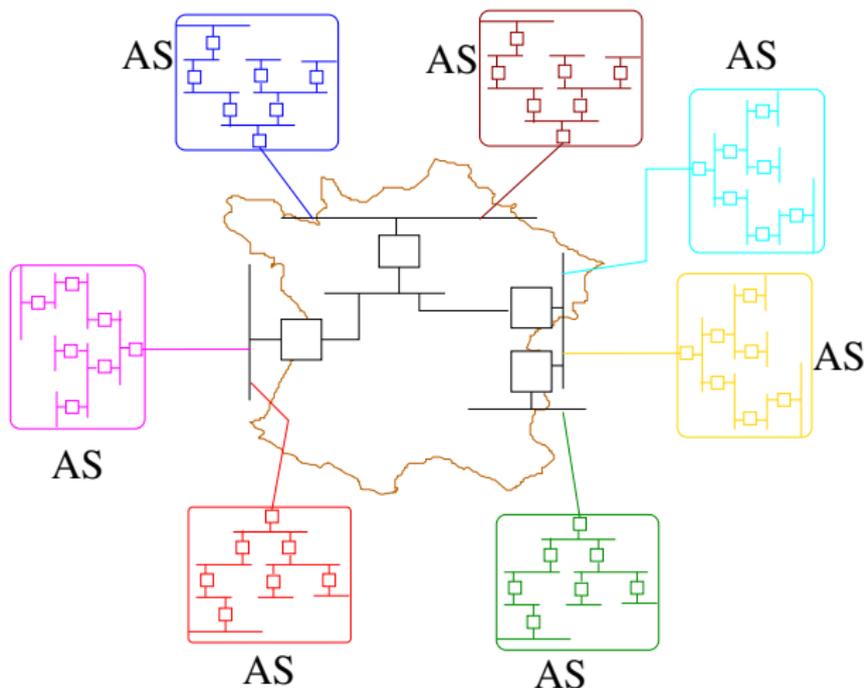
- Routage à état de lien (*Link-State*) : permettre au routeur d'avoir une vision globale du réseau et de sa topologie
- OSPF gère les limitations de RIP
- s'applique sur de très larges réseaux utilisant une architecture hiérarchique.
- mises à jour sont non périodiques et déclenchées sur des changements de topologie, ce qui entraîne un faible temps de convergence des tables de routage.

Protocole à état de lien recommandé pour remplacer RIP

- plus fiable
- hiérarchisé
- authentification
- équilibrage de charge

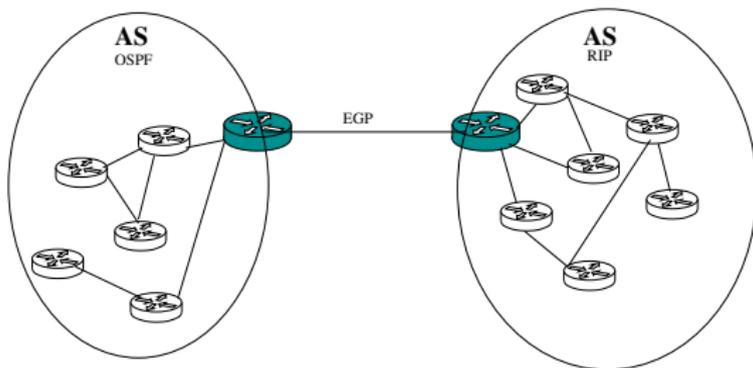
- 1 Routage statique
- 2 Principes généraux du routage dynamique
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF
- 5 Protocoles de routage externe (EGP) : BGP**
- 6 Références bibliographiques

# EGP (External Gateway Protocol)



# EGP (External Gateway Protocol)

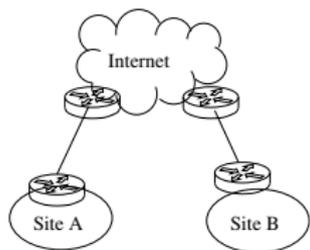
Routage inter-domaine.



Problèmes techniques :

- les métriques sont différentes suivant les protocoles internes aux AS.

# EGP (External Gateway Protocol)



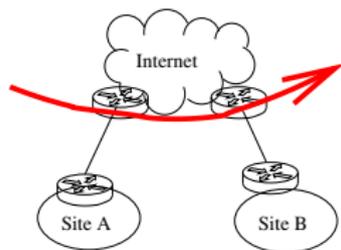
Problèmes politiques :

- s'utilise entre entités distinctes (souvent concurrentes).
- Impossibilité de prendre une décision qui s'imposera à tous.
- On n'est pas prévenu de ce que vont faire les autres.
- Idée de méfiance : le but n'est pas de trouver la meilleure route mais au contraire d'empêcher les routeurs de choisir une route dont on ne voudrait pas.
- Politique sur le trafic de transit

# EGP (External Gateway Protocol)

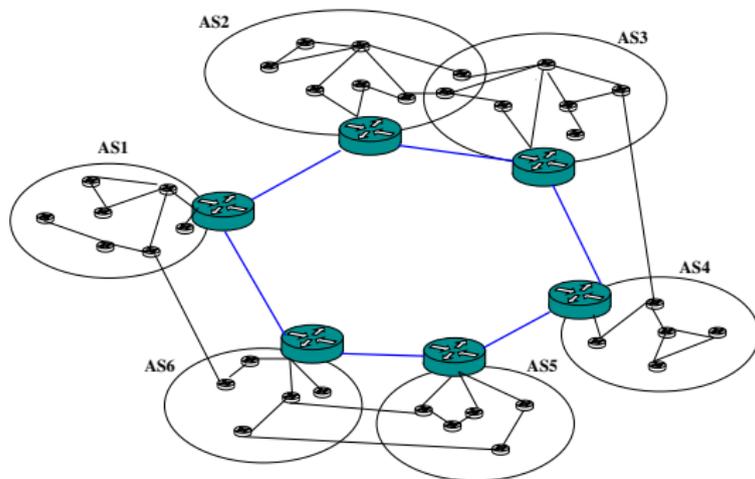
## Problèmes politiques :

- s'utilise entre entités distinctes (souvent concurrentes).
- Impossibilité de prendre une décision qui s'imposera à tous.
- On n'est pas prévenu de ce que vont faire les autres.
- Idée de méfiance : le but n'est pas de trouver la meilleure route mais au contraire d'empêcher les routeurs de choisir une route dont on ne voudrait pas.
- Politique sur le trafic de transit



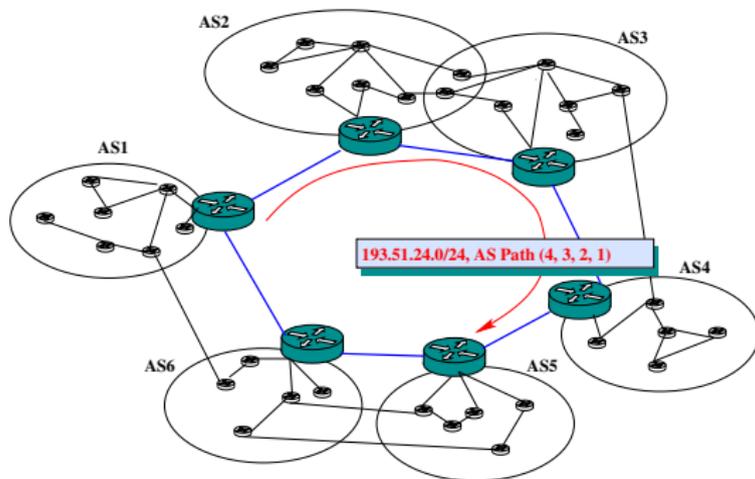
# BGP (Border Gateway Protocol)

- protocole standard de l'Internet pour les interconnexions entre opérateurs.
- on doit annoncer exactement le préfixe réseau que l'on veut diffuser.



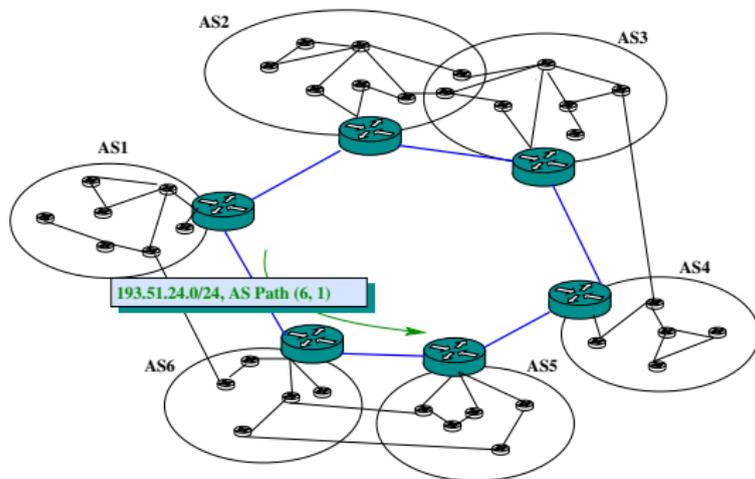
# BGP (Border Gateway Protocol)

- protocole standard de l'Internet pour les interconnexions entre opérateurs.
- on doit annoncer exactement le préfixe réseau que l'on veut diffuser.



# BGP (Border Gateway Protocol)

- protocole standard de l'Internet pour les interconnexions entre opérateurs.
- on doit annoncer exactement le préfixe réseau que l'on veut diffuser.



# BGP (Border Gateway Protocol)

- protocole point à point entre routeurs de bords de l'AS
  - établissement de session BGP (TCP port 179)
  - échange de message BGP
  - un routeur peut participer à plusieurs sessions BGP
- routage par vecteur de chemin (et pas par vecteur de distance, ni par état de lien)
- un noeud n'a pas à accepter une route apprise par ses voisins, il peut l'accepter ou la refuser.
- un noeud BGP annonce une partie de sa table de routage
- ce qui est partagé et accepté dépend de la politique de routage

# BGP (Border Gateway Protocol)

Protocole à vecteur de chemin

- trouver des chemins sans cycle entre les AS
- trouver un chemin optimal n'est pas un but en soi : ce n'est ni un protocole par vecteur de distance ni par état de lien.

Problèmes :

- pas de routage par défaut!!!
- Environ 12.000 AS sur l'Internet, soit de très grosses tables dans les routeurs BGP
- Besoin de flexibilité

# CIDR et longest prefix match

CIDR : utilisation de préfixes de longueur variable.

- Pour l'instant les tables BGP de l'Internet comportent un peu plus de 90.000 routes.
- CIDR utilisé pour réduire les tables de routage (supernetting).
- Possibilité de recouvrement des préfixes

Il est choisi le préfixe de longueur maximale pour router. Exemple : Un routeur entend :

- R1 annonce 193.51.0.0/16 et
- R2 annonce 193.51.24.0/24

Pour router :

- 193.51.24.3  $\Rightarrow$  R2 est choisit
- 193.51.33.14  $\Rightarrow$  R1 est choisit

# BGP 4

BGP est utilisé pour transporter des informations de routage entre AS :

- numéro d'AS
- liste des réseaux de chaque AS
- distance vers les sous-réseau de l'AS
- IP du routeur d'entrée vers les sous-réseaux.

Protocole de transmission fiable (TCP sur port 179). Messages échangés :

- Message d'ouverture (numéro d'AS) entre deux routeurs
- Message de mise à jour : signale chaque changement d'état et les routes inaccessibles
- Message de notification : motif de la fermeture
- Message Hello keepalive : pour signaler que le routeur est toujours vivant.

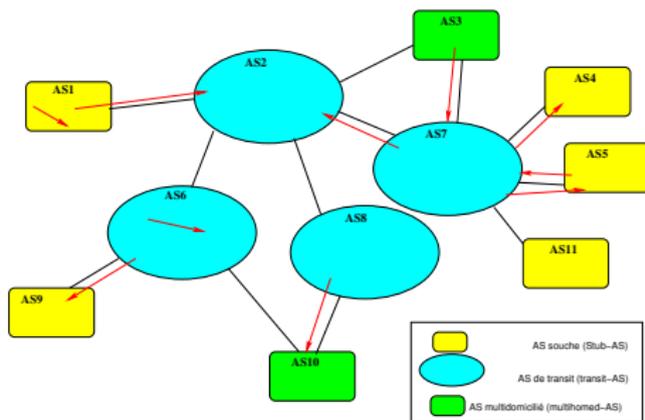
# Domaine de transit et domaine souche

## Type de trafic

- **local source ou destination au sein de l'AS**
- transit trafic passant à travers un AS.

## Type d'AS

- **AS souche (stub AS)** : connecté à un seul AS. Transporte que du trafic local.
- **AS multi-domicilié** : AS connecté à plus d'un AS. Ne transporte pas de trafic de transit.
- **AS transit** : AS connecté à plus d'un AS et transportant du trafic de transit.



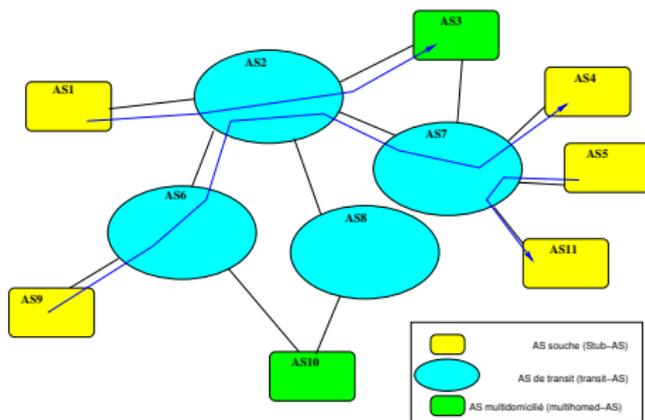
# Domaine de transit et domaine souche

## Type de trafic

- local source ou destination au sein de l'AS
- **transit trafic passant à travers un AS.**

## Type d'AS

- **AS souche (stub AS)** : connecté à un seul AS. Transporte que du trafic local.
- **AS multi-domicilié** : AS connecté à plus d'un AS. Ne transporte pas de trafic de transit.
- **AS transit** : AS connecté à plus d'un AS et transportant du trafic de transit.



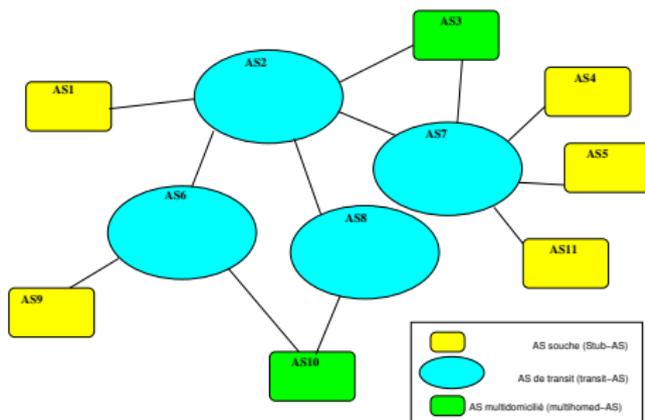
# Domaine de transit et domaine souche

## Type de trafic

- local source ou destination au sein de l'AS
- transit trafic passant à travers un AS.

## Type d'AS

- **AS souche (stub AS)** : connecté à un seul AS. Transporte que du trafic local.
- **AS multi-domicilié** : AS connecté à plus d'un AS. Ne transporte pas de trafic de transit.
- **AS transit** : AS connecté à plus d'un AS et transportant du trafic de transit.



# AS et BGP

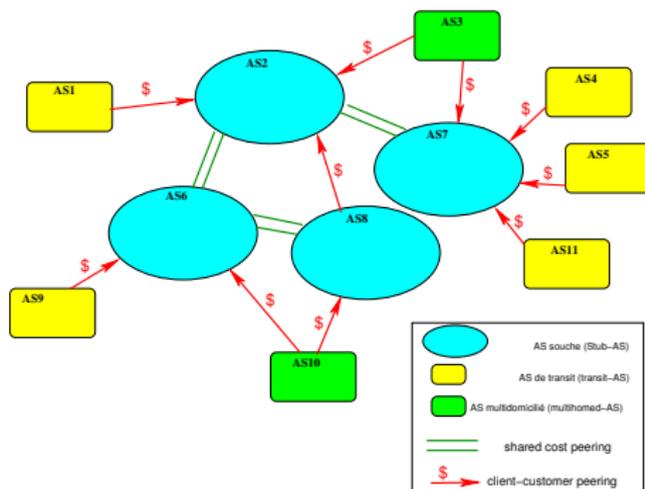
Chaque AS a

- un ou plusieurs border router gérant le trafic inter
- un BGP speaker (pour les AS participant au routage)
- un speaker BGP établit des sessions avec ses pairs et annonce :
  - les réseaux locaux
  - les autres réseaux accessibles (pour les AS de transit)
  - donne des informations sur les chemins (poids)
  - les routes supprimées

# Peering BGP

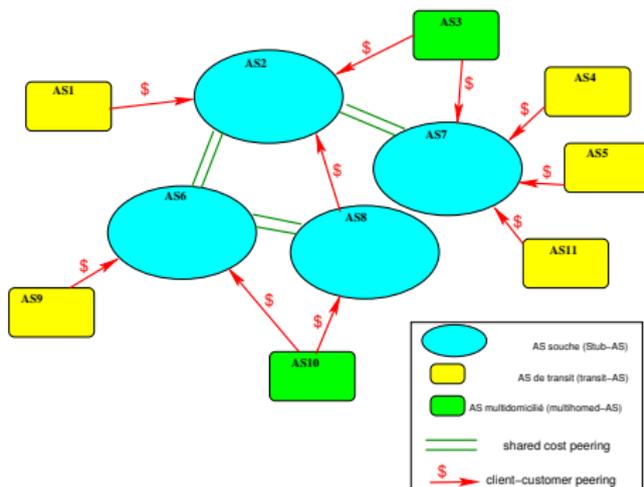
## 2 types de peering

- **client-fournisseur (customer-provider peering)** : Relation asymétrique dans laquelle un client (un domaine de routage) achète une connectivité à l'Internet auprès d'un fournisseur d'accès (un autre domaine de routage).
- **pair-à-pair (shared-cost peering)** : Relation symétrique où deux domaines de routage (souvent de même importance) acceptent d'échanger gratuitement leurs paquets à travers un point d'interconnexion.



# Peering BGP

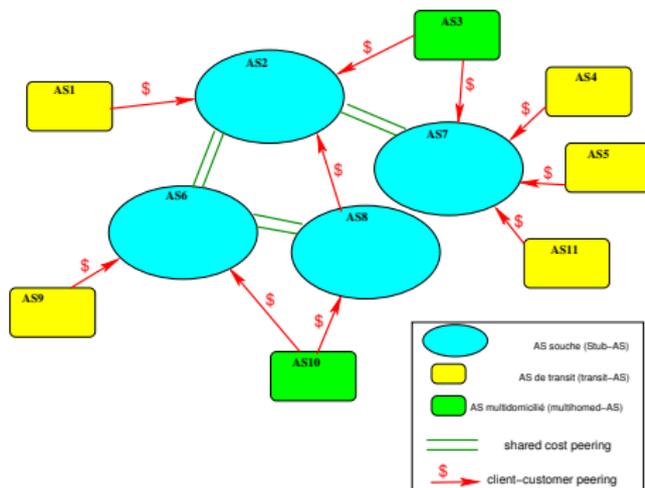
- **client-fournisseur**  
(customer-provider peering) :  
Relation asymétrique dans laquelle un client (un domaine de routage) achète une connectivité à l'Internet auprès d'un fournisseur d'accès (un autre domaine de routage).



- Le client envoie ses routes internes et les routes apprises de ses propres clients au fournisseur  
⇒ Le fournisseur annoncera ces routes sur tout l'Internet
- Le fournisseur annonce à son client toutes les routes qu'il connaît  
⇒ le client est capable d'atteindre n'importe qui sur l'Internet

# Peering BGP

- **pair-à-pair (shared-cost peering)** : Relation symétrique où deux domaines de routage (souvent de même importance) acceptent d'échanger gratuitement leurs paquets à travers un point d'interconnexion.



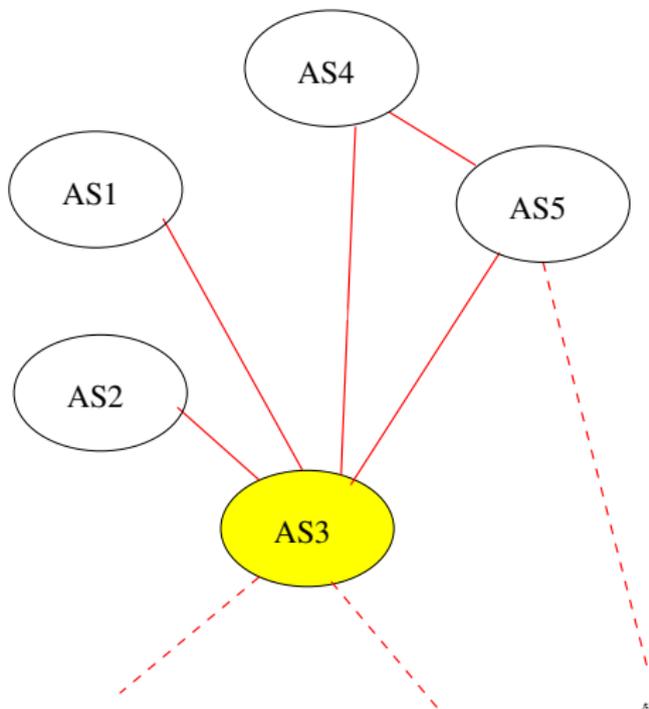
- Chaque "peer" envoie à l'autre ses propres routes et celles de ses clients
- Le point d'interconnexion sera utilisé par l'un des pair BGP pour atteindre les destinations de l'autre pair (ou de ses clients) ⇒ échange de trafic local

A.

# Interconnexion d'AS

Les routeurs des AS sont connectés :

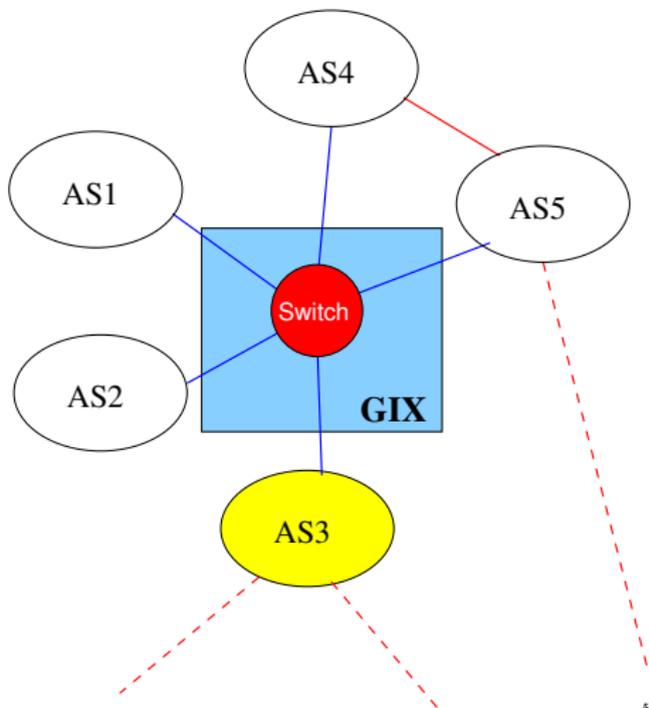
- par liaison point-à-point
  - ⇒ Seul matériel commun : la liaison.
  - ⇒ Cher si beaucoup d'AS voisins géographiquement
- par point d'échange GIX (Global Internet eXchange) également appelé IXP (Internet Exchange Point)



# Interconnexion d'AS

Les routeurs des AS sont connectés :

- par liaison point-à-point
  - ⇒ Seul matériel commun : la liaison.
  - ⇒ Cher si beaucoup d'AS voisins géographiquement
- par point d'échange GIX (Global Internet eXchange) également appelé IXP (Internet Exchange Point)



# Messages et déroulement du protocole BGP

- ① **open (1)** : Chaque routeur BGP échange avec ses voisins des messages pour ouvrir et négocier les paramètres la session BGP. Initialement les routeurs BGP échangent la totalité des information de routage.
- ② **update (2)** : seules les modifications sont transmises. Un numéro est associé à chaque version des informations collectées par un routeur. Tous les voisins BGP doivent avoir le même numéro. Ce numéro est modifié à chaque mise à jour.
- ③ **Keepalive (4)** : transmis périodiquement pour vérifier le bon fonctionnement de la session BGP.
- ④ **Notification" (3)** : messages spéciaux utilisés pour informer les voisins BGP des erreurs et des cas spéciaux.

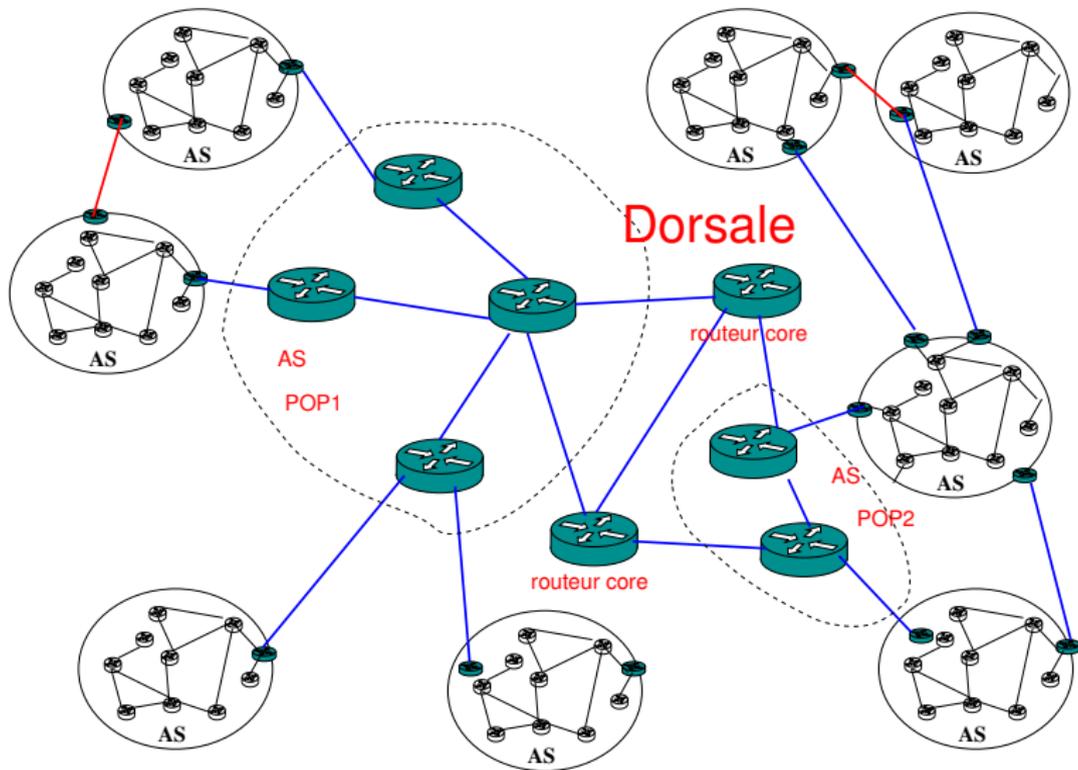
# Informations échangées par les routeurs BGP

Tableau d'accessibilité de préfixes IP (destinations). Pour chacun des préfixes :

- **AS\_path** : chemin d'AS sans boucle suivi pour atteindre la destination.
- **Next hop** : Le prochain saut pour atteindre le réseau
- **Poids (Weight)** :
- **Préférences locales** : pour influencer le processus de sélection du meilleur chemin. Interpretation locale à l'AS.
- **Multi-exit discriminator** : pour informer une préférence relatives entre différents points d'entrées.
- **Communauté (Community)** : regroupement de destination identifiée par un numéro.
- **Origine** :
- ... :

Ces informations permettent de construire un graphe formé d'AS (sans boucle) sur lequel une politique de routage peut être appliquée pour contraindre certains chemins.

# Dorsales paires



# En pratique

- chaque grand FAI est un AS
- entre 2 AS, accord d'échange de trafic entre 2 FAI : entre GIX (Internet Exchange Point) ou ligne privée louée
- communication avec ses pairs par BGP
- validation des annonces : Serveurs de routes IRR (Internet Routing Registry) infos relatives aux blocs détenus par chaque FAI

BGP est le coeur de l'Internet

- problème de validation des données des serveurs de routes.
- problème de routage temporaire, trous noirs de l'Internet
- beaucoup de mise à jour BGP chaque jour !
- trop trop trop grosses table de routage à maintenir (et ça augmente !)

5.

- 1 Routage statique
- 2 Principes généraux du routage dynamique
- 3 Protocoles de routage interne (IGP) : RIP
- 4 Protocoles de routage interne (IGP) : OSPF
- 5 Protocoles de routage externe (EGP) : BGP
- 6 Références bibliographiques

- «TCP/IP : Architectures, protocoles et applications » par Douglas Comer aux éditions Pearson education , 5ème édition.
- [http ://www-lsr.imag.fr/users/Martin.Heusse/](http://www-lsr.imag.fr/users/Martin.Heusse/)
- le support de cours de Bernard Cousin sur BGP : le routage inter-domaine
- le support de cours de Jean-Jacques Pansiot sur le routage inter-domaine et BGP
- « Réseaux et télécoms - cours avec 129 exercices corrigés » par Claude Servin aux éditions Dunod ; 2ème édition

- RFC 2453 : RIPv2
- RFC 2080 : RIPvng
- RFC 2328 : OSPFv2
- RFC 3630 : OSPF-TE
- RFC 1584 : Multicast OSPF
- RFC 4203 : OSPF pour MPLS
- RFC 5340 : OSPF for IPV6 (OSPFv3)
- RFC 4271 : BGP4