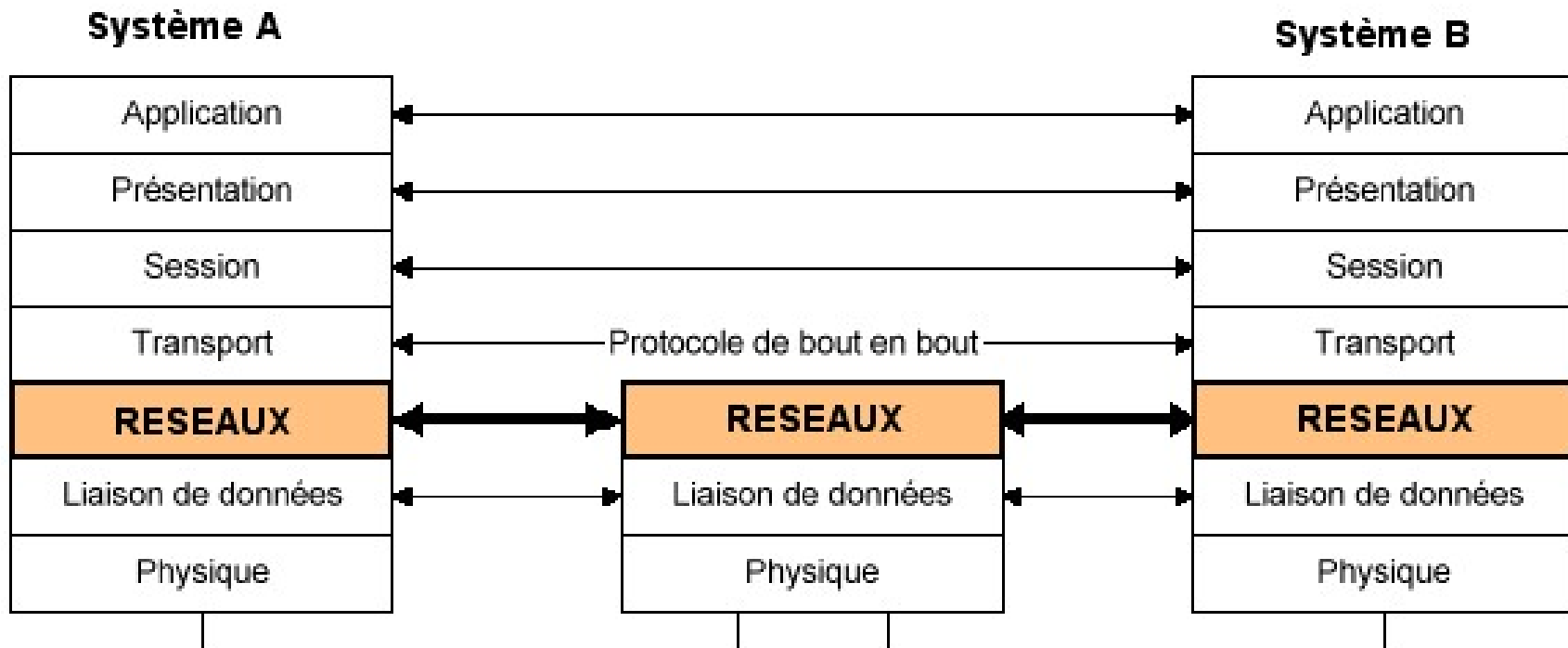


# **Internet et Routage**

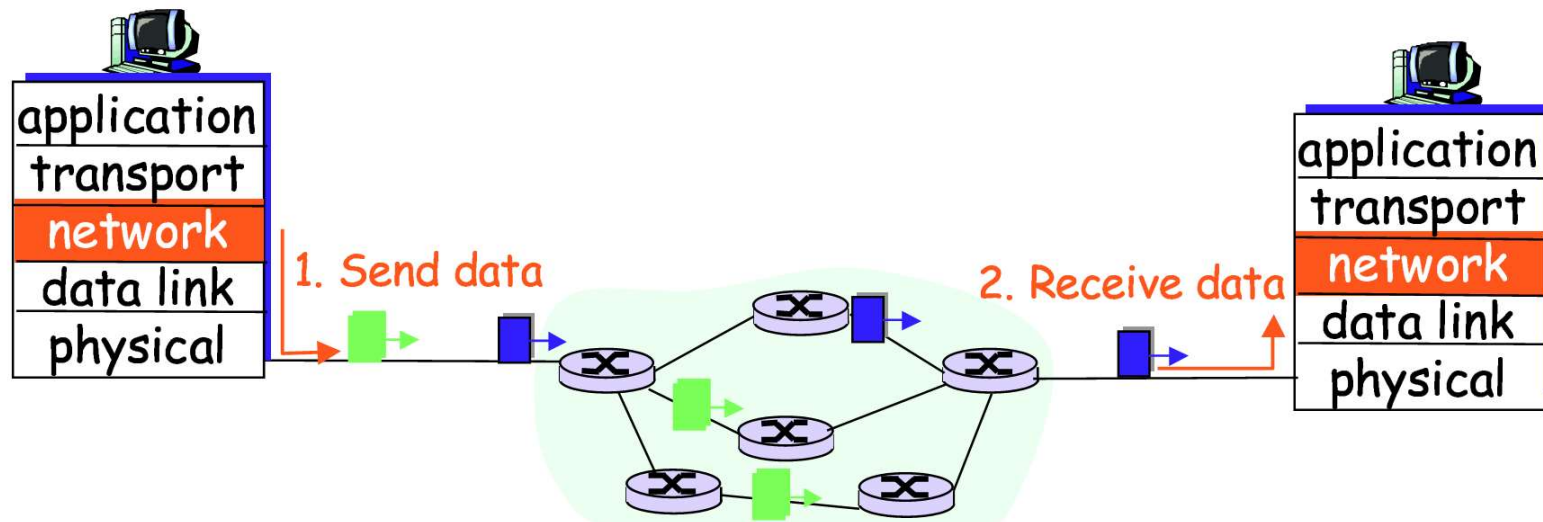
# Couche réseau



# Routage Internet

## Le mode datagramme

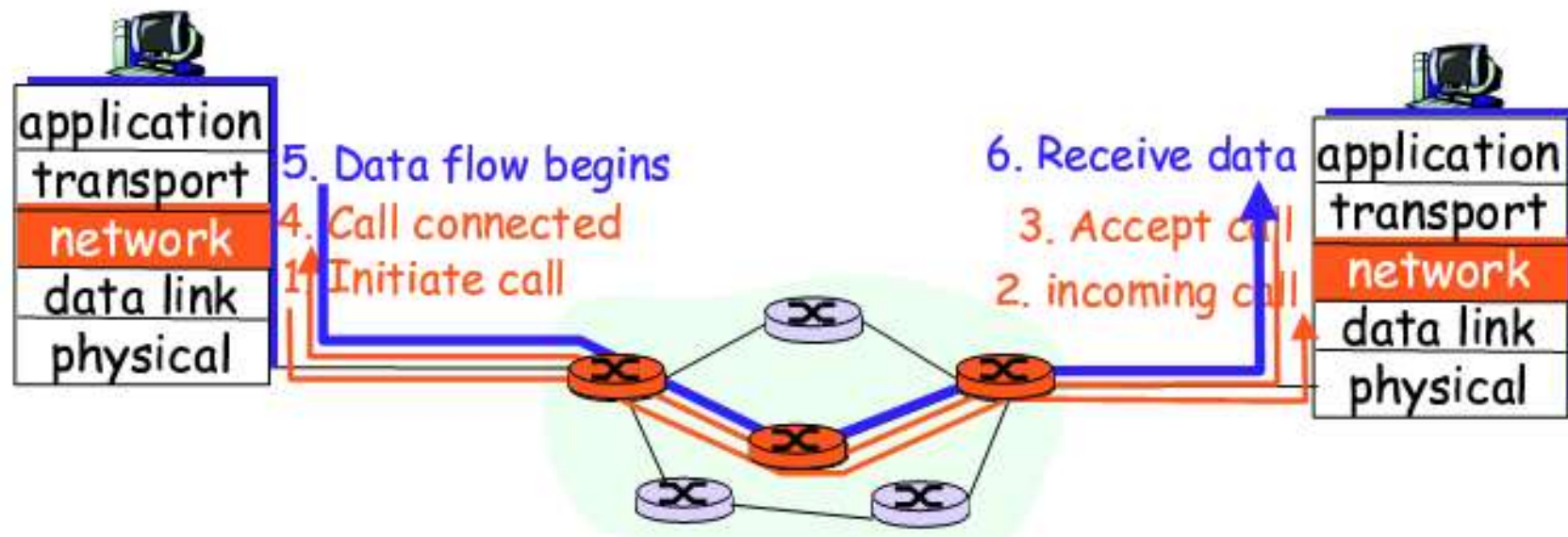
- ❑ Pas de mise en place de circuit
- ❑ routeurs: aucun état mémorisé au sujet des connexions
  - Pas de notion de connexion au niveau réseau
- ❑ Les paquets sont typiquement routé en fonction de l'adresse de destination
  - Des paquets avec la même source et destination peuvent suivre des trajets différents



# Routage Internet

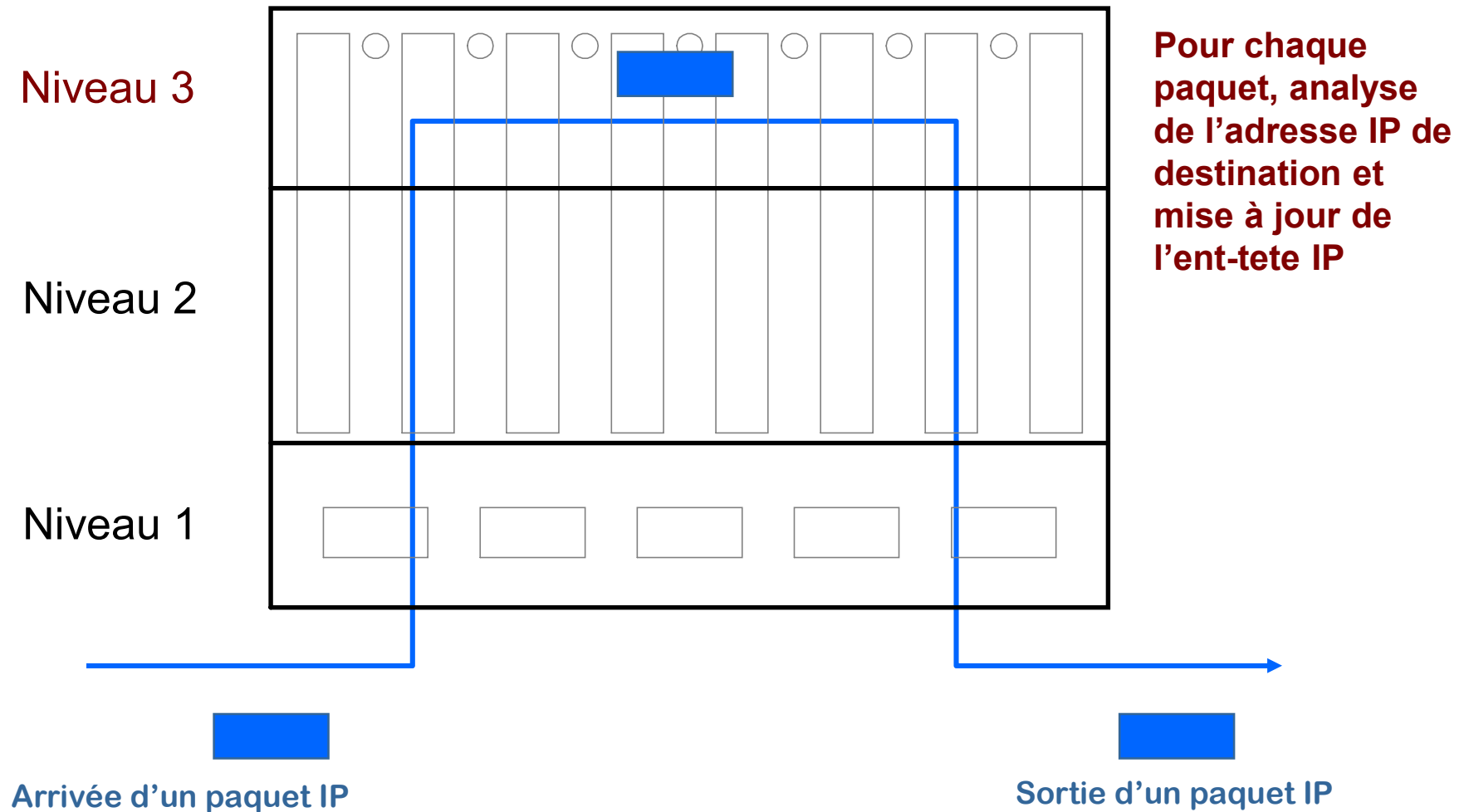
## Le mode Commutation de circuit

- ❑ Utilisés pour mettre en place et gérer un circuit virtuel
- ❑ utilisés dans ATM, frame-relay et X.25
- ❑ Ne sont pas utilisés (du moins de façon visible) dans l'Internet actuel



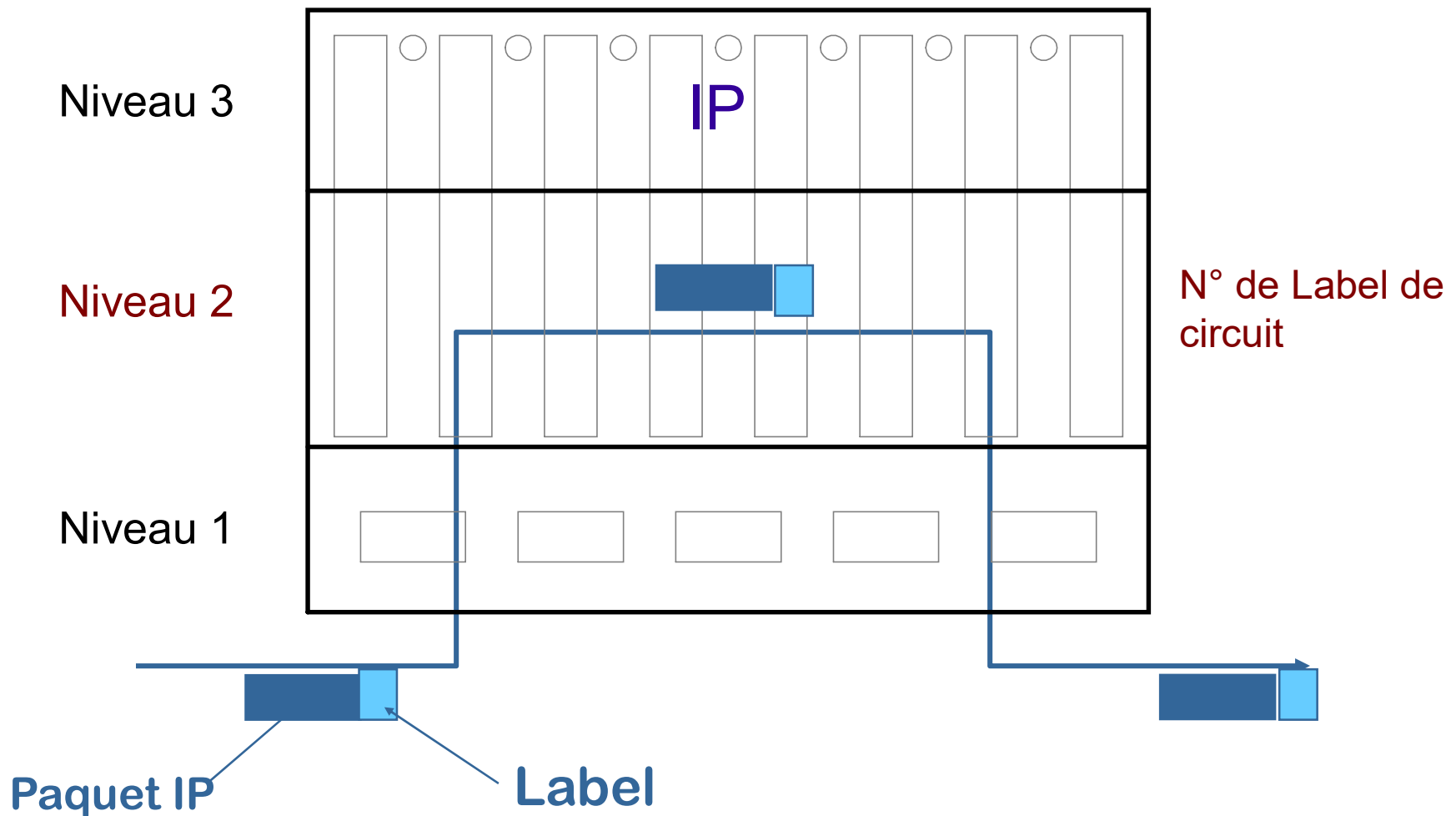
# Mode datagramme

## Routage IP



# Mode circuit virtuel

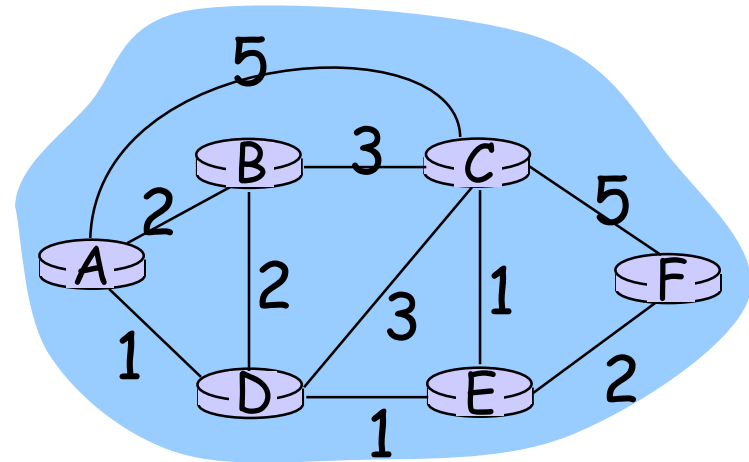
## Commutation IP par Label



# Routage: définition

**Objectif:** determine  
"meilleur" chemin  
(sequence des routeurs) à  
travers le réseau de la  
source à la destination.

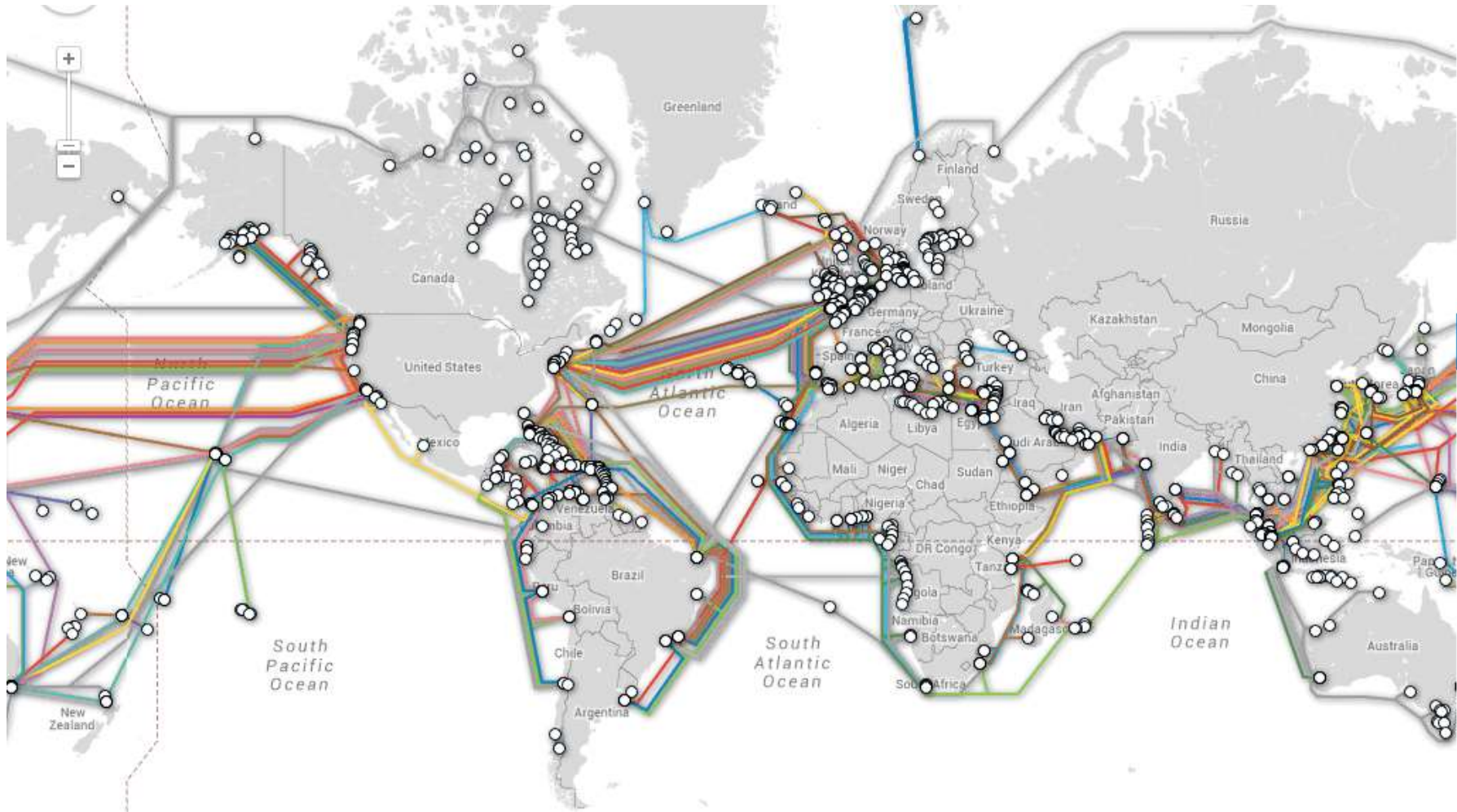
- ❑ Utilisation du Graphe  
par les algorithmes de  
routage:
- ❑ noeuds de graphe sont  
des routeurs
- ❑ Arcs de graphe sont  
des liens physique
  - poids: délai, nombre de  
sauts, debits, ...



- ❑ "meilleur" chemin:
  - Typiquement un chemin  
à coût minimum
  - autre déf possible

# Interconnexion Internet

## Cables optiques sous-marins

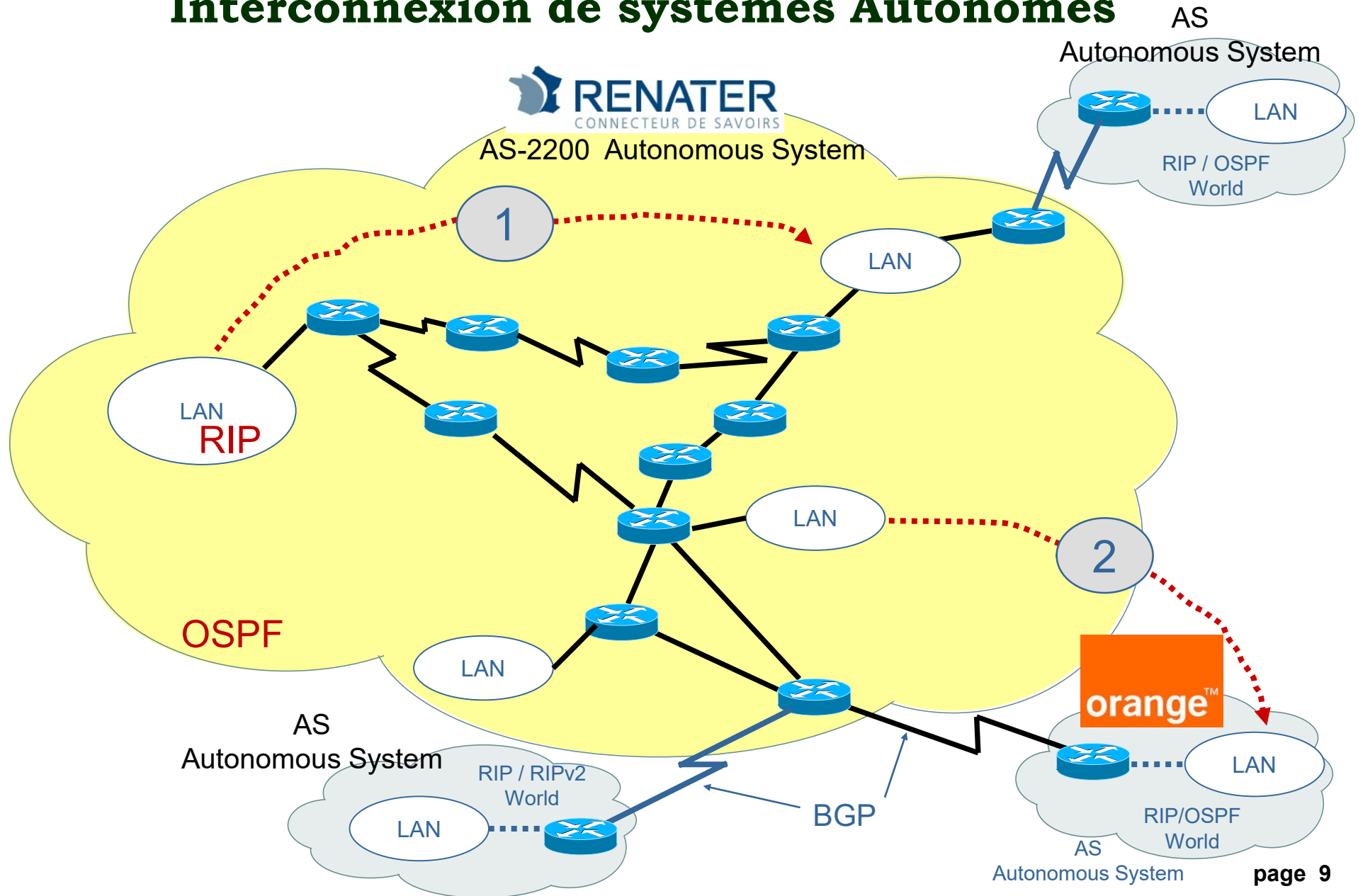


[www.submarinecablemap.com](http://www.submarinecablemap.com)  
100 Gbps par fibre



# Structure topologique de l'Internet

## Interconnexion de systemes Autonomes



# Autonomous Systems

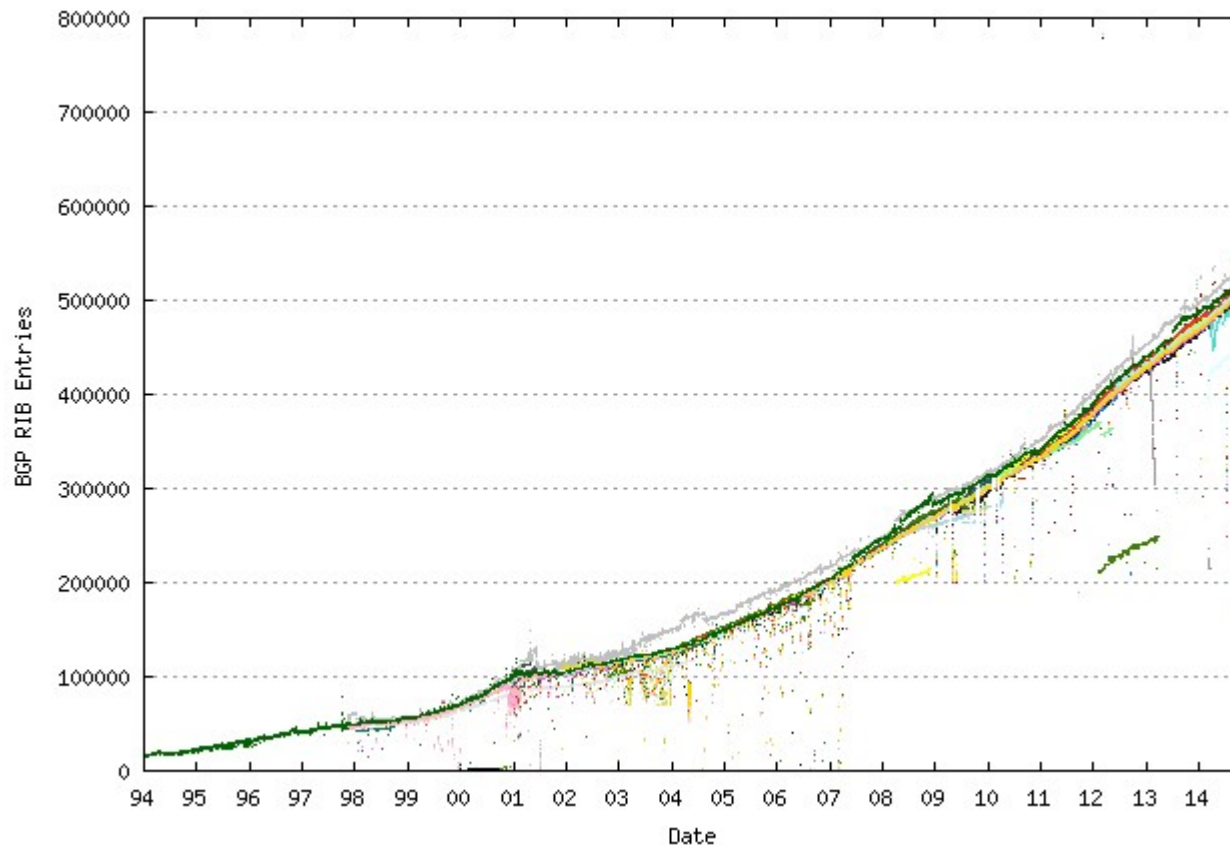
- Internet est le résultat du regroupement de **Systèmes Autonomes (AS)** interconnectés.
- un Système Autonome est "un groupe de réseaux IP possédant une *politique de routage propre et indépendante*, et qui réalise sa *propre gestion du trafic* entre celui-ci et le reste des Systèmes Autonomes qui font partie d'Internet.
- Un AS est identifié par un numéro sur 32 bits obtenu auprès des agences RIR (ICANN)
- Les **points d'échanges** (IXP's pour Internet Exchange Point) sont des associations où les opérateurs et les fournisseurs Internet se réunissent pour connecter directement leurs réseaux et échanger leur trafic Internet.

New membership receiving the following resources			Setup fee (one time)	Annual fee (paid in advance)
IPv4	IPv6	ASN		
/24 (256 addresses)	/48	1	500 AUD	1,050 AUD
/23 (512 addresses)	/48	1	500 AUD	1,373 AUD
/22 (1,024 addresses)	/32	1	500 AUD	1,796 AUD
/21 (2,048 addresses)	/32	1	500 AUD	2,350 AUD

# Statistiques de l'Internet

Nov. 2014:

- 48 802 AS (Autonomous Systems)
- 526 595 routes globales BGP
- Dernieres adresses IPv4 distribuées aux RIR en sept. 2011 par IANA



# Structure d'un réseau d'opérateur (un Système Autonome)

## □ POP

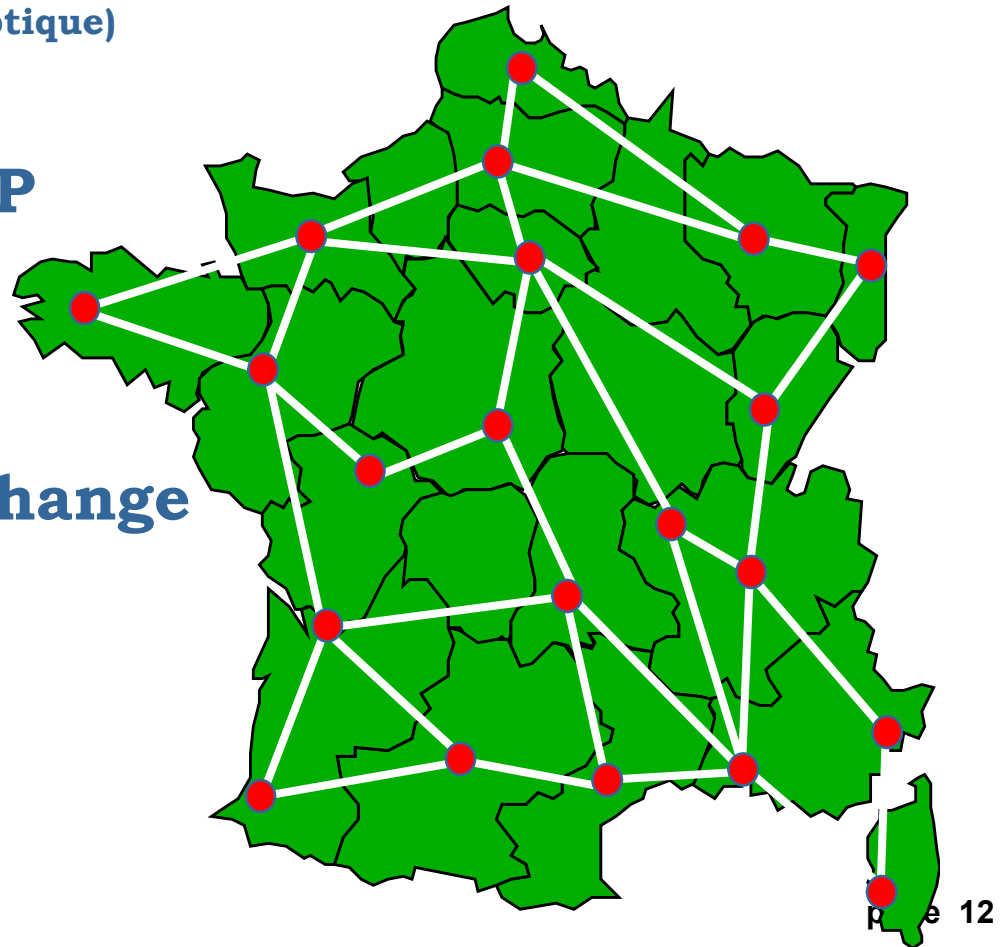
- Points de(Of) Présence (équipements commutateur, routeur, multiplexeur)
- Raccordement des clients (ISP, entreprises, particuliers) sur les POP
- Via boucles locales (cuivre, fibre optique)

## □ Interconnexion des POP

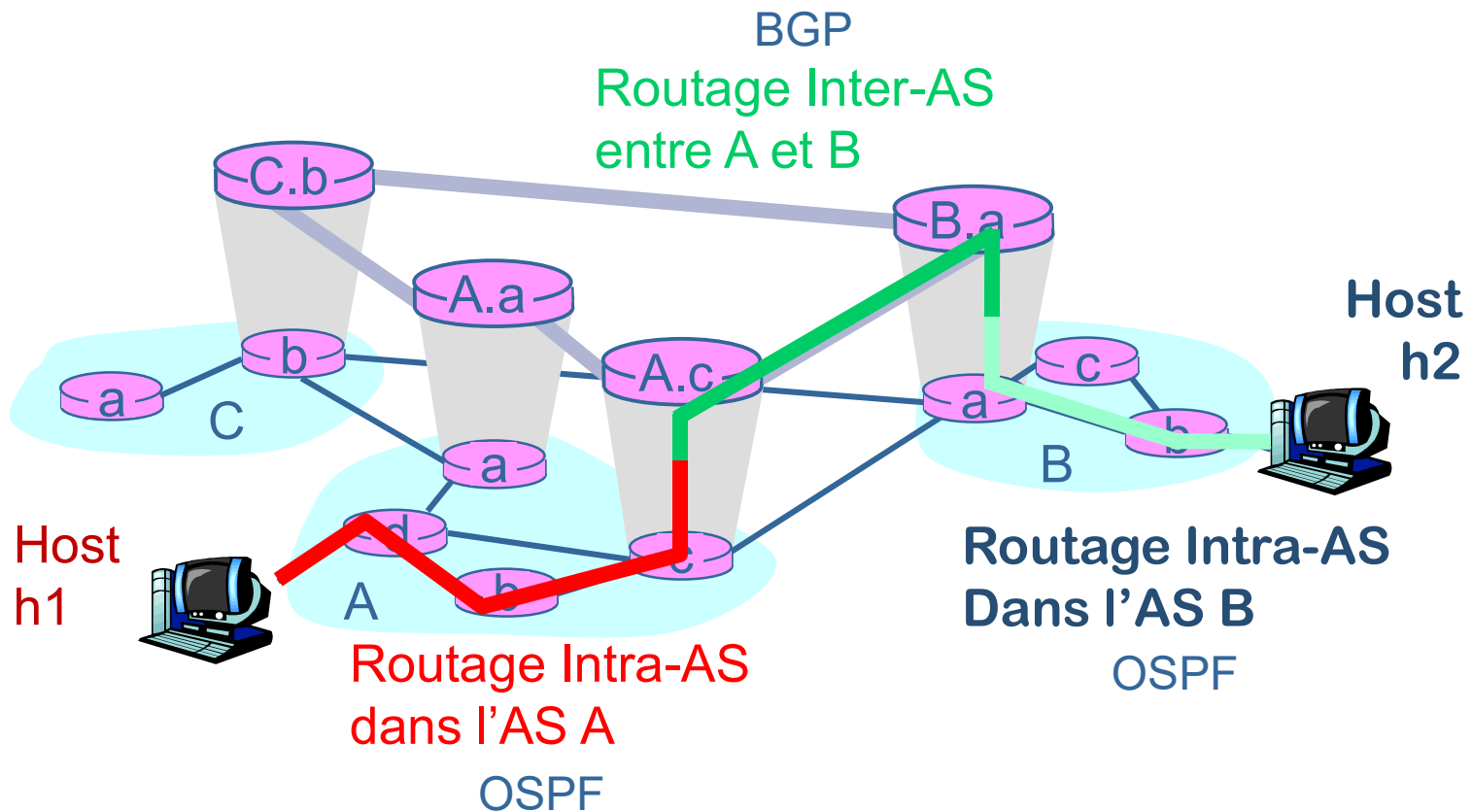
- Réseau maillé
- Fibres optiques
- Routage IP OSPF (intra AS)

## □ IPX Internet Point Exchange

- POPs dédiés
- Interconnexion entre opérateurs
- Fibres optiques
- Routage IP BGP (inter AS)



# Routage Intra-AS et Inter-AS



# Routeurs IP (opérateurs)



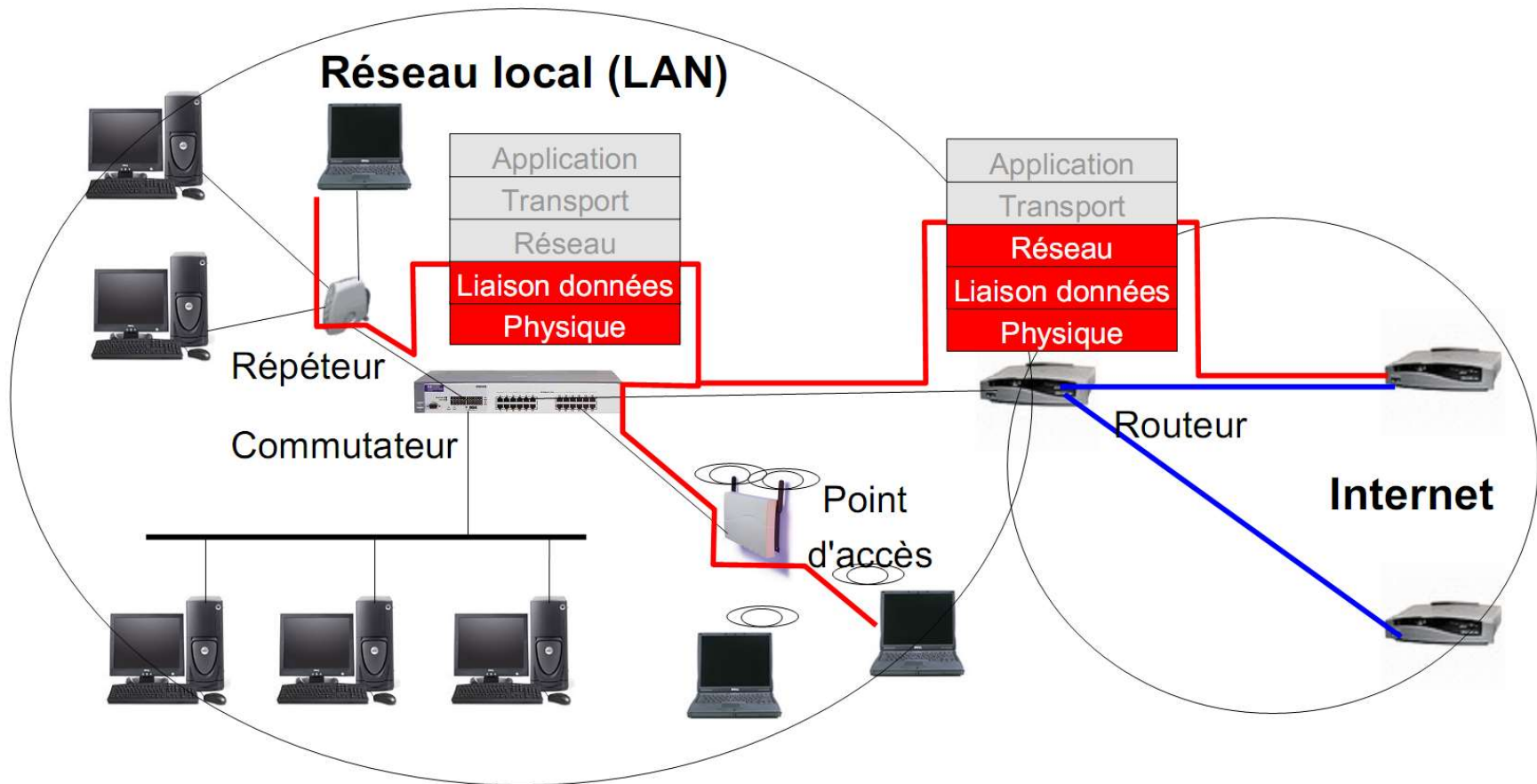
NOKIA

Alcatel-Lucent



# Routage IP dans les Intranet d'entreprises

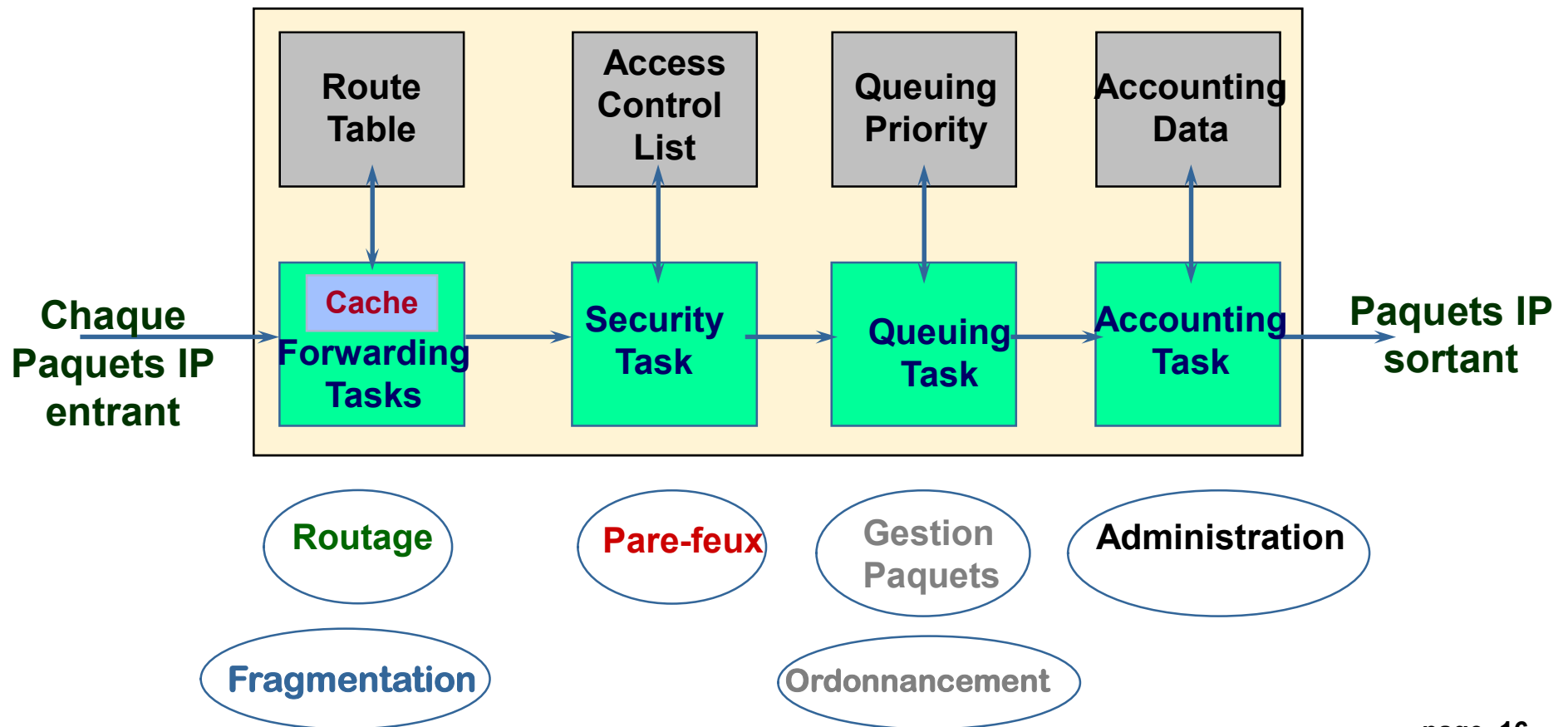
## Passerelle IP (Gateway IP)



Peu de routeurs (<10)  
Protocole de routage **RIP**



# Fonctions d'une Passerelle IP





# Tâches d'une passerelle IP

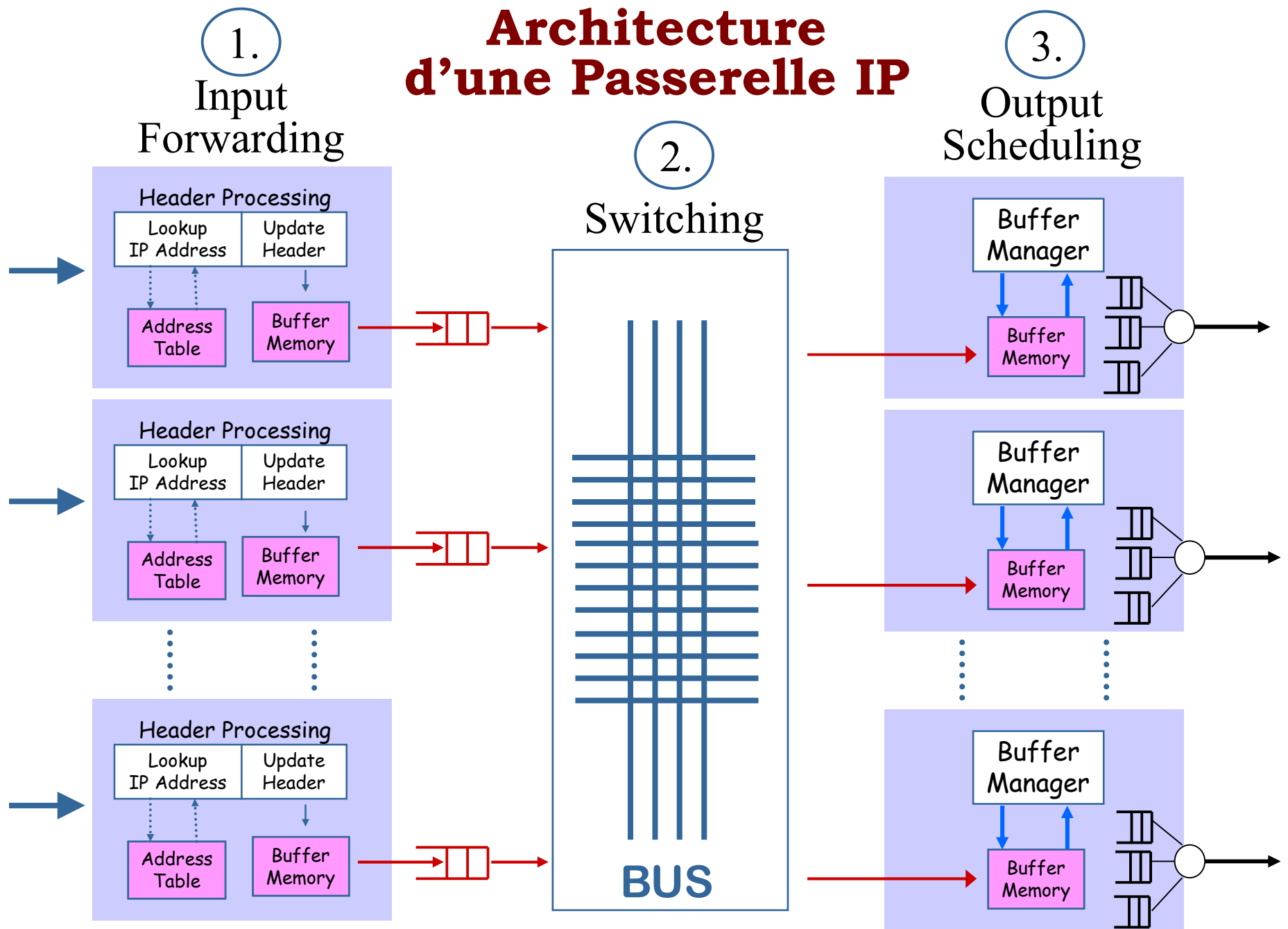
Pour chaque datagramme IP qui traverse une passerelle, le protocole IP :

1. détermine si ce sont des données utilisateur (TCP ou UDP) ou de contrôle (ICMP) destinées à la passerelle (analyse du champ « Protocole »)
2. vérifie le checksum, si faux => destruction paquet
3. vérifie la liste de contrôle d'accès (optionnel : fonction de Pare-Feux)
4. décrémente la durée de vie (TTL) du paquet, si nulle => destruction
5. **forwarding**: **décide du routage** (consulte la table de routage)
6. **fragmente** le datagramme si nécessaire (pour respecter le MTU de la prochaine liaison)
7. **reconstruit l'en-tête IP** avec les champs maj (TTL, ID, FLAG, OFFSET, Checksum)
8. **Switching**: **transmet** le ou les fragments du paquet IP vers le port de sortie à travers le bus
9. **Scheduling**: **ordonnancement** du paquet dans la file de sortie
10. Remise du paquet à la couche 2 puis à la couche 1 pour codage et transmission
11. mise à jours des statistiques de trafic (optionnel)

A réception dans l'hôte destinataire, IP :

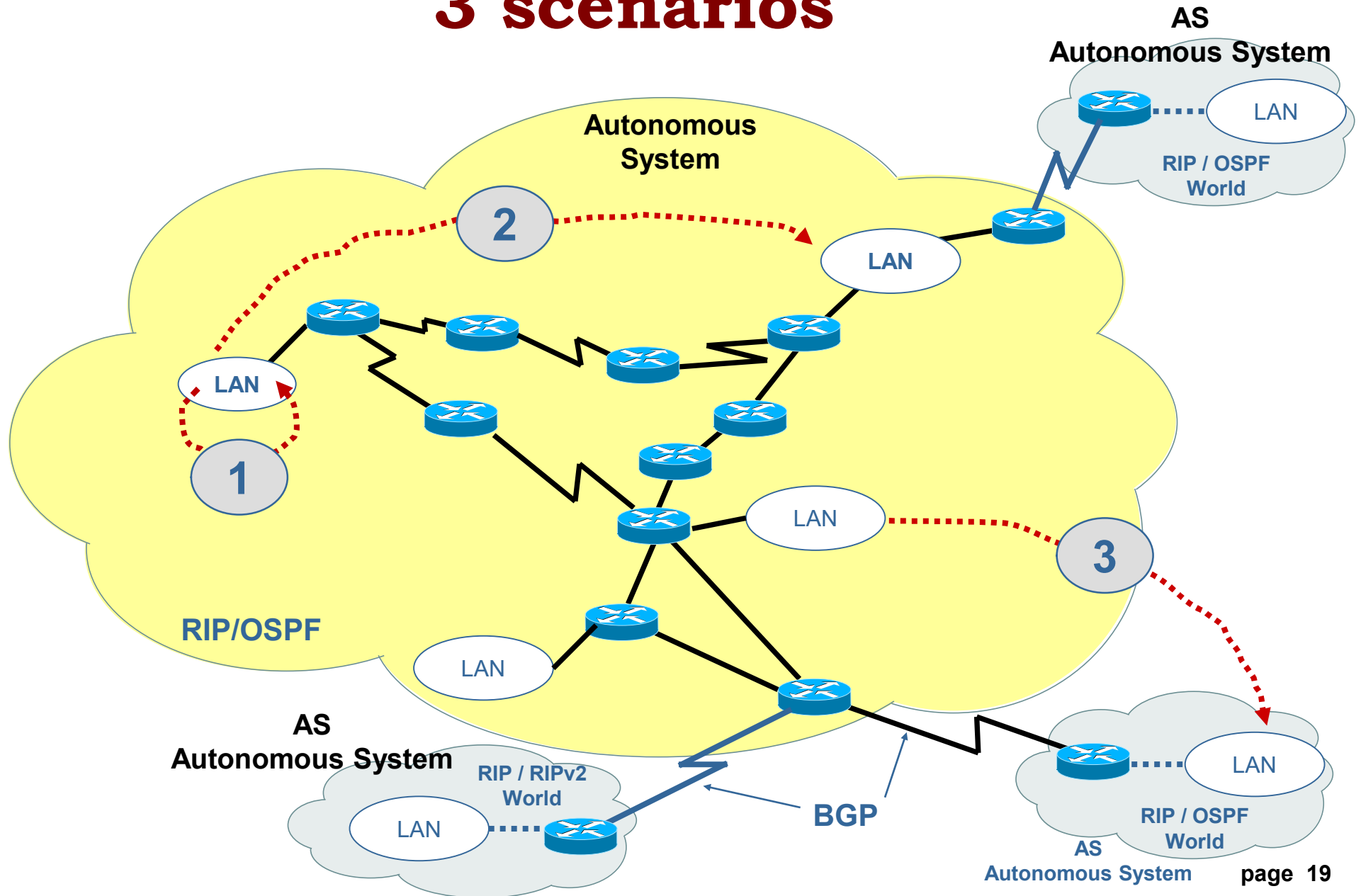
- vérifie le checksum
- s'il y a eu fragmentation, mémorise puis **réassemble**
- **délivre au niveau supérieur** (TCP, UDP) les données et les paramètres par la primitive DELIVER

# Architecture d'une Passerelle IP

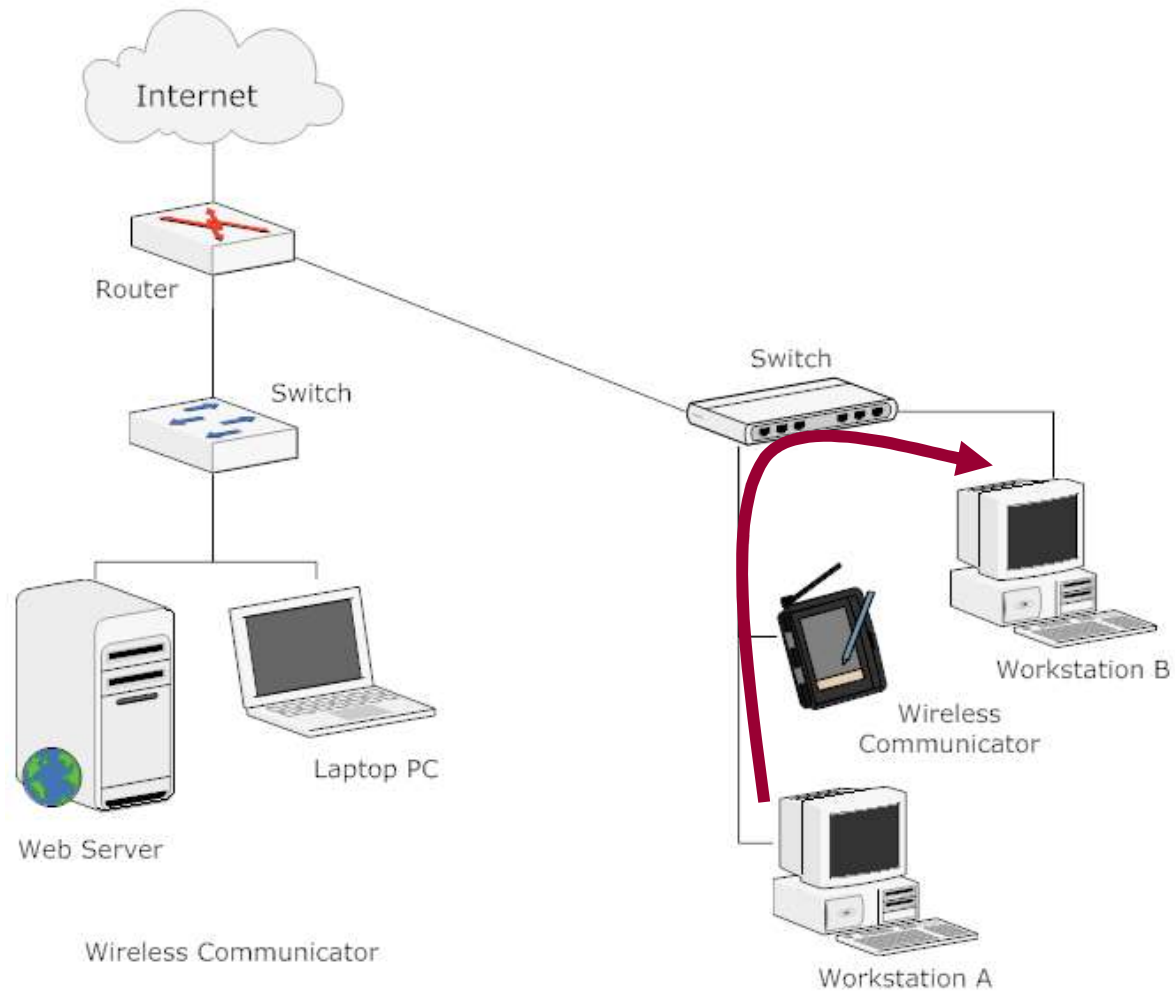


# Routage IP

## 3 scénarios

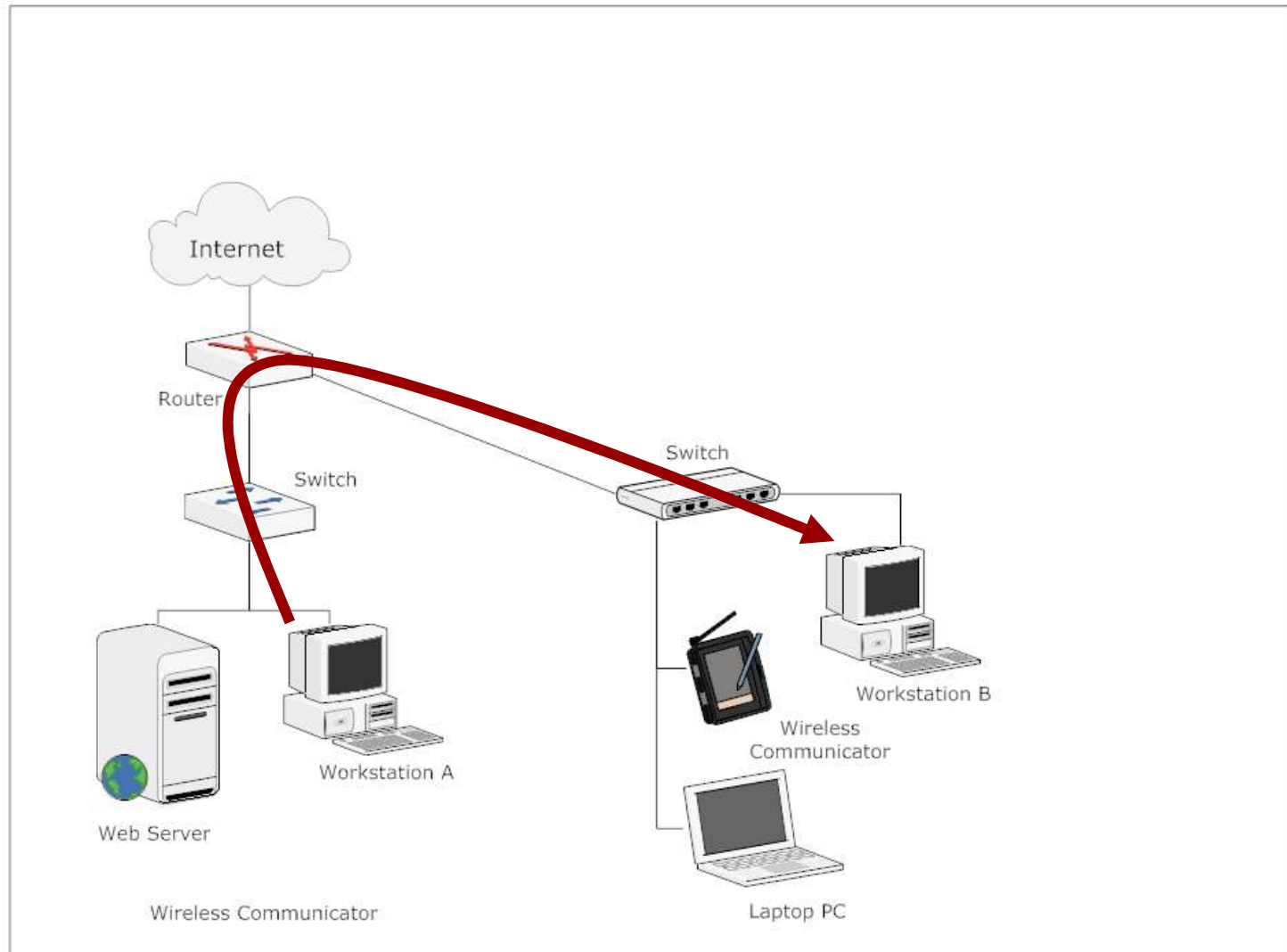


# Route direct



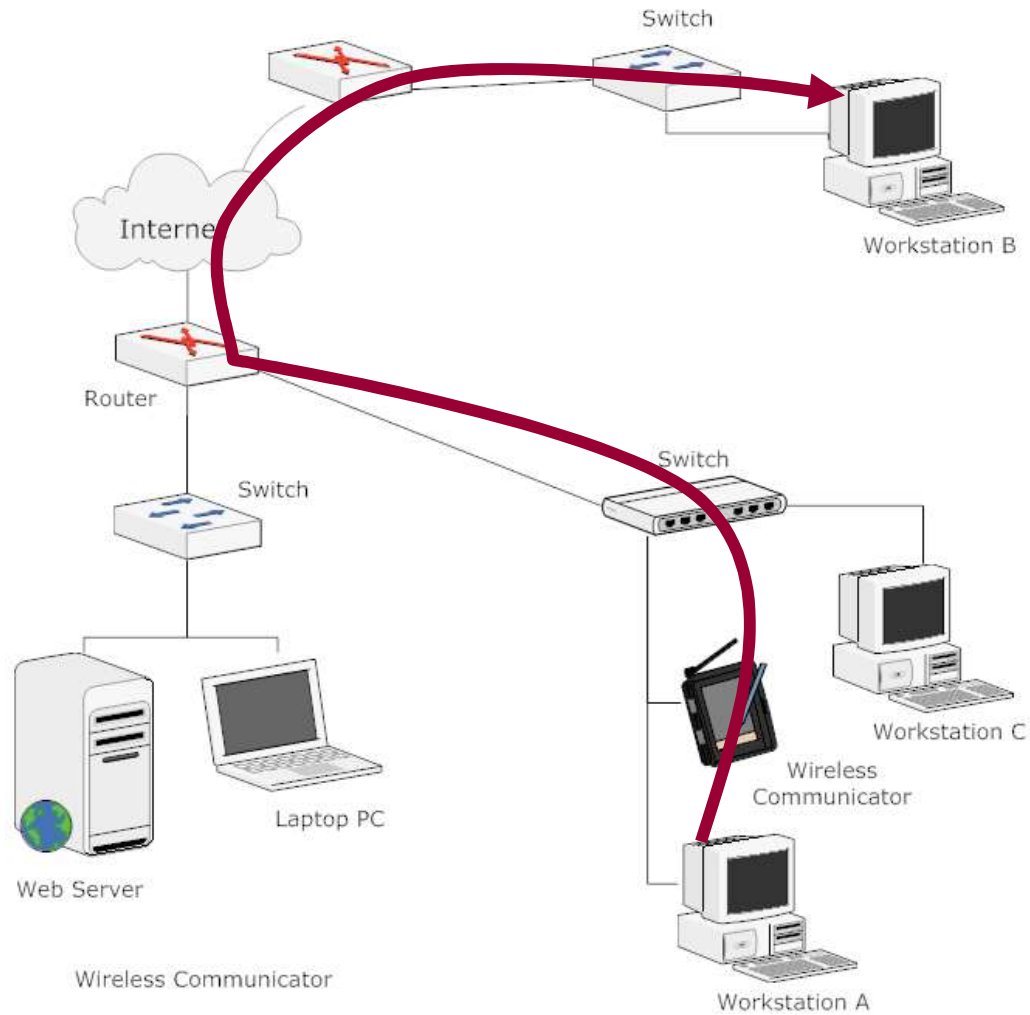
# Routage indirect

## - scénario 1 – entre sous-réseaux



# Routage indirect

## - scénario 2 – entre réseaux



# Routage IP intra-domaine

Distance vector algorithm : (utilisé avec le protocole RIP)

- algorithme simple,
- par diffusion d'un extrait des meilleurs chemins,
- (sous la forme d'un vecteur où chaque entrée contient une distance)
- entre voisins directs (de proche en proche)
- métrique simple : *hop count*.

Link state algorithm (pour information) : (utilisé avec le protocole OSPF)

- 2 phases :
  - . diffusion à tous de la connaissance sur les liaisons locales
  - . calcul local par chacun des meilleurs chemins sur les informations ainsi rassemblées
- exemple : Short Path First

# Routage IP

- Fonction qui permet de déterminer le meilleur chemin dans un réseau maillé vers une destination identifiée par une adresse IP.
- Utilisation de :
  - **TABLE DE ROUTAGE** (ou table d'acheminement) située dans chaque nœud : information nécessaire pour atteindre le prochain nœud vers la destination. Ex. Table de routage ip (netstat -r)
  - **ALGORITHME DE ROUTAGE** : fonction distribuée sur chaque nœuds qui a pour objectif de calculer les routes optimales pour atteindre une destination. Ex. Bellman-ford, Dijkstra,
  - **PROTOCOLES DE ROUTAGE** : pour rôle l'échanges des informations de routes calculées par les algorithmes de routage et qui permettent la mise à jour dynamique des tables de routage. Ex. RIP, OSPF



# Exple d'une table de routage

```
IPv4 Table de routage
=====
Itinéraires actifs :
Destination réseau    Masque réseau  Adr. passerelle  Adr. interface  Métrique
0.0.0.0              0.0.0.0        172.30.32.1      172.30.34.194    25
0.0.0.0              0.0.0.0        193.48.200.1     193.48.200.198   20
127.0.0.0            255.0.0.0      0n-link          127.0.0.1        306
127.0.0.1            255.255.255.255 0n-link          127.0.0.1        306
127.255.255.255      255.255.255.255 0n-link          127.0.0.1        306
172.30.32.0          255.255.240.0   0n-link          172.30.34.194    281
172.30.34.194        255.255.255.255 0n-link          172.30.34.194    281
172.30.47.255        255.255.255.255 0n-link          172.30.34.194    281
192.168.56.0         255.255.255.0   0n-link          192.168.56.1     276
192.168.56.1         255.255.255.255 0n-link          192.168.56.1     276
192.168.56.255       255.255.255.255 0n-link          192.168.56.1     276
193.48.200.0         255.255.255.0   0n-link          193.48.200.198   276
193.48.200.198       255.255.255.255 0n-link          193.48.200.198   276
193.48.200.255       255.255.255.255 0n-link          193.48.200.198   276
224.0.0.0            240.0.0.0       0n-link          127.0.0.1        306
224.0.0.0            240.0.0.0       0n-link          192.168.56.1     276
224.0.0.0            240.0.0.0       0n-link          193.48.200.198   276
224.0.0.0            240.0.0.0       0n-link          172.30.34.194    281
255.255.255.255      255.255.255.255 0n-link          127.0.0.1        306
255.255.255.255      255.255.255.255 0n-link          192.168.56.1     276
255.255.255.255      255.255.255.255 0n-link          193.48.200.198   276
255.255.255.255      255.255.255.255 0n-link          172.30.34.194    281
=====
```

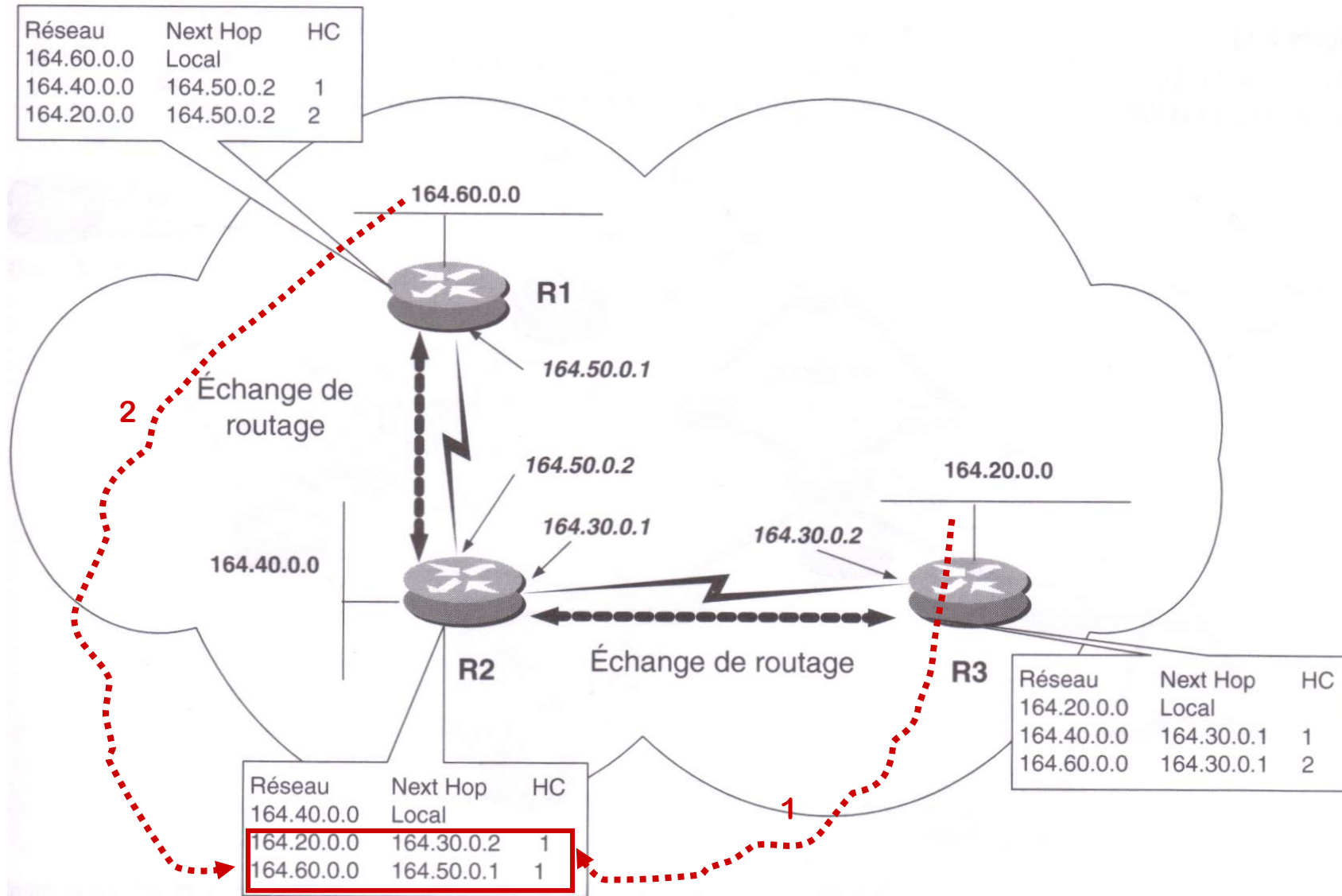
# Routage IP

- Machines et routeurs participent au routage :
  - Ils possèdent tous deux une table de routage,
  - les machines doivent déterminer si le datagramme doit être délivré sur le réseau physique sur lequel elles sont connectées (**routage direct**) ou bien si le datagramme doit être acheminé vers un routeur; dans ce cas (**routage indirect**), elle doit identifier le routeur appropriée.
  - les routeurs effectuent le choix de routage vers d'autres routeurs afin d'acheminer le datagramme vers sa destination finale.
  - **Commande : netstat -r**

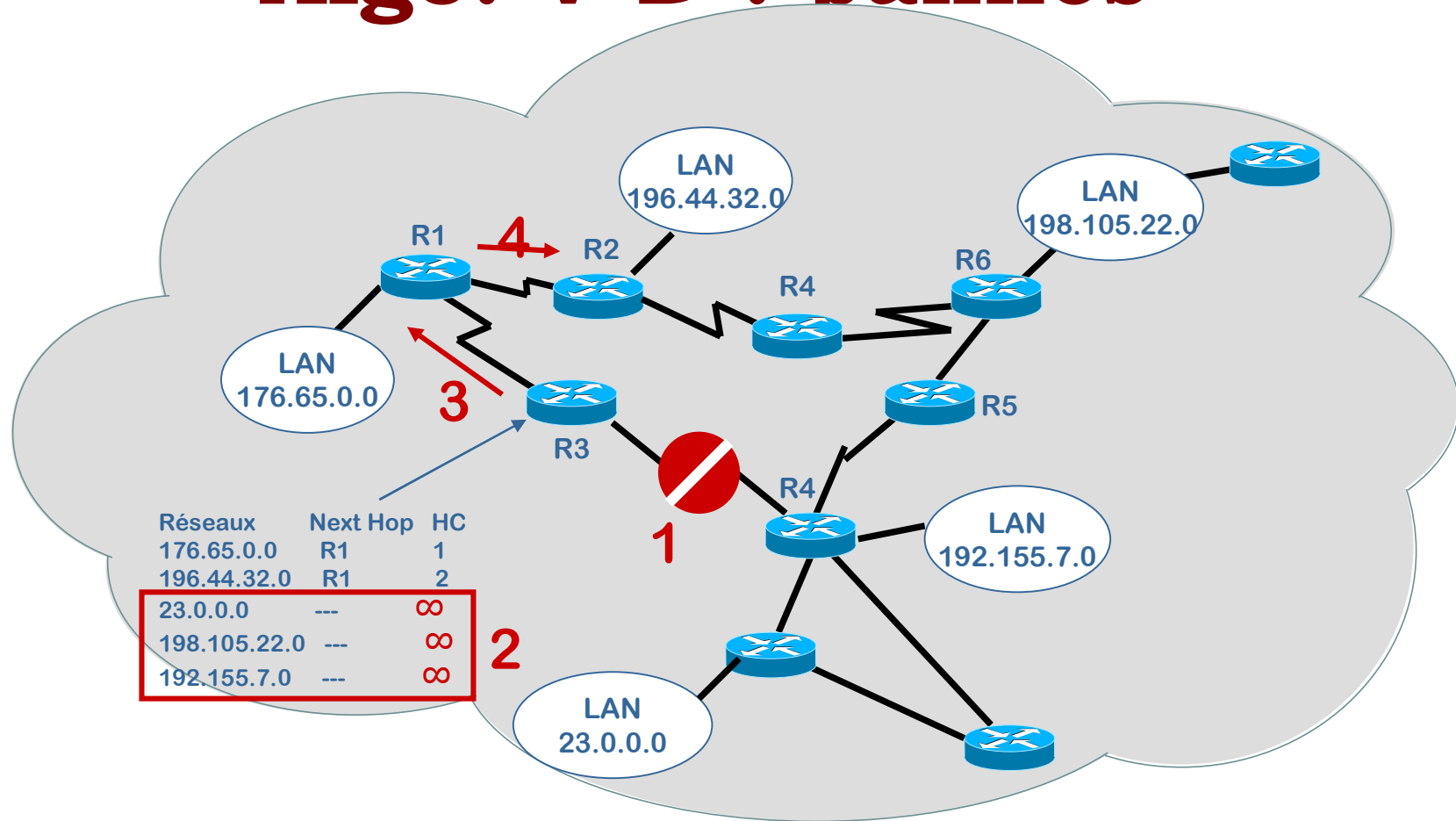
# Algorithme Distance Vector

- Basé sur l'algorithme de Belman-Ford, calcul de routes distribué.
- Un routeur diffuse régulièrement à ses voisins les routes qu'il connaît (toute les 30sec. Avec RIP).
- Une route est composée d'une adresse destination, son masque, d'une adresse de passerelle et d'une métrique indiquant le nombre de sauts nécessaires pour atteindre la destination.
- Une passerelle qui reçoit ces informations compare les routes reçues avec ses propres routes connues et met à jour sa propre table de routage :
  - si une route reçue comprend un plus court chemin (nombre de prochains sauts +1 inférieur),
  - si une route reçue est inconnue.

# Algorithme Vector distance



# Algo. V-D : pannes



1. R3 détecte qu'il n'y a plus de signal sur le port le reliant à R4;
2. R3 met toutes les routes passant par R4 à une distance **infinie** dans sa table
3. R3 envoie un nouveau vecteur de distances vers ses voisins actifs (R1)
4. La mise à jour se propage dans le réseau

# Algorithme Distance Vector

## Inconvénients :

- La taille des informations de routage est proportionnelle au nombre de routeurs du domaine,
- Métrique difficilement utilisable : lenteur de convergence,
- Bouclage, éventuellement à l'infini,
- Pas de chemins multiples
- Coût des routes externes arbitraire.

# RIP

Routing Information Protocol :

- RIP-1 : RFC 1058 - juin 1988.
- RIP-2 : RFC 1388 - juin 1993.

*routed* : Unix RIP routing deamon

commande *netstat -r* : visualise la table de routage

commande *route* : modifie la table de routage

fichier : */etc/hosts* : la table de routage initiale

## RIP + UDP + IP

- . Port n°520 (service RIP)
- . Infini = 16 hops ➡ étendue limitée
- . Période de diffusion des message de routage [15-45s]
- . Durée de validité d'un entrée (3 mn)
- . Délai aléatoire de diffusion immédiate [0-5s]

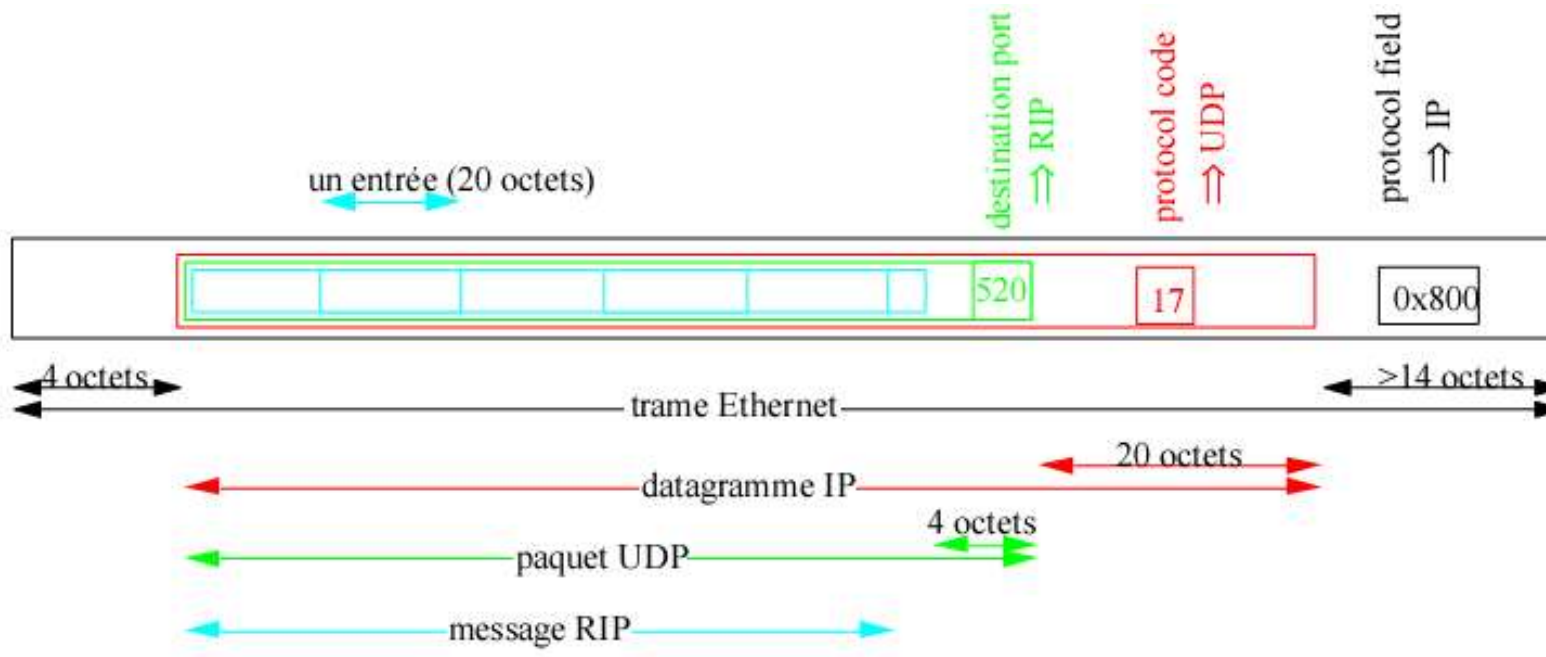
## Optimisation :

- RIP-1 utilise l'adresse de diffusion locale (255.255.255.255)
  - . Toutes les stations reçoivent une copie du message
- RIP-2 utilise l'adresse multicast réservée (224.0.0.9 : le groupe des routeurs)
  - . Seuls les routeurs RIP reçoivent une copie du message
    - ⇒ moins de surcharge pour les drivers IP des autres stations et autres routeurs.

# RIP Encapsulation

## Contraintes

- . Les messages de routage ont une longueur limitée : 512 octets
  - ⇒ le MTU par défaut des datagrammes IP est de 576 octets !
- . si les informations à transmettre sont plus longues, on diffuse plusieurs messages de routage.
- . le protocole RIP est sans mémoire (“memoryless”), ces messages ne sont pas liés (par ex. pas de n°).





# RIP principe

Etat initial :

Chaque routeur connaît son environnement immédiat :

- . son adresse, ses interfaces,
- . ses (sous-)réseaux directs : distance = 0.

Chaque routeur maintient localement une liste (BdD) des meilleures routes

⇒ table de routage <@ de destination, distance, @ du prochain routeur>

Chaque routeur actif diffuse un **extrait** de sa table de routage (message de routage) :

- Périodiquement (30s)
- A tous leurs voisins immédiats
- Une liste de couple <@ de destination, distance>

Tous les routeurs mettent à jour leur tables de routage en conséquence. L'adresse du prochain routeur est implicitement celui de l'émetteur du message de routage.

Etat des stations :

- Actif (les routeurs) diffusent leurs routes,
- Passif (les stations d'extrémité) écoutent.

# RIP Format des messages

Le champ “**command**”(8 bits) : code le type du message :

0	7	8	15	16
command		version		routing domain
address family			route tag	
IP address				
subnet mask				
next-hop address				
metric				
address family			route tag	
IP address				
subnet mask				
next-hop address				
metric				

. 1 = demande d'information

- demande partielle pour certaines destinations (dont les entrées figurent dans la demande)

- demande totale (s'il y a une seule entrée associée à la demande tel que “address family”=0 et “metric”=16)

. 2 = réponse

- l'extrait des meilleures routes du routeur

- suit à une demande, envoi périodique, envoi spontané

Le champ “**version**”(8 bits) :

. 1 = RIP-1 (⇒ les champs “routing domain”, “route tag”, “subnet address”, “next-hop address” sont inutilisés = 0)

. 2 = RIP-2

Le champ “**routing domain**”(16 bits) :

. RIP est générique :

- plusieurs domaines peuvent être gérés simultanément par le même routeur.

. 0 par défaut et obligatoire pour RIP-1

# RIP Format des messages (2)

0	7	8	15	16	31 bits
command		version		routing domain	
address family				route tag	
IP address					
subnet mask					
next-hop address					
metric					
address family				route tag	
IP address					
subnet mask					
next-hop address					
metric					

Le champ “**address family**” (16 bits) : code le format d'adressage :

- . les adresses peuvent être de longueur quelconque
- . 2 = IP  $\Rightarrow$  (32 bits)

Le champ “**route tag**” (16 bits) :

- . transmet des informations utilisées par le routage inter-domaine (EGP)
- . 0 pour RIP-1

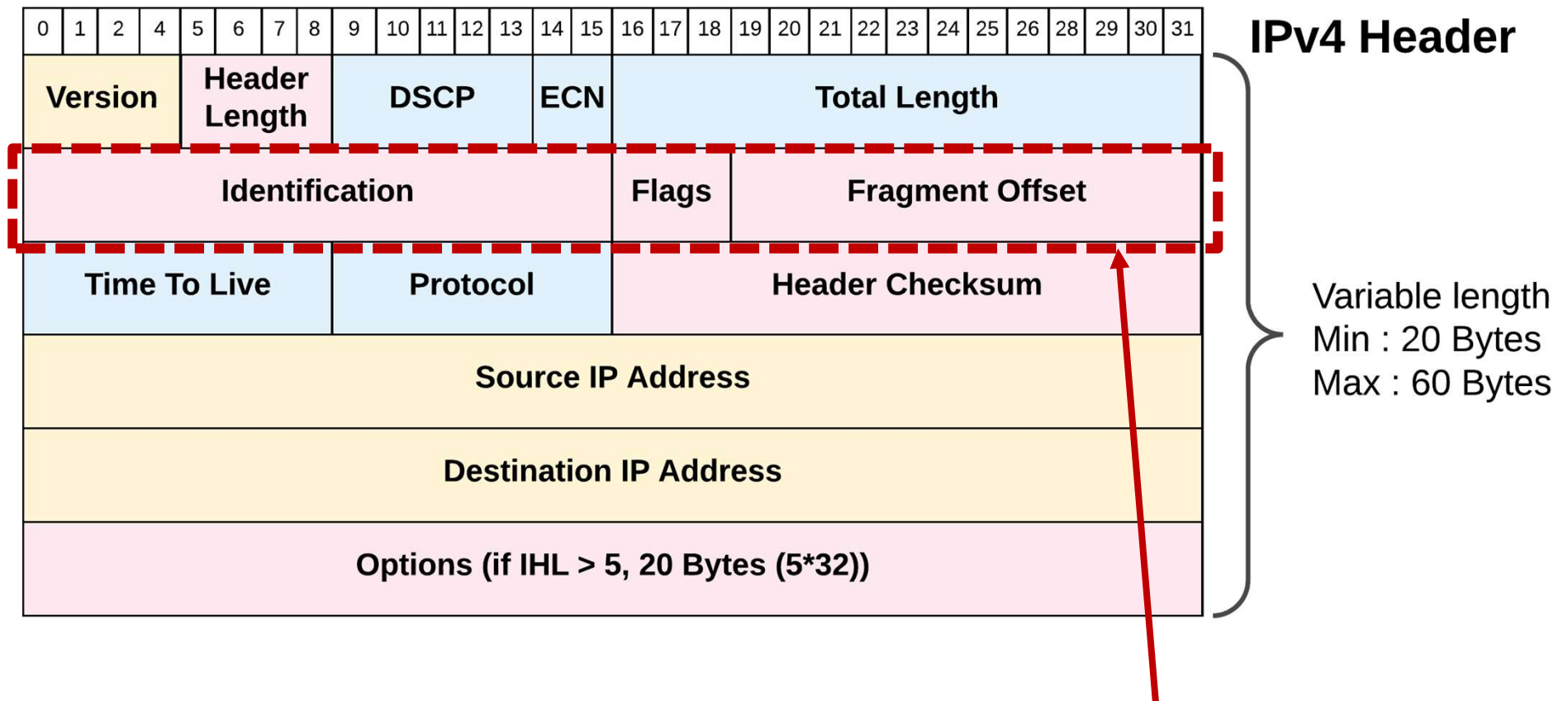
Le champ “**IP address**” (32 bits) : l'adresse de destination

- . l'adresse d'un réseau IP ( $\Rightarrow$  netid)
- . l'adresse d'un sous-réseau IP ( $\Rightarrow$  subnet mask : subnetid)
- . l'adresse d'une station ( $\Rightarrow$  @IP)
- . l'adresse par défaut ( $\Rightarrow$  n'importe quelle destination : 0.0.0.0)

Le champ “**subnet mask**” (32 bits) :

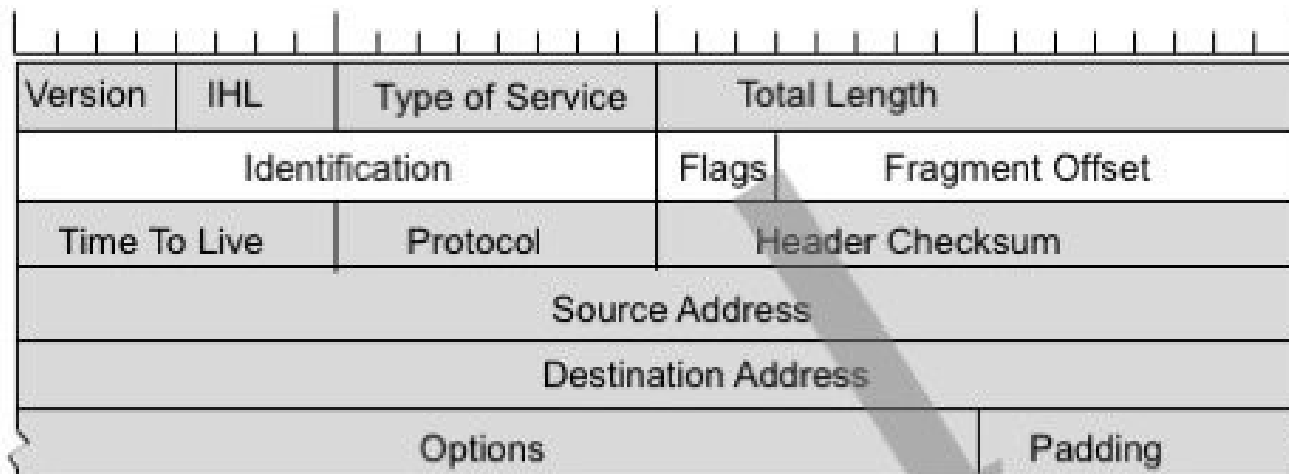
- . 0 pour RIP-1
- . spécifie la taille du champ “subnetID” dans le champ “hostID” de l'adresse IP.

# IP fragmentation des paquets



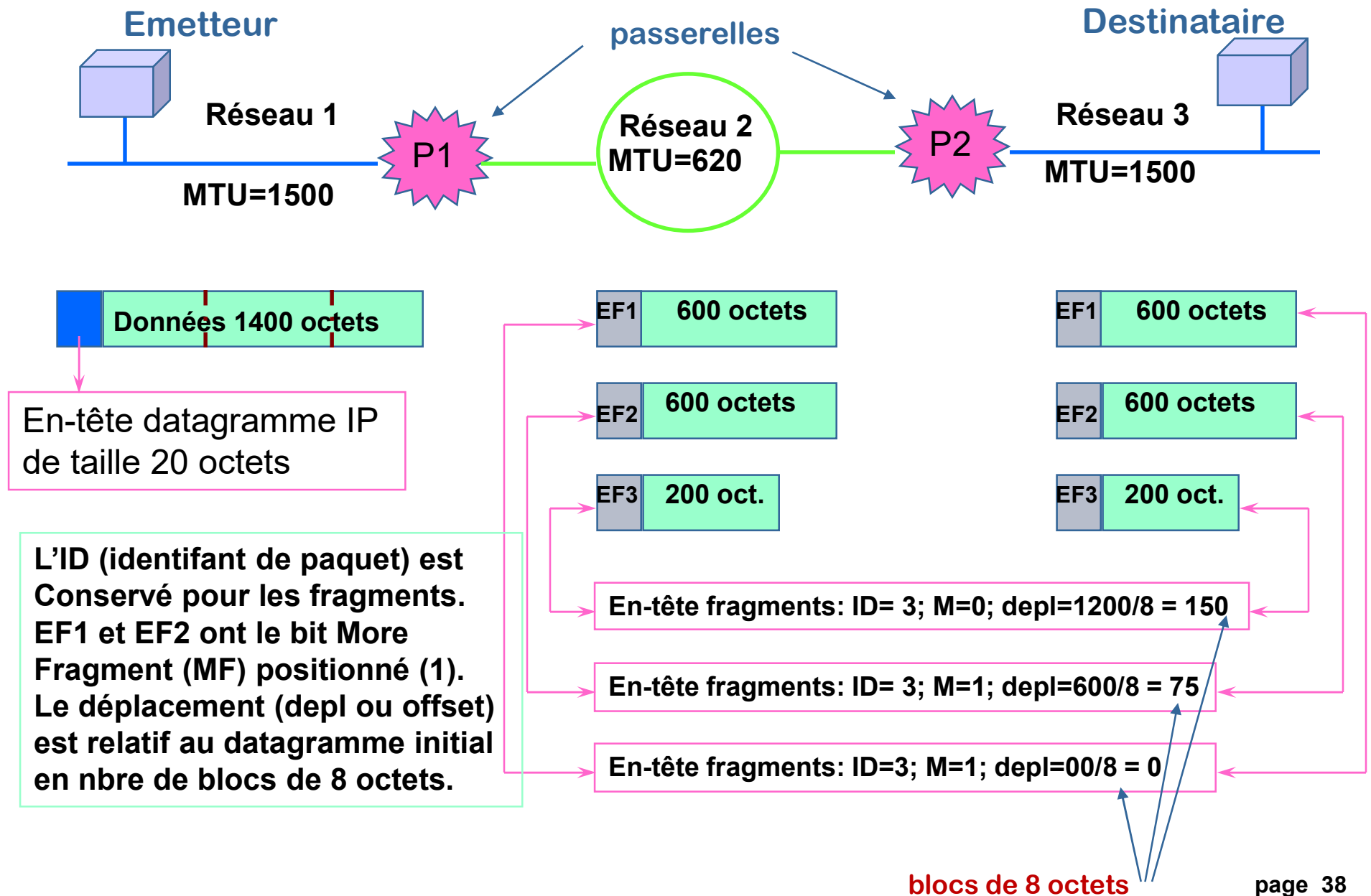
3 champs utilisés pour gérer la fragmentation des paquets IP

# IP Flags



Flags: bit 0 – Reserved  
bit 1 - Don't Fragment  
bit 2 – More Fragments

# Fragmentation des datagrammes IP





Original IP Packet			
Version	IHL	Type of Service	Total Length = 1320
Identification = 1956	0	0	Fragment Offset = 0
Time To Live	Protocol	Header Checksum	
Source Address			
Destination Address			
1300 Octets of Data			

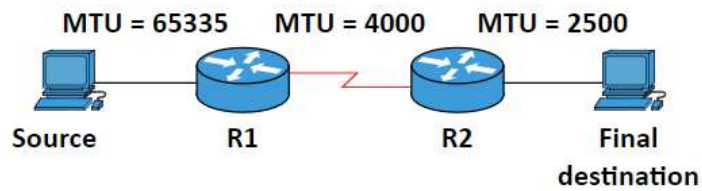
Fragmentation

1st Fragment			
Version	IHL	Type of Service	Total Length = 532
Identification = 1956	0	0	Fragment Offset = 0
Time To Live	Protocol	Header Checksum	
Source Address			
Destination Address			
512 Octets of Data			

2nd Fragment			
Version	IHL	Type of Service	Total Length = 532
Identification = 1956	0	0	Fragment Offset = 64
Time To Live	Protocol	Header Checksum	
Source Address			
Destination Address			
512 Octets of Data			

3rd Fragment			
Version	IHL	Type of Service	Total Length = 296
Identification = 1956	0	0	Fragment Offset = 128
Time To Live	Protocol	Header Checksum	
Source Address			
Destination Address			
276 Octets of Data			





## Legend



Host



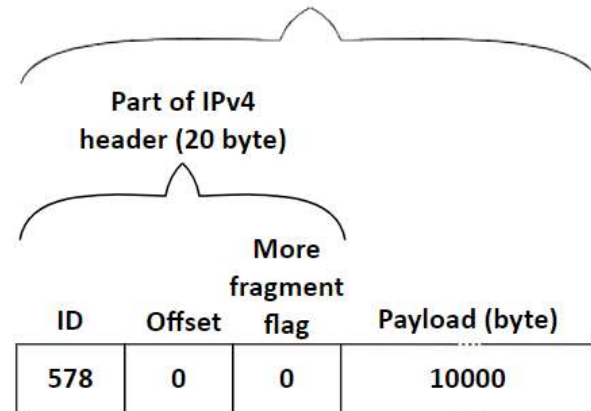
Links



Router

MTU: Maximum Transmission Unit  
(Byte)

## The original packet



First fragmentation  
R1:(4000)

First fragmentation  
R2:(2500)

