

# Chapitre 3:

## **Le routage IP**

# Le Routage IP

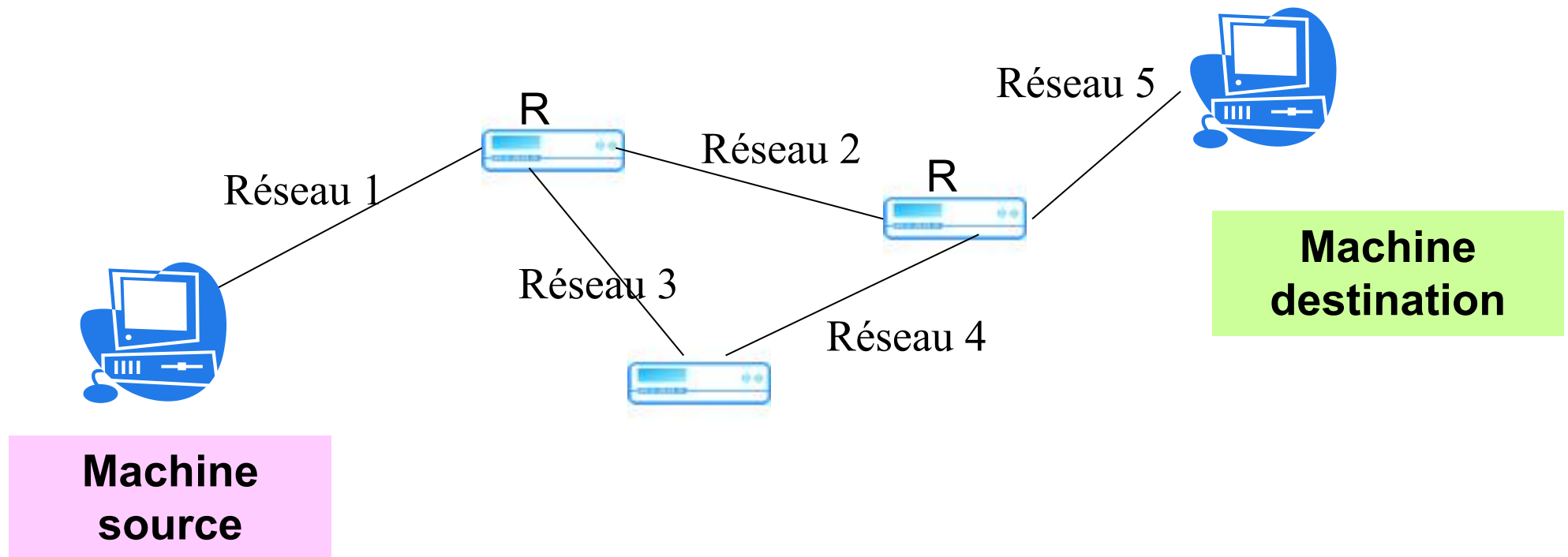
- **Routage = processus permettant à un paquet d'être acheminé vers le destinataire lorsque celui-ci n'est pas sur le même réseau physique que l'émetteur (en fonction de la politique choisie).**
- **Le routeur réalise le choix du chemin en appliquant un algorithme particulier à partir de tables de routage.**

# Le Routage IP

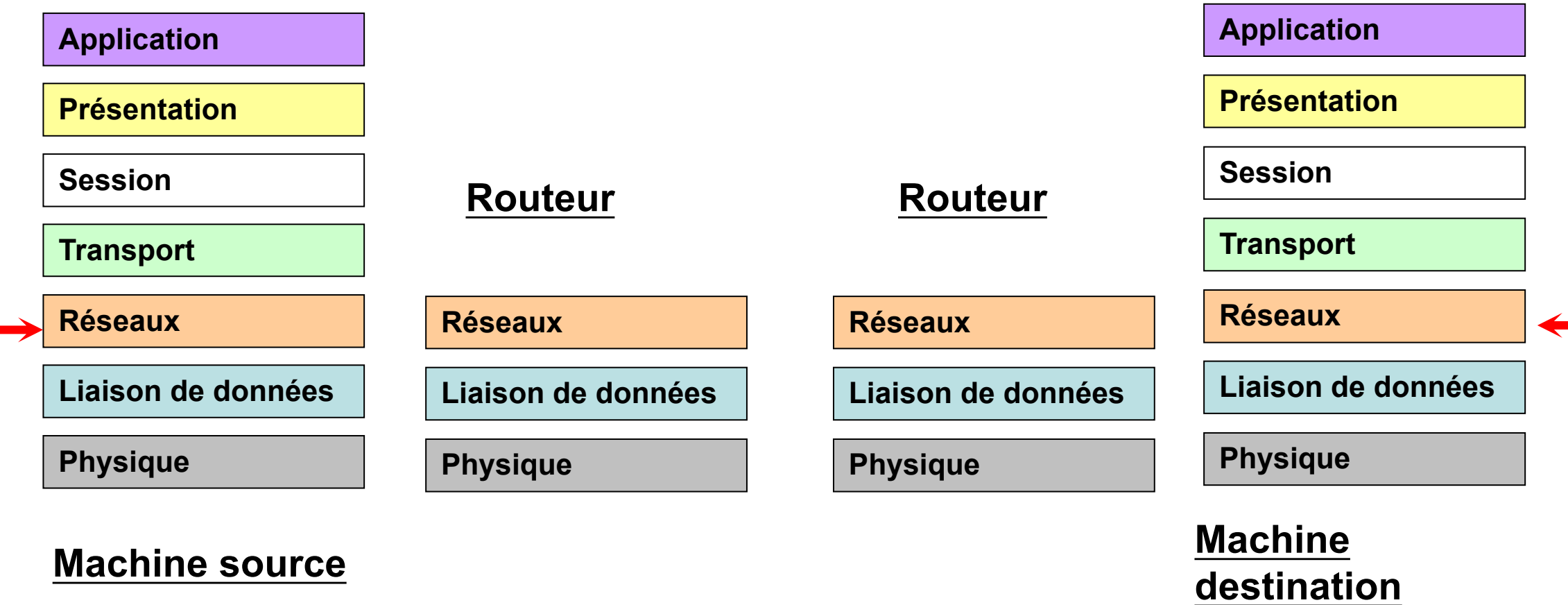
Les réseaux utilisent les adresses IP et communiquent entre eux via les routeurs .

## ❖ Définition :

un routeur est une machine qui permet de transmettre les informations d'une machine **source** vers une **destination** .



# Le Routage IP



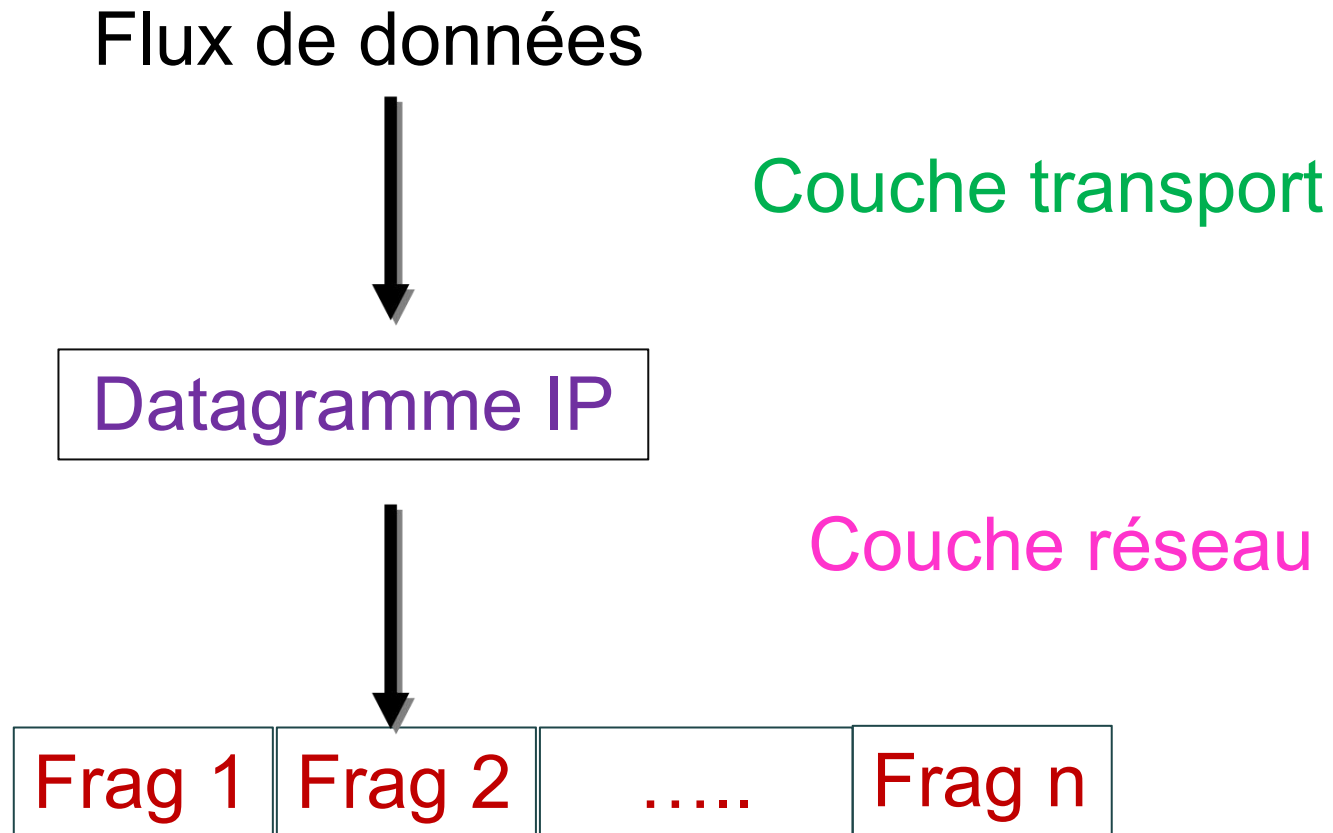
# Le protocole IP

## Introduction

Les différents réseaux hétérogènes d'Internet coopèrent grâce au protocole IP.

- IP permet **l'identification** de tout équipement (grâce à l'adressage IP).
- IP permet **l'échange de datagrammes** entre tout couple d'équipements.
- **Objectif** : faire le mieux possible pour transmettre les datagrammes de leur source vers leur destination.

# Communication via IP

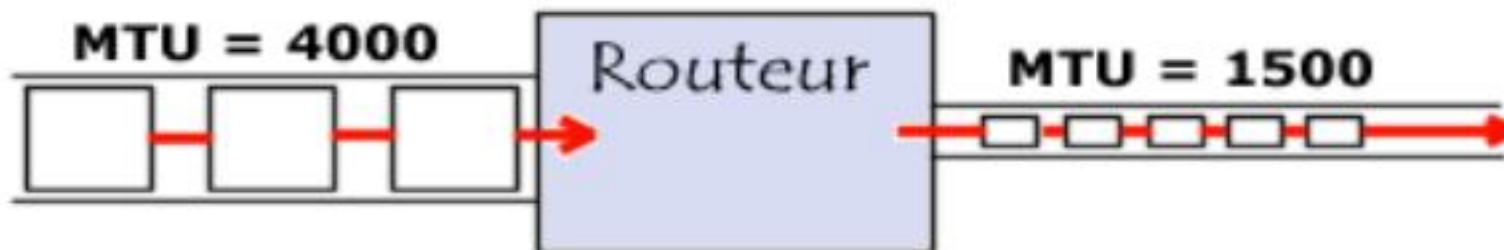


# fragmentation d'un datagramme

- La taille maximale d'une trame est appelée *MTU* (**Maximum Transfer Unit**), entraînera la fragmentation du datagramme si celui-ci a une taille plus importante que le MTU du réseau.

Type de réseau	MTU (en octets)
Arpanet	1000
<a href="#">Ethernet</a>	1500
<a href="#">FDDI</a>	4470

- $\text{taille} < \text{MTU du réseau}$
- taille du fragment soit un multiple de 8 octets.



# Communication via IP

- À destination, tous les morceaux **sont réassemblés par la couche réseau** pour recomposer le datagramme.
- La couche transport **reconstitue le flux de données** initial pour la couche application.

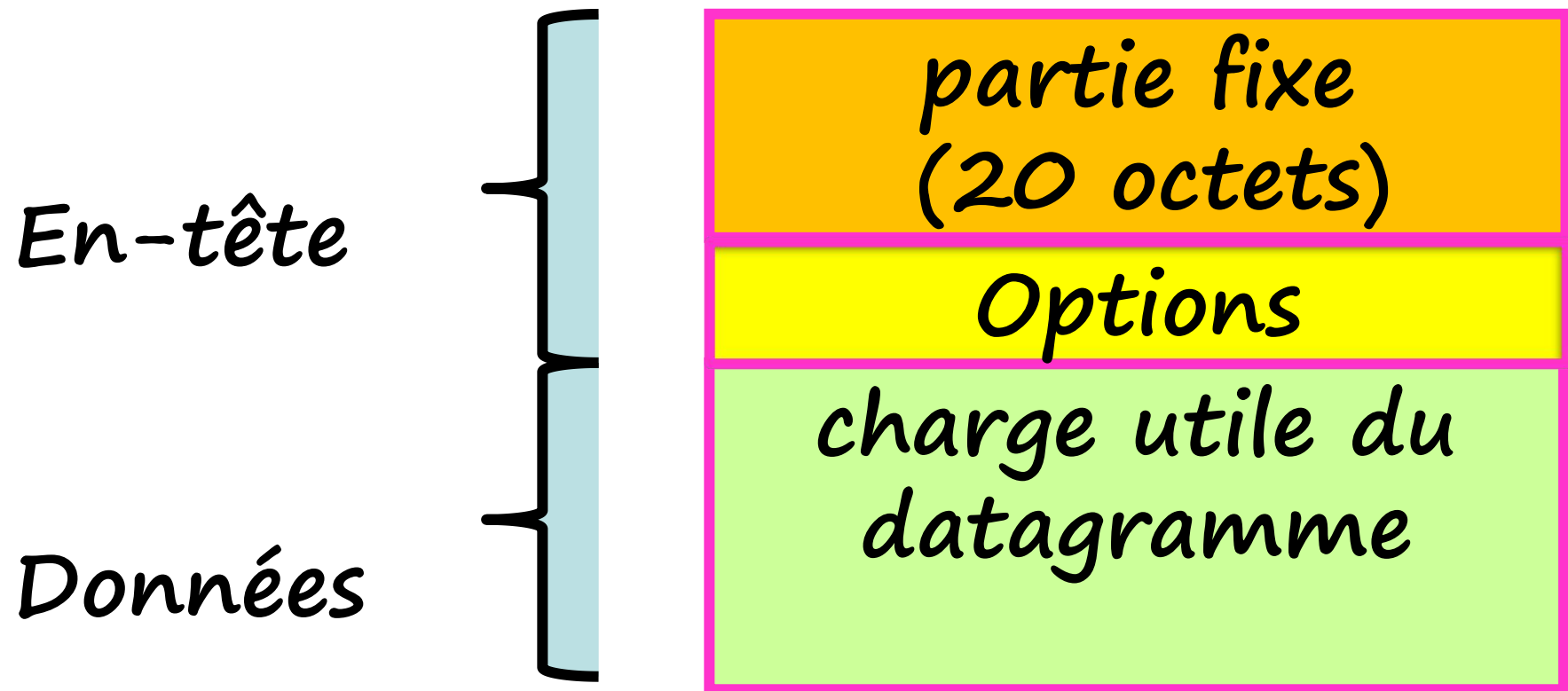


# Le service offert par IP

- Le service offert par le protocole IP est dit non fiable
- remise de paquets **non garantie**,
- **sans connexion** : paquets traités indépendamment les uns des autres
- **pour le mieux** : **best effort**, les paquets ne sont pas éliminés sans raison

# Datagramme IP

Constitué de deux parties : un entête et des données



# Format du Datagramme IP

0                      4                      8                      16                      31

Version	Longueur d'entête	Type de service	Longueur total du datagramme en octets		
Identificateur du datagramme			DF	MF	Position du fragment
Durée de vie		Protocole qui utilise IP		Total de contrôle d'entête	
Adresse IP émetteur					
Adresse IP destination					
Options ( 0 ou plusieurs mots)					
Données ...					

# Champs d'en-tête

Version : numéro de la version du protocole utilisé pour créer le datagramme (4 bits) : version 4 ou 6.

Longueur entête : longueur de l'en-tête exprimée en mots de 32 bits (= 5 si pas d'option)

Type de service : précise le mode de gestion du datagramme (**8bits**)

- Priorité : 0 (normal) ® 7 (supervision réseau) (**3 bits**)
- Indicateurs **D, T, R** : indiquent le type d'acheminement désiré du datagramme, permettant à un routeur de choisir entre plusieurs routes (si elles existent) :
  - » D (délai court)
  - » T (débit élevé)
  - » R (grande fiabilité)
- **2 bits** inutilisés

# Champs d'en-tête

- Longueur totale en octets (**16 bits**)
- Identificateur : permet au destinataire de savoir à quel datagramme appartient un fragment (**16 bits**)
- Drapeau : **3 bits**
  - DF : “ Don’t fragment ”
  - MF : “ More fragments ”
  - 1bit inutilisé

# Champs d'en-tête

Position du fragment : localisation du déplacement du fragment dans le datagramme (**13 bits**)

Durée de vie (TTL) : compteur utilisé pour limiter la durée de vie des datagrammes (**8 bits**). Nombre maximal de routeurs que le datagramme peut traverser :

décrémenté à chaque saut détruit quand il passe à 0

Protocole : indique par un numéro à quel protocole confier le contenu du datagramme (**8 bits**)

6 = TCP, 17 = UDP, 1 = ICMP.

Protocole de niveau supérieur ayant créé le datagramme

## Champs d'en-tête

Total de contrôle d'en-tête : vérifie la validité de l'en-tête, doit être recalculé à chaque saut (16 bits)

Adresse IP source : 32 bits

Adresse IP destination : 32 bits

# Les options du datagramme

- Le champ **OPTIONS** est facultatif et de longueur variable. Les options concernent des fonctionnalités de mise au point.
- Une option est définie par **un champ octet**.

C	Classe d'option	Numéro d'option
---	-----------------	-----------------

- **C (copie):**
  - **(c=1)** indique que l'option doit être recopiée dans tous les fragments
  - **(c=0)** : option copié uniquement dans le 1<sup>er</sup> fragment
- les bits classe d'option et Num d'option indiquent **le type de l'option**



# Les options du datagramme

Enregistrement de route (classe =0, option =7)

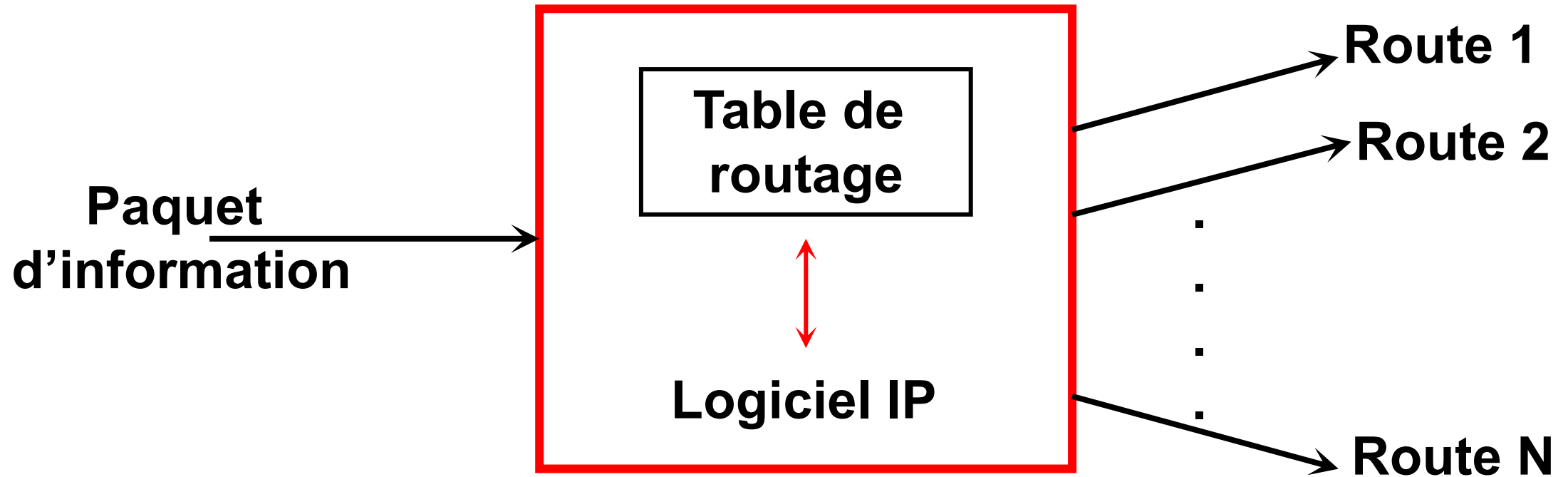
Oblige chaque passerelle/routeur d'ajouter son adresse dans la liste.

Routage strict prédéfini par l'émetteur (classe =0, option =9) prédéfinit le routage qui doit être utilisé dans l'interconnexion en indiquant la suite des adresses IP.

Routage lâche prédéfini par l'émetteur (classe =0, option =3) autorise, entre deux passages obligés, le transit par d'autres intermédiaires.

Horodatage (classe =2, option =4) permet d'obtenir les temps de passage (*timestamp*) des datagrammes dans les routeurs. Exprimé en heure et date universelle

## Schéma simplifié d'un routeur :



## Table de routage

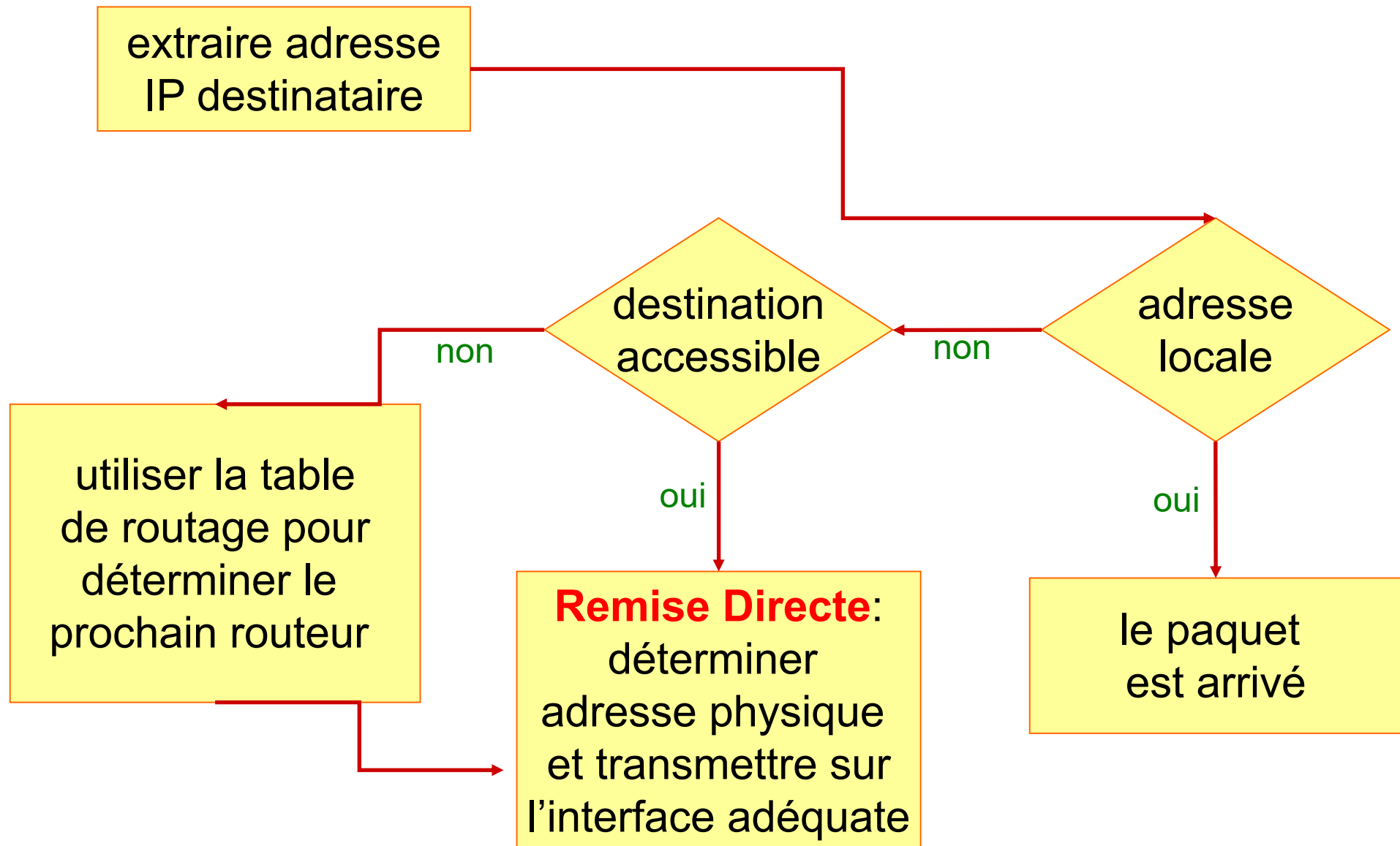
- Les routeurs décident de la route à faire suivre aux paquets IP par consultation d'une table de routage.
- La maintenance des tables de routages est une opération fondamentale. Elle peut être statique (manuelle) ou dynamique.

# Table de routage

- Ne contiennent que les identifiants réseau des adresses IP.
- La table contient, pour chaque numéro de réseau à atteindre, l'adresse IP du routeur le plus proche.
- Chaque routeur possède une liste de couples:
  - définit comment accéder à un réseau distant ou
  - comment accéder à un ordinateur du réseau local.

Destination	Routeur NEXT HOP
<b>200.1.1.0</b>	<b>Remise directe</b>
<b>201.12.5.0</b>	<b>200.1.1.3</b>
<b>202.10.1.0</b>	<b>200.1.1.6</b>
<b>203.6.5.0</b>	<b>200.1.1.9</b>

# Méthode de routage



## Route par défaut

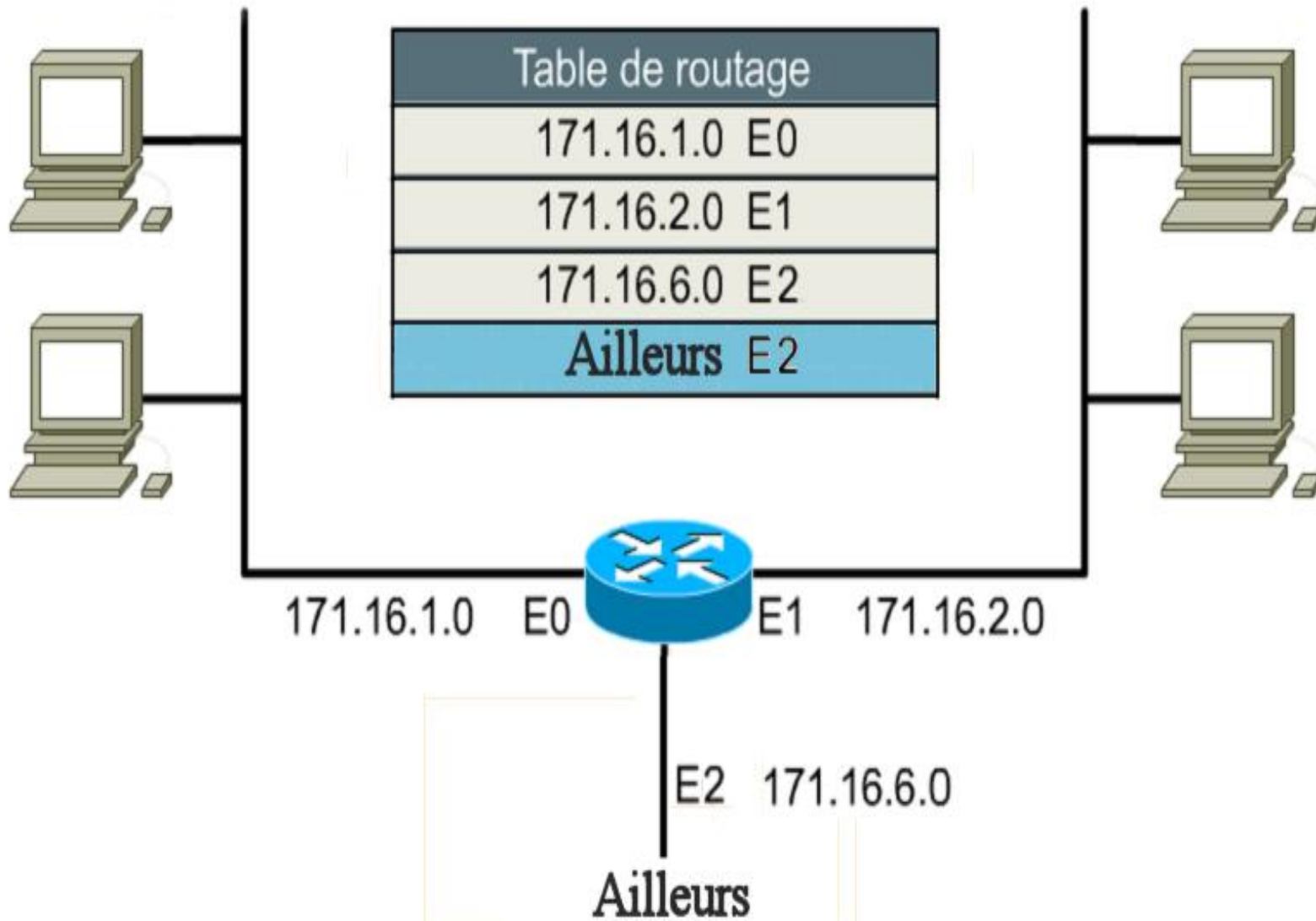
Une route par défaut permet d'acheminer un paquet dont la destination ne correspond à aucune route de la table de routage

```
/> route
```

Destination	Masque	Passerelle	Interface
172.12.0.2	255.255.255.255	127.0.0.1	127.0.0.1
172.12.0.0	255.255.0.0	172.12.0.2	172.12.0.2
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1
0.0.0.0	0.0.0.0	172.12.255.254	172.12.0.2

```
/> |
```

# Route par défaut



# Objet du routage

Trouver (calculer) le plus court chemin à emprunter  
d'une source à une destination

Distance ?

- nombre de sauts

- distance kilométrique

- temps moyen de transmission

- longueur moyenne des files d'attente



# Propriétés d'un algorithme de routage

- Exactitude
- Simplicité
- Robustesse (capacité d'adaptation aux pannes et changement de topologie)
- Stabilité (convergence vers un état d'équilibre)
- Optimisation

# Classes d'algorithmes de routage

Comment un routeur peut-il connaître les différents chemins le reliant aux autres routeurs ?

- routage statique
- routage dynamique (protocole de routage)

# Routage Statique

L'administrateur réseau spécifie manuellement la table de routage.

Inconvénients : l'administrateur doit faire les mises à jour en cas de changement de la topologie du réseau

Avantages : réduction de la charge du système, car aucune mise à jour de routage n'est envoyée

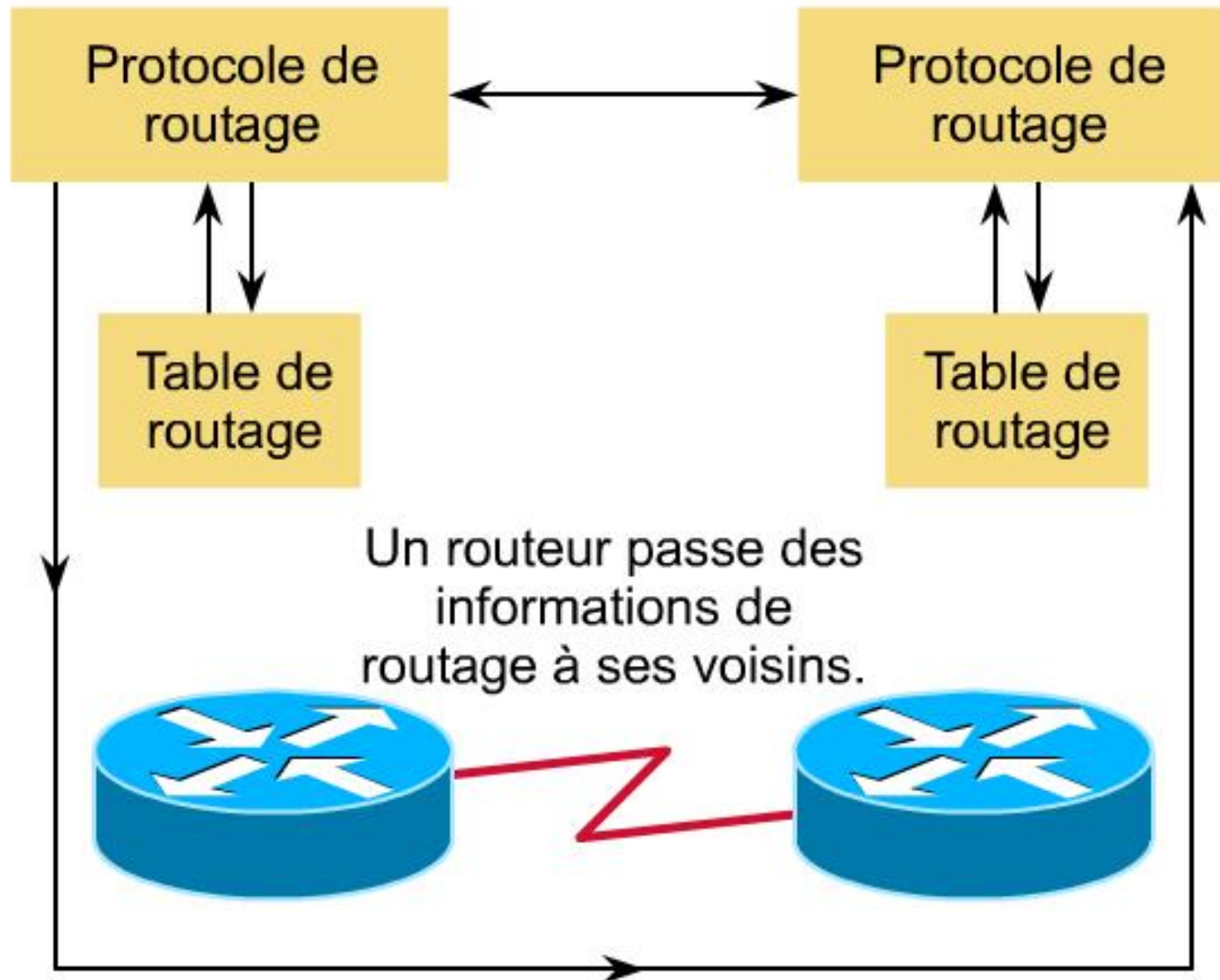
# Routage dynamique

L'administrateur réseau met en place un protocole de routage établissant automatiquement les chemins entre deux routeurs.

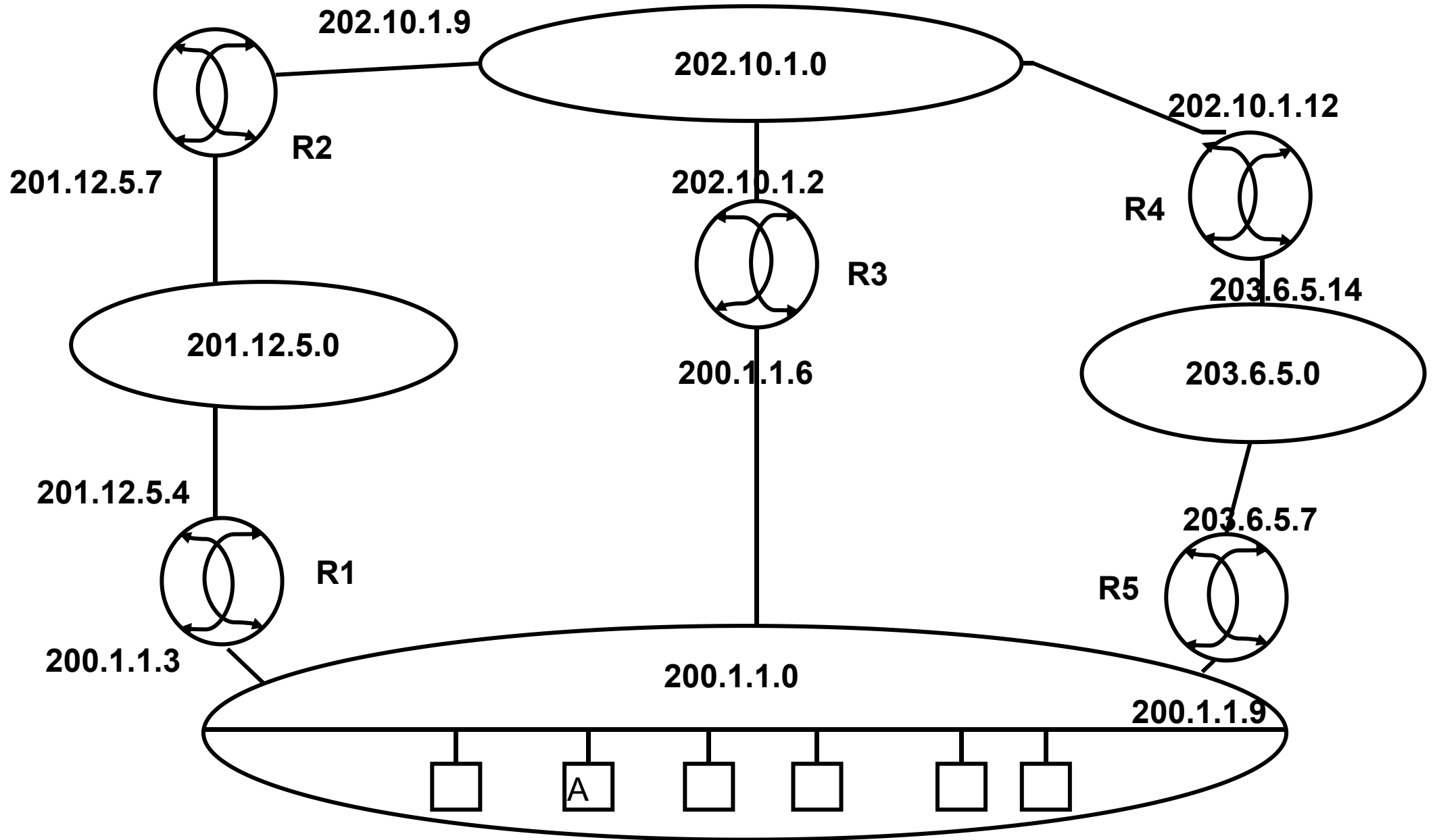
Inconvénients : augmentation de la charge du système, car des mises à jour de routage doivent être envoyées

Avantages : prise en compte automatique d'un changement de la topologie du réseau

# Routage dynamique



# Example :



## Exercice :

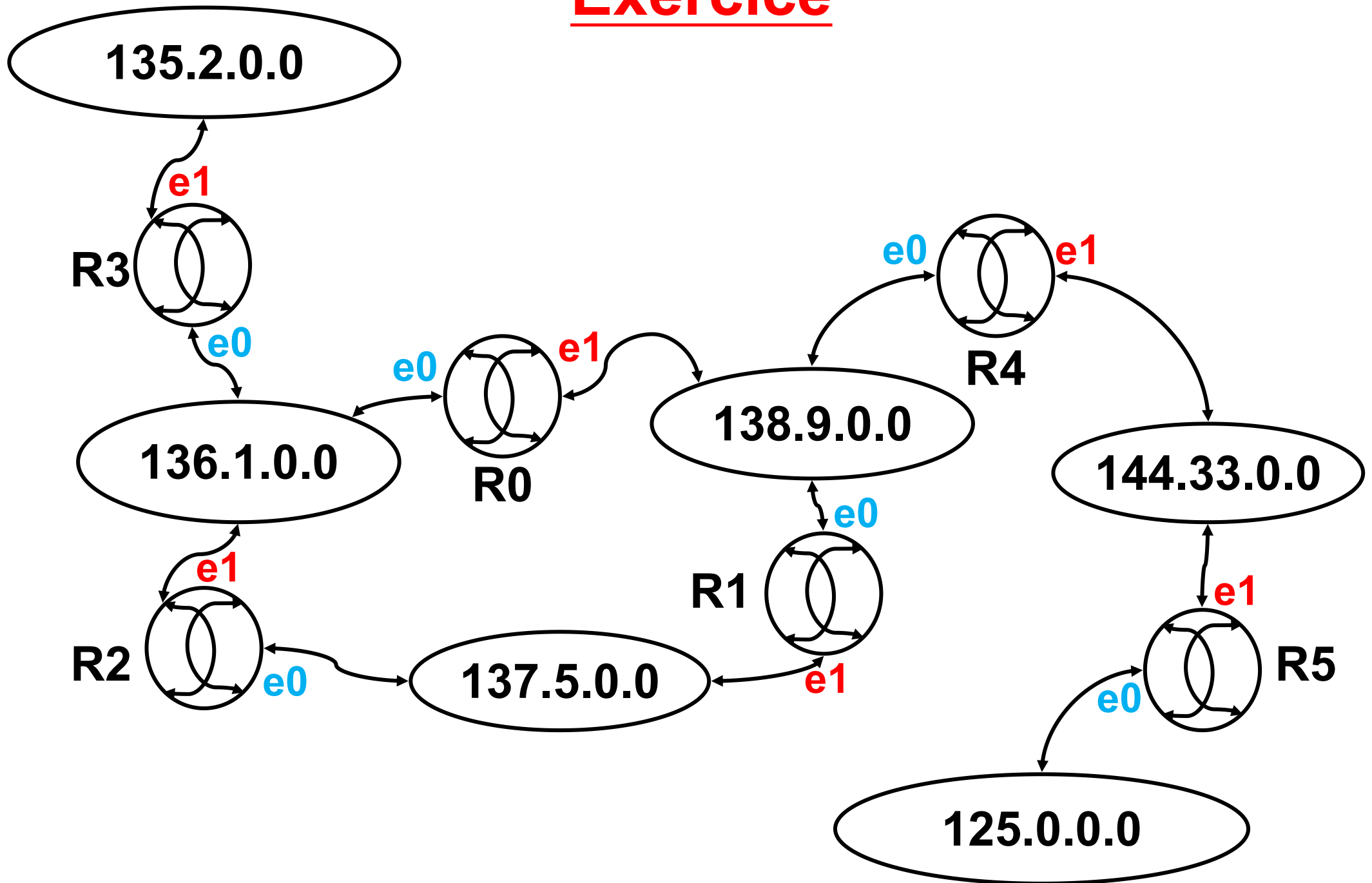
1. On vous demande de donner tous les réseaux et leurs classes .
2. On vous demande de donner tous les routeurs et leurs interfaces.
3. Proposez une table de routage pour la machine A .

La table de routage de la station A du réseau  
200.1.1.0

Destination	Routeur NEXT HOP
<b>200.1.1.0</b>	<b>Remise directe</b>
<b>201.12.5.0</b>	<b>200.1.1.3</b>
<b>202.10.1.0</b>	<b>200.1.1.6</b>
<b>203.6.5.0</b>	<b>200.1.1.9</b>

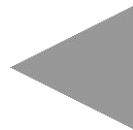


# Exercise



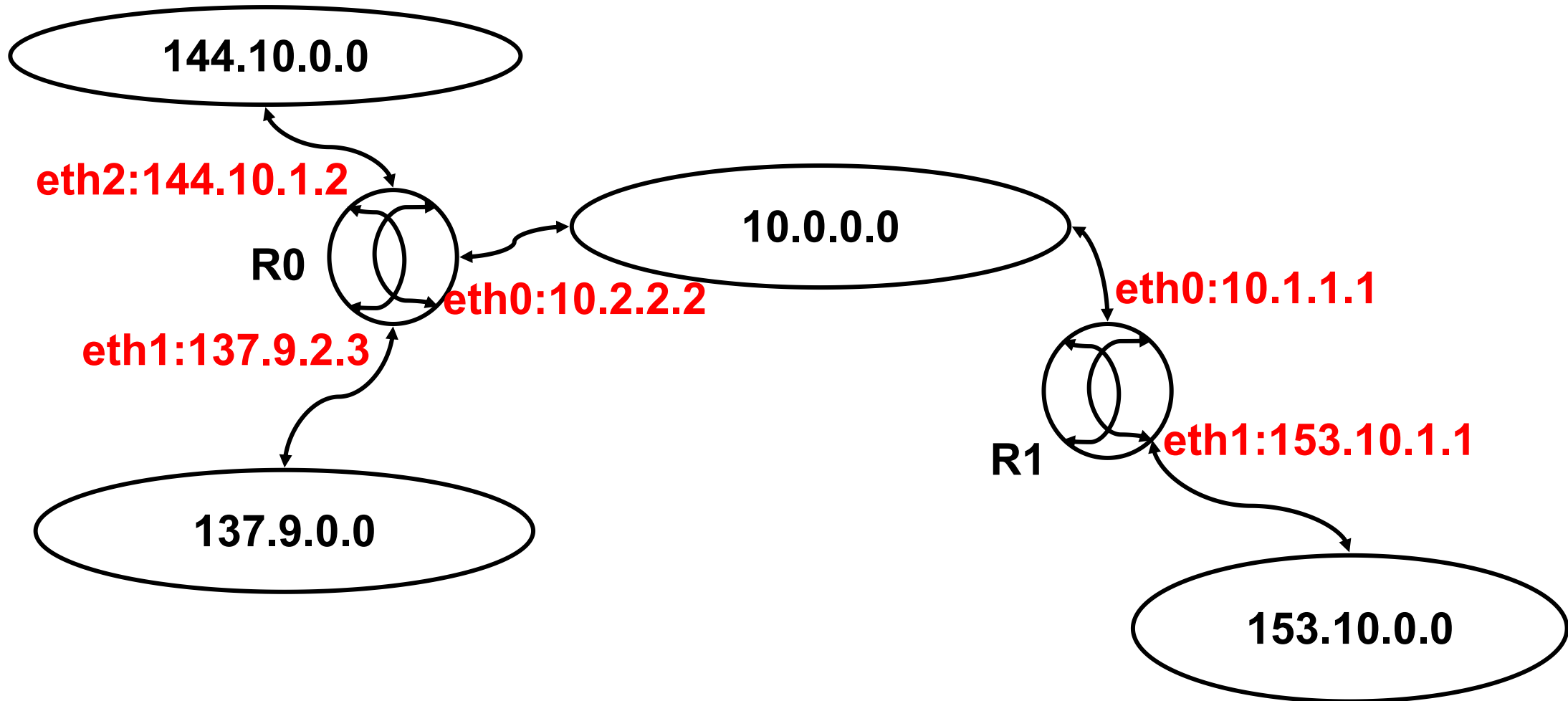
## Exercice à résoudre

- Donnez tous les réseaux et leurs classes .
- Déterminez tous les routeurs et leurs interfaces .
- Proposez une table de routage pour le routeur R0 .



# Les protocoles de routage (RIP)

- Considérons le réseau suivant :



- La table de routage de chaque routeur est constituée de réseaux voisins .

Table de routage du routeur **R0** :

Destination	Distance	RNH
144.10.0.0	1	RD
137.9.0.0	1	RD
10.0.0.0	1	RD

## Remarque

- Si un réseau est voisin d'un routeur alors la distance qui les sépare est égal à **1**

## Table de routage du routeur **R1** :

Destination	Distance	RNH
10.0.0.0	1	RD
153.10.0.0	1	RD

R1 envoie sa table de routage à R0

Le réseau **153.10.0.0** est loin du routeur R0  
d'une distance égal à **2** car :

**Règle :**

$$D(153.10.0.0, R0) = D(153.10.0.0, R1) + D(R1, R0)$$

## Explication

- R0 a déjà dans sa table de routage le réseau **10.0.0.0**
- R1 dit à R0 que le réseau **10.0.0.0** est loin de R0 d'une distance égal à **1** , donc après la règle  $D(R0, 10.0.0.0) = \mathbf{2}$
- Or dans la table de routage de R0  $D(R0, 10.0.0.0) = \mathbf{1}$  donc R0 ne vas pas mettre à jour le réseau **10.0.0.0**



## Exercice

- Donnez la table de routage de R0 après que R1 envoie sa table de routage à R0 .
- Donnez la table de routage de R1 après que R0 envoie sa table de routage à R1 .



## Solution

- Après mise à jour, la table de routage de R0 devient :

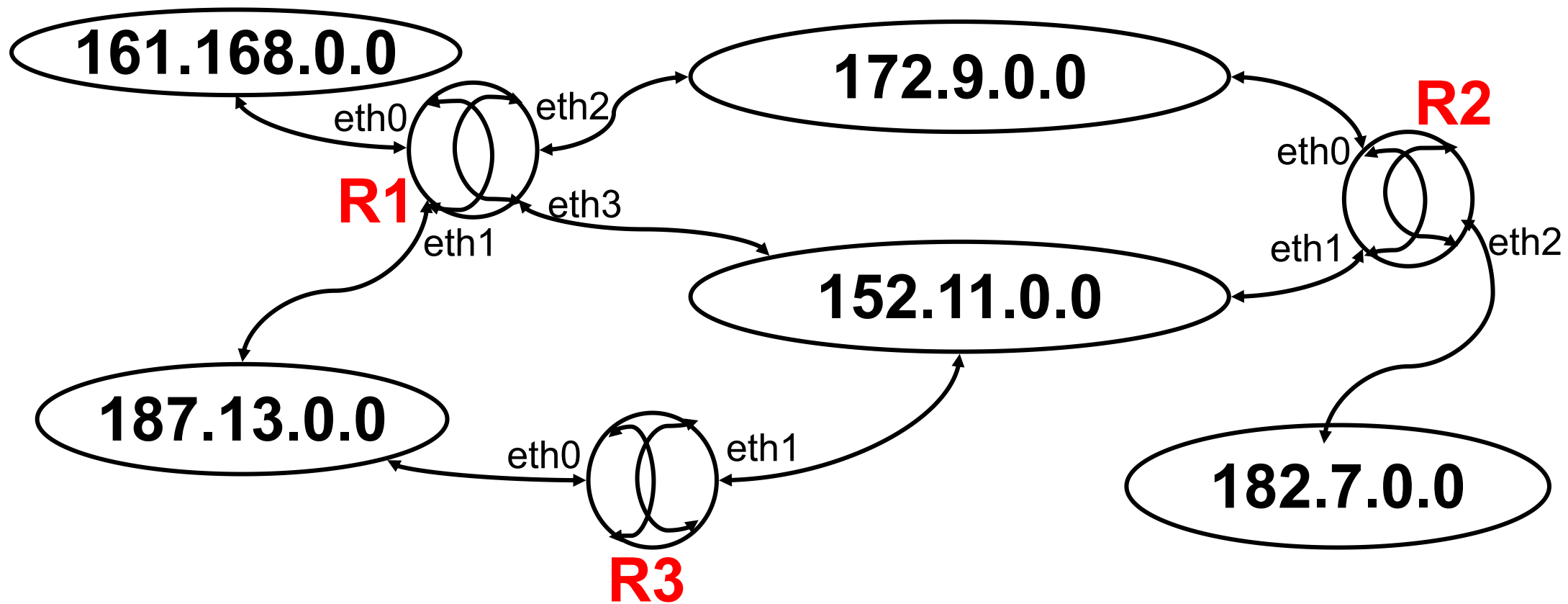
Destination	Distance	RNH
144.10.0.0	1	RD
137.9.0.0	1	RD
10.0.0.0	1	RD
153.10.0.0	2	R1(10.1.1.1)

- La table de routage de R1 devient :

Destination	Distance	RNH
<b>144.10.0.0</b>	<b>2</b>	<b>R0(10.2.2.2)</b>
<b>137.9.0.0</b>	<b>2</b>	<b>R0(10.2.2.2)</b>
<b>10.0.0.0</b>	<b>1</b>	<b>RD</b>
<b>153.10.0.0</b>	<b>1</b>	<b>RD</b>

# Exercice

- Donnez la table de routage aux différents routeur:





## Exercice

- Donnez les tables de routage de R1, R2 et R3.

On suppose que :

- R1 et R2 envoient leurs tables de routage à R3 .
- R3 et R2 envoient leurs tables de routage à R1 .
- R1 et R3 envoient leurs tables de routage à R2 .

## Table de routage de R3:

<b>Destination</b>	<b>Distance</b>	<b>RNH</b>
<b>187.13.0.0</b>	<b>1</b>	<b>RD</b>
<b>152.11.0.0</b>	<b>1</b>	<b>RD</b>
<b>182.7.0.0</b>	<b>2</b>	<b>152.11.1.1</b>
<b>172.9.0.0</b>	<b>2</b>	<b>152.11.1.1</b>
<b>161.168.0.0</b>	<b>2</b>	<b>187.13.1.1</b>

## Table de routage de R1:

Destination	Distance	RNH
161.168.0.0	1	RD
187.13.0.0	1	RD
152.11.0.0	1	RD
172.9.0.0	1	RD
182.7.0.0	<del>3</del> 2	<del>187.13.1.2</del> 172.9.1.1

## Table de routage de R2:

Destination	Distance	RNH
182.7.0.0	1	RD
152.11.0.0	1	RD
172.9.0.0	1	RD
187.13.0.0	2	172.9.1.2
161.168.0.0	2	172.9.1.2



- Cet algorithme est utilisé dans Internet et il s'appelle **Routing Information Protocol (RIP)**
- C'est un protocole de routage dynamique qui utilise le **nombre de sauts** comme mesure de la distance de routage .
- prend en compte la distance entre deux machines en termes de saut, mais ne considère pas l'état de la liaison afin de choisir la meilleure bande passante possible.

# Protocole RIP:

## *Routing Information Protocol*

- chaque routeur communique aux autres routeurs la distance qui les sépare (**nombre de saut**).
- chaque routeur qui reçoit ces infos, incrémente cette distance de 1 et communique le message aux routeurs directement accessibles.
- Les routeurs stocke l'@routeur suivant dans la TR de telle façon que le nombre de saut pour atteindre un réseau soit **minimal**.

# Protocole OSPF(Open Shortest Path First)

Chaque routeur distribue des informations sur son état du lien càd:

- ✓ interfaces utilisables,
- ✓ voisins accessibles,
- ✓ et le coût d'utilisation de chaque interface.

chaque routeur peut choisir la route la plus appropriée pour un message donné.

# **Les Sous** **réseaux**

# Les sous-réseaux

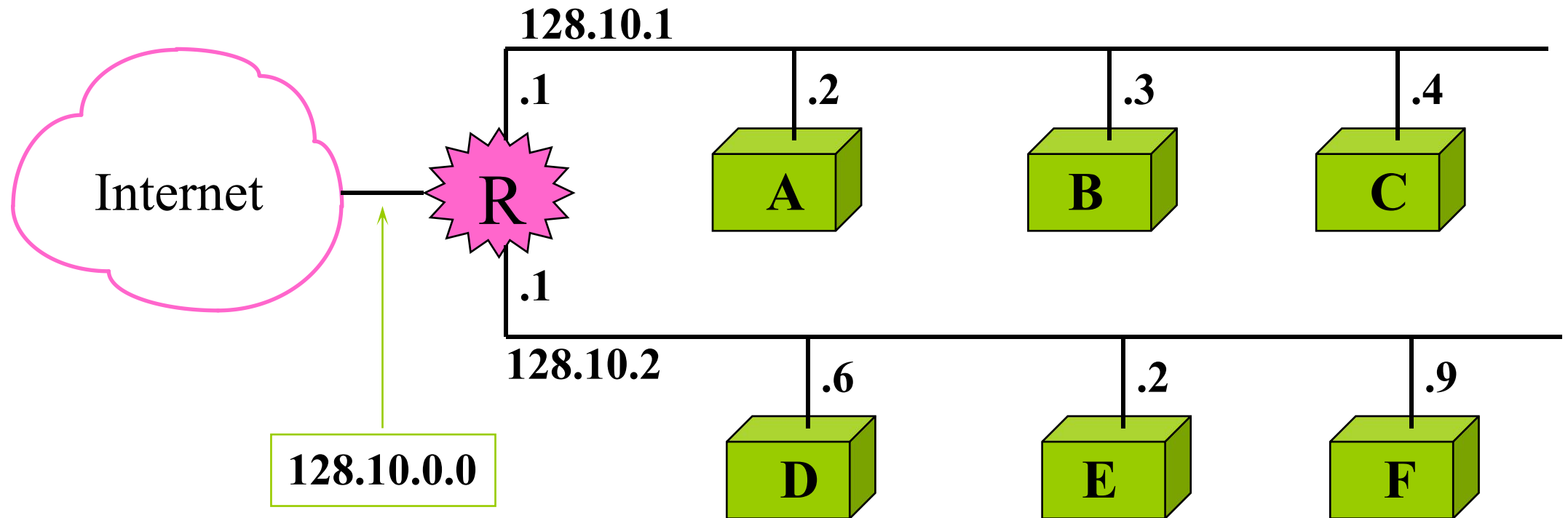
- Principes
  - A l'intérieur d'une entité associée à une @IP de classe A, B ou C, plusieurs réseaux physiques partagent cette @IP.
  - On dit alors que ces réseaux physiques sont des sous-réseaux (*subnet*) du réseau d'adresse IP.

# Les sous-réseaux

- Le sous-adressage est une extension du plan d'adressage initial
- Devant la croissance du nombre de réseaux de l'Internet, il a été introduit afin de limiter la consommation d'adresses IP
- il permet de diminuer :
  1. la gestion administrative des adresses IP,
  2. la taille des tables de routage des routeurs,
  3. la taille des informations de routage,
  4. le traitement effectué au niveau des routeurs.

# Les sous-réseaux

Les sous-réseaux 128.10.1.0 et 128.10.2.0 sont notés seulement avec le NetId, les machines seulement avec le Hostid ;  
exemple @IP(F) = 128.10.2.9



Ces deux sous-réseaux sont couverts par une seule @IP de classe B.  
R : accepte tout le trafic destiné au réseau 128.10.0.0 et sélectionne le sous-réseau en fonction du troisième octet de l'adresse destination.:

# Les sous-réseaux

- Le site utilise une seule adresse pour les deux réseaux physiques.
- A l'exception de R, tous routeur de l'Internet route comme s'il n'existait qu'un seul réseau.



# Les sous-réseaux

- Le routeur doit router vers l'un ou l'autre des sous-réseaux ; le découpage du site en sous-réseaux a été effectué sur la base du troisième octet de l'adresse
  - les adresses des machines du premier sous-réseau sont de la forme **128.10.1.X**,
  - les adresses des machines du second sous-réseau sont de la forme **128.10.2.X**.

# Les sous-réseaux

- Pour sélectionner l'un ou l'autre des sous-réseaux, R examine le troisième octet de l'adresse destination : si la valeur est 1, le datagramme est routé vers réseau 128.10.1.0, si la valeur est 2, il est routé vers le réseau 128.10.2.0.

# Les sous-réseaux

- Conceptuellement, la partie locale dans le plan d'adressage initial est subdivisée en “partie réseau physique” + “identification de machine (hostid) sur ce sous-réseau” :



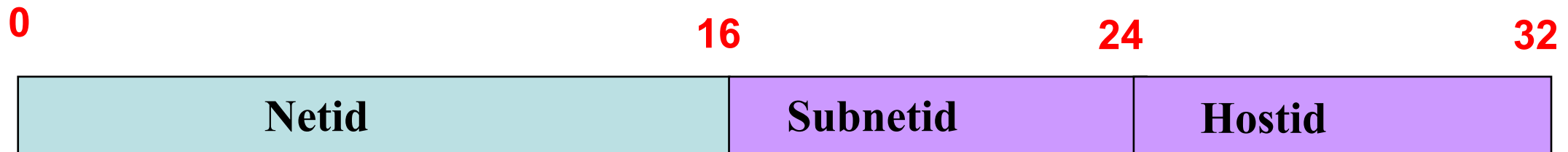
- ➡ «Partie Internet» correspond au NetId (plan d'adressage initial)
- ➡ «Partie locale» correspond au hostid (plan d'adressage initial)
- ➡ les champs «Réseau physique» et «identifieur Machine» sont de taille variable;
- ➡ la longueur des 2 champs étant toujours égale à la longueur de la «Partie locale».

# Les sous-réseaux

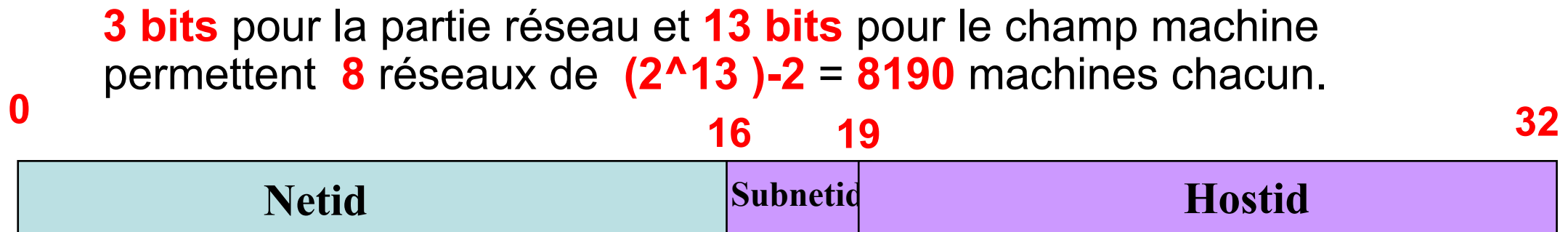
- Le choix du découpage dépend des perspectives d'évolution du site:

## Exemple

Classe B : **8 bits** pour les parties réseau et machine donnent un potentiel de **256** sous-réseaux et **254** machines par sous-réseau,



tandis que



# Les sous-réseaux

- Exemple :

Classe C : **4 bits** pour la partie réseau et **4 bits** pour le champ machine permettent **16** réseaux de **14** machines chacun.



Lorsque le sous-adressage est ainsi défini, toutes les machines du réseau doivent s'y conformer sous peine de dysfonctionnement du routage → configuration rigoureuse.

## Le masque de sous réseau

- Le masque de sous réseau est utilisé par les routeurs et les hôtes d'un sous réseau pour interpréter le champ hostid afin de déterminer combien de **bits sont utilisés pour la mise en sous réseau.**
- Le masque de sous réseau divise le champ hostid en un numéro de sous réseau et un numéro d'hôte.

## Le masque de sous réseau

- Le masque de sous réseau est un nombre de 32bits dont les valeurs sont obtenues à l'aide des règles suivantes:
  - Les chiffres 1 dans le masque de SR => à la position de l'id de réseau et du numéro de sous réseau dans l'@ IP.
  - Les chiffres 0 dans le masque de SR => à la position du numéro d'hôte dans l'@IP.

# Le masque de sous réseau

Exemple : réseau de classe B avec 8 bits pour le sous réseau

0 16 24 32

Netid	Subnetid	Hostid
-------	----------	--------

1111 1111 1111 1111 1111 1111 0000 0000

→ Masque = 255.255.255.0

Si on utilise une valeur de masque de sous-réseau de 255.255.0.0 pour un réseau de classe B → il n'y a pas de sous réseau.



## Exercice 1: masque de sous réseau aligné à l'octet.

- Utilisation d'une adresse de classe B avec un masque de sous réseau aligné à l'octet.

Adresse IP = 128.12.34.71

Masque de sous réseau = 255.255.255.0

### Question

Quelles sont les valeurs des :

Numéro de sous réseau = ????

Numéro d'hôte = ????

Adresse de diffusion dirigée = ????

## Solution 1

Adresse IP = 128.12.34.71

Masque de sous réseau = 255.255.255.0

128.12.34.71 = 1000 0000. 0000 1100. 0010 0010. 0100  
0111

255.255.255.0 = 1111 1111 1111 1111 1111 1111 0000 0000

---

128.12.34.0 = 1000 0000. 0000 1100. 0010 0010. 0000  
0000

Les valeurs sont:

Numéro de SR = **0.0.34.0**

Numéro d'hôte = **0.0.0.71**

Adresse de diffusion dirigée = **128.12.34.255**

## Exercice 2 : Masques de sous réseaux non alignés à l'octet

- Utilisation d'une adresse de classe C avec un masque de sous réseau non aligné à l'octet.

Adresse IP = 192.55.12.120

Masque de sous réseau = 255.255.255.240

### Question

Quelles sont les valeurs des :

Numéro de sous réseau = ????

Numéro d'hôte = ????

Adresse de diffusion dirigée = ????

## Solution 2

Adresse IP = 192.55.12.120

Masque de sous réseau = 255.255.255.240

192.55.12.120	1100 0000	0011 0111	0000 1100	0111 1000
255.255.255.240	1111 1111	1111 1111	<b>1111 1111</b>	<b>1111 0000</b>
@ IP du sous-réseau	1100 0000 <b>192</b>	0011 0111 <b>55</b>	0000 1100 <b>12</b>	<b>0111 0000</b> <b>112</b>
Numéro de sous-réseau	0000 0000 <b>0</b>	0000 0000 <b>0</b>	0000 0000 <b>0</b>	<b>0111 0000</b> <b>112</b>
Numéro d'hôte	0000 0000 <b>0</b>	0000 0000 <b>0</b>	0000 0000 <b>0</b>	0000 <b>1000</b> <b>8</b>
Adresse de diffusion dirigée	1100 0000 <b>192</b>	0011 0111 <b>55</b>	0000 1100 <b>12</b>	<b>0111 1111</b> <b>127</b>

# Masques de sous réseaux non alignés à l'octet

Dans la plus part des réseaux on trouve les valeurs décimales suivantes pour le dernier octet du masque de sous réseau

Taille de sous réseau en bits	Motif binaire	Valeur décimale
1	1000 0000	128
2	1100 0000	192
3	1110 0000	224
4	1111 0000	240
5	1111 1000	248
6	1111 1100	252
7	1111 1110	254

## Exercice

Quelles adresses IP se trouvent sur le même sous réseau que **130.12.127.231** si le masque de sous réseau est **255.255.192.0** ?

1. **130.45.130.1**
2. **130.22.130.1**
3. **130.12.64.23**
4. **130.12.167.127**

## Solution

$$\begin{array}{rcll} 130.12.127.231 & = & 130 & . 12 & . 0111\ 1111 & . 231 \\ 255.255.192.0 & = & 255 & . 255 & . 1100\ 0000 & . 0 \end{array}$$

---

$$130.12.64.0 = 130 . 12 . 0100\ 0000 . 0$$

---

1. 130.45.130.1

2. 130.22.130.1

3. 130.12.64.23

4. 130.12.167.127

# Le routage dans les sous réseaux

L'algorithme de routage IP utilisé par les routeurs et les hôtes, ne prenait pas en compte les masques de SR, les info de base dans une TR sont constituées des entrées suivantes :

**(adresse de destination , adresse de prochain pas)**

Adresse de destination = adresse de réseau ou d'hôte spécifié

Adresse de prochain pas = n'est pas le chemin complet jusqu'à destination, mais l'adresse du routeur vers lequel on doit transmettre le datagramme.



# Le routage dans les sous réseaux

Avec un sous réseau, la table de routage doit posséder une information supplémentaire pour indiquer les masques de sous réseaux

**(adresse de destination,  
masque de sous réseau,  
adresse de prochain pas)**

## Exemple

Soit **2 datagrammes** dont les @IP de destinations sont les suivantes: (avec un mask = **255.255.255.0**)

**144.19.74.12**

**144.19.75.21**

On détermine la partie réseau de ces adresses:

**144.19.74.12   &&   255.255.255.0   =   144.19.74.0**

**144.19.75.21   &&   255.255.255.0   =   144.19.75.0**

## Exemple

Soit l'entrée de la table de routage suivante :

<b>144.19.74.0</b>	<b>255.255.255.0</b>	<b>144.19.74.91)</b>
--------------------	----------------------	----------------------

- ➔ On trouve dans la table de routage une correspondance pour **144.19.74.12** mais pas pour **144.19.75.21**
- ➔ On utilise la route par défaut **0.0.0.0** avec un masque de sous réseau **0.0.0.0** (toute adresse IP de destination sera valide pour cette entrée)
- ➔ Les routes spécifiques aux hôtes ont un masque de sous réseau **255.255.255.255** (tous les bits de l'adresse IP sont significatifs pour le routage)

## Exemple : table de routage contenant des routes par défaut et des routes spécifiques pour hôtes

Adresse de destination	Masque de sous réseau	Adresse de prochain pas
144.19.74.0	255.255.255.0	144.19.74.91
<b>145.12.2.101</b>	<b>255.255.255.255</b>	<b>144.19.74.92</b>
<b>202.33.23.3</b>	<b>255.255.255.255</b>	<b>144.19.74.93</b>
0.0.0.0	0.0.0.0	144.19.74.91
0.0.0.0	0.0.0.0	144.19.74.94

- ➔ 2 routes spécifiques à un hôte **145.12.2.101** et **202.33.23.3**
- ➔ 2 routes par défaut pointant vers les routeurs de prochain pas **144.19.74.91** et **144.19.74.94**

### Exercice :

Quelle est la route trouvée à partir de cette table de routage pour un datagramme IP portant l'adresse de destination :

1. 202.33.23.3
2. 201.3.3.3

## Notation /n

/n à la suite d'une adresse IP indique que les n premiers bits servent au routage. n représente donc la longueur du masque de réseau.

### Exemples :

/8	255.0.0.0	11111111. 00000000. 00000000.00000000
/16	255.255.0.0	11111111. 11111111. 00000000.00000000
/19	255.255.224.0	11111111. 11111111. 11100000.00000000

# Algorithme de routage IP complet avec sous réseaux

1. à partir du champ **@IPDest**, on détermine la partie réseau **PrefixeReseau** en effectuant un **ET** bit à bit avec le masque sous réseau figurant dans l'entrée de table de routage.
2. **Si PrefixeReseau** correspond à l'ID d'un réseau directement connecté → **@IPDest** est celle d'un hôte sur ce réseau → **RD** datagramme IP.
3. **Si** aucune correspondance pour le **PrefixeReseau**, on examine la table de routage →  
**Si** cette entrée existe, on transmet le paquet vers le routeur de prochain pas indiqué dans l'entrée.
4. **Sinon** (pas de correspondance dans la table de routage) on cherche dans la table de routage l'entrée par défaut 0.0.0.0 →  
**Si** cette entrée existe on transmet le datagramme IP vers le routeur de prochain pas indiqué pour cette entrée.
5. **Sinon** le datagramme IP ne peut être routé → on signale une erreur de routage aux protocoles de couches supérieures