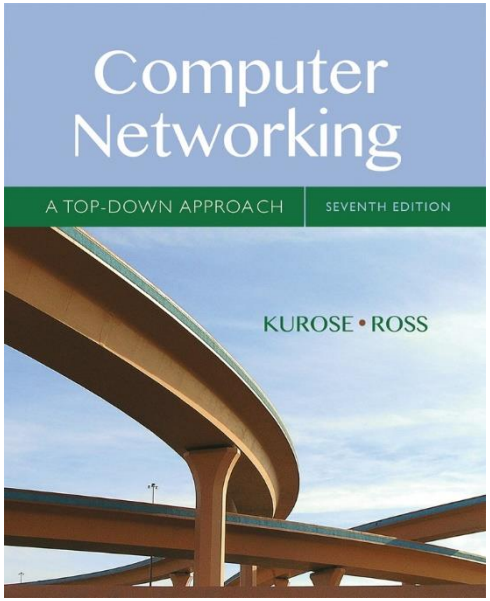
Chapitre IV
La couche réseau

Le plan de données



Computer Networking: A Top Down Approach
7ème édition
Jim Kurose, Keith Ross
Addison-Wesley
2017

A note on the use of these Powerpoint slides:
We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

Chapitre IV: plan

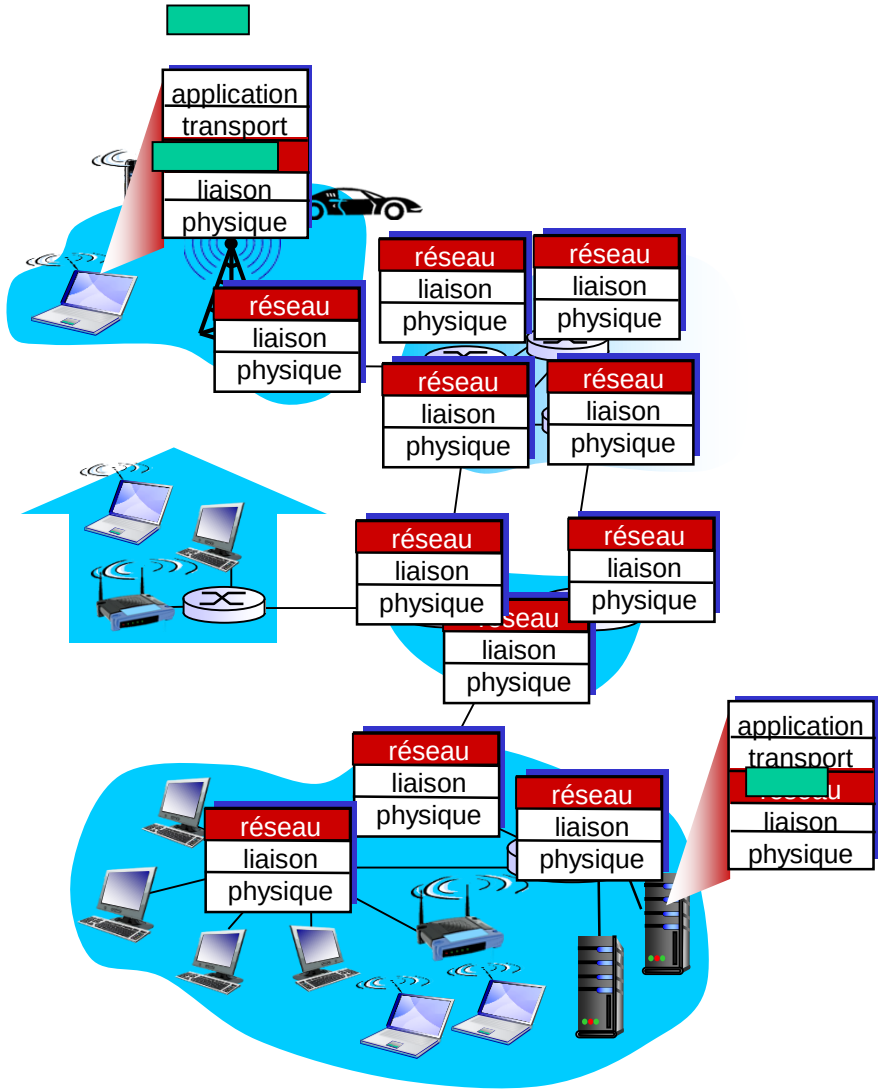
4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

La couche réseau

- transporte les segments de l'émetteur vers le récepteur
- côté émission: encapsule les segments en datagrammes
- côté réception: délivre les segments vers la couche transport
- les protocoles réseaux sont dans toutes les hôtes et les routeurs
- le routeur examine l'entête de tous les datagrammes IP qui le traversent



Deux fonctionnalités de la couche réseau

- *transfert:* déplacer les paquets d'une entrée du routeur vers la sortie appropriée
- *routage:* déterminer la route à prendre par les paquets de la source vers la dest.
  - *algorithmes de routage*

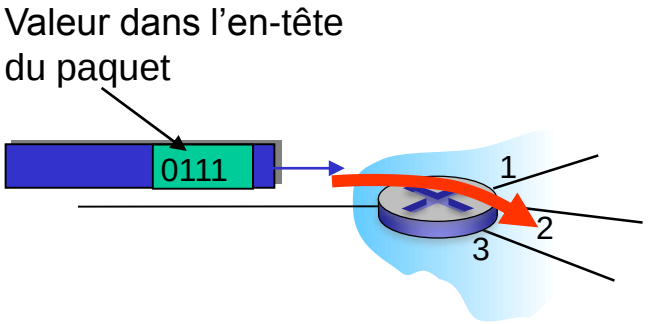
analogie: voyage

- *routage:* processus de planifier un voyage de la source vers la dest.
- *transfert:* processus de traverser un point d'interconnexion

Plan de données, plan de contrôle

Plan de données

- local
- détermine comment le datagramme qui arrive au port d'entrée est transféré au bon port de sortie
- fonction "transfert"

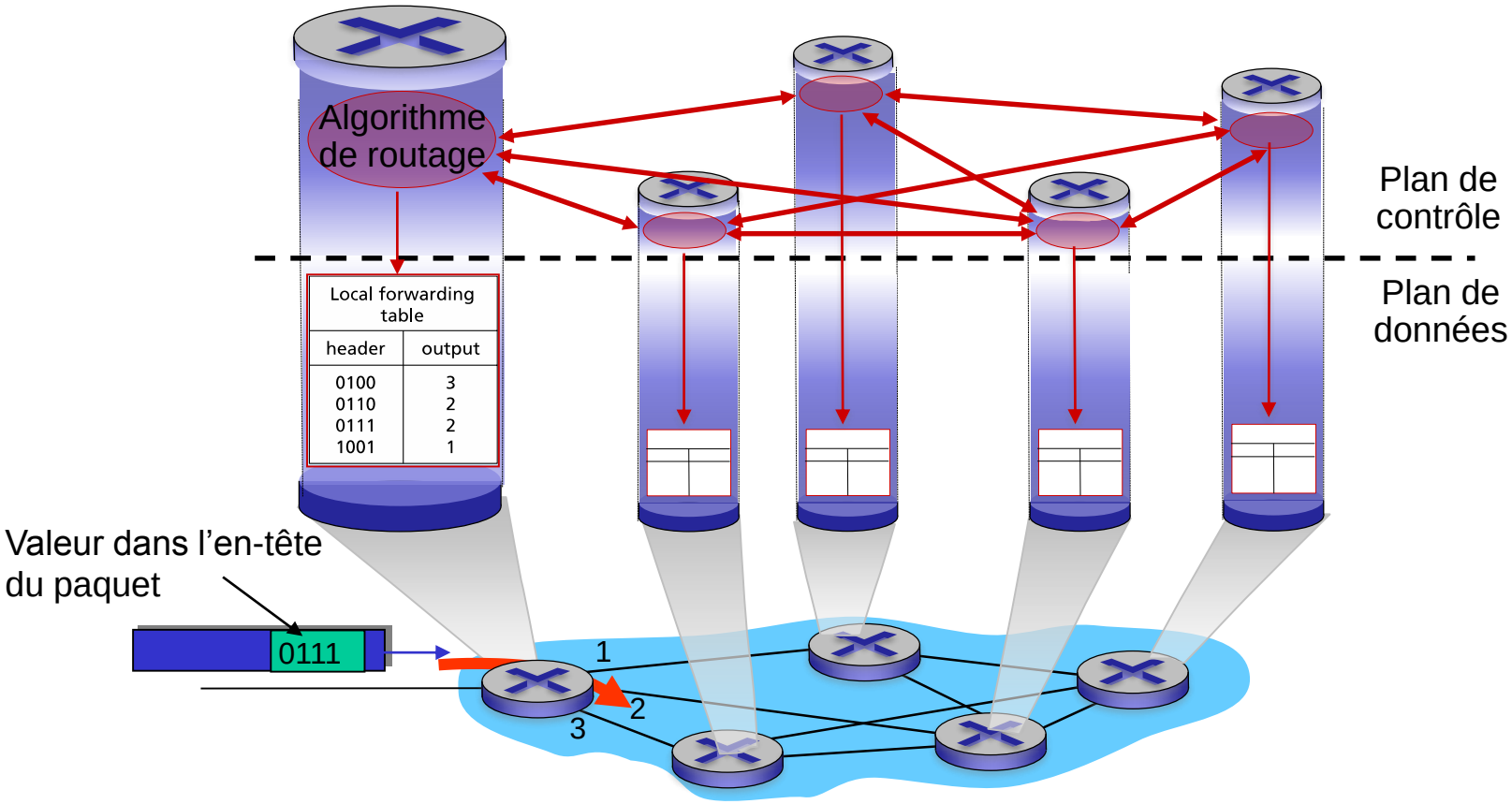


Plan de contrôle

- vue globale du réseau
- détermine la route (la série des routeurs) que doit prendre le datagramme de la source à la destination
- deux approches:
  - *Algorithmes de routage traditionnels:* implémentés dans les routeurs
  - *software-defined networking (SDN)*: implémenté au niveau de serveurs distants

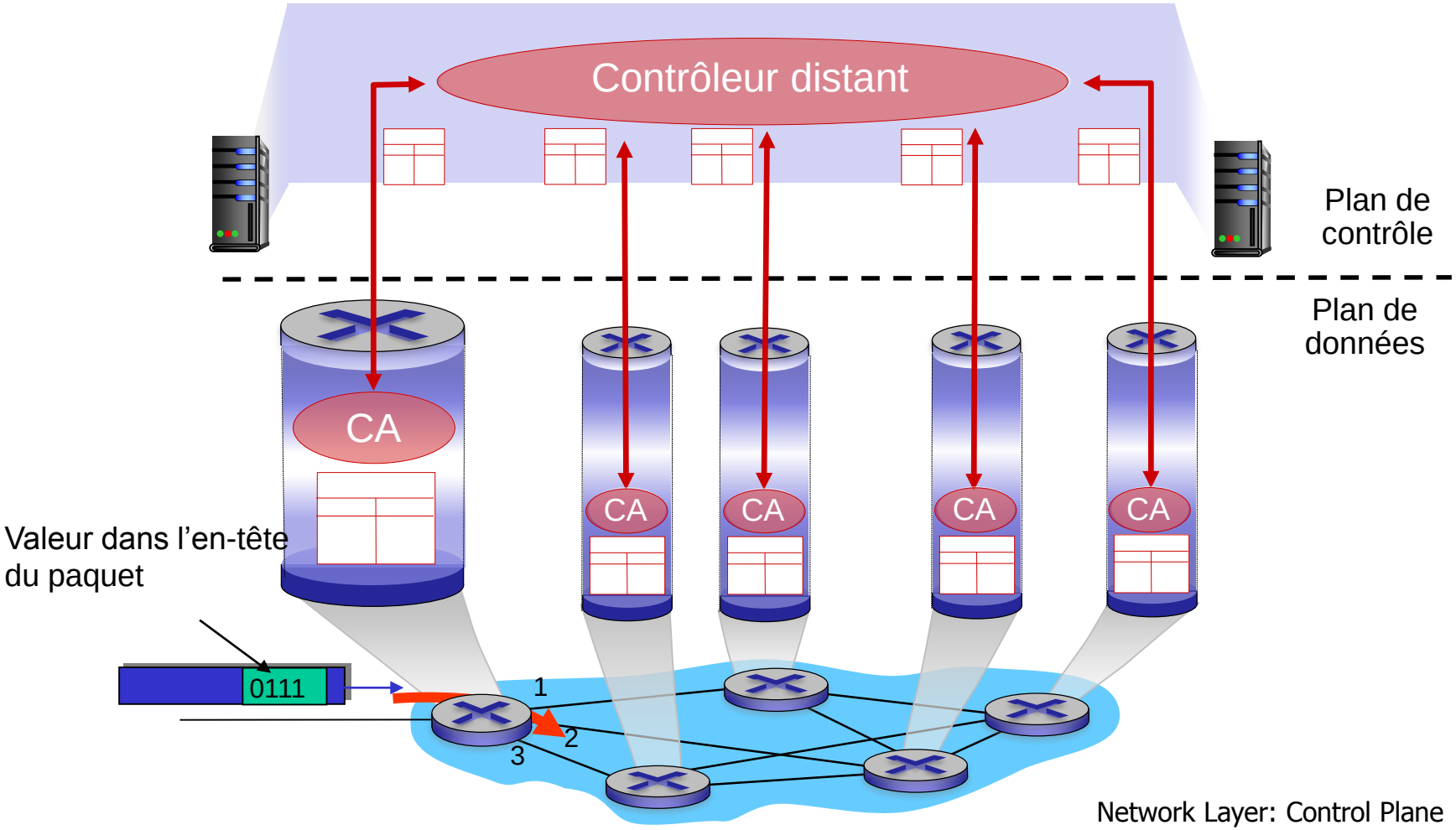
Plan de contrôle (dans chaque routeur)

Chaque routeur implémente un algorithme de routage. Les routeurs intéragissent entre eux.



Plan de contrôle (centralisé)

Un contrôleur distant interagit avec les agents de contrôle (CA) implémentés dans chaque routeur



Services de la couche réseau

Q: Quels sont les services que peut assurer la couche réseau pour le transport des datagrammes?

exemples de services par datagramme:

- garantie de livraison
- garantie de livraison dans un délai précis

exemples de services par flux de datagrammes:

- livraison en ordre
- garantie d'un débit minimal pour le flux
- garantie de la gigue (délai entre deux paquets successifs)

Comparaison des services

architecture réseau	modèle de services	Garanties ? débit	perte	ordre	délai	info sur la congestion
Internet	au mieux	non	non	non	non	non
ATM	CBR	taux constant	oui	oui	oui	pas de congestion
ATM	VBR	garantie	oui	oui	oui	pas de congestion
ATM	ABR	garantie minimum	non	oui	non	oui
ATM	UBR	non	non	oui	non	non

Chapitre IV: plan

4.1 introduction
- plan de données
- plan de contrôle

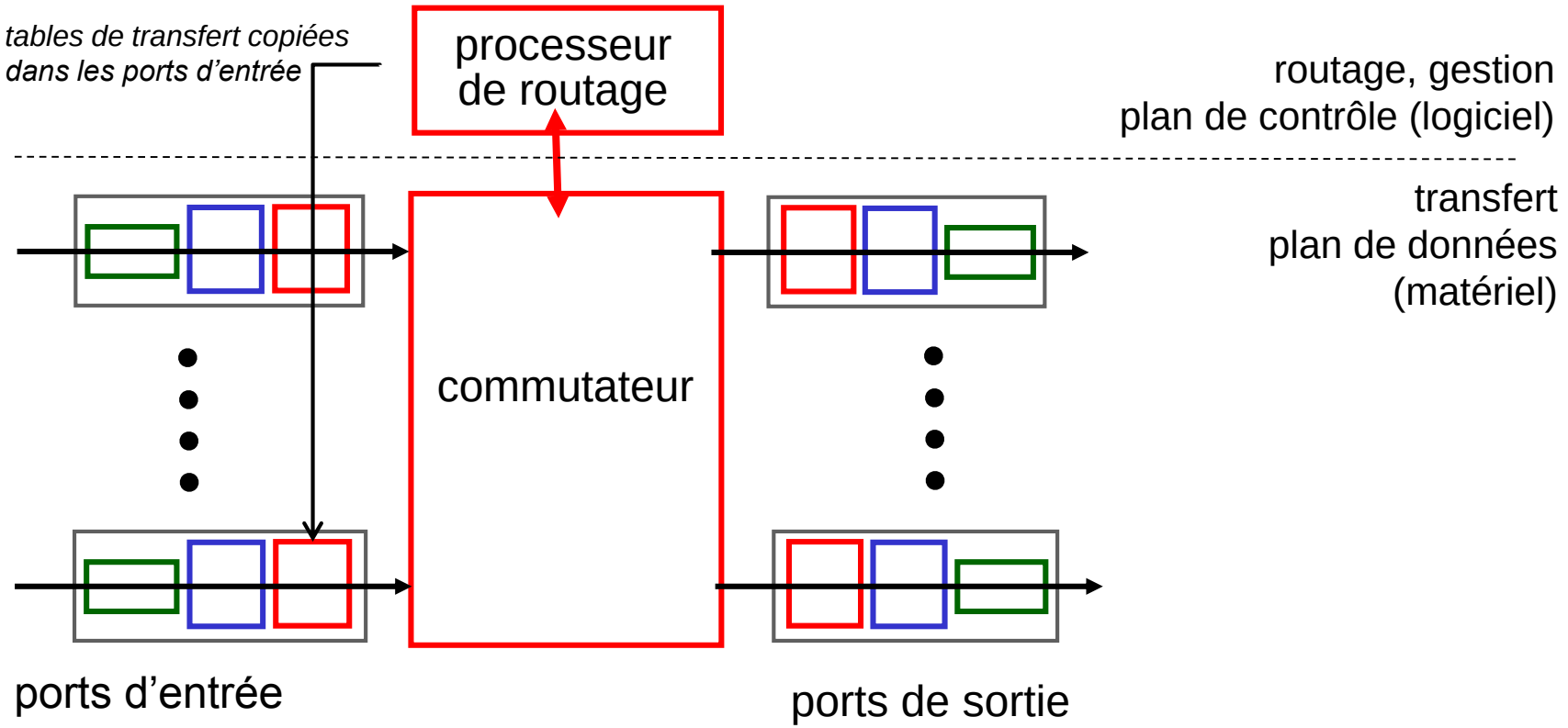
4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

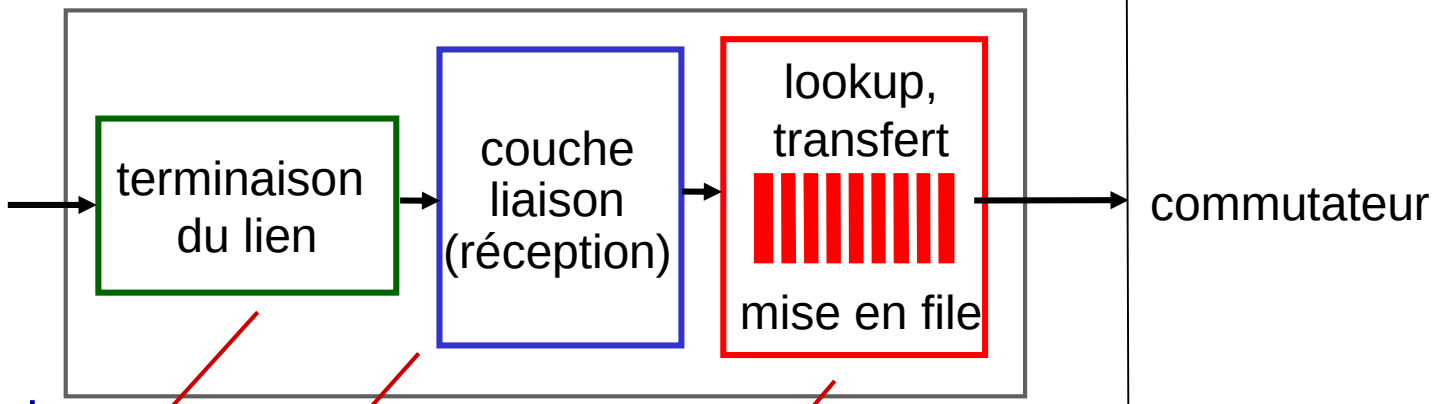
Architecture d'un routeur

Deux fonctions principales:

- exécuter un algorithme de routage (RIP, OSPF, BGP)
- *transférer des datagrammes (entrée vers sortie)*



Fonctionnement du port d'entrée



couche physique:
réception de bits

couche liaison:
ex., Ethernet

commutation décentralisée:

- en utilisant l'adresse de dest., et la table de routage, il trouve le port de sortie « match and action »
- objectif: compléter le traitement du port d'entrée rapidement 'line speed'
- mise en file: si les paquets arrivent plus rapidement que le taux de transfert vers le commutateur

Transfert selon la destination

Table de transfert

Intervalle d'adresses de destination	Interface
11001000 00010111 00010000 00000000 à 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 à 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 à 11001000 00010111 00011111 11111111	2
sinon	3

Transfert selon la destination

Règle du plus long préfixe

L'entrée dans la table de tranfert qui correspond à la destination est celle qui partage le plus long préfixe avec l'adresse de destination

Intervalle d'adresses de destination	Interface
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
sinon	3

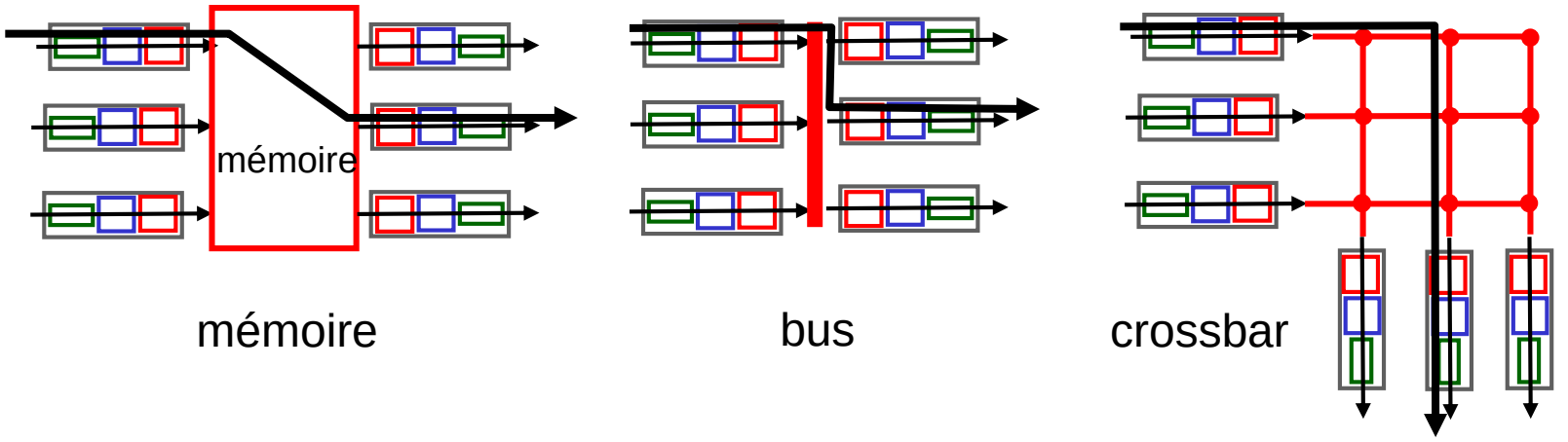
exemples:

DA: 11001000 00010111 00010110 10100001 Quel interface?

DA: 11001000 00010111 00011000 10101010 Quel interface?

Commutateur

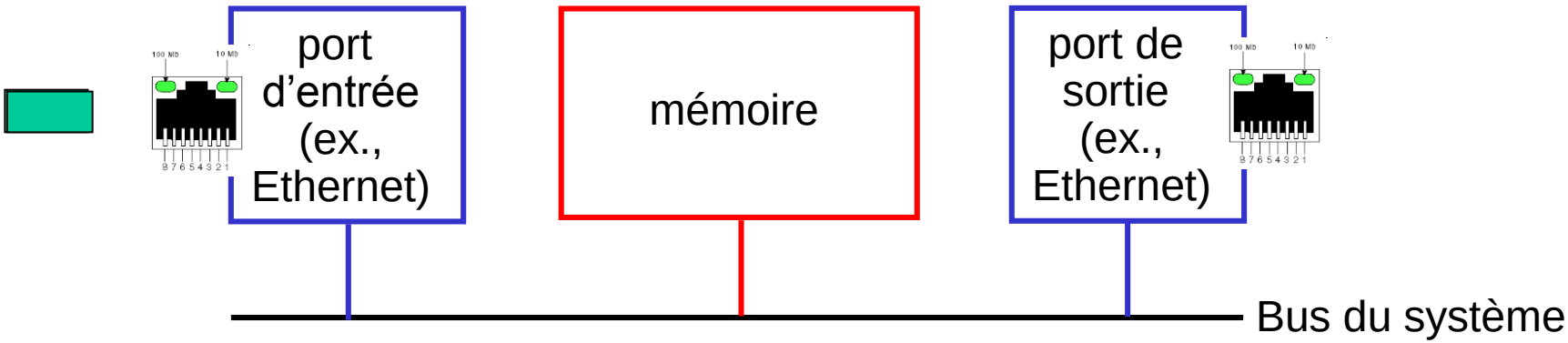
- transférer le paquet du buffer d'entrée vers le buffer de sortie approprié
- taux (vitesse) de commutation: taux selon lequel les paquets passent de l'entrée à la sortie
- trois types de commutateurs



Commutation par mémoire

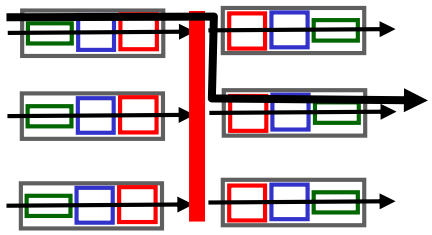
Les routeurs de 1ère génération:

- des ordinateurs traditionnels avec une commutation sous contrôle directe de la CPU
- le paquet est copié dans le mémoire du système
- la vitesse est limitée par la bande du mémoire (2 traversées du bus par datagramme)



Commutation par le bus

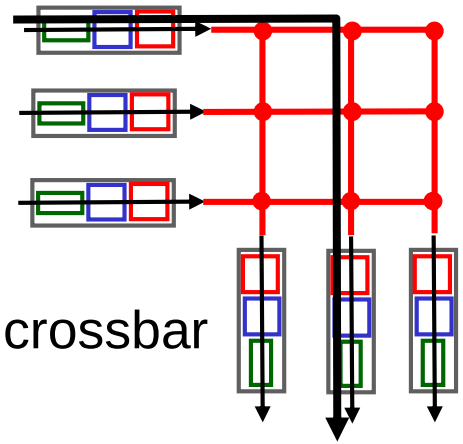
- le datagramme se déplace du mémoire du port d'entrée au mémoire du port de sortie à travers un bus partagé
- *contention dans le bus:* la vitesse de commutation est limitée par la bande du bus
- bus de 32 Gbps, Cisco 5600: vitesse suffisante pour les routeurs des entreprises et d'accès



bus

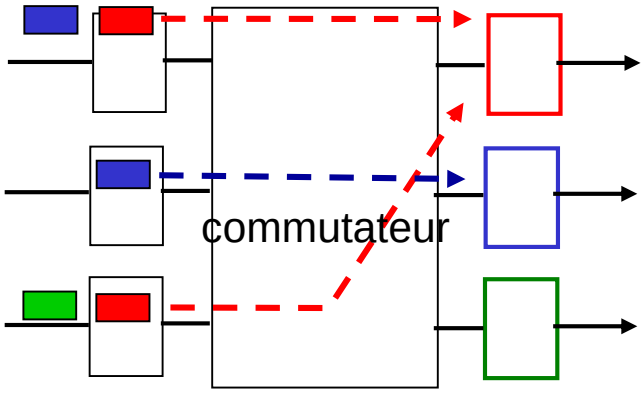
Commutation par un réseau d'interconnexion

- contourne les limitations de la bande du bus
- réseaux banyan, crossbar, d'autres réseaux d'interconnexions initialement développés pour connecter des processeurs
- conception avancée : fragmenter le datagramme en des cellules de taille fixe, commuter les cellules à travers la fabrique.
- Cisco 12000: commutent 60 Gbps à travers un réseau d'interconnexion

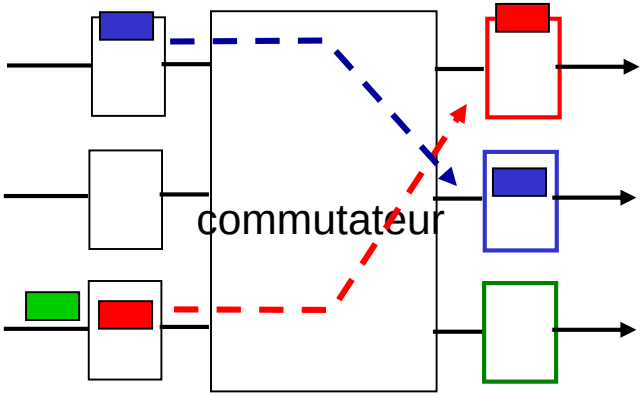


Mise en file à l'entrée

- si le commutateur est plus lent -> mise en file aux ports d'entrées
  - *délai et perte à cause du débordement des tampons d'entrée!*
- blocage en tête de ligne: les datagrammes en tête d'une file d'attente ne permettent pas le services des autres

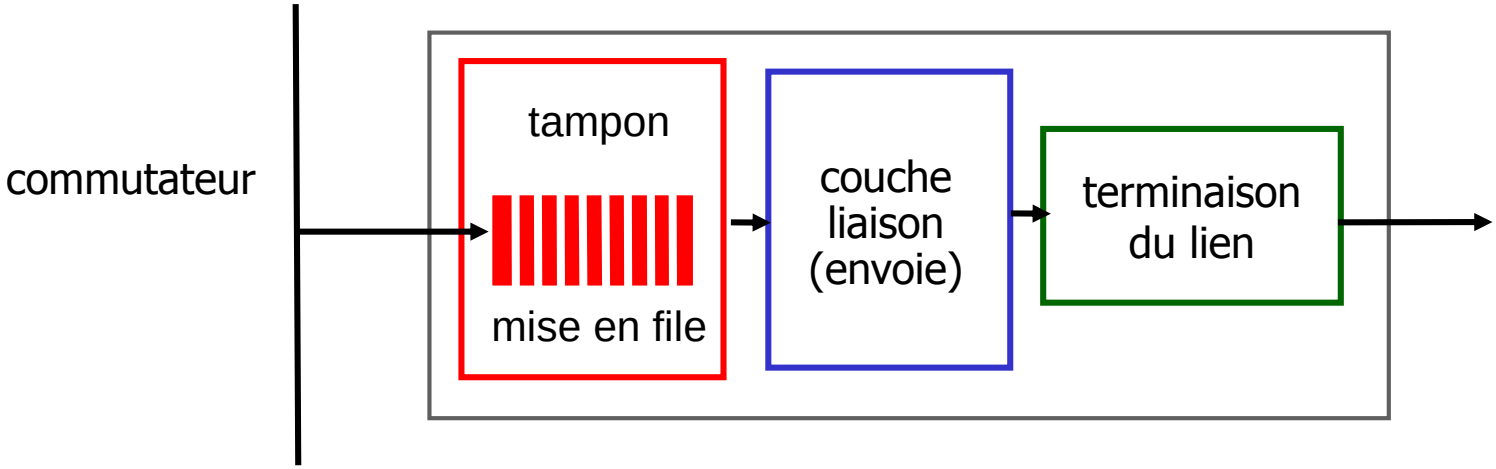


un seul paquet rouge peut passer.
le paquet rouge du bas est bloqué



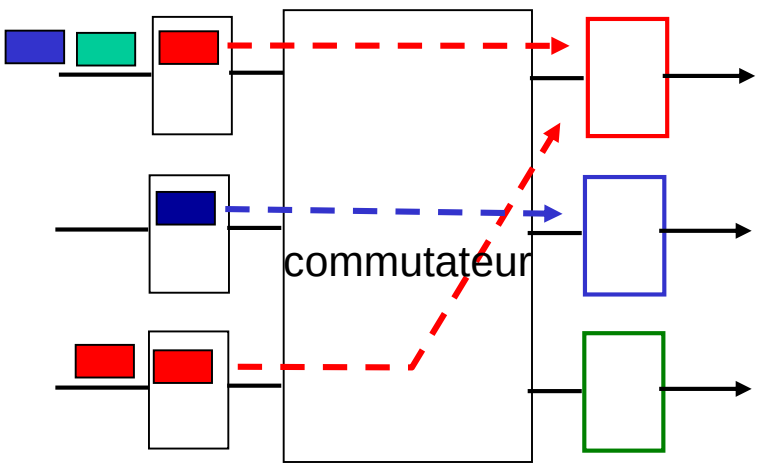
après: le paquet vert *est aussi bloqué*

Port de sortie

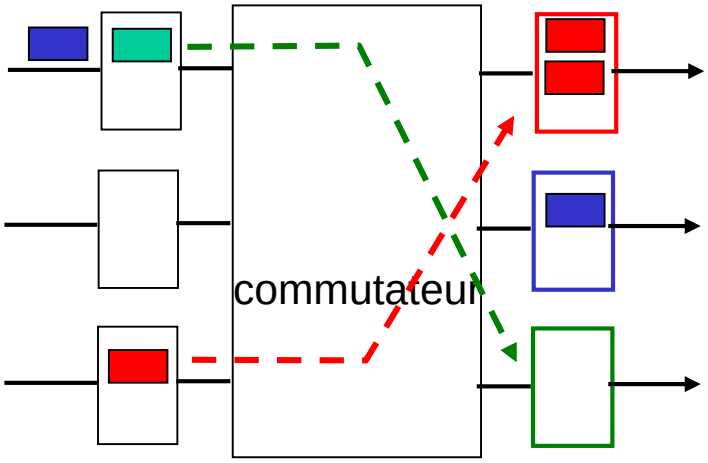


- *mise en tampon:* exigé lorsque le taux d'arrivée à partir du commutateur est plus grand que le taux de transmission ➔ perte
- *ordonnancement* choisir le datagramme à envoyer ➔ comment?

Mise en file à la sortie



à l'instant *t*, plusieurs paquets *vers la sortie*

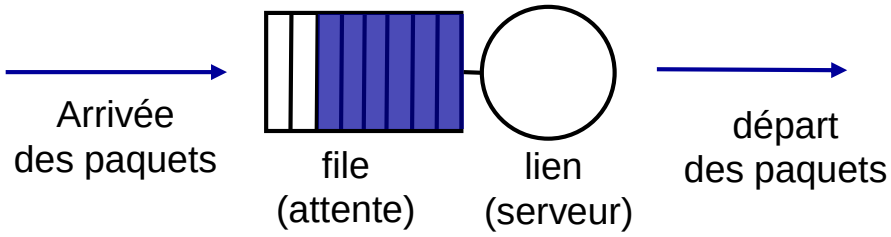


après commutation

- mise en mémoire quand le taux d'arrivée via le commutateur dépasse la vitesse de sortie
- *possiblement: attente (délai) et perte due à la surcharge de tampon de sortie!*

Mécanismes d'ordonnancement

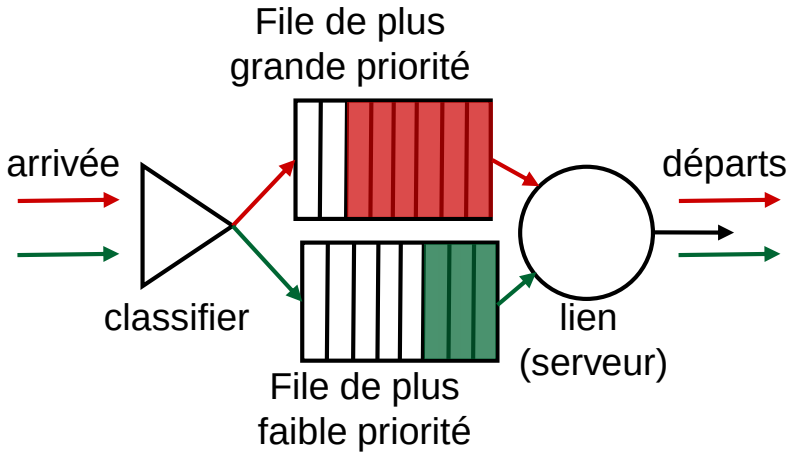
- *Ordonnancement:* choisir le prochain paquet à envoyer
- *Ordonnancement FIFO (first in first out):* choisir selon l'ordre d'arrivée à la file
  - *politique d'élimination:* si un paquet arrive à une file déjà pleine: quel paquet éliminer?
    - *tail drop:* éliminer le paquet qui vient d'arriver
    - *priorité:* éliminer selon la priorité
    - *random:* éliminer aléatoirement

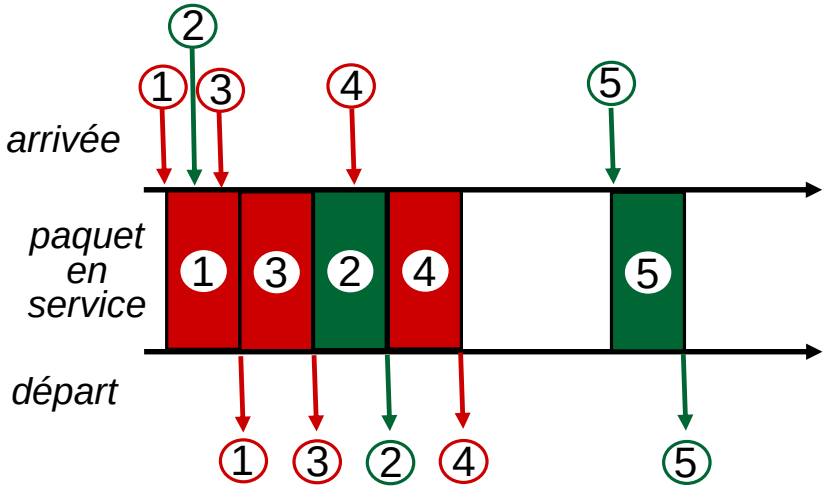


Ordonnancement par priorité

Envoyer le paquet qui dispose de la plus haute priorité

- Plusieurs *classes*, avec différentes priorités
  - Différencier les classes par un marquage ou une information au niveau de l'en-tête IP

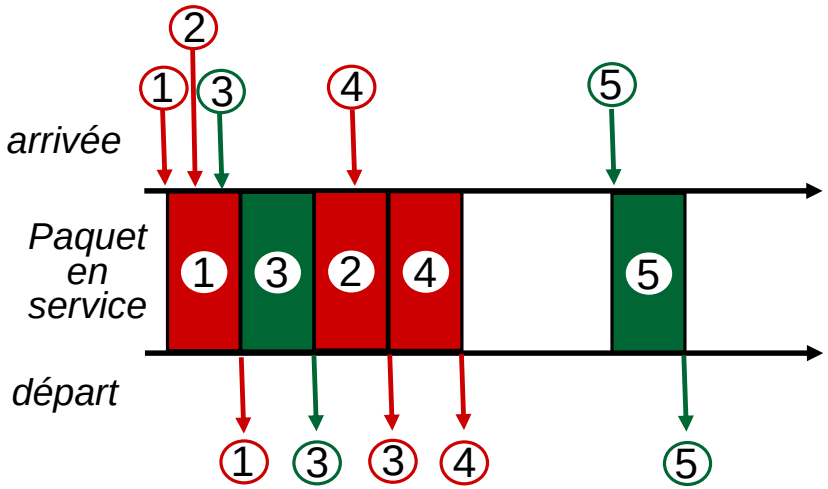




Autres mécanismes d'ordonnancement

Ordonnancement Round Robin (RR):

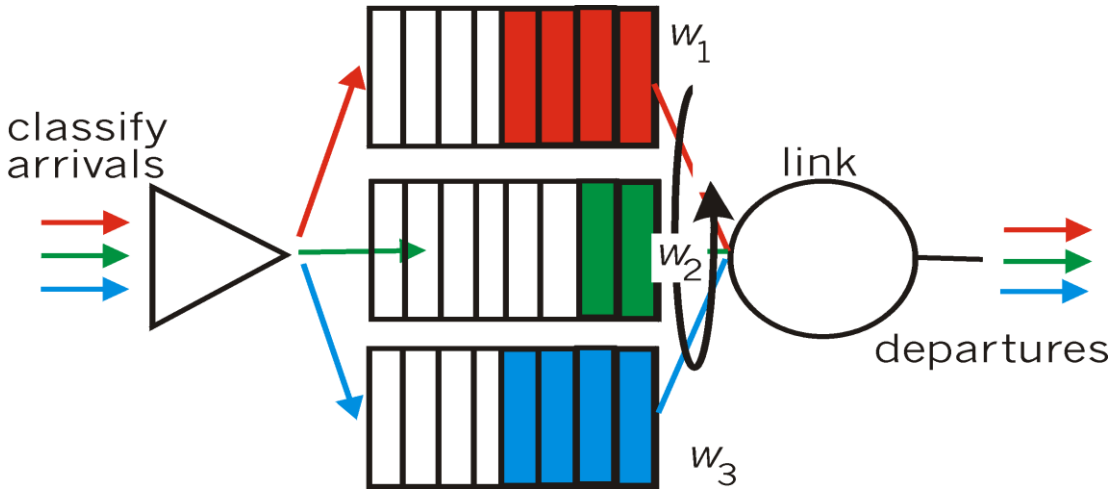
- Plusieurs classes
- vérifier les files d'une manière cyclique et envoyer un paquet de chaque classe (si disponible)



Autres mécanismes d'ordonnancement

Weighted Fair Queuing (WFQ):

- généralisation du Round Robin
- chaque classe reçoit un service proportionnel à son poids durant chaque cycle



Chapitre IV: plan

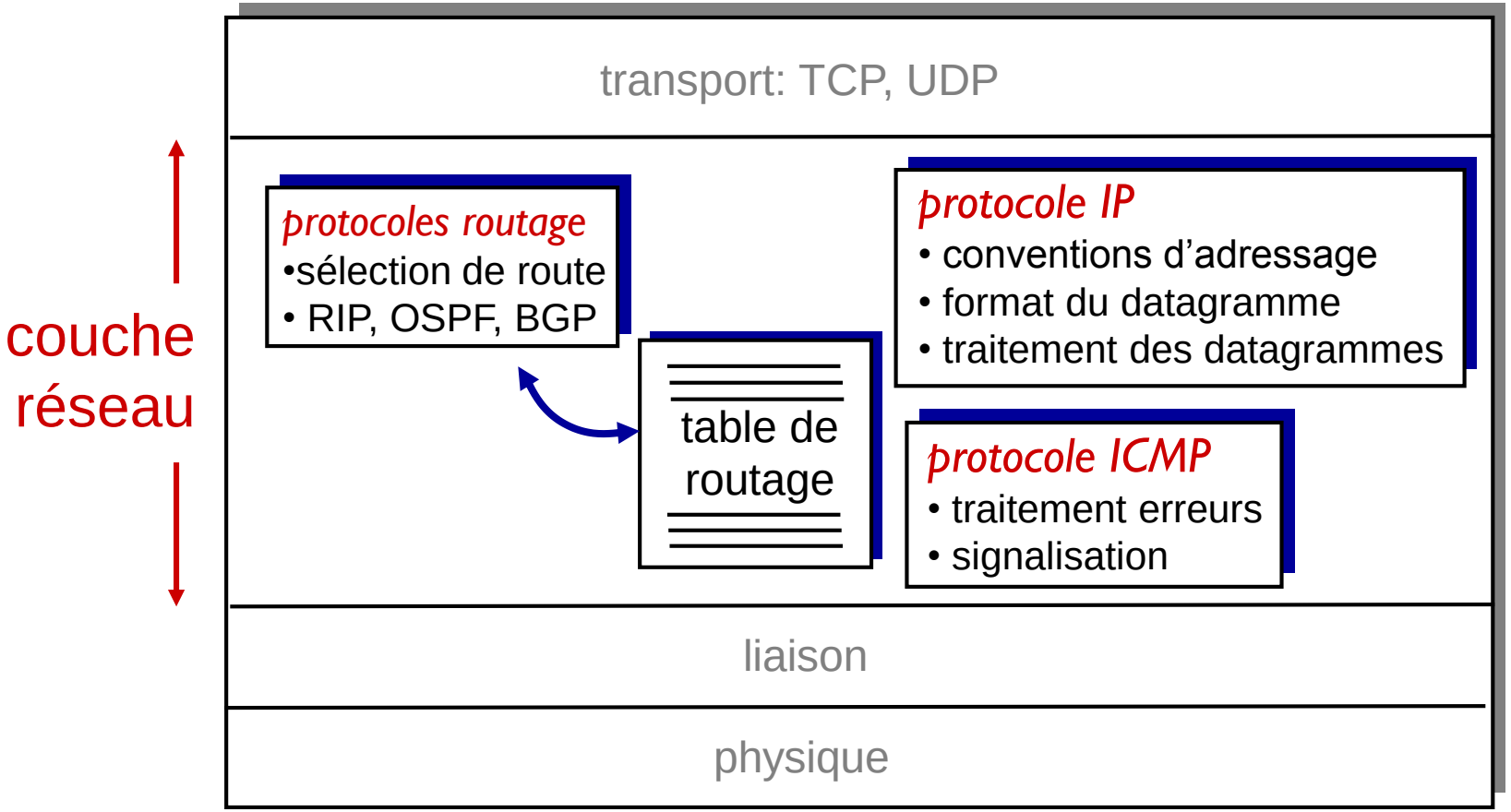
4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

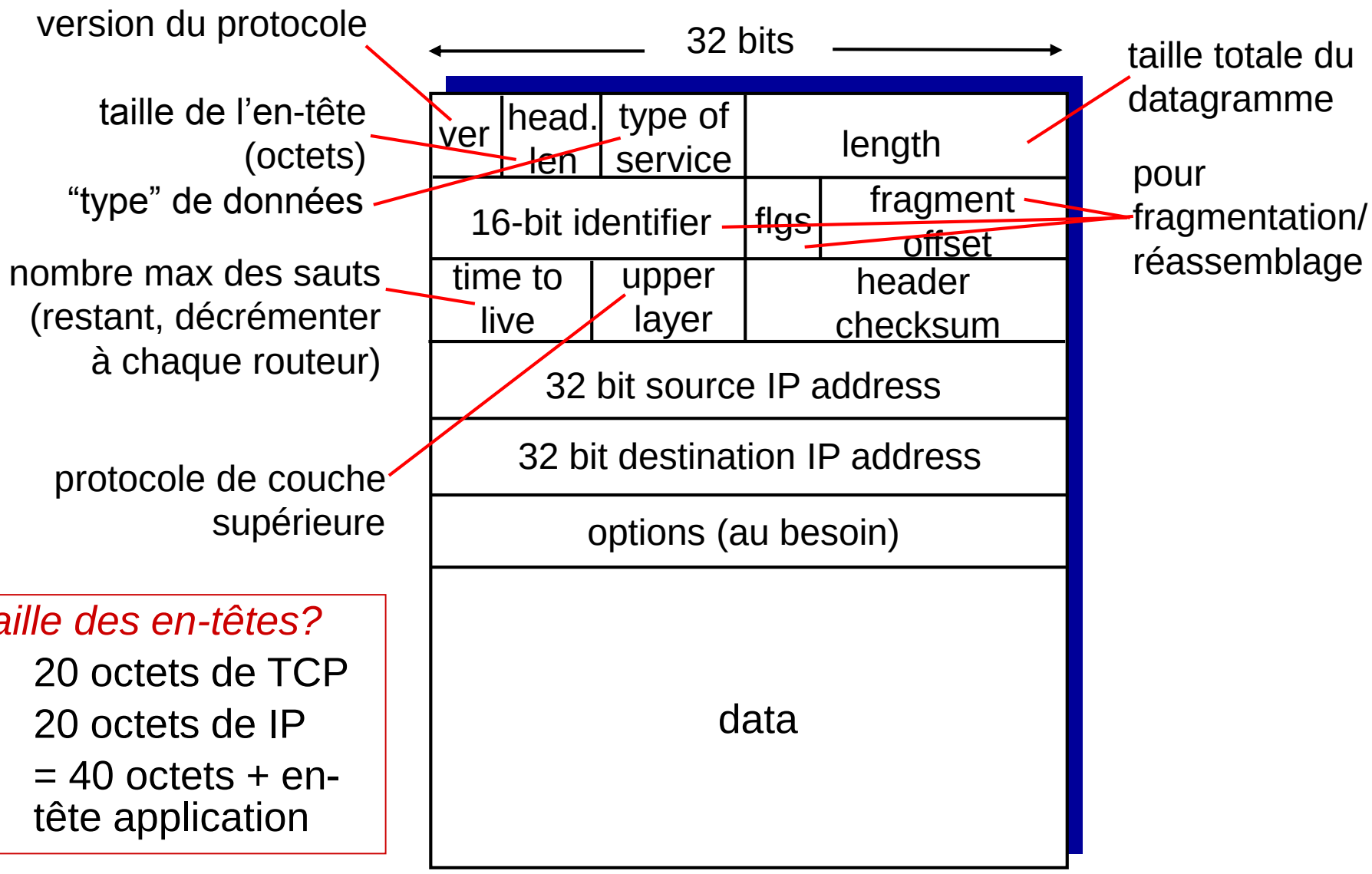
4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

La couche réseau de l'Internet

fonctions de la couche réseau:



Format du datagramme IP

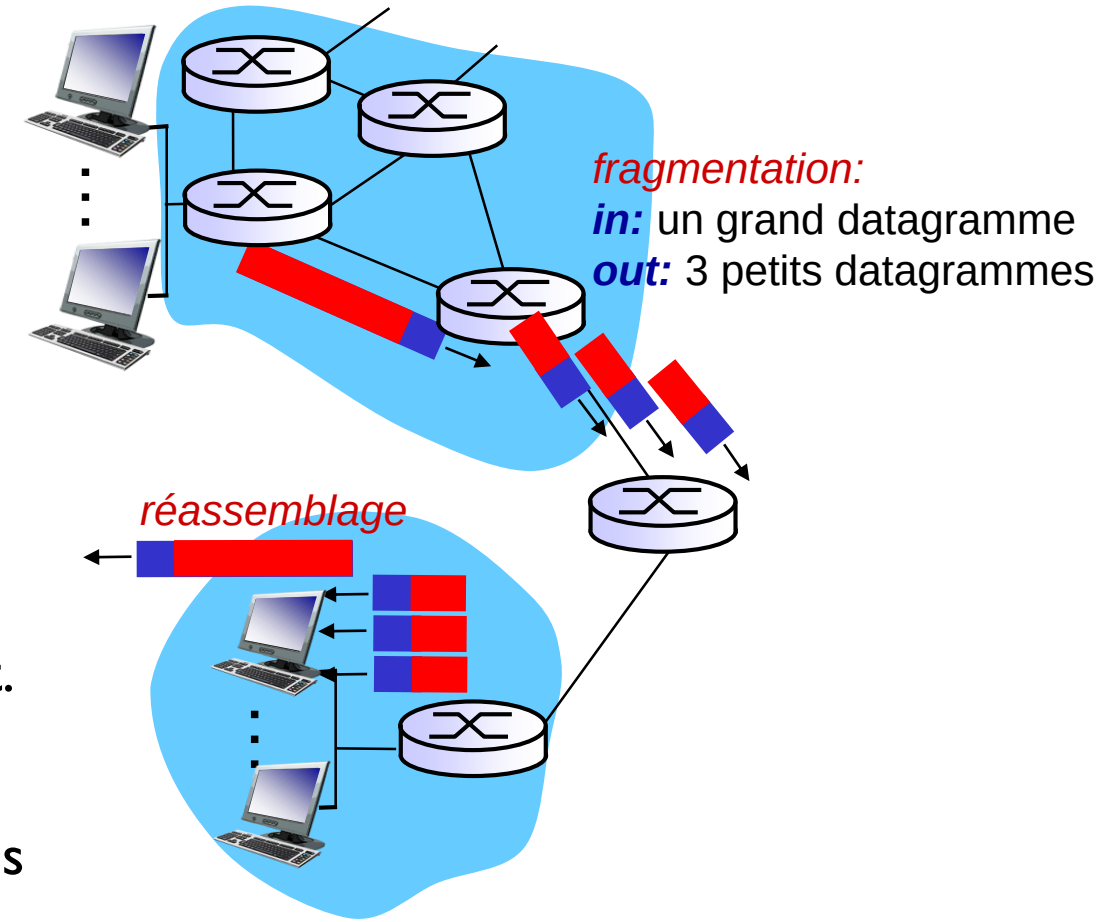


taille des en-têtes?

- 20 octets de TCP
- 20 octets de IP
- = 40 octets + en-tête application

Fragmentation IP, réassemblage

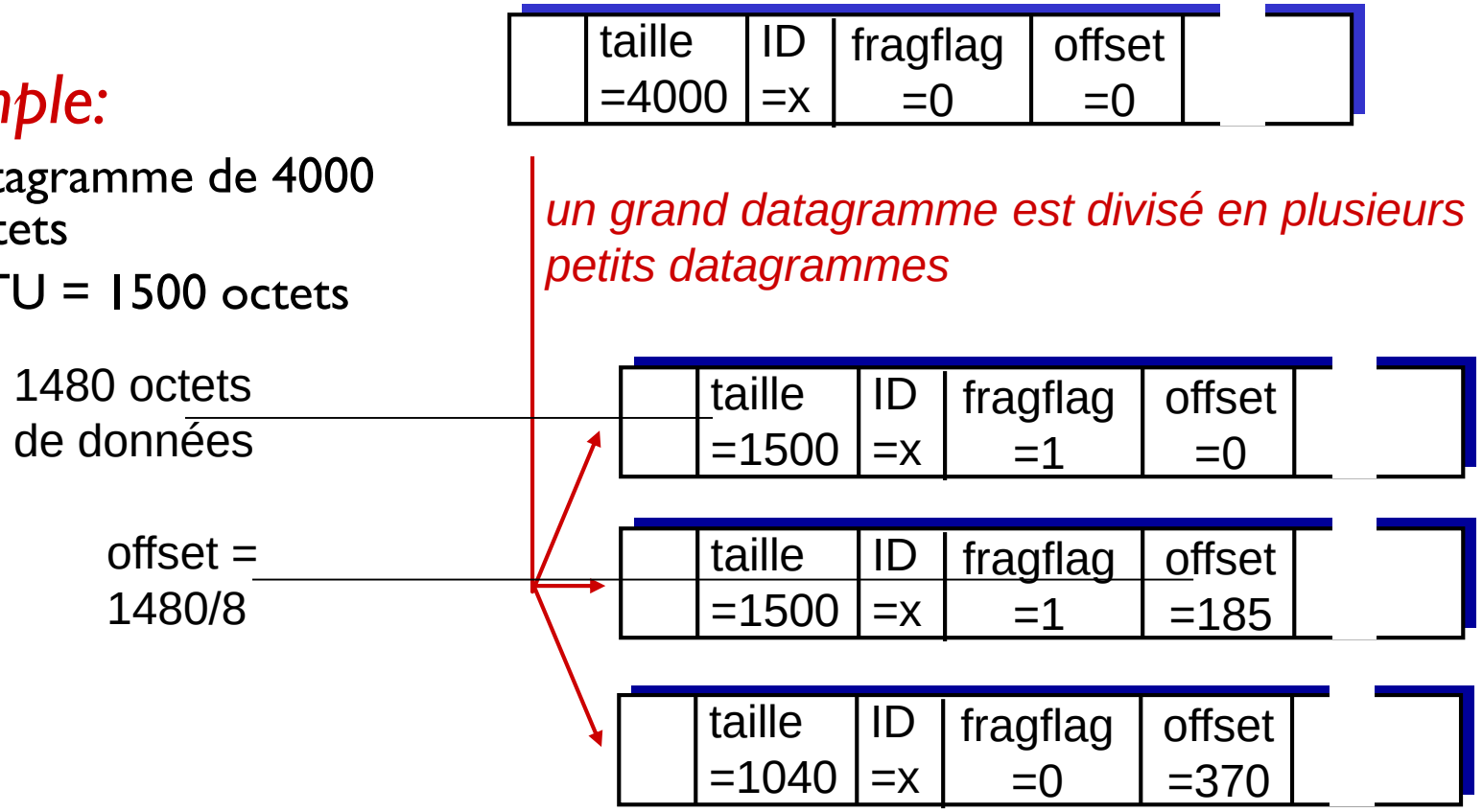
- Un lien réseau a un MTU (*max. transfer unit size*) – taille maximale du datagramme.
  - différents types de liens, différents MTUs
- Un datagramme IP large est divisé (“fragmenté”) dans le réseau
  - le datagramme devient plusieurs
  - “réassemblage” à la dest. finale seulement
  - les bits d’entête IP sont utilisés pour identifier les fragments



Fragmentation IP, réassemblage

exemple:

- datagramme de 4000 octets
- MTU = 1500 octets



Chapitre IV: plan

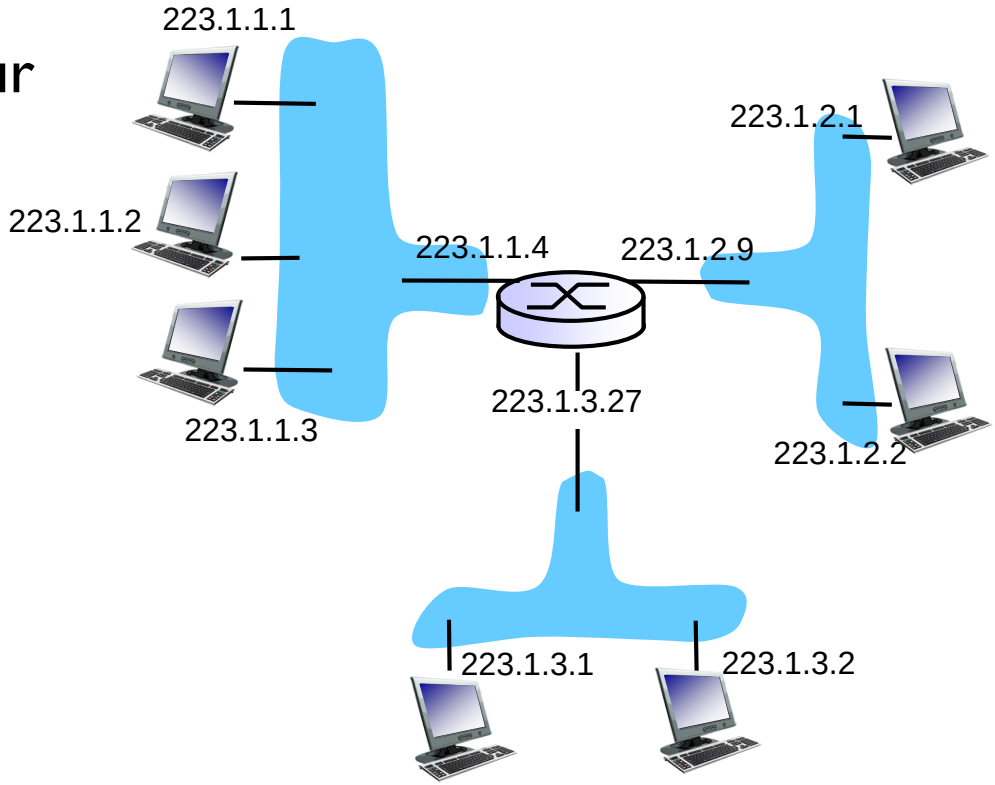
4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

Adressage IP: introduction

- *adresse IP:* identificateur d'*interface* de 32 bits
- *interface:* connexion entre l'équipement et le lien physique
  - un routeur en possède plusieurs
  - un hôte en possède très souvent un à deux
- *une adresse IP associée à chaque interface*

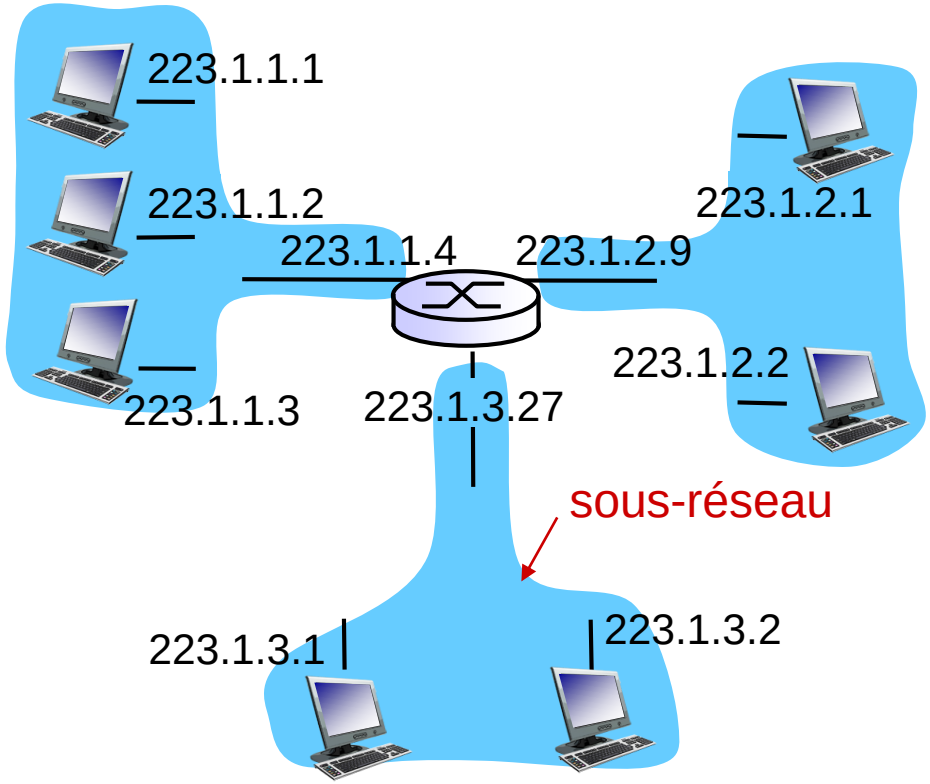


223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

Sous-réseaux

- adresse IP:
  - les bits les plus significatifs – sous réseau
  - les bits les moins significatifs – hôtes
- *un sous-réseau ?*
  - les interfaces qui ont la même partie sous-réseau dans leurs adresses IP
  - peuvent physiquement communiquer sans l'intervention d'un routeur

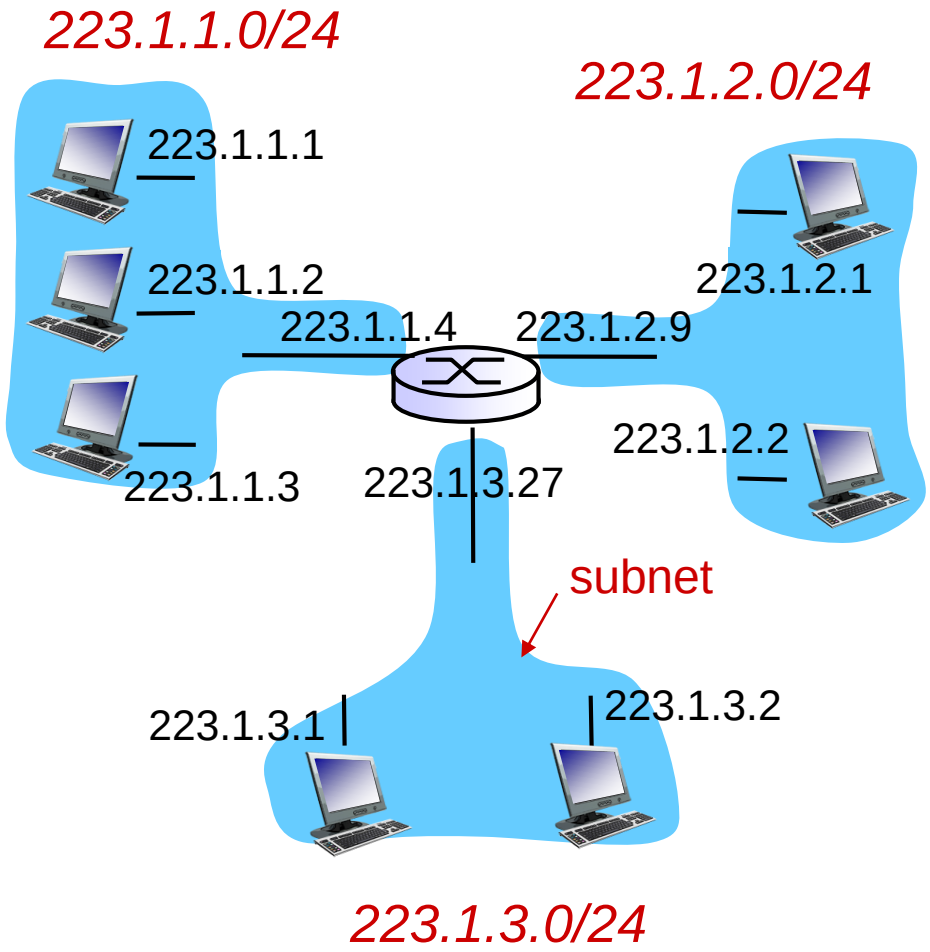


un réseau de 3 sous-réseaux

Sous-réseaux

méthode

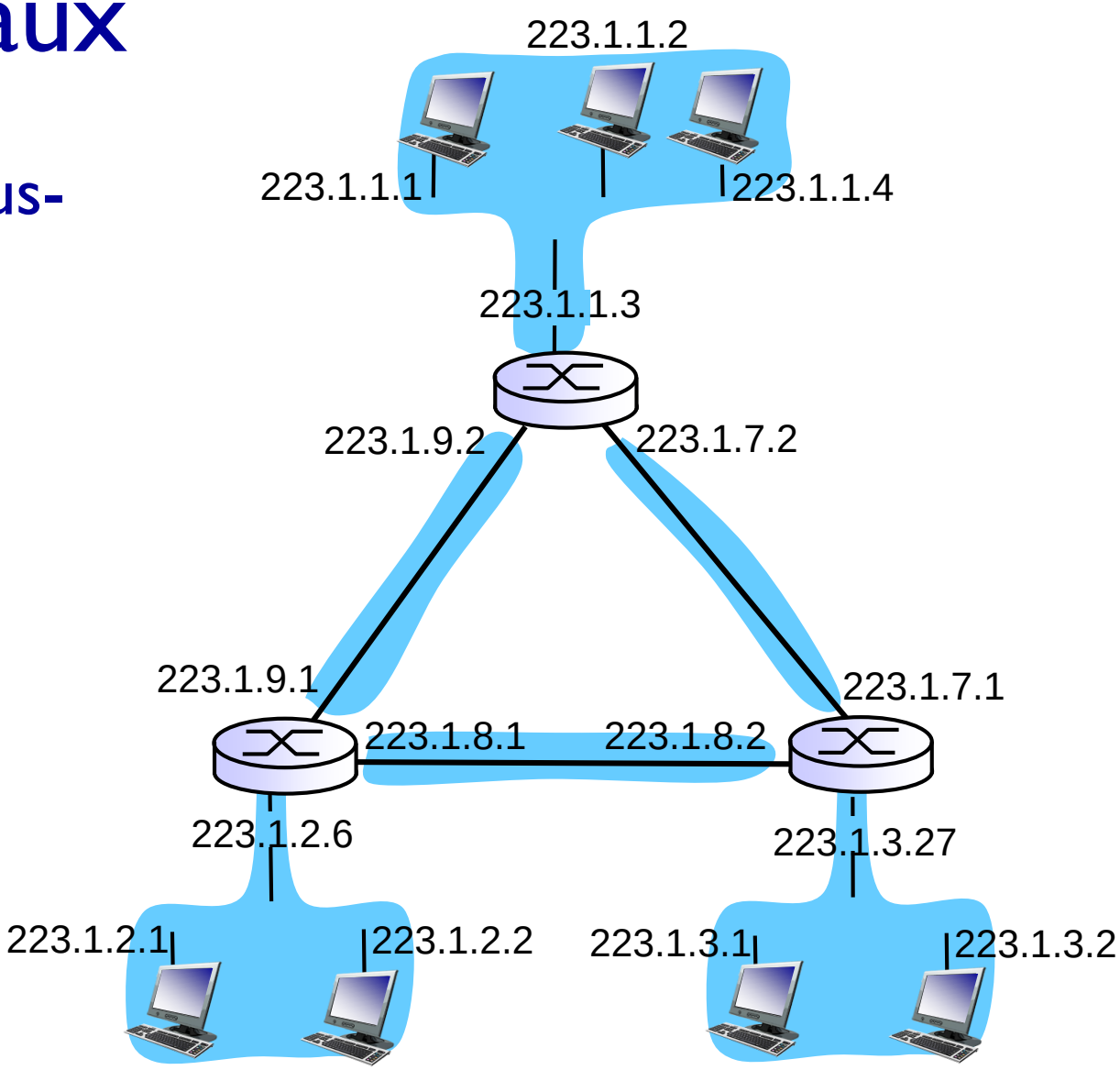
- pour déterminer les sous-réseaux, détacher chaque interface de sa machine ou routeur, en créant des iles de réseaux isolés
- chaque réseau isolé est un sous-réseau



le masque: /24

Sous-réseaux

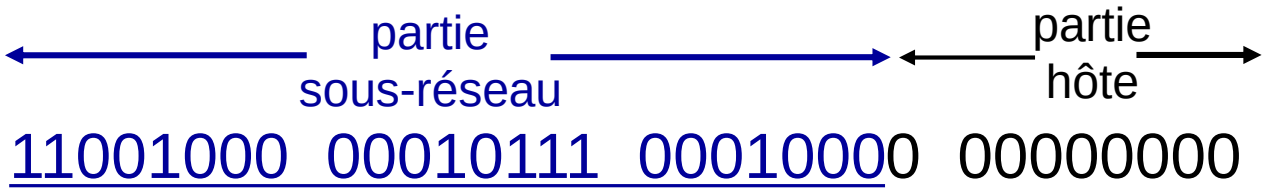
Combien de sous-réseaux?



Adressage IP: CIDR

CIDR: Classless InterDomain Routing

- La partie d'adressage sous réseau est de taille arbitraire
- Le format de l'adresse: a.b.c.d/x, où x est le # de bits dans la partie sous réseau de l'adresse



200.23.16.0/23

Adresses IP: comment en obtenir une?

Q: comment obtenir une adresse IP?

- stocker dans un fichier
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol: dynamiquement à partir d'un serveur
  - "plug-and-play"

adresses IP : comment en obtenir un?

Q: comment un réseau donne des adresses IP à ses sous-réseaux?

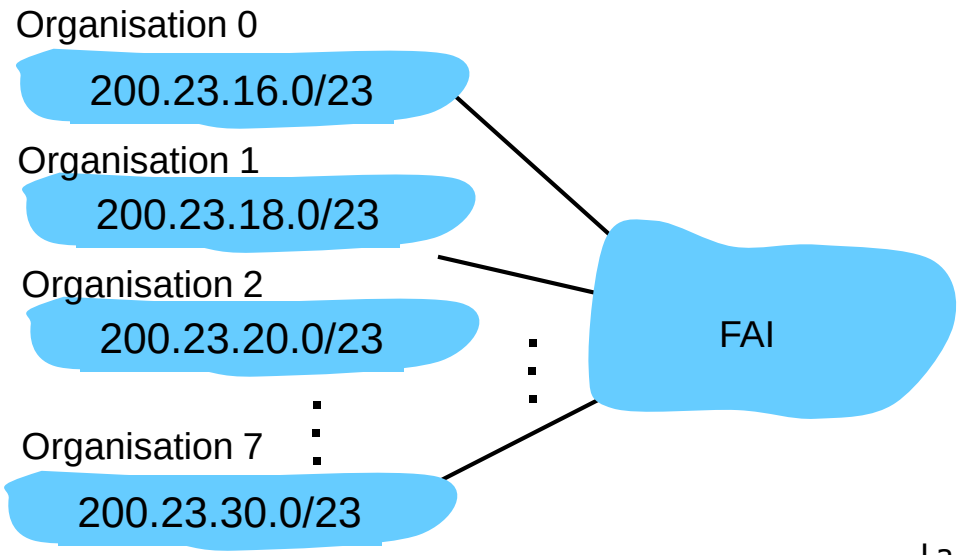
A: allouer des portions de son espace des adresses alloué par le fournisseur d'accès

Le bloc de ISP 11001000 00010111 00010000 00000000 200.23.16.0/20

Organisation 0	11001000 00010111 0001000**0**	00000000	200.23.16.0/23
Organisation 1	11001000 00010111 0001001**0**	00000000	200.23.18.0/23
Organisation 2	11001000 00010111 0001010**0**	00000000	200.23.20.0/23
...	.....	....	....
Organisation 7	11001000 00010111 0001111**0**	00000000	200.23.30.0/23

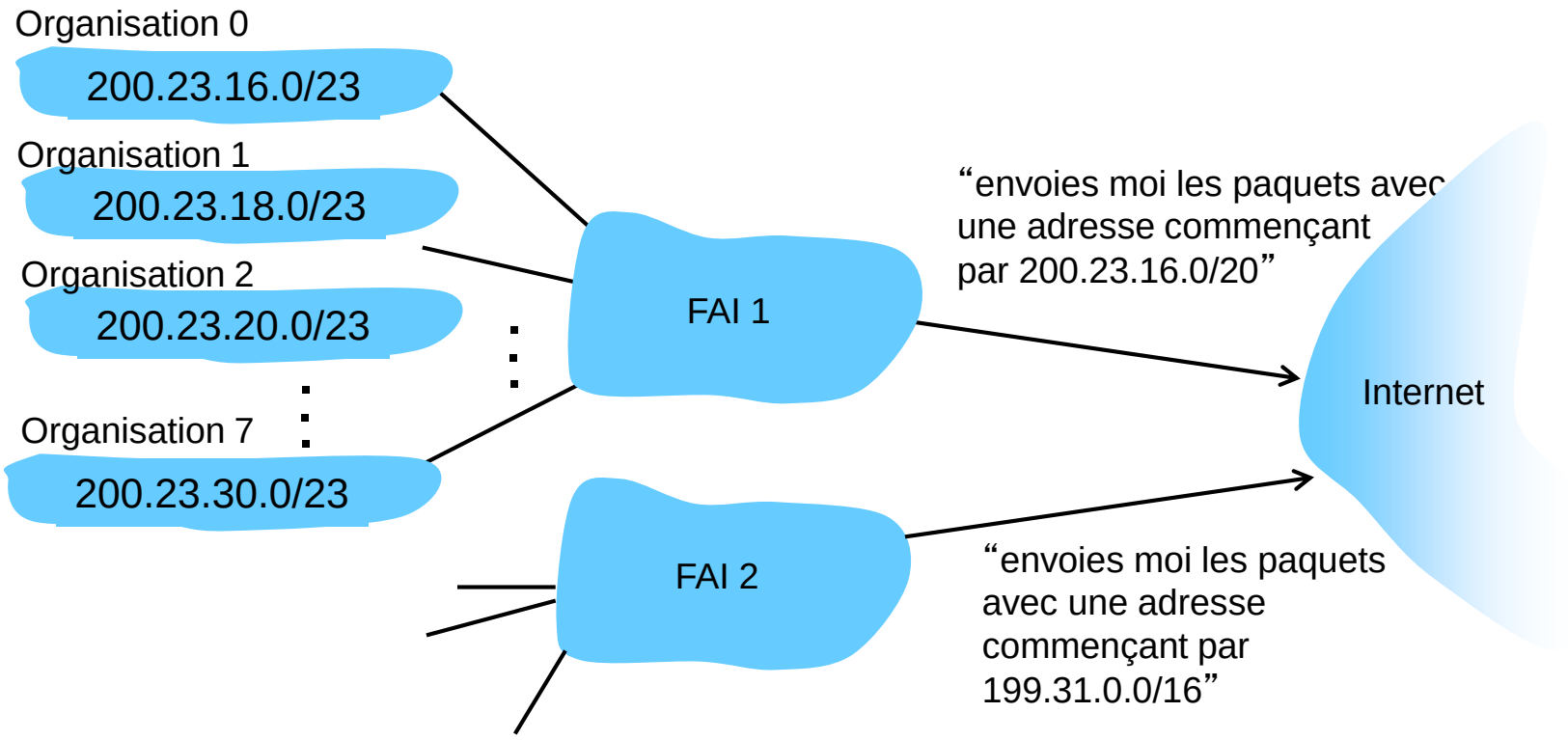
Adressage hiérarchique

Intervalle FAI	11001000 00010111 00010000	00000000	200.23.16.0/20
Organization 0	11001000 00010111 00010000	00000000	200.23.16.0/23
Organization 1	11001000 00010111 00010010	00000000	200.23.18.0/23
Organization 2	11001000 00010111 00010100	00000000	200.23.20.0/23
...	.....	....	....
Organization 7	11001000 00010111 00011110	00000000	200.23.30.0/23



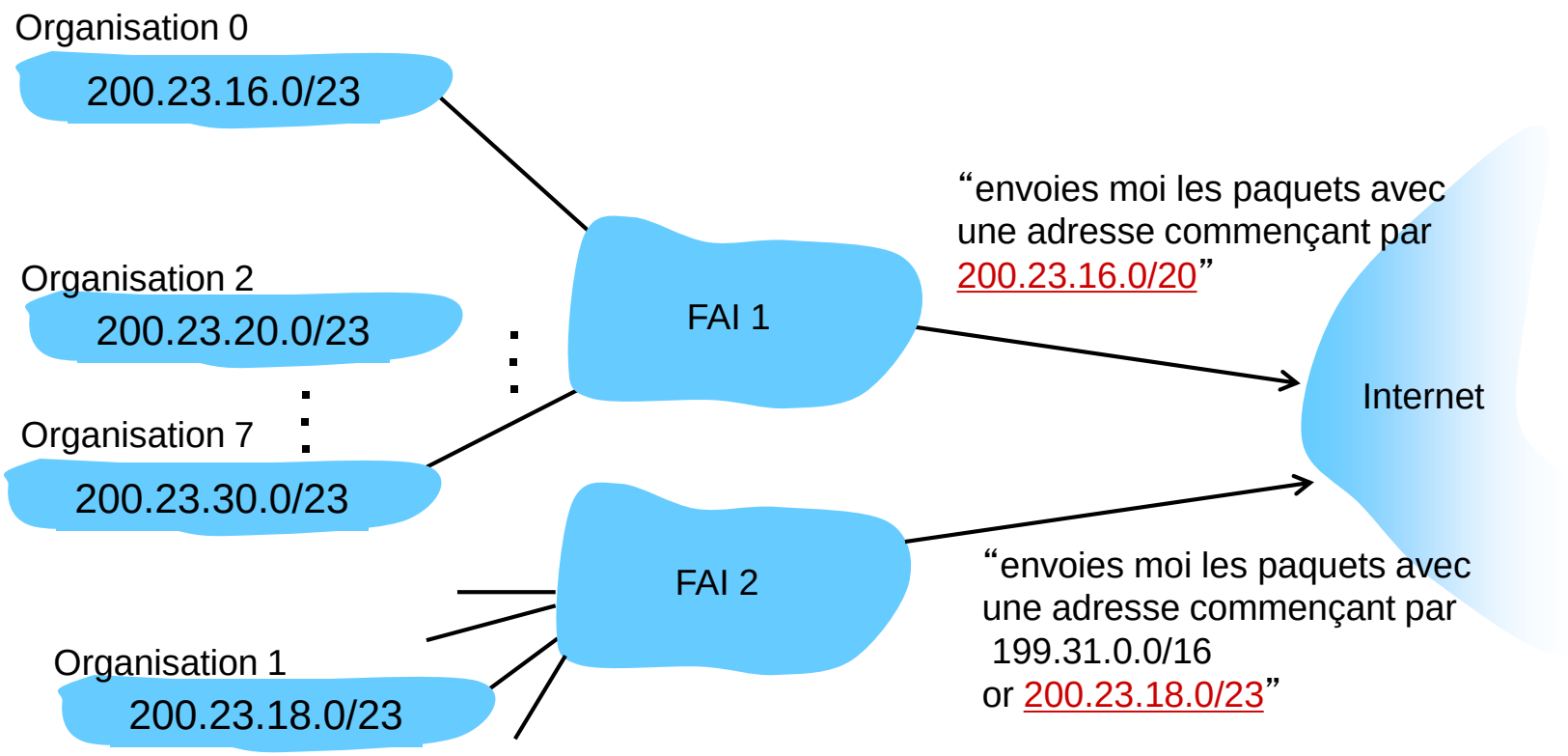
Adressage hiérarchique

L'adressage hiérarchique permet une diffusion efficace des info de routage:



Adressage hiérarchique

FAI2 a une route spécifique vers Organisation 1



Chapitre IV: plan

4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

DHCP: Dynamic Host Configuration Protocol

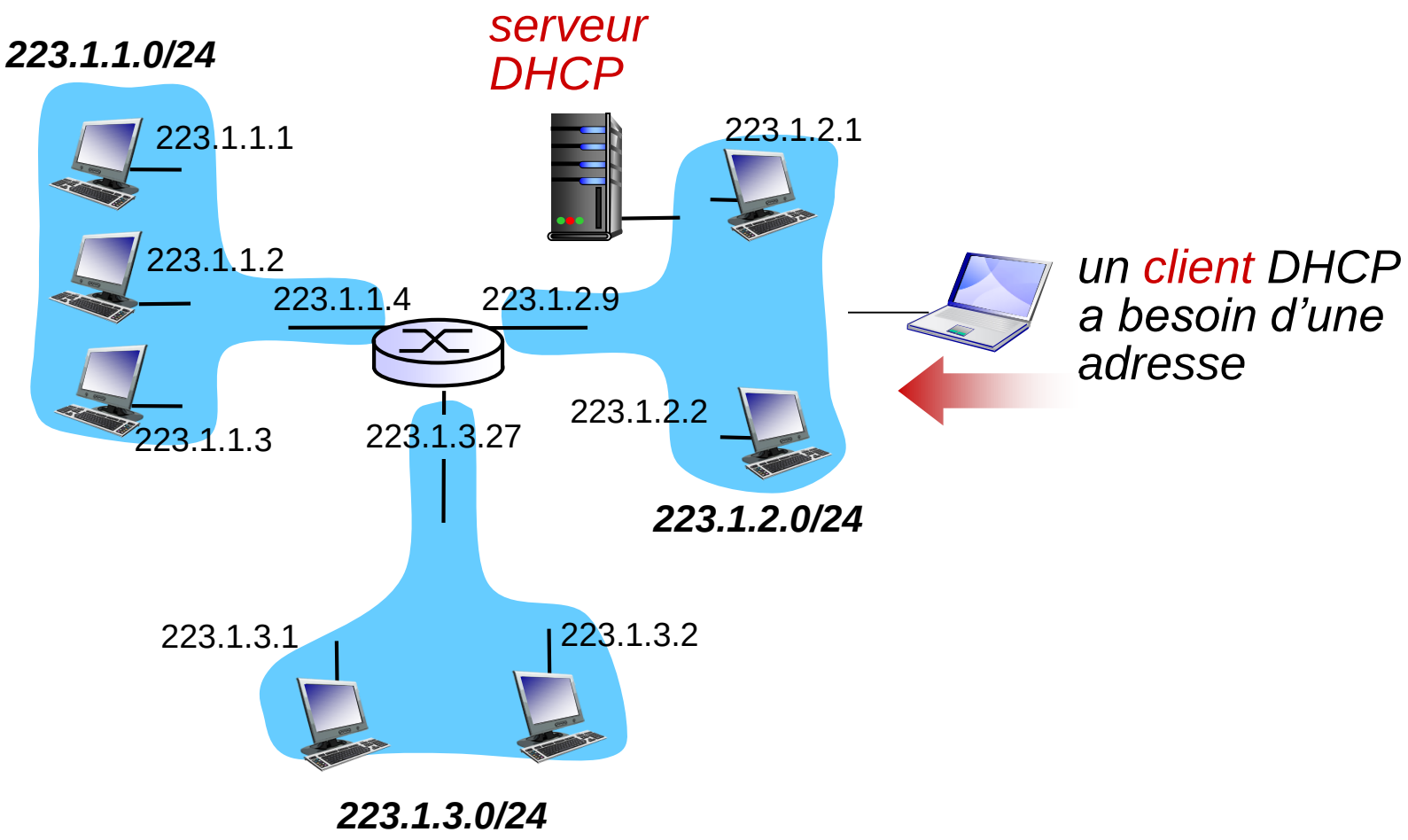
objectif : permettre à l'hôte d'obtenir une adresse IP d'un serveur quand il rejoint le réseau

- peut renouveler son usage d'adresse
- permet la réutilisation d'adresses (seulement monopoliser l'adresse quand il est connecté)
- supporte des usagers mobiles qui veulent rejoindre des réseaux (brièvement)

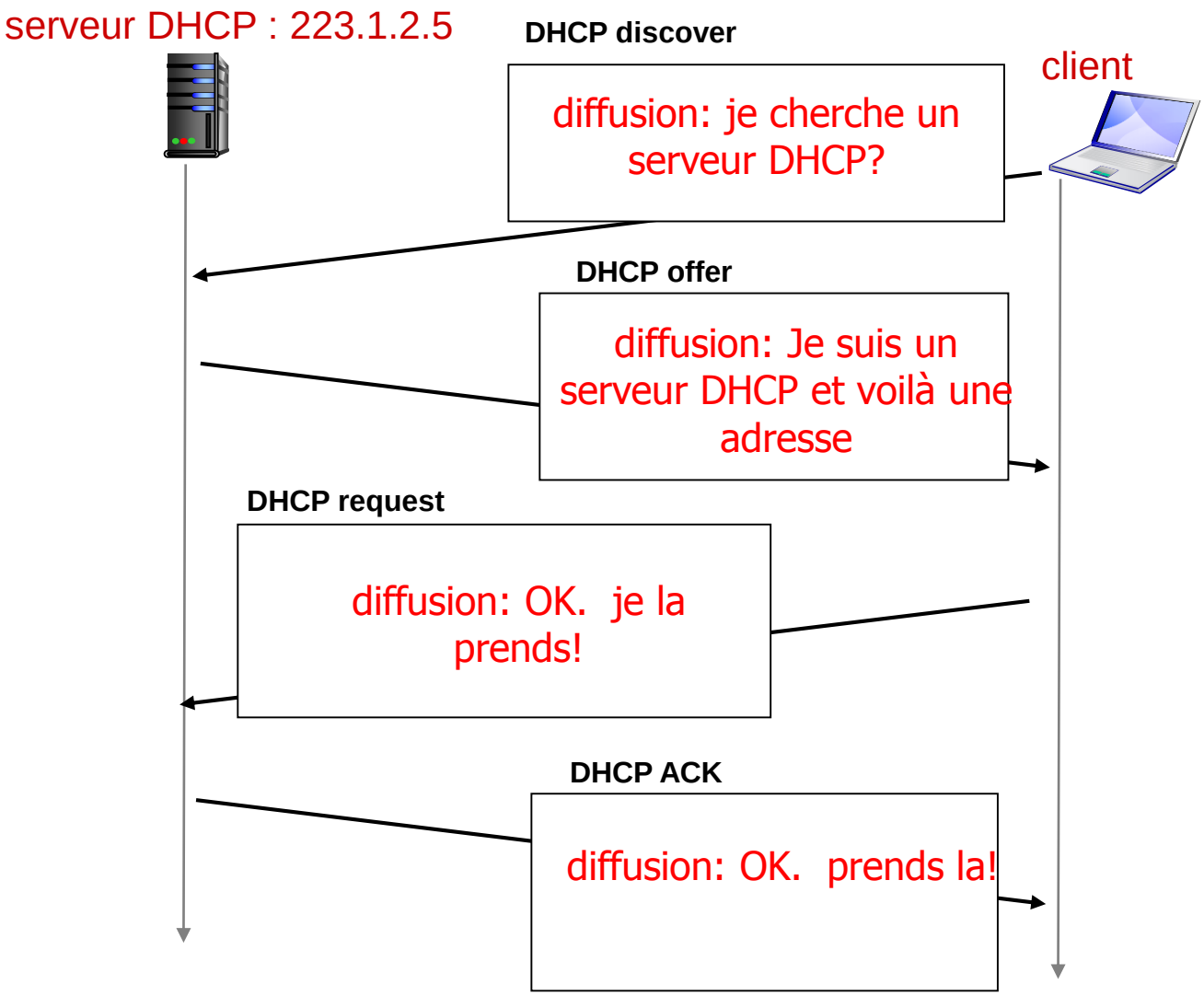
DHCP :

- l'hôte diffuse "DHCP discover"
- le serveur DHCP réponds par "DHCP offer"
- l'hôte demande l'adresse IP : "DHCP request"
- le serveur DHCP envoie l'adresse: "DHCP ack"

Scénario DHCP



Scénario DHCP

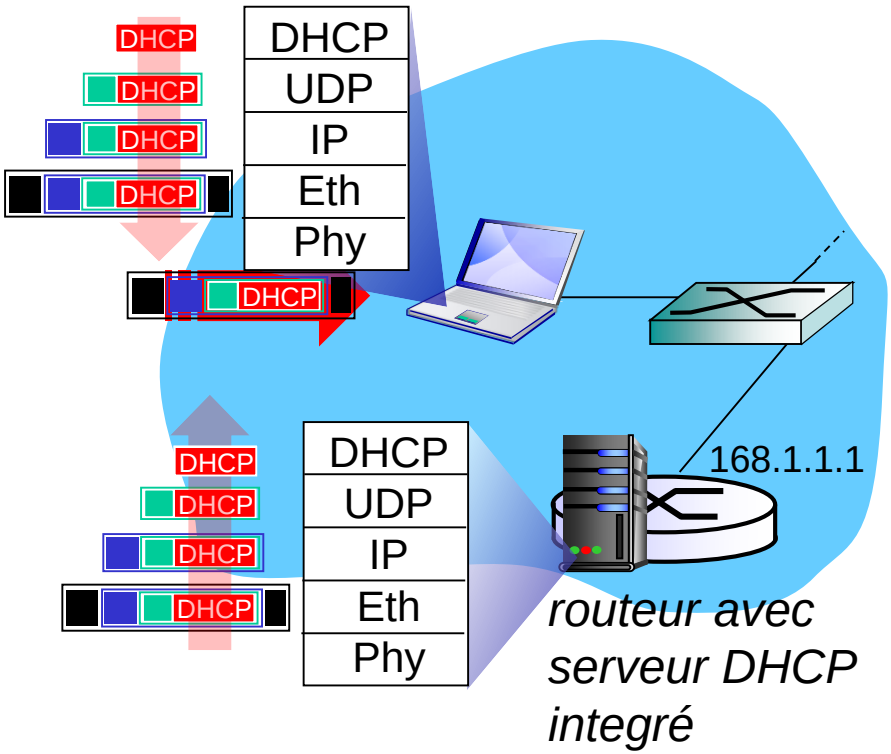


DHCP

DHCP peut retourner plus que seulement des adresses IP allouées aux sous-réseaux :

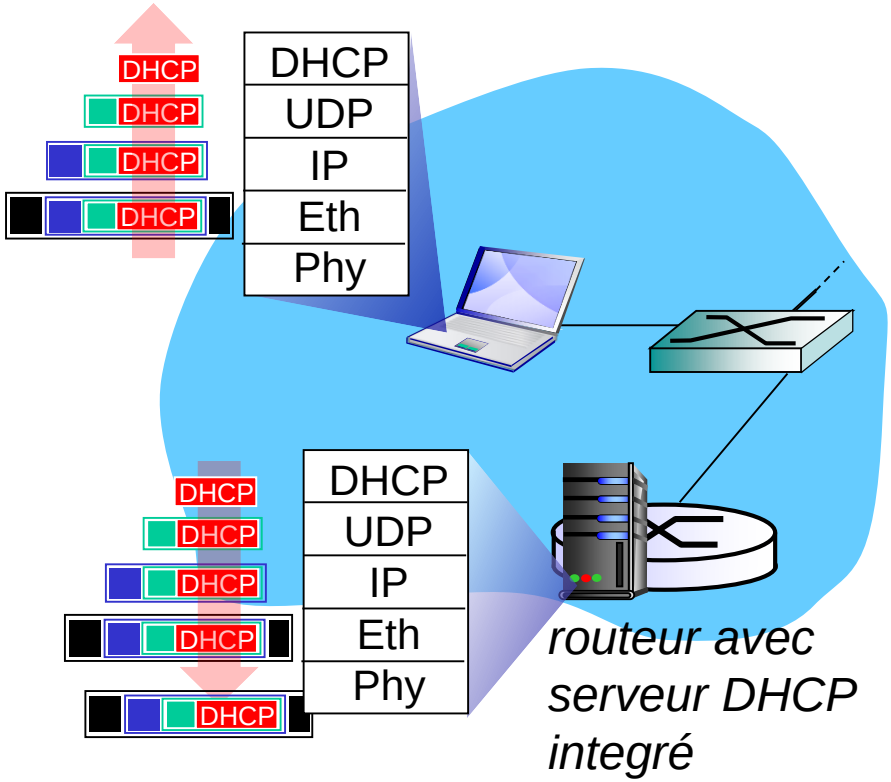
- Adresse du premier routeur pour un client
- Nom et adresse IP d'un serveur DNS
- Le masque réseau (indiquant la partie adresse réseau vs hôte)

DHCP: exemple



- connecter un ordi qui a besoin d'une @IP, @ du premier routeur, @ du serveur DNS : utilise DHCP
- la requête DHCP est encapsulé en UDP, encapsulé en IP, encapsulé en Ethernet 802.1
- la trame Ethernet est diffusé (@dest. FFFFFFFFFFFF) sur le LAN, elle est reçue par le routeur qui héberge le serveur DHCP
- la trame Ethernet décapsulé à IP, décapsulé à UDP, décapsulé à DHCP

DHCP: exemple



- le serveur DHCP envoie un Ack DHCP qui contient une adresse IP, l'adresse IP du premier routeur, nom & adresse IP du serveur DNS
- encapsulation du serveur DHCP, transfert de la trame vers le client,
- décapsuler jusqu'au DHCP chez le client
- le client connait maintenant son adresse IP, le nom et l'adresse IP du serveur DNS, l'adresse IP du premier routeur

DHCP: Wireshark (LAN maison)

requête

Message type: **Boot Request (1)**
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) **DHCP Message Type = DHCP Request**
Option: (61) Client identifier
    Length: 7; Value: 010016D323688A;
    Hardware type: Ethernet
    Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Option: (t=50,l=4) Requested IP Address = 192.168.1.101
Option: (t=12,l=5) Host Name = "nomad"
Option: (55) Parameter Request List
    Length: 11; Value: 010F03062C2E2F1F21F92B
    1 = Subnet Mask; 15 = Domain Name
    3 = Router; 6 = Domain Name Server
    44 = NetBIOS over TCP/IP Name Server
    ……

réponse

Message type: **Boot Reply (2)**
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 192.168.1.101 (192.168.1.101)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 192.168.1.1 (192.168.1.1)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP ACK
Option: (t=54,l=4) Server Identifier = 192.168.1.1
Option: (t=1,l=4) Subnet Mask = 255.255.255.0
Option: (t=3,l=4) Router = 192.168.1.1
Option: (6) Domain Name Server
    Length: 12; Value: 445747E2445749F244574092;
    IP Address: 68.87.71.226;
    IP Address: 68.87.73.242;
    IP Address: 68.87.64.146
Option: (t=15,l=20) Domain Name = "hsd1.ma.comcast.net."

ICMP: internet control message protocol

- utilisé par les nœuds pour communiquer des info de la couche 3
  - rapports d'erreurs: hôte non accessible, réseau, port, protocole
  - écho requête/réponse (utilisé par ping)
- au niveau réseau mais il est "au dessus" de IP:
  - les msgs ICMP sont portés dans un datagramme IP
- message ICMP : type, code plus les premiers 8 octets du datagramme IP causant l'erreur

Type	Code	description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

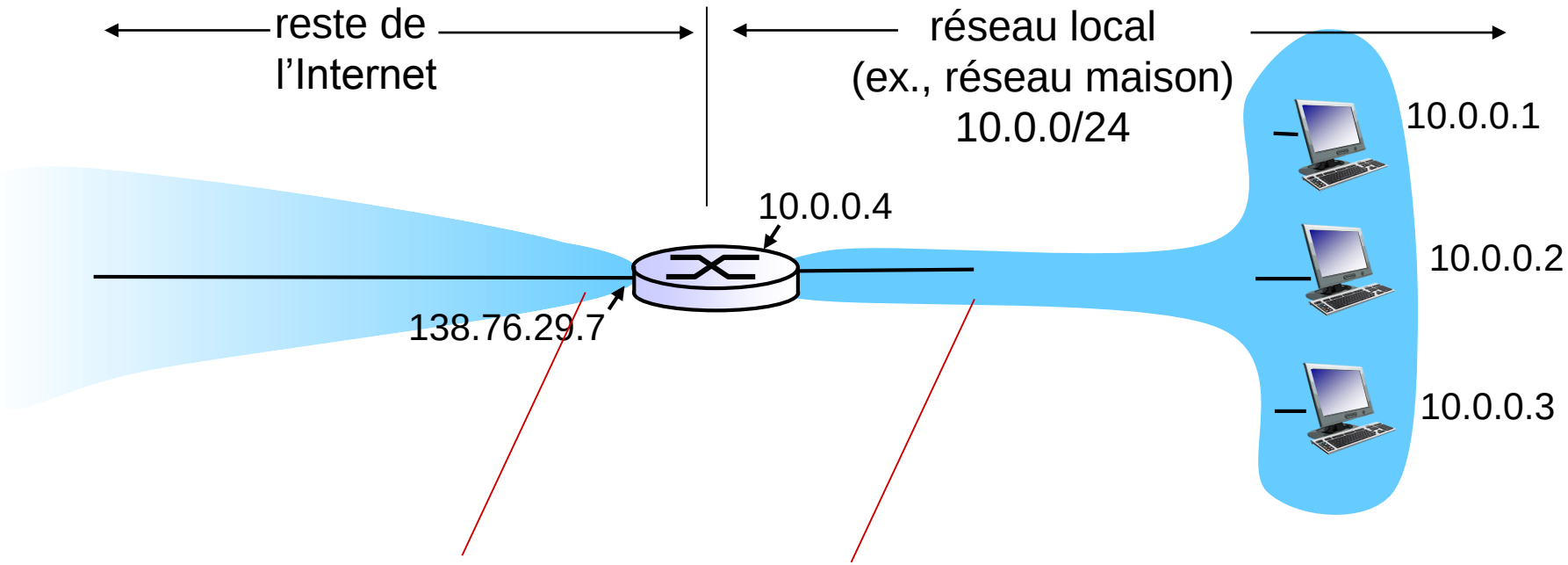
Chapitre IV: plan

4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

NAT: translation d'adresse réseau



Tous les datagrammes quittant le réseau local ont la même adresse IP source NAT: 138.76.29.7, numéros de port source différents

Les datagrammes avec source ou destination dans ce réseau Ont l'adresse 10.0.0/24 pour la source, destination (comme d'hab.)

NAT: translation d'adresse réseau

motivation: le réseau local utilise seulement une adresse IP pour l'extérieur:

- la plage des adresses non nécessaires pour le FAI: juste une adresse IP pour tous les équipements
- peut changer les adresses des équipements dans un réseau local sans en informer l'extérieur
- peut changer le FAI sans changer les adresses dans le réseau local
- les équipements dans le réseau local ne sont pas explicitement adressables, ni visible par l'extérieure (meilleure sécurité)

NAT: translation d'adresse réseau

implémentation: le routeur NAT doit:

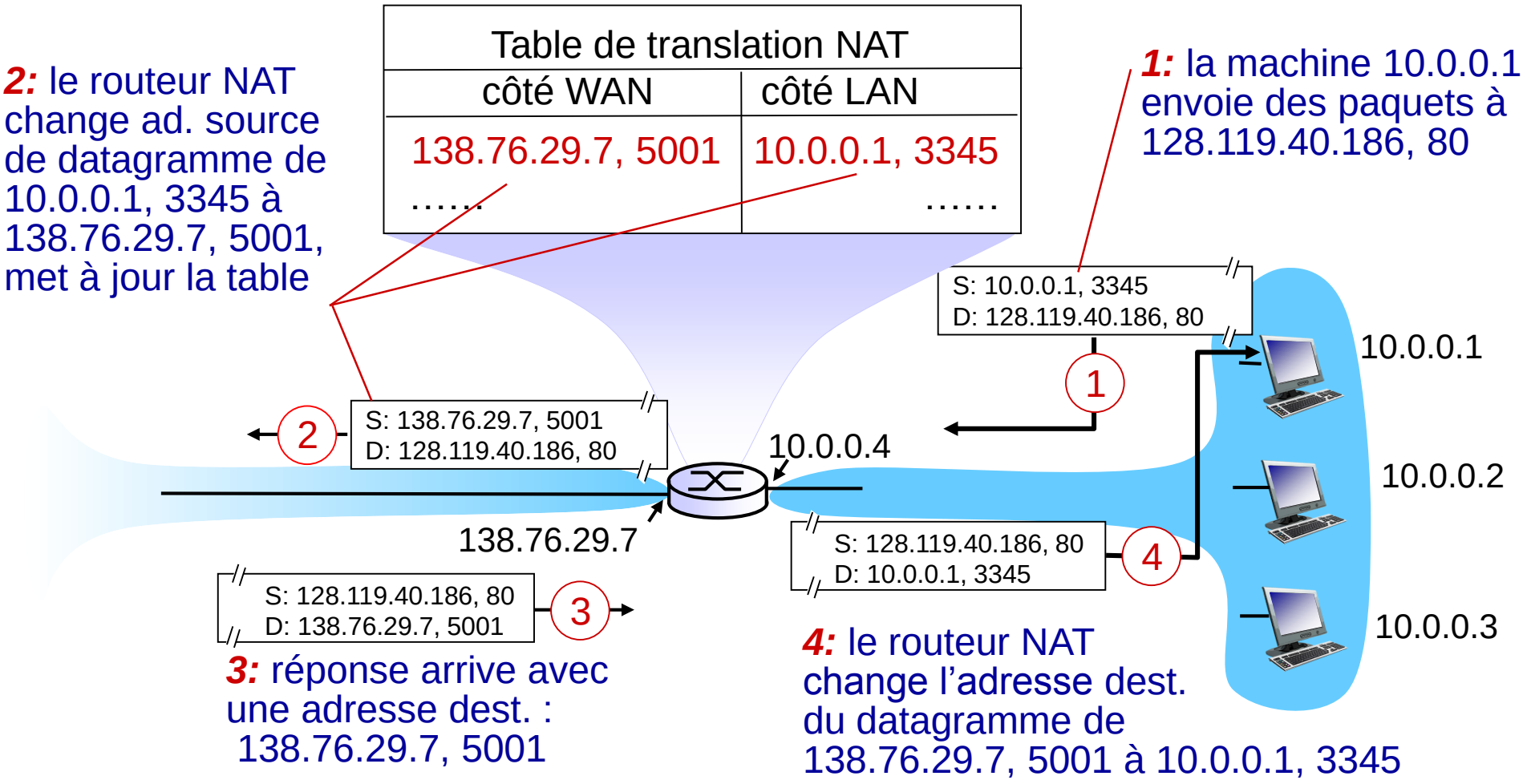
- *Datagrammes sortants :* remplacer (adresse IP source, # port) de chaque datagramme à (adresse IP NAT, nouveau # port)

  . . . les clients/servers distants répondent en utilisant (adresse IP NAT, nouveau # port) comme adresse de destination

- *Se rappeler (en utilisant la table de translation NAT)* chaque paire de translation (adresse IP source, # port) à (adresse IP NAT, nouveau # port)

- *Datagrammes entrants: remplacer* (adresse IP NAT, nouveau # port) dans le champ dest. de chaque datagramme entrant avec la valeur (adresse IP source, # port) correspondante stockée dans le tableau NAT

NAT: translation d'adresse réseau



NAT: translation d'adresse réseau

- 16-bit champ du numéro de port :
  - 60,000 connexions simultanées avec une seule adresse LAN!
- la NAT est controversée :
  - les routeurs doivent traiter seulement jusqu'à la couche 3
  - le manque d'adresses doit être résolue par IPv6
  - la NAT ne respecte pas le principe de bout-en-bout
    - La possibilité NAT doit être prise en compte par les concepteurs d'app., ex., applications P2P
  - NAT traversable: et si le client veut se connecter avec un serveur derrière un NAT?

Chapitre IV: plan

4.1 introduction
- plan de données
- plan de contrôle

4.2 à l'intérieur d'un routeur

4.3 IP: Internet Protocol
- structure du datagramme
- adressage IPv4
- DHCP
- ICMP
- translation d'adresses réseaux (NAT)
- IPv6

IPv6: motivation

- *motivation initiale:* l'espace d'adresses de 32-bit sera complétement allouée bientôt.
- motivation additionnelles :
  - Le format de l'entête devra aider à accélérer le traitement/transfert
  - L'entête devra faciliter la QoS

format de datagramme IPv6 :

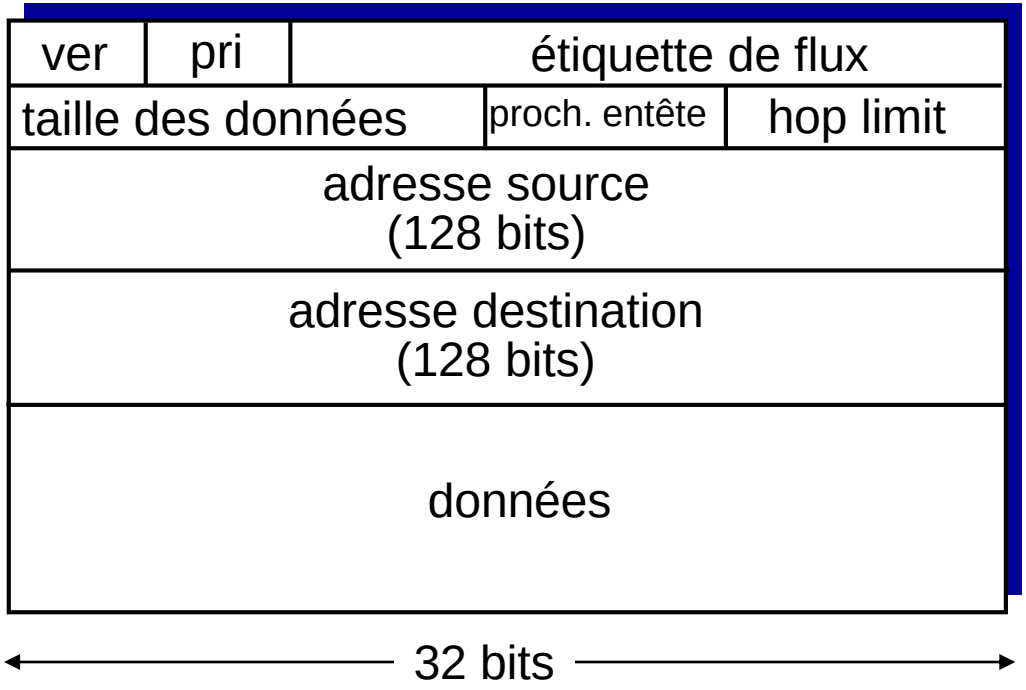
- Entête de taille fixe de 40 octets
- Pas de fragmentation possible

Format de datagramme IPv6

priorité: identifier la priorité des datagrammes d'un flux

étiquette de flux: identifier les datagrammes du même "flux" (le concept de "flux" n'est pas bien défini).

Prochain entête: identifier le protocole de la couche supérieure pour les données

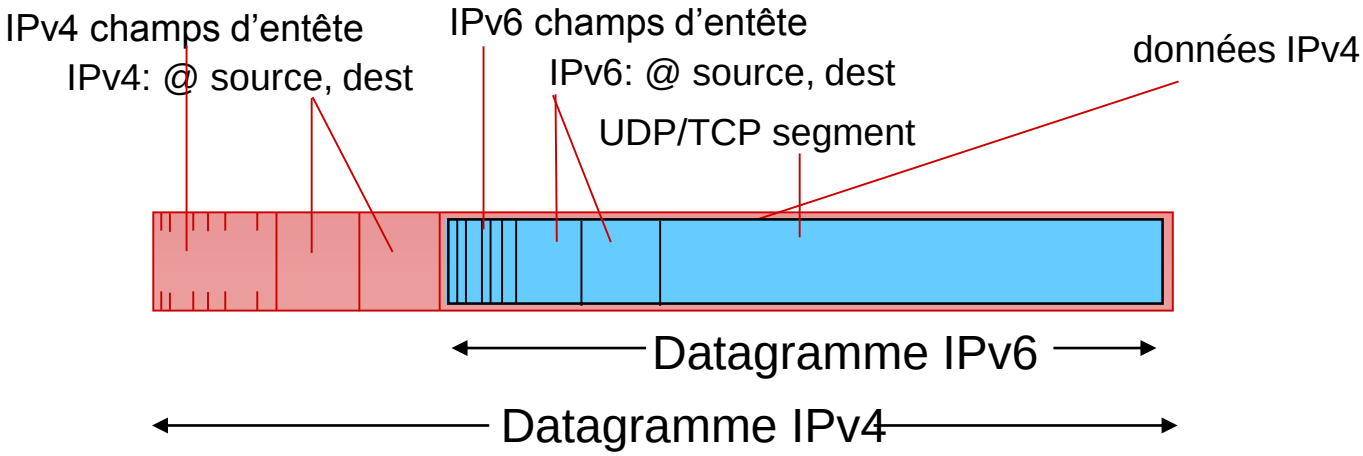


D'autres changements vs IPv4

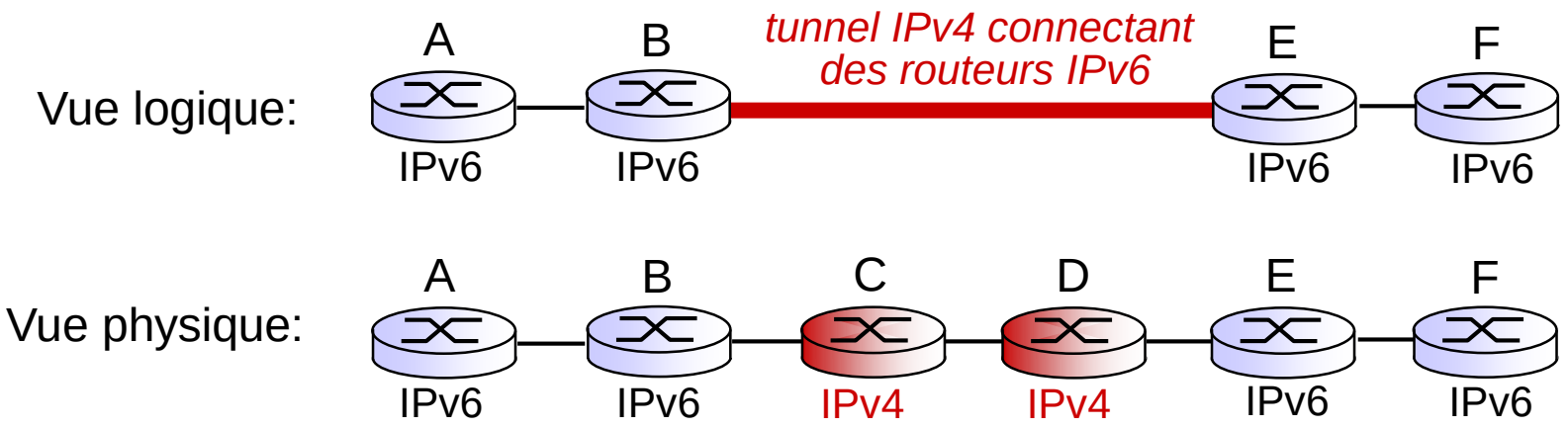
- *checksum:* enlevé pour réduire le temps fde traitement à chaque saut
- *options:* permises, mais à l'extérieure de l'entête, indiquées par le champ de "prochain entête"
- *ICMPv6:* nouvelle version de ICMP
  - types de message additionnels, ex. "Paquet trop grand"
  - Fonctions de gestion de groupe de multicast

Transition de IPv4 à IPv6

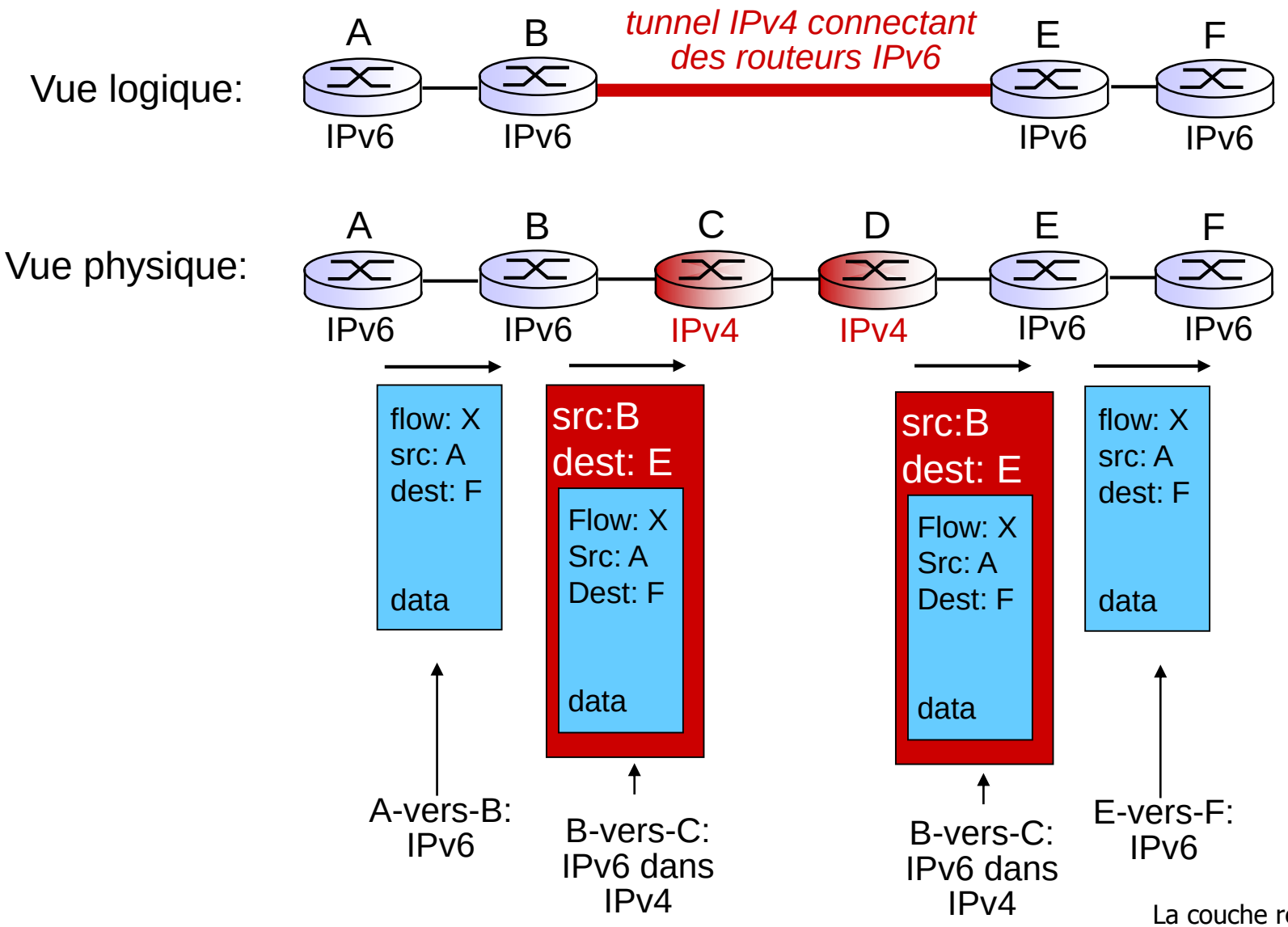
- Les routeurs ne passeront pas tous à IPv6 simultanément
  - Comment le réseau va opérer avec un mélange de routeurs IPv4 et IPv6?
- *Mise en tunnel:* les datagrammes IPv6 joueront le rôle de données dans des datagrammes IPv4 pour traverser les routeurs IPv4



Mise en tunnel



Mise en tunnel



IPv6: adoption

- Google: 8% of des services accès des via IPv6
- NIST: 1/3 des domaines du gouvernement US sont capables de fonctionner avec IPv6

- *Très longue période de déploiement, utilise*
  - 20 ans et encore!
  - les changements d'applications en 20 ans a changé énormément : WWW, Facebook, diffusion média, Skype, …
  - *pourquoi?*

Exercice

- Soit un datagramme de 2400 octets avec un id. 422 sur un lien avec MTU = 700 octets
  - combien de fragments?
  - valeurs des différents champs?

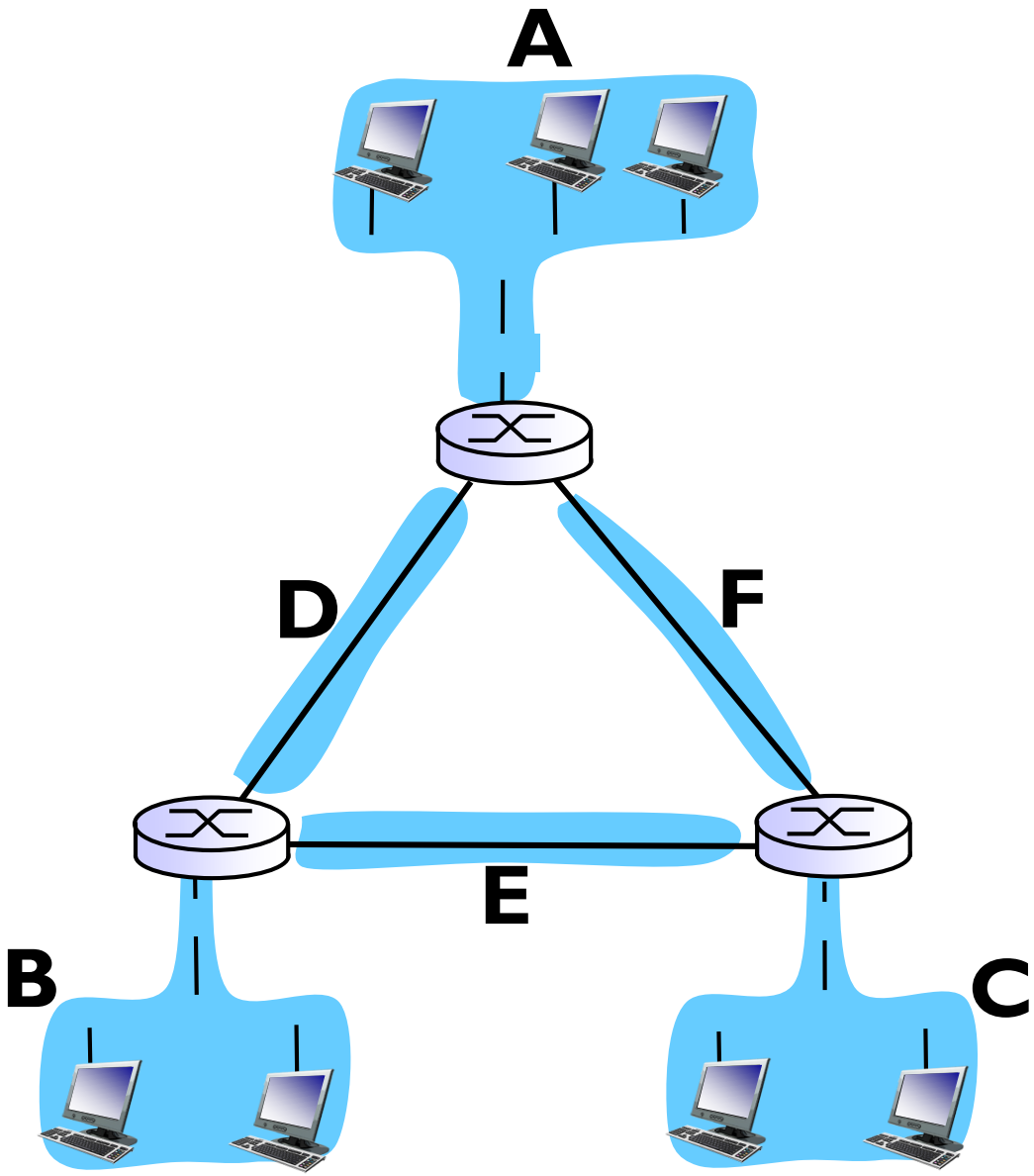
Exercice

- Soit le sous-réseau ayant l'adresse 128.119.40.128/26
  - combien peut-on assigner d'adresses IP?
  - donner la plage d'adresses.

- Soit le sous-réseau ayant l'adresse 128.119.40.64/26
  - diviser le sous-réseau en quatre sous-réseaux ayant chacun le même nombre d'adresses

Exercice

- Soit le sous-réseau ayant l'adresse 192.1.1.0/24 qui doit être diviser en
  - A : 60 interfaces
  - B : 20 interfaces
  - C : 14 interfaces
  - D : ? interfaces
  - E : ? interfaces
  - F : ? interfaces



Exercice

- Soit le sous-réseau ayant l'adresse 214.97.254.0/23 qui doit être diviser en
  - A : 250 interfaces
  - B : 120 interfaces
  - C : 120 interfaces
  - D : 2 interfaces
  - E : 2 interfaces
  - F : 2 interfaces