



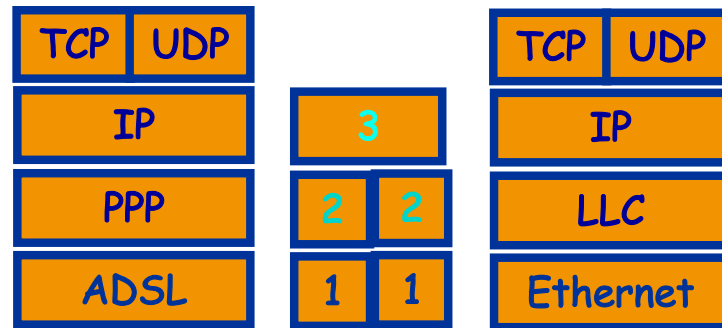
# Chapitre AR1

## Architecture

Architecture matérielle (Internet)  
Architecture logique & modèle  
OSI



# AR – Architecture



- Comment on découpe et assemble les « briques » pour combiner les fonctions nécessaires à la communication
  - Architecture en couches
- Vous comprendrez:
  - Pourquoi il y a tant de normes de réseaux et non une seule
  - Comment choisir et argumenter entre plusieurs solutions pour relier des sites d'une entreprise



# Objectifs du chapitre AR1

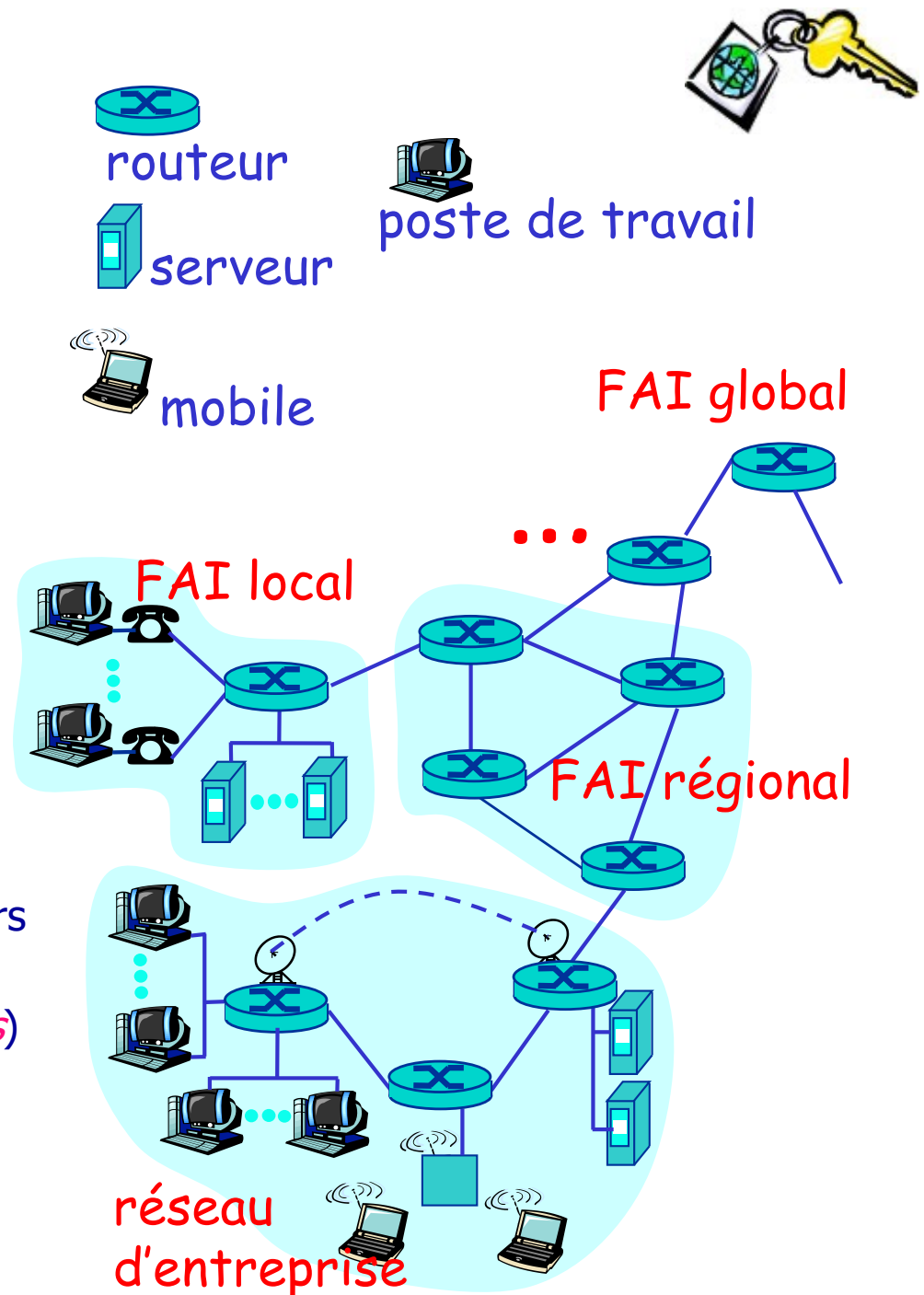
- Comprendre la notion d'architecture en couches
  - Structuration logicielle
  - Avoir une idée de ce que fait chaque couche
  - Être capable d'argumenter, proposer des alternatives de structuration
- Vocabulaire internet
  - FAI, hôte
  - paquet, trame, segment
  - Hiérarchie des réseaux

# Plan du chapitre AR1

- Architecture matérielle (Internet)
  - Vocabulaire
  - Structure hiérarchique de l'Internet
  - Modes paquet & circuit
- Architecture logique en couches
  - Motivation de l'architecture en couches
  - Modèles OSI & TCP/IP
  - Exemples de piles & de protocoles

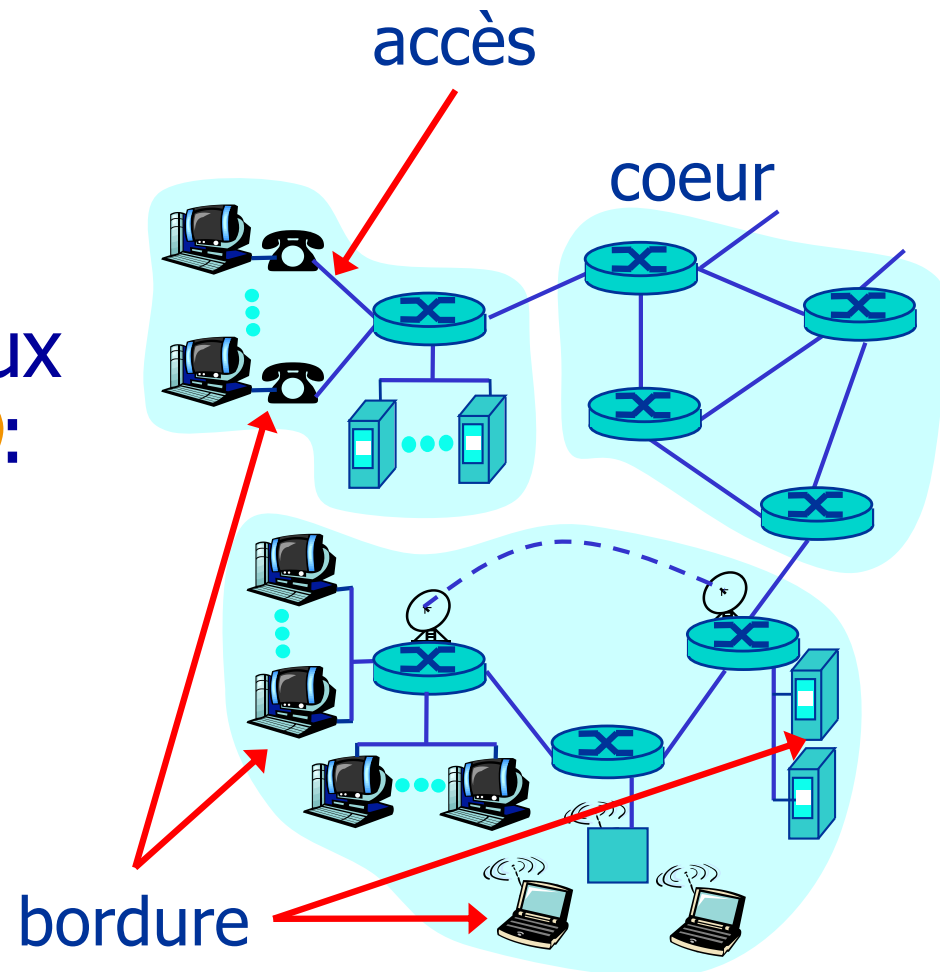
# Internet

- millions de systèmes connectés: hôtes (*hosts, end-systems*)
  - stations PC, serveurs, PDA, grille-pains...  
exécutent applications réseau
- liens de communication
  - fibre, cuivre, radio, satellite...
- routeurs: acheminent des paquets
- protocoles: contrôlent la communication entre machines
  - TCP, IP, HTTP, FTP...
- Internet: “réseau des réseaux”
  - structure hiérarchique de FAI (Fournisseurs d'Accès Internet)
  - interconnexion globale (mondiale: *carriers*)
  - **Internet** public vs. **intranet** privé
- standards d'Internet
  - RFC: Request for comments, émis par:
  - IETF: Internet Engineering Task Force



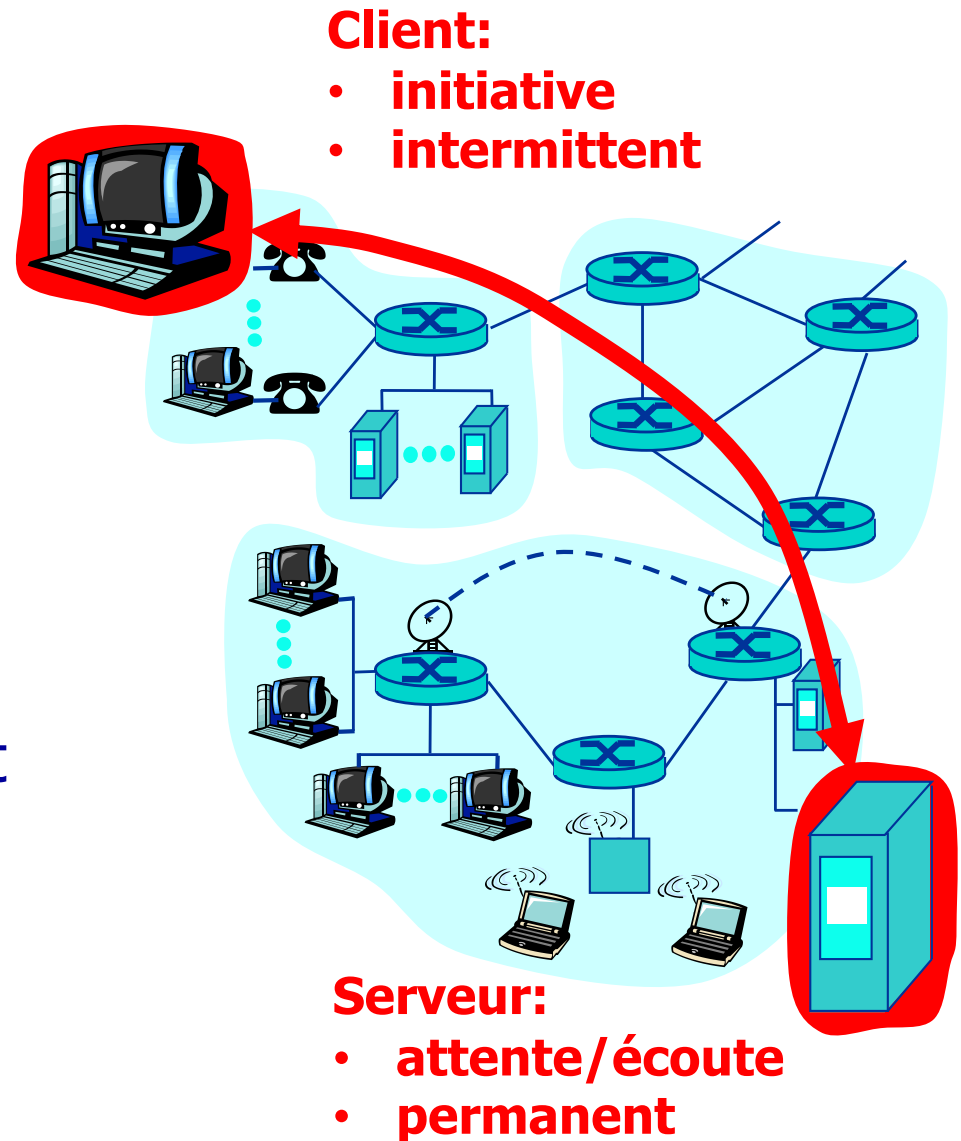
# Structure de réseau (Internet)

- bordure (*chap. AP, PR*):
  - applications & hôtes
- réseaux d'accès, canaux physique (*chap. RA, CN*):
  - liens de communication
- cœur (*cours Rx*):
  - routeurs
  - "réseau des réseaux"



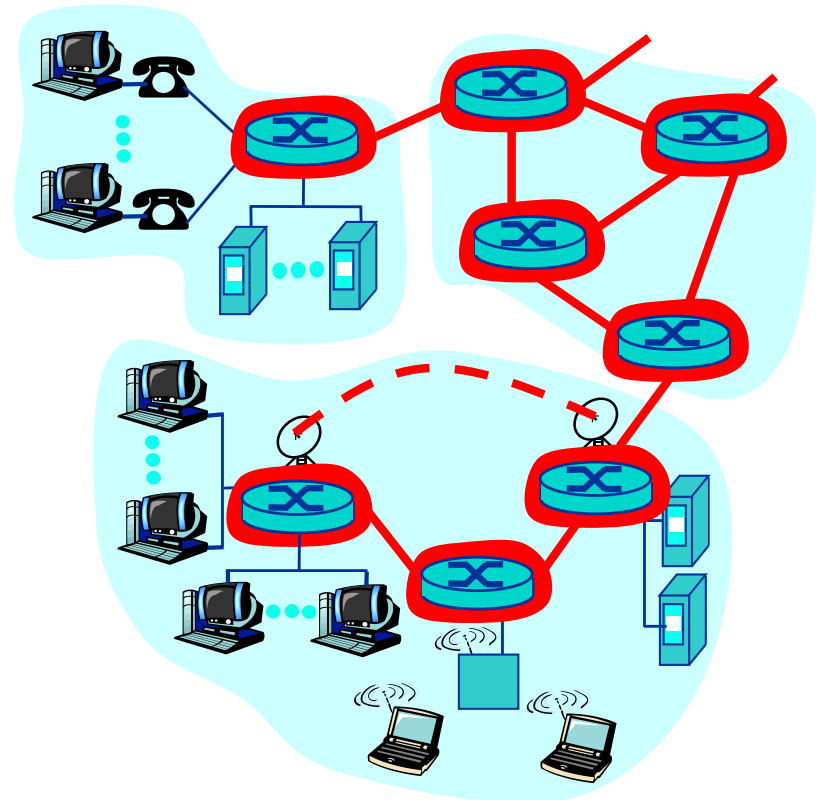
# Bordure

- Hôtes
  - exécutent des applications
    - WWW, e-mail...
  - se raccordent au cœur du réseau
- Modèle client/serveur
  - client envoie une requête
  - attend la réponse
  - reçoit un service de la part du serveur
  - e.g., client WWW (navigateur)/ serveur WWW, mais aussi: e-mail, FTP, ssh, X11...



# Cœur du réseau

- Interconnexion de systèmes intermédiaires
- *Choix crucial*: comment les données sont transportées ?
  - commutation de paquets : données découpées en paquets acheminés par les routeurs (*Internet*)
  - commutation de circuits : circuit dédié par appel et mis en place par les commutateurs (*Téléphone*)







## Mode paquet

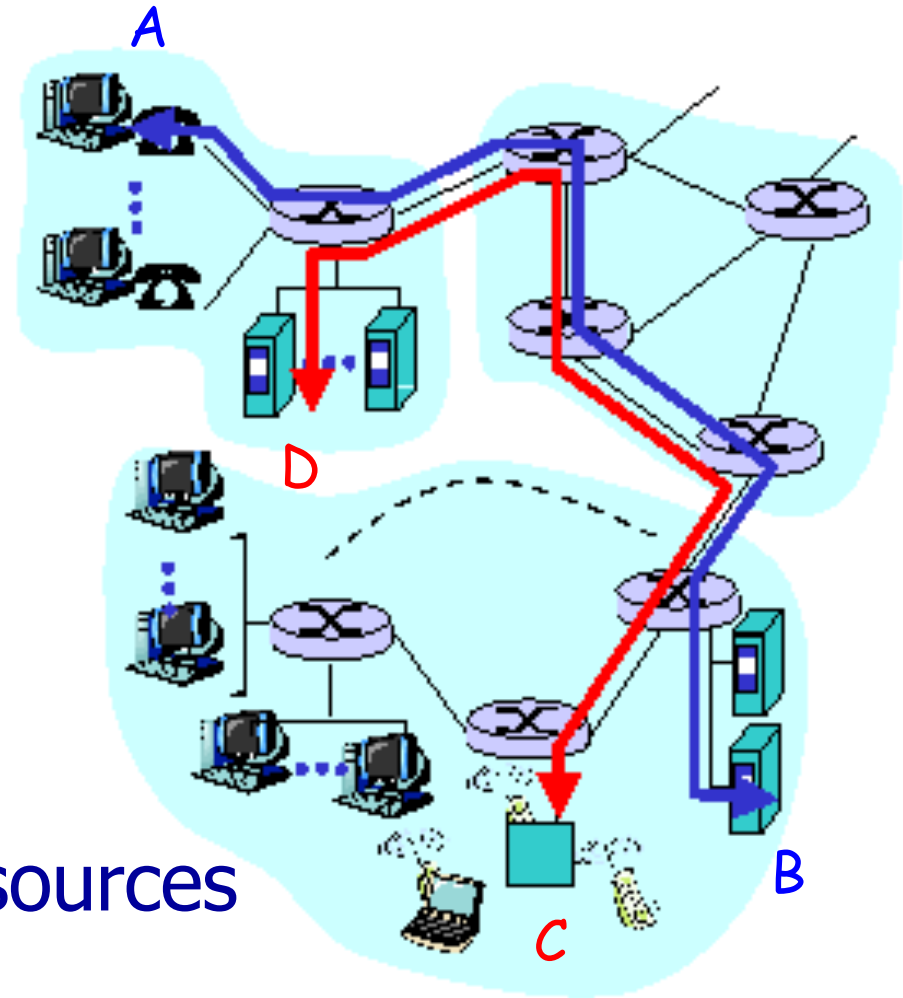
- Chaque paquet contient @exp et @dest en plus de l'info
- Chaque nœud route le paquet vers voisin en direction de @dest
- Chaque paquet suit son propre chemin, et est examiné et traité par chaque nœud

## Mode circuit

- Demande d'une connexion persistante (1 communication)
- Calcul initial d'un chemin
- Réservation des liens
- Le nœud connecte lien amont et aval
- Tous les paquets d'info suivent le chemin

# Mode circuit

- A demande une connexion à B
- Le réseau
  - trouve un chemin
  - vérifie l'accord de B
  - réserve les liens
  - aboute les liens
  - notifie A
- Envoi des données
- Fin de la connexion
- Le réseau libère les ressources



# Mode paquet

- Numérique seulement
- Surcoût d' en-têtes
  - overhead vs payload
- Partage du débit
  - optimise usage canal
- Débit aléatoire
  - « Best effort »
- Découplage ém/réc
  - Vitesses et débits  $\neq$
- Ordre non garanti

# Mode circuit

- Analogique ou num.
- Délai d' établissement
  - signalisation parallèle
- Débit réservé
  - gaspillage possible
- Débit stable garanti
  - Qualité de Service
- Synchro ém/réc
  - débit commun, phase
- Ordre d' émission préservé

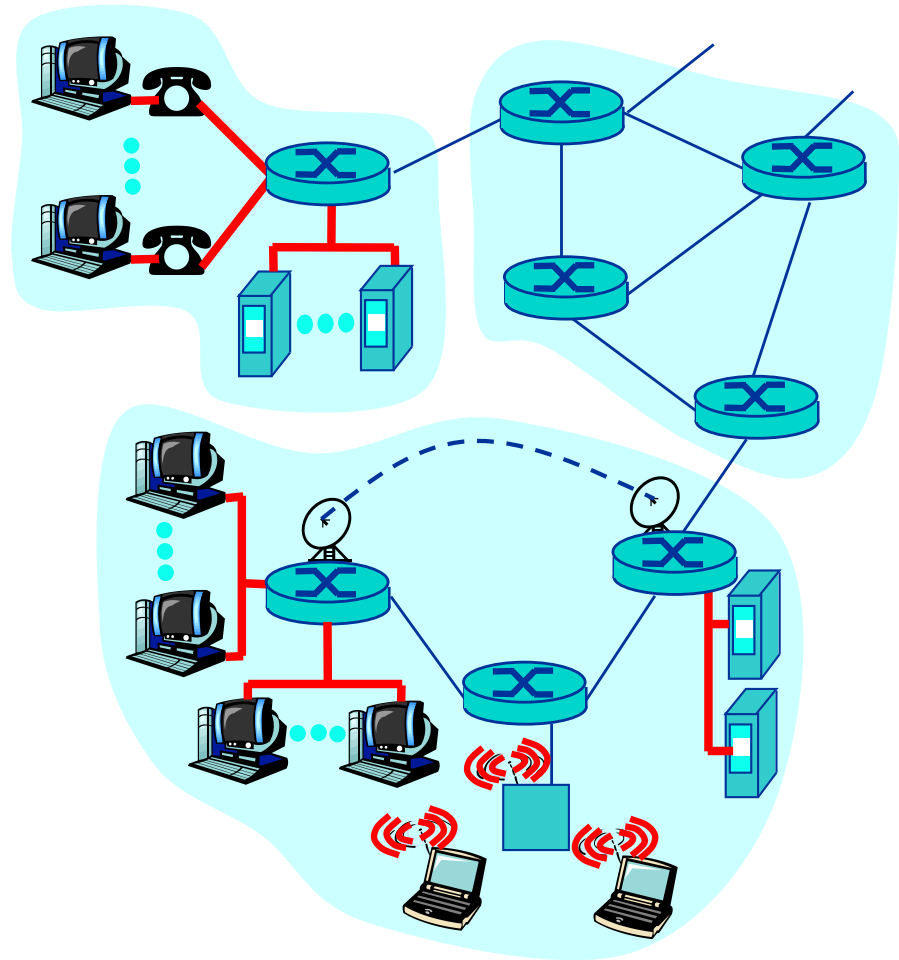
# Réseaux d'accès et canaux physiques

*Q: Comment se connecter au routeur de bordure*

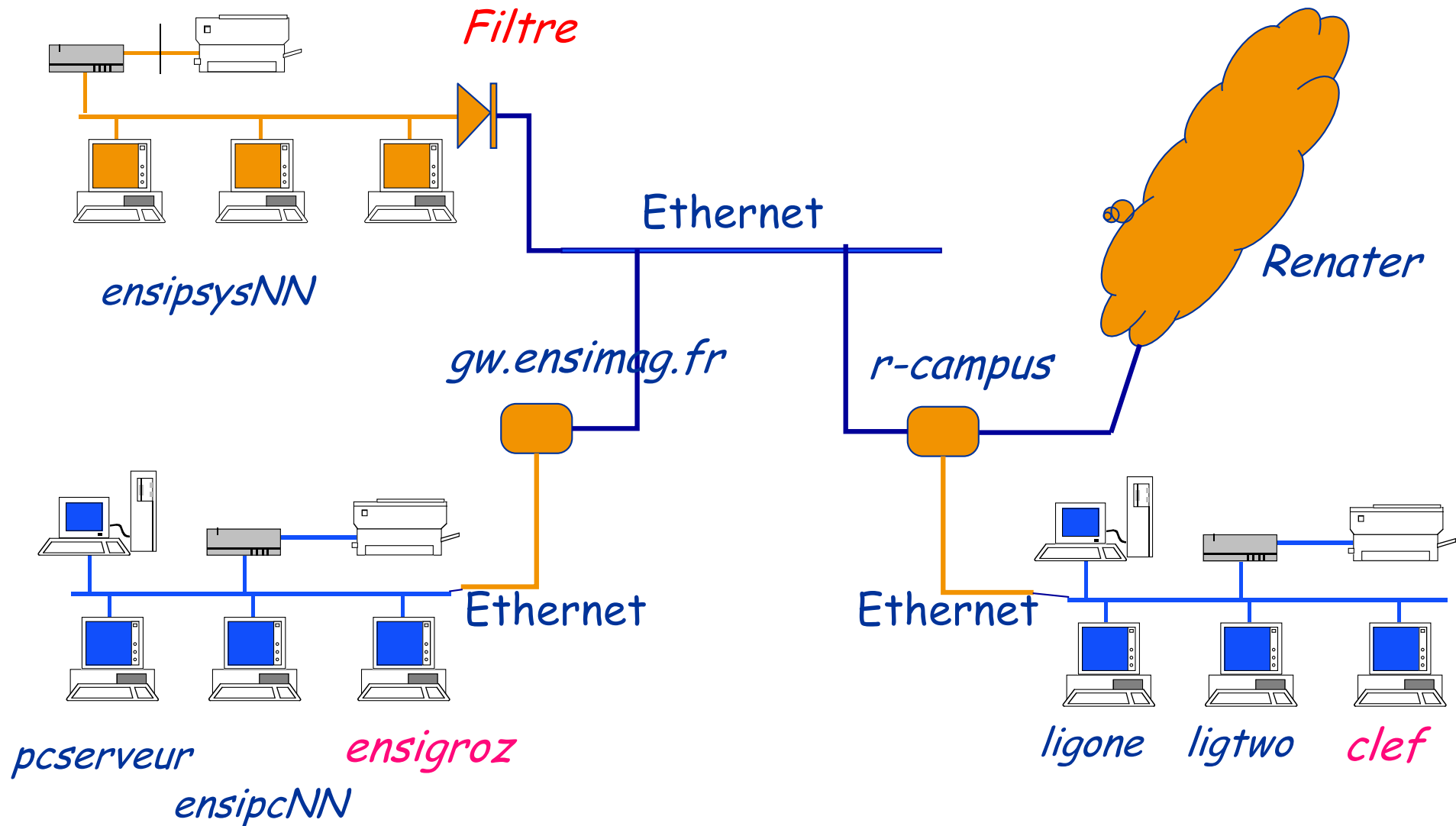
- réseaux d'accès résidentiels (modems, ADSL, Câble, FTTH...)
- réseaux d'accès institutionnels (Ethernet)
- réseaux d'accès mobiles (Wifi, GPRS, 3G...)

*Caractéristiques ?*

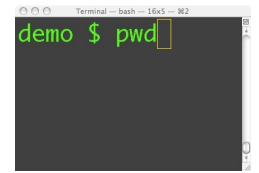
- débit nominal (33kb/s..1Gb/s)
- partagés ou dédiés



# Réseau (fixe) imag



# Connectivité avec un hôte



- Ping: utilitaire pour tester la connectivité

```
pcserveur:~ groz$ ping google.fr
```

```
PING google.fr (66.249.93.104): 56 data bytes
```

```
64 bytes from 66.249.93.104: icmp_seq=0 ttl=238 time=81.396 ms
```

```
64 bytes from 66.249.93.104: icmp_seq=1 ttl=238 time=78.160 ms
```

```
64 bytes from 66.249.93.104: icmp_seq=2 ttl=238 time=79.289 ms
```

```
64 bytes from 66.249.93.104: icmp_seq=3 ttl=238 time=77.682 ms
```

```
^C
```

```
--- google.fr ping statistics ---
```

```
4 packets transmitted, 4 packets received, 0% packet loss
```

```
round-trip min/avg/max/stddev = 77.682/79.132/81.396/1.432 ms
```



# Chemin jusqu' à un site grenoblois

Traceroute: chemin pour accéder à un hôte *(NB tracert en Windows DOS)*

```
pcserveur:~traceroute -I www.grenoble-em.com
```

```
traceroute to www.grenoble-em.com (217.167.24.44), 30 hops max, 38 byte packets
```

```
1  ensigate (195.221.228.1)  0.453 ms  0.304 ms  0.266 ms
2  r-campus.imag.fr (129.88.1.2)  0.321 ms  0.329 ms  0.251 ms
3  r-campus.grenet.fr (193.54.185.120)  0.341 ms  0.341 ms  0.231 ms
4  tigrel.grenet.fr (193.54.184.33)  0.364 ms  0.328 ms  0.360 ms
5  grenoble-g3-2.cssi.renater.fr (193.51.181.94)  1.113 ms  0.342 ms
   0.848 ms
6  lyon-pos13-0.cssi.renater.fr (193.51.179.237)  11.980
7  nri-b-pos9-0.cssi.renater.fr (193.51.179.13)  11.971 ms  11.832 ms
   11.805 ms
8  COGENT-COMMUNICATIONS-France.sfinx.tm.fr (194.68.129.164)  12.184 ms
   12.181 ms  12.090 ms
9  te3-3.mpd02.par01.atlas.cogentco.com (130.117.1.61)  13.211 ms
   13.049 ms  13.038 ms
10 tel-1.ccr01.lys01.atlas.cogentco.com (130.117.2.186)  20.087 ms
   20.176 ms  20.216 ms
11 149.6.118.50 (149.6.118.50)  21.509 ms  21.590 ms  21.519 ms
12 217.167.24.44 (217.167.24.44)  22.121 ms  22.390 ms  22.530 ms
13 217.167.24.44 (217.167.24.44)  23.126 ms  22.529 ms  22.853 ms
14 217.167.24.44 (217.167.24.44)  22.625 ms  22.689 ms  22.898 ms
telesun:~
```

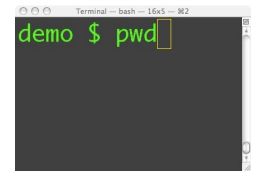
*Pourquoi passer  
par Lyon ?*

# Plan du chapitre AR1

- Architecture matérielle (Internet)
- Architecture logique en couches
  - Motivation de l'architecture en couches
  - Modèles OSI & TCP/IP
  - Cheminement des données
  - Exemples de piles protocoles



# Exemple: requête Web



`http://ensimag.grenoble-inp.fr`

Texte, images, animations:



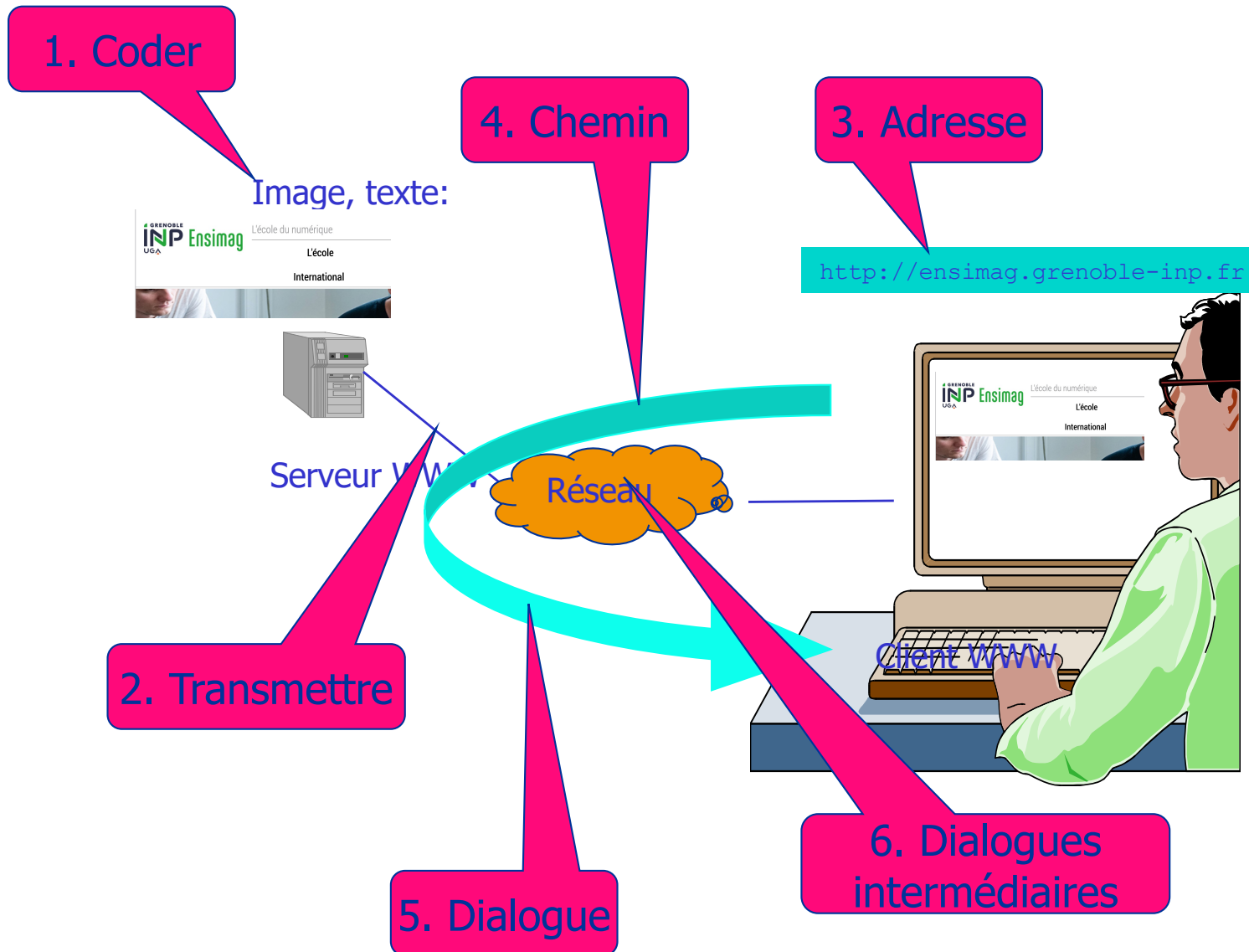
Réseau

Serveur WWW  
sur machine  
`magnolia.infra.grenoble-inp.fr`

Client WWW  
Navigateur  
(Firefox, IE...)



# Problèmes à résoudre





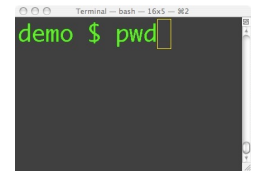
# Problèmes à résoudre

1. Représenter/numériser l'information → codage  
*cf cours Théorie de l'information  
et cours de codages (2A: Codage Audio & Vidéo etc)*
2. Produire les signaux adaptés → transmission  
*chapitre CN*
3. Allouer, puis trouver l'adresse  
*chapitres AR2 (adresses) et AR3 (annuaire DNS etc)*
4. Acheminer → routage, contrôle flux&pertes etc.  
*cours Réseaux 2A (algorithmes&protocoles, informatique)*
5. Dialoguer avec machine distante → protocoles  
*chapitre PR (protocoles)+AP (www)  
(algorithmique, logique, probas)*
6. Dialoguer via machines locales → protocoles  
*chapitre AR+PR (encapsulation, interconnexion)*

# Pb3 - Adressage

- L'utilisateur connaît « ensimag.grenoble-inp.fr »
  - Représentation textuelle
  - Explicite, compréhensible, mémorisable
  - Redondante (beaucoup de caractères: peu compact)
- Les machines sont désignées sur le réseau par des numéros (adresses IP...): 195.220.30.60
  - Acheminement des paquets selon l'architecture du réseau
  - Codage compact
- Annuaire: nom DNS <-> IP
  - Domain Name System: l'annuaire est « en ligne »

# Consultation annuaire puis web



Texte, images, animations:



Annuaire DNS



1b

195.220.30.60

1a

ensimag.grenoble-inp.fr ?

2a

Envoi à 195.220.30.60



Serveur WWW  
sur machine  
magnolia.infra.grenoble-inp.fr

Réseau

2b



Client WWW  
Navigateur  
(Firefox,IE...)

## Pb 4 - Chemins

- Chaque message va de nœud en nœud
  - Chaque paquet comporte l'adresse IP du destinataire (et de l'émetteur)
- Chaque nœud se base sur l'adresse du destinataire pour déterminer lequel de ses voisins est plus proche de la destination
- Le calcul du chemin « optimal » a été pré-calculé, enregistré dans la table de routage de chaque nœud traversé.

# Identification du chemin

*Par où est-on passé ?*

- *Satellite*
- *Câble sous-marin ?*



## • Traceroute: chemin pour accéder à un hôte (*NB tracert en Windows DOS*)

```
grozr@pcserveur:~$ traceroute ucla.edu
traceroute to ucla.edu (128.97.27.37), 30 hops max, 60 byte packets
 1  gw.ensimag.fr (147.171.104.1)  2.001 ms  1.743 ms  5.715 ms
 2  bertholet.enseeg.inpg.fr (195.220.32.1)  0.882 ms  0.753 ms  0.694 ms
 3  r-campus1.grenet.fr (193.54.184.24)  1.058 ms  1.003 ms  0.935 ms
 4  tigrel.grenet.fr (193.54.185.1)  4.618 ms  4.482 ms  4.351 ms
 5  tel-4-grenoble-rtr-021.noc.renater.fr (193.51.181.94)  1.317 ms  1.157 ms  1.027 ms
 6  te-0-1-0-12-ren-nr-lyon2-rtr-091.noc.renater.fr (193.51.180.67)  3.741 ms  3.057 ms  3.164 ms
 7  xe-0-1-7-lyon1-rtr-131.noc.renater.fr (193.51.177.167)  2.956 ms xe-1-0-6-ren-nr-lyon1-rtr-
    131.noc.renater.fr (193.51.180.56)  4.747 ms xe-1-0-7-ren-nr-lyon1-rtr-131.noc.renater.fr
    (193.51.177.242)  4.612 ms
 8  renater.mx1.gen.ch.geant.net (62.40.124.61)  5.371 ms  5.103 ms  5.282 ms
 9  ae6.mx1.par.fr.geant.net (62.40.98.183)  13.471 ms  13.334 ms  13.168 ms
10  hundredge-0-0-0-22.102.core1.newy32aoa.net.internet2.edu (198.71.45.236)  88.418 ms  90.234 ms
    90.030 ms
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  hpr-lax-aggr10--i2.cenic.net (137.164.26.200)  151.903 ms  151.776 ms  151.927 ms
20  * * *
21  bd10f1.anderson--cr10f1.anderson.ucla.net (169.232.4.6)  151.885 ms bd10f1.anderson--
    cr00f2.csbl.ucla.net (169.232.4.4)  152.250 ms  152.135 ms
22  cr10f1.anderson--sr02fb.jsei.ucla.net (169.232.8.53)  152.274 ms  152.256 ms  152.170 ms
23  128.97.27.37 (128.97.27.37)  152.380 ms !X  152.212 ms !X  152.038 ms !X
```

# Problèmes à traiter: les fonctions du réseau

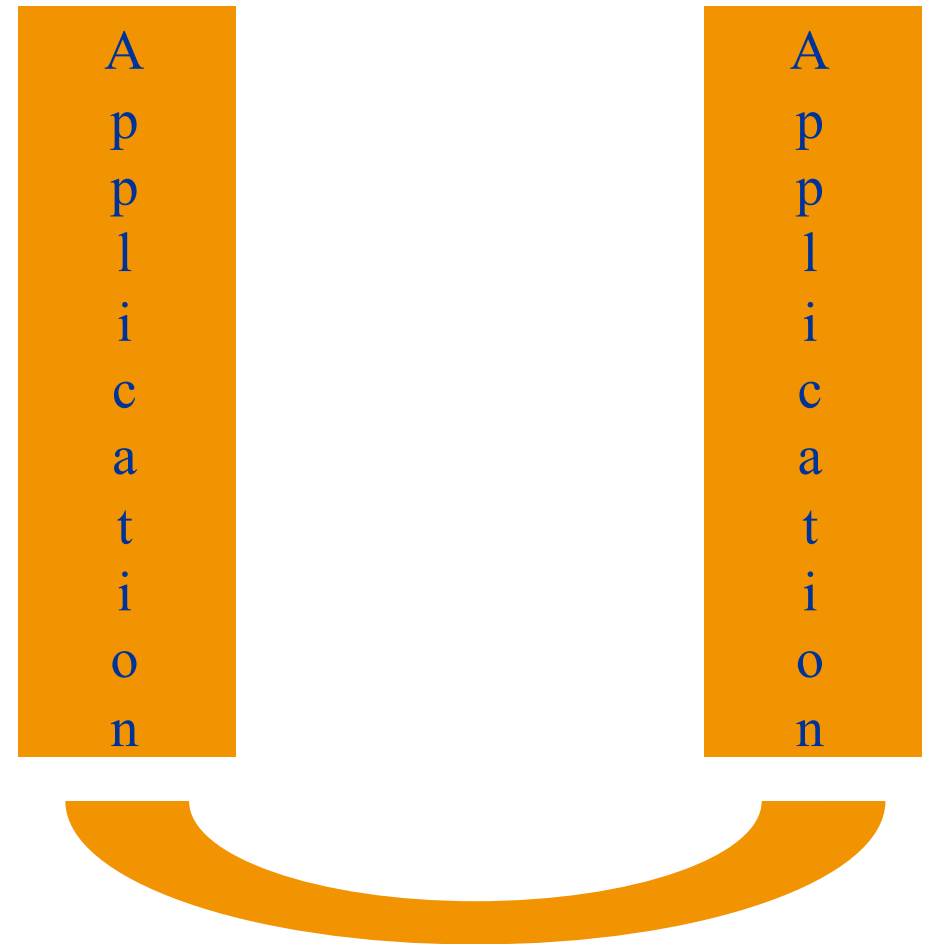
- Pb 3: Recherche d'adresses
- Pb 4: Acheminement de proche en proche
  - calcul du routage
  - correspondance adresse/liens
- Ouverture dialogue
  - Pb 5: avec extrémité
  - Pb 6: avec les noeuds intermédiaires
- Pb 7: Découpage en paquets
- Pb 8: Contrôle pertes - retransmission
- Pb 9: Contrôle de flux:  
émetteur plus rapide que dest. ou canal ou machines intermédiaires
- ...



# 1ère approche

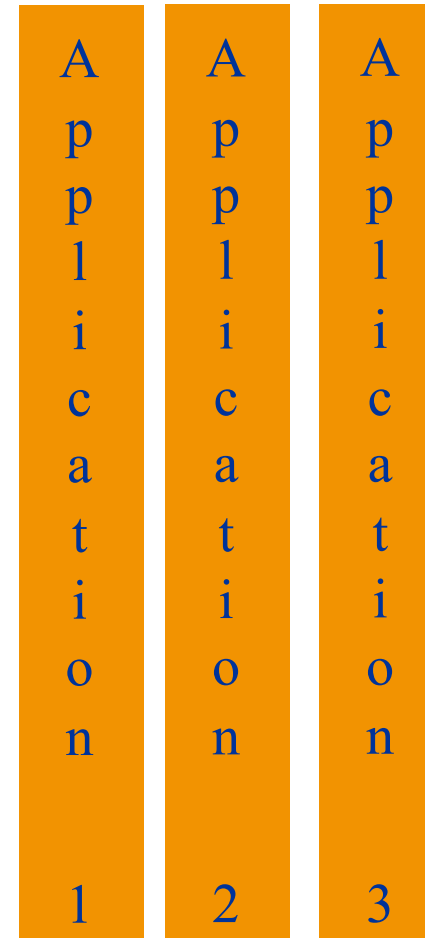
(héritage téléinfo: terminal->centre de calcul)

- Réseau de base=tuyau
  - transport de caractères
  - sur une ligne entre émetteur-récepteur
- Chaque application doit gérer son tuyau, i.e. assurer les fonctions (contr. flux, adressage, erreurs, routage etc...)



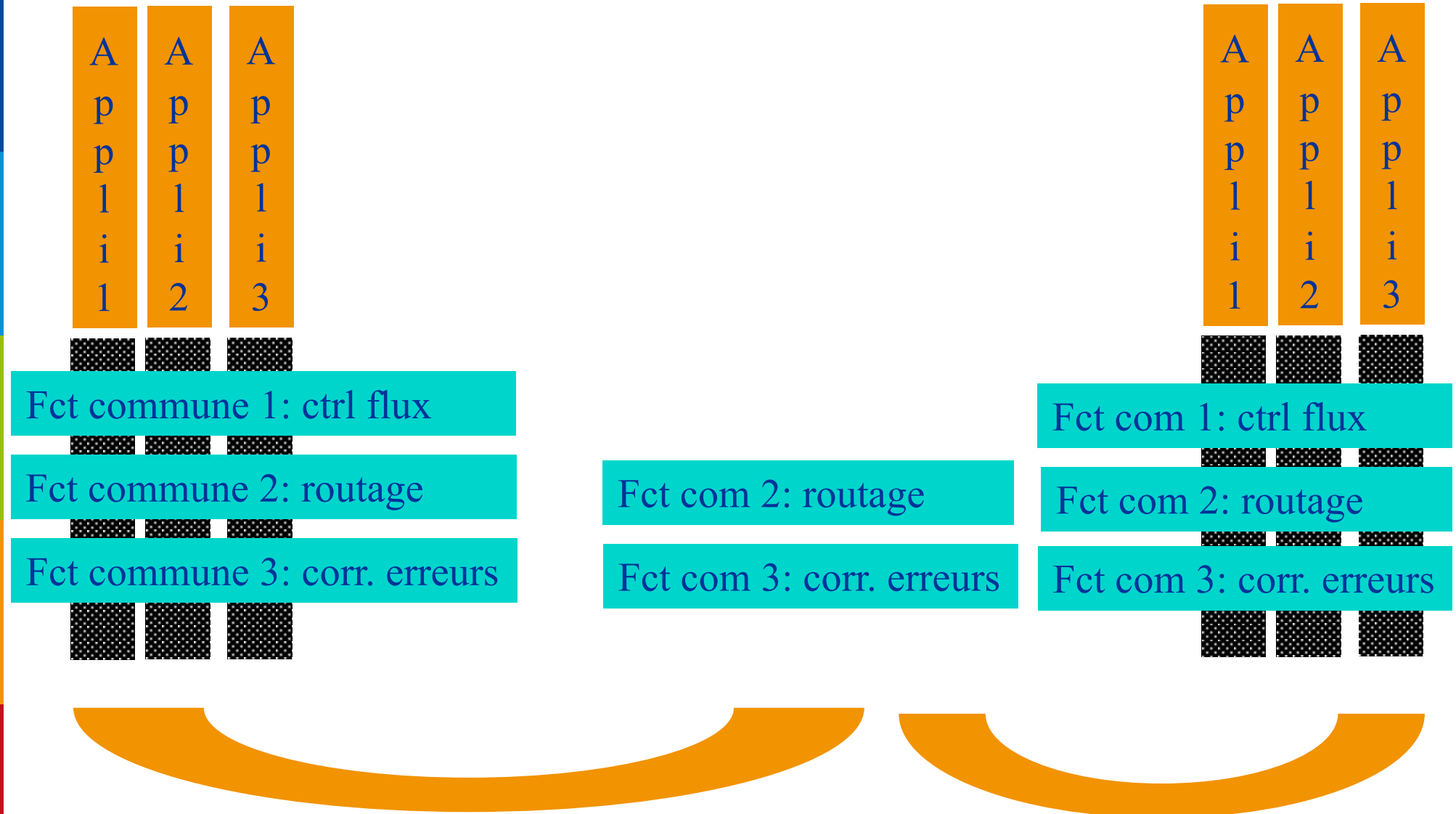
# Inconvénients de la 1ère approche

- Dialogue entre applis homogènes
- Réplication des fonctions communes
- Applications doivent « tourner » sur chaque machine intermédiaire

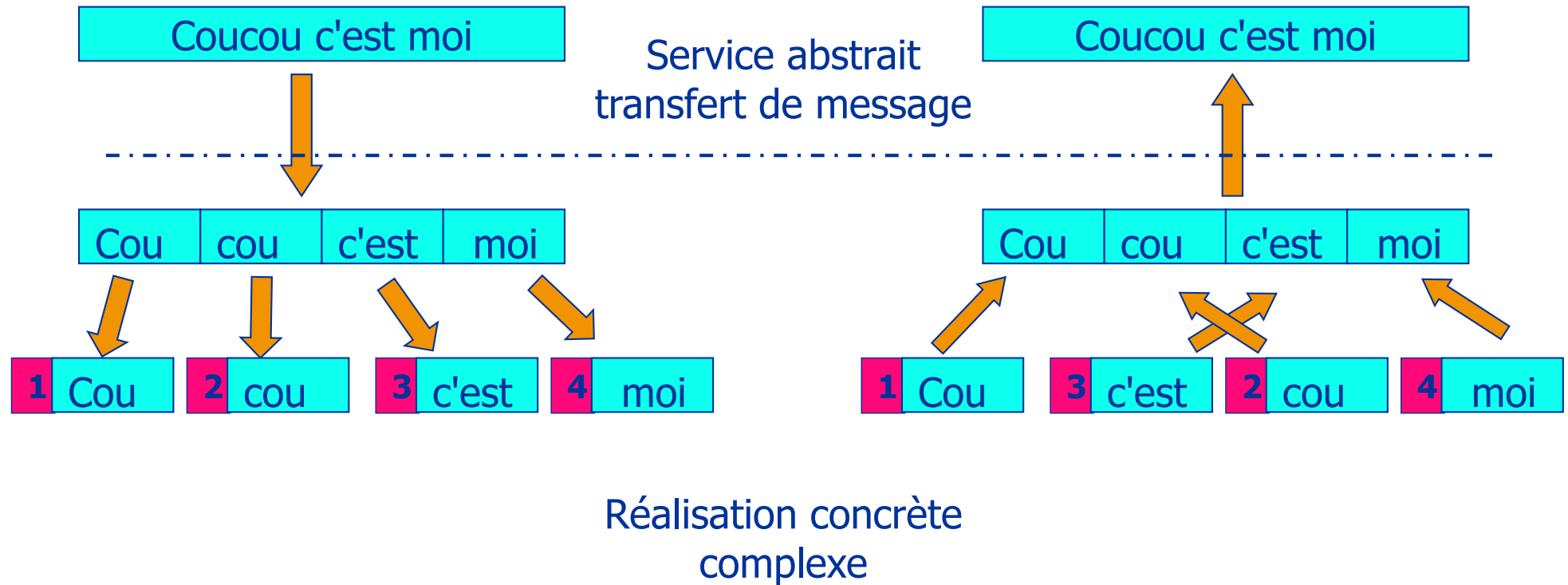


# Architecture en couches

## *Layered protocol stack*



# Pb7 - Segmentation

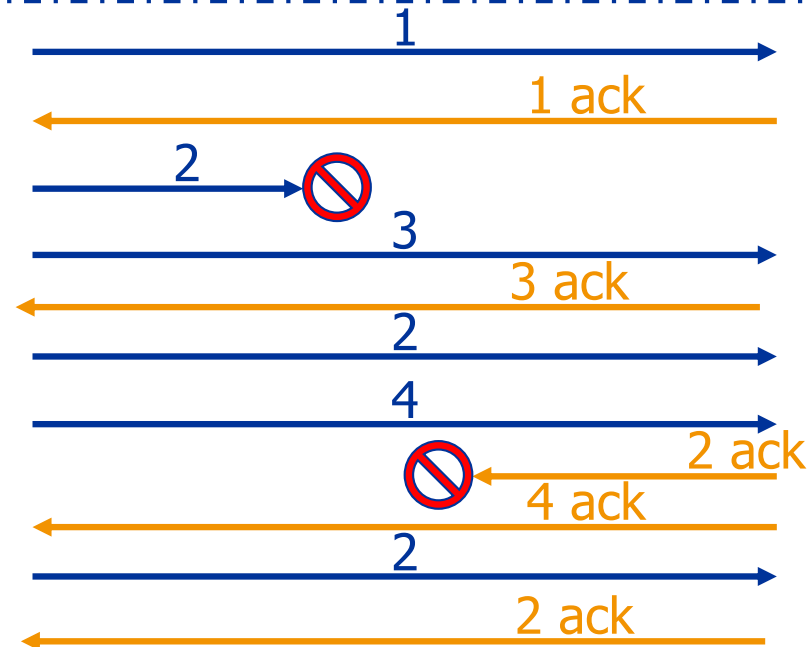


# Pb8 - Contrôle pertes & retransmission

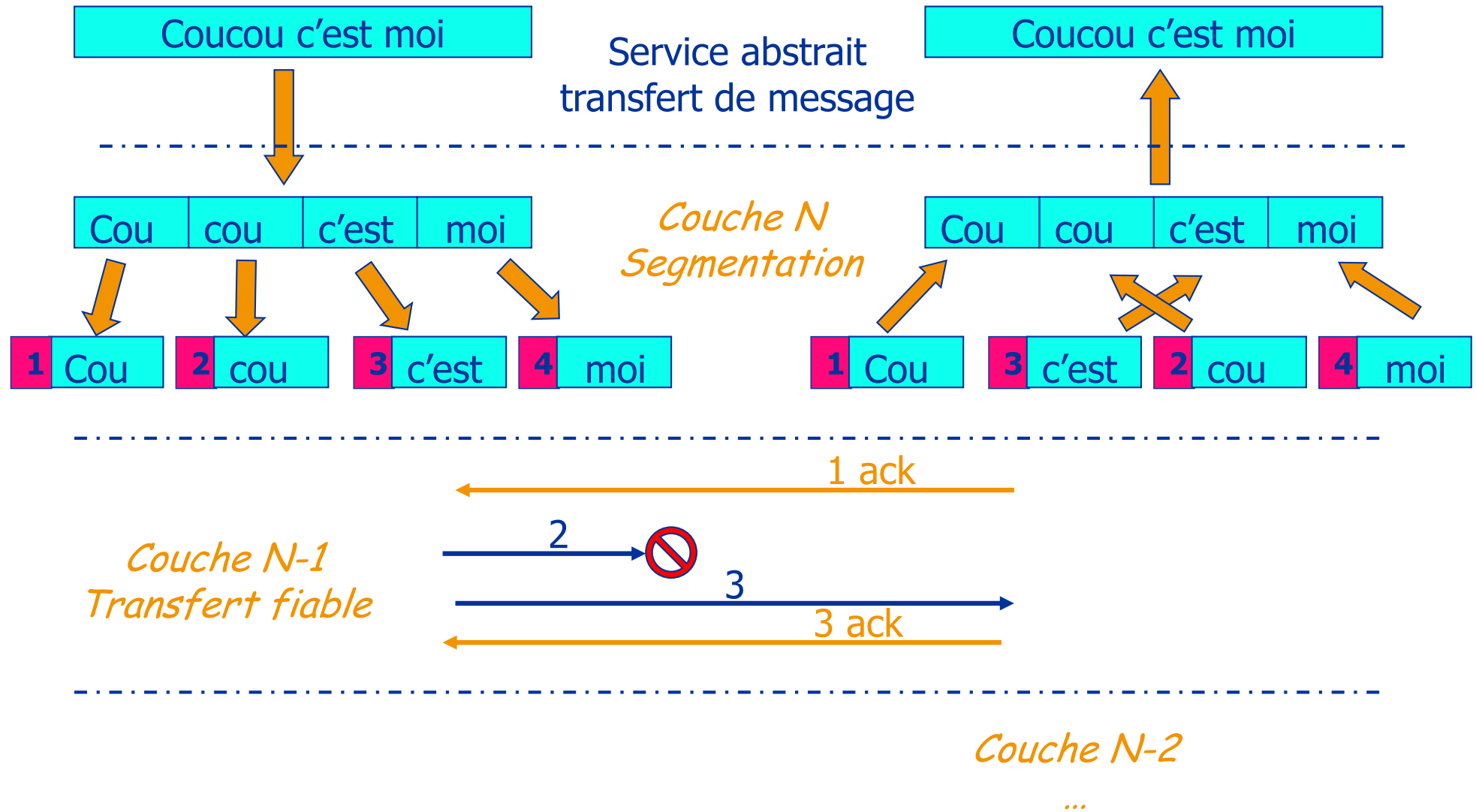
Service abstrait  
transfert sans perte



Réalisation concrète  
complexe

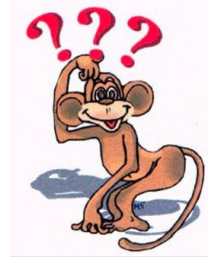


# Superposition de couches

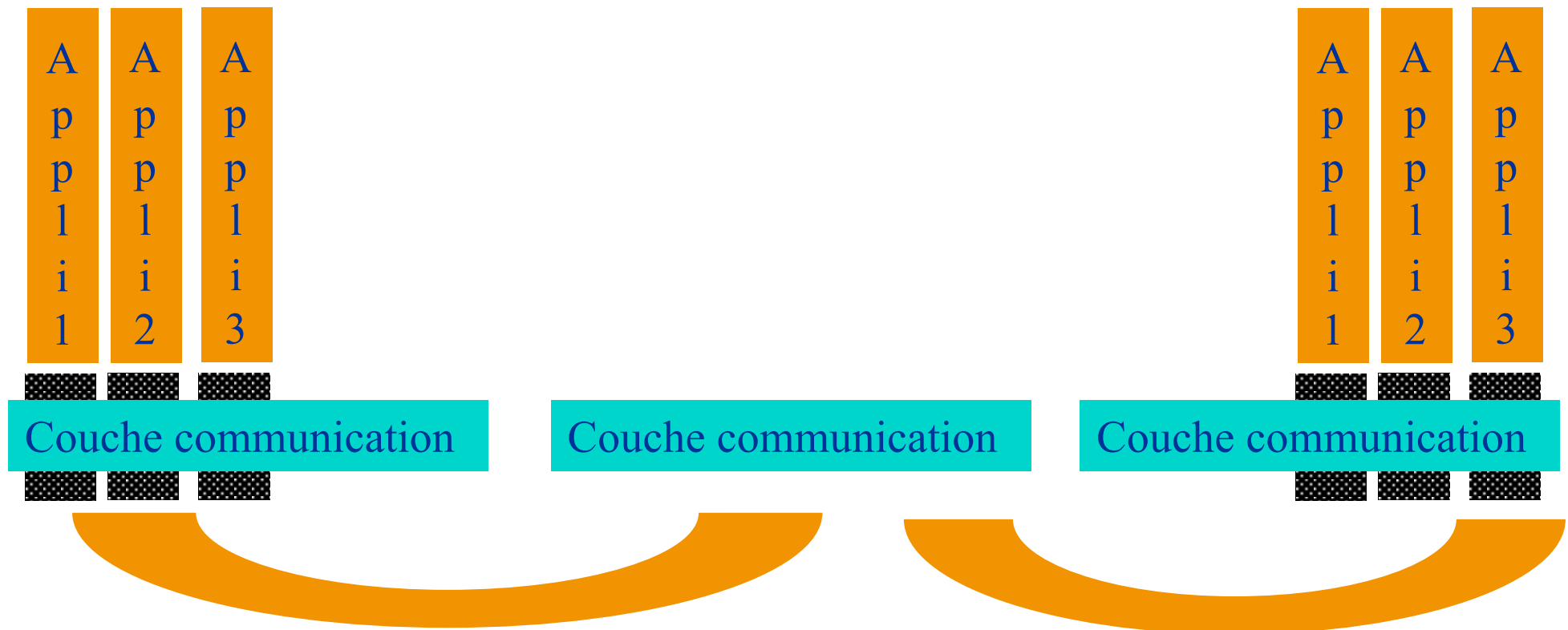


# Pourquoi plusieurs couches ?

*et non pas 1 couche « communication » unique*



*Pourquoi faire simple quand on peut faire compliqué ! :-)*





# Avantages de l'architecture en couches

- Réutilisation/partage des fonctions communes
- Facilité de relier des machines hétérogènes
- Applications en extrémité seulement  
*mais aussi*
- Séparation des problèmes
- ABSTRACTION, indépendance des particularités  
(« information hiding »)  
de lien, de réseau, de codage...  
Spécification / implémentation
- Optimisation par niveau (avantage/inconvénient)

*Inconvénient: surcoût de performance*



# Analogie: bibliothèques de calcul

Application

- Ex: Acroread, xdvi, Firefox...

Symboles

- Dessin de polices de caractères

Bibl. Graphique

- Assemblages de fct: splines, arcs...

Biblio C

- Fct math élémentaires: cos, sin, exp, log

Câblé

- Add, Sub, Mul, Div de nbres 32 bits



# Modèle OSI de l'ISO

*A connaître par coeur*

7- Application

- Fonctions communes + applis

6- Présentation

- Format interchangeable, terminal virtuel

5- Session

- Organisation du dialogue

4- Transport

- Communication fiable entre processus

3- Réseau

- Acheminement à travers le réseau

2- Liaison

- Transmission point à point (sur 1 lien)

1- Physique

- Emission/réception des signaux é-mag.



# Modèle OSI de l' ISO

## *Open Systems Interconnection*

### *Organisation Internationale de Normalisation*

Début des années 1980...

- Systèmes ouverts
  - spécifications de protocoles publics
  - Interopérabilité entre systèmes hétérogènes
- Modèle: pérenne (terminologie, concepts)
- Protocoles normalisés: en grande partie abandonnés  
(OSI rattrapé par IP et Internet)

*Années 1960-70= tâtonnements, communication non fiable, interblocages etc → pbs résolus par architecture en couches*

# Modèle TCP/IP

Application

- DNS, FTP, WWW, telnet, SMTP

Transport

- Communication fiable entre processus

Réseau

- Acheminement à travers le réseau

Liaison

- Transmission point à point sur lien

Physique

- Émission/réception électrique

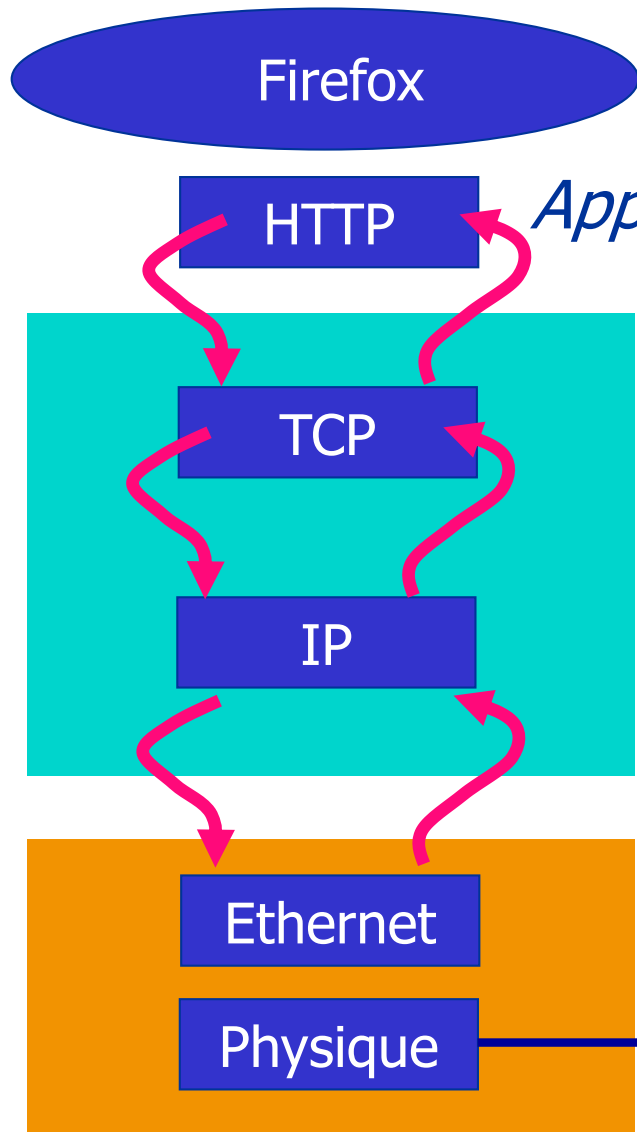
# Principales fonctions des couches 1-3

- Couche 1 physique (*chapitre CN & RA + 2A-3A*)
  - codage canal, modulation pour transmettre 1 symbole sur le lien physique
  - synchronisation émetteur/récepteur
- Couche 2 liaison (*2A-3A*)
  - assemblage de symboles (bits) en trames
  - détection erreurs de transmission
  - contrôle de flux
- Couche 3 réseau (*2A-3A*)
  - routage (pré-calculs chemins + routage paquets)
  - signalisation entre nœuds (découverte du voisinage, problèmes d'acheminement)

# Principales fonctions des couches 4+

- Couche 4 transport (*chapitre PR1 + 2A*)
  - dispatcher les flux entre processus d'une machine
  - découpe des flux, segmentation, réassemblage
  - adaptation aux pbs réseaux (pertes, congestion...)
- Couche 5 session
  - synchronisation multi-points (tour de parole...)
  - transactions (reprise sur état antérieur si pb)
  - authentification, chiffrement (*chapitres NF, AP*)
- Couche 6 présentation
  - formats communs de codage des informations (*PR3*)
  - adaptation des formats aux terminaux

# Comment l'information quitte la machine: couches logicielles



- Former la requête HTTP

appel de  
procédure  
+ tampons

- Découper en paquets, numéroté

O/S

- Attacher l'adresse de destination

- Envoyer sur réseau local au routeur

Carte

électronique

- Bits encodés en signaux

# Principaux protocoles / couche

7- HTTP, SMTP, Telnet, FTP, SSH, NFS, DNS, Whois...

6- XDR, ASN.1, MIME, XML... (formats), X11 (protocole)

5- SSL, TLS, RPC, NetBIOS, H.323

4- TCP, UDP, RTP, MPTCP, SCTP

3- IP, ICMP, OSPF, RIP, ARP, X.25, ATM

2- Ethernet, HDLC, 802.11 (Wifi), PPP, SLIP

1- Ethernet 100BaseT, 802.11 physique, ADSL, modems V.nm ...



# Exemples d'empilements

HTTP	HTTP	eMule	DNS
TCP	TCP	TCP+UDP	UDP
IP	IP	IP	IP
LLC/MAC	PPP	PPPoE	Liaison 802.11
Ethernet	Modem 56k V.92	Modem ADSL	IEEE 802.11
<i>PC virtuel sur Rx fil Ensimag</i>	<i>PC via liaison téléphonique</i>	<i>PC via ADSL</i>	<i>PC portable via liaison sans fil WiFi</i>



# Bilan chap. AR1: notions essentielles

- Terminologie Internet
  - FAI, hôtes, routeur; paquets, trames
- Mode Paquet vs Circuit
- Quelques problèmes des réseaux par paquet
  - Adressage, routage, segmentation
- Architecture en couches
  - Principes « information hiding » + « separation of concerns »
  - Numéros et noms des 7 niveaux
  - Fonctions associées à chacune des 7 couches
- *Graphiques: diagrammes de couches*