

# Introduction aux réseaux

---

**Renaud Lachaize**  
Université Grenoble Alpes  
renaud.lachaize @ imag.fr

Mai 2019

Ce cours est basé sur les diapositives de Sacha Krakowiak

# Bibliographie sommaire sur les réseaux

---

## ■ Principes

- ◆ A. Tanenbaum, D. Wetherall, *Computer Networks* (5th edition). Prentice Hall, 2010.
  - ❖ Version française : *Les réseaux*, 5ème édition, Pearson Education, 2011.
  - ❖ Remarque : les éditions antérieures ont un seul auteur (Tanenbaum)
- ◆ L. Peterson, B. S. Davie, *Computer Networks, a systems approach*, 5th edition, Morgan Kaufmann, 2011.
  - ❖ Il existe une traduction française mais seulement pour la 1ère édition actuellement : *Réseaux d'ordinateurs : une approche orientée système*, Vuibert)
- ◆ J. F. Kurose, K. W. Ross, *Computer Networking, a top-down approach featuring the Internet*, Addison-Wesley, 7th ed. 2016.
  - ❖ Il existe une traduction française d'une édition plus ancienne : *Analyse structurée des réseaux*, Pearson Education)

## ■ Programmation réseau en environnement Unix / langage C

- ◆ R. Bryant, D. O'Hallaron, *Computer Systems: A Programmer's Perspective*, Prentice Hall, 2003 (2<sup>nd</sup> edition 2010).
- ◆ S. A. Rago, W. R. Stevens, *Advanced Programming in the Unix Environment (2nd Edition)*, Addison-Wesley, 2005.
- ◆ C. Blaess, *Programmation système en C sous Linux (2ème édition)*, Eyrolles, 2005.
  - ❖ Il existe une troisième édition avec un titre différent : *Développement système sous Linux (Eyrolles, 2011)*
- ◆ J.-M. Rifflet, R. Yunès, *Unix : programmation et communication*, Dunod, 2003.

# Bibliographie sommaire sur les réseaux (suite)

---

## ■ Livres utiles pour la spécialité ISN

- ◆ Ouvrage collectif. Introduction à la science informatique pour les enseignants de la discipline en lycée. CRDP Académie de Paris, 2011.
  - ❖ Chapitre 5 sur les réseaux.
  - ❖ Disponible en ligne : <https://wiki.inria.fr/wikis/sciencinfolycee/images/5/57/LSICh5.pdf>
  
- ◆ Ouvrage collectif. Informatique et sciences du numériques. Spécialité ISN en Terminale S. Eyrolles, 2012
  - ❖ Chapitre 16 sur les réseaux.
  - ❖ Disponible en ligne : [http://wiki.inria.fr/sciencinfolycee/Fichier:Informatique et Sciences du Numérique - Spécialité ISN en Terminale S.pdf](http://wiki.inria.fr/sciencinfolycee/Fichier:Informatique_et_Sciences_du_Numérique_-_Spécialité_ISN_en_Terminale_S.pdf)

## ■ Divers

- ◆ S.Cateloin, A. Gallais, S. Marc-Zwecker, J. Montavont. Mini-manuel des réseaux informatiques. Dunod, 2012.
  - ❖ Un ouvrage assez compact conçu pour le premier cycle universitaire.
- ◆ O. Alphand, A. Duda, F. Rousseau. Computer networking - Explanation of the main concepts with example exercises. Ensimag, 2016.
  - ❖ <http://duda.imag.fr/m1/notes/index.html>
- ◆ G. Cizault (nom d'emprunt d'un collectif d'universitaires et d'ingénieurs). IPv6 – théorie et pratique. Livre disponible en ligne : <http://livre.g6.asso.fr>

# Bibliographie sommaire sur les réseaux (suite)

---

## ■ MOOCs (cours gratuits en ligne, avec vidéos, supports et exercices)

La plupart de ces cours ont de nouvelles sessions organisées régulièrement. L'inscription n'engage à rien et permet de conserver l'accès aux supports au delà des dates de la session.

- ◆ Voir notamment les cours disponibles sur la plateforme FUN (France Université Numérique) : <https://www.fun-mooc.fr>

- ❖ Principes des réseaux de données
- ❖ Les réseaux locaux
- ❖ Routage et qualité de service dans l'Internet
- ❖ Objectif IPv6 : vers l'Internet nouvelle génération

- ◆ Voir aussi les cours (en anglais) disponibles sur les plateformes EdX (<https://www.edx.org>) ou Coursera (<https://www.coursera.org>)

- ❖ N.B. : Les certificats de réussite aux cours sont payants mais l'accès aux contenus est libre/gratuit
- ❖ Par exemple :
  - ▲ The bits and bytes of computer networking (<https://www.coursera.org/learn/computer-networking>)
  - ▲ Networks illustrated: principles without calculus (<https://www.coursera.org/learn/networks-illustrated>)
  - ▲ Fundamentals of network communication (<https://www.coursera.org/learn/fundamentals-network-communications>)

# Introduction aux réseaux informatiques

---

- Un **réseau informatique** (*computer network*) est un système de communication (ensemble matériel + logiciel) qui permet à un ensemble d'ordinateurs (au sens large) d'échanger de l'information
  - ◆ sens large : points d'accès, terminaux de paiement, téléphones, capteurs divers, etc.
  
- L'échange d'information n'est pas une fin en soi. Les réseaux servent avant tout à réaliser des **services**
  - ◆ accessibles à partir de tout organe connecté au réseau
  - ◆ mis en œuvre par un ensemble d'ordinateurs sur le réseau
  - ◆ exemples de services
    - le courrier électronique (*mail*)
    - le transfert de fichiers (*ftp*)
    - l'accès à distance (*telnet*, *ssh*)
    - l'accès au World Wide Web
    - les services utilisant le Web :  
documentation, commerce électronique, etc.
  
- Lien entre réseaux et systèmes d'exploitation
  - ◆ Le réseau (support de communication) comme organe d'entrée-sortie
  - ◆ Le réseau (ensemble de serveurs) comme super-machine (*clusters*, *grids*, *clouds*)

# Le défi de la puissance de calcul à bon marché : des grappes aux nuages

---

## ■ Les grappes (*clusters*)

- ◆ Groupe de machines sur un réseau local (dizaines, centaines)
- ◆ Puissance et disponibilité pour la réalisation de services

## ■ Les grilles (*grids*)

- ◆ Calcul parallèle à grande échelle (milliers, dizaines de milliers)
- ◆ Grappes réparties sur l'Internet
- ◆ Gestion globale des ressources
- ◆ Concurrentes de supercalculateurs pour grandes applications scientifiques

## ■ Les nuages (*clouds*)

- ◆ Virtualisation des ressources de calcul

Diapositive tirée de S. Krakowiak, Histoire et actualité de l'informatique, 2011.

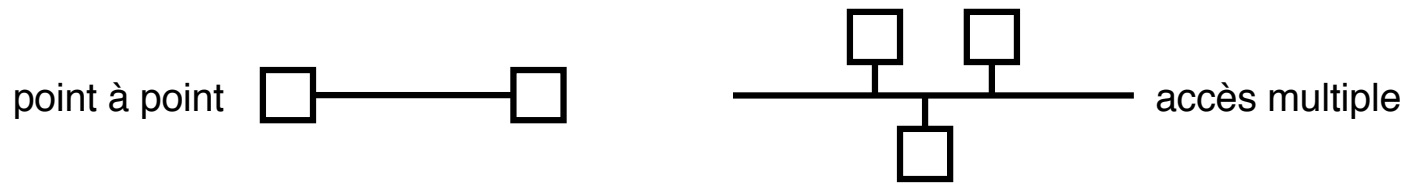
# Types de réseaux (1)

## ■ Les réseaux peuvent être classés selon différents critères

### ◆ Nature de la liaison entre les organes connectés (nœuds)

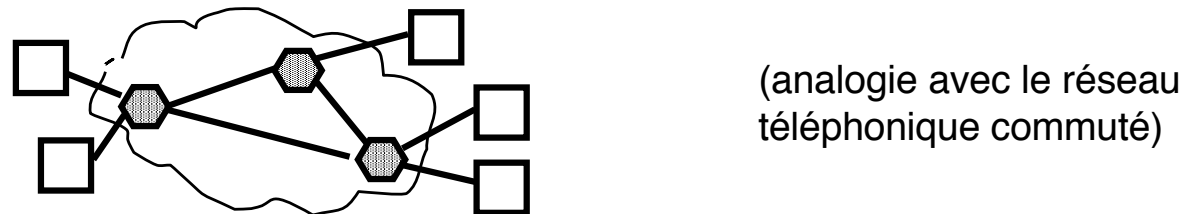
#### ❖ Liaison directe

- ▲ il y a un lien direct entre deux nœuds du réseau



#### ❖ Liaison commutée

- ▲ la liaison passe par des organes intermédiaires



#### ❖ Les supports physiques de la communication peuvent être divers

- ▲ paires de fils, câble coaxial, fibre optique, radio, infra-rouge, etc.

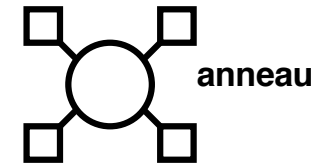
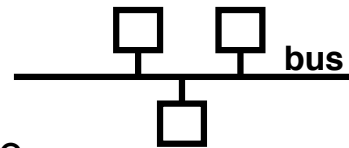
### ◆ Couverture géographique (réseau local, à grande distance, etc.)

# Types de réseaux (2)

La classification par étendue de la couverture géographique est souvent utilisée, bien que non stricte

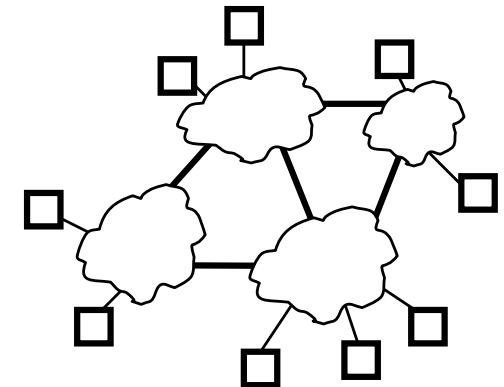
## ■ Réseaux locaux (*Local Area Networks, LAN*)

- ◆ Communication au sein d'une organisation (département d'entreprise, etc.)
- ◆ Administration unique
- ◆ Couverture géographique limitée (~1 km)
- ◆ Débit élevé, taux d'erreur faible
- ◆ Topologies diverses : bus, anneau, étoile, arbre ...



## ■ Réseaux à grande distance (*Wide Area Networks, WAN*)

- ◆ Communication entre des organisations (souvent) diverses
- ◆ Administrations (souvent) multiples
- ◆ Couverture géographique étendue : un pays, toute la planète
- ◆ Débit variable, taux d'erreur parfois non négligeable
- ◆ Topologie maillée ; interconnexion de réseaux (exemple : l'Internet)



## ■ Réseaux métropolitains (*Metropolitan Area Networks, MAN*)

- ◆ Intermédiaires entre LAN et WAN - qq dizaines de km, ville ou région

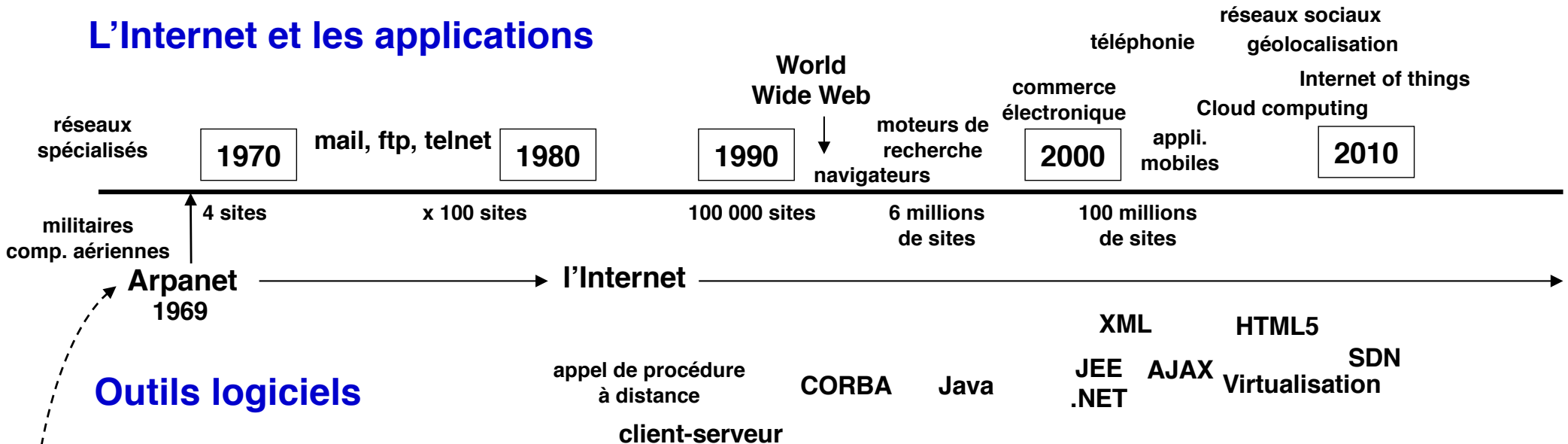
## ■ Autres réseaux

- ◆ *System & Storage Area Networks (SAN)*, *Desk Area Networks (DAN)* ...
- ◆ Réseaux de mobiles, réseaux de capteurs, ... (potentiellement ad hoc)



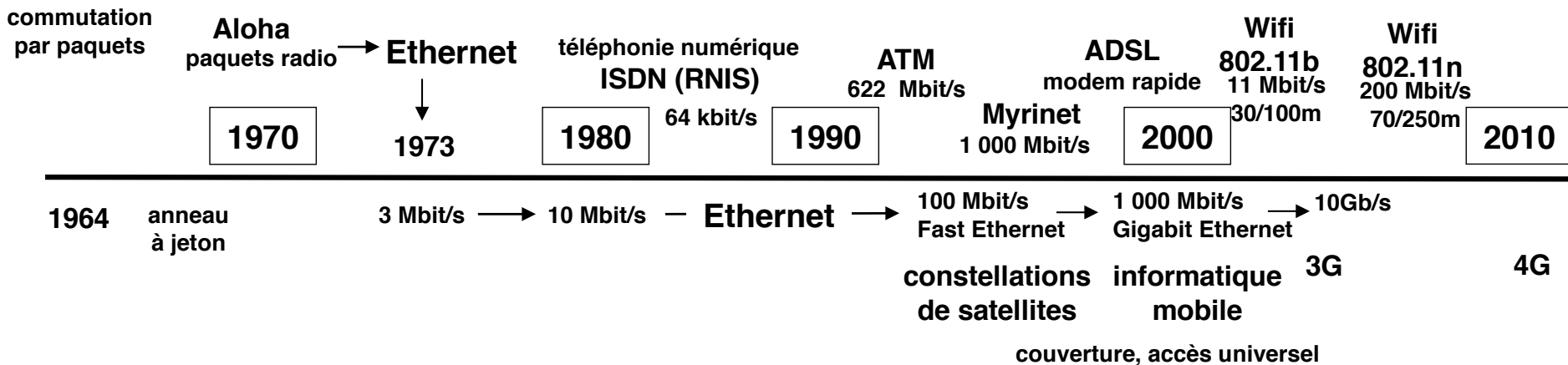
# Bref historique des réseaux informatiques

## L'Internet et les applications



## Outils logiciels

## Quelques avancées technologiques



# Performances des réseaux (1)

---

## ■ Deux mesures principales de performance

- ◆ **Débit** (*throughput*) : quantité d'information par unité de temps. Unité : bit/s (Kbit/s, Mbit/s, ...)
  - ❖ une notion corrélée au débit est la bande passante (*bandwidth*)
- ◆ **Latence** (*latency*) : temps écoulé entre l'émission d'un bit et sa réception. Unité : s (ms,  $\mu$ s, ...)
  - ❖ on s'intéresse aussi parfois au temps d'aller-retour (*round trip time* ou RTT)

## ■ Facteurs de performance

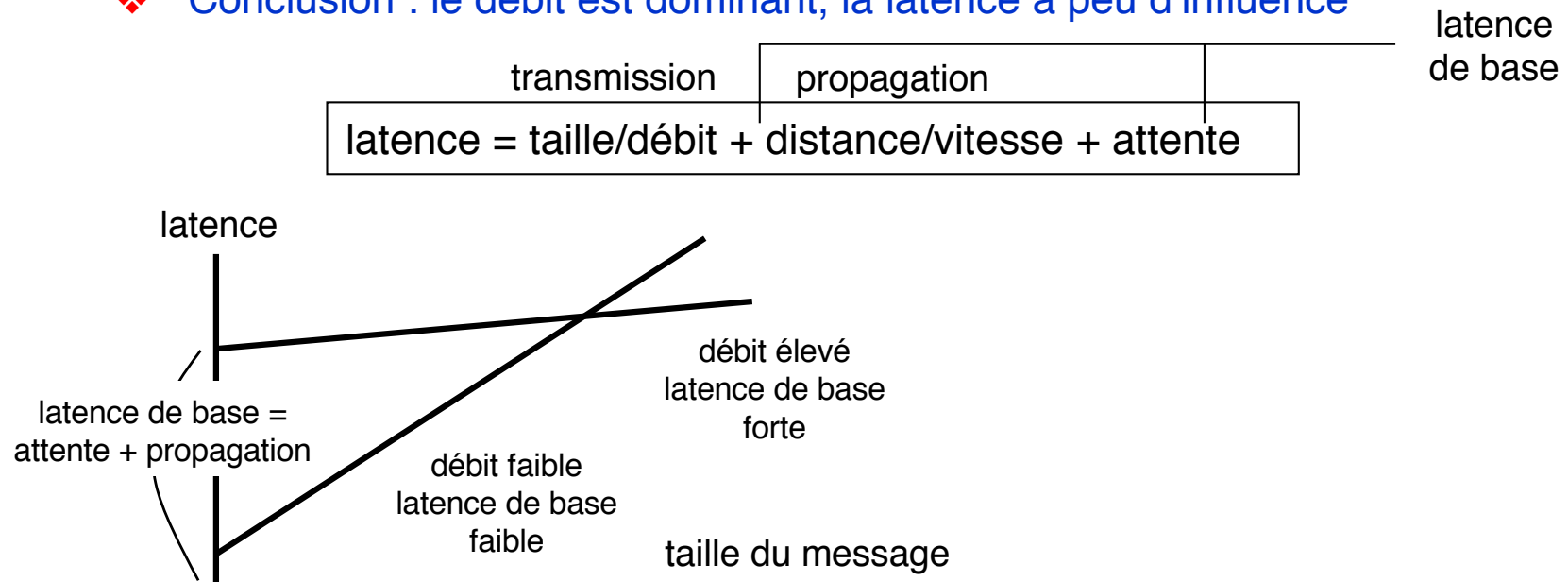
Latence = durée de transmission + temps de propagation + temps d'attente

- ◆ Durée de transmission : taille du message / débit
  - ❖ temps nécessaire pour transmettre les données (les envoyer sur le réseau)
- ◆ Temps de propagation : distance / vitesse de propagation
  - ❖ temps nécessaire pour que les données aillent de l'émetteur au récepteur
- ◆ Temps d'attente
  - ❖ temps "perdu" par le système de communication (notamment à cause de l'occupation des ressources)
- ◆ La somme (temps de propagation + temps d'attente), ou **latence de base**, est un délai incompressible (temps écoulé avant de recevoir le 1er bit d'un message)

# Performances des réseaux (2)

## ■ Débit et latence ont un impact variable selon les applications

- ◆ Exemple 1 : message bref (exemple : interaction question-réponse)
  - ❖ Envoi 1 Kbit, retour 1 Kbit, distance 10 000 km. Temps incompressible d'aller-retour = 100 ms (si vitesse =  $2c/3$ ). Temps de transmission = 1 ms à 1 Mbit/s, 0,01 ms à 100 Mbit/s.
  - ❖ Conclusion : la latence est dominante, le débit a peu d'influence
- ◆ Exemple 2 : message très gros (exemple : transfert d'un fichier multimédia)
  - ❖ Envoi 100 Mbit, distance 10 000 km. Temps incompressible de propagation = 50 ms. Temps de transmission = 100 s à 1 Mbit/s, 1 s à 100 Mbit/s.
  - ❖ Conclusion : le débit est dominant, la latence a peu d'influence



# Qualité de service

---

- La qualité de service (*Quality of Service*, QoS) désigne un ensemble de facteurs de qualité nécessaires aux besoins d'une application particulière
  - ◆ Cette définition est générique et doit être précisée dans chaque cas
  - ◆ Les besoins en QoS dépendent de la nature des applications
- Exemples
  - ◆ Stabilité de la latence (absence de gigue)
    - ❖ nécessaire pour les applications multimédia (son et vidéo)
  - ◆ Garantie d'une limite supérieure sur la latence
    - ❖ nécessaire pour les applications critiques liées au temps réel
  - ◆ Faible taux d'erreur (probabilité pour qu'un bit ou un message soit perdu ou modifié)
    - ❖ nécessaire si les données sont peu redondantes
- Garanties de qualité de service
  - ◆ Problème difficile (au moins pour les aspects quantitatifs)
  - ◆ Voies d'approche : réservation de ressources

# Qualité de service

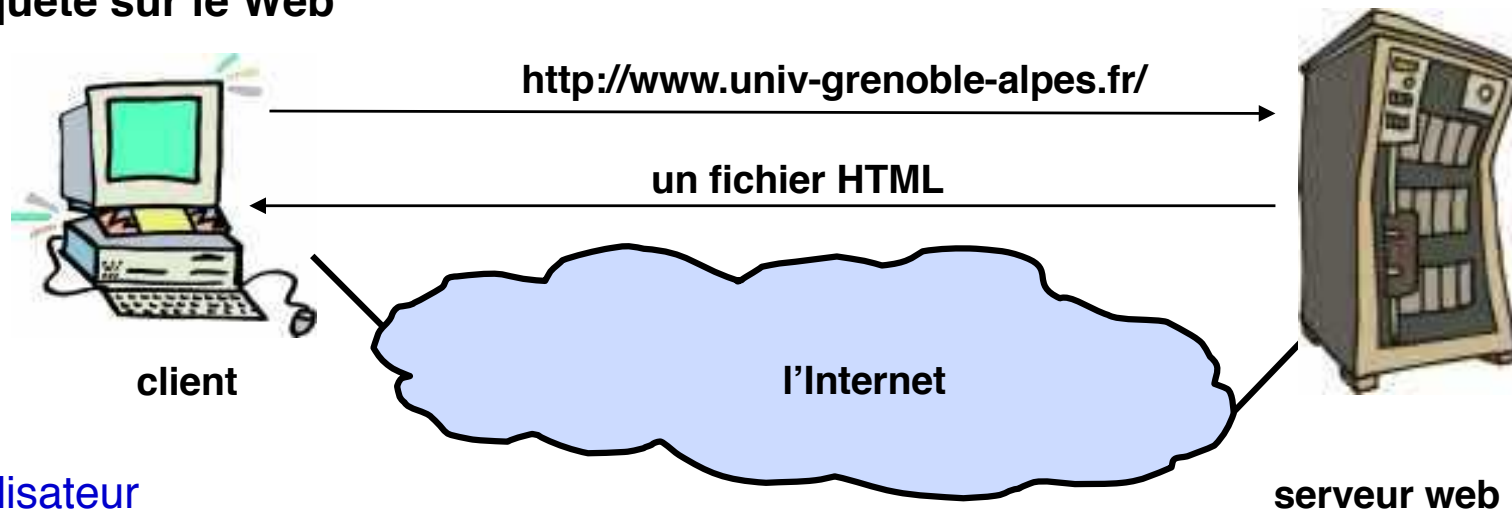
## Exemples de besoins applicatifs

Application	Perte de données tolérable ?	Débit nécessaire	Sensibilité à la latence ?
Transfert de fichiers	Non	Élastique	Non
Email	Non	Élastique	Non
Documents web	Non	Élastique (quelques Kb/s)	Non
Téléphonie / vidéoconférence	Oui	Audio : $\leq 1$ Mb/s Vidéo : 1 à kb/s à 5 Mb/s	Oui : $\leq 1$ seconde
Flux multimédia	Oui	Cf. ligne précédente	Oui : secondes
Jeu interactif	Oui	1 à 10 kb/s	Oui : $\leq 1$ seconde
Messagerie instantanée	Non	Élastique	Oui et non

Tableau emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

# Comment fonctionne un réseau ? (1)

## une requête sur le Web



### Vu de l'utilisateur

on clique sur un lien  
une "page web" s'affiche sur l'écran

### Sur la station client

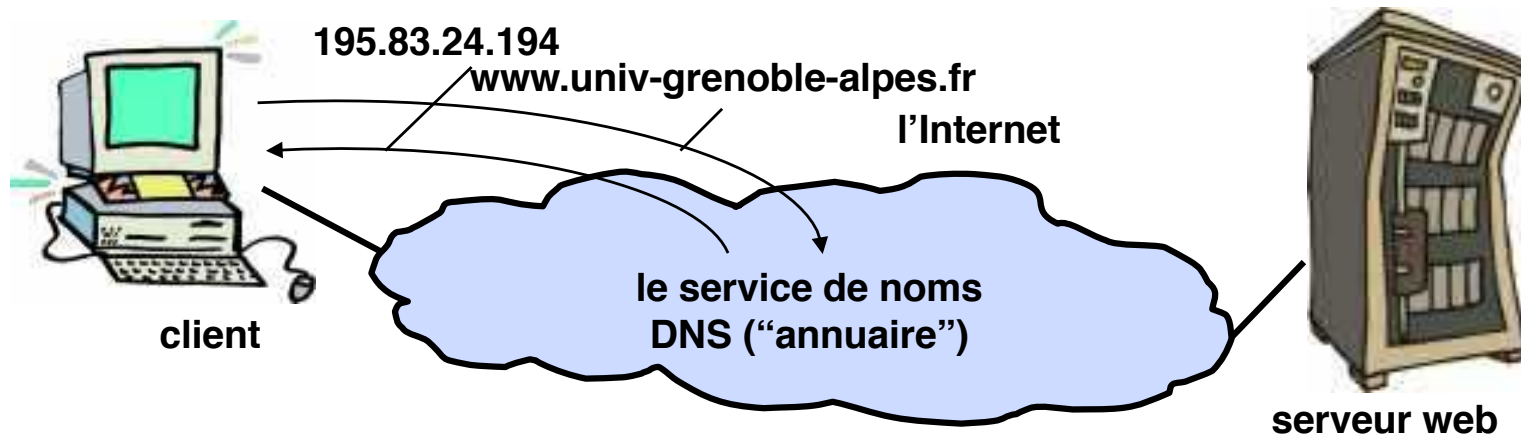
le programme navigateur envoie une requête au serveur correspondant à l'URI  
(*Uniform Resource Identifier*) associé au lien  
(si tout se passe bien) le programme navigateur reçoit un fichier HTML qu'il sait afficher sur l'écran

### Sur le réseau, il faut :

trouver le bon serveur (celui qui correspond à l'URI)  
transporter la requête depuis la station client vers le serveur  
transporter le fichier depuis le serveur à la station client

(voir aussi « what happens when ... »: <https://github.com/alex/what-happens-when/blob/master/README.rst>)

## Comment fonctionne un réseau ? (2)



### Première étape : localiser le serveur

Via le service de noms de l'Internet (DNS) qui associe un nom à une "adresse IP"  
(fonctionne comme un annuaire - détails plus loin)

Toute machine connectée à l'Internet a une **adresse IP**

Question : comment trouver l'annuaire ?

Réponse : on connaît l'adresse IP d'un point d'entrée

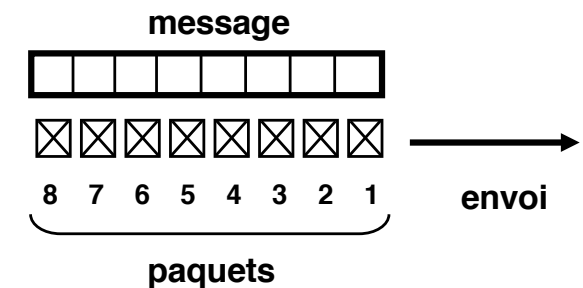
### Deuxième étape : envoyer la requête au serveur

Comment est transmise la requête ?

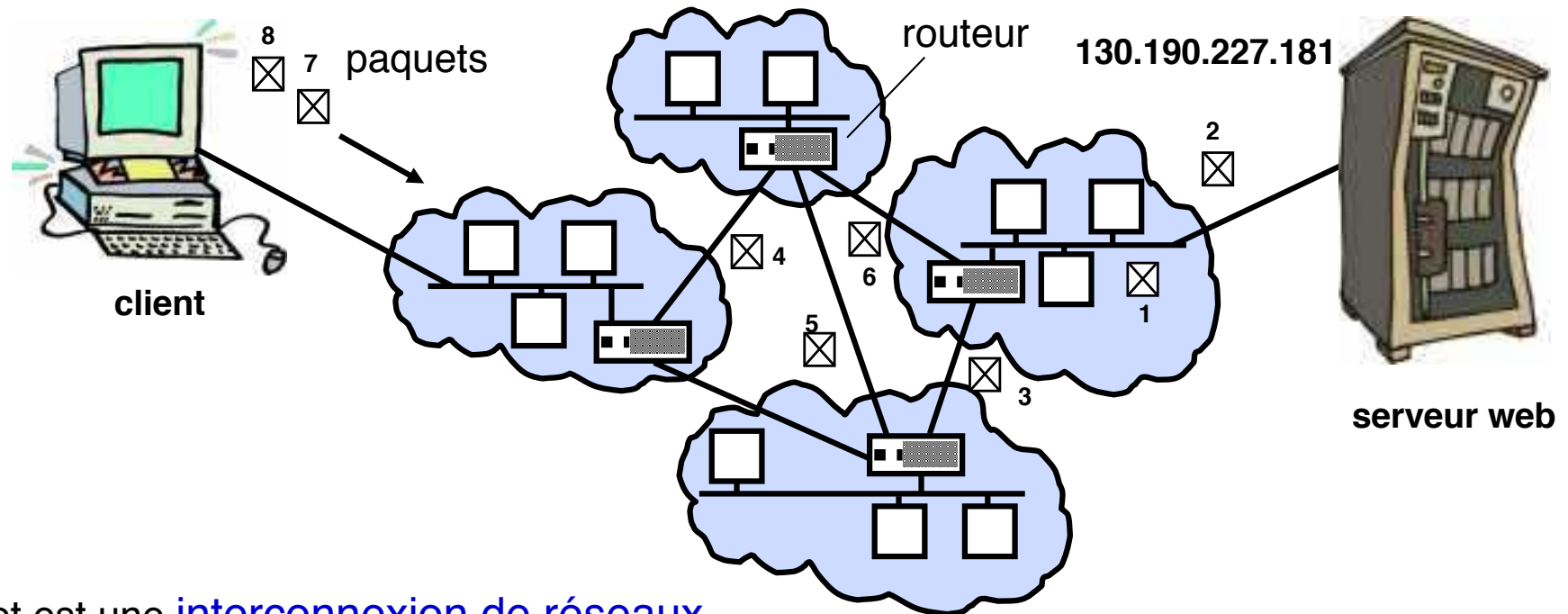
La requête est un message (une suite de bits).

Elle est découpée en "paquets" de taille fixe (bornée)

Chaque paquet est envoyé sur le réseau



## Comment fonctionne un réseau ? (3)



L'Internet est une **interconnexion de réseaux** (*internetwork*)

Les réseaux sont reliés entre eux par des **routeurs**

Chaque paquet contient son numéro et son adresse de destination. Quand il arrive sur un réseau, et que le site destinataire n'en fait pas partie, le paquet est transmis à un routeur.

Les routeurs contiennent les informations qui permettent d'acheminer le paquet vers son site destinataire



# Comment fonctionne un réseau ? (4)

---

## Fonctionnement du serveur

Le serveur reconstruit le message initial en mettant les paquets dans l'ordre de leurs numéros. Que fait le serveur à la réception de la requête ?

Le site serveur interprète la requête comme une demande de fourniture de fichier. Il envoie le fichier au client (sous forme d'une suite de paquets, comme précédemment).

Que se passe-t-il si un paquet s'est perdu, ou a mal été transmis ?

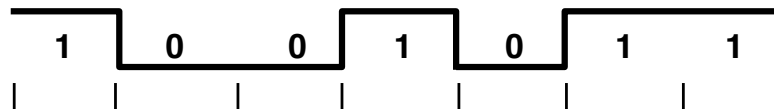
Le destinataire est capable de le détecter et demande qu'on lui renvoie le paquet manquant ou erroné.

## Transmission de l'information

Comment sont transmis concrètement les paquets ?

Chaque paquet est une suite de bits. La transmission utilise un support physique : fils, fibre optique, ondes radio, etc., sur lequel sont envoyés des signaux (électriques, lumineux, etc.). Chaque bit (0 ou 1) est représenté par une configuration particulière du signal

Exemple  
(non réaliste)



# Connaître le chemin suivi (1)

---

La commande `tracert` permet de montrer la suite de routeurs utilisés

Exemples :

```
mandelbrot:~> tracert jpp
tracert to jpp.e.ujf-grenoble.fr (195.220.82.130), 30 hops max, 40 byte packets
 1 jpp (195.220.82.130) 6.044 ms 0.138 ms 0.139 ms
```

même réseau  
pas de routeur

```
mandelbrot:~> tracert nougat.e.ujf-grenoble.fr
tracert to nougat.e.ujf-grenoble.fr (152.77.2.163), 30 hops max, 60 byte packets
 1 iron-ima-2-ima-srv-ped.ujf-grenoble.fr (195.220.82.129) 0.387 ms 0.680 ms 0.986 ms
 2 dsigu-giga-2-ima-gate.ujf-grenoble.fr (152.77.32.50) 2.166 ms 2.729 ms 3.067 ms
 3 x480-simsu-to-dsigu-giga.ujf-grenoble.fr (152.77.32.57) 3.405 ms 3.793 ms 4.142 ms
 4 x480-admin-to-x480-simsu.ujf-grenoble.fr (152.77.32.114) 4.480 ms 4.820 ms 5.124 ms
 5 iron-sciences-2-admin-giga.ujf-grenoble.fr (152.77.32.218) 5.276 ms 5.426 ms 5.719 ms
 6 nougat.e.ujf-grenoble.fr (152.77.2.163) 5.863 ms 1.191 ms 1.248 ms
```

## Connaître le chemin suivi (2)

```
mandelbrot:~> traceroute www.umass.edu
traceroute to www.umass.edu (128.119.103.13), 30 hops max, 60 byte packets
 1 iron-ima-2-ima-srv-ped.ujf-grenoble.fr (195.220.82.129)  0.408 ms  0.703 ms  1.012 ms
 2 dsigu-giga-2-ima-gate.ujf-grenoble.fr (152.77.32.50)  2.221 ms  2.554 ms  3.339 ms
 3 x480-simsu-to-dsigu-giga.ujf-grenoble.fr (152.77.32.57)  3.674 ms  4.012 ms  4.347 ms
 4 x480-admin-to-x480-simsu.ujf-grenoble.fr (152.77.32.114)  4.690 ms  4.997 ms  5.146 ms
 5 x480-bio-a-4-2-x480-admin.ujf-grenoble.fr (152.77.32.6)  5.453 ms  5.805 ms  6.110 ms
 6 smh-gate-2-x480-bio-a-4.ujf-grenoble.fr (152.77.32.46)  6.280 ms  1.231 ms  0.835 ms
 7 r-campus.grenet.fr (193.54.185.120)  1.165 ms  1.505 ms  1.846 ms
 8 tigre1.grenet.fr (193.54.184.33)  3.014 ms  3.370 ms  3.545 ms
 9 * * *
10 te1-3-lyon2-rtr-021.noc.renater.fr (193.51.189.110)  13.421 ms  9.916 ms  10.071 ms
11 te0-0-0-1-paris2-rtr-001.noc.renater.fr (193.51.189.2)  10.833 ms  11.036 ms  11.381 ms
12 te0-3-4-0-paris1-rtr-001.noc.renater.fr (193.51.189.5)  14.047 ms  26.277 ms  26.433 ms
13 renater.rt1.par.fr.geant2.net (62.40.124.69)  10.650 ms  10.807 ms  11.140 ms
14 as0.rt1.lon.uk.geant2.net (62.40.112.106)  18.449 ms  20.528 ms  19.840 ms
15 as1.rt1.ams.nl.geant2.net (62.40.112.137)  25.605 ms  25.478 ms  25.736 ms
16 xe-2-3-0.102.rtr.newy32aoa.net.internet2.edu (198.32.11.50)  112.645 ms  112.774 ms  112.772 ms
17 nox300gw1-vl-110-nox-i2.nox.org (192.5.89.221)  117.452 ms  117.309 ms  117.338 ms
18 nox300gw1-peer-nox-umass-192-5-89-102.nox.org (192.5.89.102)  120.028 ms  120.032 ms  120.033 ms
19 libr-rt-b02b-1-gi4-5.gw.umass.edu (128.119.2.202)  120.014 ms  119.648 ms  119.588 ms
20 lgrc-rt-106-6-gi1-2.gw.umass.edu (128.119.2.182)  119.722 ms  119.725 ms  119.621 ms
21 www.umass.edu (128.119.103.13)  119.956 ms  119.563 ms  119.566 ms
```

## Connaître le chemin suivi (3)

Variante de traceroute : lft (layer-four traceroute)

- Optimisations: utilisations de diverses techniques pour améliorer la découverte du chemin (vitesse, précision, robustesse)
- Fonctionnalités supplémentaires (par exemple, affichage d'informations sur les réseaux traversés)

```
bash-4.2$ lft -N www.cornell.edu
Tracing .....*..*T
TTL LFT trace to wwwcornelledu-ssl.cit.cornell.edu (132.236.204.10):80/tcp
 1 [FR-MI2S] fw-campus-r.imag.fr (129.88.1.198) 0.6ms
 2 [FR-SIMSU] r-campus1.grenet.fr (193.54.184.21) 1.2ms
 3 [FR-SIMSU] tigre1.grenet.fr (193.54.184.33) 1.5ms
** [neglected] no reply packets received from TTLs 4 through 5
 6 [GEANT] renater.mx1.gen.ch.geant.net (62.40.124.61) 3.0ms
 7 [GEANT] ae4.mx1.par.fr.geant.net (62.40.98.152) 10.2ms
 8 [GEANT] ae1.mx1.lon.uk.geant.net (62.40.98.76) 16.0ms
 9 [GEANT] ae0.mx1.ams.nl.geant.net (62.40.98.81) 20.8ms
10 [GEANT] ae2.rt1.ams.nl.geant.net (62.40.98.114) 20.8ms
11 [NLR] tge-0-5-0-4.211.newy.layer3.nlr.net (216.24.184.85) 107.7ms
12 [CORNELL-DMZ] nyc1-6500-vl4040.net.cornell.edu (192.35.82.130) 107.7ms
13 [CORNELLU-NET] core2-6500-vl12.net.cornell.edu (132.236.222.13) 144.9ms
14 [CORNELLU-NET] sf1-6500-te5-4.net.cornell.edu (132.236.222.174) 128.0ms
15 [CORNELLU-NET] [target open] wwwcornelledu-ssl.cit.cornell.edu (132.236.204.10):80 115.3ms
```

## Connaître le chemin suivi (4)

---

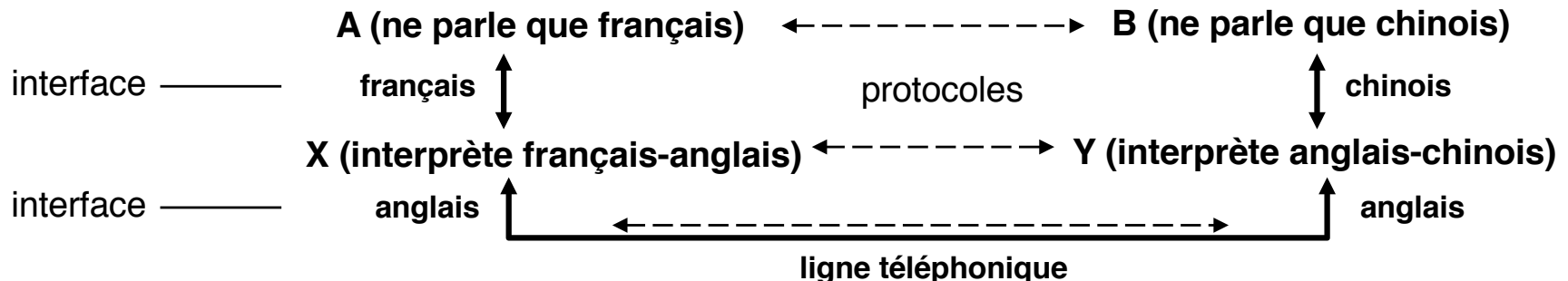
Même requête que la précédente, à partir de la même ville (Grenoble) mais depuis un autre réseau (fournisseur d'accès Internet à domicile à la place du réseau de l'Université)

```
bash-4.2$ lft -N www.cornell.edu
Tracing .....*..*.*.T
TTL LFT trace to wwwcornelledu-ssl.cit.cornell.edu (132.236.204.10):80/tcp
 1 [NULL] 192.168.0.254 1.8ms
 2 [FR-PROXAD-ADSL] 82.228.182.254 19.8ms
 3 [FR-PROXAD] grenoble-6k-1-a5.routers.proxad.net (213.228.10.62) 22.5ms
 4 [FR-PROXAD] lyon-crs16-1-be1002.intf.routers.proxad.net (212.27.59.45) 23.7ms
 5 [FR-PROXAD] lyon-crs8-1-be1000.intf.routers.proxad.net (212.27.56.157) 31.4ms
 6 [FR-PROXAD] th2-crs16-1-be2001.intf.routers.proxad.net (212.27.59.29) 41.8ms
 7 [FR-PROXAD] bzn-crs16-2-be2000.intf.routers.proxad.net (78.254.250.125) 29.4ms
 8 [COGENT-149-6-16] te0-0-0-31.363.mag21.par01.atlas.cogentco.com (149.6.160.97) 69.5ms
 9 [NET-154-54-0-0] be2040.mpd21.par01.atlas.cogentco.com (154.54.78.37) 69.3ms
10 [COGENT-EUROPEAN-OPERATIONS-001] be2273.mpd21.lon13.atlas.cogentco.com (130.117.50.93) 84.1ms
11 [NET-154-54-0-0] be2390.ccr21.bos01.atlas.cogentco.com (154.54.44.221) 151.0ms
** [neglected] no reply packets received from TTLs 12 through 13
14 [COGENT-A] 38.122.120.22 160.4ms
15 [CORNELLU-NET] core2-6500-te4-7.net.cornell.edu (132.236.222.9) 160.2ms
16 [CORNELLU-NET] sf1-6500-te5-4.net.cornell.edu (132.236.222.174) 158.7ms
17 [CORNELLU-NET] [target open] wwwcornelledu-ssl.cit.cornell.edu (132.236.204.10):80 157.0ms
```

# Notions de protocole et d'interface (1)

- L'exemple de la requête sur le web permet d'identifier divers "niveaux" d'échange entre le client et le serveur
  - ◆ le niveau de l'application : le client clique sur un lien, le serveur renvoie une page web
  - ◆ le niveau des messages : le client envoie un message contenant une URI, le serveur renvoie un message contenant un fichier HTML
  - ◆ le niveau des paquets : le message du client est découpé en paquets, les différents routeurs du réseau les acheminent vers le serveur (idem pour le retour)
  - ◆ le niveau de la transmission des bits : pour envoyer les paquets, chaque bit (0 ou 1) est transmis comme un signal électrique sur une ligne.
  - ◆ chaque niveau utilise les fonctions du niveau inférieur
- Les notions de protocole et d'interface visent à représenter ce mode de fonctionnement

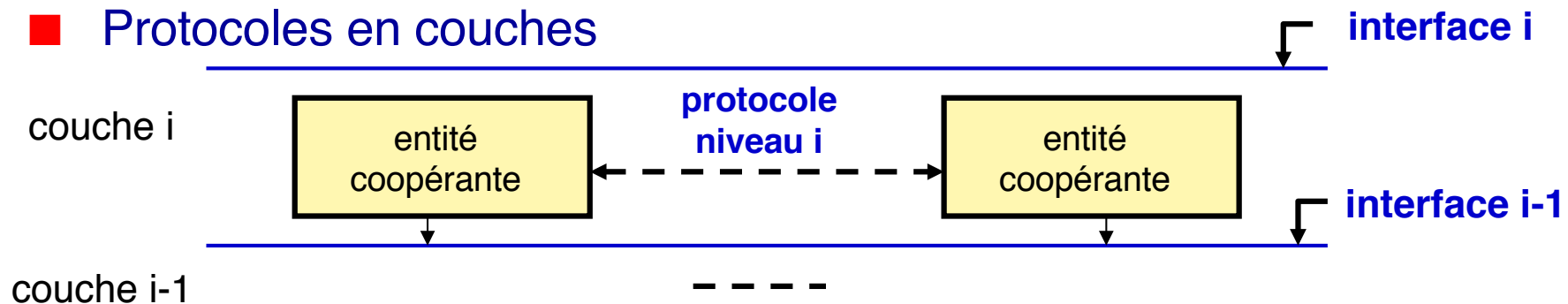
**Analogie : deux personnes peuvent dialoguer même si elles ne parlent pas la même langue**



## Notions de protocole et d'interface (2)

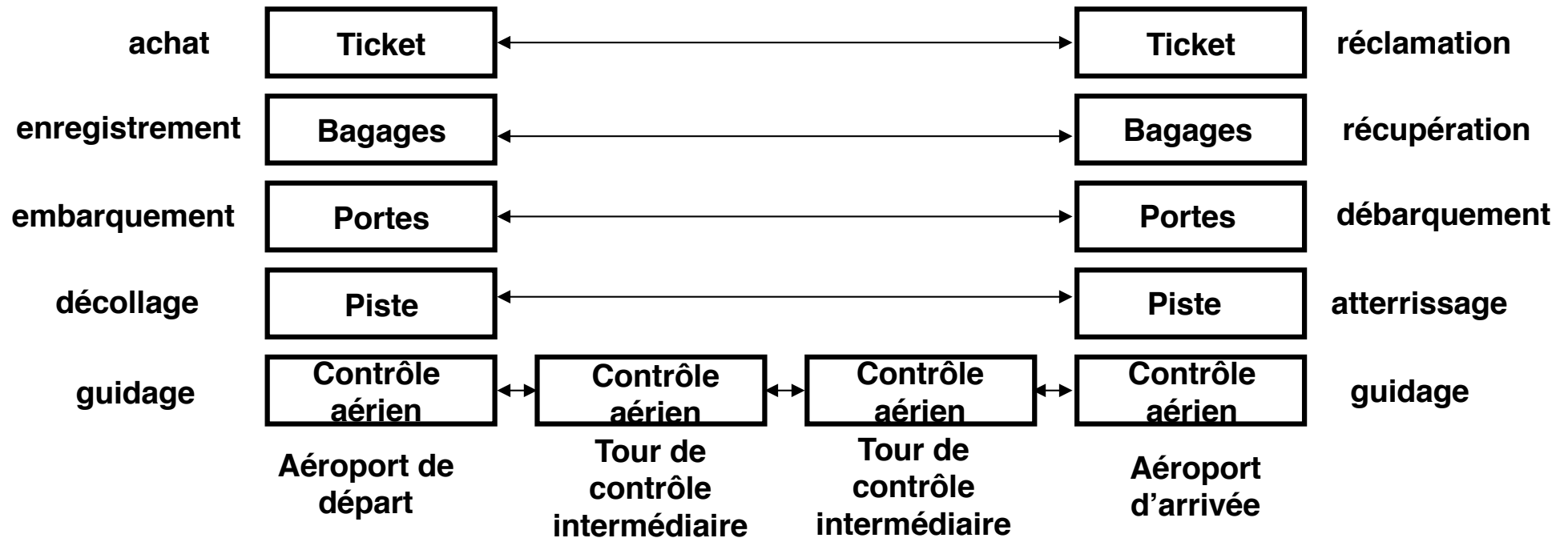
- Interface (d'un service) : ensemble de fonctions (logicielles ou matérielles) et de règles d'accès utilisables pour accéder au service
- Protocole : ensemble de conventions définissant les échanges entre des entités qui coopèrent pour réaliser un service
  - ◆ Format et séquence de messages échangés entre entités
  - ◆ Actions à effectuer suite à l'arrivée d'un message ou à l'occurrence d'un autre événement ...
- Relations entre protocoles et interfaces
  - ◆ une interface définit l'accès à un service, un protocole définit la réalisation d'un service
  - ◆ la construction d'un protocole utilise souvent des protocoles de niveau inférieur (plus élémentaires), en accédant à leurs interfaces

### ■ Protocoles en couches



# Organisation en couches (1/2)

## ■ Analogie avec le transport aérien



Exemple emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

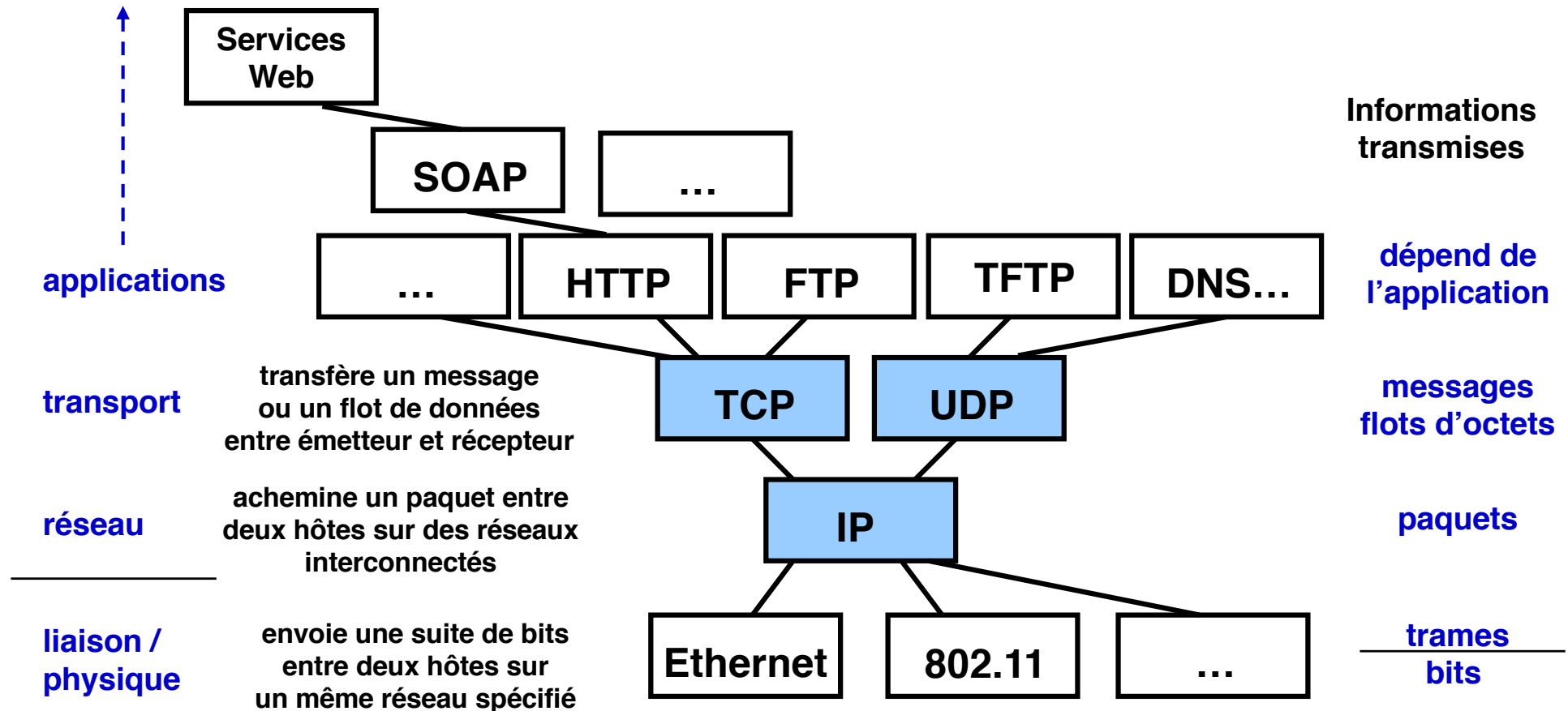


## Organisation en couches (2/2)

---

- **Approche nécessaire pour maîtriser la complexité**
- **Utilisée dans tous les systèmes informatiques et a fortiori dans les systèmes répartis**
- **Structuration explicite**
  - ◆ **Permet d'identifier les interactions entre les différents composants**
- **Modularité**
  - ◆ **Indépendance entre l'interface offerte par une couche et sa réalisation interne**
  - ◆ **Simplifie la maintenance et l'évolution du système**

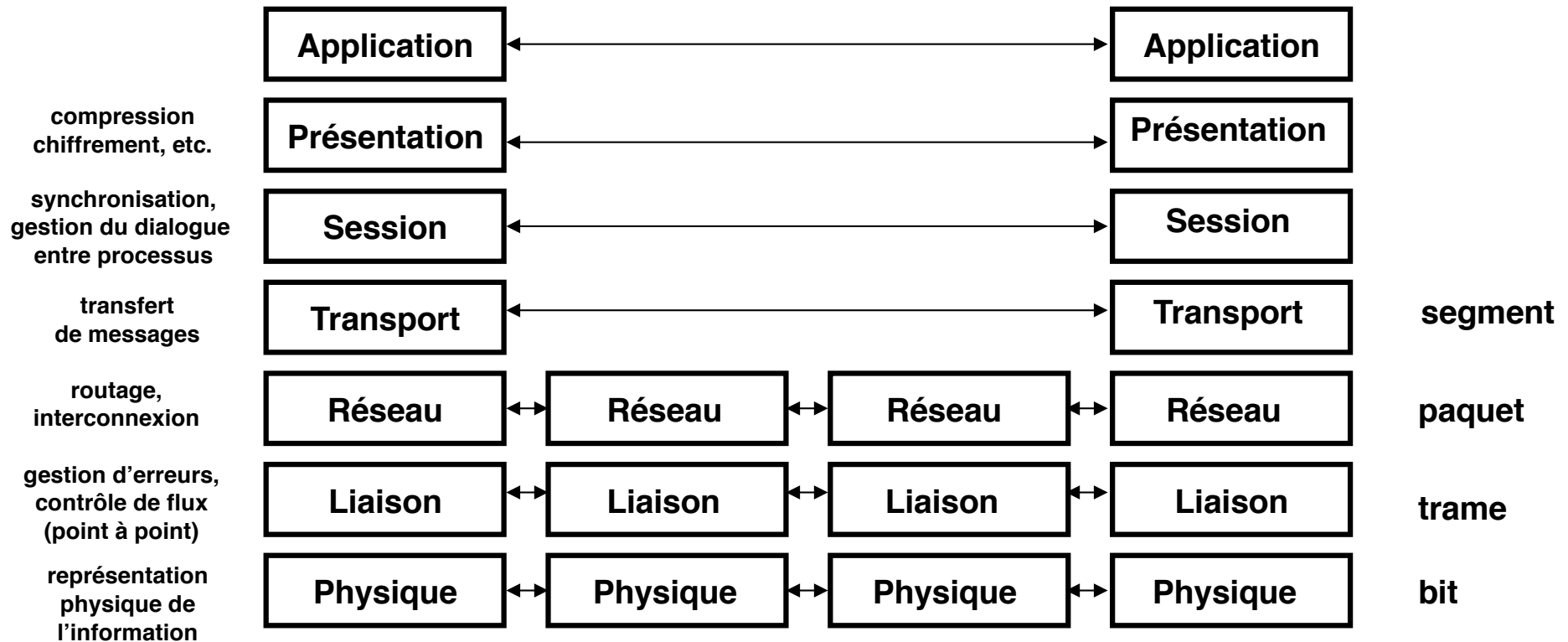
# Les protocoles de l'Internet



**HTTP** : *HyperText Transfer Protocol* : protocole du Web  
**TFTP, FTP** : *(Trivial) File Transfer Protocol* : transfert de fichiers  
**TCP** : *Transmission Control Protocol* : transport en mode connecté  
**UDP** : *User Datagram Protocol* : transport en mode non connecté  
**IP** : *Internet Protocol* : Interconnexion de réseaux, routage

# Les protocoles normalisés de l'ISO (International Standards Organisation)

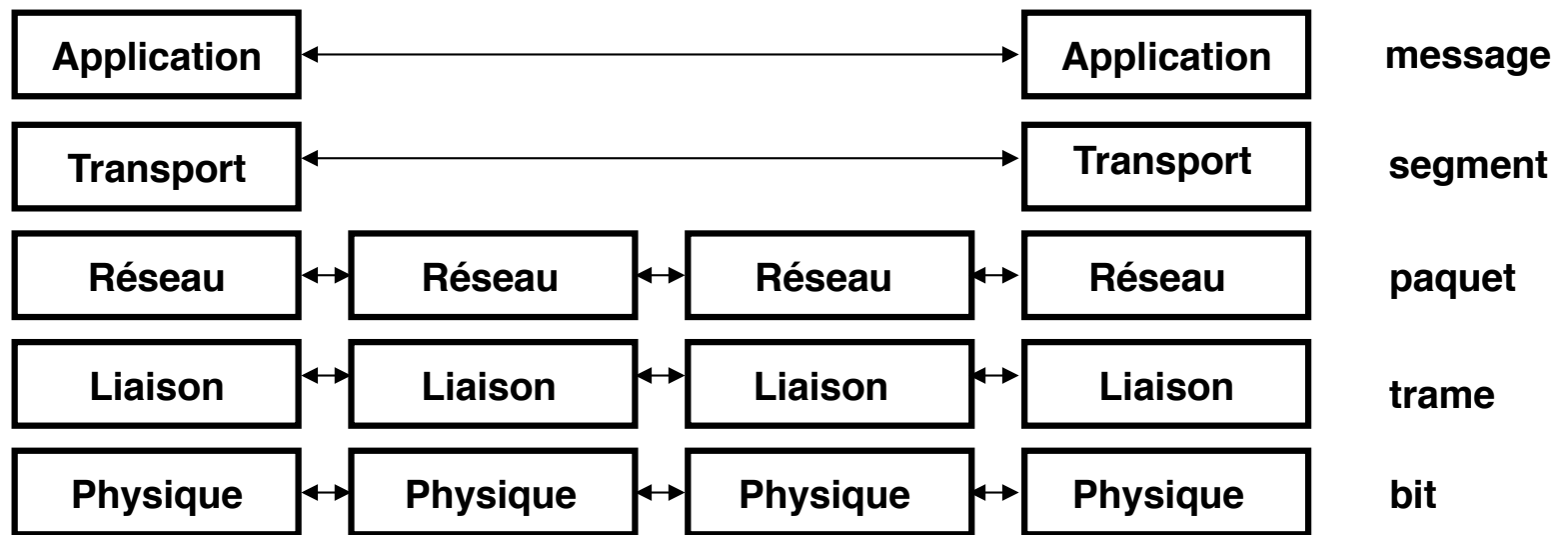
## Open Systems Interconnection (OSI)



Les protocoles OSI servent plutôt de cadre de référence pour la définition des fonctions que de normes de réalisation. La normalisation de fait est autour de TCP/IP

# Le modèle en couches de l'Internet

---



Les services des couches Session et Présentation définies par le modèle OSI sont ici intégrés dans la couche applicative (si nécessaire)

# Encapsulation

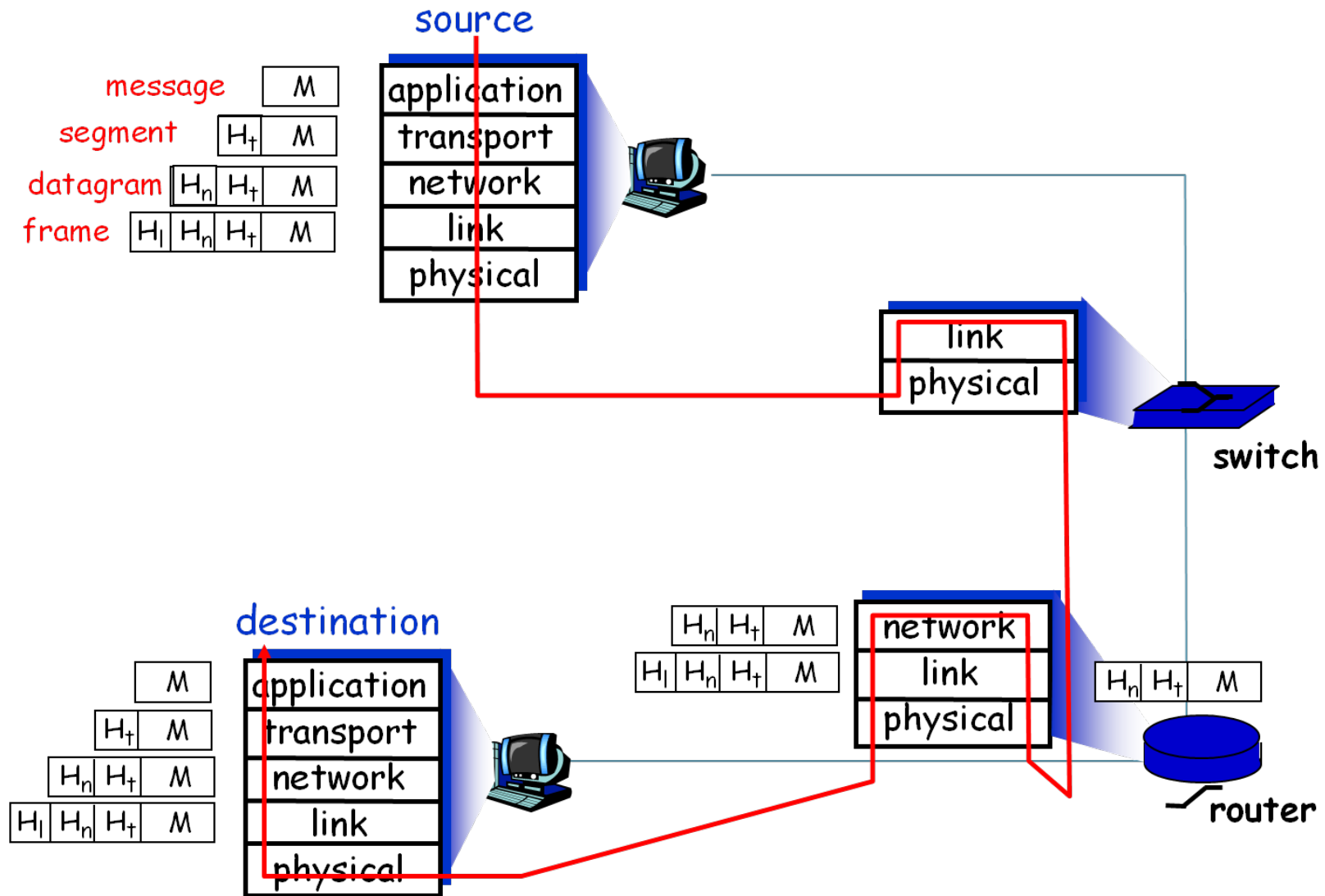


Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

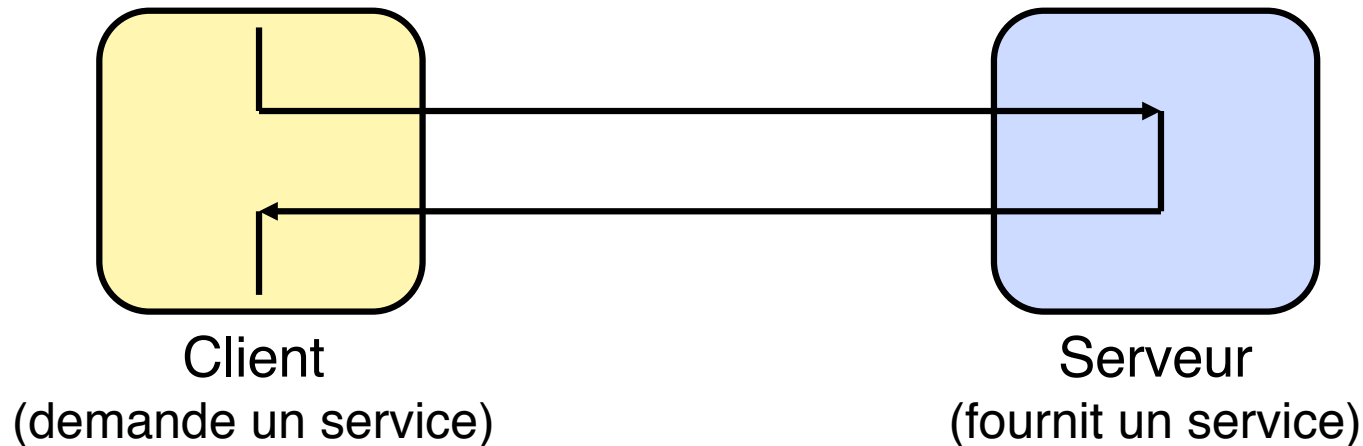
# Les services de l'Internet

---

- Services principaux (applications) disponibles sur l'Internet
  - ◆ Courrier électronique (`mail`) - protocole SMTP (*Simple Mail Transfer Protocol*)
    - ❖ inclut maintenant tous types de documents, grâce au format MIME (*Multipurpose Internet Mail Extensions*)
  - ◆ Forums de discussion (`news`) - protocole NNTP (*Network News Transfer Protocol*)
  - ◆ Transfert de fichiers (`ftp`) - protocole FTP (*File Transfer Protocol*)
  - ◆ Accès à une machine distante - protocoles *telnet*, SSH
  - ◆ X-Window, service de fenêtres pour client distant
  - ◆ Accès au World Wide Web - protocole HTTP, formats HTML, XML, ...
  - ◆ Services divers utilisant le Web (bibliothèques virtuelles, commerce électronique, installation de logiciel, ...)
  - ◆ Extensions diverses pour la sécurité (SSL, SET, PGP, ...)
- Ces services utilisent les protocoles de transport (TCP, UDP), ainsi que le service de noms DNS (*Domain Name System*)

# Le réseau vu de l'utilisateur (1)

---

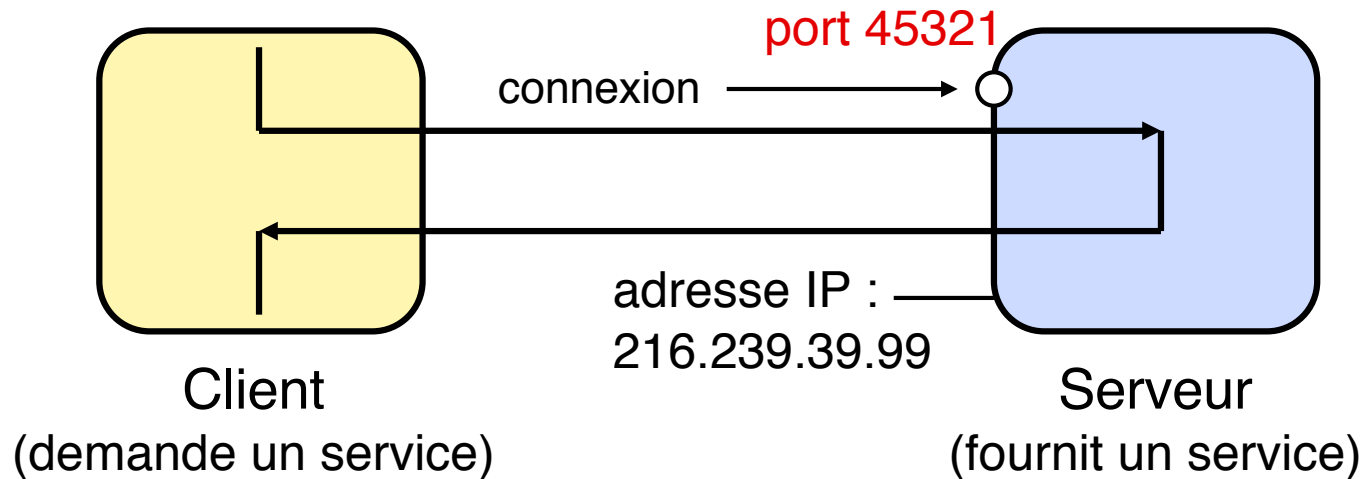


Le schéma **client-serveur** peut être mis en œuvre pour des processus sur une même machine. Ce schéma se transpose à un réseau, où les processus client et serveur sont sur des machines différentes.

Pour le client, un service est souvent désigné par un nom symbolique. Ce nom doit être converti en une adresse interprétable par les protocoles du réseau.

La conversion d'un nom symbolique (par ex. <http://www.google.fr>) en une adresse IP (216.239.39.99) est à la charge du service DNS (vu plus loin)

## Le réseau vu de l'utilisateur (2)



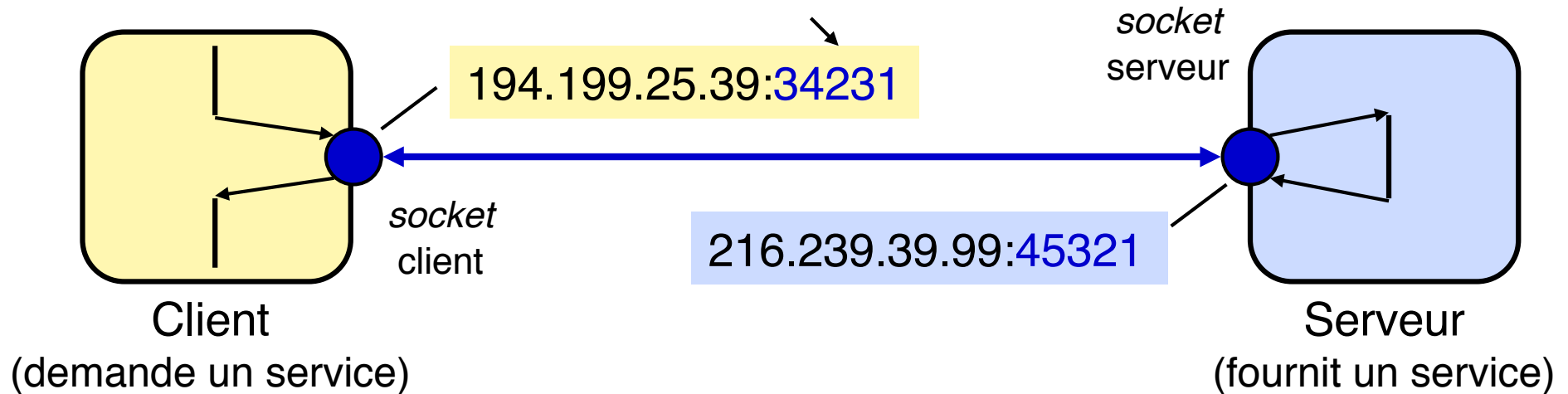
En fait, l'adresse IP du serveur ne suffit pas, car le serveur (machine physique) peut comporter différents services; il faut préciser le service demandé au moyen d'un **numéro de port**, qui permet d'atteindre un processus particulier sur la machine serveur.

Un numéro de port comprend 16 bits (0 à 65 535). Les numéros de 0 à 1023 sont réservés, par convention, à des services spécifiques. Exemples (protocole TCP) :

- |                                    |                                |
|------------------------------------|--------------------------------|
| 7 : echo                           | 22 : ssh                       |
| 23 : telnet (connexion à distance) | 80 : serveur web               |
| 25 : mail (acheminement)           | 110 : mail (consultation POP3) |



## Le réseau vu de l'utilisateur (3)



Pour programmer une application client-serveur, il est commode d'utiliser les *sockets*, qui fournissent une interface qui permet d'utiliser facilement les protocoles de transport TCP et UDP

Un *socket* est simplement un moyen de désigner l'extrémité d'une connexion, côté émetteur ou récepteur, en l'associant à un port.

Une fois la connexion (bidirectionnelle) établie via des *sockets* entre un processus client et un processus serveur, ceux-ci peuvent communiquer en utilisant les mêmes primitives (*read*, *write*) que pour l'accès aux fichiers.

# DNS : noms de domaines (1)

---

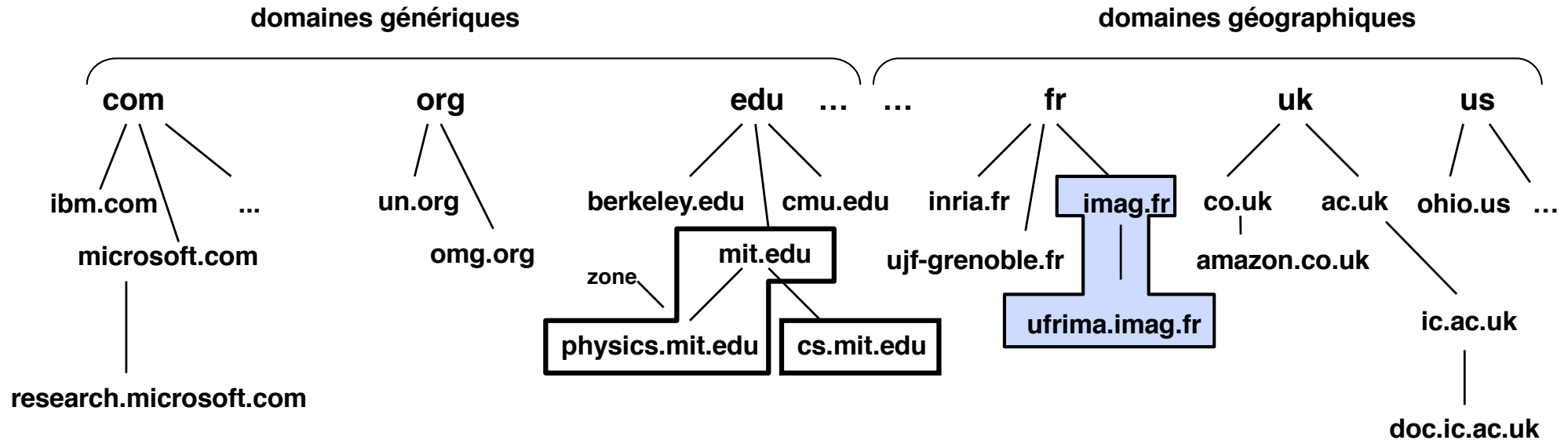
- La désignation sur l'Internet utilise un schéma hiérarchique (noms de domaines)
  - ◆ Exemples : thales.e.ujf-grenoble.fr, cmu1.acs.cmu.edu, www.w3.org, firewall.ma02.bull.com
  - ◆ Un **domaine** est une unité pour la construction et la gestion des noms (analogue à un répertoire dans un système de fichiers)
  
- La hiérarchie des noms a une racine multiple : les domaines du premier niveau
  - ◆ domaines “géographiques” :
    - ❖ fr (France), uk (Royaume Uni), de (Allemagne), jp (Japon),
    - ❖ USA est souvent compris “par défaut”, bien qu’il existe un domaine us
  - ◆ domaines “d’activité”, génériques :
    - ❖ com (commercial), org (organisations),  
edu (universités, aux USA), net (l’Internet), ...

## DNS : noms de domaines (2)

---

- Dans chaque domaine, les noms sont attribués par une autorité responsable du domaine
  - ◆ Pour établir la liste des domaines du premier niveau : l'Internet Society (ISOC) via un groupe technique ad hoc
  - ◆ Pour les domaines "publics" du premier niveau (com, org, net, ...) : une autorité centrale, l'ICANN ([www.icann.org](http://www.icann.org)), avec des autorités déléguées
  - ◆ Pour les domaines géographiques : une autorité nationale par pays - en France, l'AFNIC ([www.nic.fr](http://www.nic.fr)) - Association Française pour le Nommage sur l'Internet en Coopération)
  - ◆ Pour les domaines inclus : autorités locales (entreprise, administration, etc.)

# DNS : noms de domaines (3)



## ■ Domaines et zones

- ◆ Domaine = unité de désignation (espace de noms)
- ◆ Zone = unité de gestion administrative (serveur de noms propre à la zone)
- ◆ Le plus souvent, un domaine est aussi une zone, mais une zone peut grouper plusieurs domaines, administrés en commun

## DNS : noms de domaines (4)

---

- **Différents cas de figure sont possibles pour la correspondance entre nom et adresse**
  - ◆ **Cas simple : correspondance 1-1**
    - ❖ **kittyhawk.cmcl.cs.cmu.edu correspond à 128.2.194.242**
  - ◆ **Plusieurs noms vers la même adresse**
    - ❖ **eecs.mit.edu et cs.mit.edu correspondent tous les deux à 18.62.1.6**
  - ◆ **Plusieurs noms vers des adresses différentes**
    - ❖ **aol.com et www.aol.com correspondent à des adresses différentes**
  - ◆ **Un nom vers plusieurs adresses**
    - ❖ **voir plus loin**
  - ◆ **Certains noms valides ne correspondent pas à une adresse**
    - ❖ **cmcl.cs.cmu.edu n'est pas associé à une adresse mais c'est un suffixe valide pour d'autres noms, qui sont eux associés à une ou plusieurs adresses (exemple : kittyhawk.cmcl.cs.cmu)**

# DNS : services et principes

---

## ■ Services

- ◆ Correspondance Nom → adresse IP et adresse IP → Nom
- ◆ Gestion d'alias (alias → nom canonique)
- ◆ Distribution de charge
  - ❖ Association d'un nom canonique à plusieurs adresses
- ◆ Gestion d'alias pour les serveurs mail (sera étudié ultérieurement)

## ■ Principes

- ◆ Le DNS est un protocole applicatif qui fournit des services essentiels à d'autres protocoles applicatifs
- ◆ La gestion des noms est donc gérée aux « extrémités » du réseau, par un ensemble d'interactions entre clients et serveurs
- ◆ Cette organisation est conforme à la philosophie de conception de l'Internet : simplifier le cœur du réseau et gérer les aspects complexes aux extrémités

# DNS : mise en œuvre (1)

## Bref historique

---

### ■ Préhistorie (jusqu'à quelques milliers de sites)

- ◆ Un serveur de noms unique pour tout l'Internet, gérant une table de correspondance nom symbolique - adresse IP
- ◆ Inconvénients
  - ❖ Mauvaise résistance aux pannes
  - ❖ Goulot d'étranglement, pas de capacité de croissance
  - ❖ Distance, déséquilibre géographique
  - ❖ Maintenance

### ■ À partir de 1985 : Introduction de DNS

- ◆ Un service réparti
- ◆ ... qui a bien résisté depuis à une croissance exponentielle

# DNS : mise en œuvre (2)

## Une base de données hiérarchique et répartie

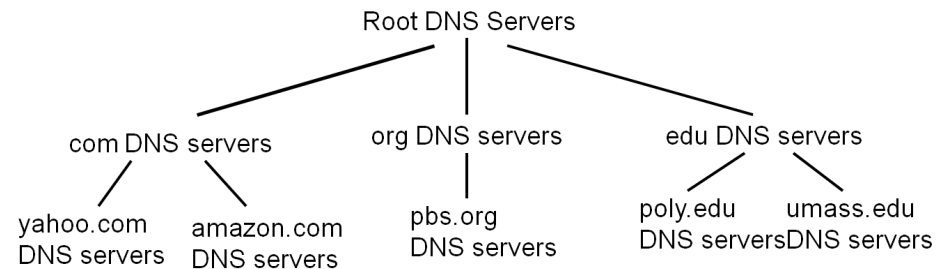
---

### ■ Idées générales

- ◆ Pas de point central : un ensemble de serveurs répartis sur le globe
- ◆ Un découpage des responsabilités

### ■ Au sein de la hiérarchie : différentes classes de serveurs

- ◆ Serveurs racine (*root DNS servers*)
- ◆ Serveurs de domaine de premier niveau (*Top-level domain* ou TLD)
- ◆ Serveurs d'autorité pour un domaine (*authoritative servers*)



### ■ En complément de la hiérarchie : serveurs locaux

Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach



# DNS : mise en œuvre (3)

## Serveurs racines

- 13 (grappes de) serveurs racines dans le monde

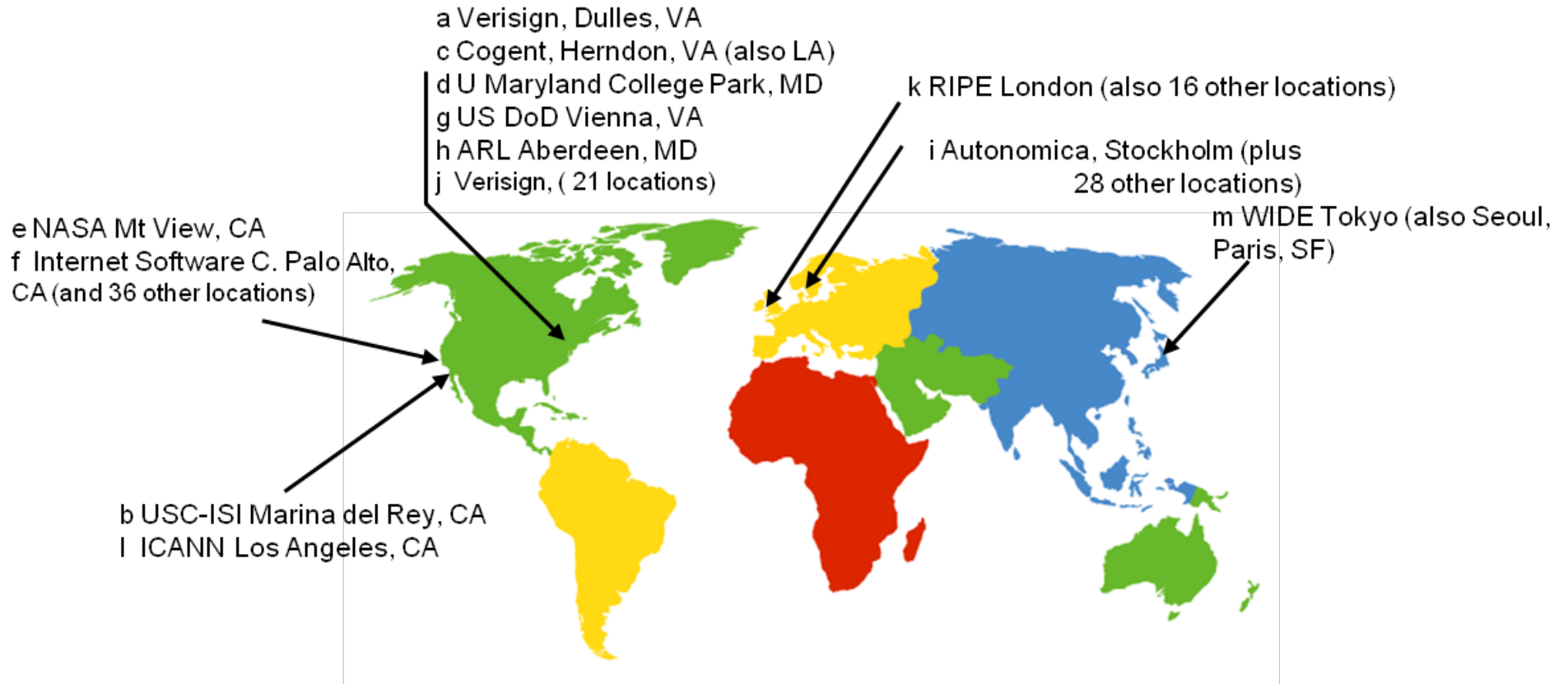


Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

# DNS : mise en œuvre (4)

## Autres classes de serveurs DNS

---

- **Serveurs DNS de premier niveau (TLD)**
  - ◆ Responsables d'un domaine de premier niveau générique (com, net, org ...) ou géographique (uk, de, fr ...)
  
- **Serveurs DNS d'autorité pour un domaine**
  - ◆ Gérés par l'organisation concernée (entreprise, université ...) ou par un prestataire externe
  
- **En dehors de la hiérarchie : serveurs DNS locaux**
  - ◆ Gérés par chaque organisation fournissant un accès à Internet
  - ◆ Une machine au sein d'un réseau envoie ses requêtes DNS au serveur local
  - ◆ Le serveur local joue un rôle d'intermédiaire auprès des serveurs de la hiérarchie

# DNS : Résolution de nom (1)

- L'hôte cis.poly.edu veut obtenir l'adresse IP associée à gaia.cs.umass.edu
- Procédure de requête itérative (étapes 2-7)
  - ◆ Si le serveur contacté n'a pas la réponse, il fournit l'identité d'un autre serveur permettant d'avancer dans la recherche

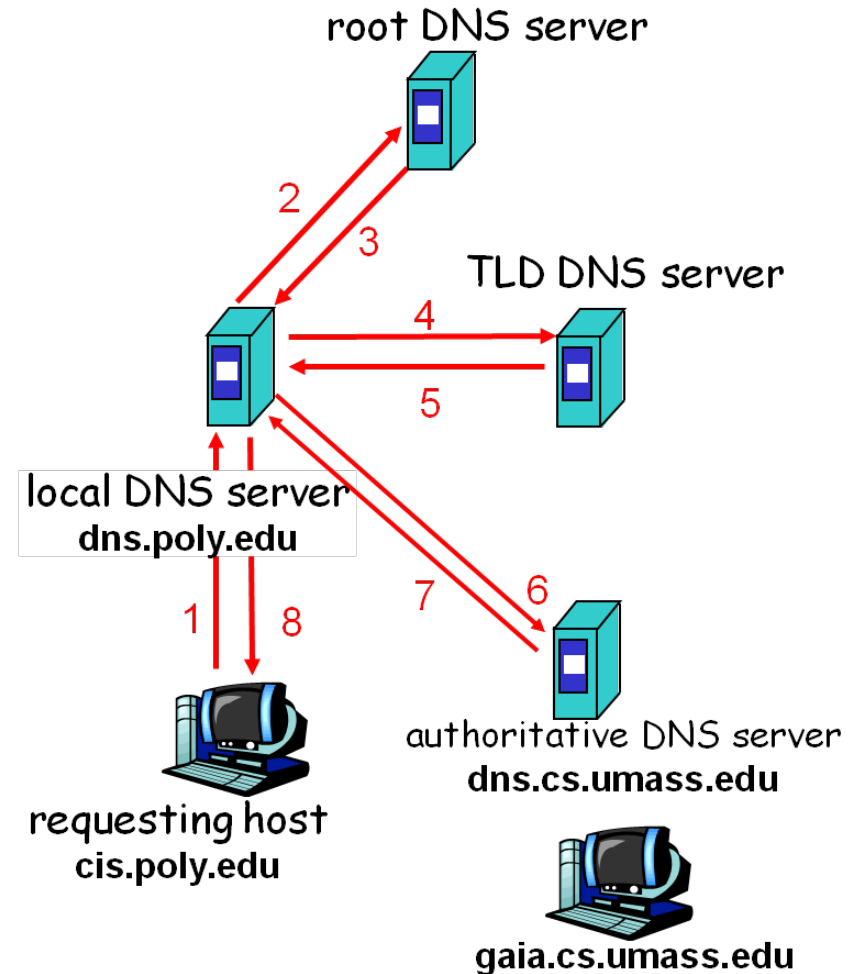


Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

## DNS : Résolution de nom (2)

- L'hôte cis.poly.edu veut obtenir l'adresse IP associée à gaia.cs.umass.edu
- Procédure de requête récursive

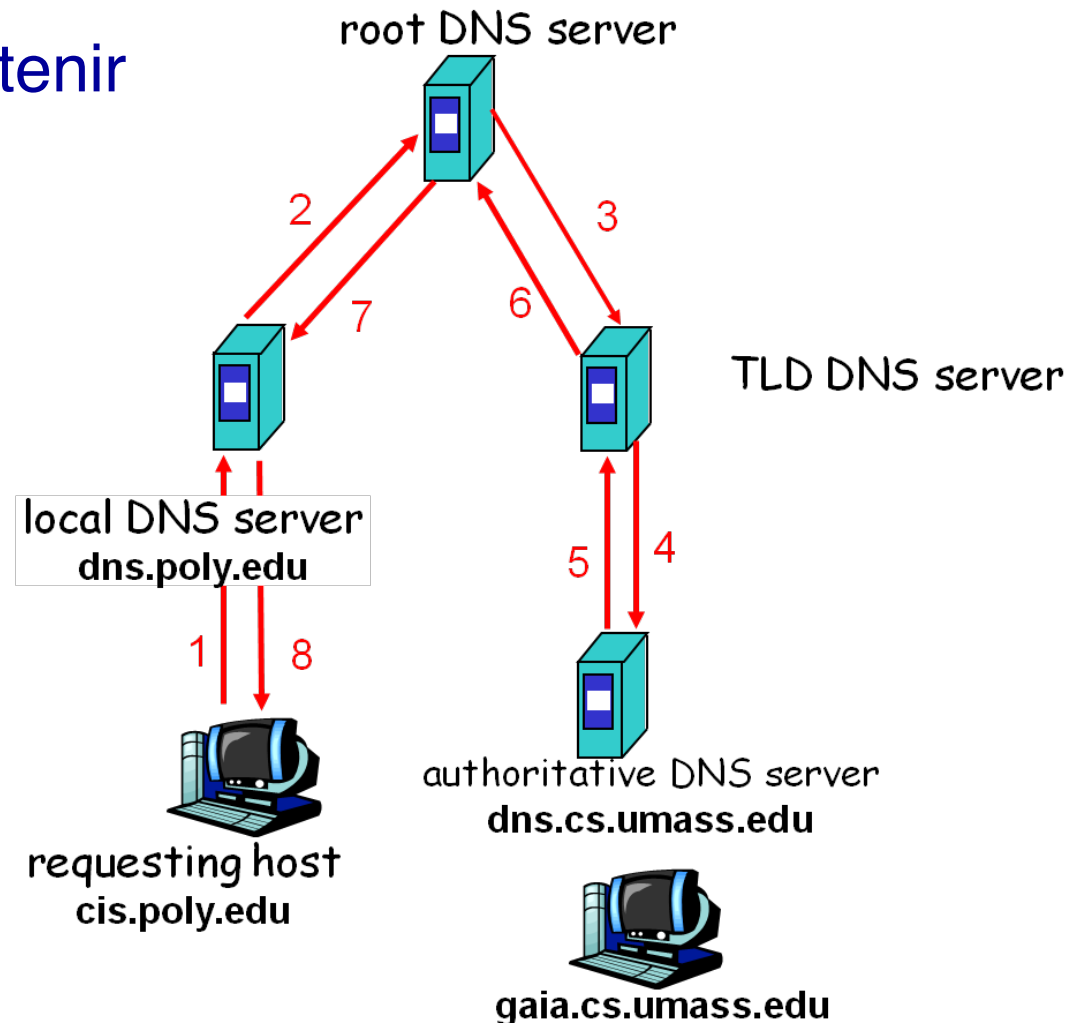


Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

# DNS : résolution de noms (3)

---

## ■ Pour amorcer une résolution

- ◆ Tout hôte doit connaître l'adresse IP d'un serveur de noms local (dans sa zone)
- ◆ Il est recommandé de connaître deux ou trois adresses, pour la tolérance aux fautes

## ■ Ces adresses :

- ◆ sont fournies aux utilisateurs de l'hôte par l'administrateur de sa zone (ou son fournisseur d'accès à l'Internet)
- ◆ sont généralement inscrites "à la main" dans les tables de configuration pour l'accès à l'Internet ...
- ◆ mais peuvent aussi être configurées automatiquement (via le protocole DHCP qui peut gérer la configuration réseau d'une machine lors de son raccordement à un réseau)

# DNS : résolution de noms (4)

---

## ■ Compléments :

- ◆ Lorsqu'un serveur apprend une information, il la conserve dans un cache local
- ◆ Les informations mises en cache ont une durée de vie limitée (pour garantir leur fraîcheur)
- ◆ Tous les serveurs DNS (même locaux) existent en plusieurs exemplaires

## ■ En pratique :

- ◆ Les serveurs ont beaucoup d'informations en cache donc :
  - ❖ une résolution de nom nécessite moins de messages que dans l'exemple précédent
  - ❖ les niveaux les plus hauts de la hiérarchie sont peu sollicités
- ◆ Les interactions pour une résolution suivent généralement le premier schéma :
  - ❖ Mode récursif entre le demandeur et son serveur local
  - ❖ Mode itératif entre le serveur local et les serveurs de la hiérarchie

# DNS : résumé des principes

---

- Algorithme de recherche décentralisé (pas de point de décision unique)
- Hiérarchie de serveurs calquée sur la hiérarchie des zones (voisines des domaines )
- Usage intensif de caches (informations dupliquées) et d'indicateurs (informations probablement valides permettant un accès rapide la plupart du temps)
  - ◆ Toute information peut être obtenue par plusieurs voies
  - ◆ La validité (« fraîcheur ») de toute information peut être confirmée si nécessaire
- Ces règles favorisent aussi la tolérance aux pannes

# DNS : détails techniques

---

■ Chaque serveur gère une table d'entrées

■ Contenu des champs d'une entrée

*Nom, Valeur, Type, Classe, Durée de vie*

- ◆ *Type* : indique comment les champs *Valeur* et *Nom* doivent être interprétés
  - ❖ *Type = A* : Nom de machine, *Valeur = adresse IP*
  - ❖ *Type = NS* : Nom de domaine, *Valeur = nom d'un serveur autorité*
  - ❖ *Type = CNAME* : *Valeur = nom "canonique" d'un hôte (le nom de référence si alias)*
  - ❖ *autres* : non traités ici (aide à la gestion du *mail*, etc.)
  
- ◆ *Classe* : IN (l'Internet), presque toujours
  
- ◆ *Durée de vie* (TTL ou *Time To Live*) : durée de validité de l'enregistrement
  - ❖ Application d'un principe général : toute information doit être périodiquement validée



# Utiliser DNS (1)

---

DNS est automatiquement utilisé par toute application sur l'Internet utilisant des noms de domaines. Mais on peut aussi y accéder directement

## Au niveau du langage de commande (*shell*)

Commande `dig` (*Domain Information Groper*) - voir `man`

```
dig www.google.com
dig -x 194.199.25.39 # -x : adresse vers nom
```

## Au niveau des appels systèmes Unix

```
#include <netdb.h>
struct hostent *gethostbyname(const char *name);
struct hostent *gethostbyaddr(const char *addr, int len, 0);
```

variantes : `getnameinfo` et `getaddrinfo`

voir Doc. technique

## Utiliser DNS (2)

```
mandelbrot:~> dig www.google.com
```

```
; <<>> DiG 9.3.4 <<>> www.google.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24632
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 7, ADDITIONAL: 7
```

```
;; QUESTION SECTION:
;www.google.com.                IN      A
```

```
;; ANSWER SECTION:
```

www.google.com.	355486	IN	CNAME	www.l.google.com.	← nom canonique
www.l.google.com.	238	IN	A	209.85.129.99	} entrées multiples
www.l.google.com.	238	IN	A	209.85.129.147	
www.l.google.com.	238	IN	A	209.85.129.104	

```
[puis noms et adresses des serveurs de noms pour le domaine
google.com ...]
```

```
;; Query time: 2 msec
;; SERVER: 152.77.2.5#53(152.77.2.5)
;; WHEN: Sun Mar  9 12:51:57 2008
;; MSG SIZE  rcvd: 324
```

## Utiliser DNS (3)

```
mandelbrot:~> dig -x 194.199.18.38

; <<>> DiG 9.7.2-P3 <<>> -x 194.199.18.38
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55325
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 2

;; QUESTION SECTION:
;38.18.199.194.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
38.18.199.194.in-addr.arpa. 7200   IN      PTR      bastion.inrialpes.fr.

;; AUTHORITY SECTION:
18.199.194.in-addr.arpa. 7200   IN      NS       ebene.inrialpes.fr.
18.199.194.in-addr.arpa. 7200   IN      NS       imag.imag.fr.
18.199.194.in-addr.arpa. 7200   IN      NS       if.inrialpes.fr.
} serveurs de noms

;; ADDITIONAL SECTION:
imag.imag.fr.      72447   IN      A        129.88.30.1
imag.imag.fr.      72447   IN      AAAA     2001:660:5301:1e::101

;; Query time: 12 msec
;; SERVER: 152.77.2.5#53(152.77.2.5)
;; WHEN: Thu Mar  3 18:34:11 2011
;; MSG SIZE rcvd: 183
```

# Résumé

---

- Introduction aux réseaux informatiques
  - ◆ Types et caractéristiques (LAN, MAN, WAN, mobiles, etc.)
  - ◆ Performances : latence, débit. Qualité de service
  
- Protocoles et interfaces
  - ◆ Fonctionnement d'ensemble d'un réseau
  - ◆ Notions de protocole et d'interface
  - ◆ Protocoles en couches, fonctions des principales couches
  
- Service de noms de l'Internet : protocole DNS
  - ◆ Désignation symbolique sur l'Internet : domaines
  - ◆ Exemple de gestion décentralisée, capacité de croissance

---

# Annexes

# Structure globale de l'Internet

---

## ■ Structure hiérarchique avec différents types de prestataires (« niveaux »/”tiers”)

- ◆ Il n'existe pas d'organisme/autorité pour définir précisément ces niveaux et leurs membres ... mais il y a cependant des états de fait (dont les caractéristiques sont plus ou moins disponibles publiquement)
- ◆ Un prestataire (*ISP: Internet Service Provider*) dispose d'un ensemble de réseaux

## ■ Idée générale

- ◆ Prestataire T1 (*”Tier-1 ISP”* ou *”Backbone networks”*)
  - ❖ Couverture internationale/mondiale
  - ❖ Chaque ISP T1 est interconnecté avec tous les autres ISP T1
  - ❖ ... et avec un grand nombre d'ISP T2
  - ❖ Très haut volume de trafic / hautes performances
  - ❖ Une dizaine (ou quelques dizaines) d'ISP T1 dans le monde
- ◆ Prestataire T2 (*”Tier-2 ISP”*)
  - ❖ Couverture souvent nationale ou régionale
  - ❖ Relié (techniquement et contractuellement) à un ou quelques ISP T1
  - ❖ Souvent aussi relié à des ISP T2 (potentiellement beaucoup)
  - ❖ Ne s'appuie en général pas majoritairement sur l'infrastructure du T1 (cf. peering/transit plus loin)
  - ❖ Quelques centaines d'ISP T2 dans le monde

# Structure globale de l'Internet (suite)

---

## ■ Idée générale (suite)

- ◆ Prestataire T3 ou de niveau supérieur ("*Tier-3 ISP*")
  - ❖ Typiquement, joue le rôle de fournisseur d'accès Internet pour les particuliers ou les PME
  - ❖ Relié à un (ou quelques) prestataire(s) T1/T2
  - ❖ Quelques milliers d'ISP T3 dans le monde
- ◆ Intégration verticale :
  - ❖ Certains prestataires fournissent des prestations à plusieurs niveaux (T1/T2) pour différents types de clients
- ◆ *Point-of-Presence (POP)*
  - ❖ Point de jonction/raccordement d'un client à un ISP, c'est-à-dire le point où les paquets émis par le client entrent dans le réseau de l'ISP
- ◆ *Internet Exchange Point (IXP)*
  - ❖ Centre/local informatique dans lequel convergent des réseaux de différents ISP ayant des accords d'interconnexion
  - ❖ Joue un rôle de « plaque tournante » pour le trafic de l'Internet
  - ❖ Un IXP peut héberger des POP

# Structure globale de l'Internet (suite)

---

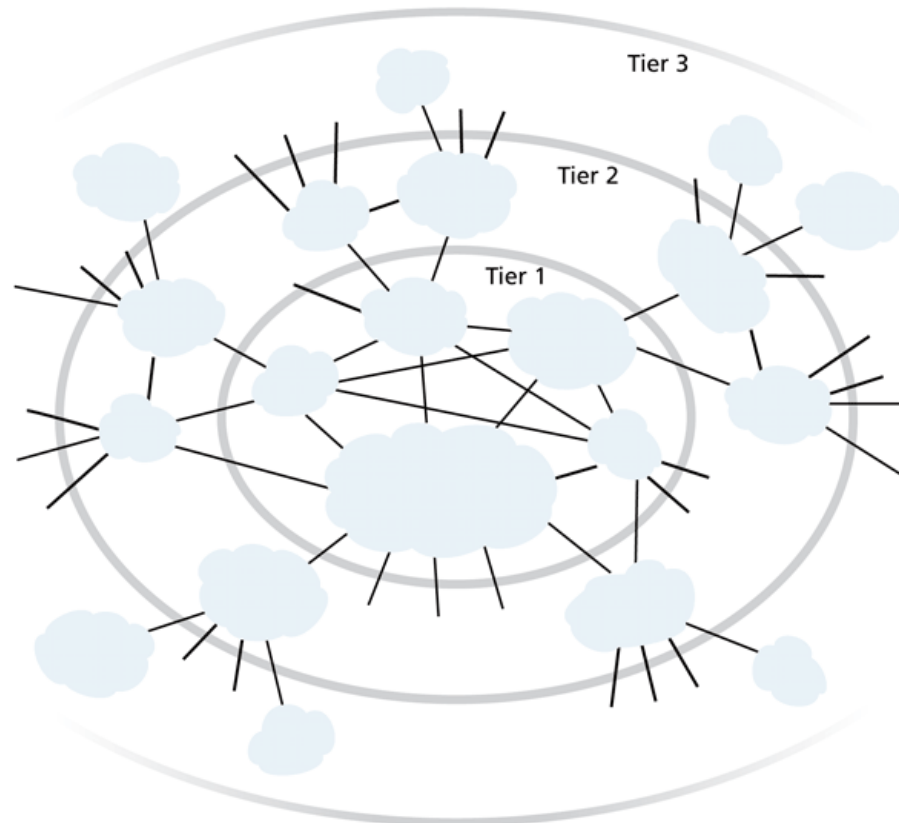
## ■ Idée générale (suite)

- ◆ Différents types d'accords entre ISP : *transit* et *peering*
- ◆ *Transit* :
  - ❖ Un ISP x paye un ISP y pour acheminer ses données vers des réseaux destinataires qu'il ne peut pas joindre par l'intermédiaire de ses propres réseaux
  - ❖ (par exemple, facturation au volume de données)
- ◆ *Peering* :
  - ❖ Échanges mutuels (sans frais) de trafic entre ISP
- ◆ En pratique :
  - ❖ Relations souvent de type transit entre ISP T2/T1 et T3/T2
  - ❖ Relations souvent de type peering entre ISP T2/T2 et T1/T1 de même envergure
  - ❖ Certains définissent la catégorie ISP T1 par le critère suivant : couverture globale de l'Internet sans nécessité d'accords de transit
  - ❖ Les détails des accords sont souvent confidentiels
- ◆ Lorsqu'un paquet est acheminé sur l'Internet d'une machine A à une machine B, en fonction des réseaux concernés (et des ISP correspondants) :
  - ❖ Le paquet ne traverse pas nécessairement un réseau T1 (ni T2)
  - ❖ Le paquet n'emprunte pas nécessairement le plus court chemin (ni en termes de distance physique, ni en termes de nombre de routeurs) en raison notamment de l'influence des accords existant (ou non) entre ISP et des choix de routage



# Structure globale de l'Internet (suite)

---



**Figure 1.11** ♦ Interconnection of ISPs

Schéma emprunté à Kurose & Ross, Computer Networking: a Top-Down Approach

# Structure globale de l'Internet (suite)

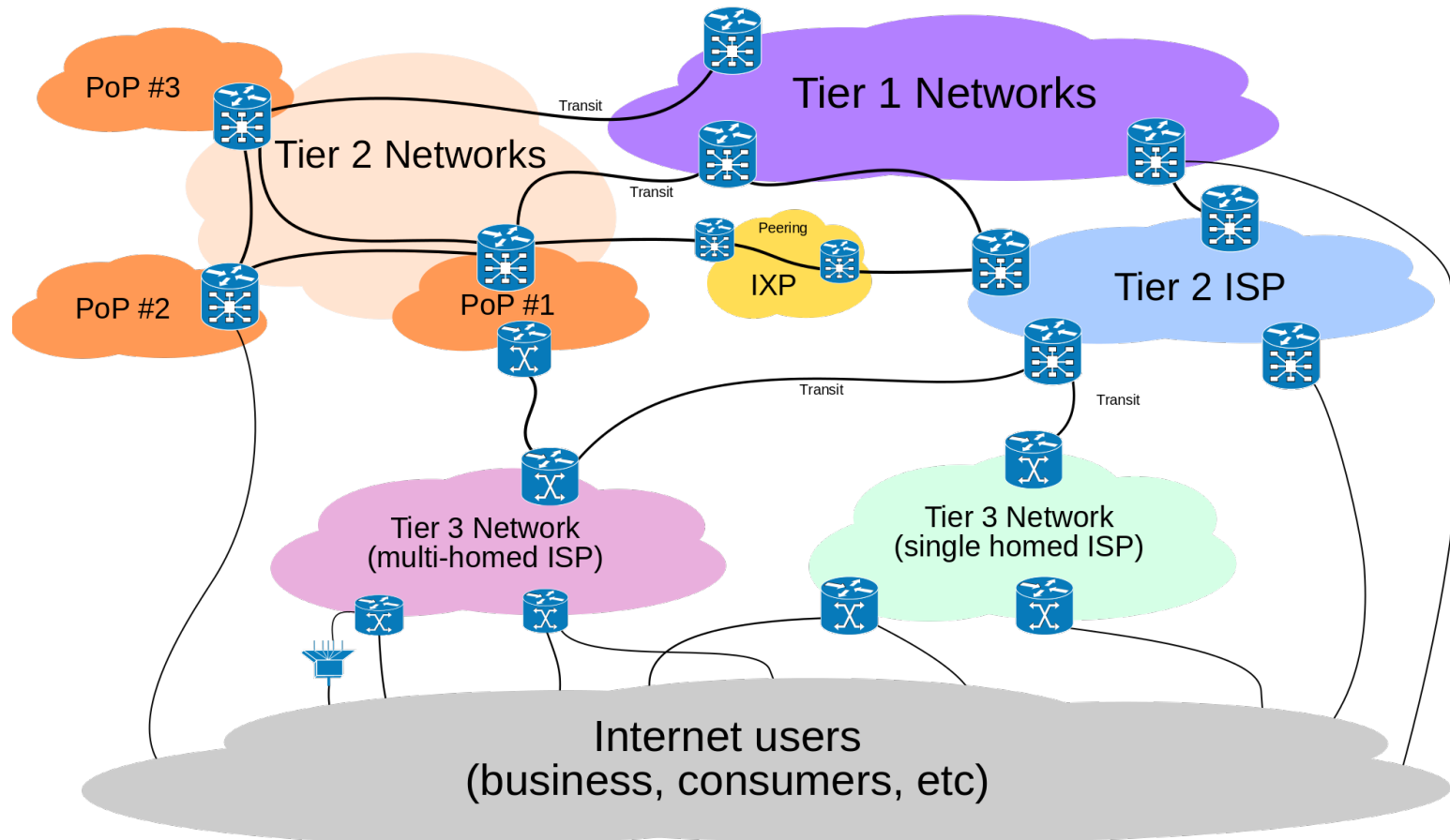


Schéma emprunté à Wikipedia : [http://en.wikipedia.org/wiki/Internet\\_service\\_provider](http://en.wikipedia.org/wiki/Internet_service_provider)

# Structure globale de l'Internet (suite)

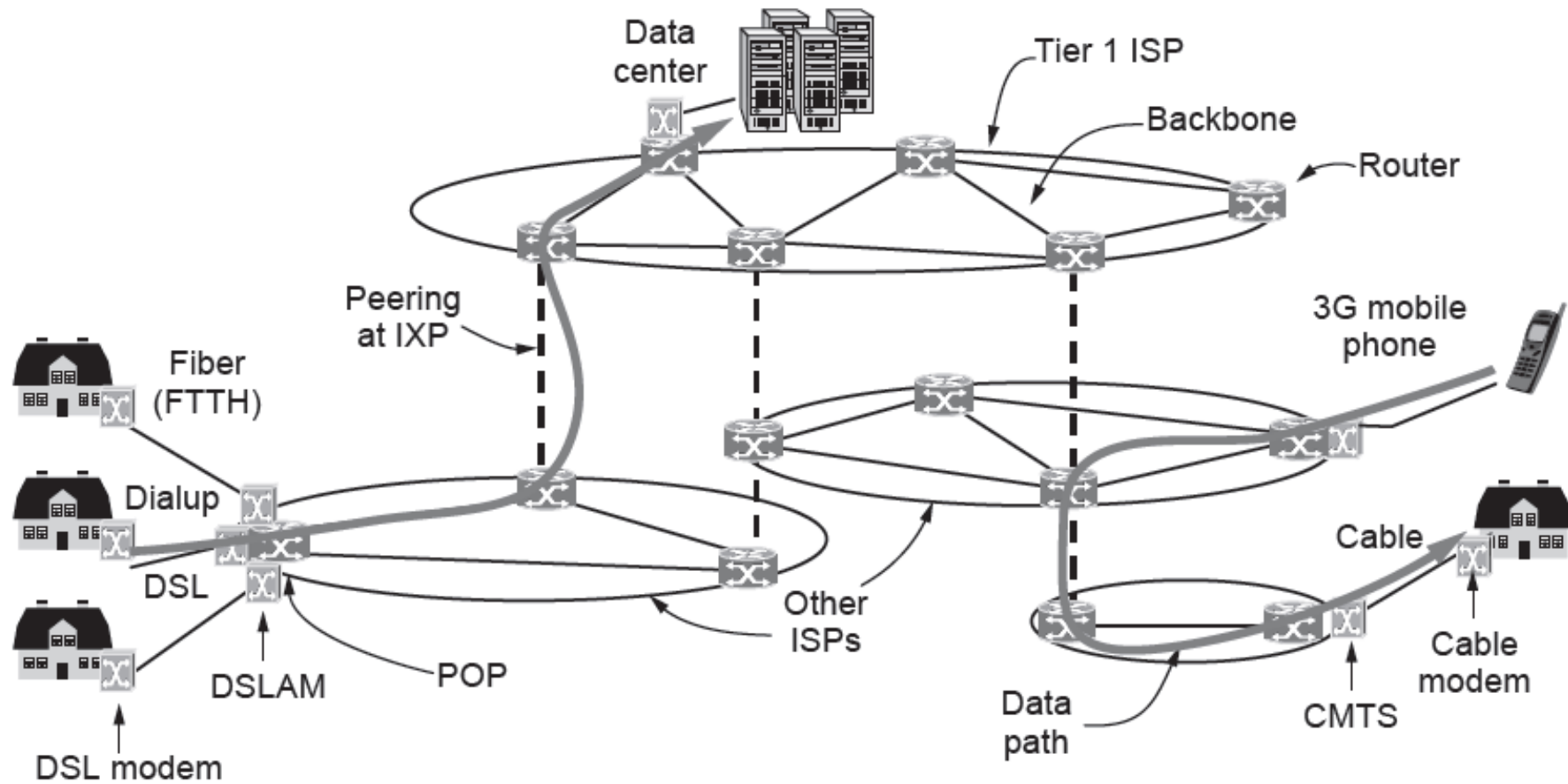


Schéma emprunté à A. Tanenbaum et D. Wetherall. Computer Networks, 5th edition, Pearson Education

# Structure globale de l'Internet (suite)

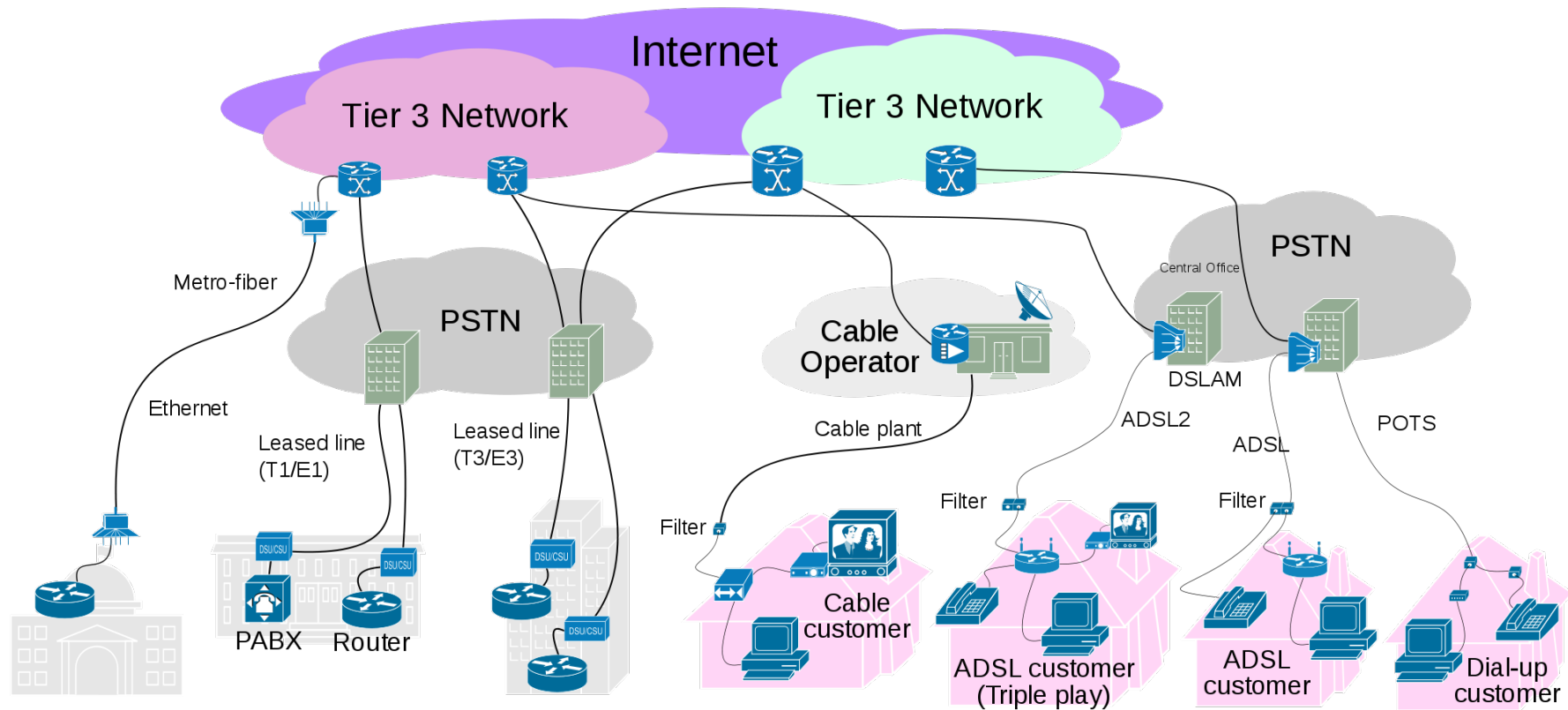


Schéma emprunté à Wikipedia : [http://en.wikipedia.org/wiki/Internet\\_service\\_provider](http://en.wikipedia.org/wiki/Internet_service_provider)

# Structure globale de l'Internet (suite)

---

## ■ Quelques ressources pour plus de détails :

### ◆ Wikipedia

- ❖ [http://en.wikipedia.org/wiki/Tier\\_1\\_network](http://en.wikipedia.org/wiki/Tier_1_network)
- ❖ [http://en.wikipedia.org/wiki/Tier\\_2\\_network](http://en.wikipedia.org/wiki/Tier_2_network)
- ❖ [http://en.wikipedia.org/wiki/Internet\\_service\\_provider](http://en.wikipedia.org/wiki/Internet_service_provider)
- ❖ [http://fr.wikipedia.org/wiki/Maillage de l%27infrastructure Internet en France](http://fr.wikipedia.org/wiki/Maillage_de_l%27infrastructure_Internet_en_France)
- ❖ [http://fr.wikipedia.org/wiki/Internet Exchange Point](http://fr.wikipedia.org/wiki/Internet_Exchange_Point)

### ◆ Données sur la topologie de l'Internet et son trafic

- ❖ Voir notamment <http://www.telegeography.com> et <http://www.pch.net>
- ❖ Exemple : Cartographie interactive des IXP
  - ▲ <http://www.telegeography.com/telecom-resources/internet-exchange-map/index.html>
  - ▲ <https://prefix.pch.net/applications/ixpdir/>

### ◆ Détails avancés sur les accords entre ISP

- ❖ *Ars Technica: How the Net works – an introduction to peering and transit (2008).* <http://arstechnica.com/features/2008/09/peering-and-transit/>
- ❖ W. Norton. The Art of Peering. <http://drpeering.net/white-papers/Art-Of-Peering-The-Peering-Playbook.html>
- ❖ Site DrPeering.net. <http://drpeering.net>

---

## **Annexes (suite)**

**Quelques illustrations graphiques et quantitative de la structure de l'Internet**

# Ressources

---

- **Cartes de la topologie du réseau et statistiques sur le trafic**
  - ◆ Voir les Telecom maps du site <http://www.telegeography.com>
  - ◆ Dont notamment:
    - ❖ Topologie planétaire et topologies continentales
    - ❖ Carte des liaisons sous-marines
  - ◆ Pour les liaisons sous-marines, voir aussi : <http://www.cablemap.info>
  - ◆ Un article/reportage assez complet sur les « entrailles » de l'Internet : <http://arstechnica.com/information-technology/2016/05/how-the-internet-works-submarine-cables-data-centres-last-mile/>
  
- **Statistiques sur la croissance du Web**
  - ◆ Voir les enquêtes périodiques de Netcraft: <http://www.netcraft.com>
  
- **Statistiques sur les noms DNS**
  - ◆ Voir les enquêtes périodiques de l'ISC: <https://www.isc.org/services/survey/>
  - ◆ <http://www.whois.sc/internet-statistics/>
  
- **Statistiques sur l'adoption d'IPv6**
  - ◆ <http://www.internetsociety.org/deploy360/ipv6/statistics/>
  - ◆ <https://getipv6.info/display/IPv6/IPv6+Penetration+Survey+Results>

# Déploiement et maintenance des liaisons sous-marines

---

- Voir par exemple “Submarine network solutions” d’Alcatel Lucent
  - ◆ <http://www.alcatel-lucent.com/solutions/submarine-networks>
  - ◆ <http://www.youtube.com/playlist?list=PL169FF67809A0C837>
- Ou encore Orange Marine
  - ◆ <http://marine.orange.com>
  - ◆ [http://www.orange.com/sirius/dossiers\\_anim/cables\\_sous\\_marins/index\\_fr.html](http://www.orange.com/sirius/dossiers_anim/cables_sous_marins/index_fr.html)
- Et aussi
  - ◆ [http://fr.wikipedia.org/wiki/Câble\\_sous-marin](http://fr.wikipedia.org/wiki/Câble_sous-marin)