

REZ 404

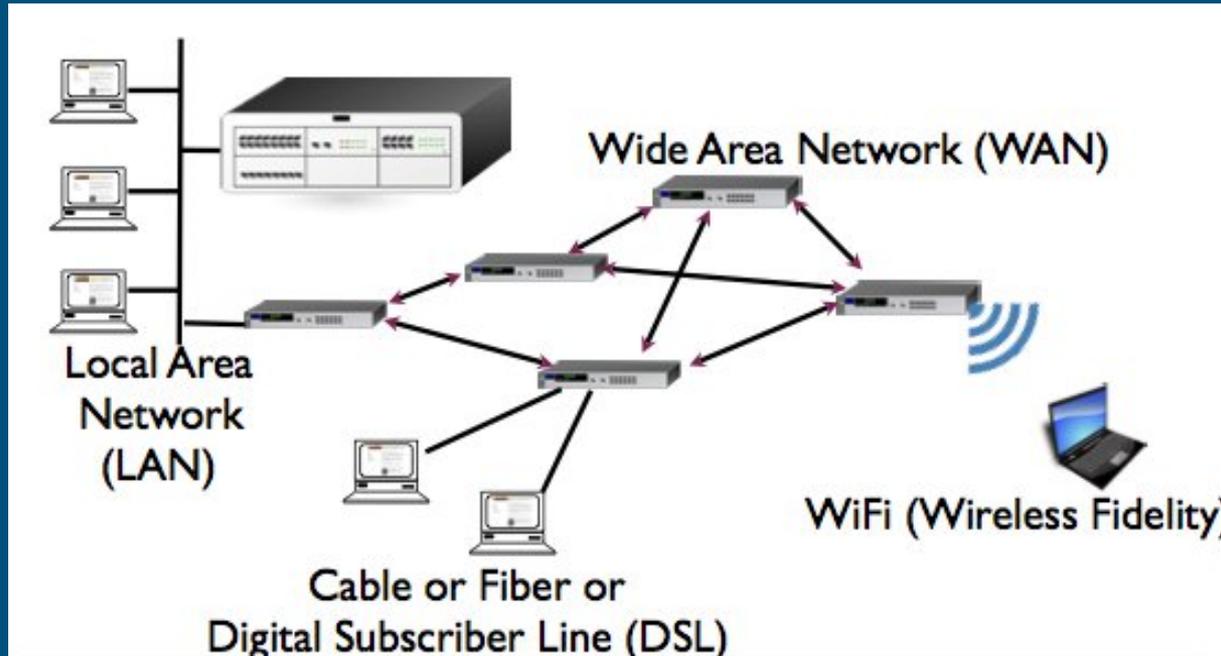
validation not found

comment valider le réseau en nainA en
apprenant qu'à Rezel c'est stylé

veuillez ne pas activer votre micro et votre
caméra, utilisez le chat pour les questions

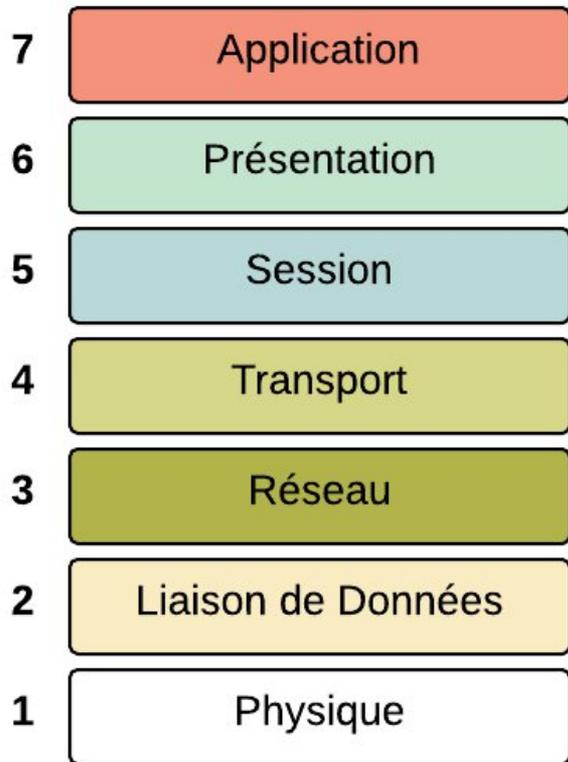


Un réseau



Le modèle par couches "OSI"

Théorie : Modèle OSI



Services applicatifs au plus proche des utilisateurs



Encode, encrypte, compresse les données utiles



Etablit des sessions entre des applications



Etablit, maintien et termine des sessions entre des périphériques terminaux



Adresse les interfaces globalement et détermine les meilleurs chemins à travers un inter-réseau

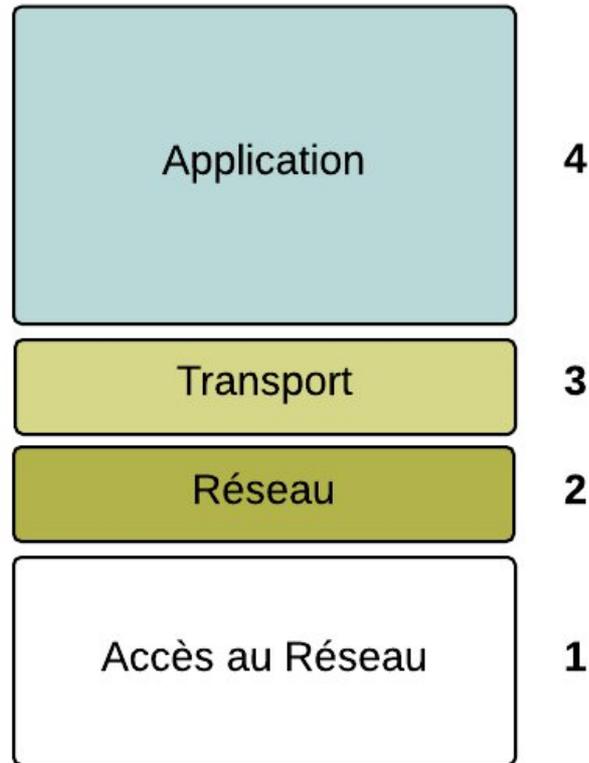


Adresse localement les interfaces, livre les informations localement, méthode MAC

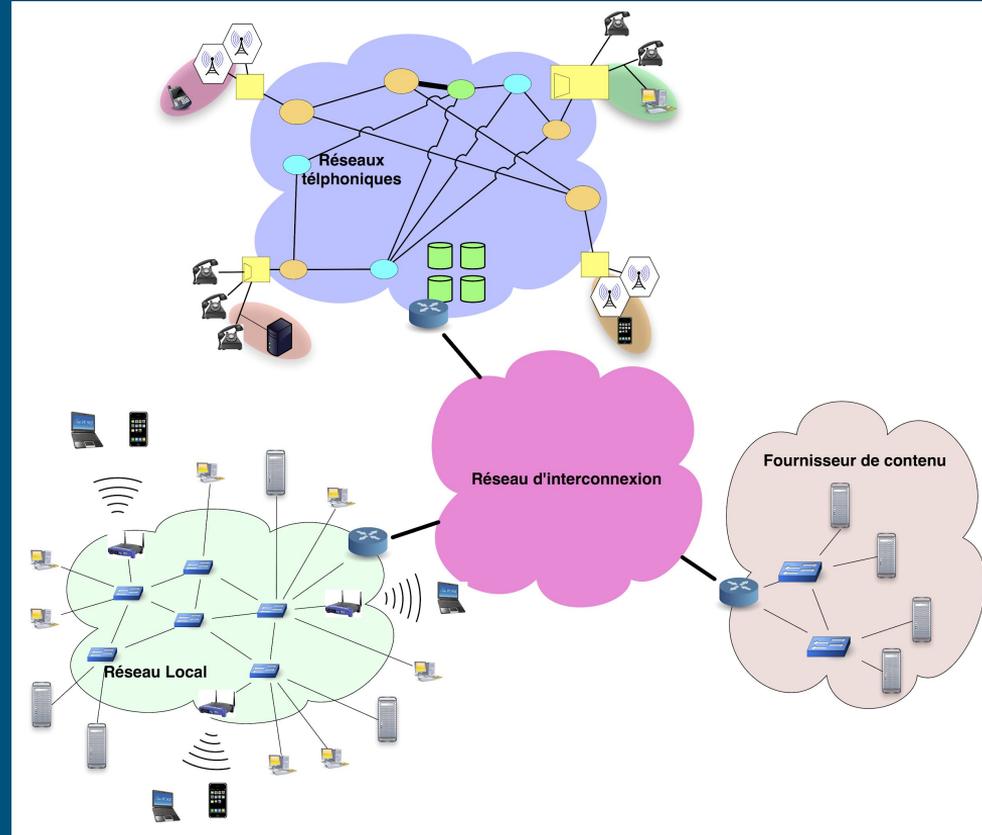


Encodage du signal, câblage et connecteurs, spécifications physiques

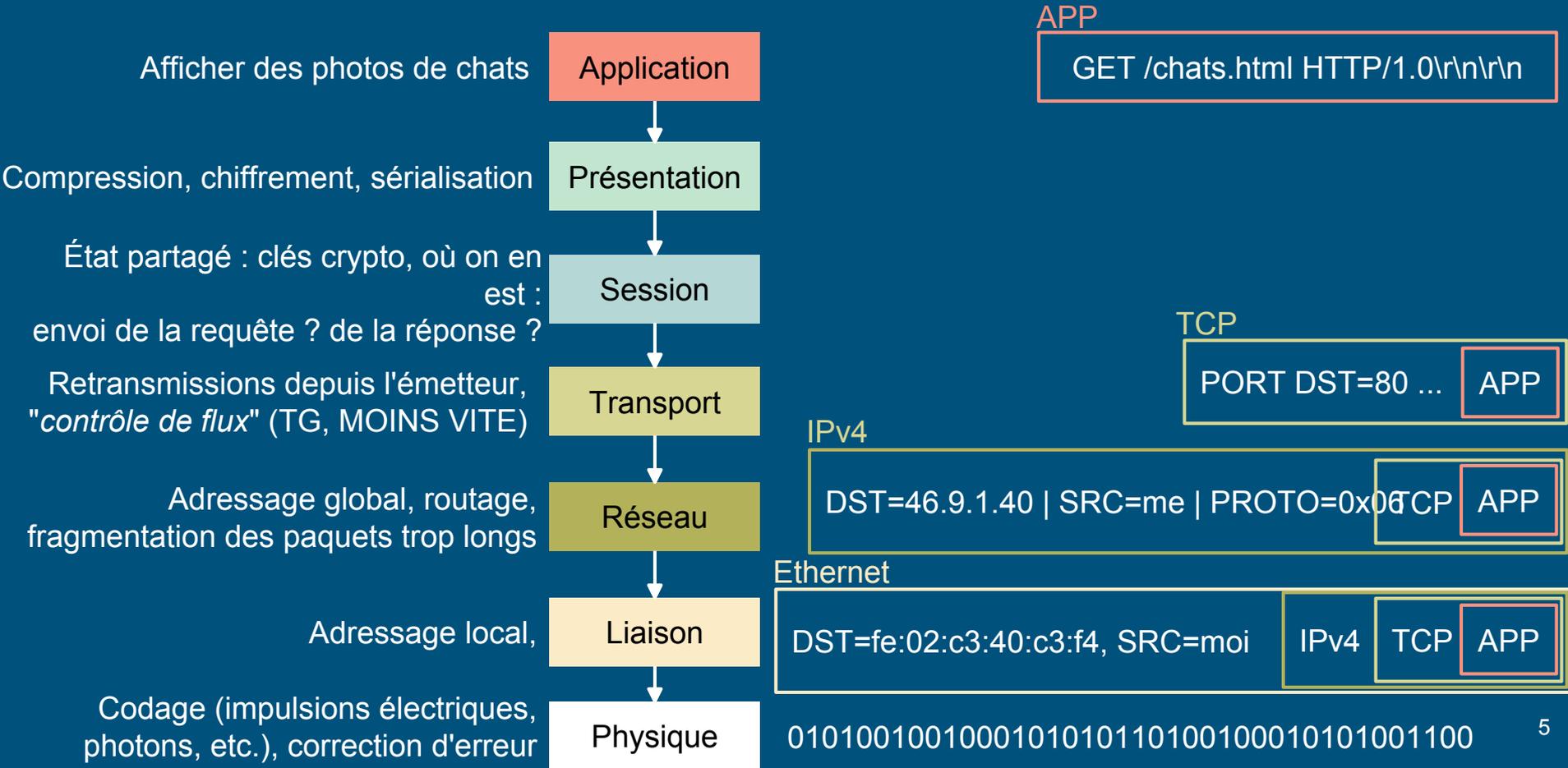
Pratique : Modèle TCP/IP



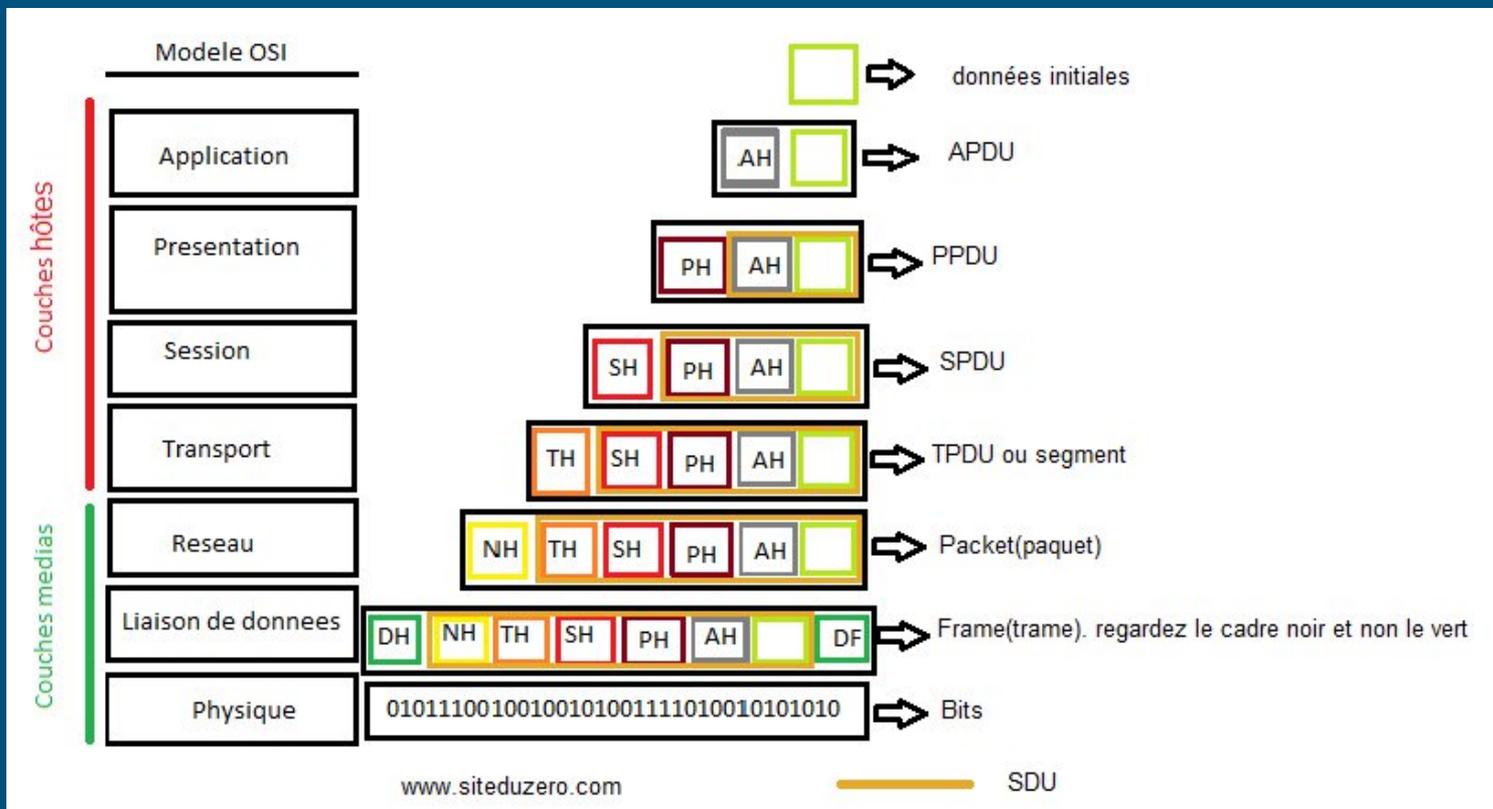
Internet



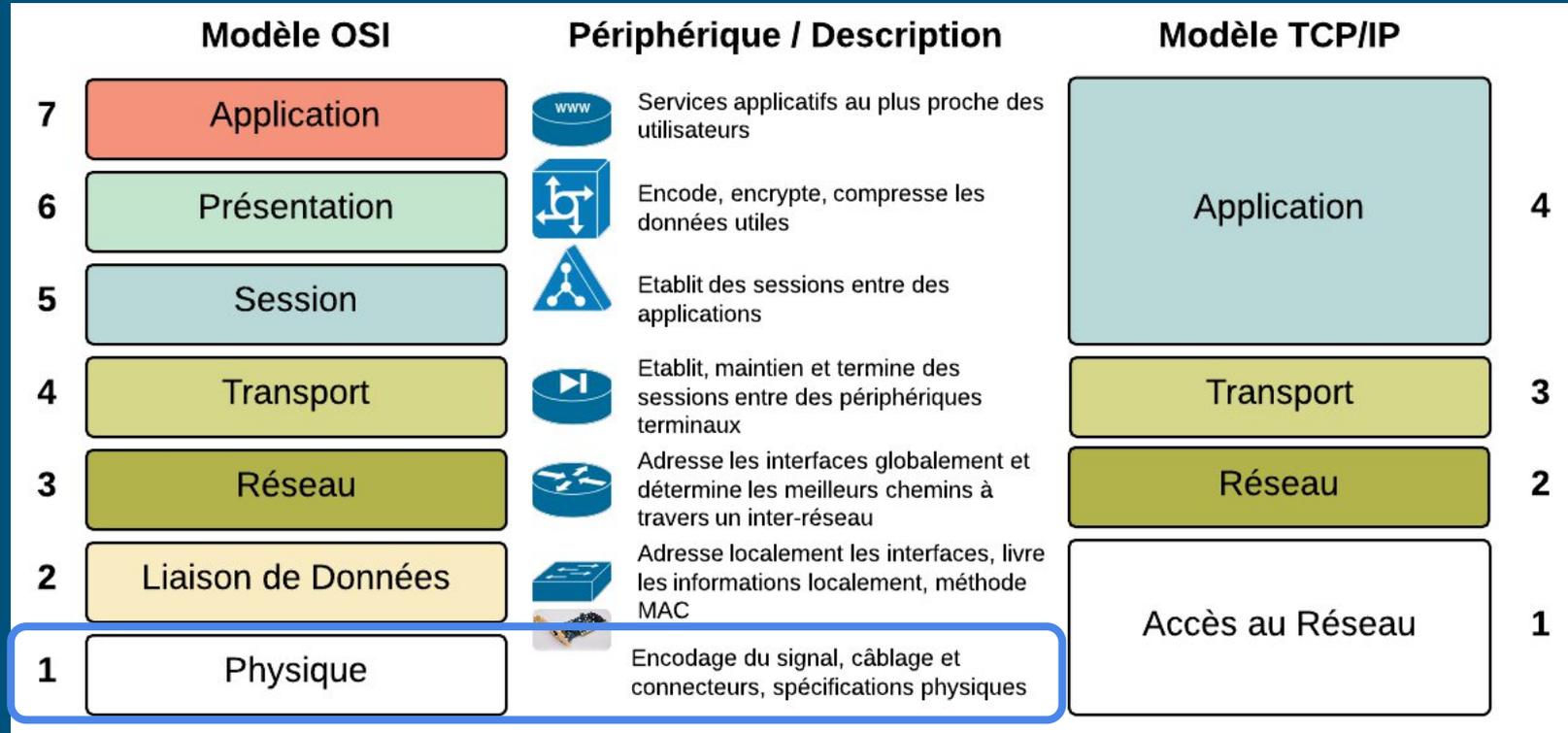
Un exemple : 99% du trafic mondial



Le modèle OSI IRL : des poupées russes



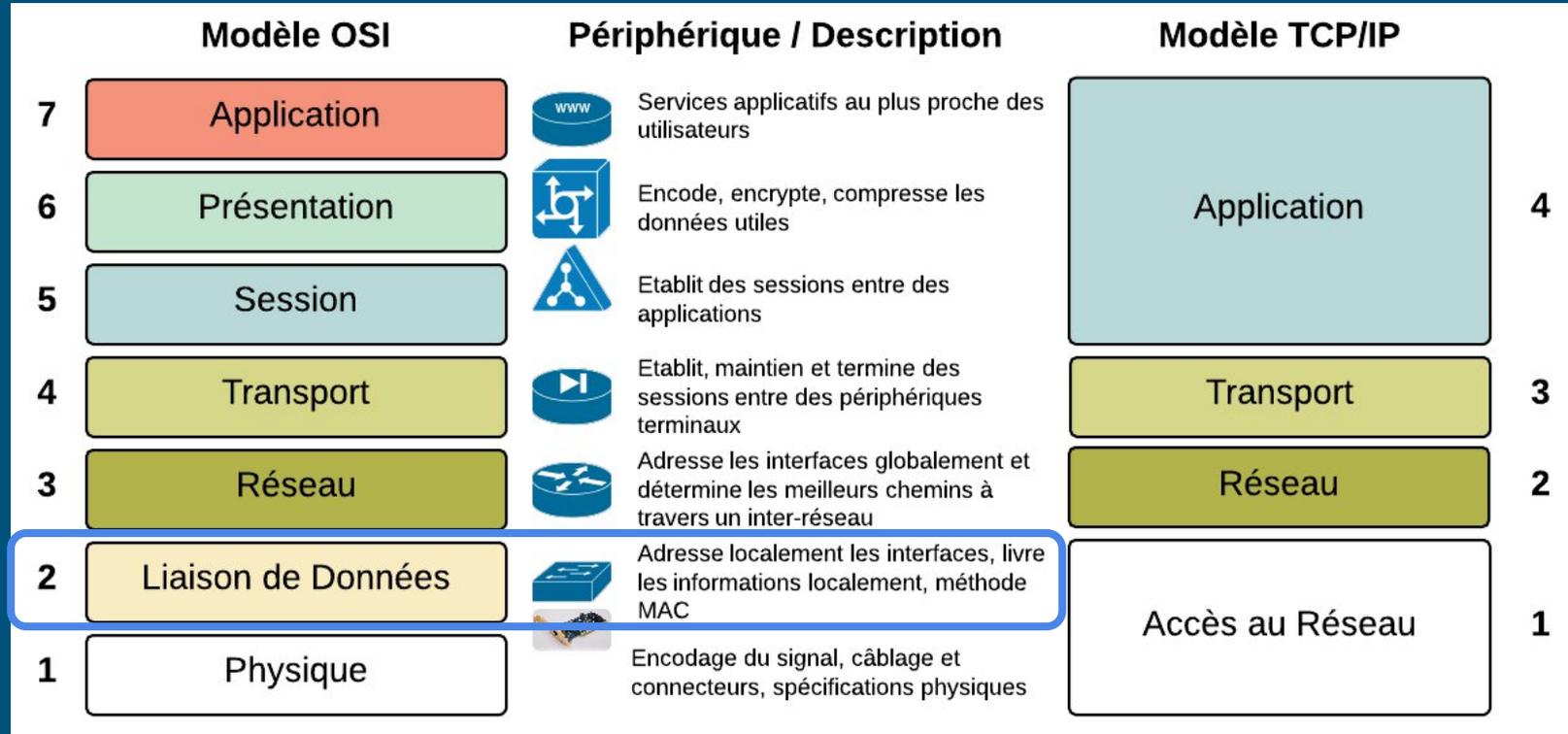
Couche 1 : Physique



Couche 1 : Physique

- des symboles, souvent des bits : des 1 et des 0
- différents mediums
 - fibre
 - fibre optique
 - paire torsadée (câble téléphone classique)
 - câble réseau (et non pas câble ethernet ...)
- Interconnecteur associé : le hub = multiprise réseau
- Balek™

Couche 2 : liaison de données



Couche 2 : liaison de données

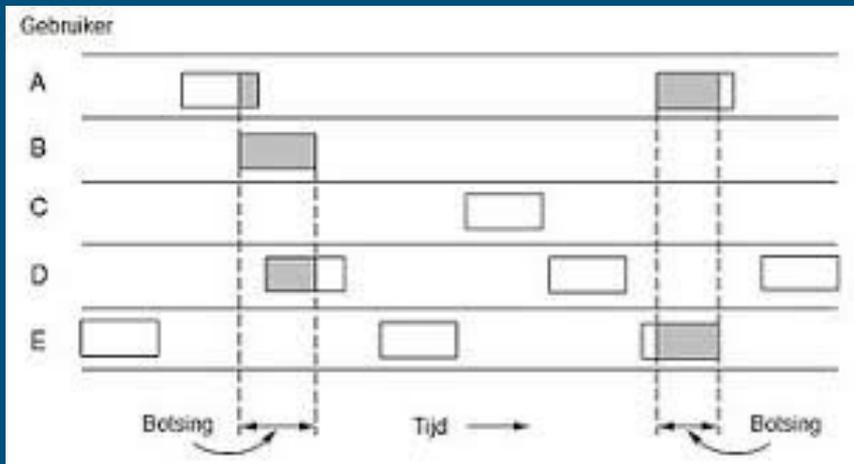
Accès partagé au support physique : quand vous discutez avec quelqu'un, ne parlez pas en même temps

Si tout le monde parle en même temps sans s'écouter cela crée des "collisions".
(canal partagé => collisions)

Comment éviter cela ?

- ALOHA
- CSMA-CD
- Ethernet

ALOHA & ALOHA Discrétisé : exemple #Balek



Des antennes émettent sur la même fréquence.

On veut parler ? alors on émet. Si collision, attente aléatoire puis réémission.

Discrétisation -> doublement du débit.

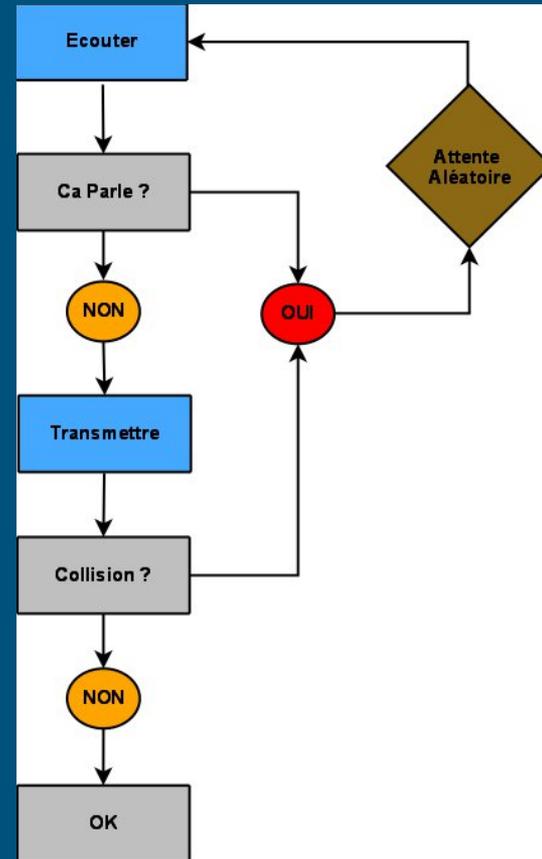
CSMA et CSMA - CD

Carrier **Sense** Multiple Access :

Écoute du canal pour savoir si quelqu'un parle

- Si on entend rien : On parle
- On écoute pendant qu'on parle
- Pas de réessai infini

Idéal pour les réseaux bondés



CSMA et CSMA - CD

Variantes principales :

- **1-persistent** : si collision, je réessaye dès que c'est libre (tout le monde va réessayer en même temps)
- **non-persistent** : si collision, j'attends un délai aléatoire pour reprendre
- **p-persistent** : même si c'est libre, on est probablement plusieurs à attendre donc je n'émet qu'avec $\text{Proba}=p$
- **CSMA-CD** : Collision Detection : arrête la transmission lors d'une collision
- **CSMA-CA** : Collision Avoidance :

à remplir avec les schémas p 18 de res101-ethernet-1p.ppt.pdf

Comment choisir p ?

Si p est petit :

- Les stations attendent longtemps avant de transmettre,
- Mais la probabilité de collision est faible (peu de retransmissions).

Si en moyenne n stations ont simultanément au moins un paquet à envoyer, le nombre moyen de stations qui émettent sur un slot libre est np :

- Si $np > 1$, la probabilité de collision est forte,
- On doit donc choisir au plus p tel que $p < 1/n$.

Plus le nombre de stations est important, plus faible doit être p et donc plus important est le délai d'accès.

Ethernet

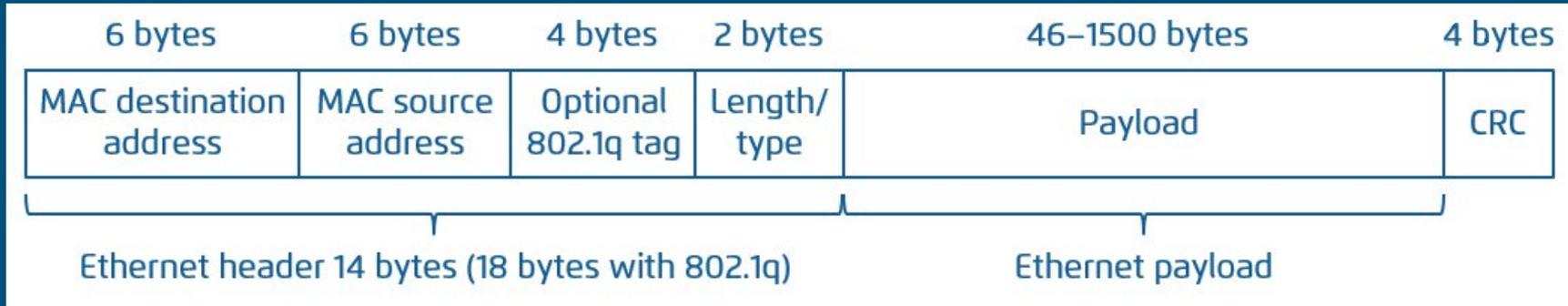
- 100BaseT : le cable que l'on connaît (possible sur fibre)
- En **lien local** (deux ordinateur connectés directement)
- Codage manchester (image p 39)
- half-duplex, full-duplex (deux paires différentes)
- Adresse MAC : ~unique pour chaque carte réseau
- f4:6d:04:0c:a0:6c
- FF:FF:FF:FF:FF:FF : broadcast
- Interconnecteur associé : le switch



Ethernet

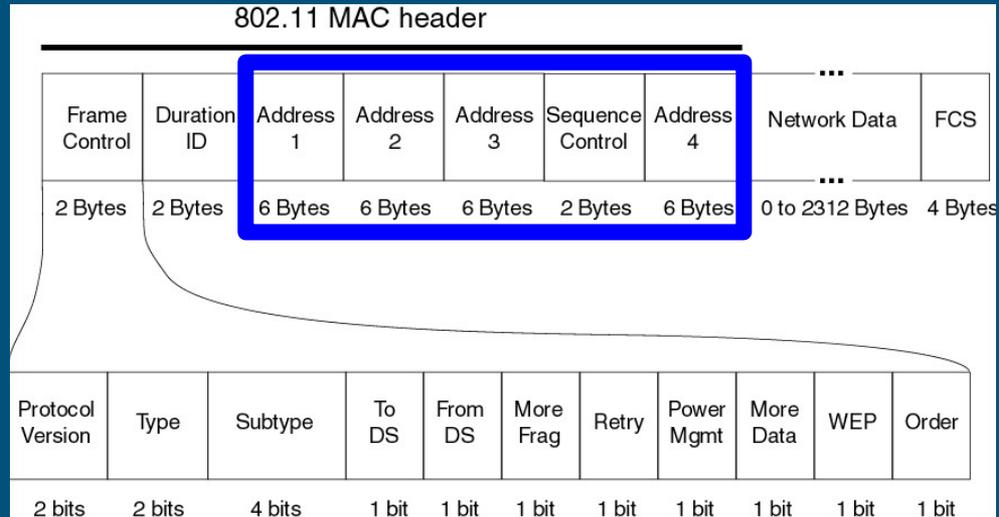
User Data : 46-1500 octets

Frame Check Sequence (FCS) : Cyclic Redundancy Check (CRC)



D'autres trucs que Ethernet

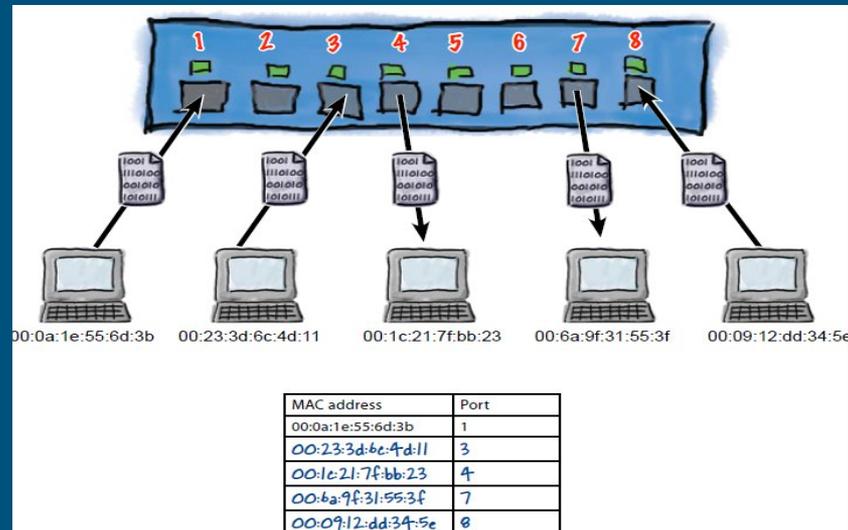
Wi-Fi (Wireless Fidelity) 802.11



Interconnexion : hub et switch/commutateurs

HUB : équivalent à une multiprise : envoi à tout le monde ;

SWITCH : fait la différence entre les clients, stocke les adresses MAC des PC connectés en mémoire, plus précisément dans une table.



Spanning Tree Protocol (STP : IEEE 802.1d)

Objectif : **éviter les boucles** : utile en cas de redondance d'équipement

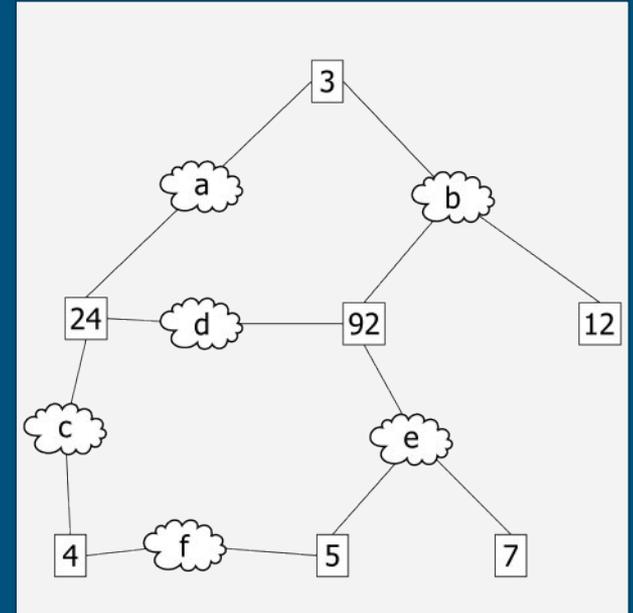
Step 1 : Election du *Root Bridge* (Le plus petit *Bridge ID* devient root)

Step 2 : Election des *Root Ports*

Step 3 : Election des *Designated Ports*

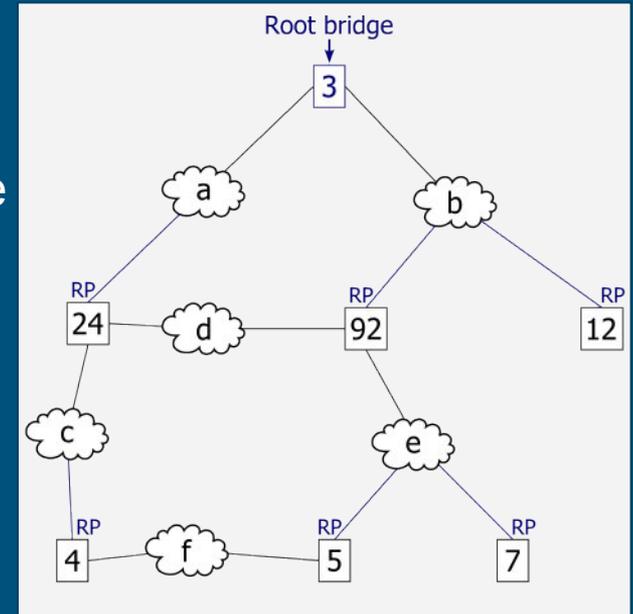
Spanning Tree Protocol (STP : IEEE 802.1d)

Éviter les boucles au niveau ethernet (les switch)



Spanning Tree Protocol (STP : IEEE 802.1d)

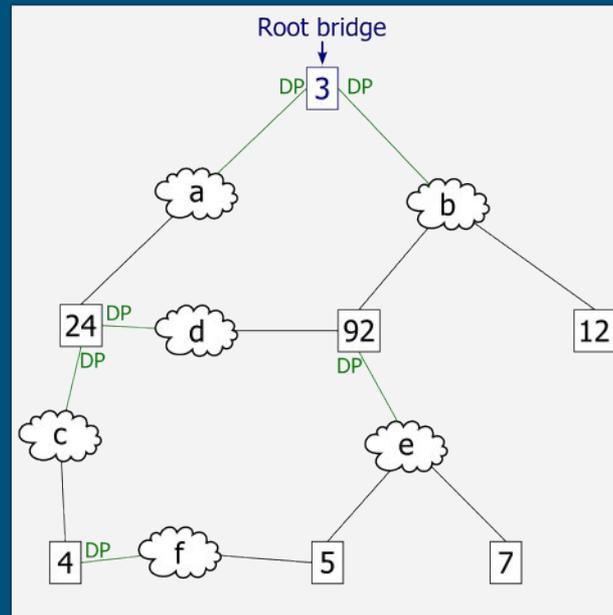
Election du **Root Bridge** => donne les **Root Ports**
pour chaque switch, le port le proche du Root Bridge



Spanning Tree Protocol (STP : IEEE 802.1d)

Election des *Designated Ports*

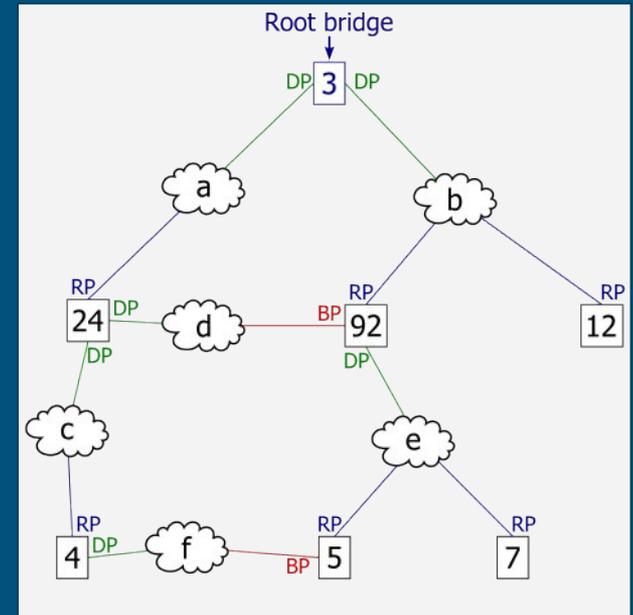
pour chaque segment Ethernet, le port qui sera utilisé pour lui parler



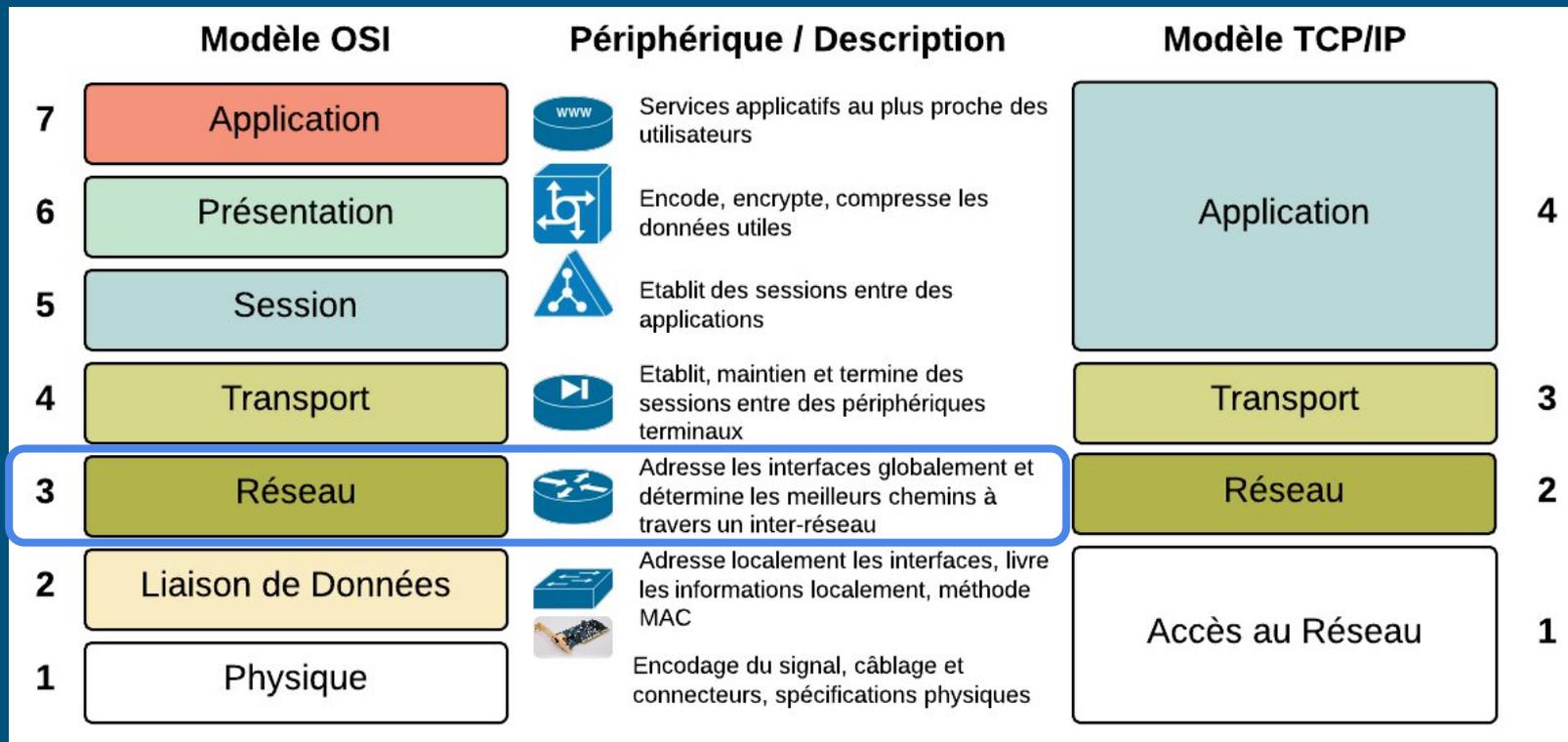
Spanning Tree Protocol (STP : IEEE 802.1d)

Détermination des *Blocked Ports*

tous les autres ports, non-designated et non-root-ports



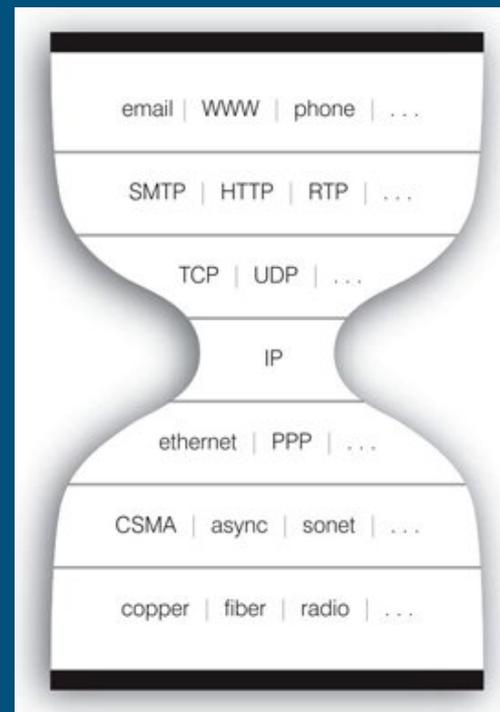
Couche 3 : Réseau (IP)



Couche 3 : Réseau (IP)

- Ethernet c'est bien mais ça marche qu'au niveau 2 (ex.: Maisel)
 - ça ne passe pas à l'échelle (car on augmente le domaine de collisions)
- IPv4 (Internet Protocol version 4) :
137.194.8.203
- IPv6 : 2006:74F1::2001:3402:3400:675F

Everything over IP & IP over Everything



IPv4

- Adresse : 4 octets, entre 0 et 255 : 192.168.0.1
- Masque de sous réseau : idem, mais suite de bits à 1 puis une suite de 0
 - Ex : 255.255.255.0 ou /24
 - Notation CIDR : nombre de 1 au début, ex : 255.255.252.0 => $4*8+6=$ /21
- Partie réseau : là où les bits sont à 1 dans le masque
- Partie machine : le reste
 - tous à 0 : adresse du réseau
 - tous à 1 : adresse de broadcast du réseau
 - le reste : les machines

IPv4

Une adresse IPv4 (notation décimale à point)

172 . 16 . 254 . 1

10101100 . 00010000 . 11111110 . 00000001

1 octet = 8 bits

32 bits (4 * 8), ou 4 octets

Telecom-Paritech.fr

137.194.0.0/16

↑
Prefix Length
(in bits)

137.194.0.0
137.194.0.1
137.194.0.2
...
137.194.255.255

$2^{16} = 65536$
addresses

Prefix	Number of Addresses
/8	~ 16 millions
/16	~ 65 000
/24	256
/25	128
/26	64

NetMask, Classes de Réseau et CIDR

Address	30.83.3.169	00011110.01010011.00000011.10101001
Mask	255.255.255.128	11111111.11111111.11111111.10000000
		AND
		00011110.01010011.00000011.10000000
In subnet	30.83.3.0 ?	00011110.01010011.00000011.00000000

- Adresses privées : 10/8, 172.16/12, 192.168/16
- Classes : A, B , C et D (attention, notion ~obsolète !)
- **Classless Inter-Domain Routing** : Masque de n'importe quelle taille

En-tête IPv4

En-tête IPv4																																							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31								
Version d'IP				Longueur de l'en-tête				Type de service								Longueur totale																							
Identification																Indicateur		Fragment offset																					
Durée de vie								Protocole								Somme de contrôle de l'en-tête																							
Adresse source																																							
Adresse destination																																							
Option(s) + remplissage																																							

IPv6

LOL c mieu, c + simpl

Version	Traffic class	Flow label	
Payload length		Next header	Hop limit
Source address			
Destination address			

Address Resolution Protocol (ARP)

Étant donnée une IP, donne l'adresse MAC à laquelle la joindre au niveau 2 :

1. Hey tout le monde, qui a @IP_{destination} ? Je suis @MAC_{source}, @IP_{source}
2. Hey @MAC_{source}, c'est moi @MAC_{destination}

Chaque machine a sa table IP → ARP

RARP : Reverse ARP protocole inverse, *!!\ obsolète*

ICMP : Internet Control Message Protocol

Ping : accessibilité d'une machine et temps de réponse

TTL : Time To Live : nombre de "saut" que le paquet peut effectuer. Un saut = un changement de réseau = un routeur

Traceroute : Permet de connaître tout les routeurs sur le chemin d'un paquet. (il peut y avoir plusieurs chemins possibles)

Fragmentation et Maximum Transmission Unit (MTU)

Chaque couche sa limite :

- Paquet IP : ≤ 65515 octets (fragmente au-delà)
- Segment TCP : ≤ 65535 octets (bufferise au-delà)
- Trame Ethernet : ≤ 1500 octets (t'envoie chier au-delà)

DHCP : Dynamic Host Configuration Protocol

PC : Bonjour, je viens de d'arriver sur le réseau, j'ai besoin d'une adresse IP pour communiquer avec le reste du monde : **DHCP Discover**

Serveur DHCP : Je peux t'offrir 137.194.8.123 : **DHCP Offer**

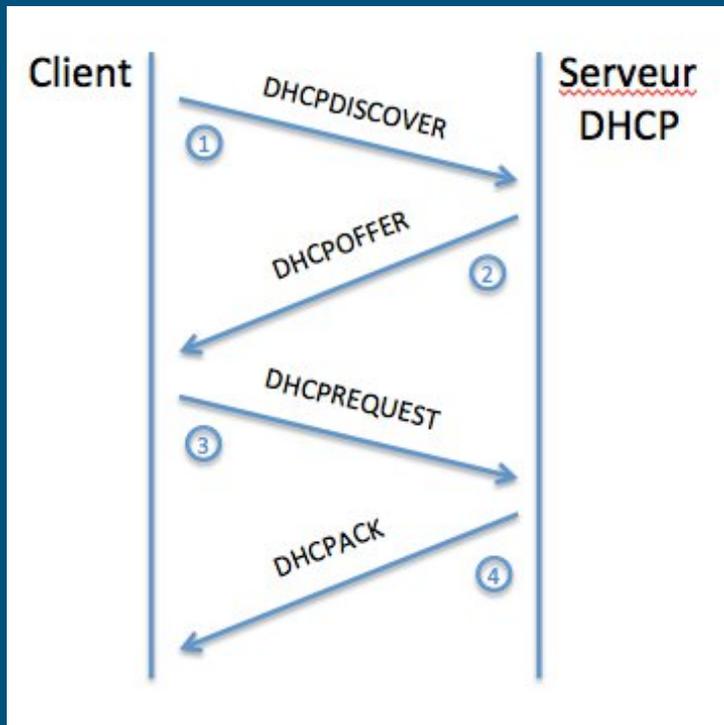
PC : Je prends ! : **DHCP Request**

Serveur DHCP : C'est bon c'est réservé pour toi : **DHCP Ack**
(ou : trop tard fallait répondre plus vite : **DHCP Nack**)

[...]

PC : plus besoin à présent, donne là à qui tu veux : **DHCP Release**

DHCP : Dynamic Host Configuration Protocol



Routage

acheminer et orienter les paquets.

Gateway : passerelle par défaut

table de routage

routage statique vs routage dynamique

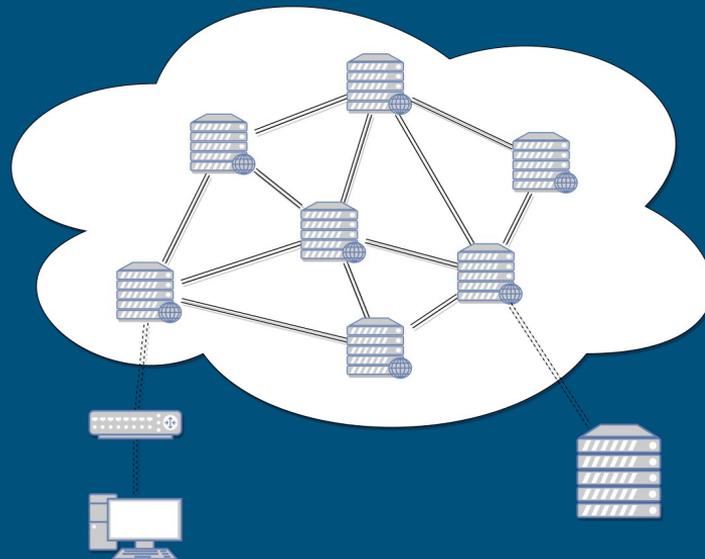


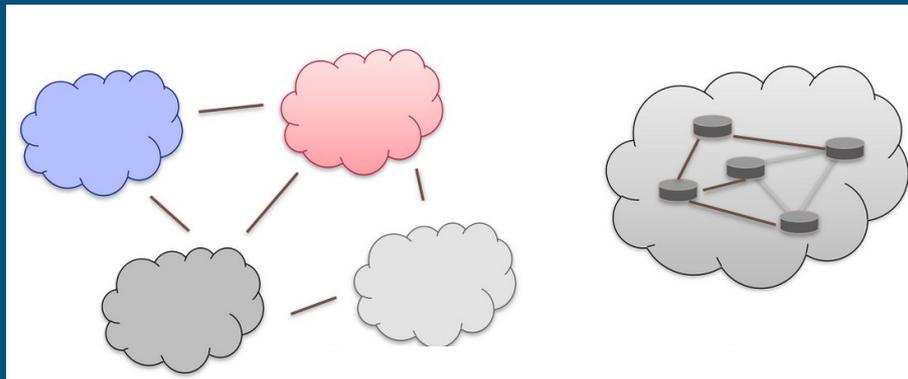
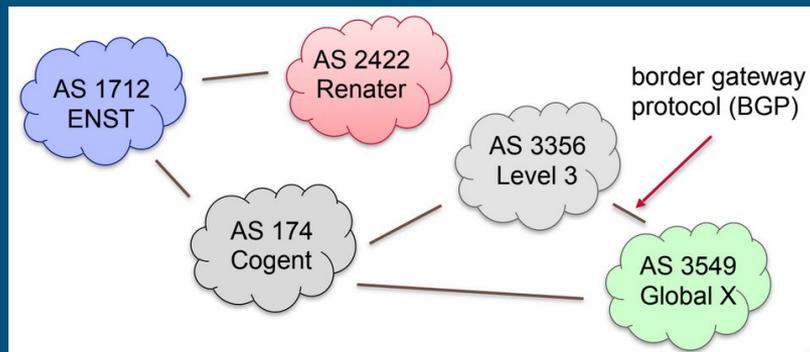
Table de routage

Address	Mask		Next Hop	Interface
17.24.0.0	255.255.0.0	(/16)	15.129.57.254	ge0
208.233.65.0	255.255.255.0	(/24)	123.79.17.254	ge1
22.11.2.0	255.255.255.0	(/24)	15.129.57.254	ge0
22.11.2.64	255.255.255.192	(/27)	15.129.57.253	ge0
15.129.57.0	255.255.255.0	(/24)	Link	ge0
123.79.17.0	255.255.255.0	(/24)	Link	ge1
67.234.14.128	255.255.255.128	(/25)	Link	fe0
0.0.0.0	0.0.0.0	(/0)	67.234.14.254	fe0

Protocoles de routage

eBGP : BGP

iBGP : OSPF / RIP



OSPF : Open Shortest Path First

Implémentation de Dijkstra

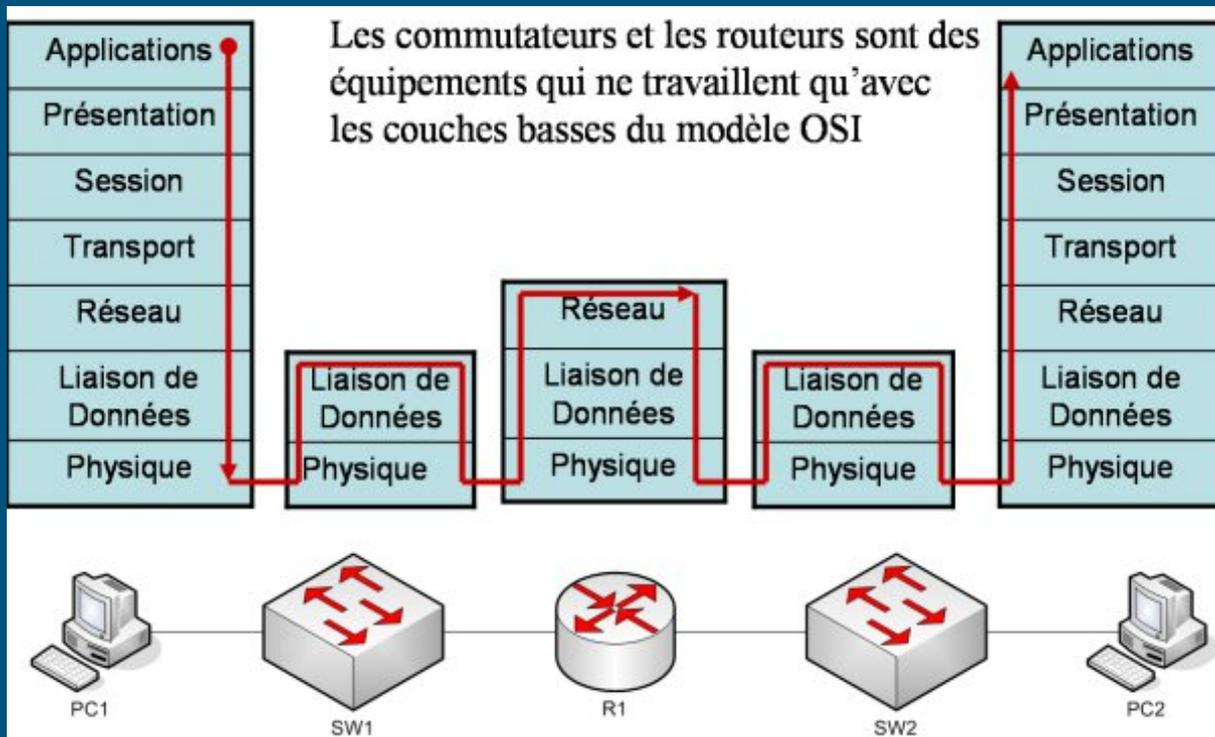
- Je dis à tous le monde mes liens/connections et leur état (up/down)
- Je dis à tous mes adresses sur chaque lien
- J'écoute les annonces des liens des autres
- Je construit une topologie complète
- Je calcule le chemin le plus court pour chaque préfixe d'adresse
- J'imagine que tout le monde à fait pareil et a le même résultat (la confiance règne)

RIP : Routing Information Protocol

Implémentation de l'algorithme de Bellman-Ford

- Je te dis toutes mes bonnes routes sa mère pour toutes les destinations que je connais et tu me racontes ta vie aussi
- Je construis une topologie simplifiée de mon point de vue
- Si une de tes routes est meilleure que la mienne, je l'utilise à la place
- Je balance l'info à mes voisins

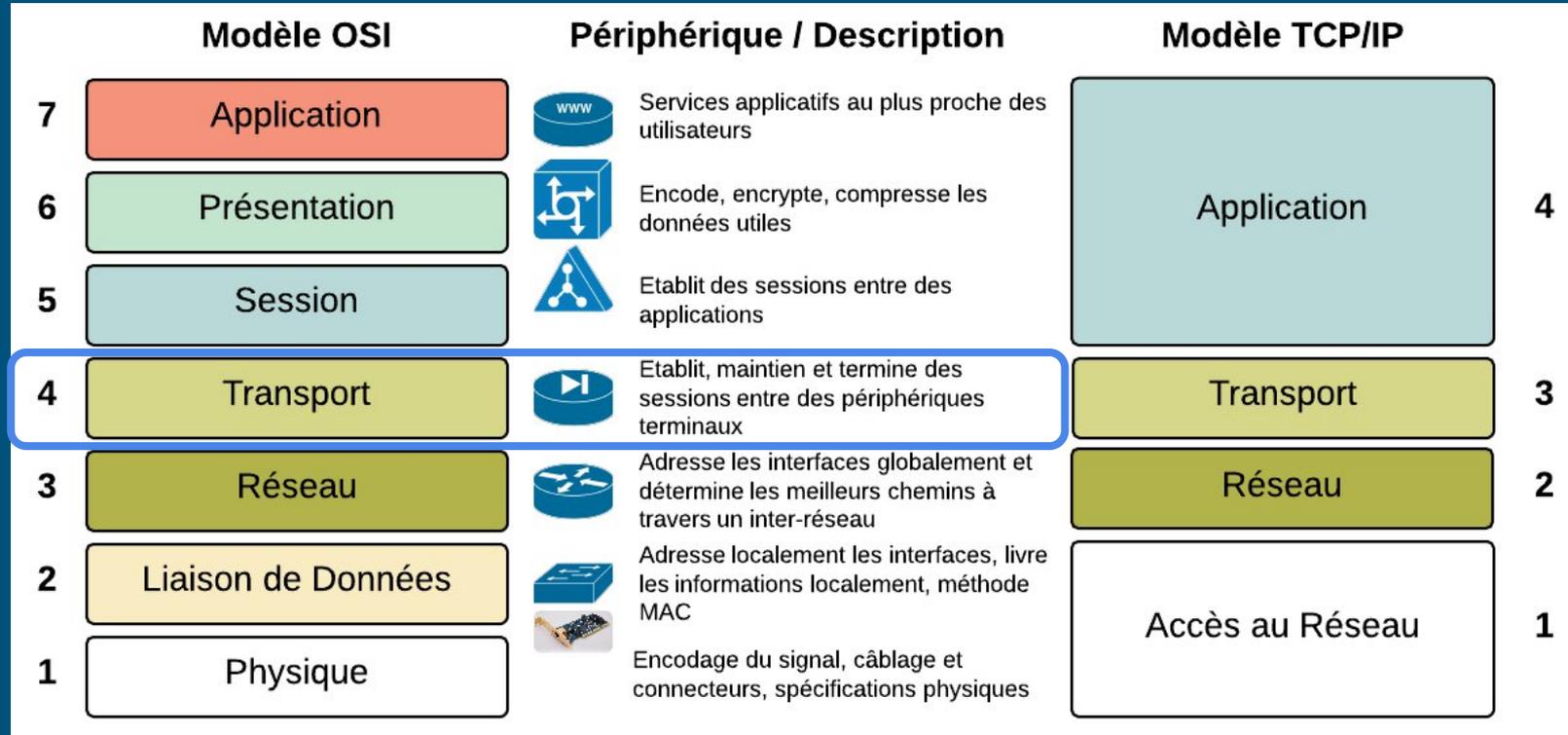
De bout en bout à travers le modèle OSI



- Routeurs : routent
forwardent en se
basant sur la couche
3 (réseau)

- Switchs : forwardent
en se basant sur la
couche 2 (réseau)

Couche 4 : Transport (TCP / UDP)



Couche 4 : Transport (TCP / UDP)

Architecture : Client / Serveur

TCP / UDP

Les ports et les services associés (les fameux sockets)

Pare-feu

IANA #BalanceTonPort

0 : Réserve ("joker" en développement réseau, tous les ports)

1 → 1023 : Ports privilégiés

1024 → 49151 : Ports alloués à des applications particulières

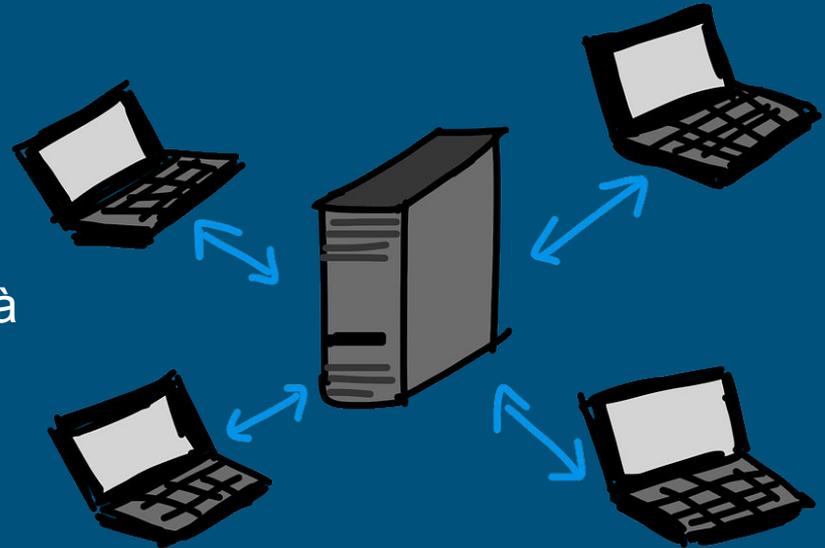
49152 → 65535 : Ports "éphémères" ou "aléatoires" utilisés comme source lors d'une connexion sortante

Architecture Client / Serveur

Client : Firefox, Chrome, Opéra...

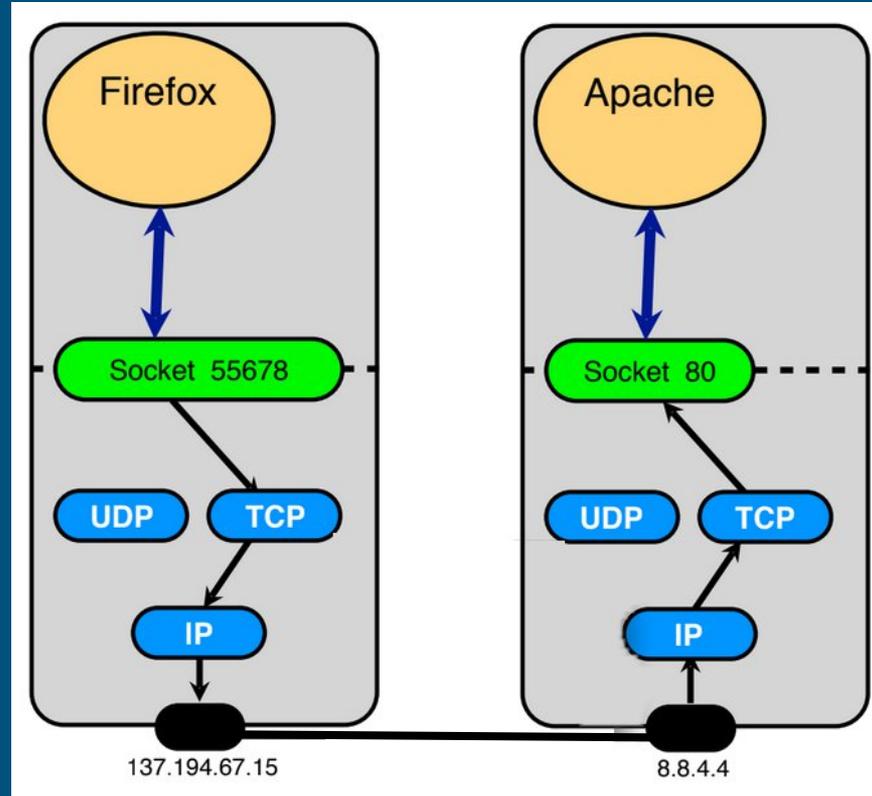
Serveur : Nginx, Apache2, IIS...

- Serveur web : port 80
- Client envoie une requête sur le port 80 à partir du port 12345 (> 49152)
- Le serveur répond depuis le 80 sur le 12345
- à chaque requête le client peut utiliser un port différent



Architecture Client / Serveur + Sockets

Client

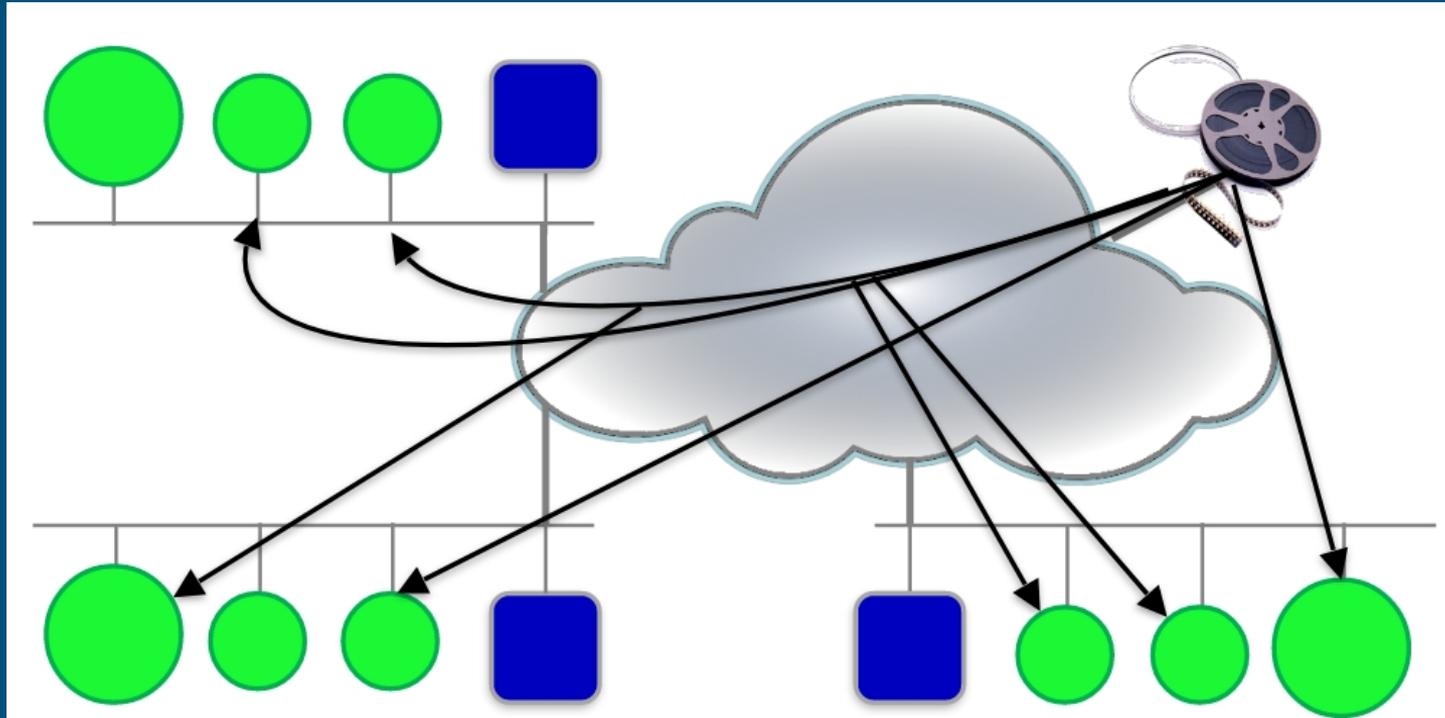


Serveur

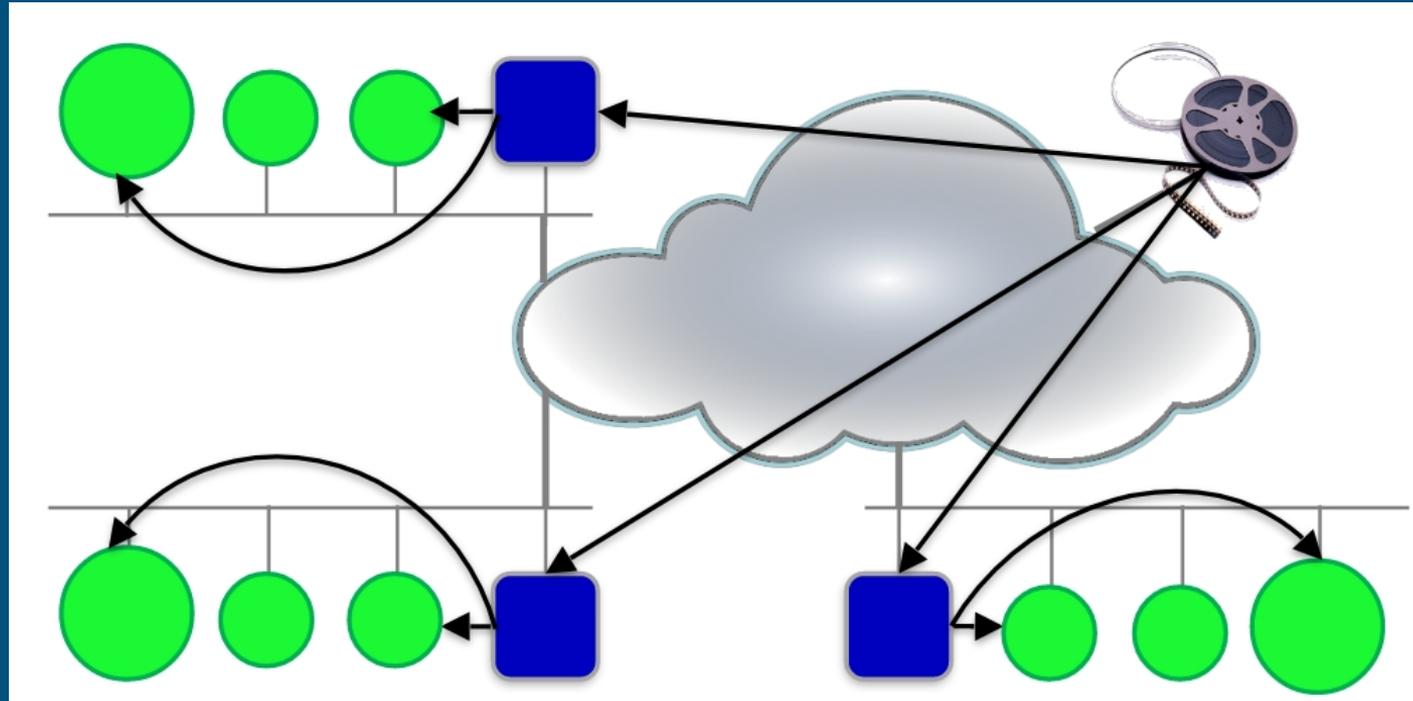
Architectures applicatives

- Client - serveur
- CDN (Content Distribution Network)
- IP Multicast
- Peer to Peer (P2P)

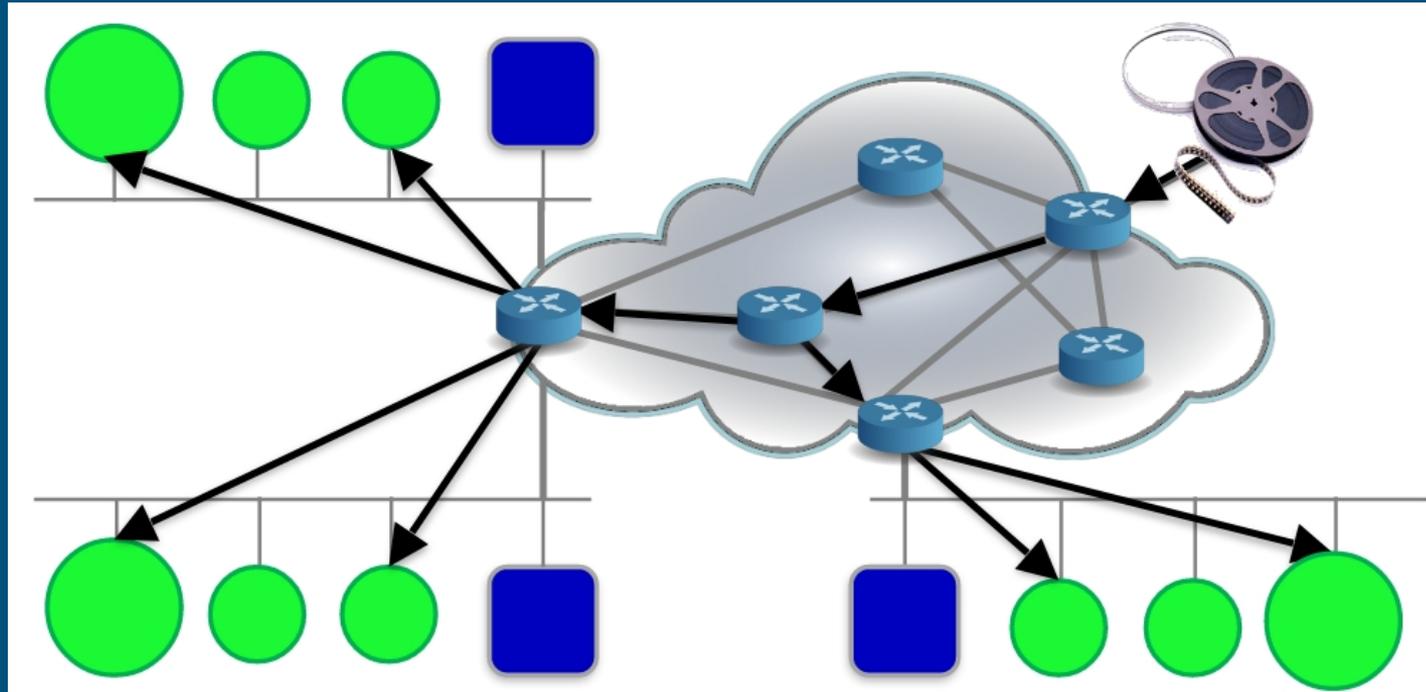
Client / Serveur



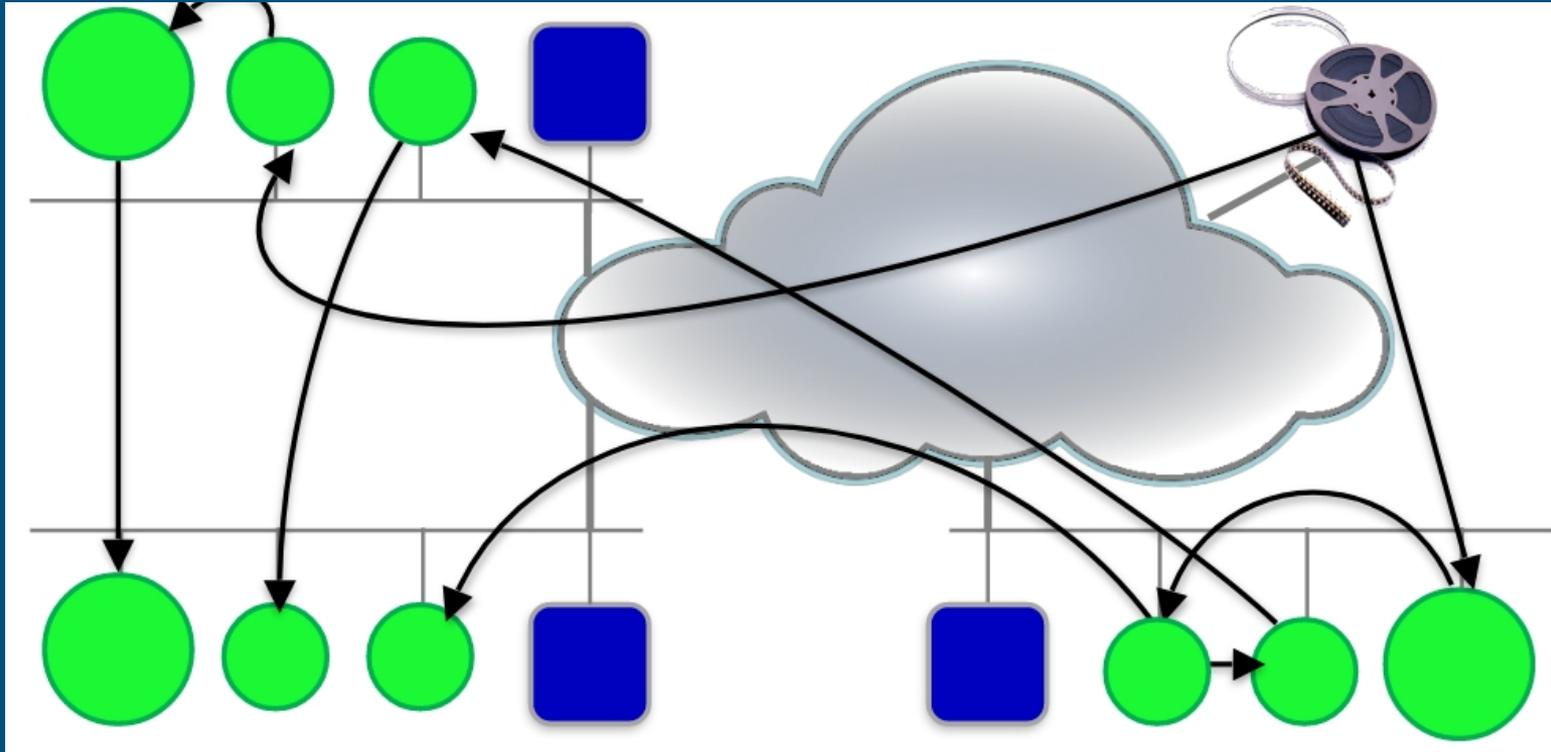
CDN (Content Delivery Network)



IP Multicast



P2P (Peer to Peer)



Pare-feu

Un service = un port

un ordinateur expose plusieurs services :

- FTP
- HTTP
- SSH

Pare-Feu bloque les ports en fonction de la source ou de la destination.

Protège et fait le tri grâce à des règles.

UDP : User Datagram Protocol

UDP Datagram Header Format

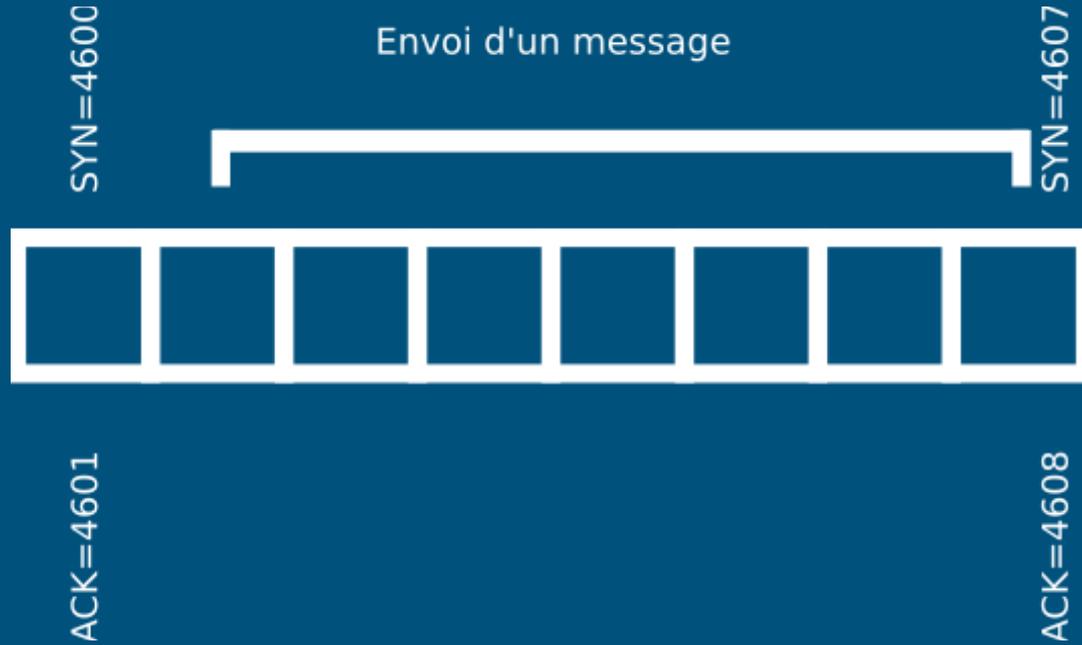
Bit #	0	7	8	15	16	23	24	31
0	Source Port				Destination Port			
32	Length				Header and Data Checksum			

TCP : Transport Control Protocol

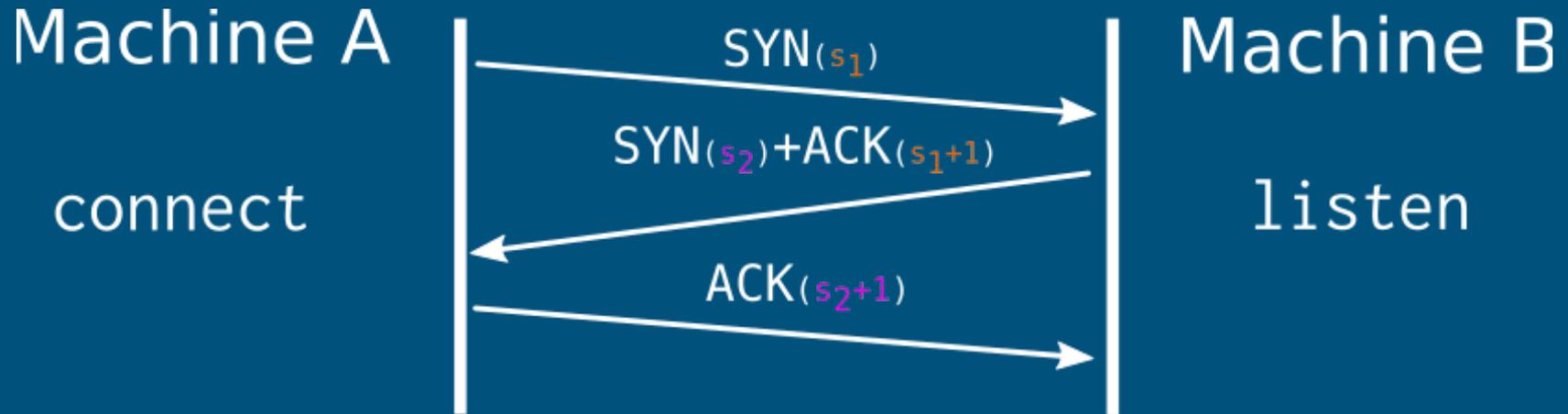
TCP Segment Header Format

Bit #	0	7	8	15	16	23	24	31
0	Source Port				Destination Port			
32	Sequence Number							
64	Acknowledgment Number							
96	Data Offset	Res	Flags			Window Size		
128	Header and Data Checksum				Urgent Pointer			
160...	Options							

TCP - Mécanisme SYN/ACK



Three-way handshake

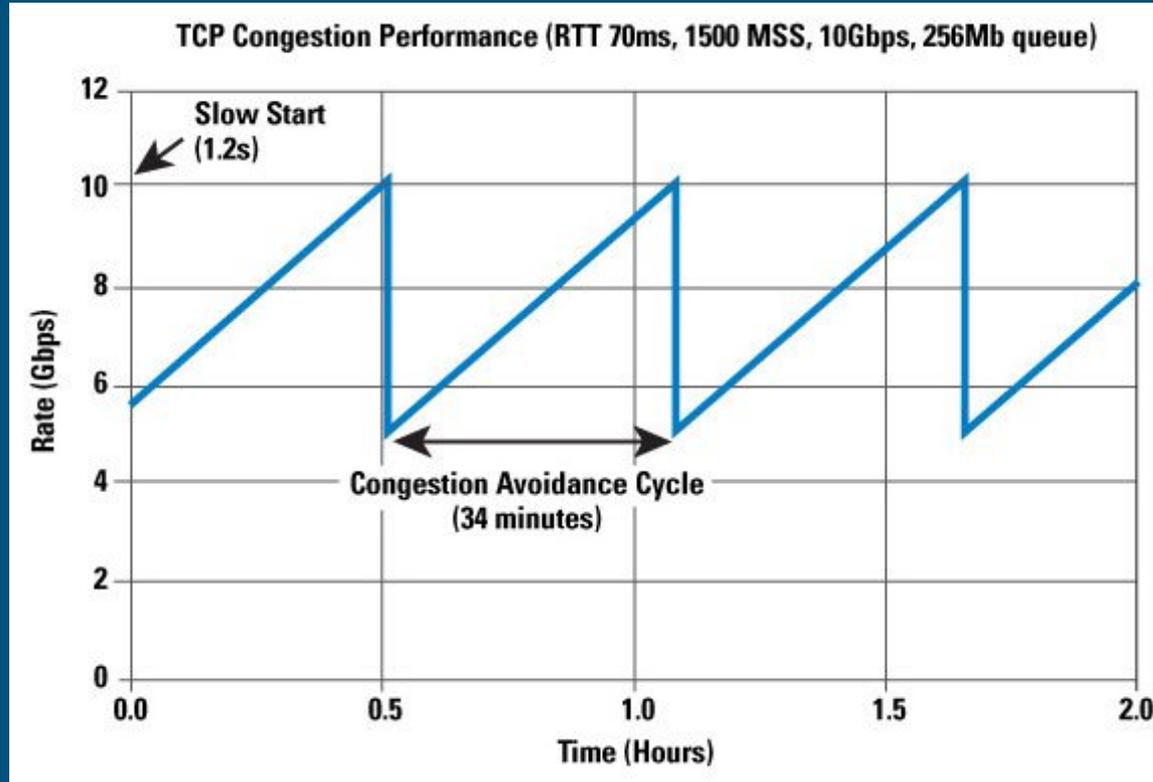


$\text{SYN} \rightarrow \text{SYN ACK} \rightarrow \text{ACK} \rightarrow \text{DATA} \rightarrow \text{ACK} \rightarrow \text{FIN} \rightarrow \text{FIN ACK}$

Fenêtre TCP



Augmentation de la fenêtre (AIMD)



TCP : gestion de flux et de congestion

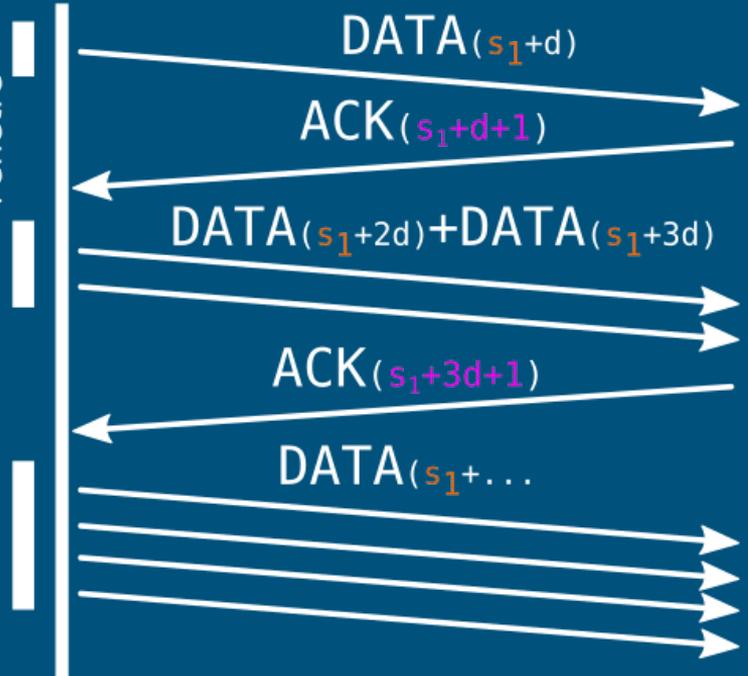
- Différentes techniques utilisées dans différents cas
- En pratique != Windows et Linux actuel
- Étendent les mécanismes d'ACK et de fenêtre glissante

TCP Tahoe : Slow start

Machine A

send

Fenêtre



Machine B

recv

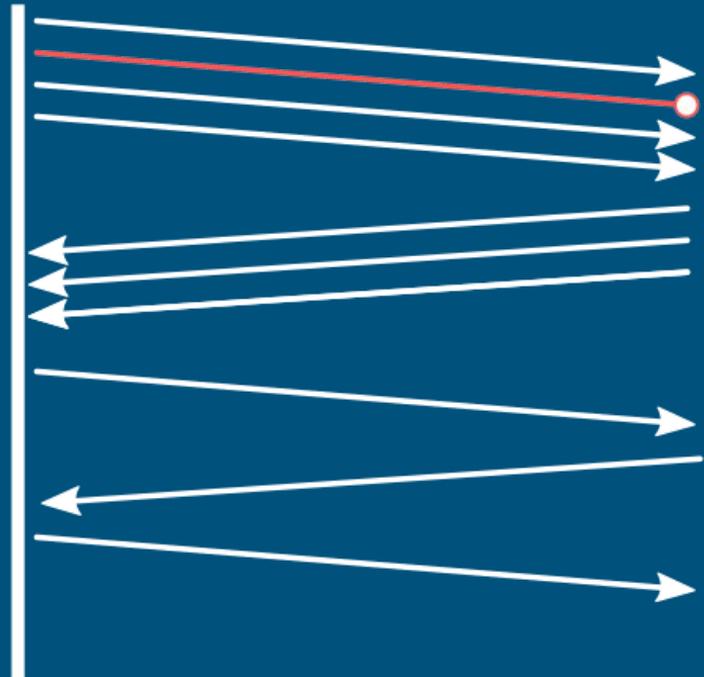
TCP Tahoe : Fast retransmit

Machine A

On envoie
1, 2, 3, 4

On envoie 5

FAST RETRANSMIT
4 ACK(2)
On retransmet 2



Machine B

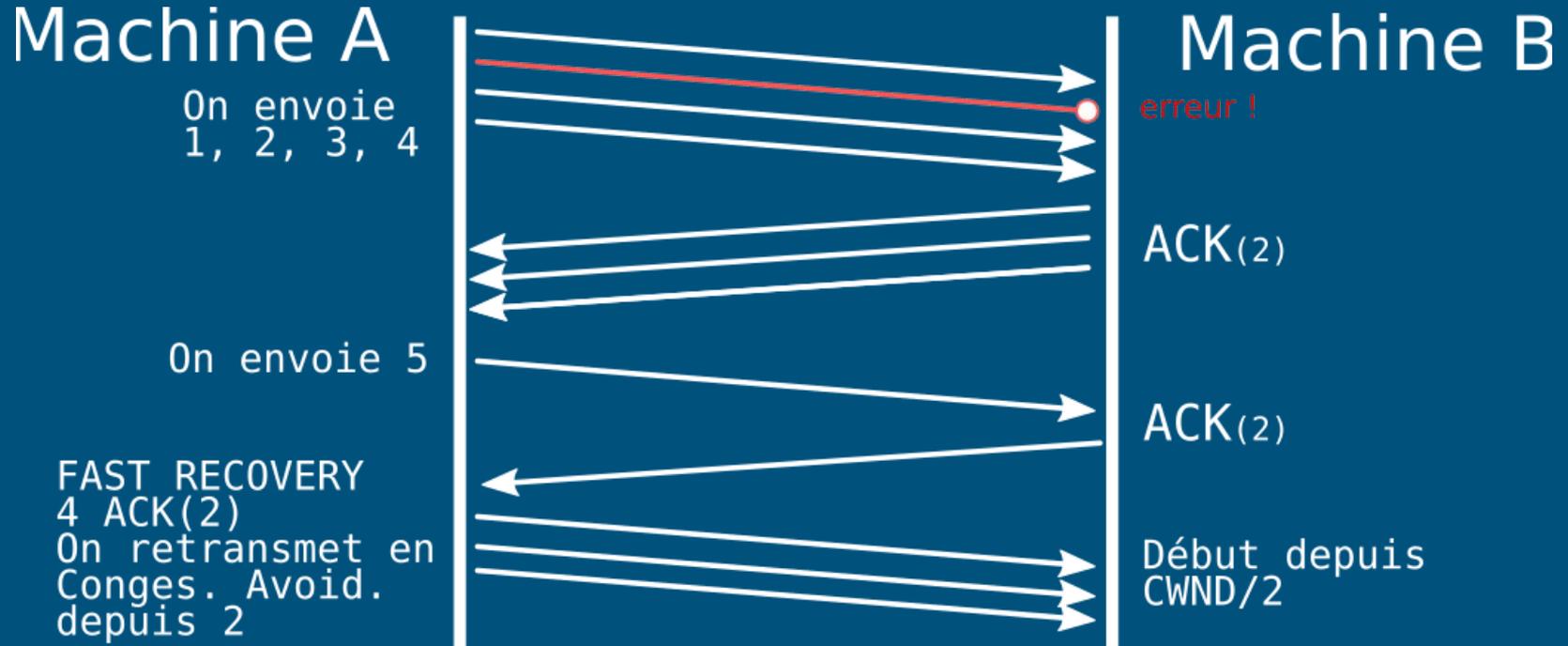
erreur !

ACK(2)

ACK(2)

Début du
slow start

TCP Reno: Fast recovery



La Mega formule à retenir

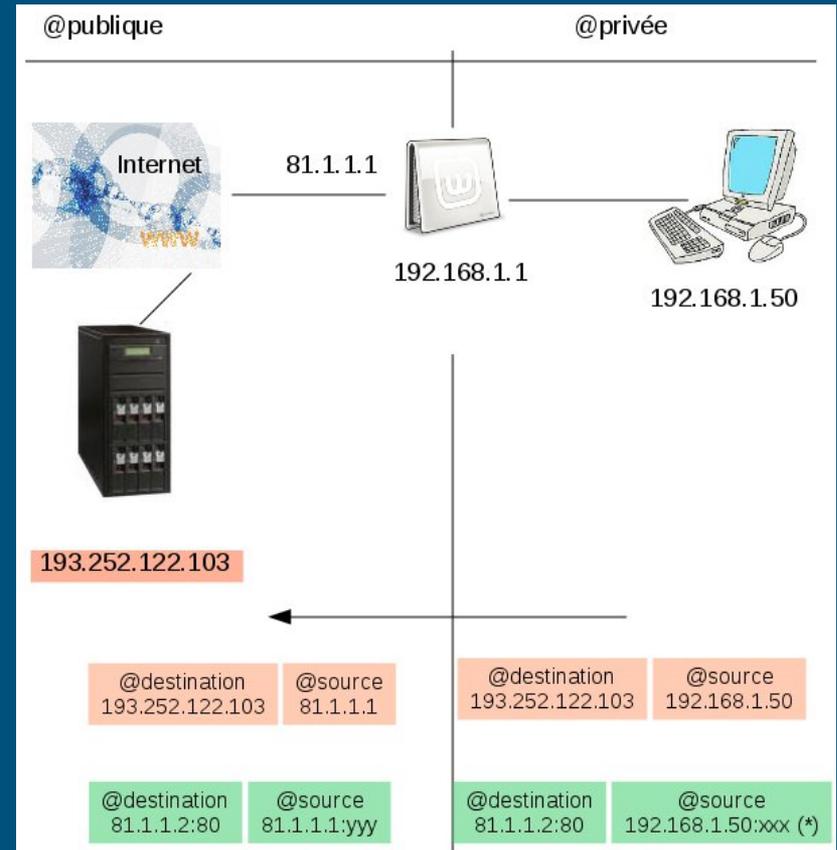
$$\sqrt{\frac{3}{2}} \times \frac{1}{\text{RTT} \sqrt{\rho}}$$

NAT - Network Address Translation

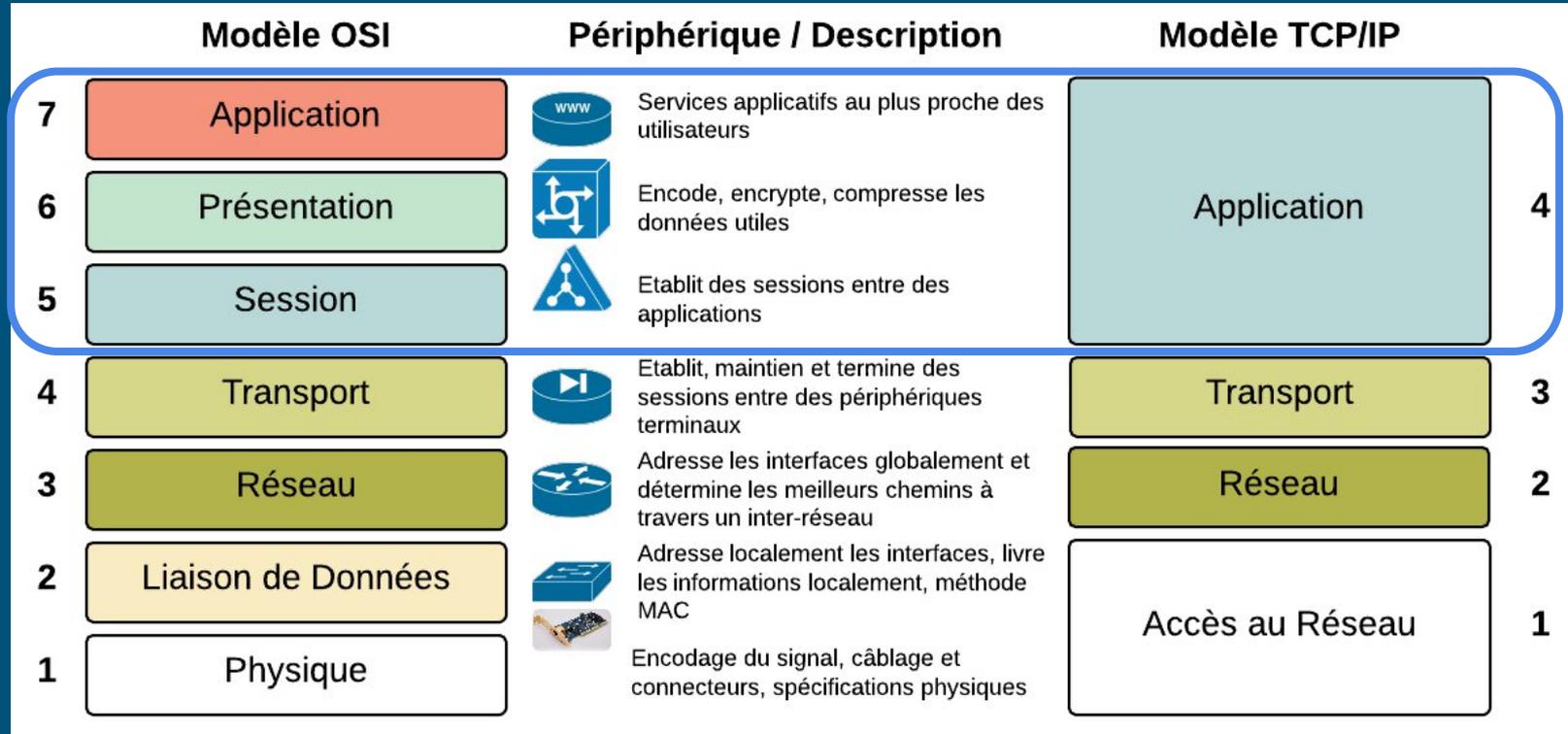
Internet est trop petit pour le nombre d'appareil qui veulent y accéder

=> On multiplexe plusieurs machines derrière la même IP en leur répartissant les ports TCP/UDP

(parfois plusieurs étages de NAT...)



Couches supérieures : Application : Service



DNS

Domaine Name Server : permet de traduire des noms de domaines en adresses IP et vice versa.

- Je veux accéder à www.telecom-paristech.fr, quelle est son adresse ?

```
www.telecom-paristech.fr is an alias for hermes.enst.fr.
```

```
hermes.enst.fr has address 137.194.52.7
```

```
hermes.enst.fr has IPv6 address 2001:660:330f:34::7
```

tout est stocké dans des tables.

DNS

Différents types d'entrées (Ressources Records) :

- NS (Name Server) Identifie le serveur de nom de la zone
 - telecom-paristech.fr -> ns6.enst.fr
- A : lie un nom de domaine à une adresse IPv4
 - vili.enst.fr -> 137.194.52.8
- AAAA : lie un nom de domaine à une adresse IPv6
- CNAME (Canonical Name) : Alias : lie un nom de domaine à un autre
 - www.telecom-paristech.fr -> vili.enst.fr
- MX (Mail eXchanger) : lie un nom de domaine à un serveur mail

DNS - fonctionnement

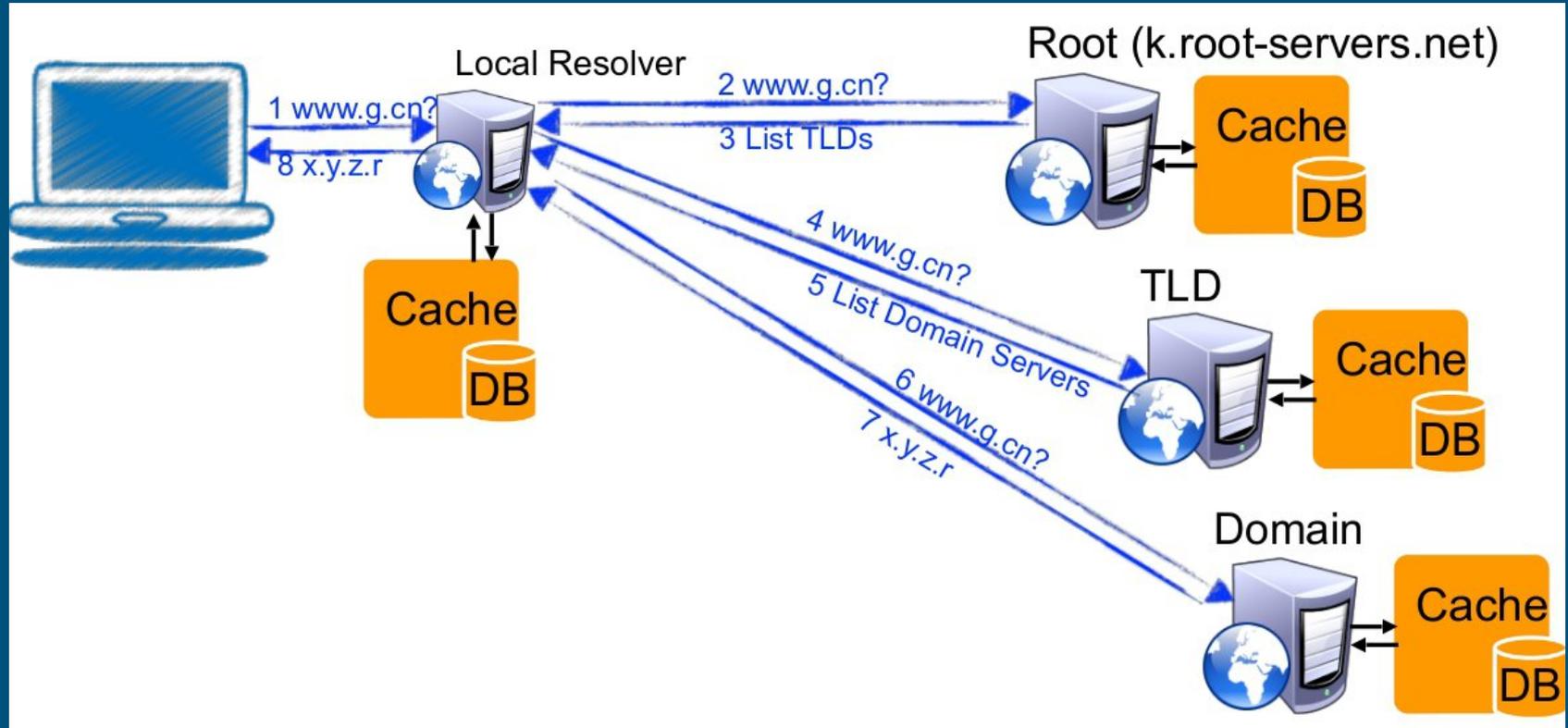
Plusieurs types de requêtes :

- Itératif
- Récursif
- Hybride

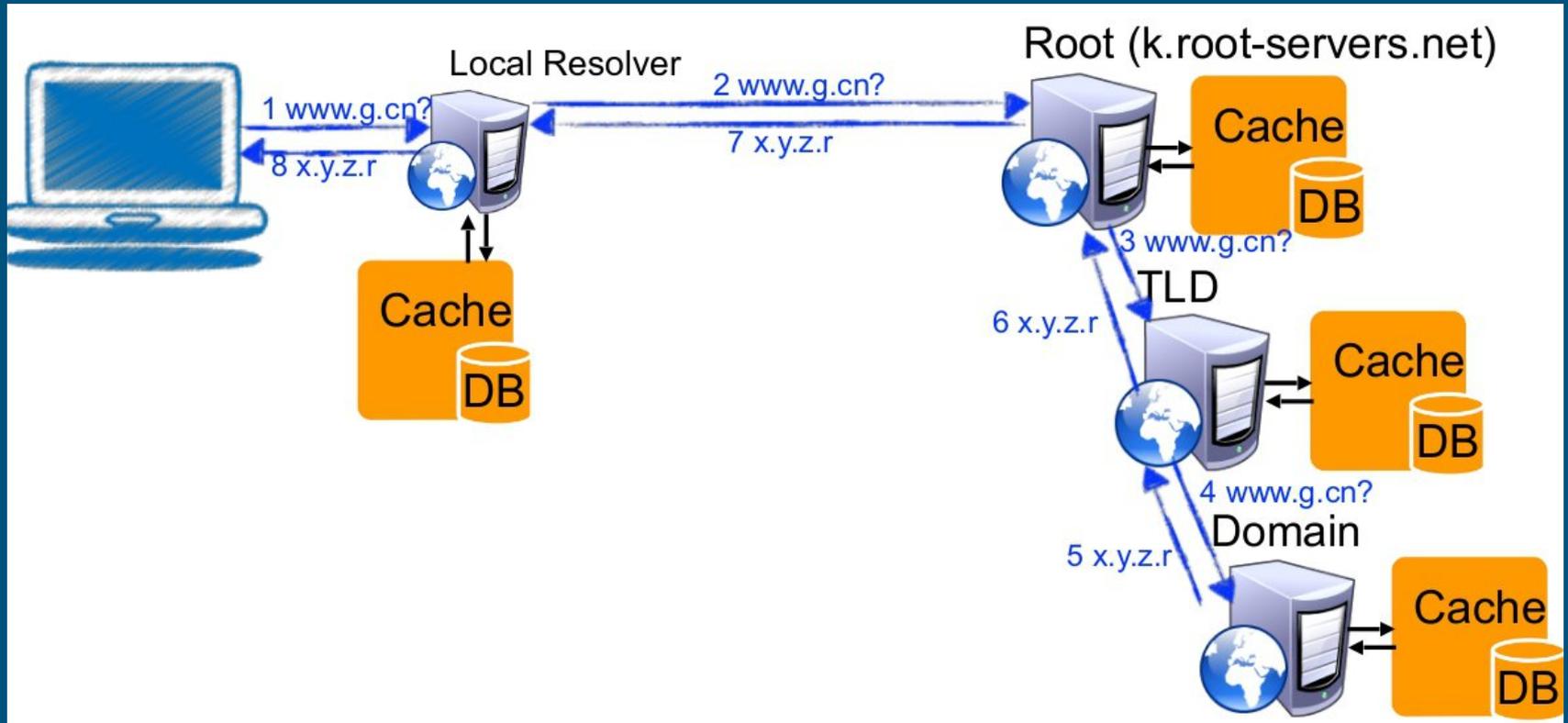
Sur un serveur :

- Cache

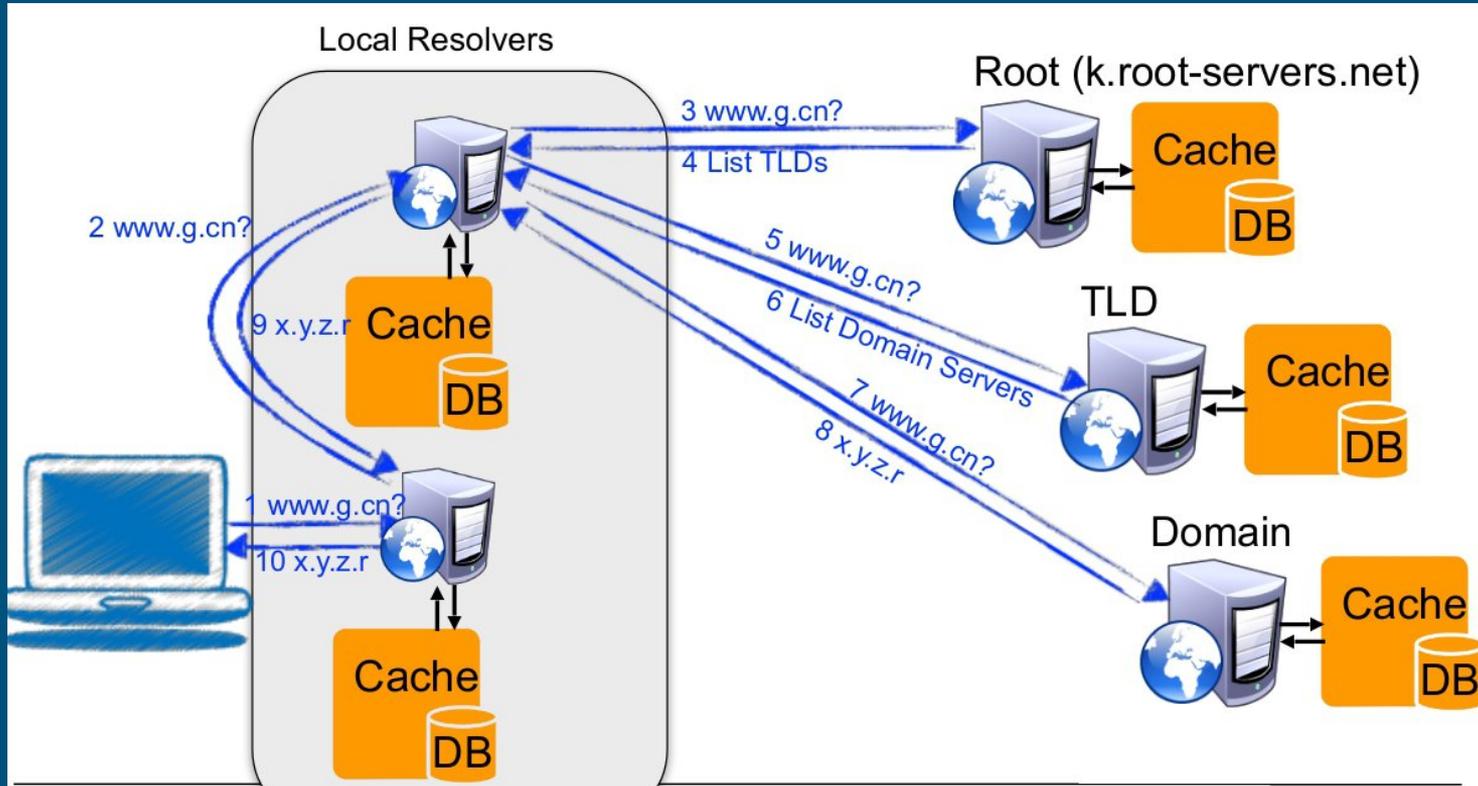
DNS - itératif



DNS - récursif



DNS - hybride



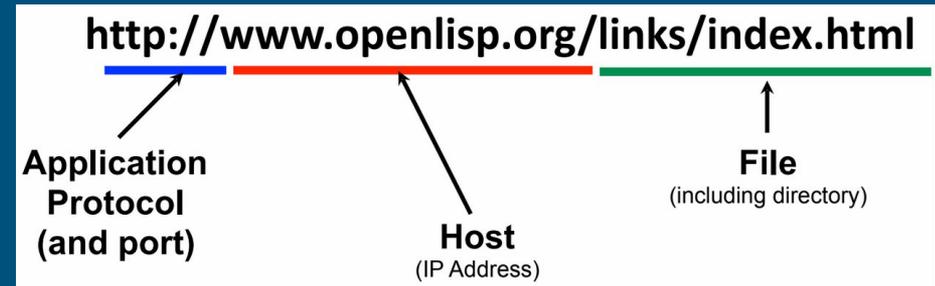
WWW : World Wide Web

différent de internet

Internet = web + chat + mails + etc.

HTTP / HTTPS

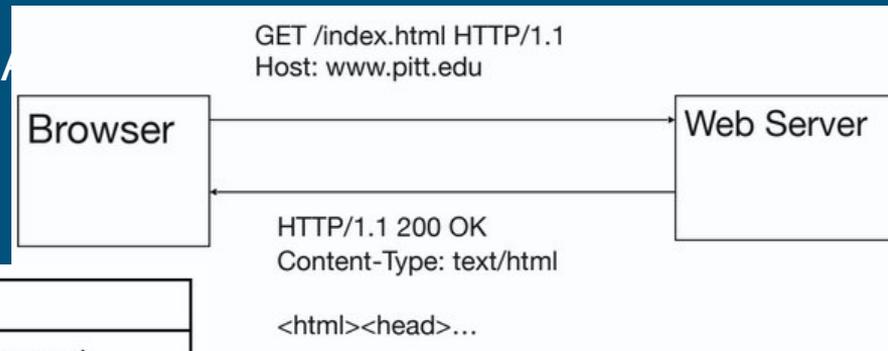
URL : Uniform Resource Locator



HTTP

Protocol sans état (pas de maintien d'état coté serveur)

Verbes : GET, PUT, POST, DELETE, UPDATE



Code	Meaning	Examples
1xx	Information	100 = server agrees to handle client's request
2xx	Success	200 = request succeeded; 204 = no content present
3xx	Redirection	301 = page moved; 304 = cached page still valid
4xx	Client error	403 = forbidden page; 404 = page not found
5xx	Server error	500 = internal server error; 503 = try again later

HTTP

Plusieurs versions :

- HTTP/1.0
- HTTP/1.1
- HTTP/2.0

Status	Method	Domain	File
	GET	www.gstatic.com	 createjs-2015.11.26.min.js
200	GET	 ssl.gstatic.com	 i1_1967ca6a.png
	GET	www.google.com	 rs=ACT90oFPDA_ht-Zifqk48ExFDewlFTWLKw
200	GET	 www.google.com	 favicon.ico

Mode d'architecture REST

Representational State Transfer : échanges sans états (stateless) se basant sur un ensemble d'opérations prédéfinies (ex : GET, POST, DELETE...)

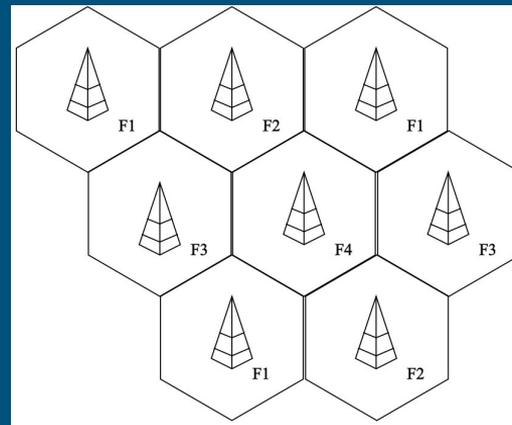
Formats utilisés : JSON, XML, HTML

“Restful” API

Réseaux sans fil

Wi-Fi : Wireless Fidelity

3G / 4G : se base sur des “cellules”



- problématique de division du temps et de la fréquence
- problématique du changement de cellule
- problématique de débit et de latence (outdoor, indoor, incar)
- UMTS, EDGE, W-CDMA, HSPA (3,5G), HSPA+ (3,75G)
- LTE (4G)

QoS / QoE

Paramètres de la Qualité de Service :

- Vitesse de transfert
- Taux d'erreur binaire, Taux d'erreur de paquet
- Disponibilité
- Délais de Propagation

Paramètres de la Qualité d'Expérience :

- Le client est content

Métriques : dimensionnement / benchmarks

- **Bandwidth** (bande passante) : sur un lien physique, débit "instantané" (dépend du bruit sur le lien, de la quantité de bits de parité, etc.)
- **Throughput** (débit) : vu depuis une app, débit mesuré (moyenné, dépend de la charge du lien, de la charge des routeurs, etc.)
- **Latence** (latency) : temps entre envoi et réception (dépend de la charge des équipements, de la qualité du lien, etc.)
- **Round Trip Time** (RTT) : temps aller + retour (le routage est ASYMÉTRIQUE sur Internet : pas aller x2)
- **Gigue** (jitter) : variation dans les timing de paquets
=> "Please wait while your video is *buffering*..."

Question Bonus : Les organismes

ICANN : Internet Corporation for Assigned Names and Numbers

IANA : Internet Assigned Numbers Authority (département de l'ICANN)

IETF : Internet Engineering Task Force

RIPE : registre régional d'adresses IP. Dessert l'Europe et une partie de l'Asie.

Question Bonus : Standard ou norme ?

Standard :

- Est un ensemble de recommandations ou de préférences préconisées par un groupe d'utilisateurs caractéristiques et avisés.

Norme :

- Est un ensemble de règles de conformité ou de fonctionnement légiféré par un organisme de normalisation mandaté, comme l'ISO au niveau international.

Merci de votre attention

Des questions ? maintenant ou par mail : rez404@rezel.net

Sinon on passe aux annales le Lundi 25 Mai 2020 de 19h à 20+h