

# ***Introduction au Web***

*D'après les cours de Andrea G. B. Tettamanzi*

---



**Fabien Givors**

Université de Nice Sophia Antipolis

Département Informatique

fabien.givors@unice.fr

*CM - Séance 1*

# **Le Web et ses protocoles**

# *Plan*

- Présentation du cours
- Qu'est-ce que c'est que le Web
- Hypertextes
- Notions de bases sur Internet
- Protocole HTTP

# *Fonctionnement*

- Structure de l'UE
  - HTML + CSS
  - 2 séances CM → 3 séances TD sur machine → TD noté
- Contrôle des connaissances
  - TD noté sur HTML (30%)
  - TD noté sur CSS (30%)
  - Contrôle final écrit (40%)

# *Ce que vous apprendrez*

- D'un point de vue pratique
  - Suivre la définition de langage
  - Structurer / Concevoir
- A faire des pages webs
  - html « statiques »
  - CSS
  - Applications pour la gestion des pages en ligne (blog, etc.)

# *Ce que vous n'apprendrez pas*

- Pas de génération dynamique (côté serveur)
- Pas d'interaction dynamique (côté client)
- Pas de spécificités HTML 5
  - Pas encore totalement fini
  - Peu de différences pour ce qui nous concerne

# Matériel

- Page Web du cours :
  - <http://www.i3s.unice.fr/~fgivors/>
- Définition officielle des standards :
  - <http://www.w3.org/standards/webdesign/htmlcss>

# Le Web et Internet



- WWW = World-Wide Web (Toile d'araignée mondiale)
  - Communément appelé le Web ou la Toile
- Un système **hypertexte** public basé sur **Internet**
- Créé au CERN par **Tim Berners-Lee** en 1990
- Idée initiale : créer un système hypertexte distribué sur le réseau informatique pour que les collaborateurs puissent partager les informations au sein du CERN
- Le 30 avril 1993, le CERN verse dans le domaine public toutes les technologies développées autour du WWW
- NCSA Mosaic : premier « navigateur »
- Comprendre le Web c'est comprendre ses standards

# Hypertexte



- Hypertexte = un texte doté de liens qui permettent de sauter immédiatement d'une partie à l'autre
- En 1945 l'ingénieur et conseiller scientifique américain **Vannevar Bush** publie sur Atlantic Monthly l'article « *As We May Think* »
- Dans les années 1960 :
  - Doug Engelbart crée le prototype NLS (oN-Line System), qui permet de rédiger et parcourir un hypertexte
  - Ted Nelson invente le terme « hypertexte »

# Systeme hypertexte



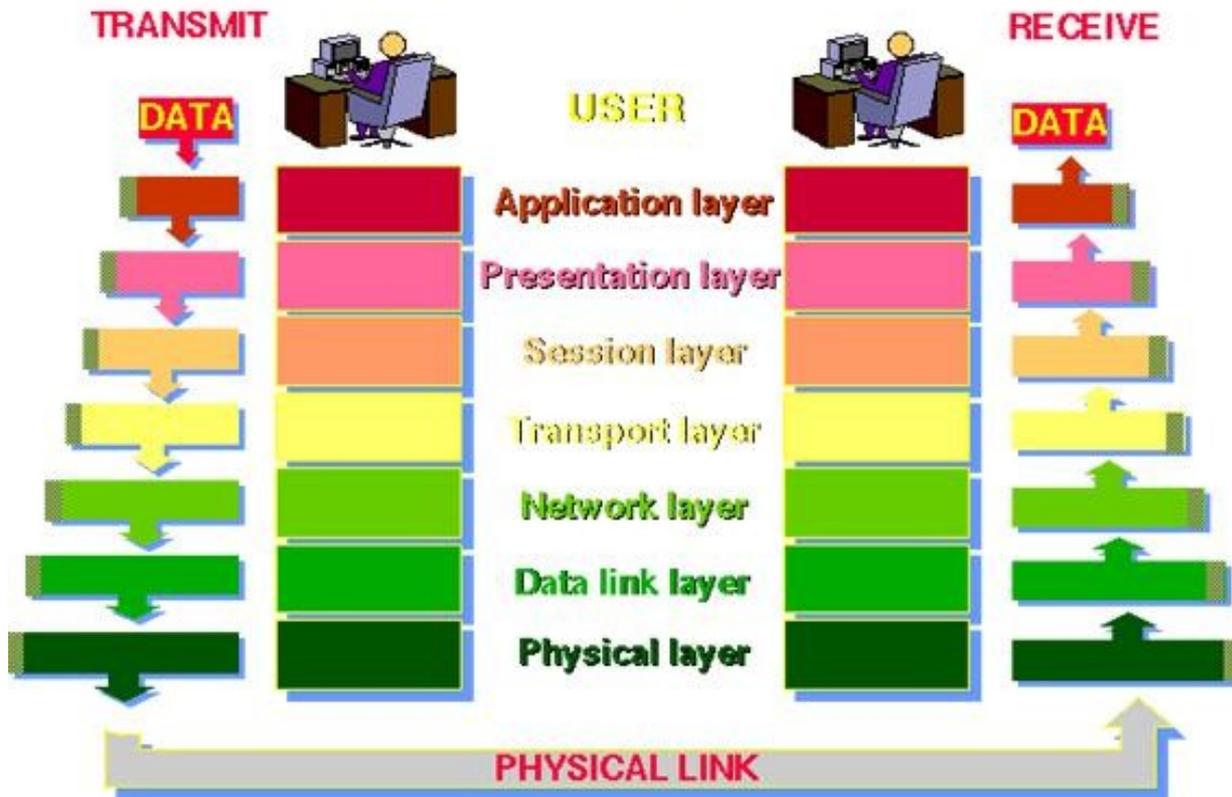
- Ensemble de nœuds liés entre eux par des hyperliens permettant de passer automatiquement d'un nœud à l'autre
- Nœud = unité minimale d'information, partie du texte
- Les liens entre les parties du texte sont gérés par ordinateur
  - Accès à l'information de manière associative
  - Navigation non linéaire, personnalisée
- Lorsque les nœuds sont aussi audiovisuels on peut parler de système hypermédia
- **Ted Nelson** : « *Let me introduce the word 'hypertext' to mean a body of written or pictorial material interconnected in such a complex way that it could not conveniently be presented or represented on paper* » (Proc. 20th ACM Nat'l Conf, 1965).

# Internet

- Internet = inter-réseau, réseau de réseaux
- Système d'interconnexion de machines
  - réseau informatique mondial
  - ensemble standardisé de protocoles de transfert de données
  - sans centre névralgique, robuste en cas d'attaque/panne
- Issu d'un projet de la DARPA : ARPANET (1968)
  - Conception d'un réseau militaire de communication à paquets
  - Les premiers nœuds installés dans des universités
- IP (Internet Protocol)
- Protocoles de transport: TCP, UDP

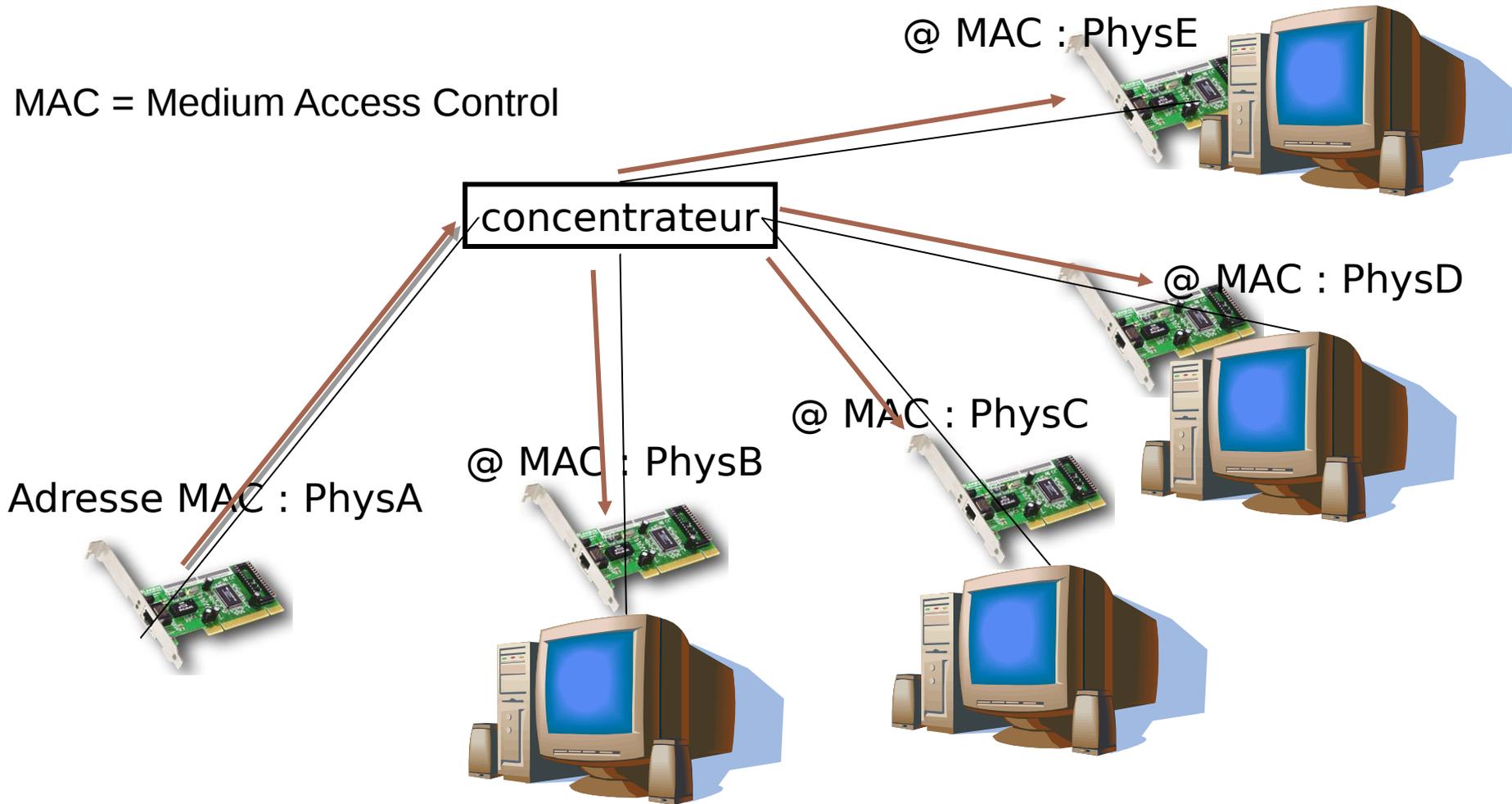
# Architecture à niveaux

## THE 7 LAYERS OF OSI

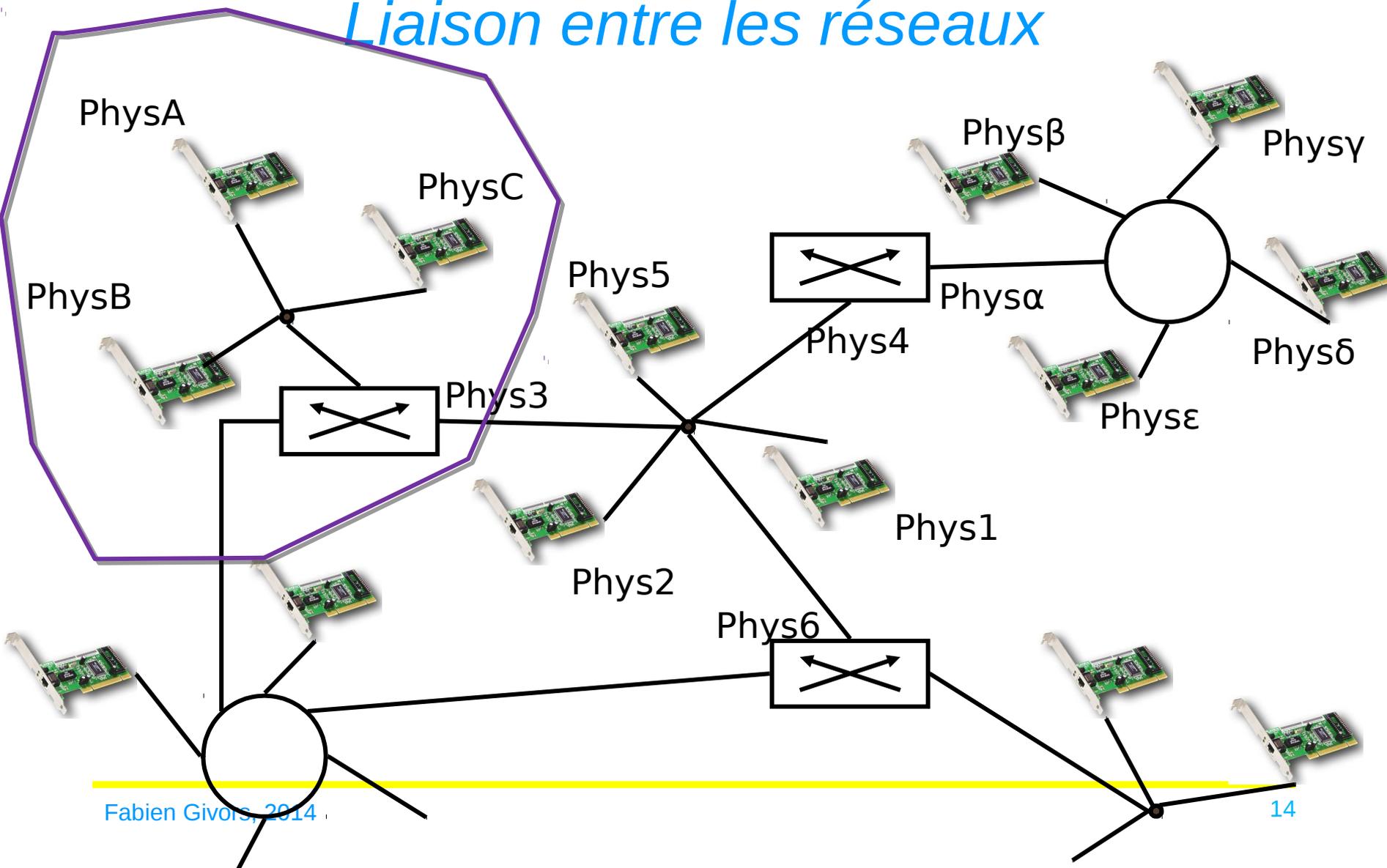


# Réseau local (LAN)

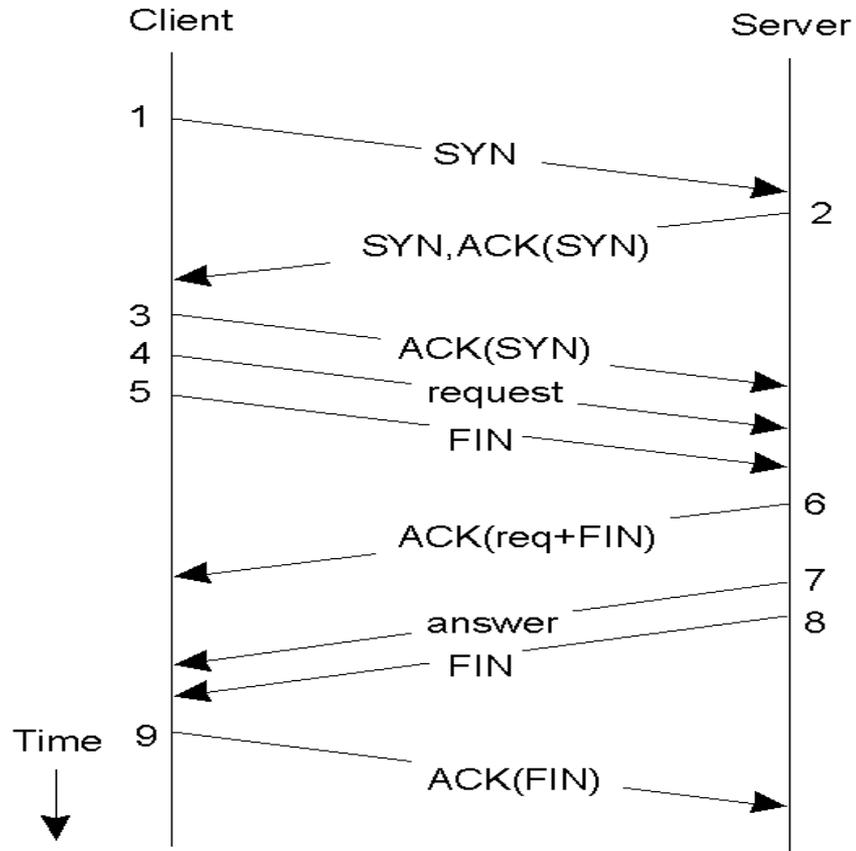
MAC = Medium Access Control



# Liaison entre les réseaux



# TCP (Niveau transport)



(a)

# Adressage

- But : fournir un service de communication universel permettant à toute machine de communiquer avec toute autre machine de l'interconnexion
- Une machine doit être accessible aussi bien par des humains que par d'autres machines
- Une machine doit pouvoir être identifiée par :
  - une adresse = identificateur universel d'un point d'accès,
  - un nom (mnémonique, pour les utilisateurs),
  - une route précisant comment la machine peut être atteinte.

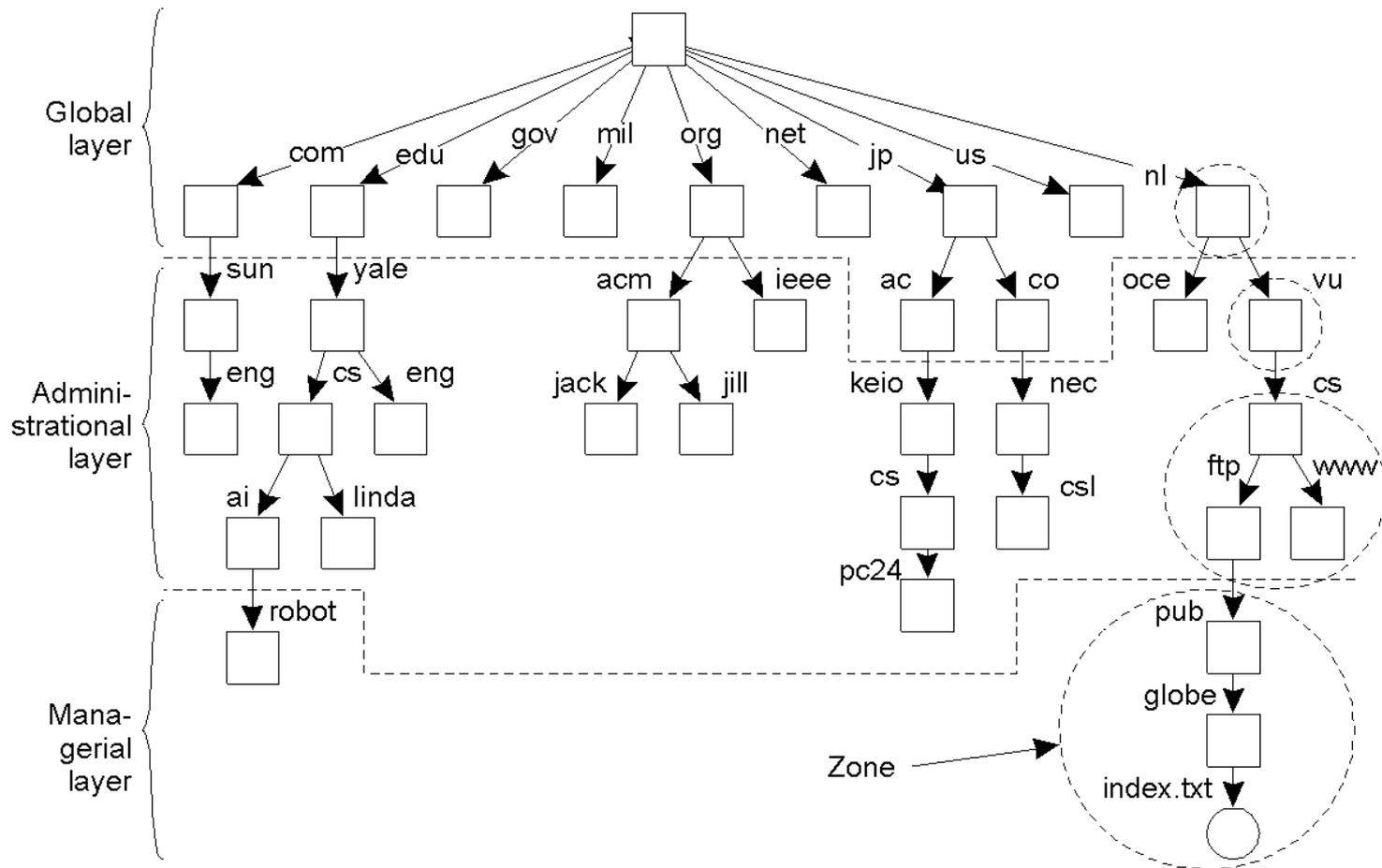
# Adresse IP

- IP version 4 : chaque interface réseau est identifiée par une adresse de 32 bit
- Cette adresse a une structure interne de type hiérarchique :
  - Les bits les plus significatifs identifient un sous-réseau
  - Les bits les moins significatifs identifient une interface dans le sous-réseau
- En notation décimale, lisible aux humains :
  - quatre entiers décimaux (de 0 à 255) séparés par un point
  - chaque entier représentant un octet de l'adresse IP
  - Ex. : 11000000 00001011 00000110 00011111  
est écrit 192.11.7.31
- Limitation : passage à IP version 6

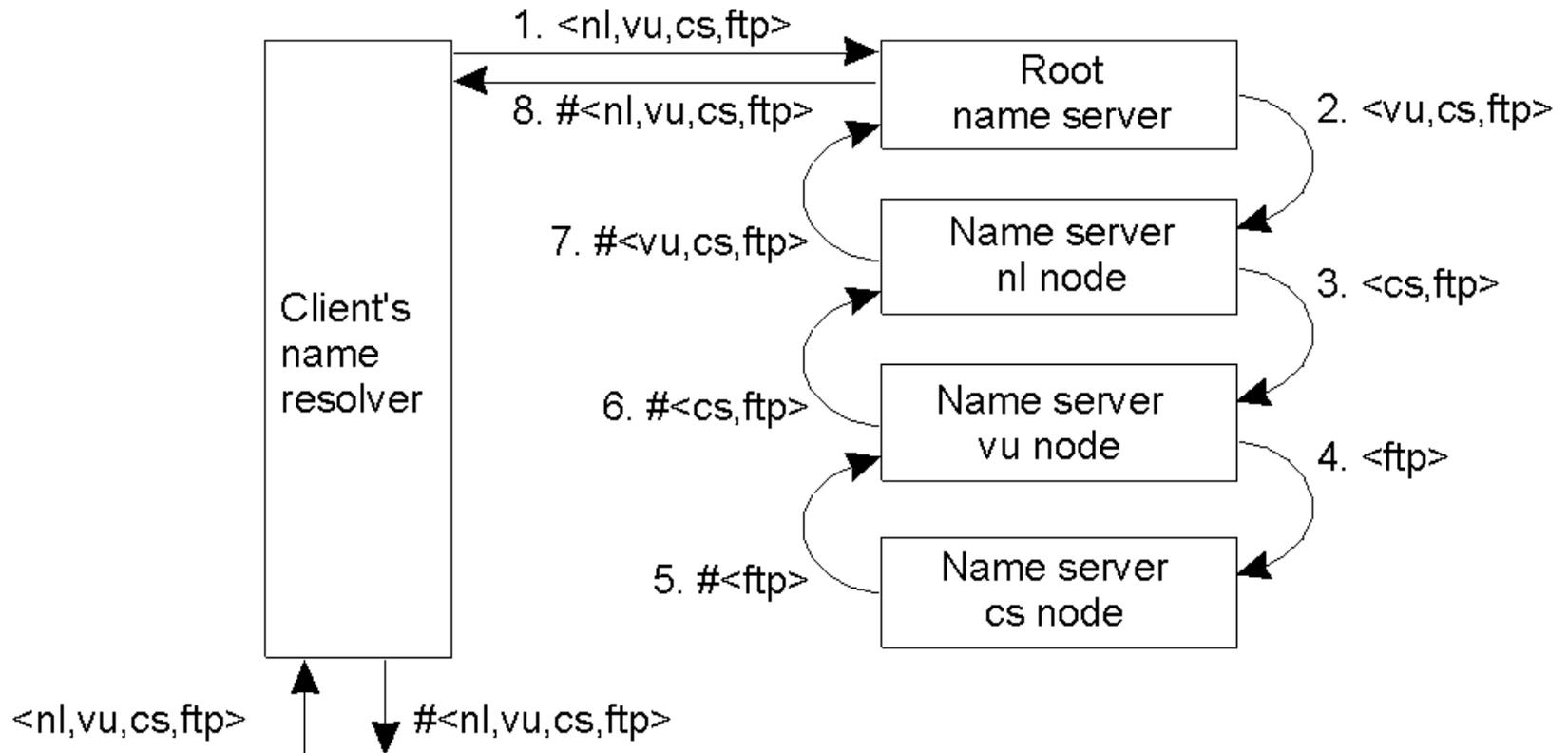
# Domain Name Server : Besoins

- Internet constitué de sous-réseaux (millions)
- IP permet l'accès aux machines par leur adresse IP
- Pratiquement impossible aux humains de connaître les adresses IP des machines auxquelles ils veulent accéder
- Le système DNS permet d'identifier une machine par un (des) nom(s) structuré(s) représentatif(s) de la machine et du (des) réseau(x) sur le(les)quel(s) elle se trouve ; exemple :
  - deptinfo.unice.fr
- Le système est mis en œuvre par une base de données distribuée au niveau mondial
- Les noms sont gérés par un organisme mondial : l'InterNIC et les organismes délégués : RIPE, NIC France, NIC Angleterre, etc.

# Espace de noms distribué (1)

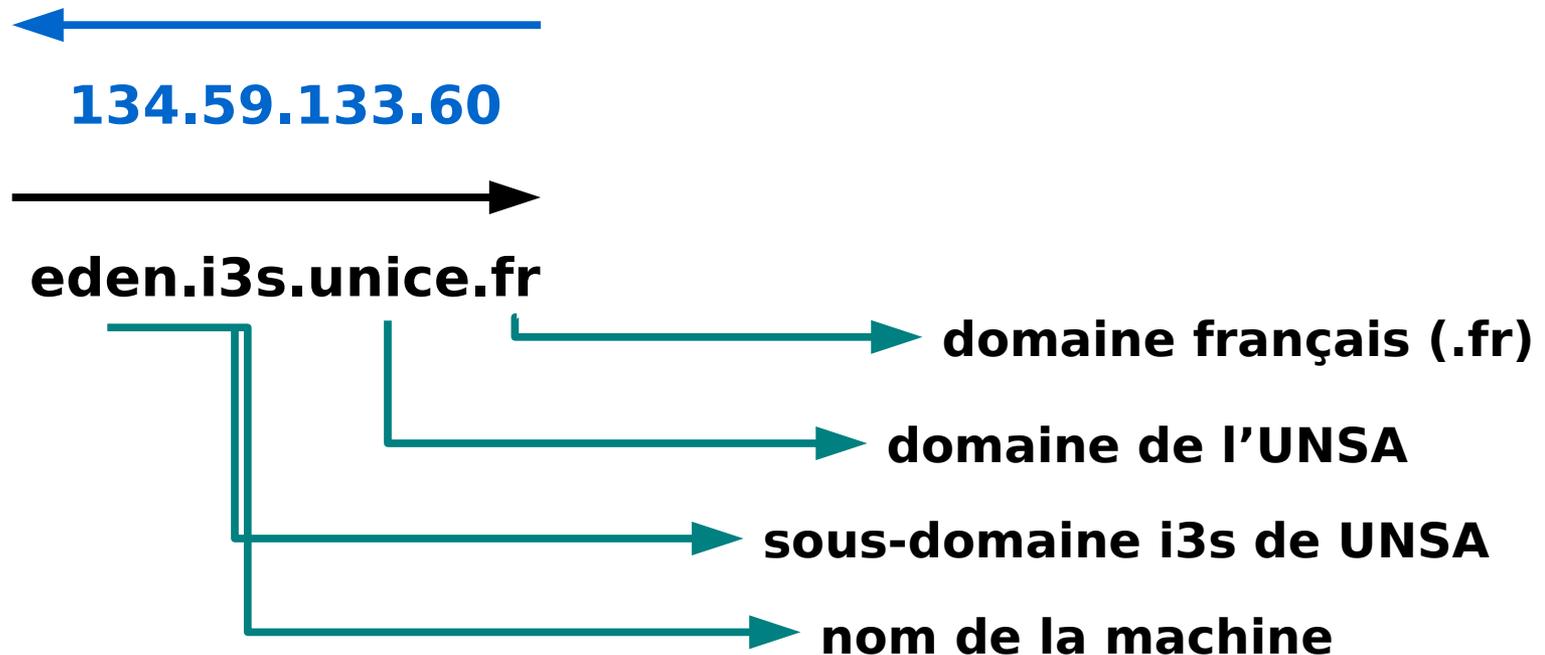


# Résolution d'un nom



# Structure d'une adresse IP symbolique

- A l'inverse de l'adressage IP numérique, la partie la plus significative se situe à gauche de la syntaxe :



# Protocole HTTP

- HTTP = Hypertext Transfer Protocol
- Fonctionne en dessus des protocoles TCP et IP
- Permet aux serveurs Web de transmettre les pages aux clients
  - Serveur : hôte qui contient des pages Web
  - Client : navigateur
- Un serveur HTTP est réalisé par un programme (ex. : apache2) qui tourne sur une machine hôte
- Le navigateur est un programme (ex. : Mozilla Firefox) qui tourne sur la machine de l'utilisateur
- Une page Web est un document

# *Désigner un document sur le Web : URL*

- URL = Uniform Resource Locator
- Une URL identifie
  - l'endroit où une ressource est stockée
  - le protocole pour y accéder
- Une ressource peut-être stockée à plusieurs endroits, elle possède donc une ou plusieurs URL
- Une ressource peut être un répertoire, un document, ou une partie d'un document, une image, un fichier audio/vidéo, etc...

# Format d'une URL

Protocole://	Hôte	[:Port]	Chemin	Nom	[#Ancre]	[?Paramètres]
--------------	------	---------	--------	-----	----------	---------------

◆ Exemple 1 : une ressource statique est un document HTML

http://	deptinfo.unice.fr		/~stouati/	L1/	#sec3	
---------	-------------------	--	------------	-----	-------	--

◆ Exemple 2 : une ressource dynamique générée avec des paramètres

http://	iihm.imag.fr		/cgi-bin/Vitesse2/	vitesse2.bat		? Keywords=unsa&SearchEngine=Google&Kind=Search&InfoSpace=&MaxInfoNumber=100&VitesseMode=Win
---------	--------------	--	--------------------	--------------	--	---

# Codage d'une URL

- Les parties d'une URL sont des séquences de caractères alphanumériques plus les deux caractères – et \_
- Un espace blanc est normalement remplacé par un +
- Séquences d'échappement pour représenter des caractères spéciaux :
  - %xx, où xx est le code hexadécimal du caractère en ASCII
  - %20 = espace
  - %7E = ~
  - %2B = +
  - %25 = %
  - Etc.

# Protocole HTTP

- HTTP est protocole le plus utilisé sur Internet depuis 1990.
- Version 0.9 : uniquement destinée à transférer des données sur Internet (en particulier des pages Web écrites en HTML).
- Version 1.x (les plus utilisées) : permet de préciser des en-têtes décrivant le contenu du message en utilisant un codage de type MIME et de spécifier le nom d'hôte pour les serveurs hébergeant plusieurs sites.
- Transfert de fichiers identifiés grâce à une URL entre un navigateur (le client) et un serveur Web (apache2).
  - Le navigateur effectue une requête HTTP
  - Le serveur traite la requête puis envoie une réponse HTTP
- RFC 2616 - Hypertext Transfer Protocol -- HTTP/1.1

# Commandes HTTP

Commande	Description
GET	<b>Requête de la ressource située à l'URL spécifiée</b>
HEAD	Requête de l'en-tête de la ressource située à l'URL spécifiée
POST	Envoi de données au programme situé à l'URL spécifiée
PUT	Envoi de données à l'URL spécifiée
DELETE	Suppression de la ressource située à l'URL spécifiée

# Réponse HTTP

- Syntaxe

```
VERSION-HTTP CODE EXPLICATION<crLf>
```

```
EN-TETE : Valeur<crLf>
```

```
...
```

```
EN-TETE : Valeur<crLf>
```

```
Ligne vide<crLf>
```

```
CORPS DE LA REPONSE
```

- Exemple de réponse HTTP:

```
HTTP/1.0 200 OK
```

```
Date : Sat, 15 Jan 2000 14:37:12 GMT
```

```
Server : Microsoft-IIS/2.0
```

```
Content-Type : text/HTML
```

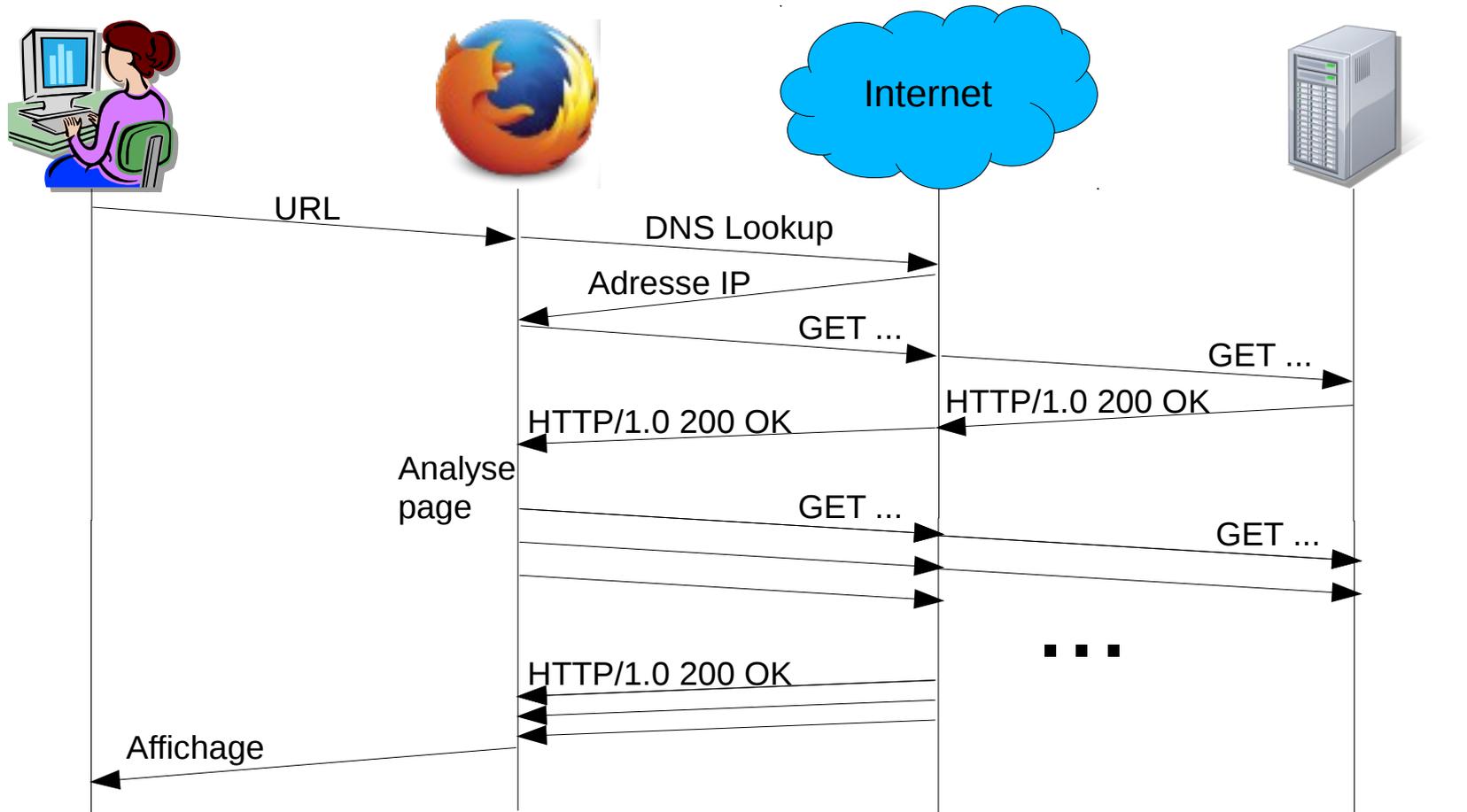
```
Content-Length : 1245
```

```
Last-Modified : Fri, 14 Jan 2000 08:25:13 GMT
```

# Codes réponse HTTP

Code	Message	Description
10x	<b>Message d'information</b>	Ces codes ne sont pas utilisés dans la version 1.0 du protocole
20x	<b>Réussite</b>	Ces codes indiquent le bon déroulement de la transaction
30x	<b>Redirection</b>	Ces codes indiquent que la ressource n'est plus à l'emplacement indiqué
40x	<b>Erreur due au client</b>	Ces codes indiquent que la requête est incorrecte
403	<b>FORBIDDEN</b>	L'accès à la ressource est tout simplement interdit
404	<b>NOT FOUND</b>	Classique! Le serveur n'a rien trouvé à l'adresse spécifiée.
50x	<b>Erreur due au serveur</b>	Ces codes indiquent qu'il y a eu une erreur interne du serveur

# En résumé



*Merci de votre attention*

