

# Réseau / Web / HTTP-CSS

Ryan Cassel  
[cassel@vimsr.fr](mailto:cassel@vimsr.fr)  
 Université Paris XI

# Réseau

## Réseau

### Qu'est ce qu'un réseau ?

- C'est un ensemble d'objets interconnectés les uns avec les autres
- Il permet de faire circuler des éléments entre chacun de ces objets selon des règles bien définies.

## Classification des réseaux

- LAN - Local Area Network
  - 1 m à 2 Km
  - 2 à 200 machines
  - 1 à 100 Mbits/s
- MAN – Metropolitan Aera Network
  - 1 m à 100 Km
  - 2 à 1000 machines
  - 1 à 100 Mbits/s
- WAN – Wide Aera Network
  - + de 1000 Km
  - Des milliers de machines
  - 50 bits à 2 Mbits/s

## Réseau

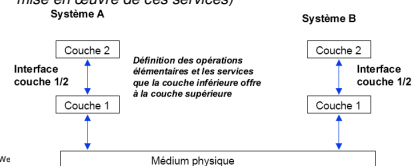
### Intranet et Extranet

- Intranet
  - Utilisations des protocoles de l'Internet mais dans un cadre privé et dans un lieu géographique limité (salle, établissement, entreprise, état)
- Extranet
  - Ouverture vers des partenaires (fournisseurs, clients, ...)

## Architectures des réseaux - généralités

### Organisation en série de couches ou niveaux

- Leur nombre et leur nom, leur fonction varie selon les réseaux
- L'objet de chaque couche est d'offrir certains services aux couches plus hautes (ces dernières ne connaissant pas la mise en œuvre de ces services)



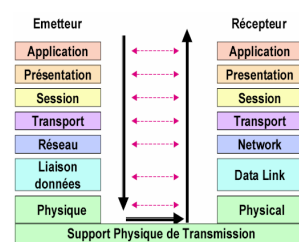
## Modèle de référence OSI

### ○ Open System Interconnection

- Modèle fondé sur le principe :
  - Diviser pour mieux régner
- Le principe de base est la description des réseaux sous forme d'un ensemble de couches superposées les unes aux autres
- L'étude du tout est réduit à celle de ses parties, l'ensemble devient plus facile à manipuler

## Modèle de référence OSI

### ● Modèle à 7 couches



# Protocoles de communication

## Couche réseau du modèle OSI (niveau 3)

- IP – Internet protocol
- ARP – Address Resolution Protocol
- RARP – Reverse Address Resolution Protocol
- ICMP – Internet Control Message Protocol

# Adressage IP

- Une adresse IP :
  - 4 octets (32 bits)
  - Notation « décimale pointée » A.B.C.D
    - Exemple 192.168.1.2
- Elle doit être unique au monde
  - Configurable par logiciel (ifconfig sous UNIX)
  - Associée à chaque interface réseau
- Attribution des adresses réseau en France :
  - Classe A et B par le NIC (Network Information Center) de l'Internet
  - Classe C en France :
    - NIC
    - Renater

# Adressage IP

- Découpée en deux :
  - Adresse de réseau, ou *network id*
    - Assignée par une autorité, identifie le réseau
  - Identificateur local de machine ou *host id*
    - Assigné par l'administrateur du réseau identifie la machine sur le réseau
  - Le découpage précis dépend de la classe d'adresse...

# Adressage IP

- Classification :
  - Classe A : N.H.H.H
  - Classe B : N.N.H.H
  - Classe C : N.N.N.H
    - Avec
      - N : adresse réseau
      - H : adresse locale
  - Classe D : cas particulier
    - pas de distinction Network/host
- L'espace d'adressage n'est pas hiérarchisé ou arborescent

# Adressage IP

## ● IP classe A

- 7 bits pour l'adresse du réseau
  - 1.0.0.0 à 126.0.0.0
- 24 bits pour l'adresse des machines
  - 254\*3 adresses locales possibles (16 277 214)

# Adressage IP

## ● IP classe B

- 16 bits pour l'adresse du réseau
  - 128.1.0.0 à 191.255.0.0
- 16 bits pour l'adresse des machines
  - 254\*2 adresses locales possibles (65 534)

# Adressage IP

## ● IP classe C

- 24 bits pour l'adresse du réseau
  - 192.0.1.0 à 223.255.255.0
- 8 bits pour l'adresse des machines
  - 254 adresses locales possibles

# Adressage IP

## ● IP classe D

- Adresse multicast
  - Transmission point à multi points; exemple vidéo conférence
- Réseaux 224 à 231
- Pas de structuration...
  - Car utilisée de façon très spéciale, ponctuelle, sans contrainte d'unicité, sans organisation géant leur attribution

## Adressage IP

### Adresses particulières

- Sois même : 127.0.0.1 (loopback ou localhost)
- Tous les bits de la partie machine à 0 → le réseau
  - Exemple : 130.190.0.0 désigne le réseau de classe B 130.190
- Tous les bits de la partie machine à 1 → tous les hosts du réseau
  - Diffusion, broadcast IP
  - 130.190.255.255 désigne toutes les machines du réseau 130.190
- 0.0.0.0 : une machine ne connaît pas son adresse

## Réseaux

### Principe Client / Serveur

- Des machines clientes contactent un serveur

### Avantages

- Ressources centralisées
- Meilleure sécurité
- Administration au niveau serveur
- Réseau évolutif (des clients peuvent disparaître sans perturber le fonctionnement du réseau)

### Inconvénients

- Coût élevé du serveur
- Le serveur est le maillon faible

## Réseaux

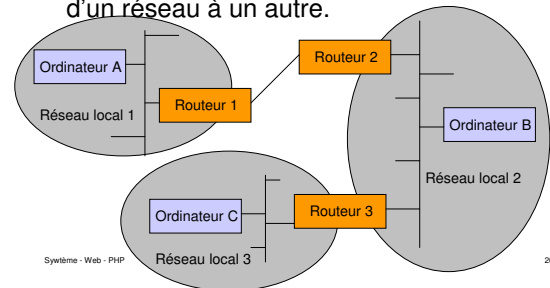
### Exemple serveur de mail

- Serveur :
  - Se charge de recevoir et d'envoyer des mails
- Client :
  - Se connecte au serveur pour récupérer ses mails
  - Se connecte au serveur pour lui faire envoyer ses mails

### Un serveur est connu par toutes les machines d'un réseau

## Routage

### Routage sert à rediriger des informations d'un réseau à un autre.



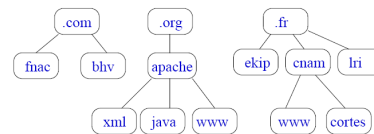
## DNS

Établit une correspondance noms/adresses

- Jusqu'en 1984, tout était centralisé dans un seul fichier *hosts.txt* !
- Remplacé par le DNS: une organisation hiérarchique des noms de domaine
- La gestion du DNS est décentralisée: un sous-domaine est géré par un serveur de nom

## Structure du DNS

Les serveurs de nom collaborent entre eux pour convertir un nom de machine en adresse IP.



## Réseaux

### Commandes UNIX

- Nslookup (informations sur une IP)
- netstat (informations sur le réseau)
- Ifconfig (configuration du réseau local)
- Ping (test des connexions)
- Traceroute (chemin des paquets vers IP)
- Tcpdump (trafic réseau)

*Rappel : utilisez souvent la commande : « man »*

Internet

## Qu'est ce que l'Internet ?

- internet
  - Réseau d'interconnexion entre deux ou plusieurs réseaux.
- L'Internet
  - L'ensemble des réseaux internet utilisant le protocole TCP / IP en respectant les standard en vigueur

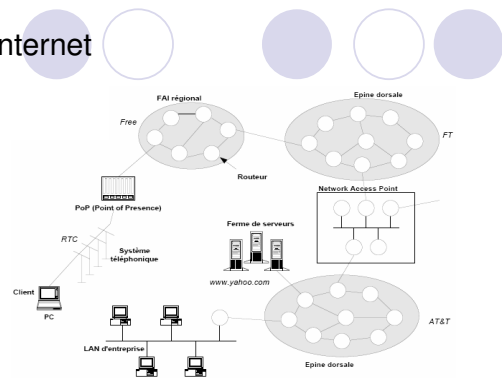
## Internet

- Ensemble de logiciels et de protocoles
  - Basé sur l'architecture TCP/IP
  - Fonctionne en mode client/serveur
  - Offre un ensemble de services (e-mail, transfert de fichiers, connexion à distance, www, ...)

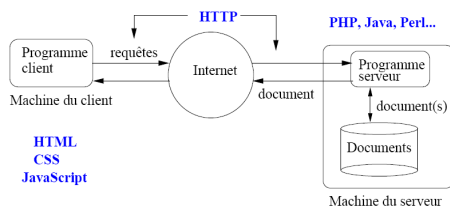
## Internet

- Construction par le bas
  - Réseau local (laboratoire, département,...)
  - Réseau local (campus, entreprise,...)
  - Réseau régional
  - Réseau national
  - Réseau mondial

## Internet



## Fonctionnement du web



## Internet - Bref Historique

- 1959-1968 : Programme ARPA
  - Ministère américain de la défense : lancer un réseau capable de supporter les conséquences d'un conflit nucléaire.
- 1969 : ARPANET (l'ancêtre d'Internet)
  - Les universités américaines s'équipent de gros ordinateurs et se connectent au réseau ARPANET.
- 1970-1982 : ouverture sur le monde
  - Premières connexions avec la Norvège et Londres
- 1983 : Naissance d'Internet
  - Protocole TCP/IP → tous les réseaux s'interconnectent, les militaires quittent le navire

## Internet - Bref Historique

- 1986 : Les autoroutes de l'information
  - La National Science Foundation décide de déployer des super-ordinateurs pour augmenter le débit d'Internet
- 1987-1992 : Les années d'expansion
  - Les fournisseurs d'accès internet apparaissent et les entreprises se connectent au réseau
- 1993-2006 : Explosion d'Internet
  - Ouverture au grand public
  - Avènement du courrier électronique et du Web
  - Marché considérable

## Services d'Internet

- Un service = une application qui utilise un protocole et un numéro de port
- Fonctionnement en mode client/serveur au dessus de TCP/IP

## Internet

- Une version du modèle OSI (très simplifié)
  - Service
    - ftp, www, telnet, ...
  - Transport
    - TCP, UDP (entre deux processus aux extrémités)
  - Réseau
    - IP (routage)
  - Transmission
    - Entre 2 sites : pas de protocoles spécifiques

## Services de base de l'Internet

- telnet (TErminAL NETwork)
  - non sécurisé !
  - Accéder à une machine distante
- FTP (File Transfert Protocol)
  - Transfert de fichier
- E-mail (Electronic mail ou messager électronique)
  - SMTP (Simple Mail Transfert Protocol)
  - POP (Post Office Protocole)
- News
  - NNTP (News Network Transport Protocol)

## Services de base de l'Internet

- World Wide Web (ou Web)
  - Le web repose sur l'hypertexte
  - Une page hypertexte est écrite en HTML
  - Ces pages se trouvent sur un serveur Web
  - Le protocole utilisé est le HTTP (Hyper Text Transport Protocol)

## Web

- Architecture pour accéder à des documents liés entre eux et situés sur des machines reliées par Internet
- Architecture basée sur 3 concepts :
  - La localisation : URL
  - Le protocole : HTTP
  - Le langage : HTML
- Popularité due à :
  - Interface graphique conviviale
  - Très grande quantité d'informations
  - Grande diversité d'informations

## Web

- Une page web:
  - Contient des objets
  - Désignés par une adresse (URL)
- Les pages web contiennent en général:
  - Du code HTML
  - Des objets référencés
- L'URL a au moins deux composantes :
  - Le nom de l'hôte contenant la page web
  - Le chemin d'accès sur l'hôte

[www.free.fr/perso/web](http://www.free.fr/perso/web)
- L'agent utilisateur pour le web est le browser (navigateur) :
  - Netscape
  - Mozilla
  - Thunderbird
  - Internet Explorer
  - Opera
  - ...
- Le serveur web
  - Apache (domaine publique)
  - MS Internet Information Server

## Fonctionnement du web

- Le client (browser) dialogue avec un serveur web selon le protocole HTTP
- Le serveur vérifie la demande, les autorisations et transmet l'information
- Le browser interprète le fichier reçu et l'affiche
- A ceci peut s'ajouter des contrôles ou des exécutions (côté client ou côté serveur)

## Adressage de documents

- Il faut nommer localiser et accéder à une page:
  - Quoi ? Où ? Comment ?
- Solution :
  - URL Uniforme Ressource Locator : Adresse Universelle de Ressource
  - En 3 parties : le protocole (comment), le nom DNS (où), le nom du document (quoi).
- URI (Uniforme Ressource Identifier) est un sur ensemble d'URLs
- Exemple d'URL :
  - <http://www.perdu.com/index.html>

## Adressage de documents

- Différentes composantes d'une URL

`protocol://host_name:port/path/extrapath?arguments`
- La racine « / » de path est définie par la configuration du serveur web (à ne pas confondre avec la racine du système de fichier sur le serveur)
- `extrapath?arguments` permettent de passer des informations à des programmes s'exécutant sur le serveur

## Adressage d'un document

- URL relative :
  - un lien vers "images/new.gif" dans la page  
`http://www.monsite.fr/projet/doc.html`  
est - par défaut - un lien vers  
`http://www.monsite.fr/projet/images/new.gif`
  - le navigateur client reconstruit l'URL absolue pour faire la requête
  - la balise HTML `<BASE href="url">` permet de positionner la racine absolue pour les URLs relatives du document contenant cette balise

## Vision côté client

- Le Web est un ensemble de pages (documents) pouvant contenir des liens vers d'autres pages n'importe où dans le monde
- Consultation des pages via un navigateur
- L'utilisateur suit ces liens par simple click → notion d'hypertexte (information répartie)
- Le navigateur (browser)
  - analyse l'URL demandée
  - demande au DNS l'adresse IP du site distant
  - établit une connexion TCP vers le numéro de port de l'URL (80 par défaut)
  - formule la requête au serveur

## Vision côté serveur

- Le navigateur (browser)
  - va rechercher la page demandée
  - interprète les commandes de formatage et de mise en forme (police, gras, couleurs...)
  - va rechercher et affiche des images
  - animation (code JavaScript, gifs...)
  - affiche la page correctement formatée
- Paramétrage à plusieurs niveaux
  - valeurs par défaut du navigateur
  - valeurs fixées dans le document
  - préférences de l'utilisateur (navigateur)
  - exemples : couleur des liens (visités ou non), du texte, fond de la page, polices...

## Vision côté serveur

- Le serveur est en permanence à l'écoute des requêtes formulées par les clients (qui peuvent être très nombreux !)
- Il vérifie la validité de la requête...
  - adresse correcte (URL)
  - client autorisé à accéder au document
- ...et y répond : envoi du texte, des images, du code à exécuter sur le client, d'un message d'erreur, d'une demande d'authentification, ...
- Il peut exécuter un programme localement qui va générer une réponse HTML (pages dynamiques)

## Web

- Les termes à connaître :
  - HTTP
  - HTML
  - URL
  - URI
  - Upload
  - Download
  - Serveur
  - Client
  - Apache
  - PHP
  - Browser
  - Editeur

HTML

## HTML

- En 1994 création du WWW Consortium par le CERN et le MIT
- Les différents HTML
  - HTML
  - DHTML
  - XHTML
  - ...
- Pour tout savoir sur les normes du HTML :  
[www.w3.org](http://www.w3.org)

## Principe de HTML

- Structuration d'un texte à l'aide de balises

# HTML

## • Hyper Text Markup Language

- un langage de description permettant de structurer et d'afficher différents objets sur un écran (textes, tableaux, images, vidéos, ...)
- une implémentation simple de SGML (Standard Generalized Markup Language)
- un document HTML doit pouvoir s'afficher dans n'importe quel navigateur **MAIS**
  - les affichages ne sont pas identiques selon le contexte et le navigateur utilisé

## • Pourquoi apprendre le HTML ?

# HTML

- Définit la structure logique d'un document WEB
- Composé d'un ensemble de commandes de formatage

- Basé sur la notion d'**environnement** possédant un début et une fin

→ **délimiteurs** : balises ou marqueurs

- Pour mettre en oeuvre du HTML, il suffit de

- un éditeur de texte
- un navigateur WEB

# HTML

- Les balises sont définies entre <>

○ <BALISE>...</BALISE>

- La plupart des environnements peuvent être imbriqués selon des règles bien définies

○ <H1><B>Mon titre</B></H1>

- Il n'est pas permis de faire chevaucher des environnements

○ <H1><B>Mon titre</H1></B>

- Les minuscules/majuscules n'interviennent pas dans la définition des balises

# HTML – Les bases

- Les balises dénotent des "constructions« documentaires

- styles de paragraphe (normal, énumérations, titres...)
- tableaux
- styles de caractères (gras, italique, souligné...)
- références à des images
- références hyper-texte
- formulaires
- etc...

- Fichiers par défaut : Welcome.html, index.html, test.htm

# Les règles de « bonne conception »

- **Mise en page et styles** : maintenir une certaine uniformité sur l'ensemble du site
- **Clarté** : éviter la confusion, présenter de façon simple avec des menus, éviter les textes longs...; l'information doit être trouvée dès les premières secondes
- **Hyperliens** : ne pas en abuser, faire en sorte qu'ils indiquent clairement où ils mènent
- **Identité** : le visiteur doit savoir facilement sur quel site il est (quelle que soit la page visitée) et savoir à qui s'adresser → logos, @mail, date de dernière maj...
- **Réactivité** : ne pas abuser des images, exécutions de code...(temps de chargement/exécution)
- **Navigateurs** : faire attention à leurs particularités → optimiser le site pour IE et Netscape...

# Structure « minimale » d'un document

```
<HTML>
  <HEAD>
    <TITLE>titre-fenetre</TITLE>
  </HEAD>
  <BODY>
    corps du document
  </BODY>
</HTML>
```

- HEAD : pour différencier du reste du texte, contient les titres
- TITLE : affiché en haut de la fenêtre, utilisé dans les signets (bookmarks)
- BODY : contient le document

# Balises et attributs

```
...
  <BODY>
    <ADDRESS>
      Jean Dupont - Tel: 01 23 45 80 80
    </ADDRESS>
  </BODY>
...
```

- Exemple de balise : <ADDRESS>
  - un style "adresse", généralement affiché en italique
- Les commentaires : <!-- ne sera pas interprété -->
- Comme pour tout langage,

**RESPECTER L'IDENTATION**

# Balises et attributs

- Attributs avec affectation d'une valeur ou non
  - <BALISE attribut1="10" attribut2="blue">
- Attributs "de coeur" (disponibles pour la plupart des balises)
  - class="name" : applique un style au contenu
  - id="name" : donne un nom unique à la balise
  - lang="langage" : spécifie la langue du contenu (ISO639)
  - des attributs liés aux événements : onclick="action", ondblclick, onkeydown, onkeypress, onkeyup, onmouseover, onmouseout, ...

## Quelques balises

- Liste : `<OL> <LI> ... </LI> </OL>`
- `<P>`: nouveau paragraphe
- `<Hn>` (0 < n < 8) pour les titres
- `<TABLE>` pour des tableaux
- `<FORM>` pour des formulaires
- `<IMG SRC='url'>` pour insérer une image
- `<B>`, `<IT>`, `<TT>`: gras, italique, courier

## Feuilles de style / CSS

## CSS

- Apparu en 1996
- Qu'est ce que CSS ?
  - Cascading Style Sheet
  - Feuille de style

## Principe des feuilles de style

- Même idée que les styles des traitements de texte
  - distinction entre la structuration du texte et les propriétés typographiques que l'on applique au texte
  - les styles permettent d'appliquer certaines propriétés à des balises HTML
  - plusieurs documents peuvent partager les mêmes instructions typographiques
  - homogénéité de typographie et de présentation des pages auxquelles s'appliquent les styles
  - l'apparence des documents peut facilement être modifiés en utilisant les styles définis

## Principe des feuilles de style (à partir de HTML 4)

- Les styles
  - peuvent être définis directement dans la page HTML
  - mais sont généralement stockés dans un fichier séparé (c'est la feuille de style !)
  - permettent d'accélérer le chargement des pages (la feuille de style est chargée une fois pour toute !)
- Deux façons de décrire des feuilles de style
  - **CSS - Cascading Style Sheets**
    - langage déclaratif type HTML
  - JASS
    - issu du langage Javascript
    - plus orienté vers la manipulation dynamique des propriétés décrivant un style

## Principe des feuilles de style

- Mise en cascade des styles
  - possible d'imbriquer dans un même document plusieurs styles avec un niveau de priorité différent
  - les propriétés non définies pour la balise courante sont héritées des balises qui contiennent la balise courante
  - un style défini directement dans une page HTML sera plus prioritaire qu'un style défini dans un fichier externe
  - idée : on peut imaginer que plus tard, le lecteur (client) pourra redéfinir ses propres styles qui seront prioritaires sur les styles définis par le concepteur de la page

## Exemple d'une feuille CSS

```
/* Le fond est toujours en blanc */
BODY {background-color: white}

/* Couleur rouge pour ancrs et en-têtes */
A, H1, H2, H3 {color: #ca0000}
CAPTION {font-size:large;color: #ca0000}

/* Classes pour alterner les couleurs des lignes */
TR.A0 {background-color:white}
TR.A1 {background-color:yellow}
.attention {background-color:red;color:white}
```

## Quelques commentaires

- Des instructions globales: A, H1, H2, H3  
`{color: #ca0000}`
  - → le style sera appliqué à toutes les balises
- Des classes pour raffiner les styles TR.A0  
`{background-color:white}`
  - → on peut appliquer `<TR CLASS='A0'>`
- Les classes peuvent être génériques :  
`.attention {background-color:red}`
  - → Ex : `<B CLASS='attention'>`