

# Bachelier en Informatique de Gestion

## WEB : principes de base

Enseignement supérieur économique de type court

Code FWB : 7534 29 U32 D1

Code ISFCE : 4IWPB



# Table des matières

## Généralités

- 01. Introduction au web
- 03. Outils
- 05. Frameworks

## Côté Client

- 12. Structure HTML
- 13. Formulaire HTML
- 14. Mise en forme CSS
- 15. Adaptabilité
- 17. Javascript
- 18. Framework jQuery
- 19. AJAX

## Côté Serveur

- 21. Middleware PHP
- 22. Traitement du formulaire
- 23. Architecture MVC
- 24. Base de données SQL
- 25. Données XML
- 26. Données JSON



# Ressources



- Interne
  - [www.burotix.be](http://www.burotix.be) / support de cours / Programmation / Web
  - [moodle.isfce.be](http://moodle.isfce.be)
- Tutoriel
  - <https://openclassrooms.com/>
  - <http://www.commentcamarche.net/contents/webmasters-4>
  - <http://developpez.com>
- Documentation
  - <http://php.net/manual/fr/>
  - <http://www.w3schools.com/>
  - <https://developer.mozilla.org/en-US/docs/Web>

# Le Professeur, c'est qui ?



Alain Wafflard

alain.wafflard@gmail.com



Ingénieur civil

25 ans dans le privé  
Processus d'entreprise  
Automatisation, informatisation



Enseignant

ISFCE Etterbeek  
mais auparavant : IFOSUP Wavre, IFC Ixelles,  
IEPSCF Evere, IPAM La Louvière, EPFC  
Bruxelles



Activité complémentaire

burotix.be  
formation en tableur  
conception de base de données  
développement de templates



# 01. Introduction au web

Accessibilité d'Internet	Modèles OSI et TCP/IP
Paradigme Client - Serveur	Les différentes technologies du web



= BUROTIX ()

# Accessibilité d'Internet

surface web

dark web

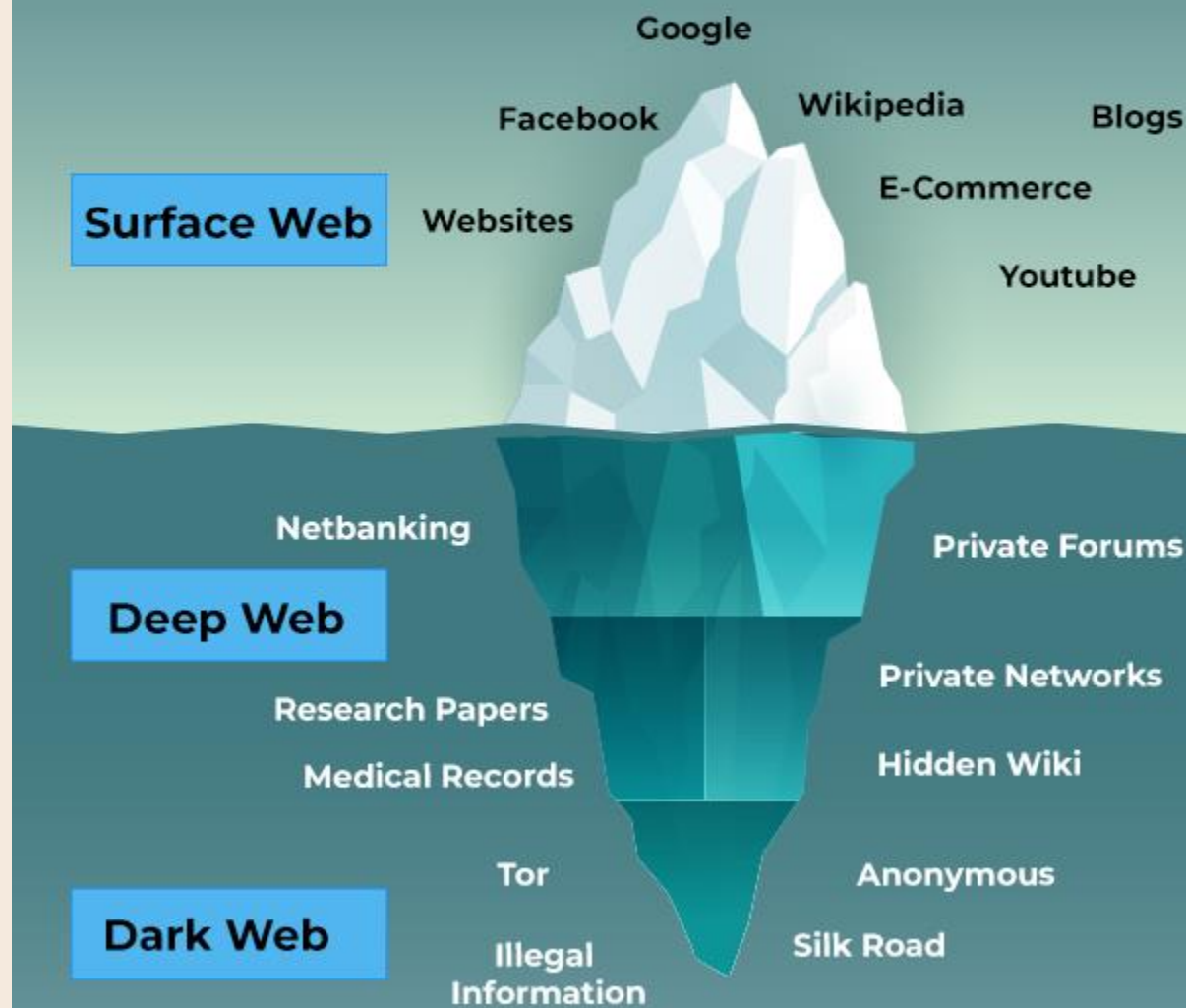
deep web



= BUROTIX ()

# Accessibilité d'Internet

## SURFACE WEB VS. DEEP WEB VS. DARK WEB



# Accessibilité d'Internet

## Surface Web

- partie d'internet accessible à tous via les moteurs de recherche (Google, Bing, etc.)
- sites publics et indexés

## Deep Web

- partie d'internet non indexée par les moteurs de recherche
- contenus accessibles via des connexions privées
- e-mails, bases de données, services en ligne, etc.
- contenu légal

## Dark Web

- sous-partie du Deep Web accessible uniquement via des logiciels spécifiques (comme Tor)
- souvent associé à l'anonymat et parfois à des activités illégales



= BUROTIX ()



# Le paradigme Client-Serveur

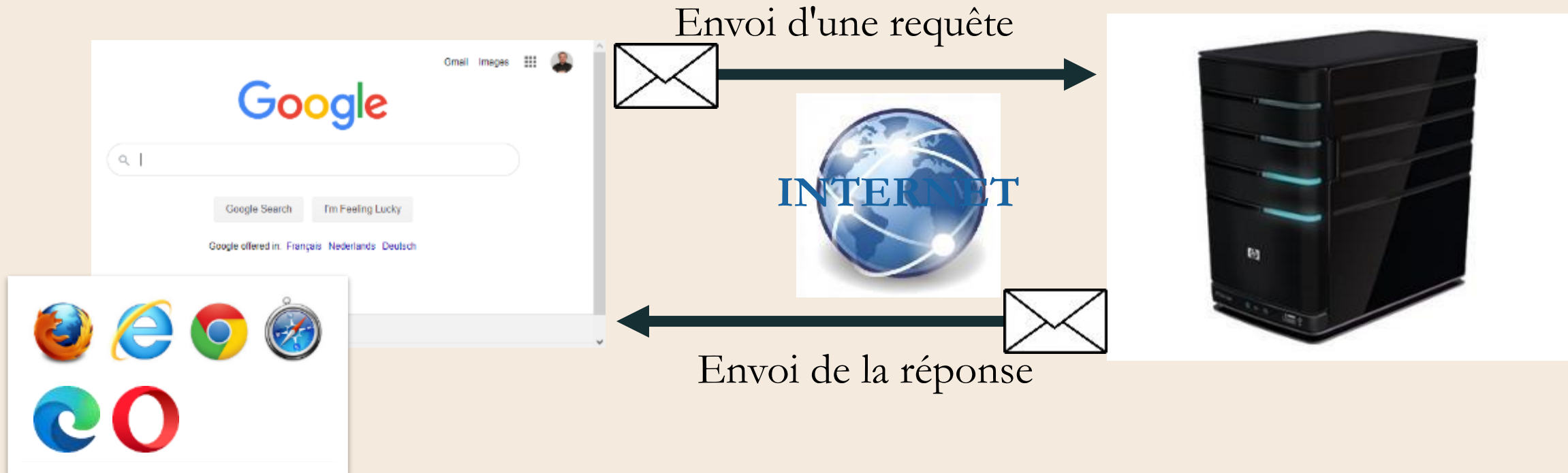


= BUROTIX ()

# Paradigme Client-Serveur

CLIENT

SERVEUR WEB



# Paradigme Client-Serveur

CLIENT

SERVEUR WEB

Envoi d'une requête

INTERNET

Envoi de la réponse



Google

Gmail Images

Q |

Google Search

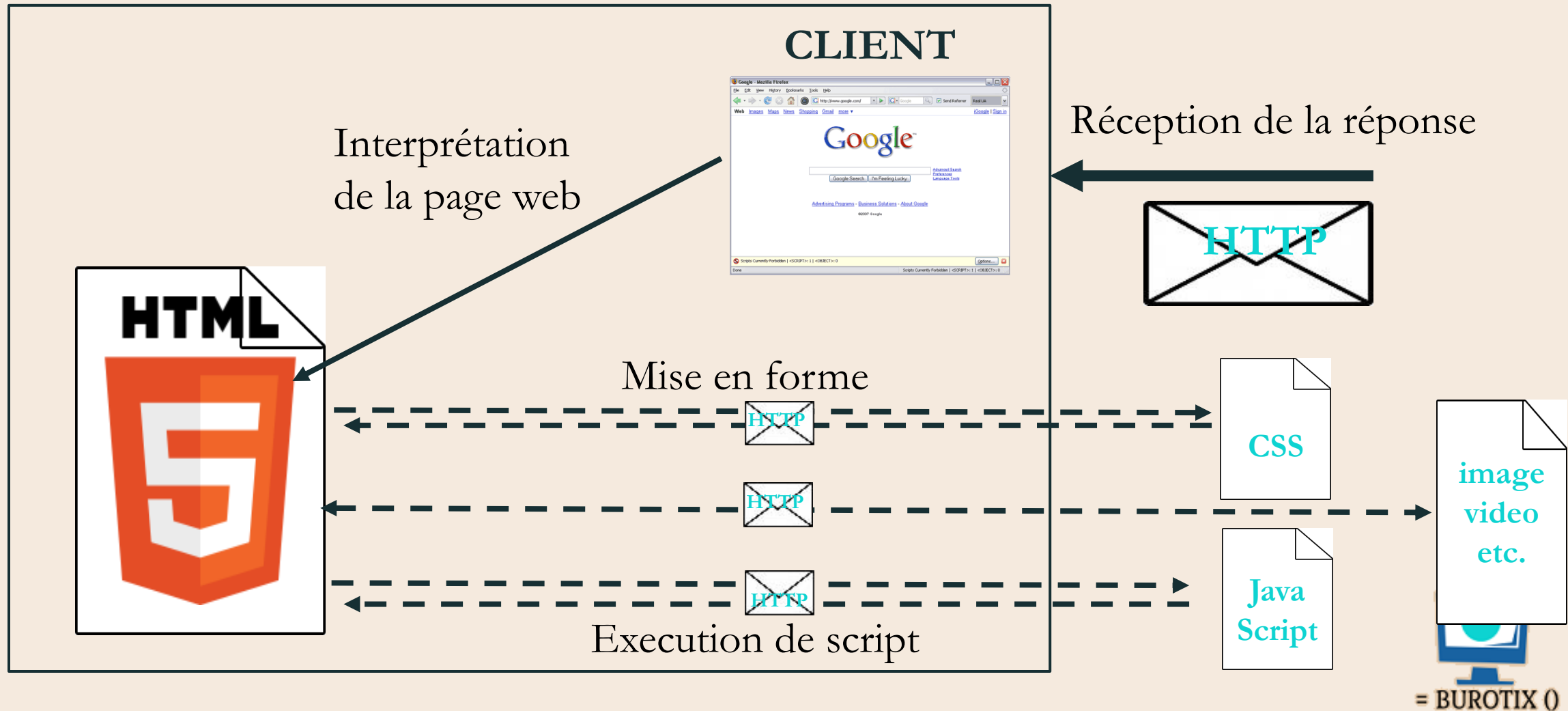
I'm Feeling Lucky

Google offered in: Français Nederlands Deutsch



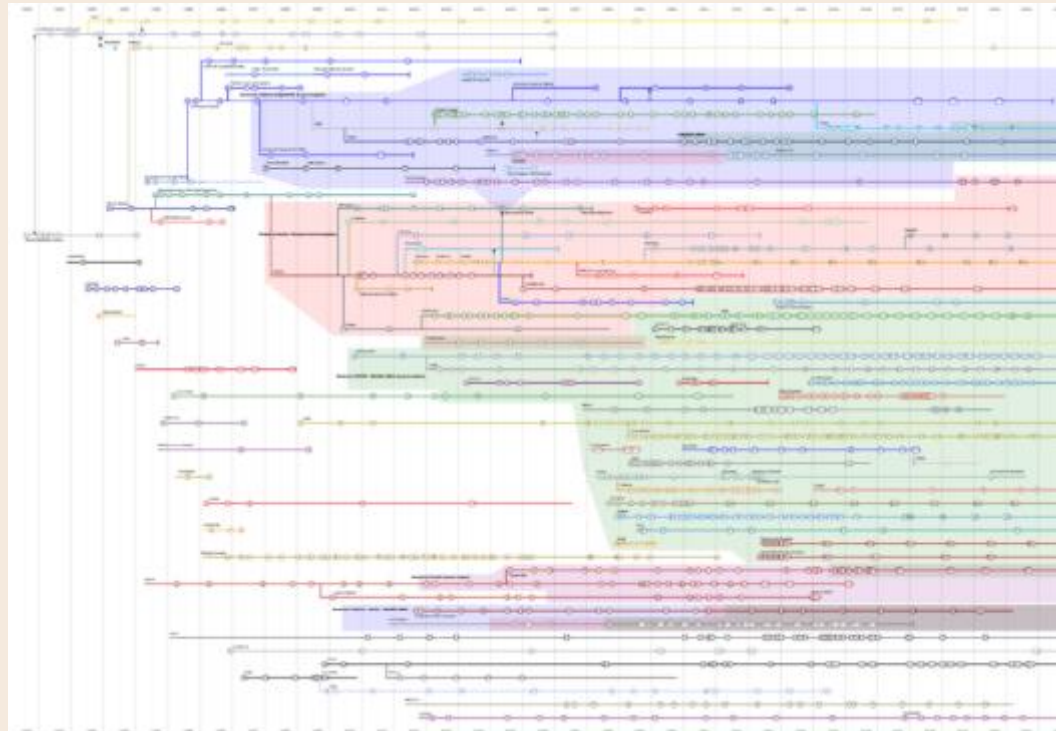
= BUROTIX ()

# Paradigme Client-Serveur



# Exo : découvrir le "client", navigateur web

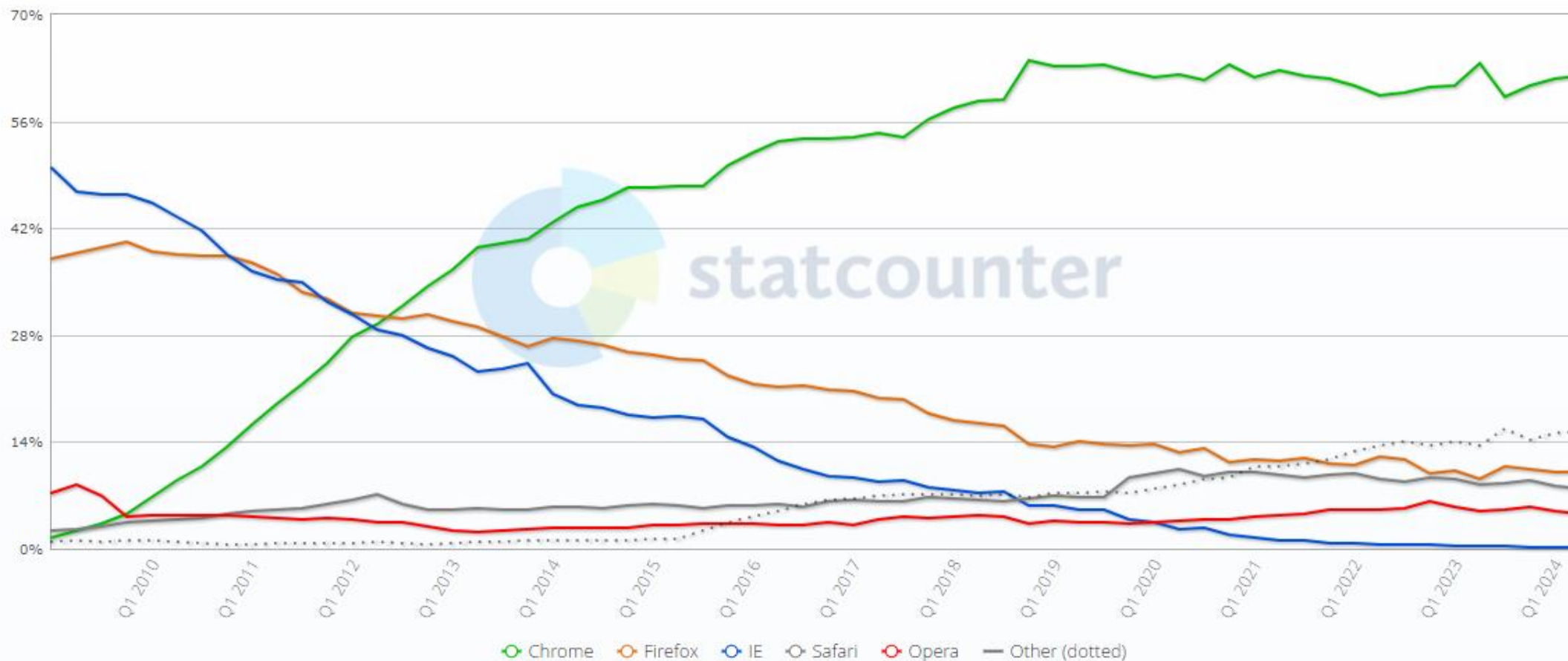
- Les principaux navigateurs
- Leur usage dans le temps



# Desktop Browser Market Share Europe

Q1 2009 - Q2 2024

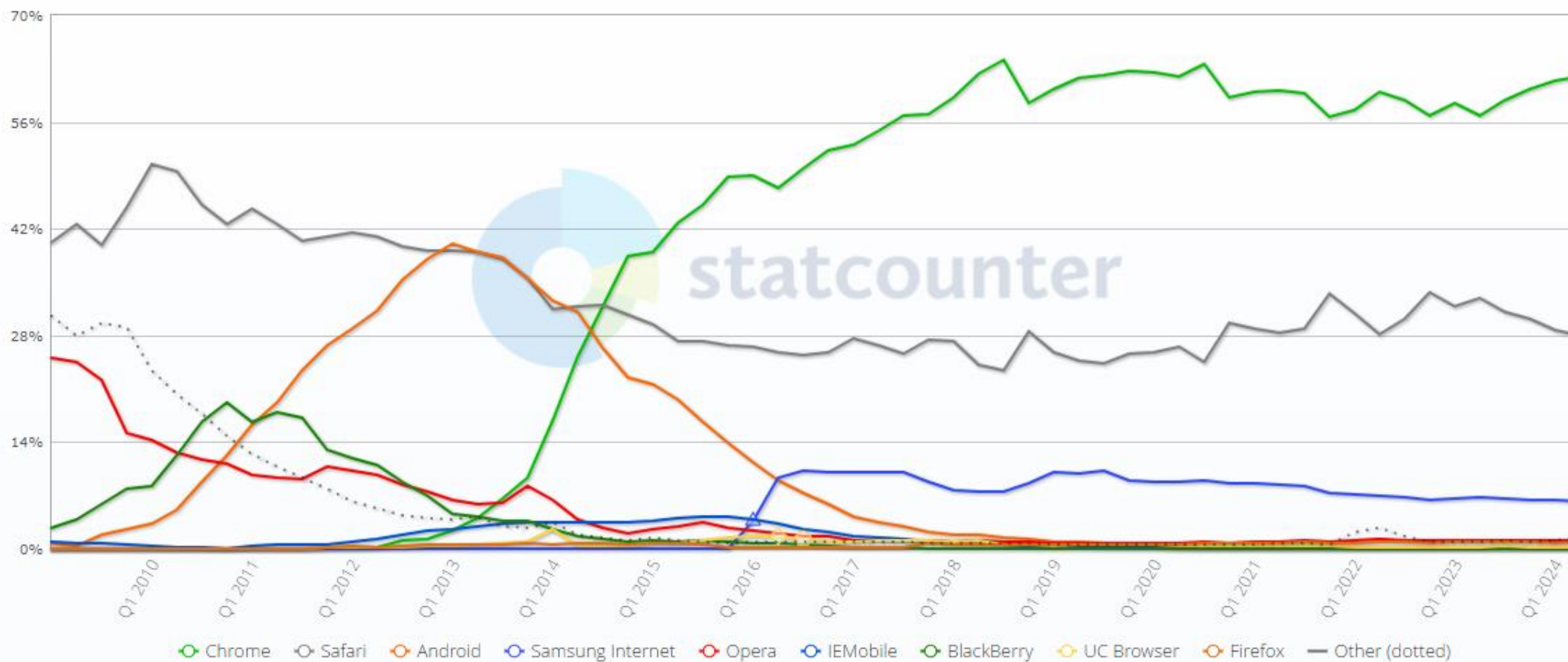
Edit Chart Data



# Mobile Browser Market Share Europe

Q1 2009 - Q2 2024

Edit Chart Data

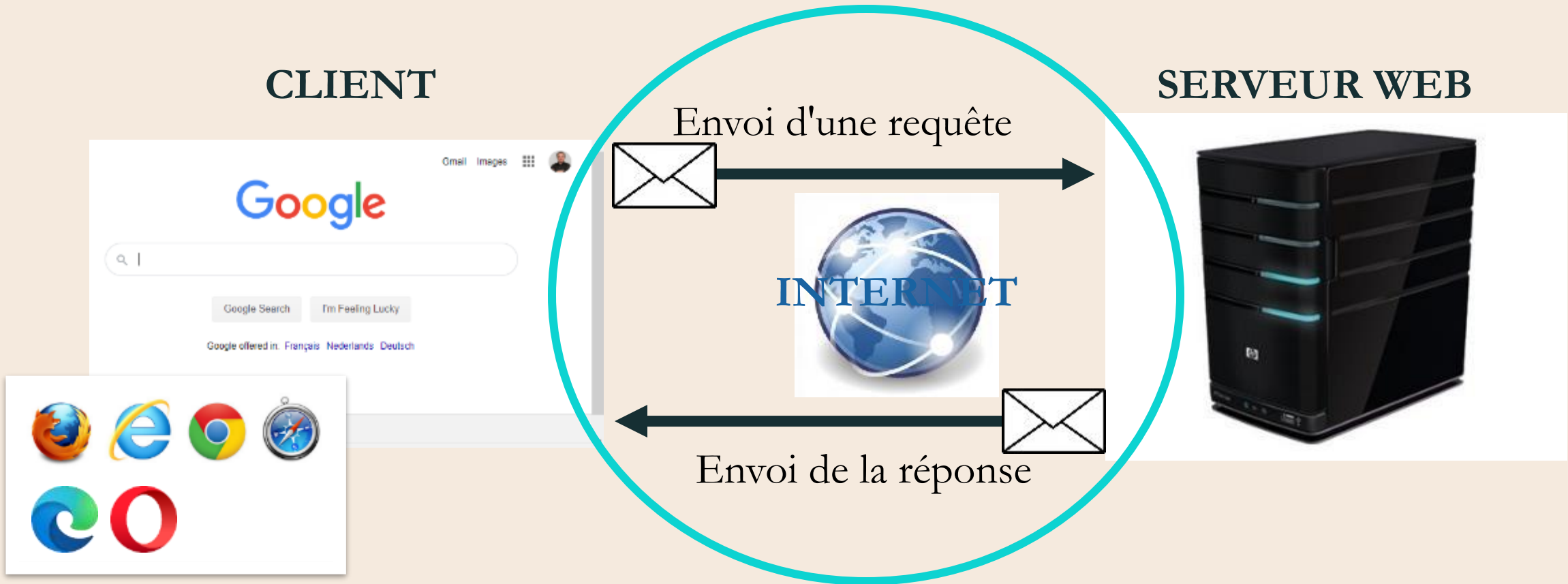




# Paradigme Client-Serveur

CLIENT

SERVEUR WEB





# Paradigme Client-Serveur



- Protocole de communication:
  - = spécification de règles pour un type de communication particulier
  - Idem pour courrier postal:
    - Position précise pour l'adresse formatée suivant certaines règles
  - HTTP (HyperText Transfer Protocol)
  - HTTPS (HyperText Transfer Protocol Secure)
  - FTP (File Transfer Protocol)
- Adresse : URL (Uniform Resource Locator)
  - <https://fr.wikipedia.org/w/index.php?search=fondateur+informatique>
- Résolution du nom de domaine (wikipedia.org) grâce au DNS (transformation en adresse IP: 208.80.154.224)



# Paradigme Client-Serveur



- Exemple requête HTTP :

```
GET /imghp?hl=fr&tab=wi HTTP/1.1
Host: images.google.fr
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; fr;
rv:1.9.0.9) Gecko/2009040821 Firefox/3.0.9 (.NET CLR 3.5.30729)
FirePHP/0.2.4
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: fr,fr-fr;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
```

- Cookie :

```
rememberme=false;VAR=-120
```



# Paradigme Client-Serveur



## Exemple 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

Code:

200 - OK

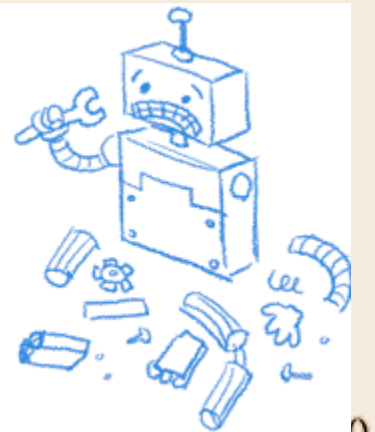
400 - BAD REQUEST

404 - NOT FOUND

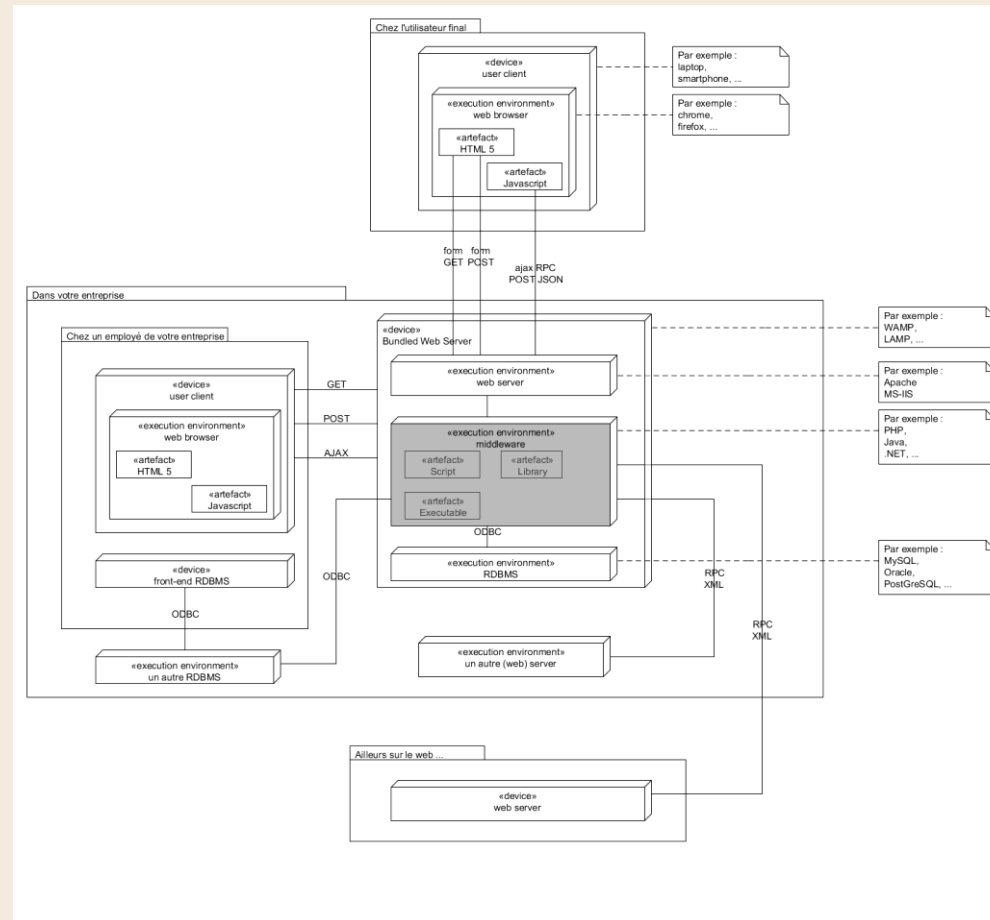


**404.** That's an error.

The requested URL /nonexistent was not found on this server. That's all we know.



# Une architecture client-serveur complexe



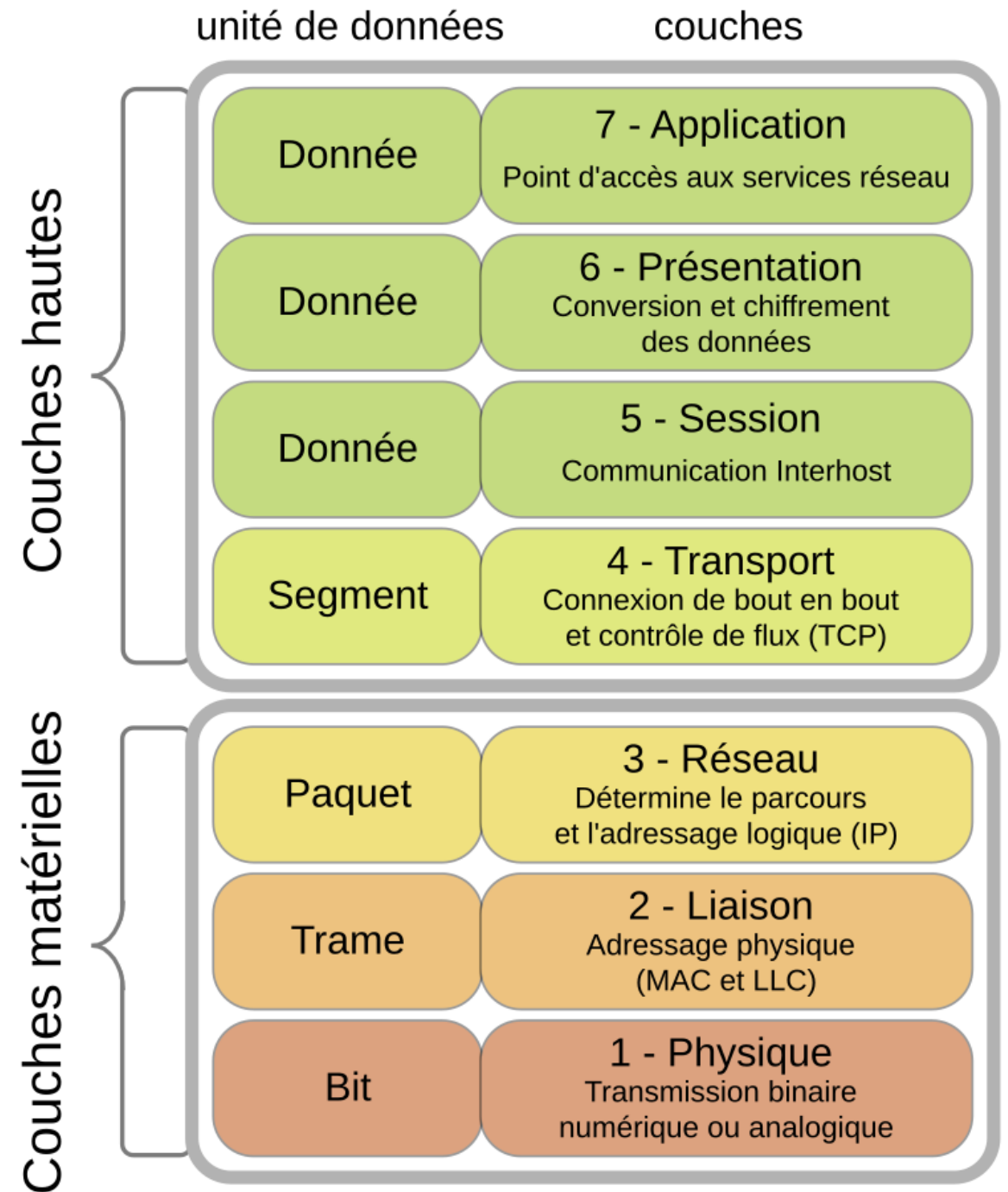
# Modèles OSI et TCP/IP



= BUROTIX ()

# Modèle OSI

- Approche en couche
  - Une couche n'a pas besoin de savoir comment celle du dessous fonctionne
  - Une couche peut se limiter à faire ses propres tâches (son “core business”).



# Modèle TCP/IP



- Variante du modèle OSI
  - plus souple et mieux adaptée à Internet

<b>Application</b>	HTTP, FTP, DNS	Spécifie les mécanismes de communication au sein des applications
<b>Transport</b>	TCP, UDP	Résout des problèmes comme la fiabilité des échanges Assure que les données arrivent dans l'ordre correct Désigne l'application de destination (via le port)
<b>Réseau</b>	IP	Résout le problème de l'acheminement de paquets à travers un seul réseau
<b>Liaison</b>	Ethernet, Token Ring, Wireless	Spécifie comment les paquets sont transportés (encapsulation dans une trame)
<b>Physique</b>	Lignes tél., RTC, ADSL, fibre optique	Définit les types de câbles, la modulation du signal...

# Analogie postale



## ■ Application

- Si une personne veut quérir une information à une autre (ex. Ores qui souhaite connaître les index des consommateurs d'énergies en Wallonie), celle-ci envoie une lettre à l'utilisateur en lui demandant de répondre avec l'information demandée.
- Il y a des règles à suivre pour composer l'enveloppe :
  - Format de l'adresse : nom, rue, code postal, pays
  - Place de l'adresse : en bas à droite
  - Place du timbre : en haut à droite
- Internet : La couche application me permet juste de dire à qui j'envoie l'information sans devoir dire comment celle-ci va arriver (par quel chemin, etc...).

## ■ Transport

- Que faire si la lettre n'arrive pas (retour à l'expéditeur, etc...)
- Que faire si elle arrive (accusé de réception nécessaire, notification).
- Internet: Si l'enveloppe a été découpée en petit morceau, dans quel ordre recoller les morceaux.

## ■ Réseau

- Déterminer le chemin que va prendre l'enveloppe
  - Relève de la boîte aux lettres à 6h
  - Direction centre de tri
  - Envoi à un bureau local
  - Passage par les routes X, Y, Z
  - Arrivée chez le destinataire
- Internet : Routage des paquets

## ■ Liaison

- Une partie du chemin est fait dans un camion, une autre par camionnette et finalement le facteur fera à vélo la partie restante.

## ■ Physique

- L'enveloppe va voyager sur une route goudronnée, puis sur une autoroute, puis sur un trottoir, etc...
- Internet : fibre optique, ligne téléphonique, etc.



= BUROTIX 0



# Exemples de protocoles "application"

- Transfert de fichiers
  - File Transfer Protocol (FTP)
  - SMB (Microsoft)
- Messagerie
  - Simple Mail Transfer Protocol (SMTP)
  - Post Office Protocol (POP)
  - Internet Message Access Protocol (IMAP)
- Session à distance
  - Telnet
  - Secure Shell (SSH)
- Envoi de pages HTML
  - Hyper Text Transfer Protocol (HTTP)
- Exploitation et gestion du réseau
  - Domain Name System (DNS) : résolution d'adresse
  - Simple Network Management Protocol : supervision



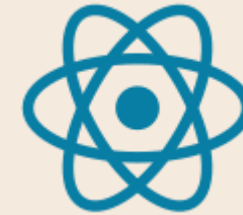
# Les technologies du web



= BUROTIX ()

# Technologies du web

- HTML
- CSS
  - Bootstrap
  - responsivity
- Javascript
  - Jquery
  - AJAX
- PHP
  - ASP.NET
  - Java
  - Python
- Données
  - SQL
  - XML
  - JSON
  - API
- Framework
  - Laravel, CodeIgniter, Django
  - React.js, Angular.js, Vue.js



# HTML

```
<h1>Ceci est un titre</h1>
```

- HyperText Markup Language
- Langage de balisage conçu pour structurer les pages web
- Permet de:
  - Créer des documents interopérables (valides sur des systèmes multiples)
  - Donner une structure sémantique
  - Inclure du contenu multimedia (images, vidéos...)
  - Créer des formulaires de saisies
  - Intégrer des éléments programmables (JS)
- Permettait la mise en forme du contenu
  - Fonctionnalités transférées au CSS
  - cf <http://www.w3.org/TR/html5-diff/#obsolete-attributes>
- Interprété côté CLIENT

# CSS

```
h1 {  
    font-size: 200%;  
    color: navy;  
    text-align: center;  
}
```

- Cascading Style Sheets
- Langage informatique décrivant la présentation des documents HTML
- Permet de:
  - séparer structure et mise en forme
  - décliner les styles de présentation selon le récepteur
  - combiner différentes sources de mise en forme
- Interprété côté CLIENT

```
document.getElementById("demo").innerHTML = "Hello JavaScript!"
```

# JavaScript

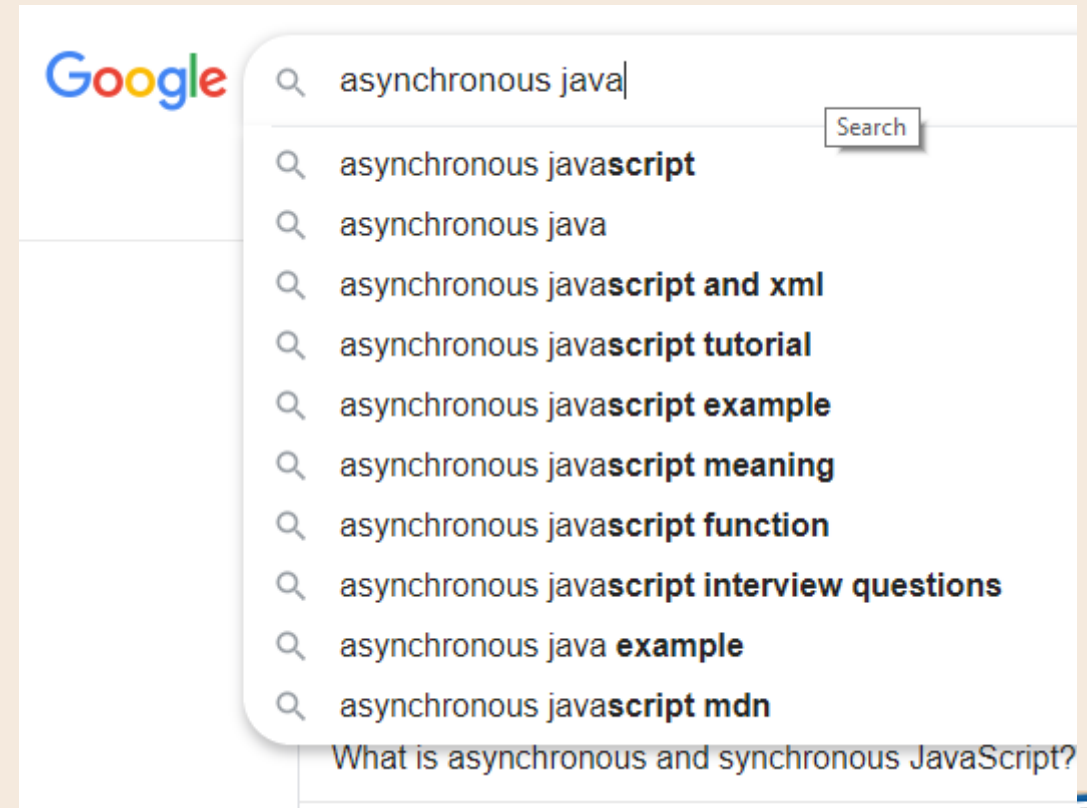
- Langage de scripts principalement utilisé dans les pages web interactives
- Permet de:
  - contrôler les données saisies dans des formulaires HTML
  - interagir avec le document HTML via l'interface Document Object Model
  - réaliser des services dynamiques (cosmétiques, ergonomiques...)
- Interprété côté **CLIENT**



= BUROTIX 0

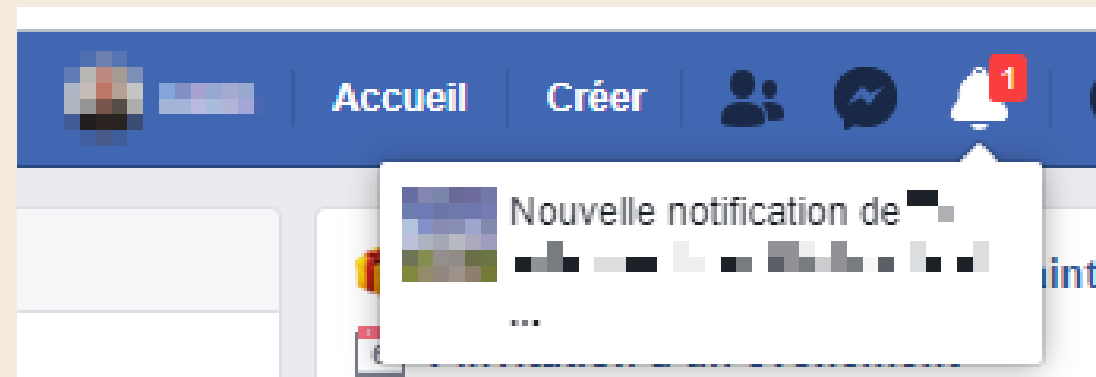
# Ajax

- Le principe par l'exemple :  
les suggestions de recherche
  - vous saisissez quelques mots-clés
  - le moteur propose spontanément d'autres mots-clés
  - sans rafraîchissement de la page
  - très rapidement



# Ajax

- "Asynchronous JavaScript And XML"
- Architecture informatique (Ajax n'est pas un langage)
- Combine HTML, JavaScript, CSS, XML, ...
- Interprété coté **CLIENT** pour interroger le **SERVEUR**
- Permet d'améliorer la maniabilité des applications web
- Utilisé dans les grandes applications web :
  - Facebook
  - Gmail, ...





# PHP

- "PHP Hypertext Preprocessor"
- Scripts principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP
- Langage de programmation impérative
- Orienté-objet depuis la version 5
- Utilisation très répandue
  - Facebook, Yahoo!, ...
- Autres solutions: Java, ASP.NET, C#, Python, ...
- Interprété coté **SERVEUR**

