M2 CCI – M2 GEOMAS - 2026-2027

Cours
Technologies du Web

http://lig-membres.imag.fr/genoud/teaching/PL2AI/

Philippe GENOUD (LIG-STeamer)
Philippe.Genoud@imag.fr



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

A qui s'adresse l'enseignement TW ?

- Enseignement commun au M2 CCI et au M2 GEOMAS

- M2 CCI (Master 2 Compétences Complémentaires en Informatique)
  - *formation courte sur 1 an visant à fournir de solides connaissances de base en informatique à des étudiants ayant suivi un cursus scientifique (niveau M1 ou plus) autre que l'informatique*

- Un certain nombre des cours sont partagés avec le master GÉOMAS (GÉOMatique et Analyse Spatiale)
  - *formation en 2 ans (M1 et M2) visant à fournir des connaissances théoriques, méthodologiques et techniques sur les méthodes et outils dédiés au traitement, à la gestion, l'analyse, la visualisation et le partage de données spatialisées* ***avec une forte composante informatique***
    - *une partie des étudiants de M2 GEOMAS sont en apprentissage*
      - *De septembre à fin février*
        - *2 jours par semaine (lundi et mardi) en entreprise et 3 jours (mercredi, jeudi, vendredi) à l'université*
      - *En mars à temps plein à l'université avec les M2 CCI (Projet Intégration)*
      - *D'avril à septembre à temps plein en entreprise*

Ensemble des enseignements partagés M2 CCI &GEOMAS

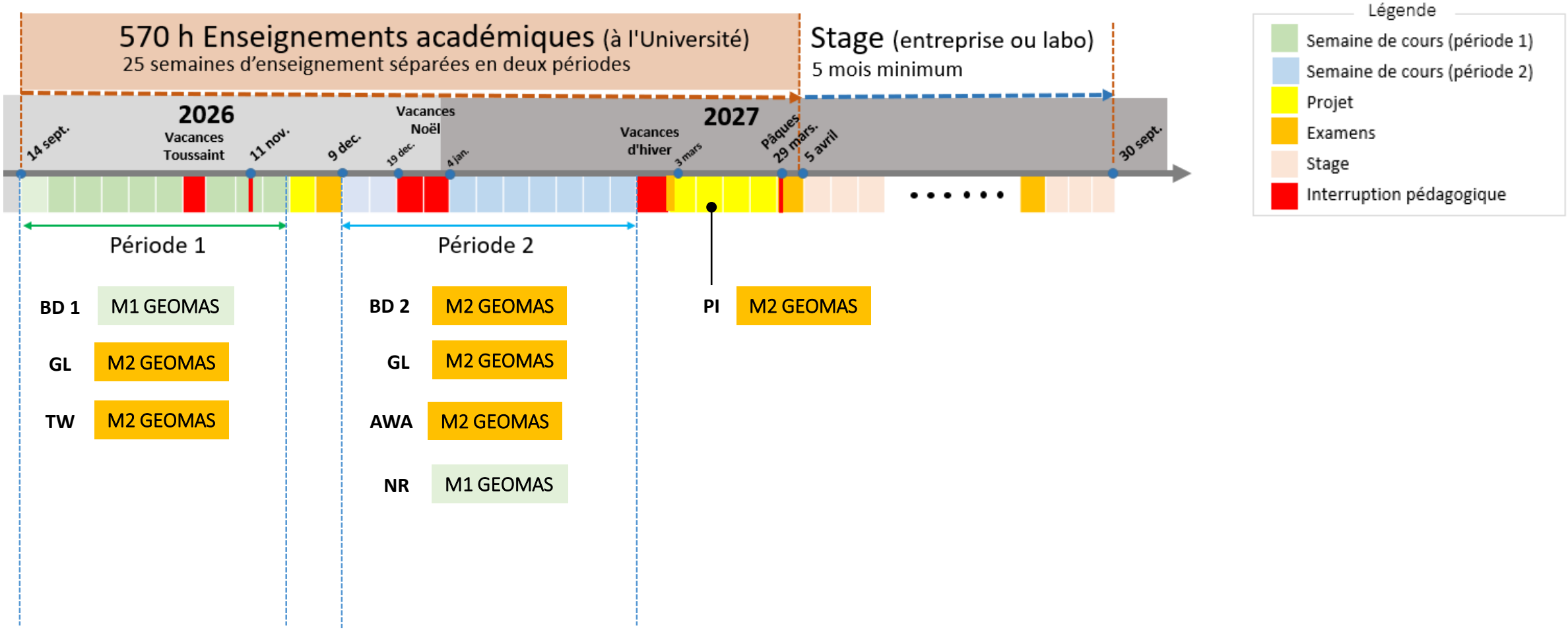
- M2 CCI : 570 h cours réparties en 8 UE (Unités d'Enseignement)

Semestre 9	30 ECTS	432 h		
Algorithmique (AL)	6 ECTS	93 h		
Programmation (PL) et Langage Machine (LM)	6 ECTS	78 h		
Bases de données (BD)	3 ECTS	45 h	M1 GEOMAS	M2 GEOMAS
Génie Logiciel (GL)	3 ECTS	45 h	M2 GEOMAS	
Systèmes (SYS) et Réseaux (RES)	3 ECTS	48 h		
Programmation et Applications Internet (PLAI)	6 ECTS	96 h		
• *Technologies du Web (TW)*	*3 ECTS*	*48h*	M2 GEOMAS	
• *Programmation Orientée Objets (POO)*	*3 ECTS*	*48h*		
Applications Web Avancées (AWA)	3 ECTS	27 h	M2 GEOMAS	
Semestre 10	**30 ECTS**	**138 h (hors stage)**		
Projet	6 ECTS	138 h		
• *Numérique Responsable (NR)*		*12h*	M1 GEOMAS	
• *Projet Intégration (PI)*		*126h*	M2 GEOMAS	
Stage	24 ECTS	5 mois minimum		

** ECTS : European Credits Transfer System*

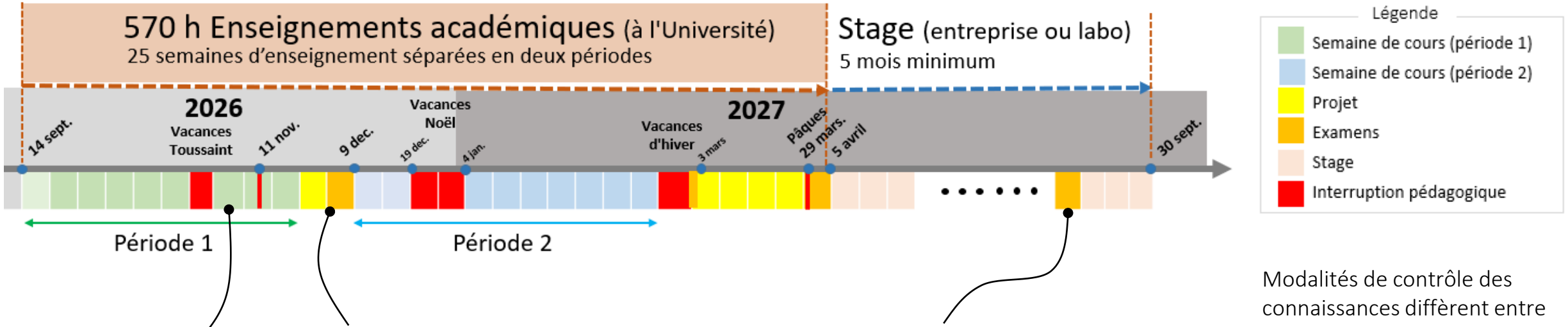
Calendrier de l'année

- Les cours sont répartis sur 2 périodes de 9 semaines chacune
- Chaque période est suivie d'un projet (une ou plusieurs semaines bloquées) puis d'examens
- Le cours Technologies du Web se déroule sur la période 1, l'examen ayant lieu à l'issue de celle-ci



Contrôle des connaissance du cours TW

- Les cours sont répartis sur 2 périodes de 9 semaines chacune
- Chaque période est suivie d'un projet (une ou plusieurs semaines bloquées) puis d'examens
- Le cours Technologies du Web se déroule sur la période 1, l'examen ayant lieu à l'issue de celle-ci



Contrôle Continu (note CC)

- pendant séance de TP *un devoir Surveillé (QCM, exercices de codage sur PC)*
- *+ éventuellement 1 ou 2 petits QCM*

Examen Terminal 1ère session (note ET1)

- semaine du 30 novembre 2026
- durée 3h
- *questions écrites*
- *exercices codage sur machine*

Examen 2ème session (note ET2)

- semaine début septembre 2027
- durée 3h
  - *questions écrites*
  - *exercices codage sur machine*

Modalités de contrôle des connaissances diffèrent entre M2 CCI et M2 GEOMAS

- M2 CCI : compensation possible avec autre matière de l'UE (Cours POO)
- M2 GEOMAS : ?

Note 1ère session
Note Finale = 30% Note CC + 70% Note ET1

Note 2ème session :
Note Finale = Max(Note ET2, 30% Note CC + 70% Note ET2)

Attention, les épreuves sur machine (QCM, codage) se font sur les machines de l'UFR IM²AG en environnement restreint

Compétences visées pour le cours TW

- L'objectif global de cet enseignements est double :
  - *Fournir une compréhension générale du fonctionnement du web et des technologies de base sur lesquelles il s'appuie à savoir les langages*
    - *HTML*
    - *CSS*
    - *JavaScript*
  - *Renforcer les compétences en programmation au travers d'un apprentissage approfondi du langage JavaScript.*

A l'issue de ce cours, les étudiants doivent être capables de

- concevoir un site web statique intégrant des éléments d'interaction gérés par JavaScript dans le navigateur web.
- d'utiliser JavaScript dans l'environnement Node.js

Detail contenu

Articulation avec les autres enseignements

- Le cours TW sera complété par d'autres enseignements :
  - *Applications Web avancées (AWA),*
    - *partagé avec les M2 GEOMAS.*
    - *introduction de notions plus avancées de JavaScript (programmation asynchrone, API fetch, exceptions, modules ...) ainsi que le développement d'applications Web modernes au travers d'une introduction au framework réactif Vue.js*
  - *Programmation Orientée Objet (POO), introduction au langage Java*
  - *Projet d'intégration (PI)*
    - *partagé avec les M2 GEOMAS.*
    - *Mobilisera l'ensemble des connaissances acquises en BD, GL, TW, AWA, POO, NR + BD spatiales + Web Mapping, Analyse spatiale*

Objectif global :

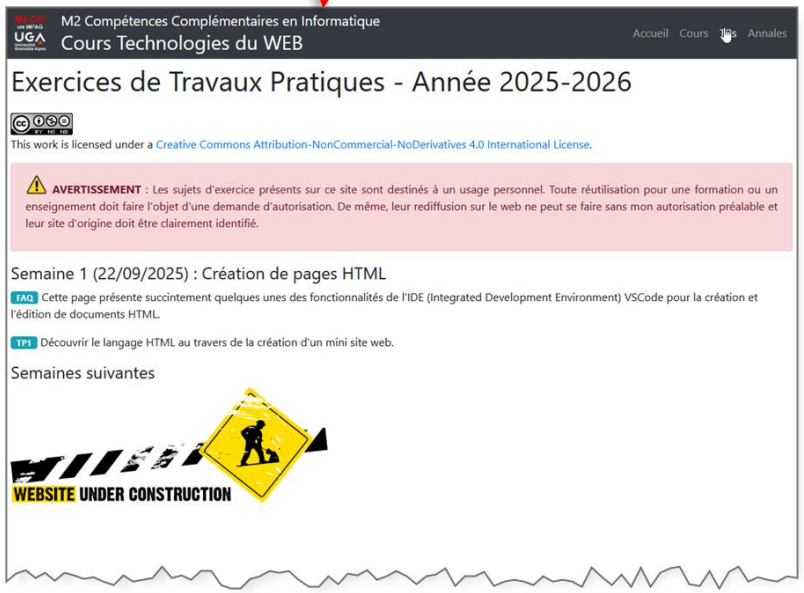
- acquérir une compréhension globale du fonctionnement des applications Web et avoir suffisamment de connaissances en programmation pour pouvoir aborder sereinement l'ensemble des activités de développement logiciel.
- sans être un spécialiste de telle ou telle technologie, avoir toutes les clés pour pouvoir s'approprier plus facilement, selon sa propre évolution future, les langages, cadriciels (frameworks) et outils nécessaires au développement d'applications modernes.

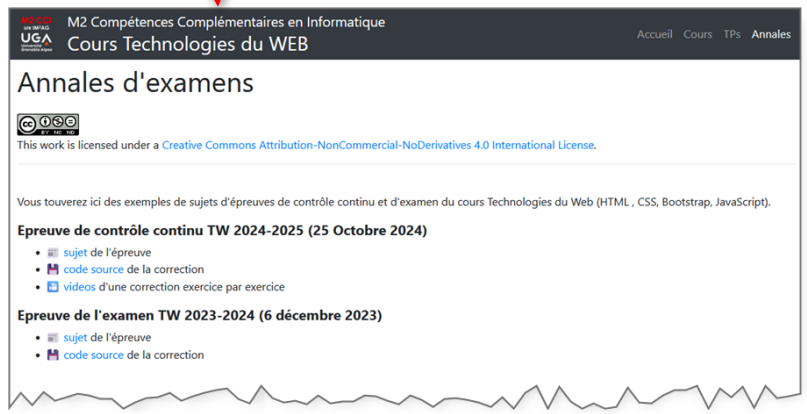
Site Web du cours

http://lig-membres.imag.fr/genoud/teaching/PL2AI/









Objectifs de cette 1ère séance

- Avoir un vision globale des points qui seront abordés dans les cours TW puis AWA
  - Comprendre les principes de fonctionnement de l'internet
  - Comprendre l'architecture du World Wide Web et le rôle de ses différents composants



Sondage

Parmi tous les termes de ce nuage de mots, pour combien seriez-vous capable de donner une définition ?

A	Tous ou presque tous (plus de 26 - 80% à 100%)
B	Une bonne partie (de 19 à 26 - 60% à 80%)
C	Environ la moitié (de 13 à 19 - 40% à 60%)
D	Un petite partie (de 8 à 13 - 25% à 40%)
E	Très peu (moins de 8 -moins de 25%)

Internet / Web

- Internet et le web est-ce la même chose ?



=



?

Internet / Web

- Internet et le web est-ce la même chose ?



≠





https://www.youtube.com/watch?v=scWj1BMRHUA
durée 10:54min

Internet / Web

- Internet et le web est-ce la même chose ?



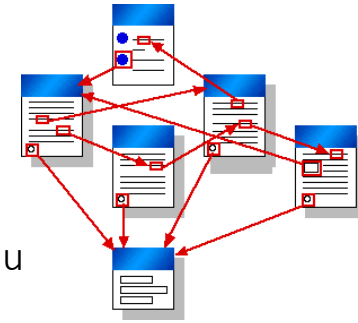
Internet n’est pas le Web

- **Internet** : un réseau informatique mondial qui permet aux ordinateurs qui y sont connectés de communiquer entre eux et d'échanger de l'information.



- c'est l'**infrastructure** globale, sur laquelle s'appuient de nombreux autres services pour échanger des données.
  - *courrier électronique (e-mail)*
  - *échange de fichiers (FTP File Transfert Protocole)*
  - *le Web (World Wide Web)*

- **Web** : un système conçu au départ pour le partage et l'échange de documents multimédia hypertextes (des documents liés entre eux par des hyperliens) en utilisant le réseau internet



Un peu d'histoire

1961
Leonard Kleinrock (MIT) théorise la **communication par paquets**

1971
1er **e-mail** envoyé

1973
1ère connexion internationale vers l'université de Londres

1983
Adoption du protocole **TCP/IP,** du système **DNS** et du mot « **Internet** »

2003
Tim O'Reilly consacre l'expression « **Web 2.0** »

Militaire → **Universités** → **Grand public**

1957
Création de l'**ARPA*** du DoD des USA

*Advanced Research Projects Agency Department of Defense

1969
Création d'**ARPANET**

1974
Définition du protocole **TCP/IP**

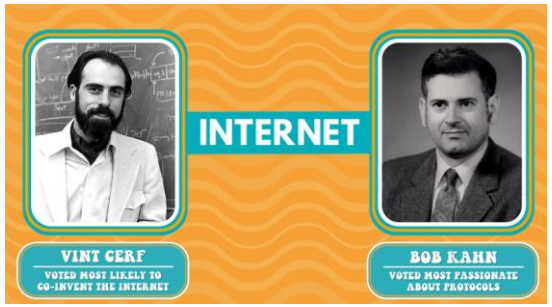
1980
Création du protocole **FTP**

1990
Disparition d'ARPANET et naissance du **World Wide Web** avec Tim Berners-Lee et le protocole **HTTP**

1994
Naissance du **W3C**

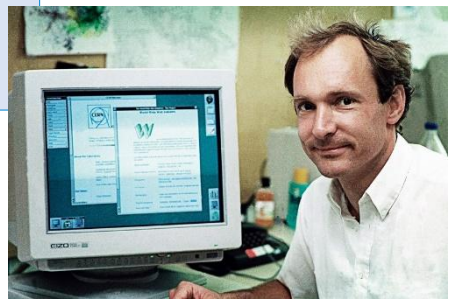
Then
« **Web 3.0** » ?

Now
« **Web 4.0** » ?



code.org
The Internet:
IP Addresses & DNS

https://www.youtube.com/watch?v=5o8CwafCxnU&index=3&list=PLzdnOPI1iJNfMRZm5DDxco3UdsFegvuB7



The birth of the web

https://home.cern/science/computing/the-birth-of-the-web/

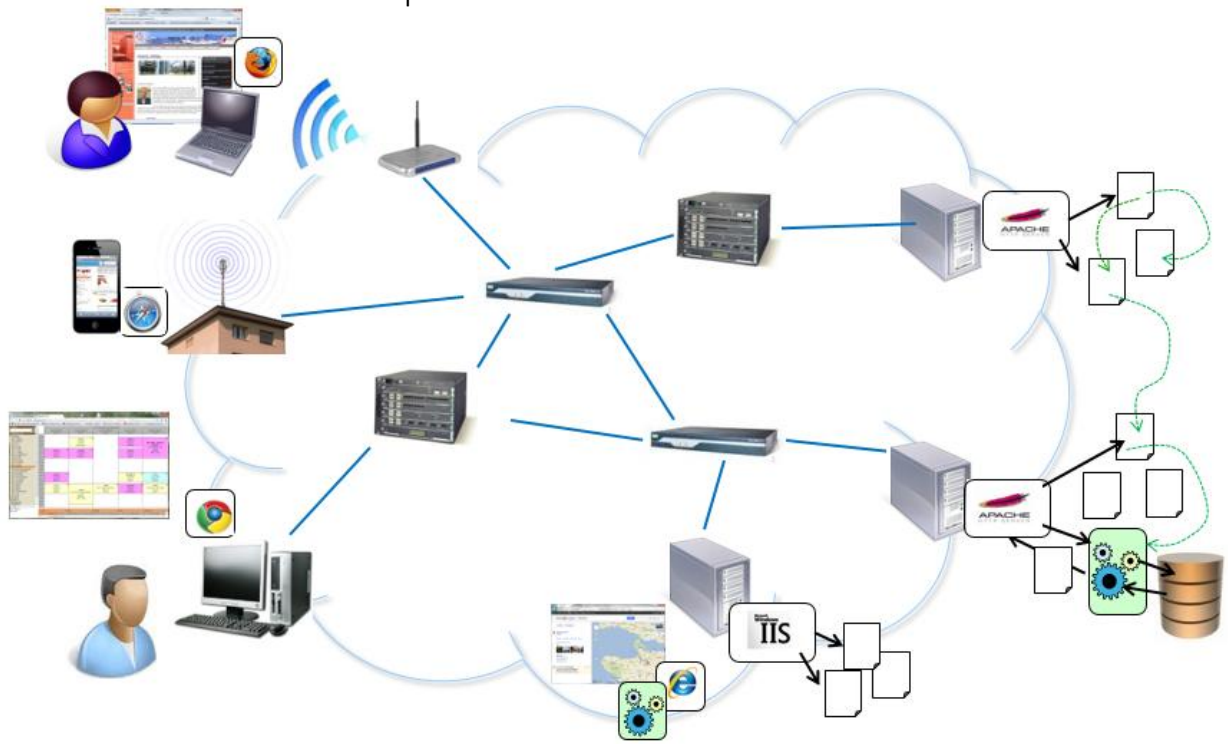
© Philippe GENOUD - UGA- M2 CCI - M2 GEOMAS - Technologies du Web 2026-2027

Objectifs de cette 1ère séance

- Avoir un vision globale des points qui seront abordés dans les cours TW puis AWA
  - Comprendre les principes de fonctionnement de l'internet
  - Comprendre l'architecture du World Wide Web et le rôle de ses différents composants



prendre chacun des termes du nuage de mots, le définir et le positionner sur le schéma ci dessous

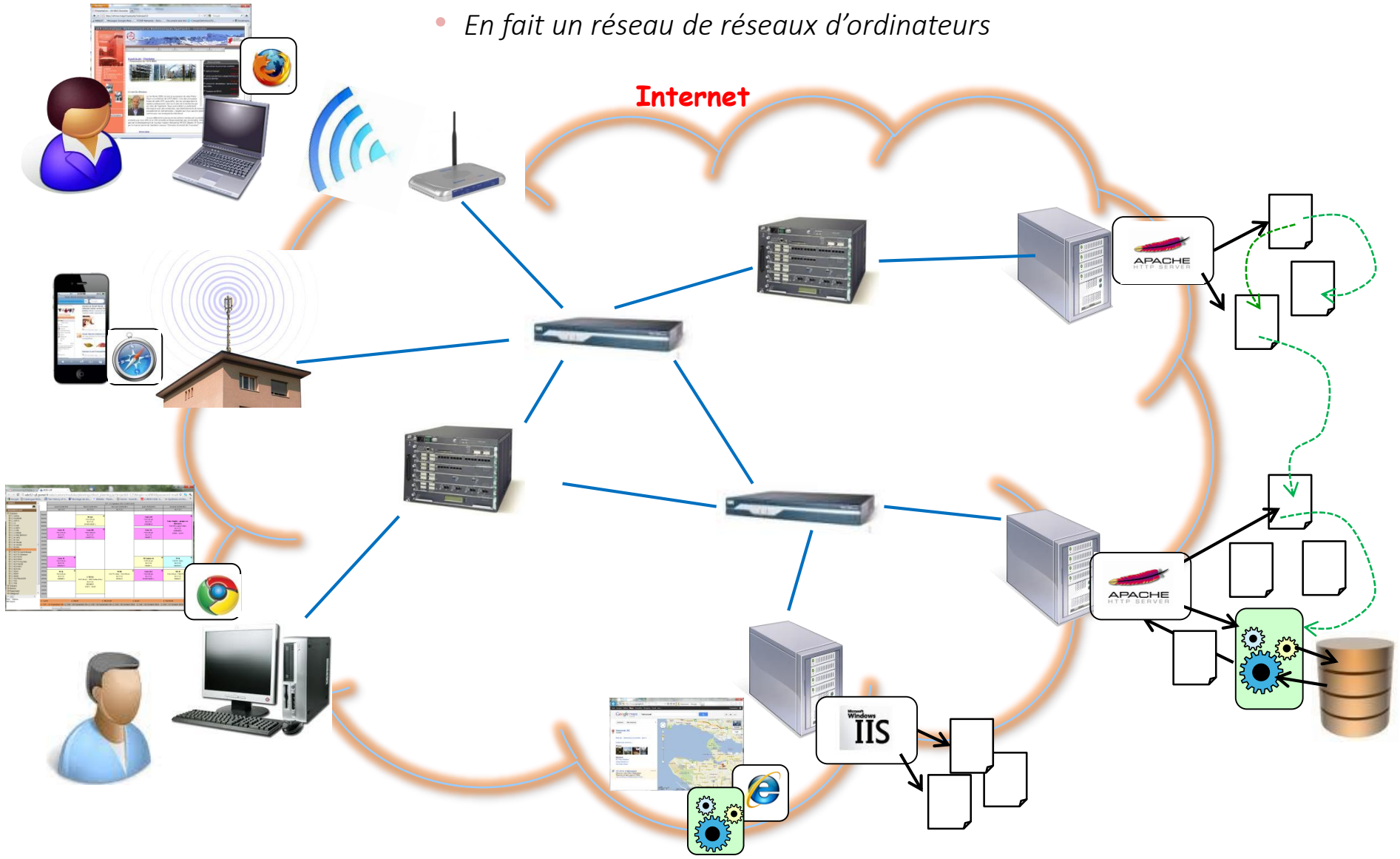


Internet


URL
Serveur-d'applications
IDE
WHATWG
API
Pages-dynamiques
REST
AJAX
Routeur
Java
HTTP
W3C
JSP
CSS
Web
Client-Web
Lien
Hypertexte
Script
Internet
PHP
Numéro-IP
Protocole
JavaScript
TCP/IP
Pages-statiques
Lien
Serveur-Web
Serveur
HTML
Servlets
CMS
DNS
Nom-IP
Framework

Internet

- réseau mondial connectant des ordinateurs et différents appareils électronique (smartphones, capteurs…)
  - *En fait un réseau de réseaux d'ordinateurs*



Internet

- réseau mondial connectant des ordinateurs
  - *En fait un réseau de réseaux d'ordinateurs*



Connexions : câble téléphonique, câble coaxial, fibre optique, câble téléphonique, liaison sans fils (ondes radio, infrarouge, micro-ondes, satellite)

Réseau local (Ethernet) (LAN Local Area Network)
ex : réseau UFR IM²AG, réseau du LIG

Réseau sans fils (WLAN Wireless Local Area Network)
ex : wifi-campus

WAN (Wide Area NetWork)
ex: grenet

RENATER Réseau national de télécommunications pour la technologie, l'enseignement et la recherche

Réseau satellitaire
ex : StarLink

5G
4G

ADSL
Réseau téléphonique
Fibre

FAI
(Fournisseur d'accès Internet)

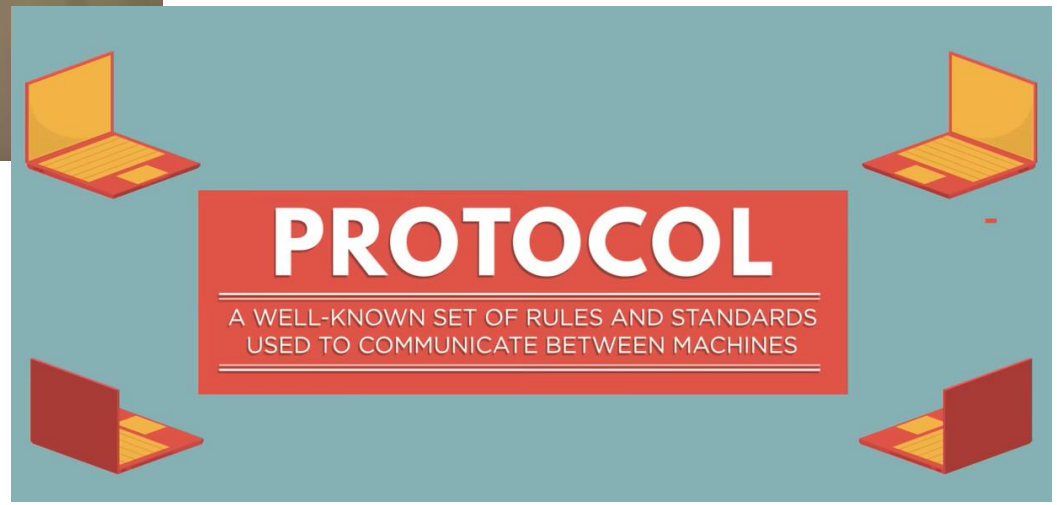
Internet

- Pour pouvoir communiquer sur des réseaux différents les ordinateurs doivent utiliser un même langage : **protocole de communication**





Vint Cerf
Code.org : The Internet: IP Addresses & DNS
https://youtu.be/5o8CwafCxnU?list=PLzdnOPI1iJNfMRZm5DDxco3UdsFegvuB7&t=39



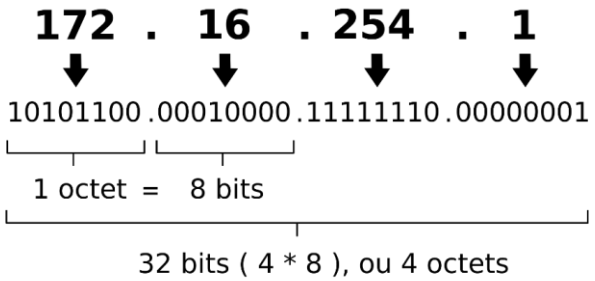
- TCP/IP : Transfert Control Protocol – Internet Protocol
  - *Le protocole de communication entre les ordinateurs connectés à internet*
  - *Transmission par paquets*
  - *Mis au point dans les années 70 par Vint Cerf et Bob Khan*

Technologies de l'internet

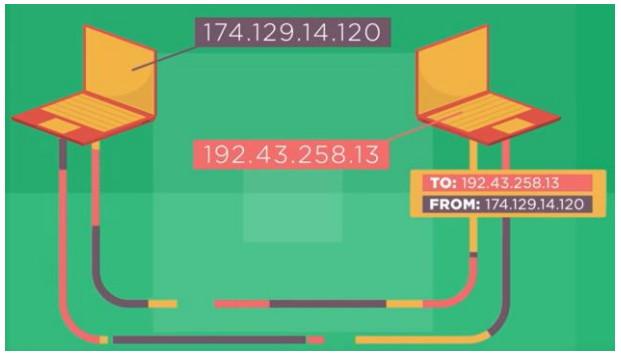


Internet Protocol : Numéro IP

- Adresse ou Numéro IP (Internet Protocol) : identifie par une adresse unique chaque ordinateur connecté au réseau
  - *IPv4 adresse sur 32 bits (4 octets)* En théorie 2^{32} = 4 294 967 296 adresses différentes peuvent être définies
    - *xxx.xxx.xxx.xxx* (*xxx représente un nombre entre 0 et 255)*



source: https://fr.wikipedia.org/wiki/IPv4



décimal : base 10

2	1	6
2×10^2	1×10^1	6×10^0
= 200	= 10	= 6

200 + 10 + 6

= 216

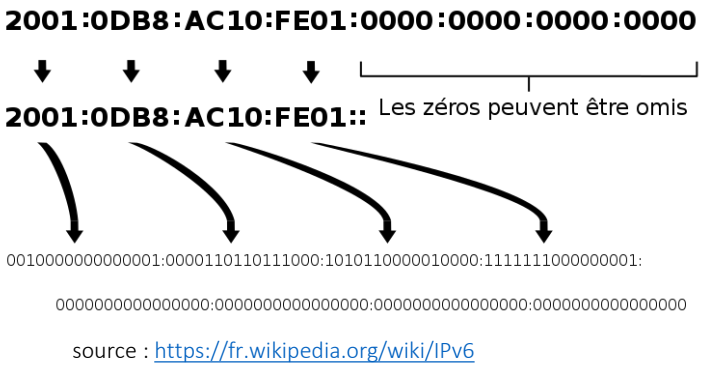
binaire : base 2

1	1	0	1	1	0	0	0
1×2^7	1×2^6	0×2^5	1×2^4	1×2^3	0×2^2	0×2^1	0×2^0
= 128	= 64	= 0	= 16	= 8	= 0	= 0	= 0

128 + 64 + 0 + 16 + 8 + 0 + 0 + 0

= 216

  - *IPv6 adresse sur 128 bits* En théorie 2^{128} , un peu plus de 340 sextillions (10^{36}) adresse différentes
    - *2001:0db8:0000:85a3:0000:0000:ac1f:8001* *(8 groupes de 2 octets, écriture en héxadecimal, base 16)*



source : https://fr.wikipedia.org/wiki/IPv6

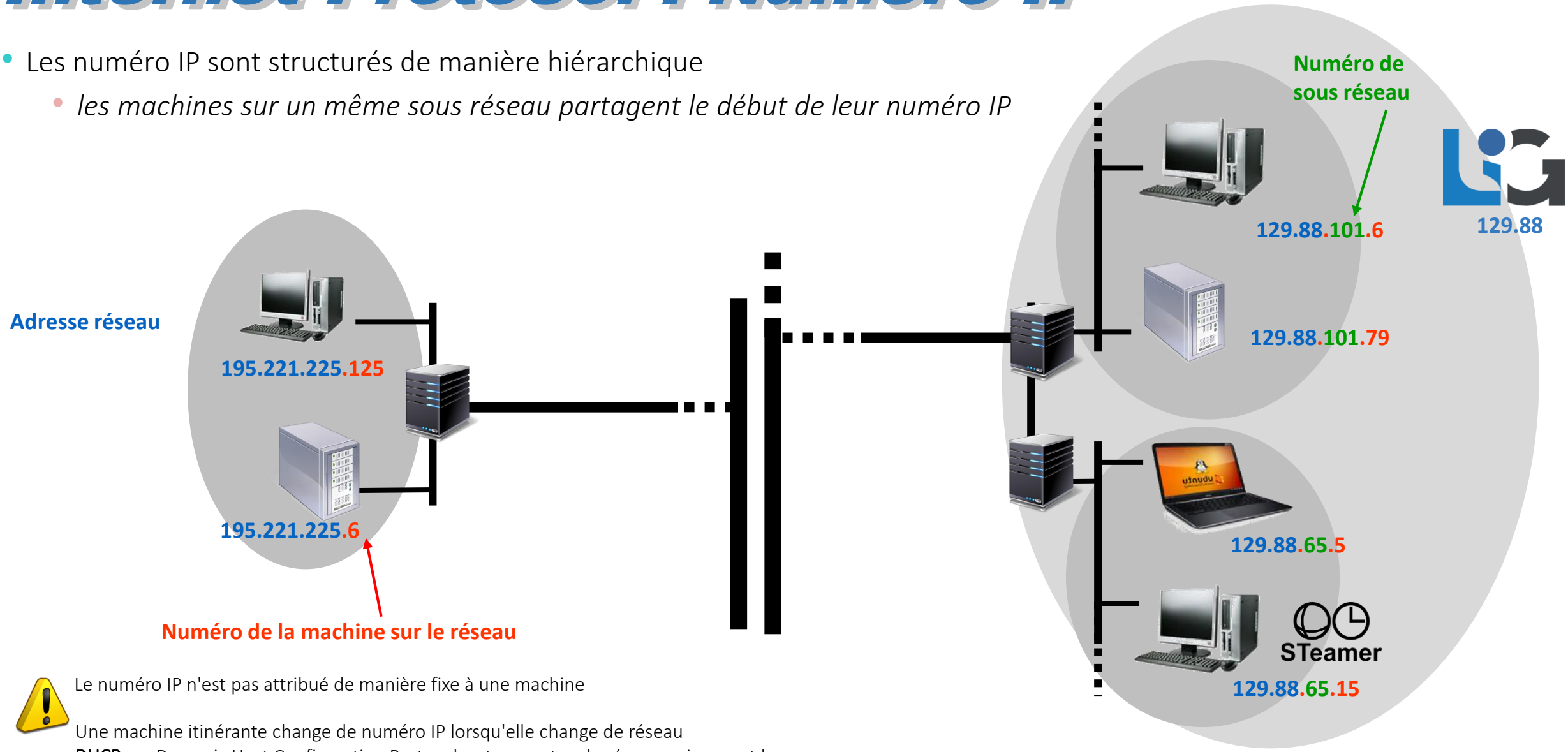
hexadécimal : base 16

Nombre en binaire	10101101010110011110111						
Groupes de 4 chiffres binaires, en partant de la droite	1	0101	1010	1010	1100	1111	0111
Chiffres hexadécimaux	1	5	A	A	C	F	7
Nombre en hexadécimal	15AACF7						
Nombre en décimal (cf. ci-dessous)	22719735						

sources :
https://mikkegoes.com/computer-science-binary-code-explained/
https://fr.wikipedia.org/wiki/Syst%C3%A8me_hexad%C3%A9cimal

Internet Protocol : Numéro IP

- Les numéro IP sont structurés de manière hiérarchique
  - *les machines sur un même sous réseau partagent le début de leur numéro IP*

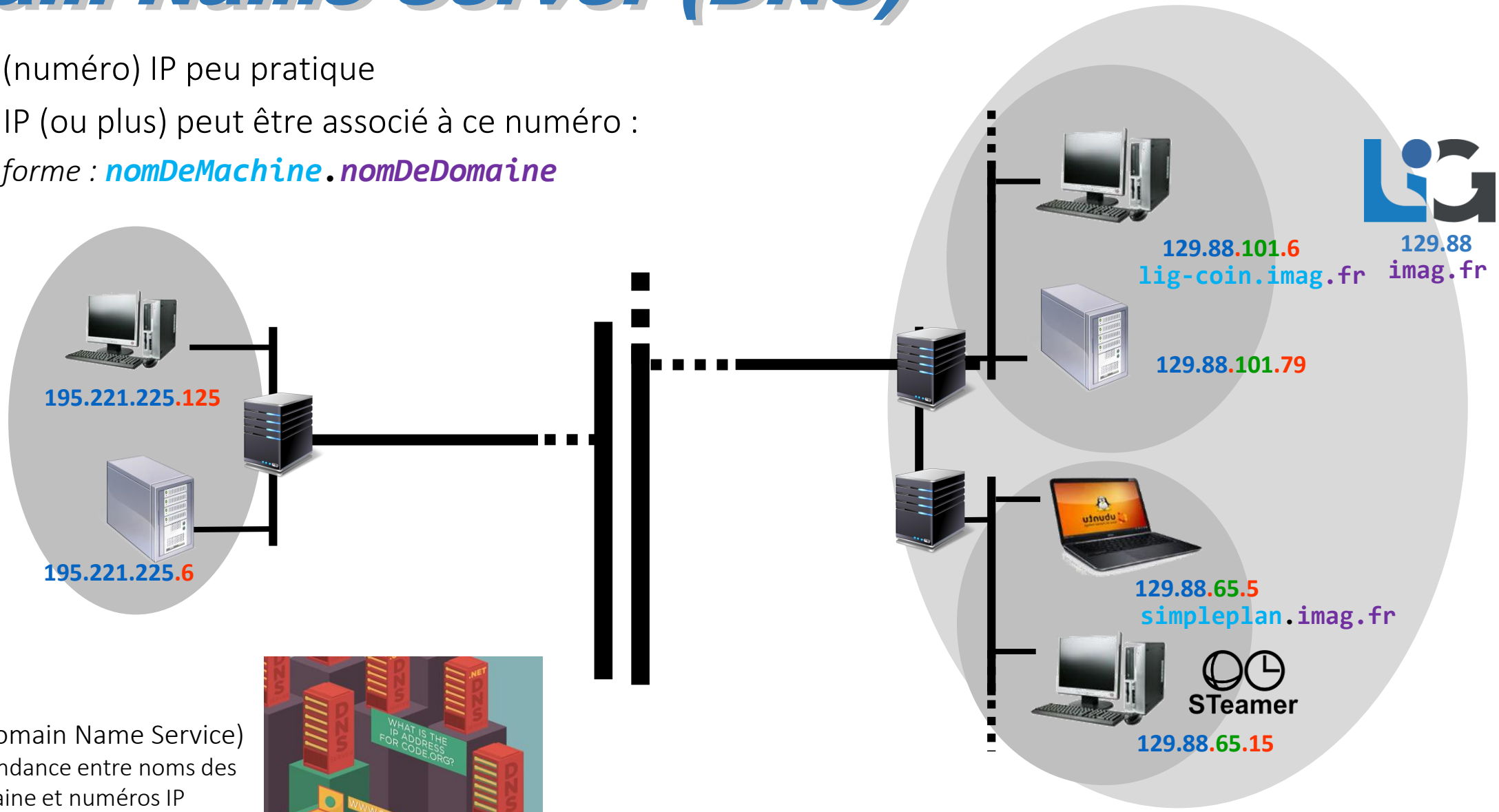


Le numéro IP n'est pas attribué de manière fixe à une machine

Une machine itinérante change de numéro IP lorsqu'elle change de réseau
DHCP, ou Dynamic Host Configuration Protocol, est un protocole réseau qui permet la configuration automatique des adresses IP d'un réseau informatique

Domain Name Server (DNS)

- Adresse (numéro) IP peu pratique
- Un nom IP (ou plus) peut être associé à ce numéro :
  - *de la forme : **nomDeMachine.nomDeDomaine***

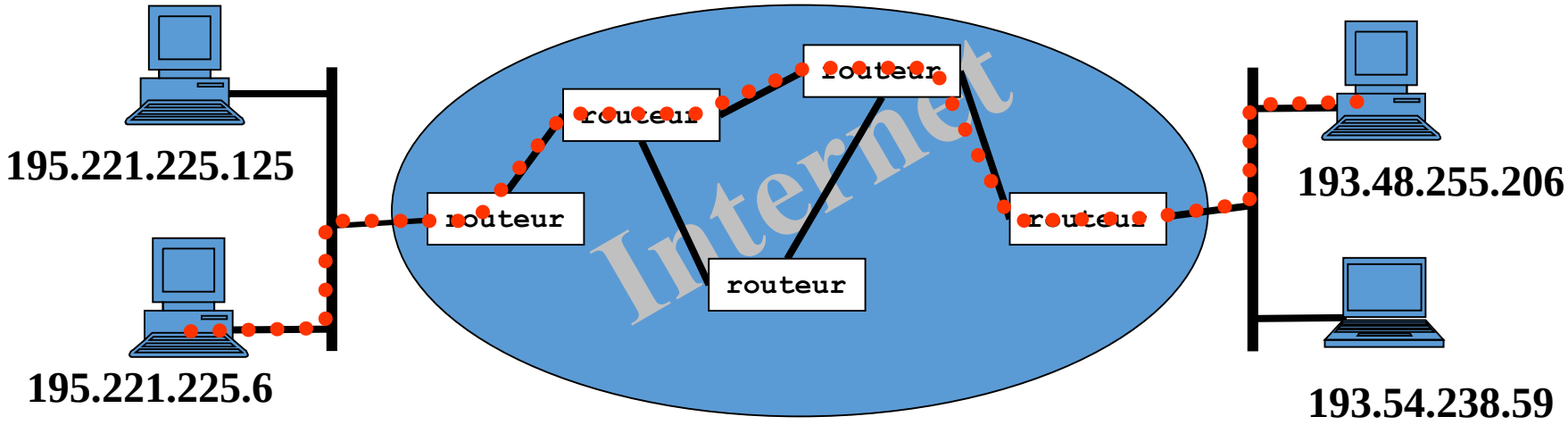


Serveurs DNS(Domain Name Service)
Assurent correspondance entre noms des machines du domaine et numéros IP

https://youtu.be/5o8CwafCxnU?list=PLzdnOPI1iJNfMRZm5DDxco3UdsFegvuB7&t=264

Transport Control Protocol : TCP

- Routeurs :
  - *Relais de l'information : charger de transporter l'information à travers le réseau internet*
- Passerelles :
  - *Machines qui interconnectent deux réseaux*



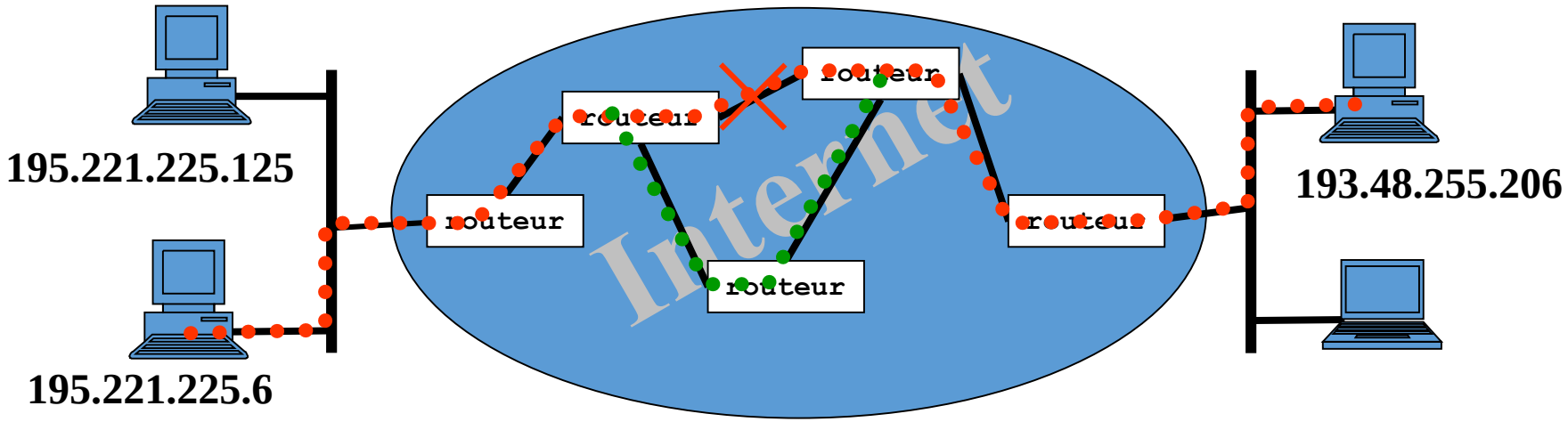
Code.org: The Internet: Packets, Routing & Reliability
https://www.youtube.com/watch?v=AYdF7b3nMto&feature=youtu.be



Transport Control Protocol : TCP

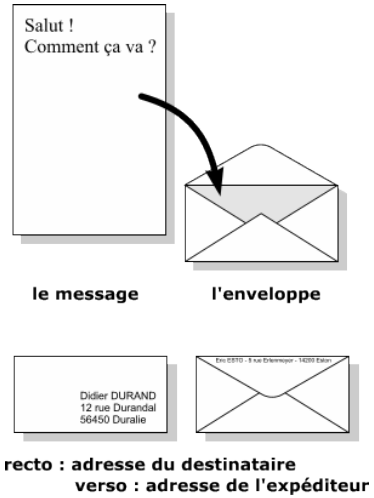
- En cas de rupture d'une connexion . . .
  - . . . le réseau peut modifier l'acheminement (le routage) de l'information

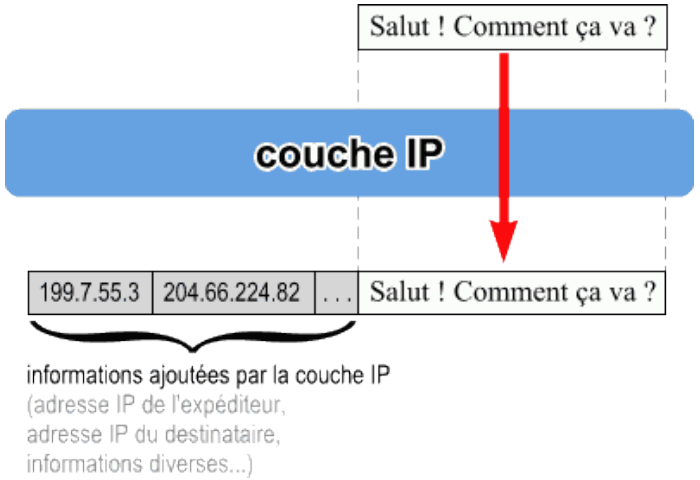
  ➔ FIABILITE (Reliability)

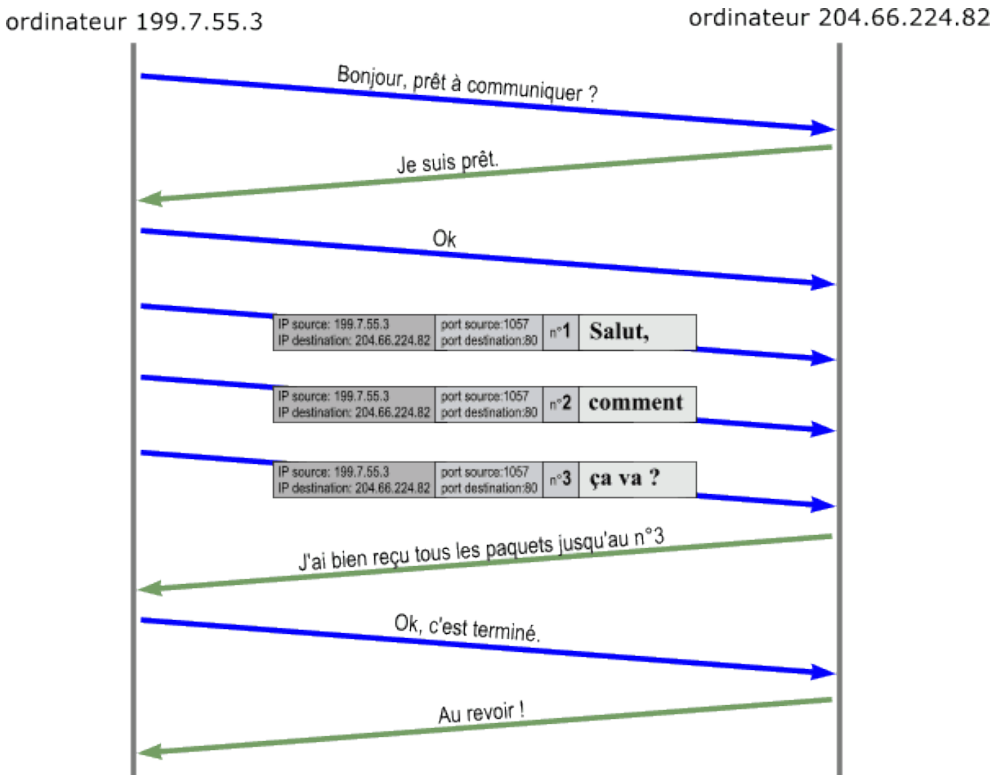


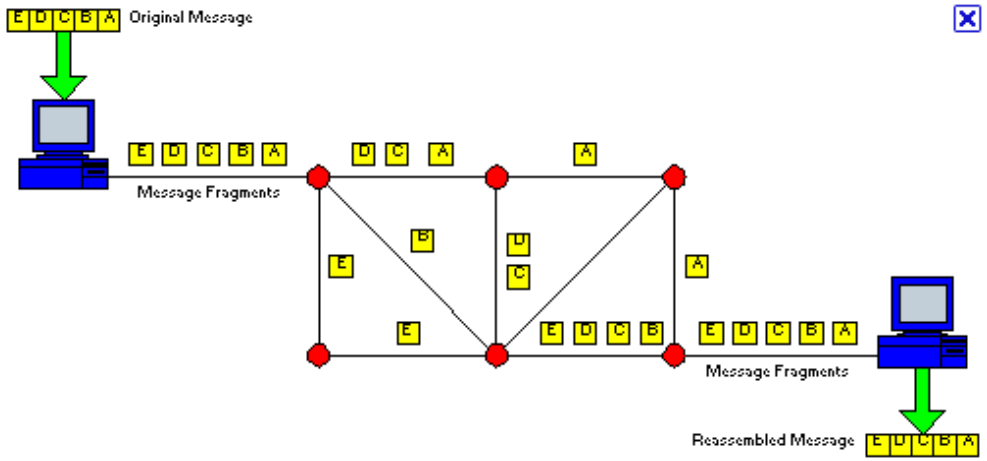
Transport Control Protocol : TCP

- TCP/IP
  - Basé sur commutation de paquets (Packets Switching)
  - Information découpée en paquets transmis séparément
  - Les paquets ne suivent pas nécessairement la même route



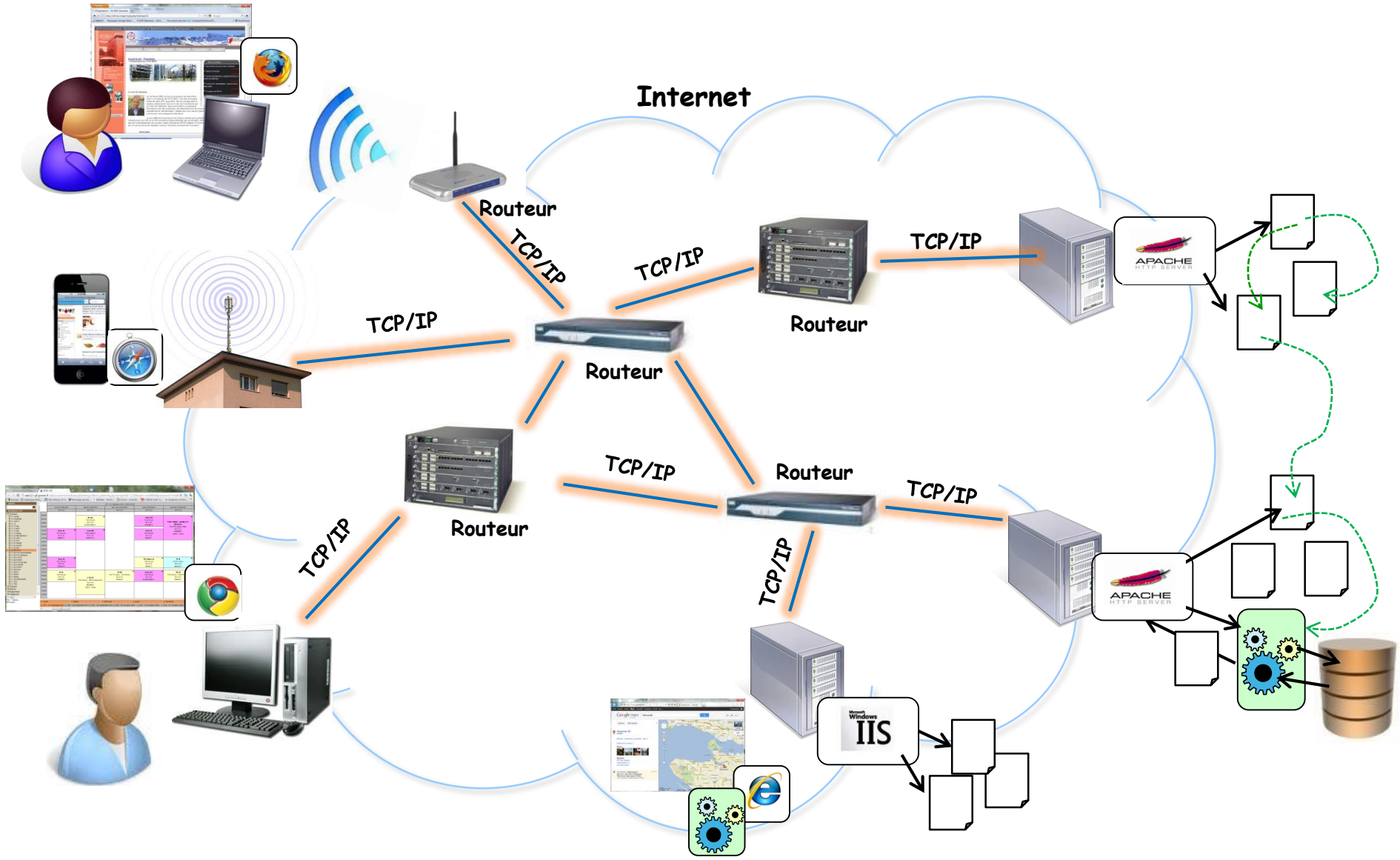






http://www.pbs.org/opb/nerds2.0.1/geek_glossary/packet_switching_flash.html

Technologies de l'internet



Le Web

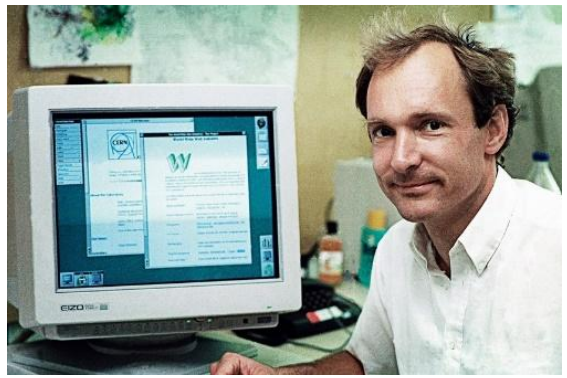

URL
Serveur-d'applications
IDE
WHATWG
API
Pages-dynamiques
REST
AJAX
W3C
Routeur
Java
HTTP
Client-Web
JSP
CSS
Web
Hypertexte
Lien
Internet
Script
PHP
Numéro-IP
Protocole
JavaScript
TCP/IP
Pages-statiques
Lien
Serveur-Web
Serveur
HTML
Servlets
CMS
DNS
Nom-IP
Framework

Le Web

- The **World Wide Web** (abbreviated as WWW or W3 and commonly known as the Web) is a **system of interlinked hypertext documents** accessed via the Internet. *(wikipedia)*

  - *Qu'est ce que l'hypertexte ?*
    - *Définition : document non linéaire constitué de pages permettant de passer de l'une à l'autre par des liens (liens hypertexte)*

  - *Web = hypertexte + internet*
    - *Inventé au début des années 90 par Tim Berners Lee au CERN (Genève) en développant trois principales technologies :*
      - *les adresses Web (ou URL Uniform Resource Locator),*
      - *le protocole HTTP (HyperText Transfer Protocol),*
      - *et le langage HTML (HyperText Markup Language).*





World Wide Web

The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists , Policy , November's W3 news , Frequently Asked Questions .

What's out there?
Pointers to the world's online information, subjects , W3 servers, etc.
Help
on the browser you are using
Software Products
A list of W3 project components and their current state. (e.g. Line Mode ,X11 Viola , NeXTStep , Servers , Tools , Mail robot , Library)
Technical
Details of protocols, formats, program internals etc
Bibliography
Paper documentation on W3 and references.
People
A list of some people involved in the project.
History
A summary of the history of the project.
How can I help ?
If you would like to support the web..
Getting code
Getting the code by anonymous FTP , etc.

http://info.cern.ch/hypertext/WWW/TheProject.html

Page d'accueil du 1er site (ensemble de pages) web mis en ligne par le CERN en décembre 1990

source : The birth of the web
https://home.cern/science/computing/birth-web

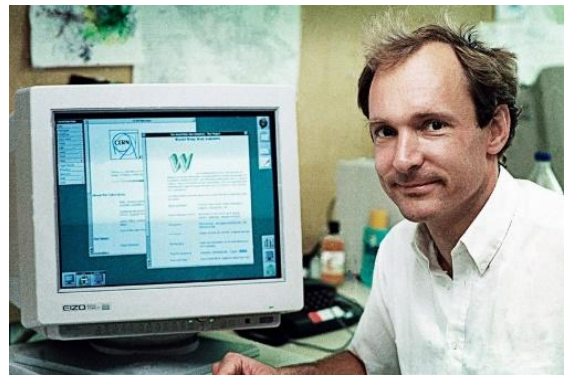
Le Web

- The **World Wide Web** (abbreviated as WWW or W3 and commonly known as the Web) is a **system of interlinked hypertext documents** accessed via the Internet. *(wikipedia)*

  - *Qu'est ce que l'hypertexte ?*
    - *Définition : document non linéaire constitué de pages permettant de passer de l'une à l'autre par des liens (liens hypertexte)*

  - *Web = hypertexte + internet*
    - *Inventé au début des années 90 par Tim Berners Lee au CERN (Genève)*

  - *Web ≠ Internet qui supporte de nombreux autres services*
    - *messagerie électronique (ou e-mail) : POP ou IMAP*
    - *transfert de fichiers : FTP*
    - *listes de discussion : "news"*
    - *forum de discussion en ligne : IRC (Internet Relay Chat)*
    - *connexion sur une machine distante : Telnet, rlogin, ssh (Secure Shell)*
    - *…*



Web : W3C - WHATWG



Web : W3C

- **W3C** : World Wide Web Consortium



  - *organisme de **standardisation** à but non-lucratif,*
  - *fondé en octobre 1994*
  - *consortium chargé de promouvoir la compatibilité des technologies du World Wide Web telles que*
    - *HTML, XHTML, HTML5*
    - *XML,*
    - *RDF,*
    - *CSS,*
    - *PNG, SVG*
    - *SOAP…*
  - *standards **indispensables** au bon fonctionnement du Web*

Standardisation du Web : W3C

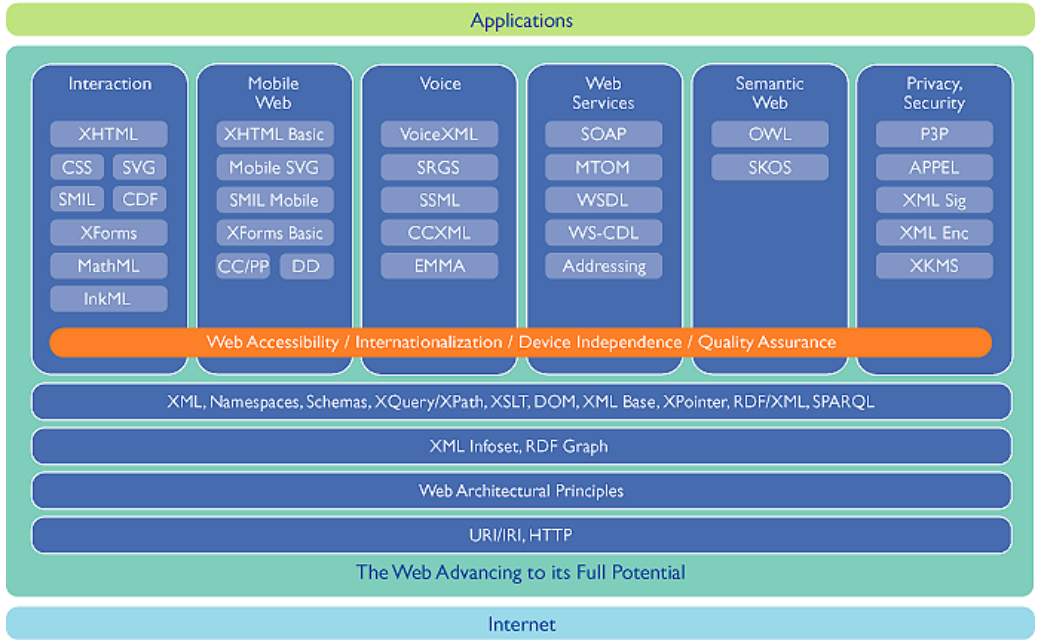
- 1994 creation of the W3C
- International consortium, around 400 Members
- Standardization of core Web technologies, publication of guidelines, technical notes, etc.

Based on a transparent and public Process



http://www.w3.org/2013/dd-epasorg.htm#%288%29

Standardization activities (2008)

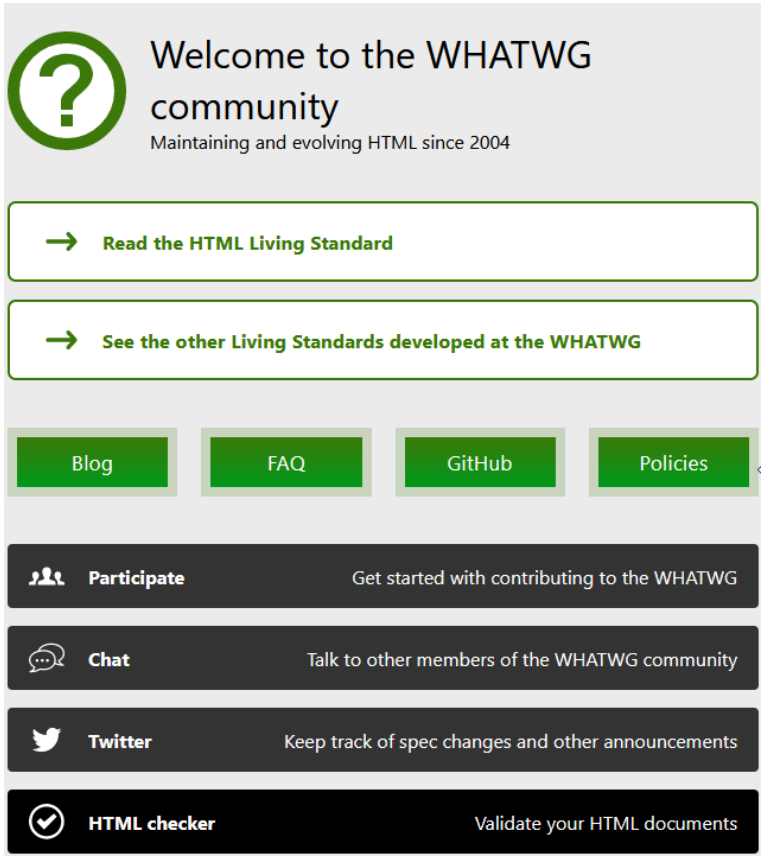


W3C All Standards and Drafts

https://www.w3.org/TR/

Standardisation du Web : WHATWG

WHATWG



- Web Hypertext Application Technology Working Group
  https://whatwg.org/
  - *Communauté de personnes intéressées dans l'évolution de HTML et des technologie reliées*
  - *2004 : Fondation par des membres d'Apple, de la fondation Mozilla et d'Opéra Software impliqués dans le développement de navigateurs Web*
    - *Groupe formé en réaction*
      - *à la lenteur supposée du développement des standards par le W3C*
      - *au caractère supposé trop fermé de son processus interne d'élaboration de spécification.*
  - *2007 groupe de travail HTML du W3C adopte propositions de WHATWG comme base de travail du futur HTML5*
  - *2017 Google et Microsoft rejoignent le WHATWG*
  - *2019 W3C et WHATWG se mettent d'accord pour que le WHATWG soit le seul éditeur des standards HTML et DOM (Document Object Model)*

Web : HTML – liens hypertexte



HTML: Hyper Text Mark Up Language

- Web = ensemble de pages hypertextes

https://lig-membres.imag.fr/genoud/teaching/exemple1/index.html

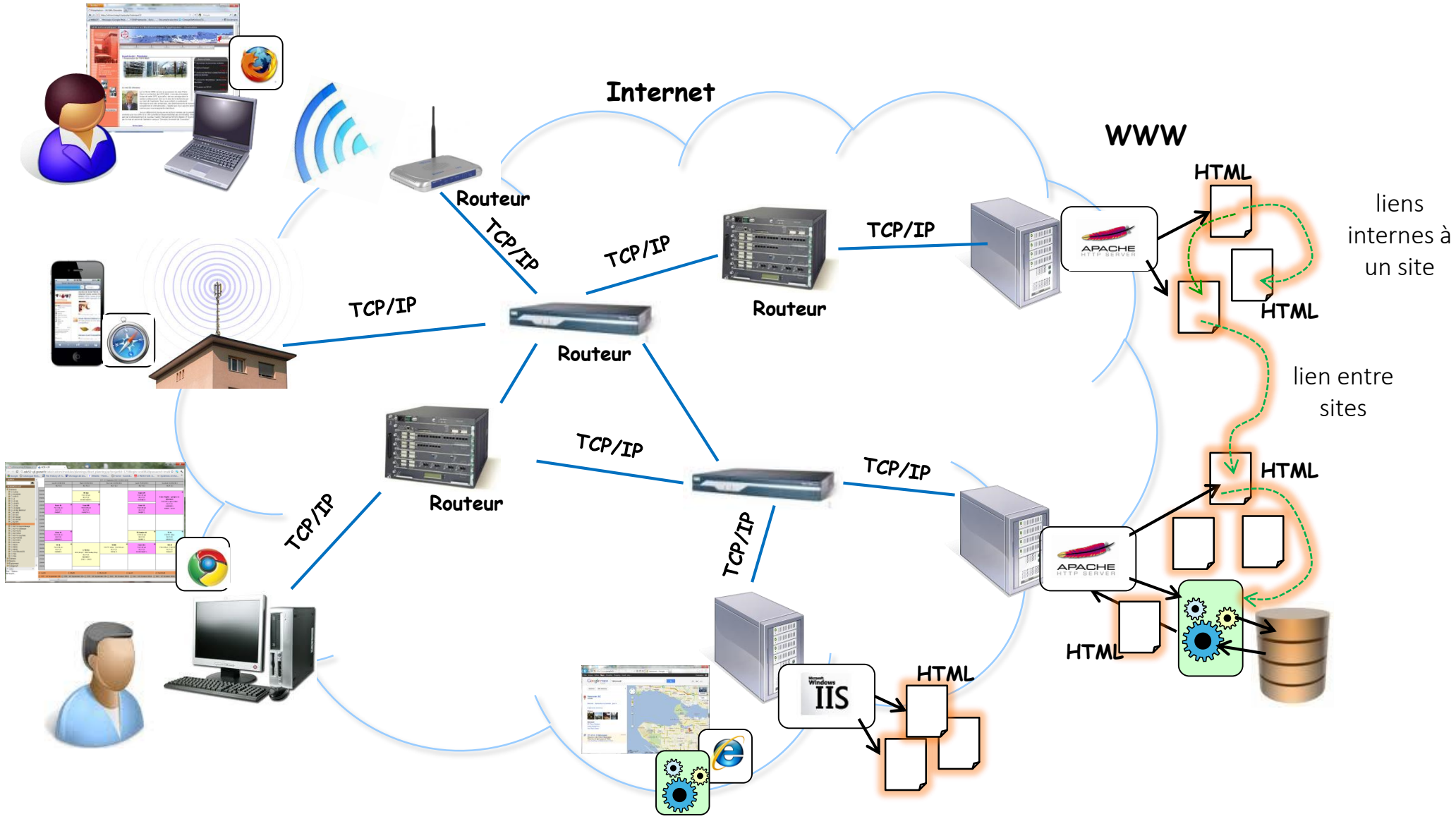
fichier source HTML

```
<!DOCTYPE html>
<html>
    <head>
        <title>Exemple HTML5</title>
        <meta charset="UTF-8">
    </head>
    <body>
        <h1>Le HTML5 c'est facile</h1>
        <hr/>
        <p>Dans un doucment HTML, on peut mettre :</p>
        <ul>
            <li>
                des liens hypertext pour naviguer :
                en cliquant <a href="http://www.w3schools.com/html/">ICI</a>
                vous accédez au tutorial du W3C sur HTML5.
            </li>
            <li>
                des <b>images</b> pour faire joli.
                <br/>
                <img src="./images/logo-uga.png" alt="le logo uga">
            </li>
            <li>
                <p>et même des vidéos !!</p>
                <iframe width="320" height="180" src="https://www.youtube.com/embed/kBXQZMmiA4s
            </li>
        </ul>
        <p>
            <a href="http://validator.w3.org/check?uri=referer">
              <img src="images/html5validation.png" alt="Valid HTML5" height="31" width="88">
            </a>
        </p>
    </body>
</html>
```

Description textuelle
Balises (tags) = structure + directives de présentation
Contenu = textuel + multimédia
Liens : un document peut faire référence à d'autres documents



Web : HTML – liens hypertexte



Web : CSS


URL
Serveur-d'applications
IDE
API
Pages-dynamiques
REST
AJAX
Java
HTTP
W3C
Routeur
JSP
CSS
Web
Client-Web
Lien
Hypertexte
Internet
Script
PHP
Numéro-IP
Protocole
JavaScript
TCP/IP
Pages-statiques
Lien
Serveur-Web
Serveur
HTML
Servlets
CMS
DNS
Nom-IP
Framework

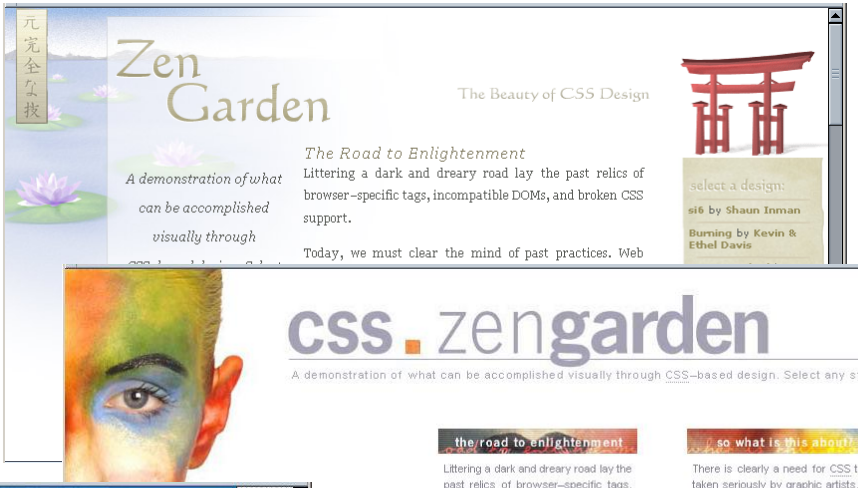
Web : CSS (Cascading Style Sheets)

- séparer contenu de la présentation
  - *HTML -> structure et contenu du document*
  - *CSS -> présentation du document*

```
html {
  font-family: sans-serif;
  -ms-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
}
body {
  margin: 0;
}
a:focus {
  outline: thin dotted;
}
a:active,
a:hover {
  outline: 0;
}
```

```
/* basic elements */
html {
    margin: 0;
    padding: 0;
    }
body {
    font: 75% georgia, sans-serif;
    line-height: 1.88889;
    color: #555753;
    background: #fff url(http://csszengarden.com/001/blossoms.jpg) no-repeat bottom right;
    margin: 0;
    padding: 0;
    }
p {
    margin-top: 0;
    text-align: justify;
    }
```









http://www.csszengarden.com/

Un même document HTML
des feuilles CSS différentes

Web : CSS (Cascading Style Sheets)

- séparer contenu de la présentation
  - *HTML -> structure et contenu du document*
  - *CSS -> présentation du document*

```
/* basic elements */
html {
    margin: 0;
    padding: 0;
    }
body {
    font: 75% georgia, sans-serif;
    line-height: 1.88889;
    color: #555753;
    background: #fff url(http://csszengarden.com/001/blossoms.jpg) no-repeat bottom right;
    margin: 0;
    padding: 0;
    }
p {
```

style1.css

```
<body id="css-zen-garden">
<div class="page-wrapper">

    <section class="intro" id="zen-intro">
        <header role="banner">
            <h1>CSS Zen Garden</h1>
            <h2>The Beauty of <abbr title="Cascading Style Sheets">CSS</abbr> Design</h2>
        </header>

        <div class="summary" id="zen-summary" role="article">
            <p>A demonstration of what can be accomplished through <abbr title="Cascading Style Sheets">
            style sheet from the list to load it into this page.</p>
            <p>Download the example <a href="/examples/index" title="This page's source HTML code, not t
            ="/examples/style.css" title="This page's sample CSS, the file you may modify.">css file</a>
```

index.html




```
html {
  font-family: sans-serif;
  -ms-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
}
body {
  margin: 0;
}
a:focus {
  outline: thin dotted;
}
a:active,
a:hover {
  outline: 0;
}
h1 {
```

style2.css

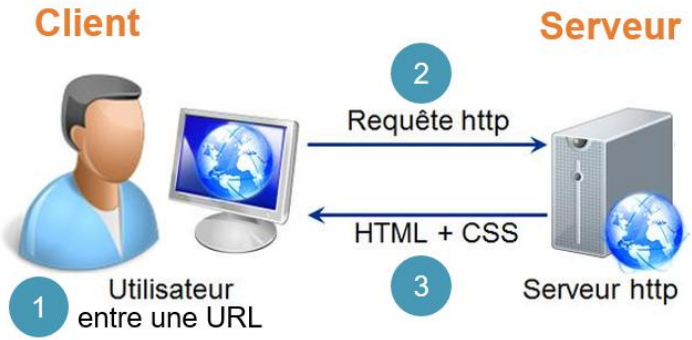


Web : HTTP – client Web – serveur Web



Web : HTTP – client Web – serveur Web

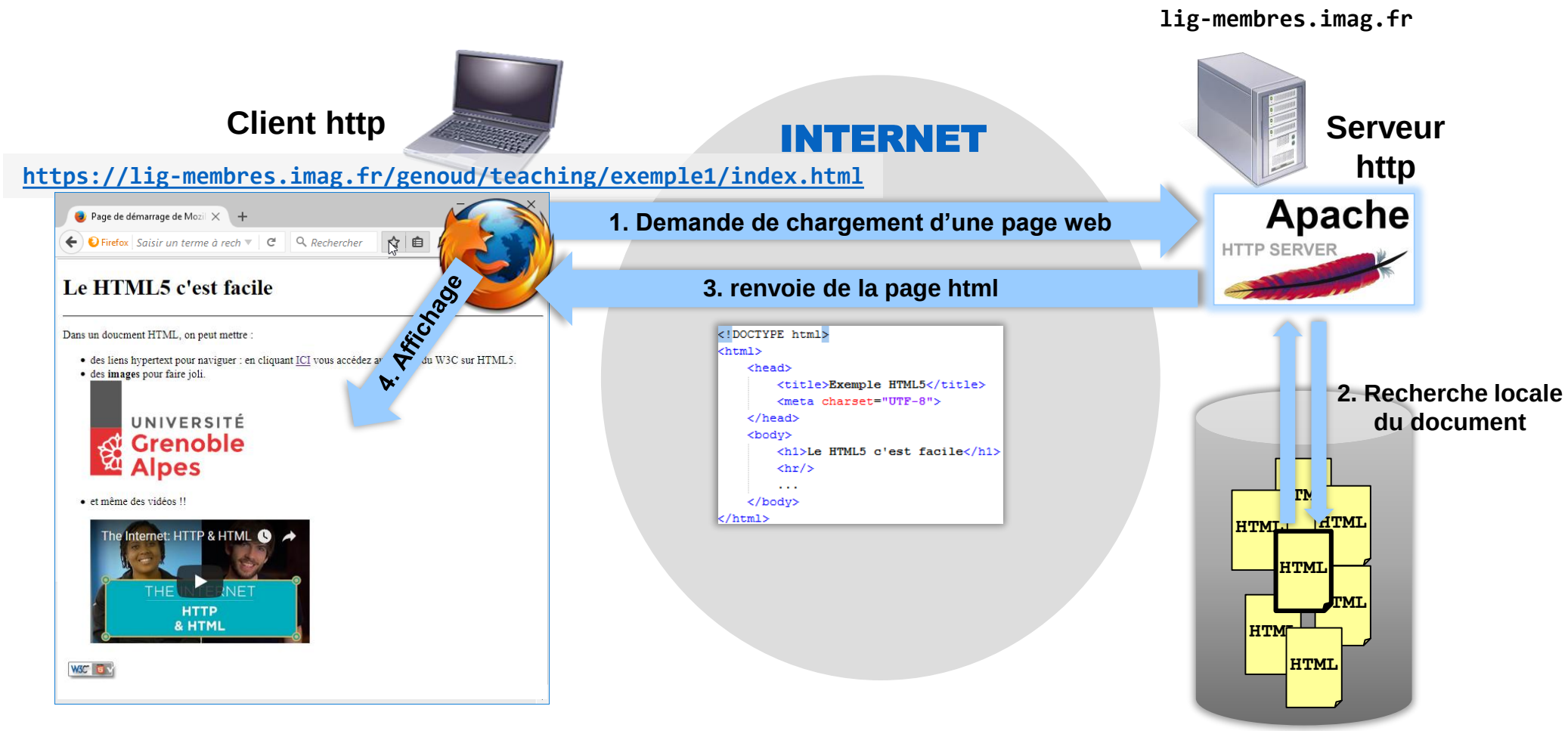
- **HTTP** Hyper Text Transfert Protocol
  - *Protocole utilisé pour chaque transaction web (au dessus de TCP/IP)*
  - *Fournit aux ordinateurs un moyen standardisé pour communiquer entre eux*
    - *Spécifie :*
      - *Comment les clients demandent les données*
      - *Comment les serveurs répondent à ces requêtes*



- **Navigateur Web** :
  - *client HTTP qui se charge de l'affichage d'un document HTML*
  - *FireFox (Mozilla), Chrome (Google), Safari (Apple), Edge (Microsoft), Opera…*

- **Serveur Web** :
  - *serveur HTTP qui transmet aux clients web les documents demandés*
  - *Apache, Nginx, Microsoft IIS, Tomcat, Jetty……*

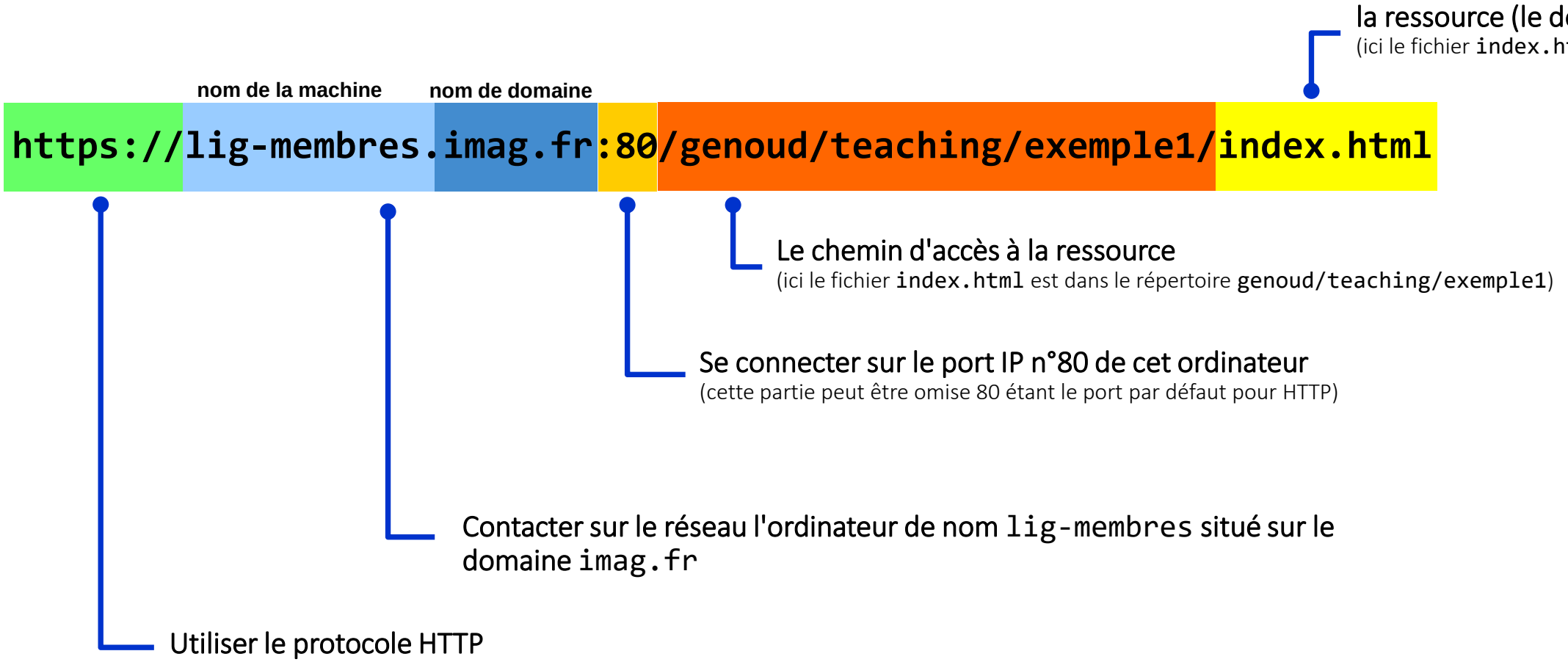
Web : protocole HTTP



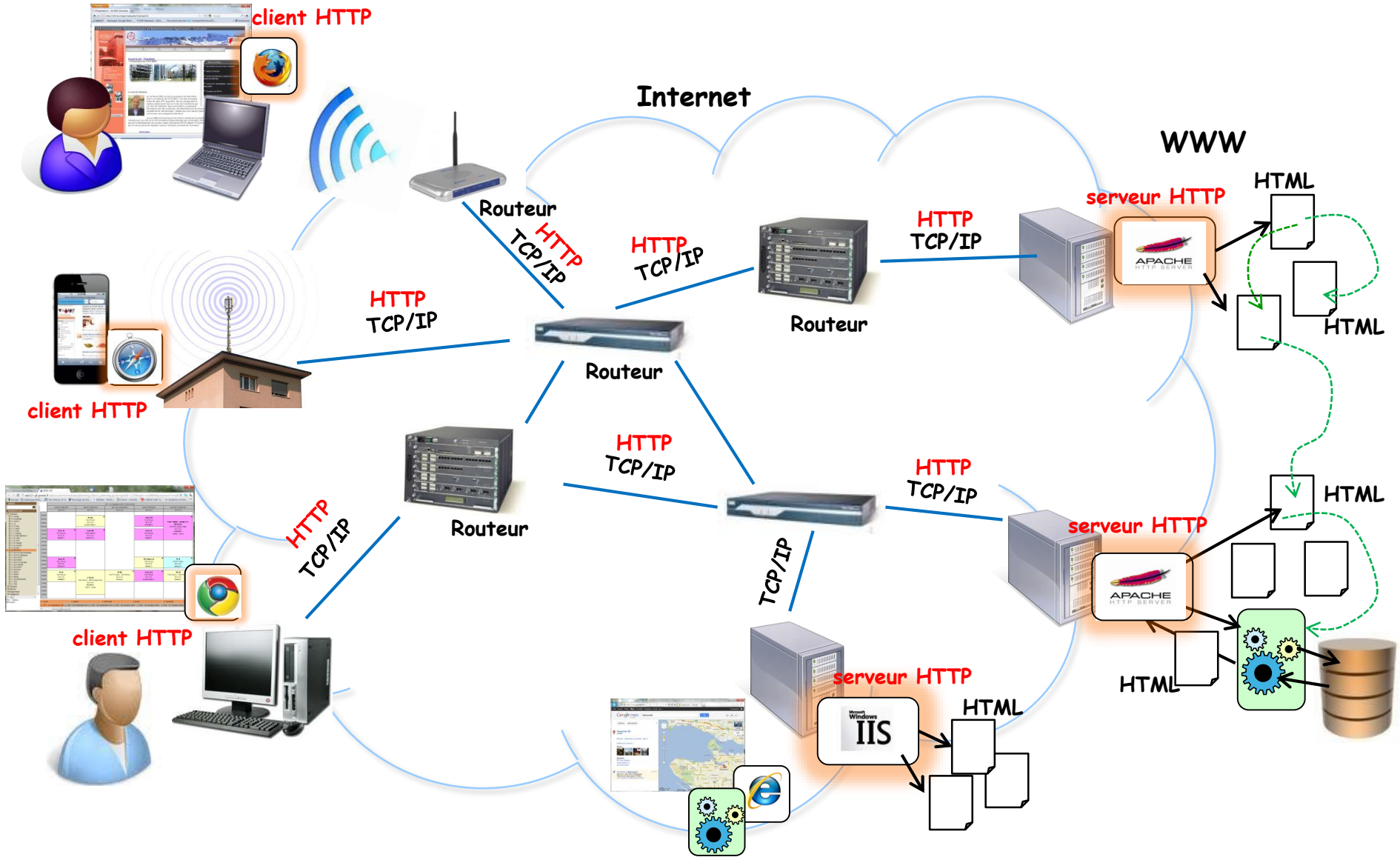
Quelques requêtes HTTP : GET, HEAD, POST, …

Uniform Resource Locator

- Localisation d'un document sur le Web :



Web : HTTP – client Web – serveur Web

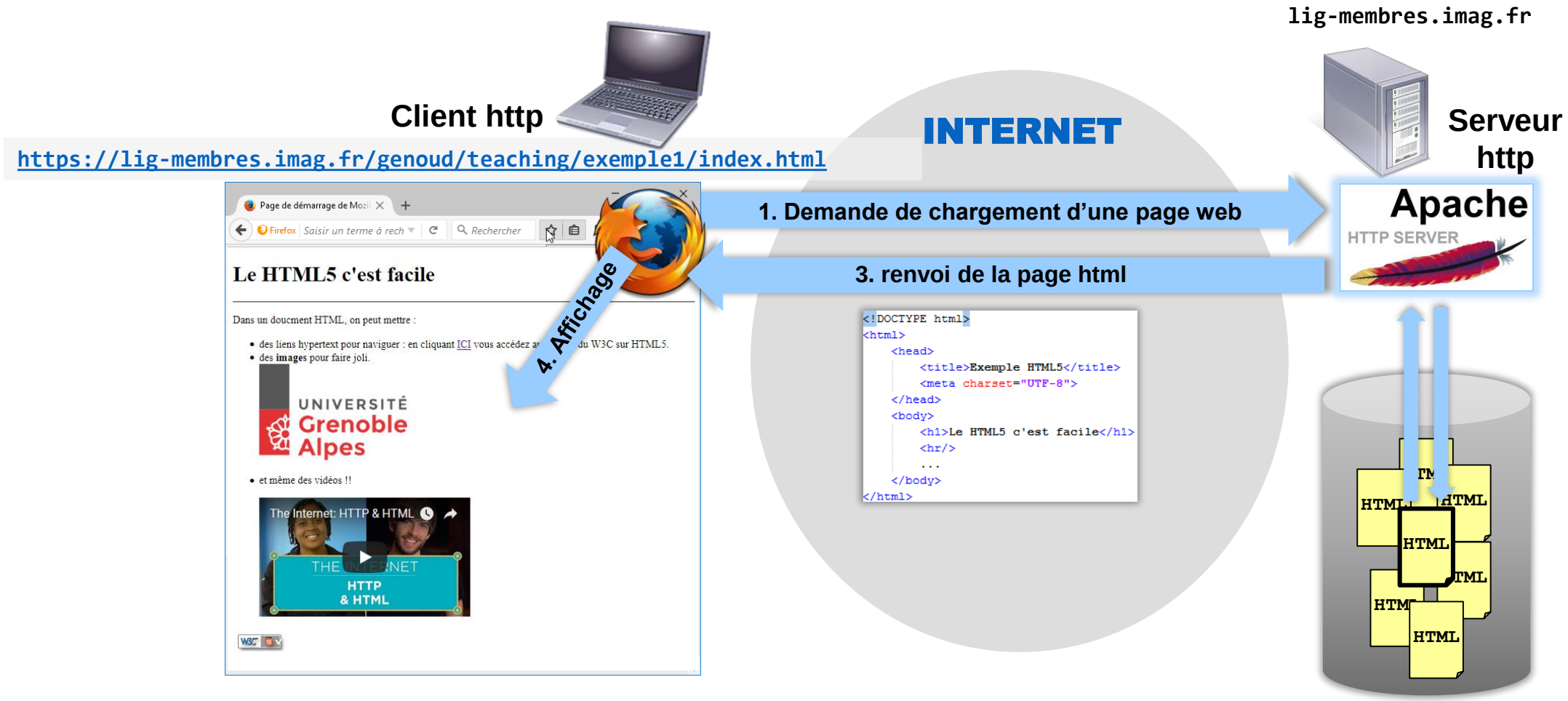


Web : pages statiques / pages dynamiques



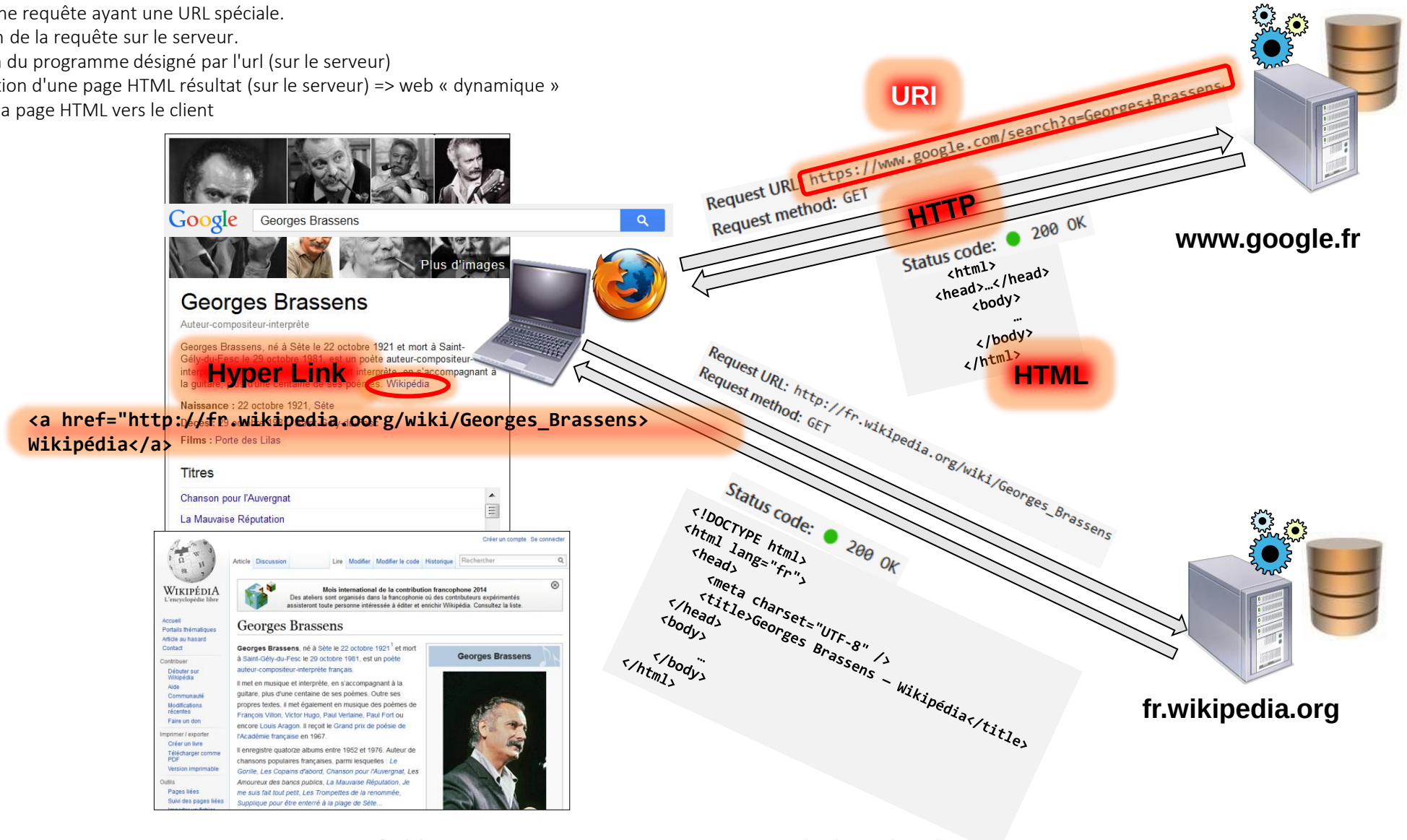
Web : pages statiques / pages dynamiques

- Pages statiques
  - *Documents HTML sous la forme de fichiers*
  - *Transmis tels quels par le serveur HTTP*

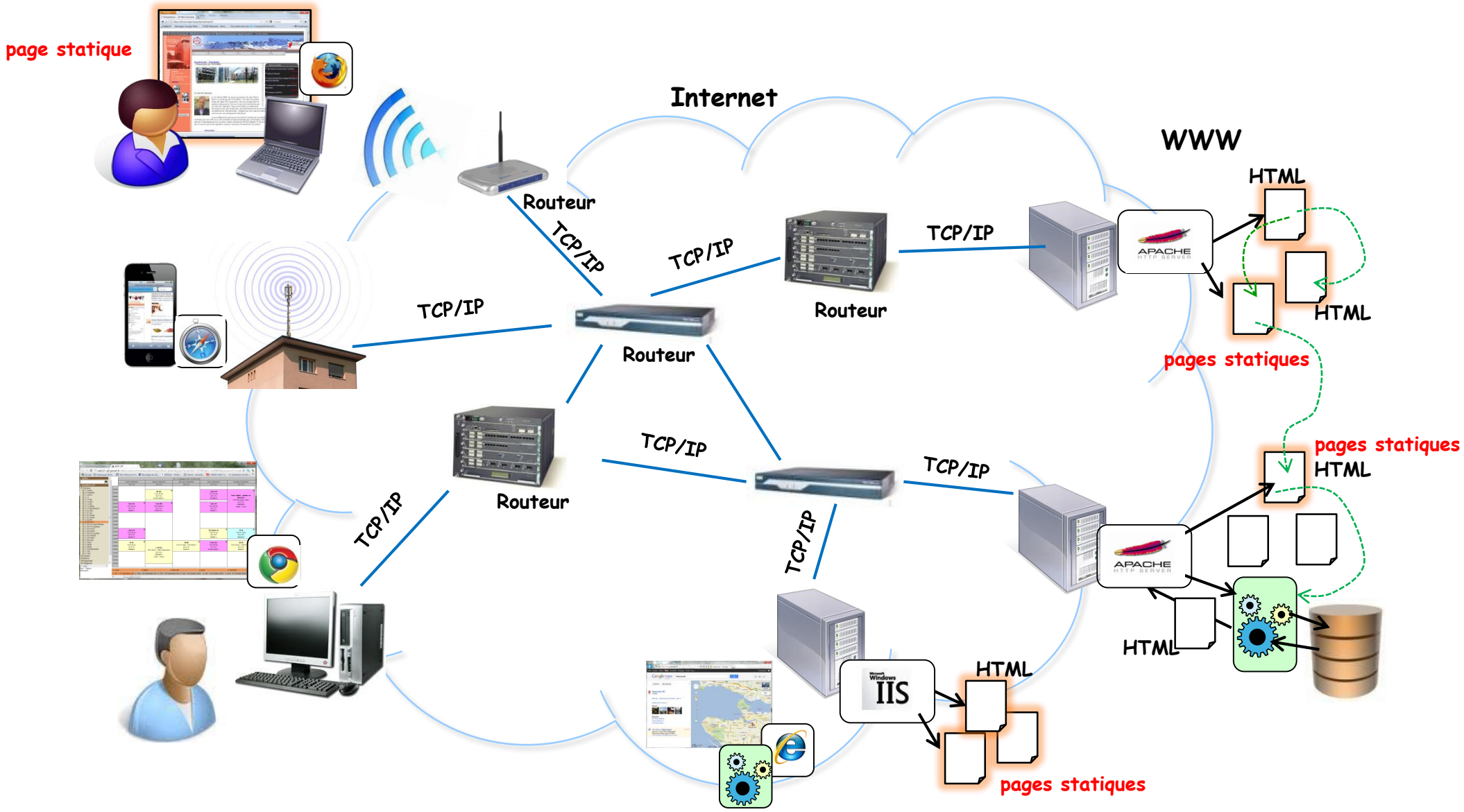


Web : pages statiques / pages dynamiques

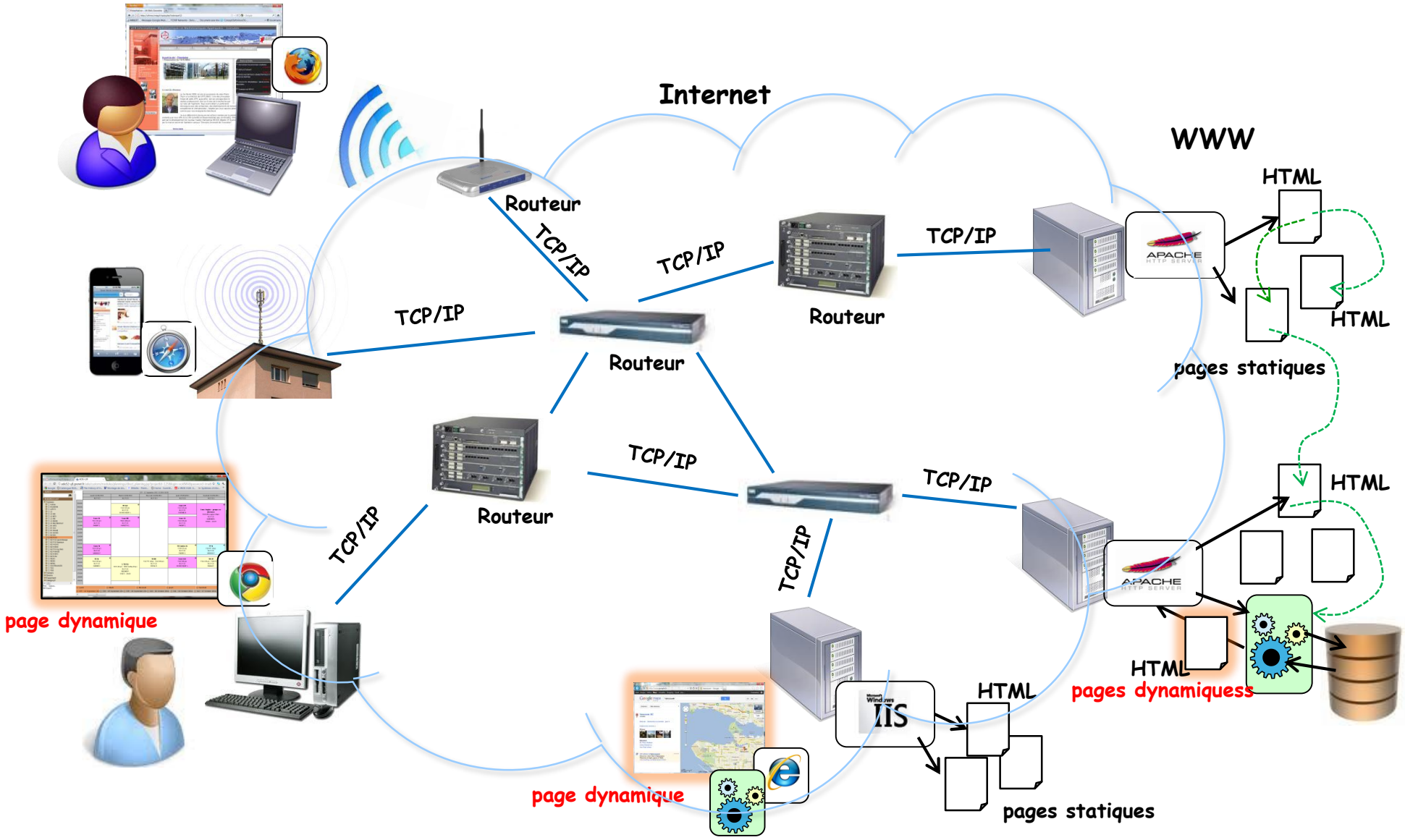
- Documents HTML construits dynamiquement (à la volée) par le serveur
  1 - Envoi d'une requête ayant une URL spéciale.
  2 - Réception de la requête sur le serveur.
  3 - Exécution du programme désigné par l'url (sur le serveur)
  4 - Construction d'une page HTML résultat (sur le serveur) => web « dynamique »
  5 - Envoi de la page HTML vers le client



Web : pages statiques / pages dynamiques



Web : pages statiques / pages dynamiques



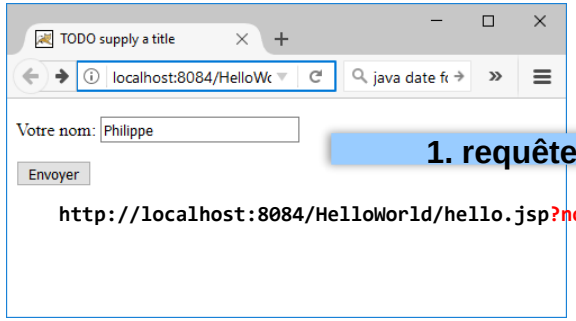
Vocabulaire



Web : Script Serveur (Web 1.0)

- Script serveur
  - *Programme s'exécutant du côté serveur HTTP*
  - *Construit à la volée la partie dynamique d'un document HTML*

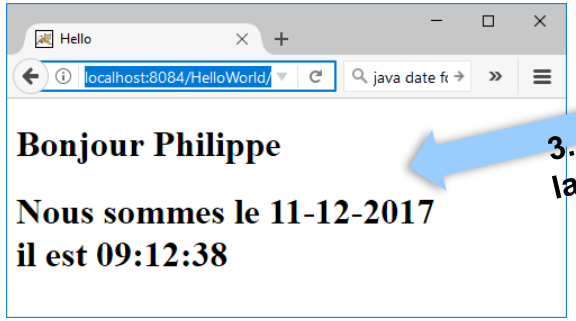
- Exemples de technologies script serveur :
  - *PHP : Personal Home Page*
  - *ASP : Microsoft Application Server Pages*
  - *JSP : Java Server Pages*



1. requête

```
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Hello</title>
    </head>
    <body>
        <h1>Bonjour ${param.nom}</h1>
        <% SimpleDateFormat dt = new SimpleDateFormat("dd-mm-yyyy '<br>il est ' hh:mm:ss"); %>
        <h1>Nous sommes le <%=dt.format(new Date())%></h1>
    </body>
</html>
```

2 exécution coté serveur

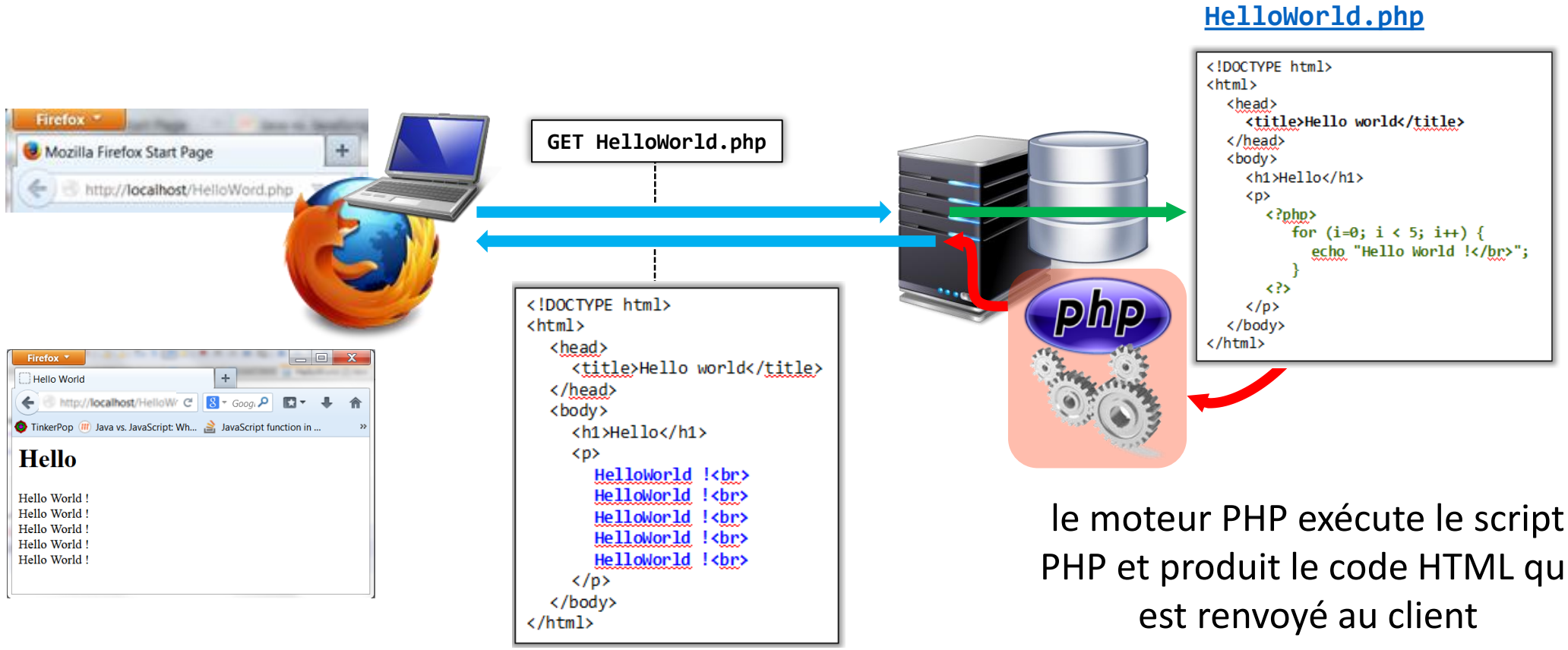


3. renvoi de la page html

- Java Server Pages
  - *Fichier texte qui décrit comment créer une réponse à partir d'une requête particulière.*
  - *tags HTML + extensions + JAVA comme langage de script*

Web : Script Serveur (Web 1.0)

- Exemple : Script PHP



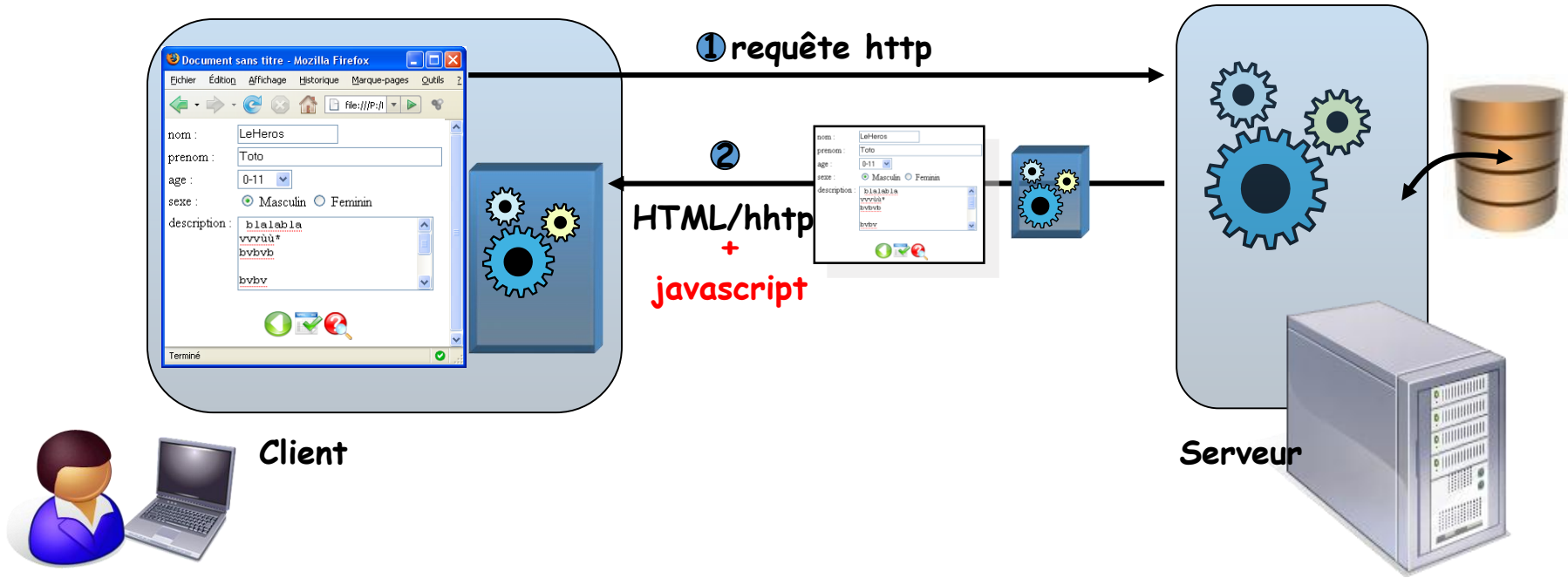
le moteur PHP exécute le script PHP et produit le code HTML qui est renvoyé au client

Web : Script Client – JavaScript



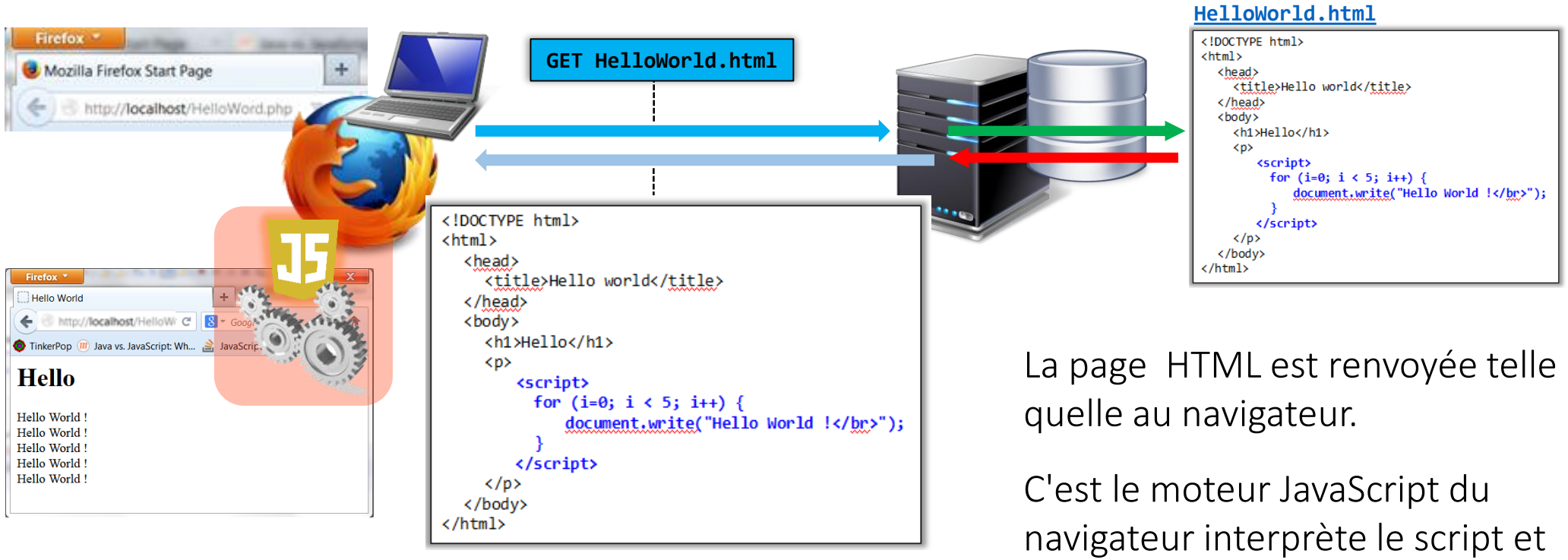
Web : Script Client – JavaScript

- Script client
  - *Programme s'exécutant du côté client HTTP (dans le navigateur)*
  - *Transmis avec le document HTML lors d'une requête HTTP*
- Exemples de technologies client:
  - *JavaScript*
  - *Flash – Action Script (Adobe)*



Web : Script Client – JavaScript

- Programme qui s'exécute dans le navigateur

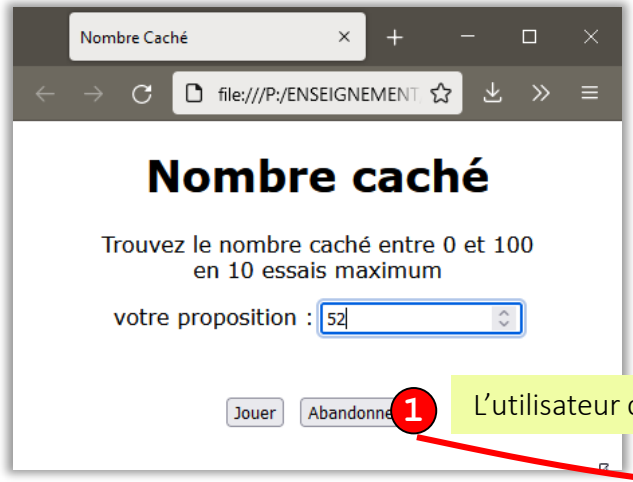


La page HTML est renvoyée telle quelle au navigateur.

C'est le moteur JavaScript du navigateur interprète le script et modifie le DOM (Document Object Model) de la page HTML

Web : Script Client – JavaScript

- JavaScript peut accéder au document HTML et permet de rajouter de l'interactivité en le modifiant



1 L'utilisateur clique sur le bouton

nombreCaché.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Nombre Caché</title>
        <meta charset="UTF-8">
        ...
        <h1>Nombre caché</h1>
        <p>Trouvez le nombre caché entre 0 et 100<br>
            en 10 essais maximum</p>
        <label>votre proposition : </label>
        <input type="number" id="proposition" min="0" max="100" value="0" placeholder="0" />
        <br>
        <!-- élément dans lequel sera affiché un message en fonction de la progression du jeu -->
        <div id="message">
            <br>
            <br>
        </div>
        <!-- élément dans lequel se trouvent les boutons 'Jouer' et 'Abandonner' -->
        <div id="partieEncours">
            <button id="jouerBtn" onclick="jouer()">Jouer</button> 
            <button id="finPartieBtn" onclick="abandonner()">Abandonner</button>
        </div>
        <!-- élément dans lequel se trouve le bouton 'Nouvelle Partie', au début
          cet élément est caché (style CSS invisible). Il est affiché (style CSS
          display:block) lorsqu'une partie est terminée, alors que l'élément
          précédent est caché.
        -->
        <div id="partieTerminee" class="invisible">
            <button id="debutPartieBtn" onclick="debuterPartie()">Nouvelle Partie</button>
        </div>
        <script src="js/nombreCache.js"></script>
    </body>
</html>
```

Un code JavaScript est exécuté

2

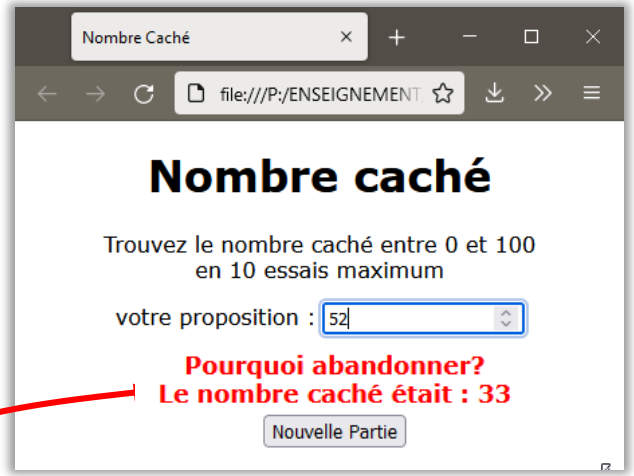
nombreCaché.js

```
    }
    return "";  // le else est inutile car il ya un return dans le if
}

/**
 * Action effectuée lorque l'utilisateur clique sur le bouton 'abandonner'.
 * Le nombre caché est révélé et le bouton qui permettra de relancer une
 * nouvelle partie est affiché.
 */
function abandonner() {
    const message = "Pourquoi abandonner?<br>Le nombre caché était : " + nombreCache;
    document.getElementById("message").innerHTML = message;
    afficherBoutonRejouer();
}
```

Web : Script Client – JavaScript

- JavaScript peut accéder au document HTML et permet de rajouter de l'interactivité en le modifiant



Mise à jour de la page

nombreCaché.html

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Nombre Caché</title>
        <meta charset="UTF-8">
        <h1>Nombre caché</h1>
        <p>Trouvez le nombre caché entre 0 et 100<br>
            en 10 essais maximum</p>
        <label>votre proposition : </label>
        <input type="number" id="proposition" min="0" max="100" value="0" placeholder="0" />
        <br>
        <!-- élément dans lequel sera affiché un message en fonction de la progression du jeu -->
        <div id="message">
            <br>
            <br>
        </div>
        <!-- élément dans lequel se trouvent les boutons 'Jouer' et 'Abandonner' -->
        <div id="partieEncours">
            <button id="jouerBtn" onclick="jouer()">Jouer</button> 
            <button id="finPartieBtn" onclick="abandonner()">Abandonner</button>
        </div>
        <!-- élement dans lequel se trouve le bouton 'Nouvelle Partie', au début
          cet élément est caché (style CSS invisible). Il est affiché (style CSS
          display:block) lorsqu'une partie est terminée, alors que l'élément
          précédent est caché.
        -->
        <div id="partieTerminee" class="invisible">
            <button id="debutPartieBtn" onclick="debuterPartie()">Nouvelle Partie</button>
        </div>
        <script src="js/nombreCache.js"></script>
    </body>
</html>
```

nombreCaché.js

```
    }
    return "";  // le else est inutile car il ya un return dans le if
}

/**
 * Action effectuée lorque l'utilisateur clique sur le bouton 'abandonner'.
 * Le nombre caché est révélé et le bouton qui permettra de relancer une
 * nouvelle partie est affiché.
 */
function abandonner() {
    const message = "Pourquoi abandonner?<br>Le nombre caché était : " + nombreCache;
    document.getElementById("message").innerHTML = message;
    afficherBoutonRejouer();
}
```

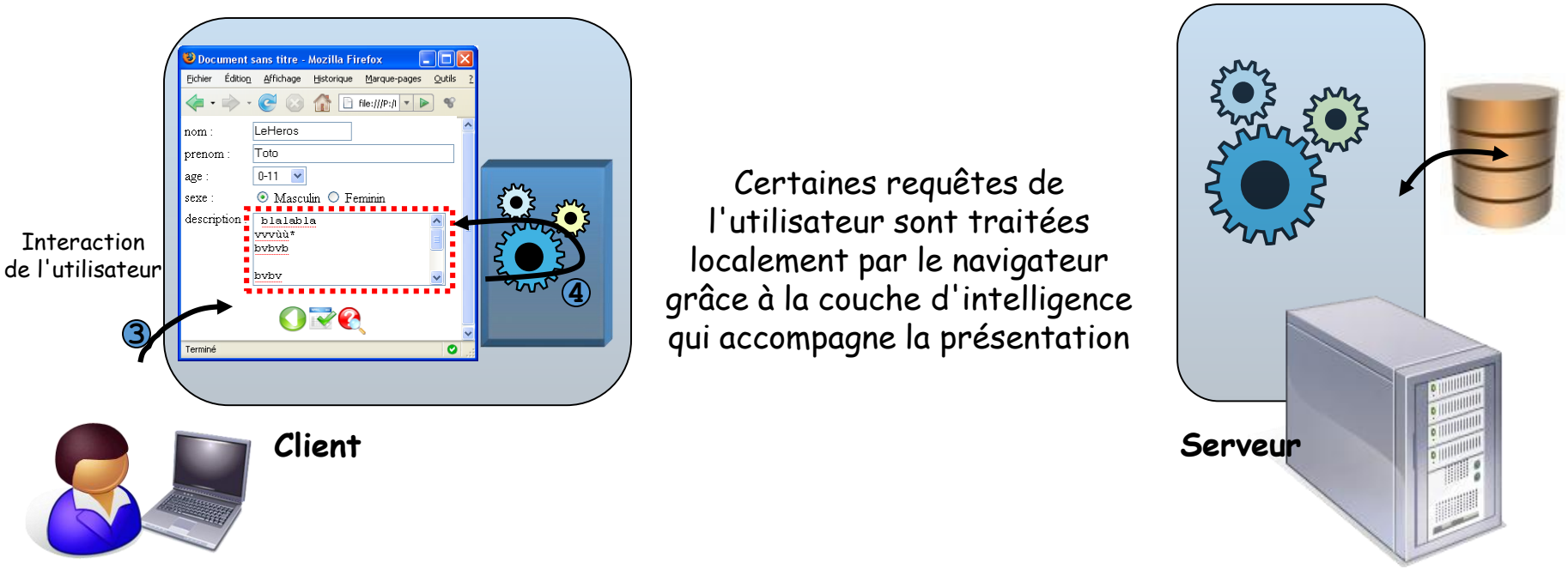
3

Web 2.0 : AJAX


URL
Serveur-d'applications
IDE
WHATWG
API
Pages-dynamiques
REST
AJAX
Java
HTTP
W3C
Routeur
JSP
CSS
Web
Client-Web
Hypertexte
Lien
Internet
Script
PHP
Numéro-IP
Protocole
JavaScript
TCP/IP
Pages-statiques
Lien
Serveur-Web
Serveur
HTML
Servlets
CMS
DNS
Nom-IP
Framework

Web 2.0 : AJAX

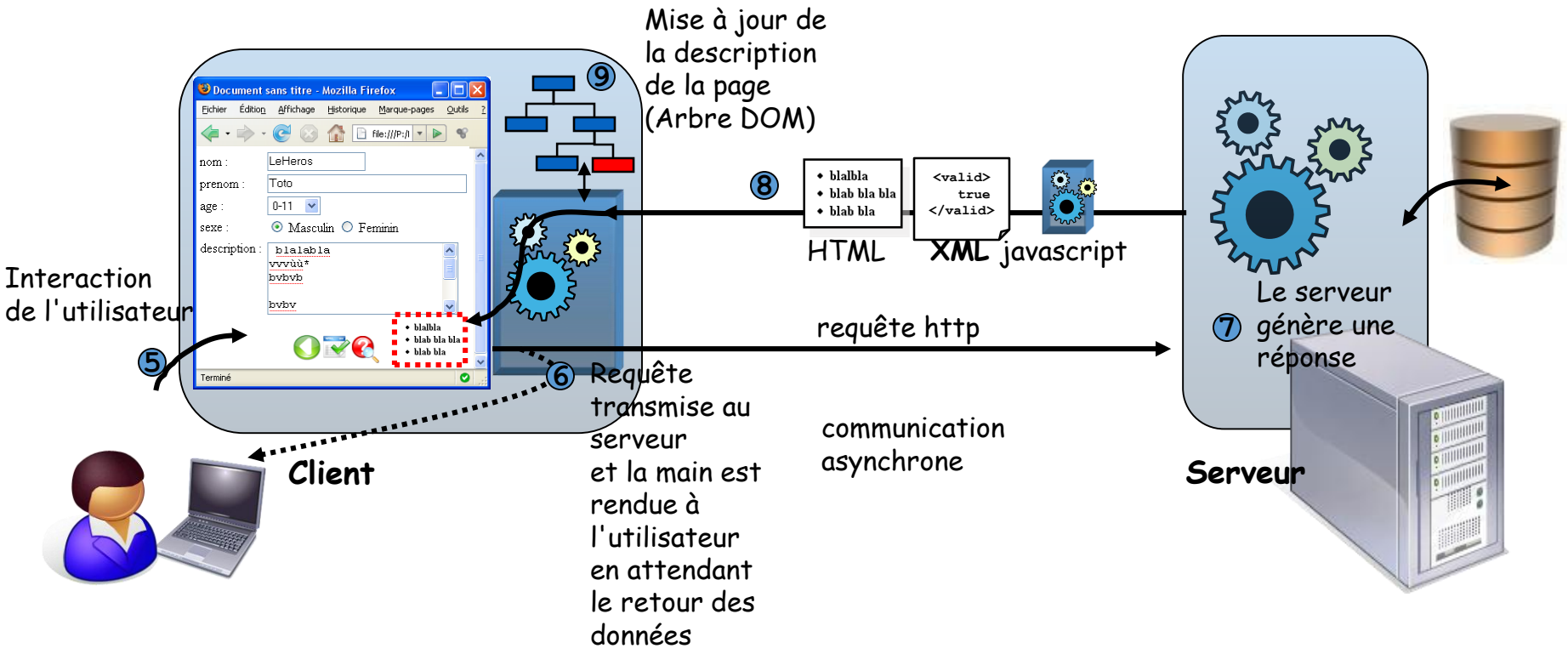
- JavaScript : Une partie de l'intelligence fonctionnelle de l'application est déportée vers le navigateur



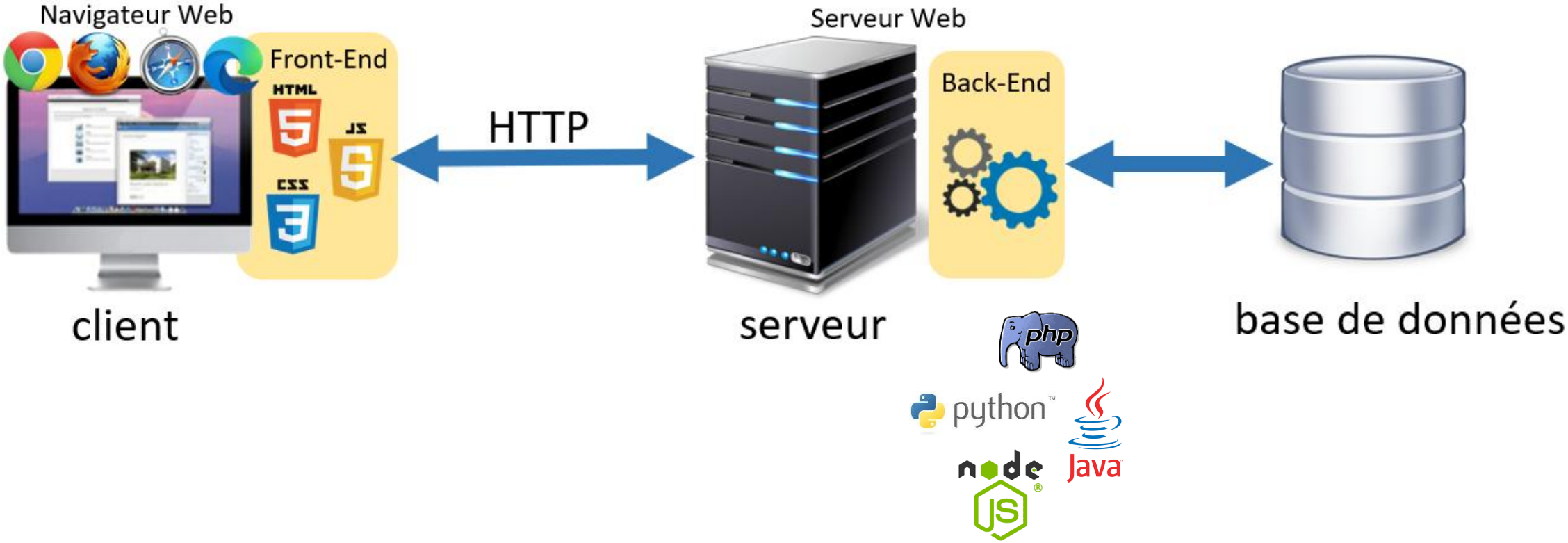
- Web 2.0 : utiliser aussi JavaScript pour communiquer avec un serveur et mettre à jour la page

Web 2.0 : AJAX

AJAX: Asynchronous JavaScript and XML



Application Web



Web: CMS


URL
Serveur-d'applications
IDE
WHATWG
API
Pages-dynamiques
REST
AJAX
Routeur
Java
HTTP
W3C
JSP
CSS
Web
Client-Web
Lien
Hypertexte
Script
Internet
PHP
Numéro-IP
Protocole
JavaScript
TCP/IP
Pages-statiques
Lien
Serveur-Web
Serveur
HTML
Servlets
CMS
DNS
Nom-IP
Framework

Web: CMS

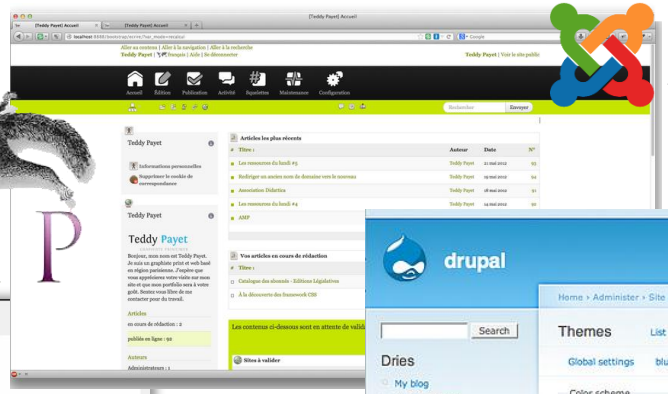
- CMS (***C**ontent **M**anagement **S**ystem) ou* SGC (**S**ystème de **G**estion de **C**ontenu)
  - *programme qui facilite la création et la gestion d'un site web*
  - *interface WYSIWYG (What You See Is What You Get)*
  - *modèles de sites*
  - *univers graphiques*
  - *…*

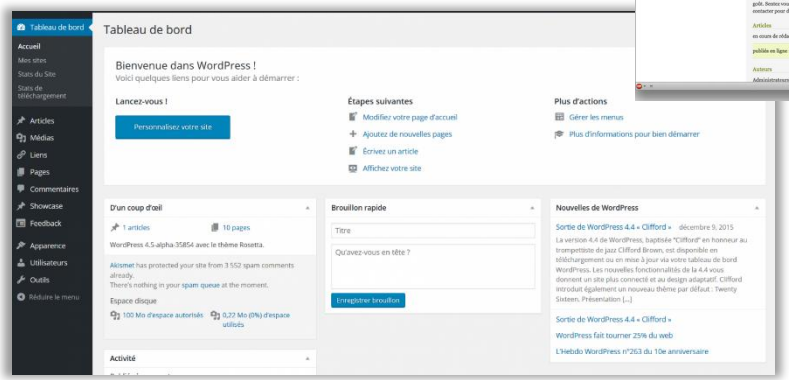
→ Techno : HTML, CSS,… : mais l'utilisateur ne voit pas (ou peu) ce code, il crée son site via une interface















Web : Framework



Web : Framework

- Framework (de développement informatique) - Cadriciel 
  - *un ensemble d'outils et de composants logiciels éprouvés et réutilisables (bibliothèques de programmes, classes, helpers…) organisés conformément à un plan d'architecture bien défini (« squelette » ou « canevas »).*
  - *Il est souvent fourni sous la forme d'une bibliothèque logicielle et accompagné d'une documentation sur l'architecture du Frameworkun définissant ensemble de préconisations pour le development d'applications*

- Intérêts
  - *Bonne organisation du code source grâce à l'architecture proposée (imposée ?) par le framework*
  - *Rapidité de développement*
    - *au lieu de faire un développement complet à partir zéro on s'appuie sur les composants du framework et on peut plus facilement se concentrer sur la partie métier*
  - *Meilleures maintenabilité*
    - *l'organisation du Framework facilite la maintenance du logiciel et la gestion des évolutions.*

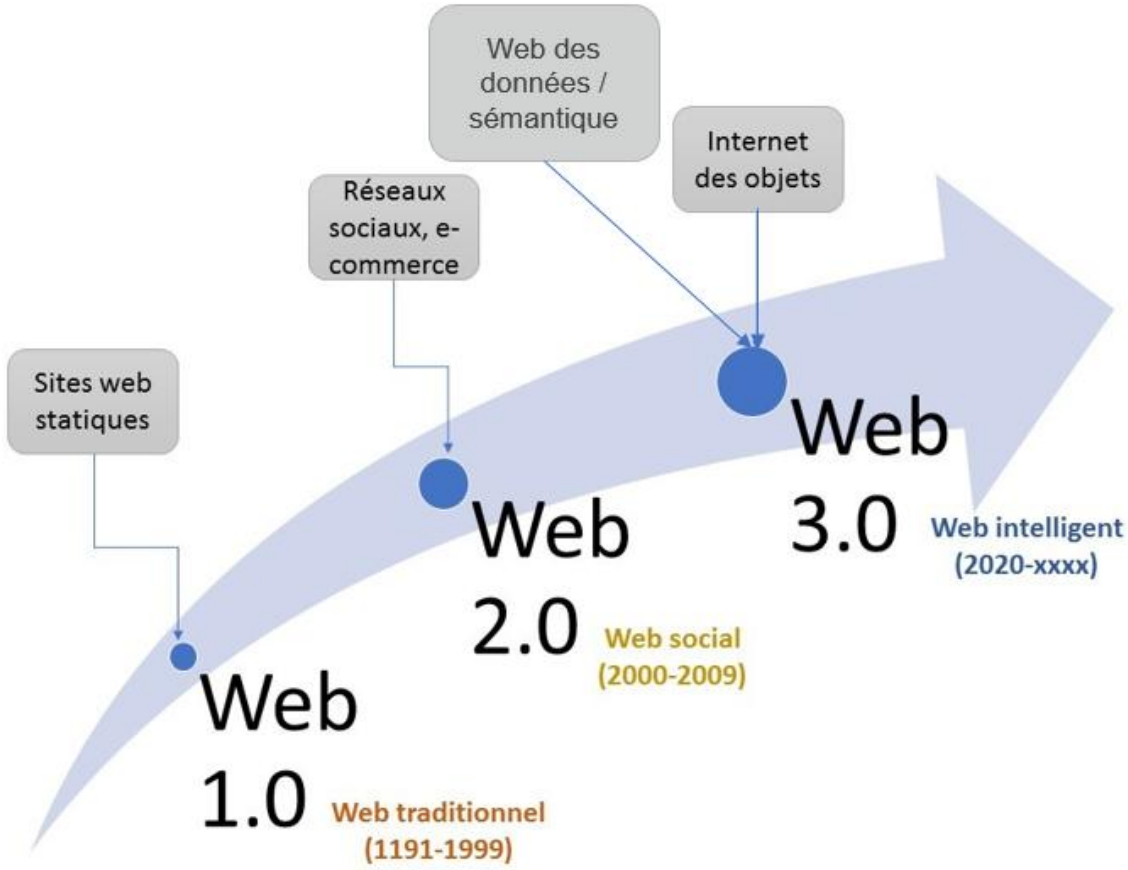
Web : Framework

- Nombreux Frameworks web



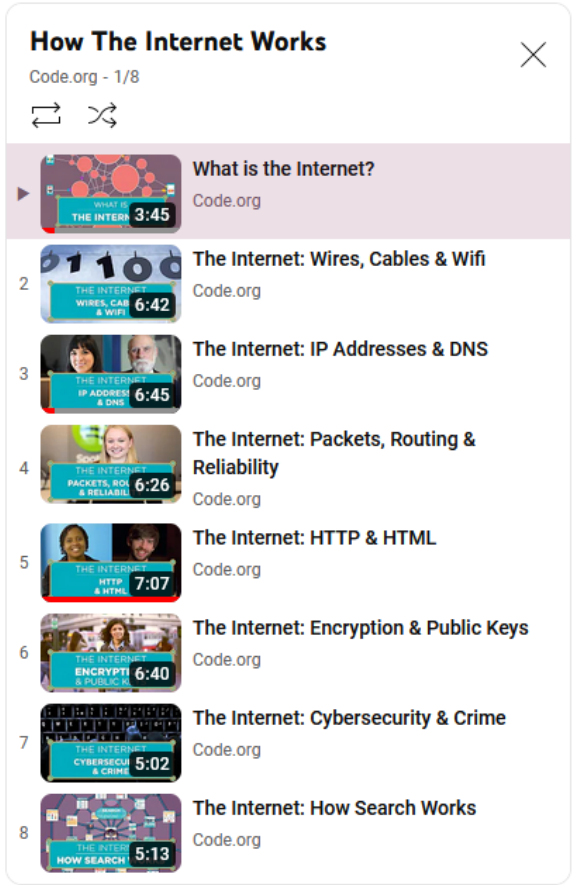
Evolutions du Web

- Le Web 1.0
  - *autrement appelé Web des documents*
- Le Web 2.0
  - *autrement appelé Web social, participatif*
- Le Web 3.0
  - *autrement appelé Web des données / Web Sémantique*
  - *D'autres voient le Web des objets comme le Web 3.0*
- Le Web 4.0
  - *autrement appelé Web des agents / Web intelligent*
    https://la-ligne-web.com/2024/11/22/qu-est-ce-que-le-web-4/



D'autres vidéos

https://www.youtube.com/playlist?list=PLzdnOPI1iJNfMRZm5DDxco3UdsFegvuB7





https://www.youtube.com/watch?v=dFmTbZMGRdQ
durée 4:29 min



https://www.youtube.com/watch?v=hrzXdKvG1CY
durée 2:53 min



https://www.youtube.com/watch?v=xhjgvhZQ-j8
durée 2:39 min

Contenu du Cours TW

- Syllabus 2025/2026
- Celui de2026/2027 devrait être très similaire
- Cette année : tentative de pédagogie inversée. Les cours dans la mesure du possible seront mis en ligne une semaine en avant et devront avoir été étudiés avant la séance

Semaine 1 (22/09/2025) :

Cours 1 **Cours 1 : Présentation générale du cours Technologies du Web**
Cours 2 **Cours 2 : Introduction à HTML**
Complément Vidéo HTML et encodage des caractères

Semaine 2 (29/09/2025) :

Cours 3 **Cours 3 : Introduction aux CSS**

- principes de base
- sélecteurs de styles
- couleurs
- boites (marges, bordures)

Cours 4 **Cours 4 : Introduction aux CSS - suite**

- positionnement des éléments (float, absolu, relatif)
- flex boxes
- grid layout (*cette partie est incomplète et en cours de rédaction*)

Semaine 3 (06/10/2025) :

Cours 5 **Responsive Web Design - Introduction Bootstrap**

- Responsive Web Design
- Bootstrap 5
  - le framework
  - la grille boostrap

Semaine 4 (13/10/2025) :

Cours 6 **Introduction à JavaScript (1ère partie)**

- Historique et évolution de JavaScript
- JavaScript côté client et JavaScript côté serveur
- Types de bases et Variables

Cours 7 **Introduction à JavaScript (2ème partie)**

- Fonctions
  - déclaration / appel
  - instruction return

Semaine 5 (20/10/2025) :

Cours 8 **Introduction à JavaScript (2ème partie - suite)**

- expression fonctions
- fonction flêchées
- fonctions d'ordre supérieur
- Portée (scope des déclarations)
  - portée bloc, globale, locale
  - **let** vs. **var** (hoisting)
- Bonnes pratiques de codage

Cours 9 **Modules JavaScript**

Semaine 6 (3/11/2025) :

Cours 10 **Programmation Objet en JavaScript (1ère partie)**

- Objets
- Constructeurs et méthodes
- Prototypes

Cours 11 **Programmation Objet en JavaScript (2ère partie)**

- Chaîne de prototypes
- La classe Object (ES5)
- Classes (ES6)

Semaine 7 (10/11/2025) :

Cours 11 **Programmation Objet en JavaScript (2ère partie fin...)**

- Classes en JavaScript (ES6)

Cours 12 **JSON** (JavaScript Object Notation)

Cours 13 **Programmation Evénementielle en JavaScript**

- Principes
- Gestionaires d'Evénements
- Objet Event
- Propagation des événements

Semaine 8 (17/11/2025) :

Cours 14 **Tableaux en JavaScript**

- Création de tableaux (expression littérale, constructeur Array)
- Parcours de tableaux
- Méthodes statiques d'Array
- Méthode d'instances (Array.prototype)