

Développement Web

Introduction au web dynamique

Halim Djerroud <hdd@ai.univ-paris8.fr>

LIASD - Université Paris 8

2018

Plan de cours général

8 semaines de cours :

- 1 Introduction au web dynamique
- 2 Gestion de rendu et décoration
- 3 Le duo magique : PHP et MySql
- 4 Programmation OO et les design patterns
- 5 TP noté pour faire une pause et un peu de JS avec ses saveurs (jQuery).
- 6 Les frameworks et Symfony4
- 7 Symfony4 et ses rouages
- 8 Correction de projets

Plan de cours

- 3 heures de cours
 - 30 minutes pour la distribution de projets et explications.
- 1 Web dynamique
 - 2 Protocole HTTP
 - 3 Introduction HTML, PHP et JavaScript
 - 4 Outils de développement
 - 5 Distribution des projets

Support de cours

- `http://djerroud.halim.info/index.php/cours/programmation-web/`

Web statique

- Protocole d'échange de documents (fichiers) appelés pages web
- Les fichiers (documents) sont écrit dans un format spécial **HTML**, spécifié par **W3C**
- Les fichiers sont stocké sur un serveur appelé serveur Web
- Les clients se connectent au serveur via un client web appelé Navigateur
- Les ressources sont identifiées par des URL ex :
`http://djerroud.halim.info/index.php/cours/
programmation-web/`
- Le navigateur web (moteur de rendu) récupère les pages et assure le rendu graphique de la page

Fonctionnement

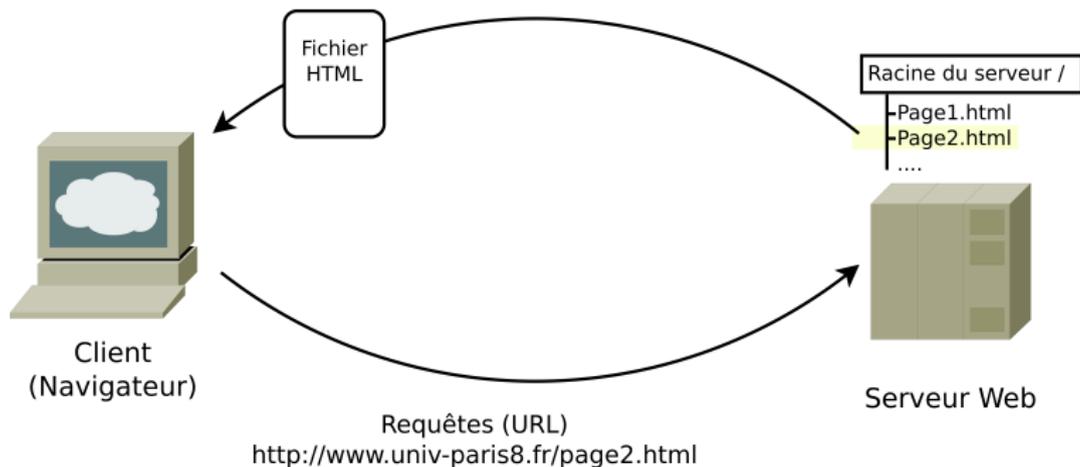


FIGURE – Fonctionnement Web statique

Exemple page HTML

```
<html>
  <head>
    <title> Ma page </title>
  </head>
  <body>
    <h1> Ma Page Web statique </h1>
    <p> Contenu de la page Statique </p>
  </body>
</html>
```

Exemple page HTML



FIGURE – Rendu de la page Web sur un navigateur

Limites du Web statique

On entend par page statique, non pas une page sans mouvements ou sans animations, mais une page visible telle qu'elle a été conçue.

Limites :

- Obsolescence très rapide (Ex : affichage d'une date sur une page web)
- Information toujours présentées de la même façon
- Interaction impossible
- etc.

Web dynamique

On entend par page dynamique, une page web générée à la demande.

Avantages :

- Affichage en fonction des requêtes des utilisateurs
- Le contenu du site est généralement stocké dans une base de données ce qui permet de faire une manipulation de données avant affichage
- La connexion avec la base de données et la réalisation de traitements dynamiques s'effectue au moyen d'un langage de programmation (ex : PHP, Python etc.)
- etc.

Fonctionnement

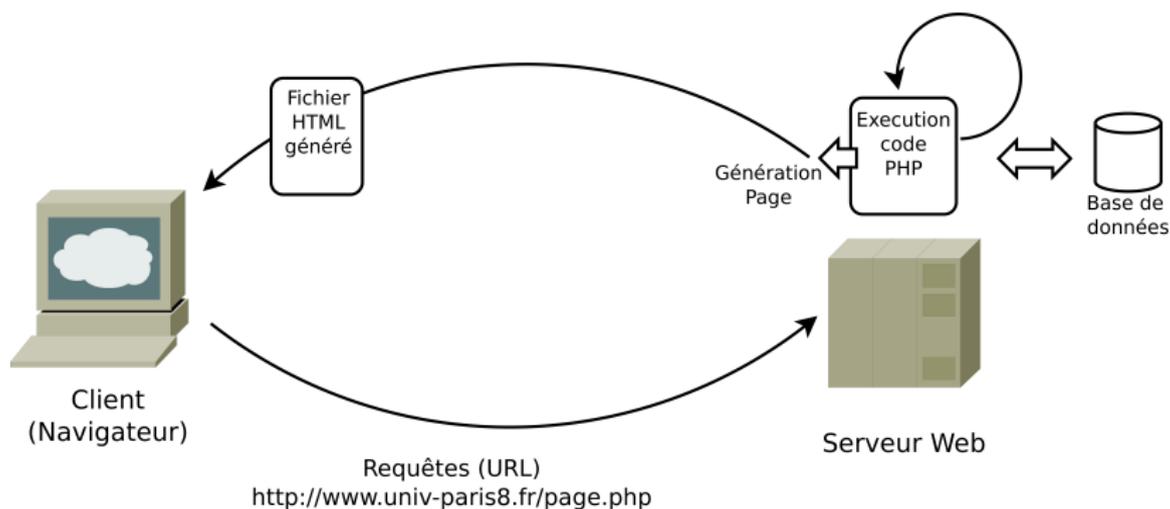


FIGURE – Fonctionnement Web dynamique

Exemple page PHP

```
<html>
  <head>
    <title> Ma page </title>
  </head>
  <body>
    <p> On est le :
    <?php
      echo date();
    ?>
    </p>
  </body>
</html>
```

Exemple page HTML générée

```
<html>
  <head>
    <title> Ma page </title>
  </head>
  <body>
    <p> On est le :
    04/06/2018
    </p>
  </body>
</html>
```

Développement full web

Habituellement on développe les applications de gestion avec des langages de programmation comme JAVA et C/C++ etc.

- Il faut installer le logiciel sur la machine (les machines) du client
- devient difficile si le client dispose de différents systèmes d'exploitations et différentes architectures

Avec le web dynamique (Web 2.0) il est possible décrire ces application avec la technologie Web

- Un navigateur Web suffi pour utiliser l'application

Architecture 2-tiers

- Aussi appelé l'architecture «client-serveur» ou client lourd
- Une partie des traitements était effectués sur le poste client et la seconde sur le serveur
- On peut distinguer deux types d'applications (application client et application serveur)
- Dans les applications clients, la plupart des traitements sont effectués sur le poste du client qui possède une IHM dite "lourde"
- etc.

Fonctionnement

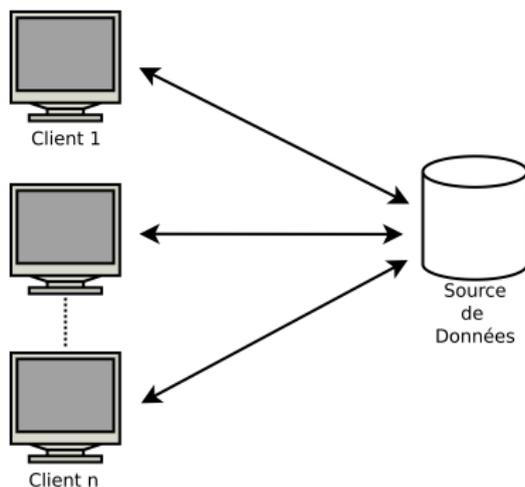


FIGURE – Architecture 2-tiers

Architecture 3-tiers

- Un nouveau niveau est apparu «Serveur d'application»
- Le client est très légers étant donné qu'il n'a aucun rôle de traitement, il permet l'affichage, généralement il dispose uniquement d'un navigateur
- Le client ne sert qu'à envoyer des requêtes et à afficher les réponses du serveur
- etc.

Fonctionnement

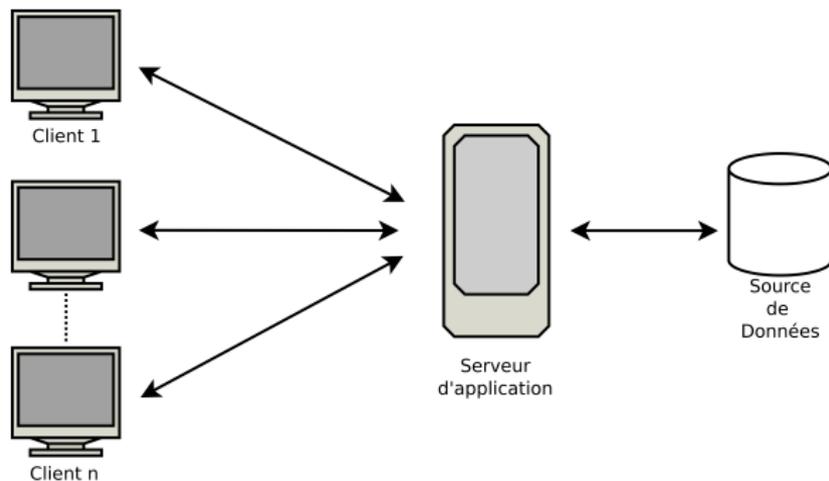


FIGURE – Architecture 3-tiers

Protocole HTTP et URL

- HTTP (HyperTex Transsmision Protocol) est un protocole de communication entre le navigateur et (client) et le serveur Web, il est basé sur les liens **hypertextes**
- Les Hyperliens permette d'adresser une ressource
- Les URL (Uniform Resource Locators) sont les noms données au Hyperliens

Exemple URL :

- `http://www.fsf.org`
- `ftp://www.univ-paris8.fr`
- `file:/home/user/test.txt`
- `mailto:hdd@ai.univ-paris8.fr`

Protocole HTTP

- Lorsqu'un navigateur Web demande un document **HTML**, une connexion **TCP** est établie entre le navigateur et le serveur web, si le port n'est pas spécifié alors le port par défaut est choisi **80**
- Pour utiliser un autre port il faut le spécifier dans l'**URL**, ex :
`http://www.halim.info:8080`
- Une fois la connexion établie, le navigateur envoie une requête via le protocole HTTP ex :
`GET /dossier/page.htmlHTTP/1.0`
- Une fois la séquence requête/réponse achevée, la connexion **TCP** est fermée
- Une nouvelle connexion est établie pour chaque requête/réponse

Contenu requête HTTP

- La méthode à utiliser pour récupérer le document (GET)
- Le nom du fichier demandé (/dossier/page.html)
- La version du protocole HTTP à invoquer (HTTP/1.0)
- D'autres méthodes peuvent être utilisées pour n'obtenir que l'entête associée avec la méthode **HEAD** ou-bien la méthode **POST** pour envoyer les données d'un formulaire au serveur

Réponse du serveur

- Le serveur répond à la requête en envoyant une réponse composé de trois parties
 - **L'état de la réponse** : (HTTP/1.0 200 OK) est une ligne de texte avec la version du protocole HTTP est un code d'état du résultat (200 pour acquittement).
 - **L'entête de la réponse** : (Date ...) Contient des information sur le serveur, l'encodage utilisé pour le contenu et une ligne vide pour annoncer les données
 - **Les données** :(<HTML>...)généralement du HTML

Exemple réponse requête HTTP

```
HTTP/1.0 200 OK
Date: Fri, 31 Dec 1999 23:59:59 GMT
Server: Apache/0.8.4
Content-Type: text/html
Content-Length: 59
Expires: Sat, 01 Jan 2000 00:59:59 GMT
Last-modified: Fri, 09 Aug 1996 14:21:40 GMT

<TITLE>Exemple</TITLE>

...
```

Exemple copié ici [https:](https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol)

[//fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol](https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol)

Code erreurs HTTP

Les codes les plus courants sont :

- 200 : succès de la requête ;
- 301 et 302 : redirection, respectivement permanente et temporaire ;
- 401 : utilisateur non authentifié ;
- 403 : accès refusé ;
- 404 : page non trouvée ;
- 500 et 503 : erreur serveur ;
- 504 : le serveur n'a pas répondu.

Liste complète :

https://fr.wikipedia.org/wiki/Liste_des_codes_HTTP

Serveur Web

- Apache
- Nginx
- Tomcat
- etc.

Clients Web ou Navigateurs

Les Navigateurs les plus courants

- Mozilla Firefox
- Chromuim
- Opea
- Konqueror
- Navigateur sur mobiles et tablettes
- etc.

HTML

- HTML (Hyper Texte Markup Language) est d'un langage de description (et non pas d'un langage de programmation) qui va nous permettre de décrire l'aspect d'un document, d'y inclure des informations variées (textes, images, sons, animations etc.) et d'établir des relations cohérentes entre ces informations grâce aux liens hypertextes

HTML

Les avantages du langage HTML sont nombreux :

- Peux coûteux en effet un simple éditeur de texte suffit à écrire ses premiers documents HTML
- Relativement facile à aborder,
- Il représente en outre un bon moyen de dépasser les problèmes de compatibilité entre des systèmes et des formats informatiques différents.

HTML

- La description d'un document HTML passe par l'utilisation de BALISES (ou "TAGS" en anglais).
- Une balise est délimitée par les signes "<" et ">" entre lesquels figure le nom de la balise.

Exemple :

```
<nomBalise> .... </nomBalise>
```

PHP

- HyperText Preprocessor
- Open source
- Langage de programmation coté serveur
- Développent POO

JavaScript

Le Javascript est un langage de script, normalement inclus dans la page Web et non-nécessaire à la page, il permet des choses simple

....

- Sources publique dans la page (non-protégé)
- Pas toujours très compatible ou imprévis
- Pas toujours très utile
- Plutôt adapté à du cosmétique mais peut s'avérer très pratique

GNU/Linux

- Système d'exploitation
- Installation en natif ou en machine virtuelle
- Une distribution au choix de préférence GNU/Linux Debian
- Démo

Serveur Apache2

- Un serveur HTTP
- Libre
- Plusieurs modules disponibles (`mod_ssl`, `mod_php`, `mod_python` etc.)
- Gestion du `virtual host` complet (IP alias, HTTP 1.1 etc.)
- Configuration simple
- Portable
- Démo

Installation et configuration Apache2

- Utiliser de préférence le système de gestion de package fourni par votre distribution
- Activation du mode debug
- Virtual Host
- Démo

Navigateur

- Firefox
- Modules supplémentaire
- Web Developer
- Débogueur JS

Eclipse

- IDE (integrated development environment) en français
environnement de développement intégré
- Développer rapidement
- Débogueur intégré
- Coloration syntaxique
- Navigation
- etc.
- Démo

Projets

- Distribution des projets

Fin

- Questions ?