

PHP

Découverte programmation PHP



HTML

Envoyer

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Titre </title>
</head>
<body>
  <button id="bouton" >Envoyer</button>
</body>
</html>
```

- Déclaratif
- Langage de balisage spécifier la structure et le contenu
- une balise HTML.

CSS

Envoyer

```
<style>
#bouton {
color:red;
}
</style>
```

- Déclaratif
- les feuilles de style en cascade : Mise en forme des éléments.
- apparence à un bouton, couleur , rollover...

JAVASCRIPT

Cette page indique
bienvenue

OK

```
<script>
function myFunction(
){
  alert("bienvenue")
}
</script>
```

- Programmation
- le comportement des éléments
- Suite un événement par ex. clic
- Pas forcément exécuté

Rappel sur les langages fondamentaux :

- Création d'un site statique (Rappel, limites ?).
- Algorithme possible, limité ?
- Données statiques / Dynamiques?
- Comment créé un e commerce ? Autre site dynamique ?

► Site dynamique

- **Gabarit de page pour chaque article à vendre (Serveur PHP) => objet de ce cours**
- On pioche dans la base de donnée (Serveur SGBD)
- On le présente sous forme HTML (Serveur HTTP)

1. PARTAGE DE DONNEES

2. MODELE DE PAGE

Architecture	Structure de données	Les scripts
Langage serveur	Algorithme	Création de fonctions
Serveur Apache	Les variables	Architecture des scripts
SGBD	Les types	Include
Serveur Mysql	Les tableau	Require
Architecture Client Serveur	Les fonctions	Typage des paramètres
Wamp	Les conditions	Les opérateurs
Hébergement	Les boucles	Concaténation

► **Objectif : Créez un site internet dynamique en PHP**

► **Problème :**

- Site statique ?
- Données ?

► **Résolution**

- Site dynamique : Langage PHP (Framework C)
- Données : Base de données MYSQL
- Architecture
- Algorithmique en PHP



Architecture client-serveur et installation de serveurs locaux - Apache et MySQL



Structures de données (notamment les tableaux et les tableaux associatifs)



Structures de contrôle



Méthode d'utilisation des fonctions de la documentation officielle de PHP 7



Création de fonctions personnalisées et typage des paramètres et valeur de retour



Inclusion de scripts séparés par include et require



Exercices : implémenter des algorithmes en PHP

- ▶ Site statique: HTML CSS JAVASCRIPT
 - ▶ Consultation du site sans possibilité d'interagir
 - ▶ L'administrateur ajoute du contenu statique
 - ▶ Exemple : Site vitrine: Datayom.com
- ▶ Si il fallait réaliser un site dynamique :
- ▶ Ex : sur un site e commerce :
 - ▶ Contenu différent selon l'utilisateur qui se connecte sans crée une page pour chaque utilisateur / Interaction
 - ▶ Autre exemples ?
 - ▶ La documentation officielle de PHP : <https://www.php.net/manual/fr/> (en français)
 - ▶ OpenClassrooms : Concevez votre site web avec PHP et MySQL (en français)

- ▶ Le PHP (Hypertext Preprocessor) est un langage de programmation (impératif) à l'inverse des langages HTML et CSS qui sont des langages déclaratifs (structure et mise en forme de page web).
- ▶ Le PHP est compilé, c'est-à-dire transformé en langage binaire, au moment de l'exécution du script.
- ▶ La première version du langage PHP fut créée en 1994 par Rasmus Lerdorf et repris en 1997 par deux étudiants : Andi Gutmans et Zeev Suraski.
- ▶ Dans le monde, plus de 300 millions de sites reposent sur la technologie Php. Ce chiffre ne cesse d'augmenter.
- ▶ <https://fr.wikipedia.org/wiki/PHP#Histoire>

► Définir

- Un client
- Un serveur
 - Comment est t il identifié ? (IP / PORT)
- Expliquez le principe de l'architecture client serveur en général avec l'exemple de l'imprimante
 - Quelles sont les protocoles de communications
- Une page statique
 - Expliquez le principe de l'architecture client serveur avec HTML CSS JAVASCRIPT
- Une page dynamique
 - Expliquez le principe de l'architecture client serveur avec PHP

- Les applications développées pour les sites internet reposent sur le modèle client serveur.



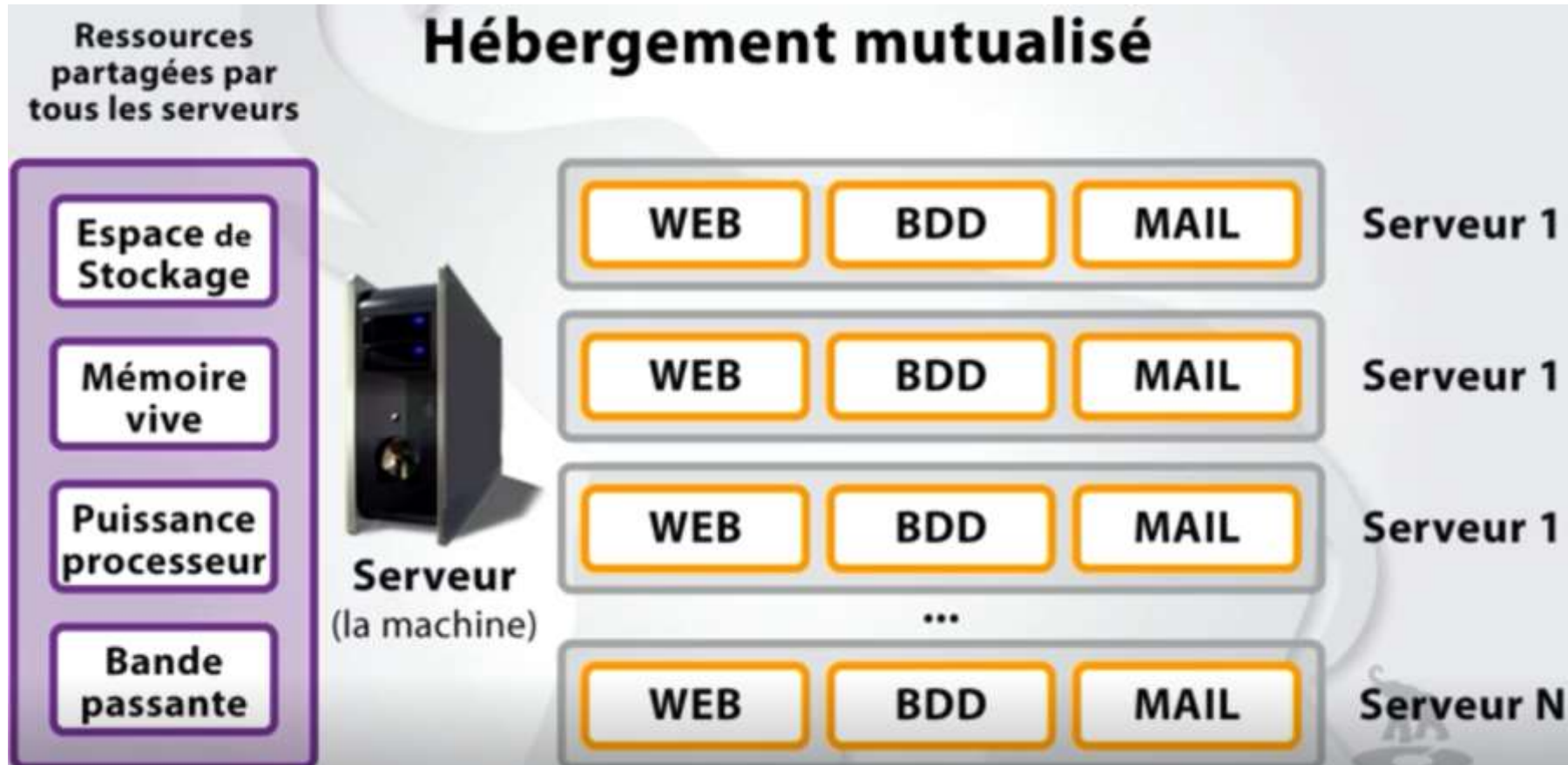
- ▶ Transposé dans le monde informatique :
 - ▶ Le client est une application qui s'exécute sur un ordinateur personnel
 - ▶ Le serveur est une autre application
 - ▶ Qui gère des ressources partagée (les livres)
 - ▶ Et qui s'exécute le plus souvent sur un ordinateur distant (établit une relation entre les clients et les ressources disponibles en bibliothèque)
 - ▶ Lorsque nous accédons à un site internet, nous utilisons la technologie Client-Serveur

Centre d'hébergement



Définition de l'hébergement : Mise à disposition d'un serveur (machine) disposant d'un espace disque et de serveurs d'applications (web, mail, FTP...) accessibles en permanence depuis Internet

Hébergement mutualisé



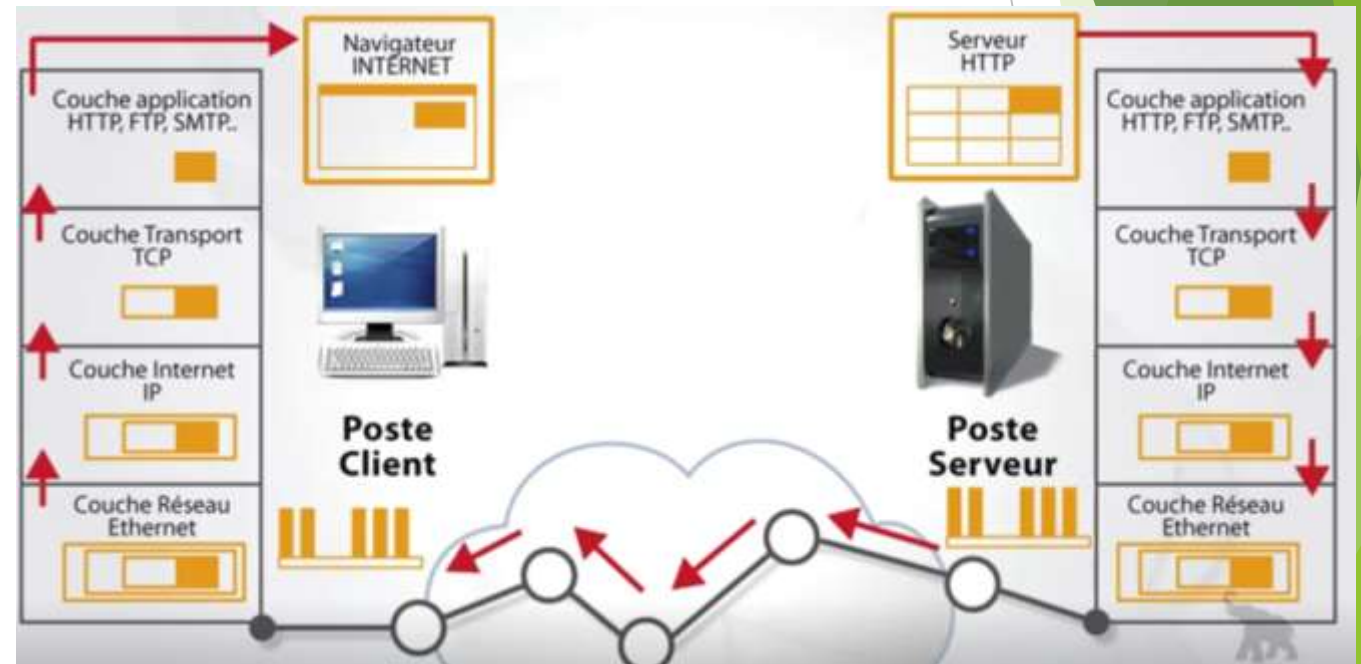
• Offre compacte incluant l'hébergement le domaine, les mails et une base de données pour un prix très attractif (entre 1 à 20 € par mois)

ouvre

ent, mail & BD

- ▶ Pour les sites on peut travailler en local mais nous avons
- ▶ [Nevada - Free HTML Bootstrap Template](#)
- ▶ une architecture sur le réseau internet même uniquement avec les langages
 - ▶ HTML
 - ▶ CSS
 - ▶ JAVASCRIPT
- ▶ => HTTP:// Serveur ?

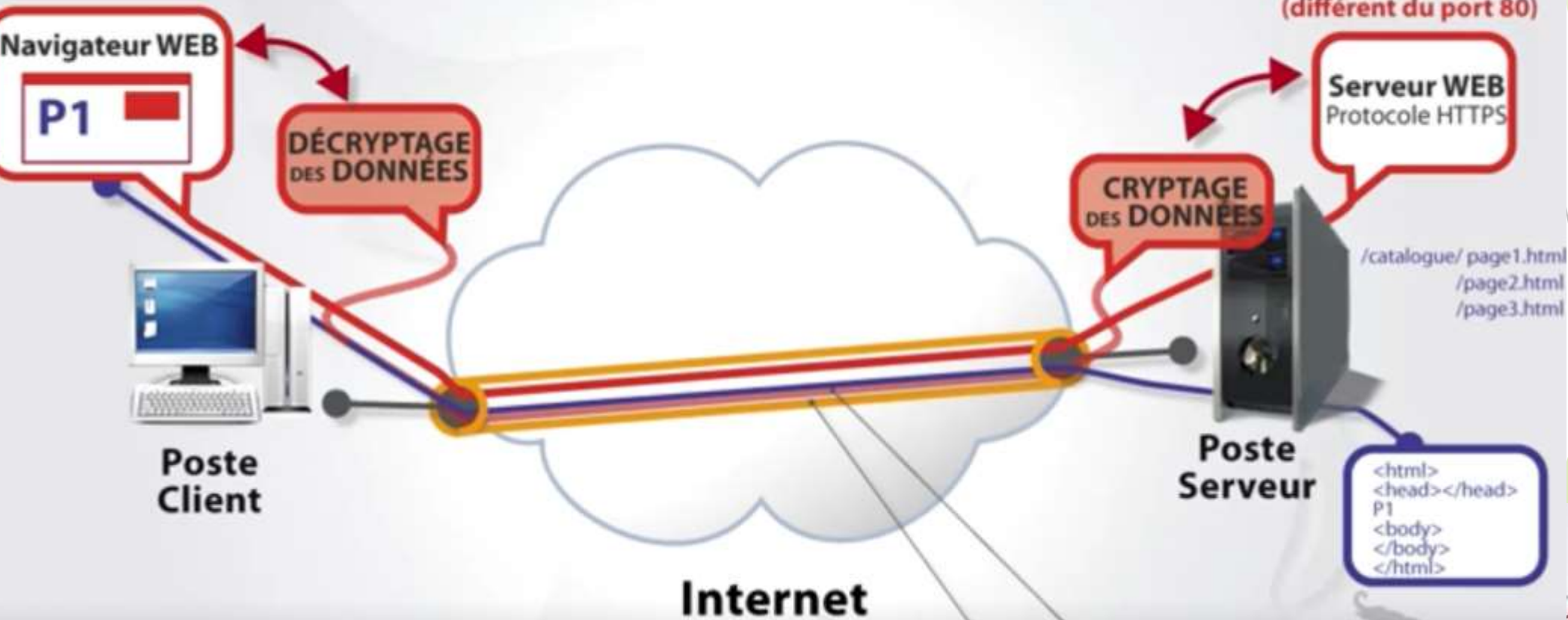
- Les 4 couches tcp / IP
- Permet de transporter un datagramme / paquet d'un poste à l'autre
- Couche Application : Chaque application est associée à un protocole : Découpe les paquet avec entête indiquant le protocole en entête.
 - Navigateur internet : HTTP
 - Transfert de fichier : FTP
 - Envoie de mail : SMTP
- Couche TCP / IP : Même protocole TCP / IP installé. Nouveau paquet indiquant le protocole.
- Couche réseau : Protocole ethernet : Mettre en forme les trames numériques qui permettront de transférer une information à l'autre. Nouveau paquet transformant en trame binaire.



- ▶ Serveur
 - ▶ https://fr.wikipedia.org/wiki/Serveur_informatique
- ▶ Serveur HTTP
 - ▶ https://fr.wikipedia.org/wiki/Apache_HTTP_Server
- ▶ Serveur Apache
 - ▶ https://fr.wikipedia.org/wiki/Apache_HTTP_Server

L'application WEB : protocole HTTPS

(HyperText Transfer Protocole Secured)



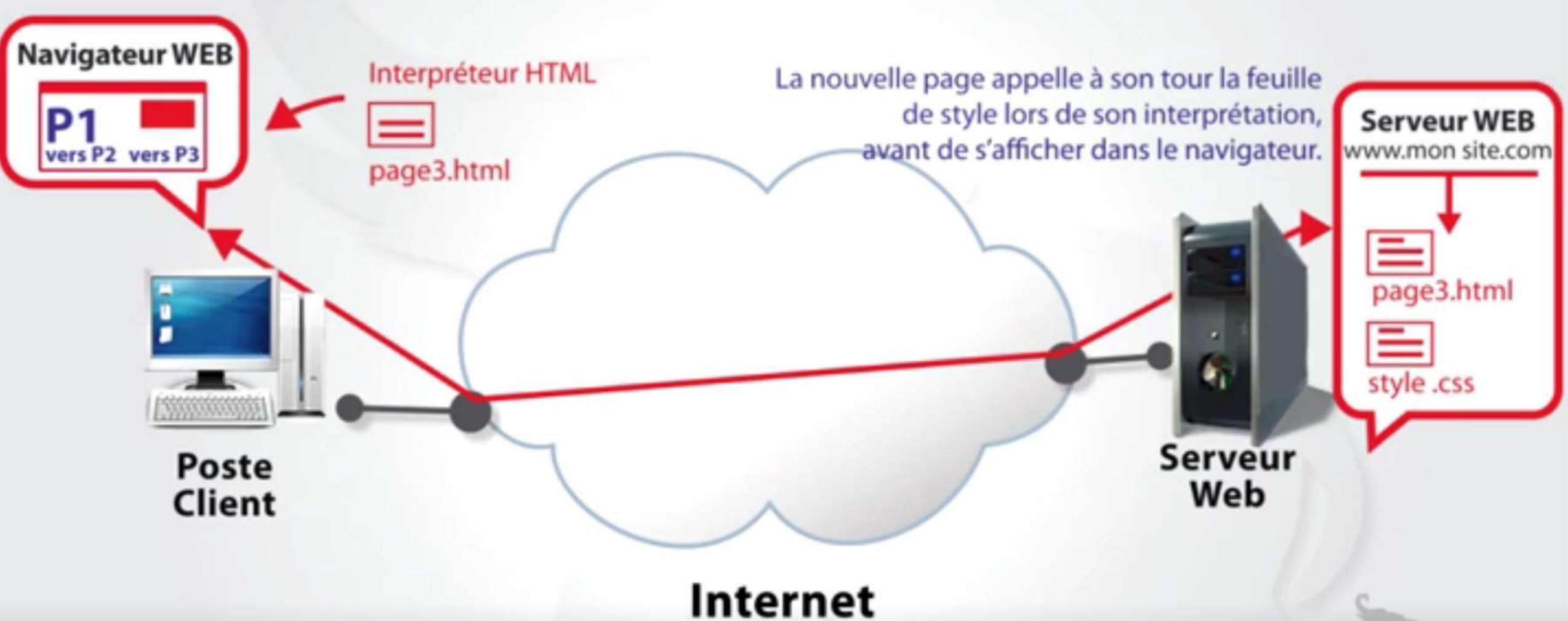
Barre d'adresse :

				
Nom :	FireFox	Internet Explorer	Safari	Chrome
Editeur :	Mozilla	Microsoft	Apple	Google

Barre d'adresse :



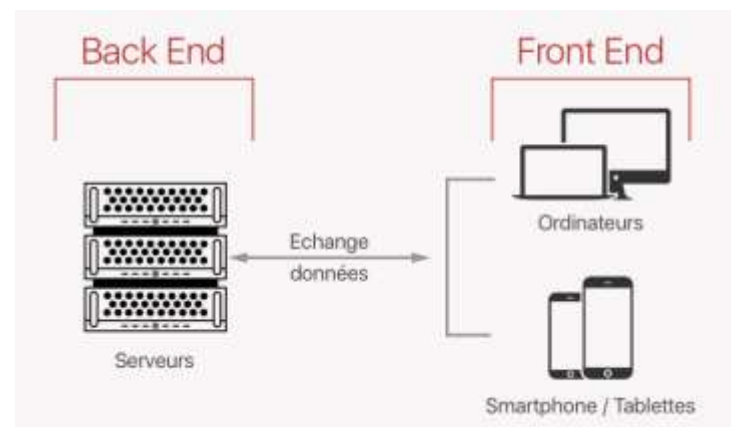
<http://www.monsite.com/page3.html> - Site web statique html



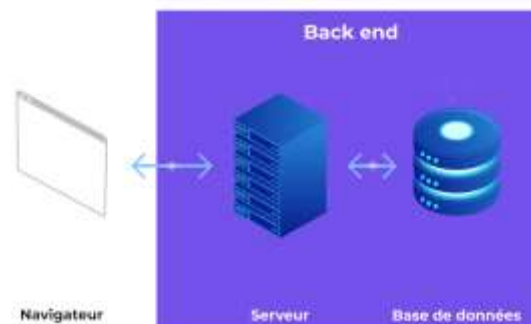
Site web statique html

- ➖ Tout le contenu HTML de la page est envoyé au navigateur car son interprétation est faite côté client. L'utilisateur peut donc accéder librement au code source de la page.
- ➕ Désormais, le HTML (Hyper Text Markup Langage) est remplacé par le xHTML (eXtensible Hyper Text Markup Langage), une évolution du précédent conforme aux contraintes du XML.
- ➕ Pour le xHTML strict, une feuille de style CSS doit obligatoirement accompagner la page Web. La page xHTML définissant la structure des données et la feuille de style CSS leur présentation.
- ➖ L'interactivité d'une simple page xHTML **est limitée à l'usage des liens hypertextes**. Pour créer plus d'interactivité entre l'utilisateur et le système, **il faut faire appel à d'autres technologies**.

FRONT END / BACK END







Frontend  Backend



Site web interactif côté client

- Le client peut désactiver le JAVASCRIPT

-  L'ajout d'un programme Javascript dans une page Web **ne nécessite pas de modifier la structure du serveur**
-  Comme le code JavaScript s'exécute sur le poste client, **il est très rapide** et conviendra aux programmes où la réactivité est importante (comme un roll-over par exemple).
-  Par contre, le fait qu'il s'exécute sur le poste client le rend **tributaire d'éventuels problèmes de compatibilité** avec la version du navigateur et le type d'ordinateur.
-  Le code du programme étant visible dans le code source de la page, il peut être copié, voir modifié par une personne mal intentionnée, ce qui constitue de réels **problèmes de protection des sources et de sécurité.**

► Site statique

- Exemple : un profil utilisateur spécifique
- HTML , CSS , JAVASCRIPT : Je crée une page pour chaque utilisateur...

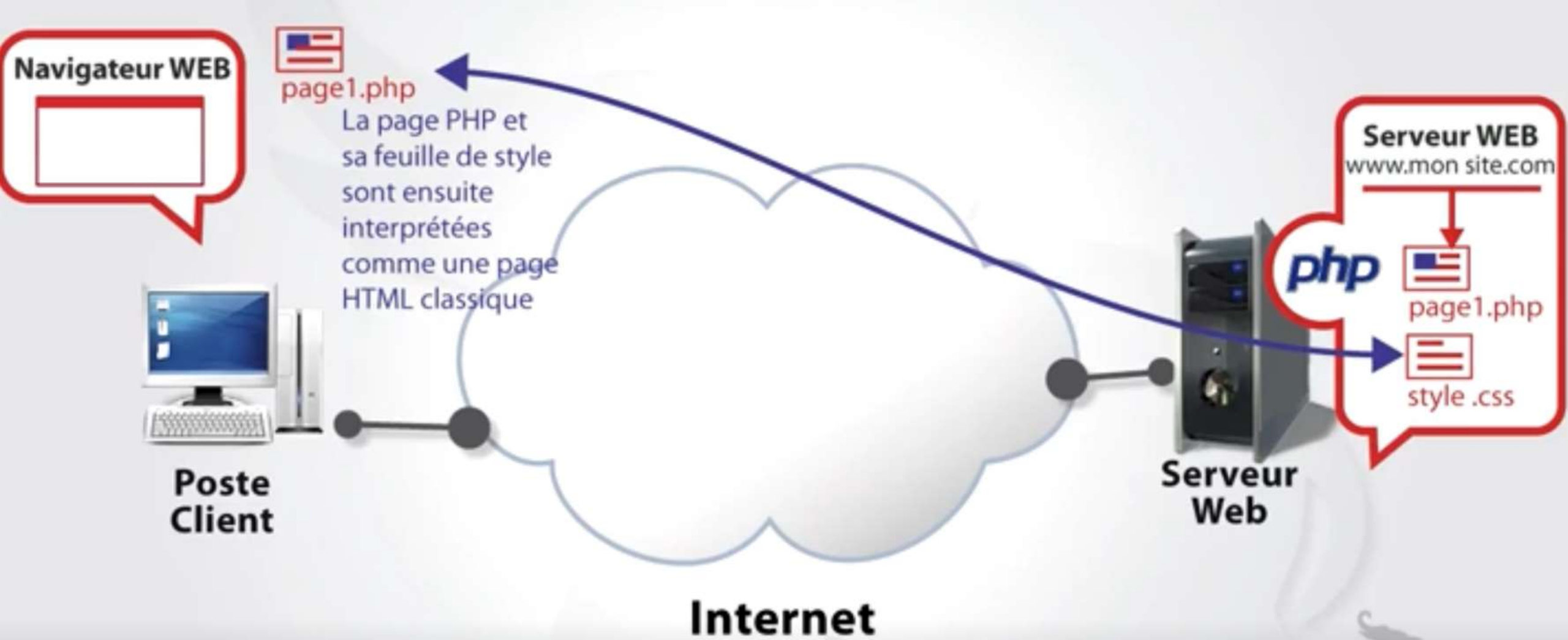
► Pour faire un site dynamique :

- Site dynamique : Il faudrait :
 - un générateur de page HTML, CSS, JAVASCRIPT qui se génère selon chaque utilisateur automatiquement à l'aide d'une page standard de profil utilisateur
 - Une Base de données qui affiche l'utilisateur correspondant.

- ▶ un générateur de page HTML, CSS, JAVASCRIPT qui se génère selon chaque utilisateur automatiquement à l'aide d'une page standard de profil utilisateur
- ▶ PHP
 - ▶ <https://fr.wikipedia.org/wiki/PHP>
- ▶ Base de donnée
 - ▶ https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es
 - ▶ <https://fr.wikipedia.org/wiki/MySQL>

Site web interactif côté serveur : PHP

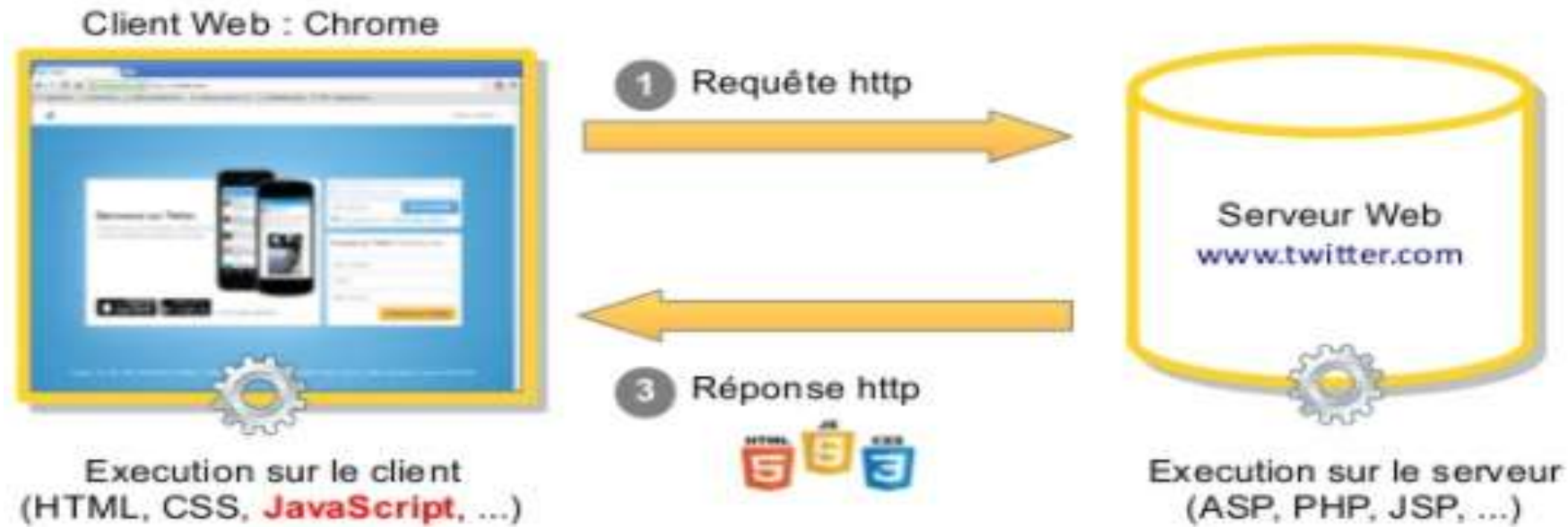
<http://monsite.com/page1.php>



PHP : Présentation : Langage côté serveur

Les fichiers JavaScript sont hébergés sur un serveur, à côté de vos fichiers HTML et CSS :

1. Le navigateur fait une requête vers le serveur.
2. le PHP, côté serveur : langages exécutés sur le serveur.
3. Le navigateur affiche une page HTML issue du PHP





Architecture client serveur avec BDD



- ▶ Le poste client qui se connecte sur une page PHP
- ▶ <https://ymelki.go.yj.fr/mapage.php>
- ▶ Le Serveur PHP va lire le code PHP va l'interpréter.
 - ▶ Si il trouve du code SQL, il va interroger un serveur SQL
- ▶ Il va continuer à interpréter le code PHP.
- ▶ Il va transformer tout ce qu'il a lu en HTML et le renvoyer au client.
- ▶ Le client va lire uniquement ce qu'il a besoin

Site web interactif côté serveur : PHP

- A chaque exécution une requête http est exécuté : Temps de réponse long VS JAVASCRIPT

-  Le programme s'exécutant sur le serveur Web, **il n'est donc pas sensible aux problèmes de compatibilité avec les navigateurs et le type d'ordinateur côté client.**
-  Après l'exécution du programme PHP par le préprocesseur, aucune ligne de code PHP ne subsistera dans le code source de la page envoyée au navigateur. **Des programmes de sécurité peuvent donc être créés en PHP.**
-  Les pages envoyées au navigateur ne contenant que du HTML, il n'y a aucun risque que l'exécution du programme soit désactivée par l'utilisateur. **Il n'est pas utile de prévoir des solutions alternatives au programme.**
-  Pour que le programme PHP soit exécuté sur le serveur, **il est nécessaire de disposer d'un processeur PHP.** Néanmoins, actuellement la plupart des hébergeurs proposent dans leur offre un processeur PHP par défaut.

Site web interactif côté serveur avec base de donnée: PHP / mysql

-  Les sites dynamiques ont les **mêmes avantages** et **inconvénients** que les sites interactifs côté serveur puisqu'ils sont basés sur la même structure de serveur Web.
- 
-  Ils ont en plus l'avantage de pouvoir intégrer dans les pages PHP des informations stockées dans une base de données, ce qui permet de **mettre en place une interface d'administration en ligne pour que l'utilisateur puisse mettre à jour facilement les différentes informations et média du site.**

- ▶ Serveurs HTTP (ou serveur web) :
 - ▶ **Apache (PHP)**, Tomcat (Java, JEE), IIS (C#, .Net)
- ▶ Langages côté serveur :
 - ▶ SQL, **PHP**, C#, Java, Python, Ruby, etc
- ▶ Serveurs SGBD :
 - ▶ MySQL, Oracle, SQL-Server, PostgreSQL, MariaDB, SQLite
- ▶ Framework
 - ▶ Symfony, Laravel, Zend,

- ▶ “PHP est un langage de script exécuté côté serveur, ainsi que son interpréteur.”
- ▶ Historique
- ▶ Définition
- ▶ Langage : Liste ...
- ▶ Programme : Algorithme, traitement de données

► Architecture

- Page statique : page renvoyé au client tel qu'elle est écrite sur le serveur
- Page dynamique : Page généré par le serveur
- Schéma
- Notion de serveur
 - Machine / IP / Hostname / Port
 - Architecture client Serveur

► Environnement PHP

- WAMP

The background of the slide features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect.

► Historique

- <https://fr.wikipedia.org/wiki/PHP#Histoire>

► Exercice : Définir

- Un client
- Un serveur
 - Comment est t il identifié ? (IP / PORT)
- Expliquez le principe de l'architecture client serveur en général avec l'exemple de l'imprimante
 - Quelles sont les protocoles de communications
- Une page statique
 - Expliquez le principe de l'architecture client serveur avec HTML CSS JAVASCRIPT
- Une page dynamique
 - Expliquez le principe de l'architecture client serveur avec PHP

- ▶ Base de donnée SQL : Stock les données par exemple des articles du site e commerce
- ▶ Page PHP: Page générique de gabarit de visualisation d'un article
- ▶ HTML / CSS / Javascript : Structure de la page renvoyé au navigateur

Architecture client serveur - installation de serveurs locaux

- ▶ Visual studio code
- ▶ Travailler en local
 - ▶ <https://www.wampserver.com/>
 - ▶ **W**INDOWS
 - ▶ **A**PPACHE
 - ▶ **M**YSQL
 - ▶ **P**HP
- ▶ <https://www.php.net/>

► Lancer wampserver

- Localhost
- C:\wamp64
- Dossier www / htdocs
- Créez un dossier « cours_php »
- Afficher Hello world
 - On remarque que le html est généré
 - Balise PHP
 - Insérer du HTML
 - Se termine par un ;

► Exercice

1. Quelles sont limites d'un site statiques html / javascript / css ?
2. Qui a été créé et comment a été créé PHP ?
3. Quel est l'architecture d'un site statique ?
4. Quel est l'architecture d'un site dynamique ?

- ▶ Hello world dans une page.php
 - ▶ Ca affiche « hello world » pourquoi ?
 - ▶ Du html
 - ▶ Modèle ou gabarit ?
- ▶ Le cheminement qui se déroule lorsque vous lancez votre page php
 - ▶ Qu'est ce qui s'exécute côté serveur ?
 - ▶ Les serveurs concernés ?

- ▶ Les commentaires :
 - ▶ `//` une seule ligne
 - ▶ `/*` plusieurs ligne: `*/`
- ▶ Variables : commence par '\$' puis un caractère alphabétique sans espace.
 - ▶ `Echo $variable ;`
 - ▶ `var_dump($variable)`
 - ▶ <https://www.php.net/manual/fr/function.var-dump.php>
 - ▶ `strtoupper` : renvoie la chaîne de caractères mise en majuscules
 - ▶ `strtolower` : renvoie la chaîne de caractères en minuscules
 - ▶ `str_shuffle` : renvoie les caractères de la chaîne mélangés
 - ▶ Type de variable

- ▶ Les variables : commence par un \$
 - ▶ Définition : Nom règlementé (voir doc)
 - ▶ Affectation : Valeur
 - ▶ Faiblement typée
 - ▶ Concaténation : avec le point
- ▶ Structure de données
 - ▶ Nombre : sans Guillement - Calcul
 - ▶ String : entre Guillement - Concaténation / double quote interprète les variables vs simple quote
 - ▶ Boolean : true / false - Algo
 - ▶ Tableau : entre crochet Adressage / Gestion
 - ▶ <https://www.php.net/manual/en/function.gettype.php>
- ▶ Opérateur : + addition - : soustraction - * : multiplication - / : division

► Les conditions : IF ELSE : echo \$a==\$b

Example	Name	Result
\$a == \$b	Equal	true if \$a is equal to \$b after type juggling.
\$a === \$b	Identical	true if \$a is equal to \$b, and they are of the same type.
\$a != \$b	Not equal	true if \$a is not equal to \$b after type juggling.
\$a <> \$b	Not equal	true if \$a is not equal to \$b after type juggling.
\$a !== \$b	Not identical	true if \$a is not equal to \$b, or they are not of the same type.
\$a < \$b	Less than	true if \$a is strictly less than \$b.
\$a > \$b	Greater than	true if \$a is strictly greater than \$b.
\$a <= \$b	Less than or equal to	true if \$a is less than or equal to \$b.
\$a >= \$b	Greater than or equal to	true if \$a is greater than or equal to \$b.
\$a <=> \$b	Spaceship	An int less than, equal to, or greater than zero when \$a is less than, equal to, or greater than \$b, respectively.

► Les conditions : IF ELSE ou Switch case

Example	Name	Result
\$a and \$b	And	true if both \$a and \$b are true .
\$a or \$b	Or	true if either \$a or \$b is true .
\$a xor \$b	Xor	true if either \$a or \$b is true , but not both.
! \$a	Not	true if \$a is not true .
\$a && \$b	And	true if both \$a and \$b are true .
\$a \$b	Or	true if either \$a or \$b is true .

► Exercice

- Créez une page « user.php »
- Insérer du html correspondant à une page html minimaliste servant à afficher un utilisateur qui pourra servir de modèle.
- Ajouter du code PHP
 - En mettant les balises PHP
 - Les variables générique à un utilisateur
 - Nom / Prenom / age / Email
 - Afficher les variables avec
 - Echo / Var_dump / Si l'age est > 17 => Majeure
 - Vous les concatenez dans un code html
- Expliquez ce qu'il se passe au niveau de serveur et de l'exécution du code

► Exercice

1. Afficher les nombre de 0 à 10 à l'aide d'une boucle
2. Afficher les nombre de 5 à 15 à l'aide d'une boucle
3. Afficher le caractère * 10 fois à l'aide d'une boucle

- For
- While

- afficher la table de 2 avec une boucle
- afficher la table de 2, 3 ... 10 avec deux boucle
- Afficher le caractère * 10 fois en ligne et 10 fois en colonne à l'aide de 2 boucles
- Faire une boucle permettant de faire la factorielle d'un nombre affichant directement la résultat & affichant les étapes
- Faire un triangle d'étoile avec 5 étoile

► Les tableau en PHP


► Array (indice)

- Echo pas adapté
- Print_r
- []= //affectation
- [3]=

► Associatif (clé valeur)

- \$joueur["nom"] = "Doe";
- \$joueur["prenom"] = "John";

► Fonction sur les tableau

- 
- ▶ Les boucles
 - ▶ FOR
 - ▶ FOREACH
 - ▶ DO WHILE
 - ▶ WHILE

- ▶ Les fonctions
 - ▶ Appels
 - ▶ Retourne ou non des résultats
 - ▶ Portée des variables : Variable locale / Variable globale
 - ▶ Appel / Déclaration / Arguments /
 - ▶ Addition
 - ▶ Echanger
 - ▶ Vérifier
 - ▶ Corp
 - ▶ <https://www.php.net/manual/en/indexes.functions.php>
 - ▶ Date
 - ▶ PHP.net function chaine

- ▶ Découper son code pour éviter d'avoir un code qui se répète autant de fois qu'on a de pages. (dynamiser le site)
 - ▶ `include 'footer.php';` // Ca inclut le fichier ; s'il existe pas => warning
 - ▶ `require 'footer.php';` // Ca inclut le fichier ; s'il existe pas => erreur fatale
 - ▶ `include_once 'footer.php';` // Comme include ; UNE seul fois
 - ▶ `require_once 'footer.php';` // Comme require ; UNE seule fois
- ▶ Créez une page html qui va se nommer « accueil ». Récupérer des codes HTML
 - ▶ Découpez la page au moins selon le header et le footer

- ▶ créer deux variables de type integer
- ▶ Additionner / Multiplier puis diviser les deux variables et afficher le résultat de l'opération
- ▶ créer deux variables de type String
- ▶ Concaténer les deux variables « chaîne de caractère » puis les afficher
- ▶ Enregistrer une variable prix et une variable tva puis calculer la tva puis revenir sans tva
- ▶ Réaliser cela sous forme de fonctions
 - ▶ Add / mult / div / moyenne / somme
 - ▶ Concatener
 - ▶ Pnet / Pbrut
 - ▶ Pair
 - ▶ calculerAge
 - ▶ SalaireEvol

- ▶ Créer une fonction pour calculer l'âge d'une personne
- ▶ - produire l'affichage des entiers de 1 à 10 à l'aide d'une boucle while / FOR
- ▶ Le mettre sous forme de fonction de 1 à 10 / de 1 à N / de m à n.
- ▶ Créer une boucle affichant tous les numéros de 1 à 20 en ajoutant une instruction de sortie de boucle si le numéro correspond à un numéro gagnant (généré aléatoirement)

- ▶ Formes géométrique d'un carré d'étoile
- ▶ écrire une boucle qui produit une ligne horizontale de 8 étoiles
- ▶ imbriquer ce code dans une nouvelle boucle pour produire un carré de 8 lignes horizontales, chacune contenant 8 étoiles
- ▶ Le mettre sous forme de fonction de 1 à 10 étoile / de 1 à N étoile / de m à n étoile

- ▶ Header Footer : Sépare les codes transverses des codes spécifiques
- ▶ Mettre les fonctions dans un fichier externe et les appeler avec include
- ▶ Variable civile : Bienvenue Mr / Mme / Fonction titre / Afficher age [generernb(40)
- ▶ Fonction des affichages Affiche des nombres de 1 à 100
- ▶ Fonction qui retournent une information ex : ispair(4) => true ispair(3)=> false return
- ▶ Valeur absolue | pair [trouverage
- ▶ Jeu du plus ou du moins
- ▶ Abs - floor - round - modulo
- ▶ Explode - implode - str replace - trim - substr - ucfirst
- ▶ Time date
- ▶ Ecrire une fonction pour multiplier 2 nombres entiers
- ▶ Ecrire une fonction pour additionner 2 nombres entiers
- ▶ Ecrire une fonction pour diviser 2 nombres entiers
- ▶ Ecrire une fonction pour compter
- ▶ Vérifier majeur
- ▶ Echanger variable
- ▶ Var_dump
- ▶ Page menu switch

- ▶ Page produit en HTML / CSS
- ▶ Tableau de produit
- ▶ Créez un tableau indicé de 5 utilisateurs / associatif
 - ▶ Trier
 - ▶ puis affichez la.
 - ▶ Boucle dessus en affichant les clé et valeur
 - ▶ Afficher le nombre de caractère de chaque utilisateur
- ▶ Faire un tableau associatif qui comprend des prix H.T. de différents articles
 - ▶ Un code qui va afficher puis modifier les prix en TTC
 - ▶ Mettez ce code sous forme de fonction
 - ▶ Un code qui va afficher les articles / afficher un prix ajouter une somme de 10€ / Modifier dans le tableau / Modification dans une fonction
 - ▶ Créez un tableau de plusieurs valeurs numérique
 - ▶ Grâce aux fonctions prédéfinis PHP afficher le nombre maximum / Minimum
 - ▶ Afficher la sommes de ces nombres. Ajouter un nombre à la fin . Vous triez . Vous refaites la somme.
 - ▶ Array_sum => somme . Créez une fonction somme équivalente / puis une fonction moyenne
 - ▶ Créez une fonction / code qui permet de générer des nombres aléatoire pour construire un tableau
 - ▶ Créez un tableau d'âge des utilisateur. Créez une fonction affichant l'âge et indiquant si la personne est majeur ou non.

- ▶ Ecrire une fonction pour trier
- ▶ Ecrire une fonction pour chercher
- ▶ Trouver_max
- ▶ Trouver_min
- ▶ ajouter
- ▶ somme d un tableau
- ▶ max / min d'un tableau
- ▶ trouver premier
- ▶ copiertab
- ▶ reduire

► Connexion à la base de donnée

- - Se connecter à la base de données : `mysqli`
- - Exécuter une requête : `query`
- - Préparer une requête avec des arguments
- - “Exécuter” une requête préparée
- - Récupérer les résultats d’une requête
- - Parcourir les résultats d’une requête
- - Compter les résultats d’une requête ou le nombre de lignes affectées par celle-ci
- - Clore une connexion à une base de données
- - Récupérer les erreurs intervenant à chacune de ces étapes