

Cours théorique de PHP



Jean-Luc Collinet

Haute École Léonard de Vinci : Paul Lambin
Bloc 1 du Bac en Informatique

Cours 1b

Pourquoi architecturer ? (1/2)

- **Non** à avoir **tout le code PHP, HTML et CSS dans un seul fichier PHP** par page HTML



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <title>Un premier script PHP</title>
6 </head>
7 <body>
8   <p><?php echo "Bonjour tout le monde !" ?></p>
9   <p>Nous sommes le <?php echo date('j/m/Y'); ?>.</p>
10  <p>Il est <?php echo date('H:i:s'); ?>.</p>
11  <p>
12    <form action="helloworld.php" method="GET">
13      Valeur à convertir
14      <input type="text" name="valeur" size="15" value="">
15      <br/><input type="radio" name="choix" value="toBTC" checked>Euros vers Bitcoin
16      <br/><input type="radio" name="choix" value="toEUR">Bitcoin vers Euros
17      <input type="submit" value="Convertir">
18    </form>
19  </p>
20 </body>
21 </html>
```

Pourquoi architecturer ? (2/2)

- **Non à la redondance**
 - Oui à un menu écrit une seule fois
 - Oui à un accès sécurisé écrit une seule fois
- **Non à la dépendance**
 - Oui à un changement facile de style
 - Oui à un changement facile de base de données

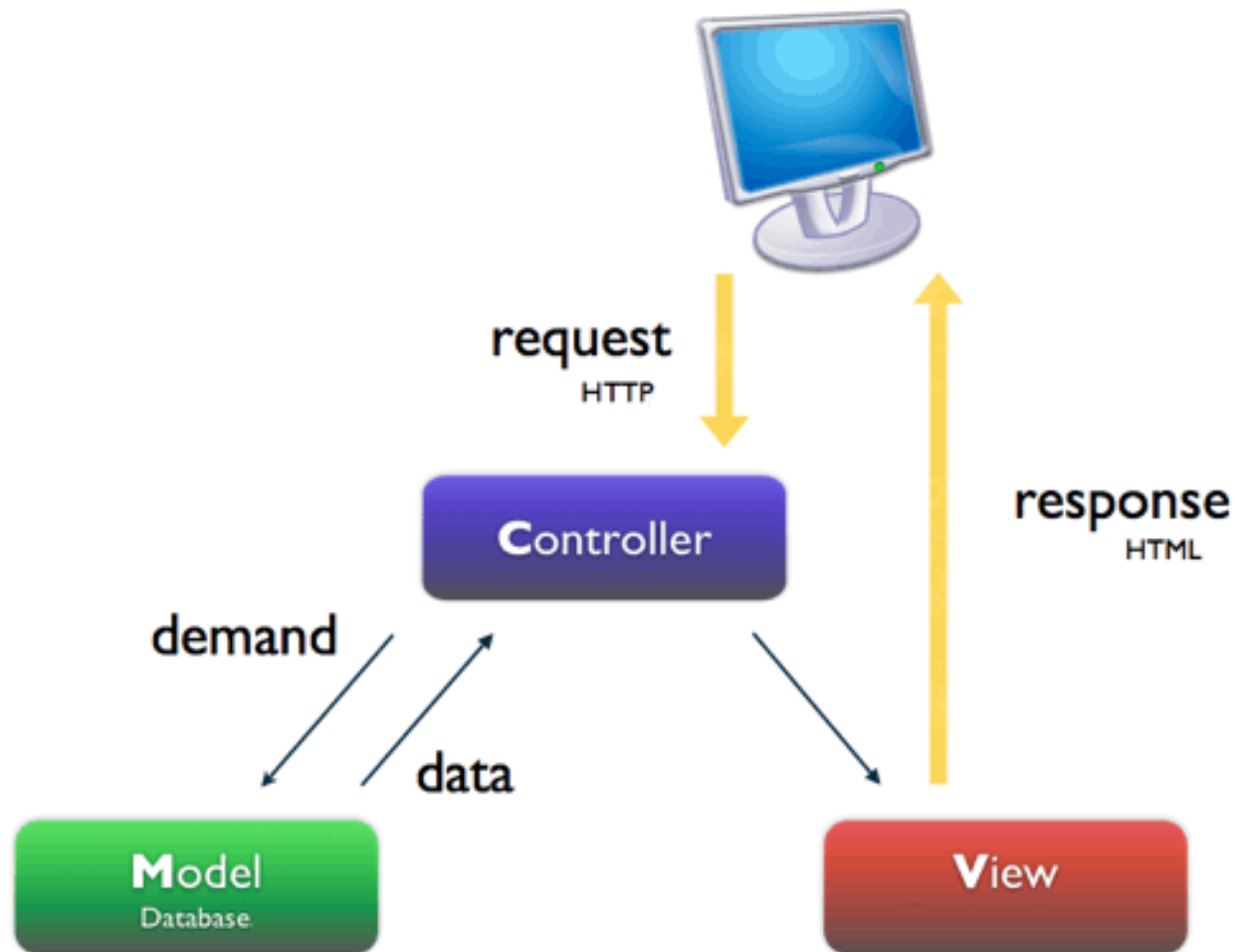
Architecture MVC

- Approche professionnelle tout de suite !
- Oui à la clarté de la structure
- Oui à la séparation des couches
 - Les données \equiv *Model*
 - La présentation \equiv *View*
 - Les traitements \equiv *Controller*

Architecture MVC

- Principe de conception de site Web basé sur la séparation de 3 fonctions essentielles en 3 parties distinctes
 - Modèle : gérer les données
 - Vue : gérer l'affichage (UI : user interface)
 - Contrôleur : gérer les actions

Schéma de fonctionnement



Un 1^{er} exemple de site Web

- Semaine 1 - Ex. PHP 2 - Code Source/(m)vc
 - index.php
 - controllers/AccueilController.php
 - views/header.php
 - views/accueil.php
 - views/footer.php
- **Démonstration en pratique...**

Architecture : Racine d'un site

- Unique pour un site Web
- Le fichier `index.php`
- Initialise des variables globales
- Dispatche le travail algorithmique vers un des contrôleurs selon l'action voulue

Architecture : les Contrôleurs

- Conçus en PHP OO, autant de classes que nécessaires
- Implémentent des actions, des traitements, des algorithmes
- Appellent des fonctions de gestion des données
- Incluent à la fin le code d'une vue à afficher

Architecture : les Vues

- Conçues en **HTML** et **CSS** à côté
- **Affichent les résultats** des traitements effectués par un contrôleur
 - **Code PHP admis** dans une vue
 - Des **echo** de variables
 - Des **tests** pour afficher ou pas qqchose
 - Des **boucles** pour afficher les éléments d'un tableau
 - **Code NON admis** dans une vue
 - Traitements et accès aux données de la DB !

Architecture : les Vues

- *Template* : découpe en 3 parties
 - Le *header* : l'en-tête commun à toutes les vues
 - Le *corps* différent pour chaque vue
 - Le *footer* : le pied commun à toutes les vues
- Le *menu principal* du site est écrit une seule fois dans le *header*

Architecture : le Modèle

- Les données sont **persistantes** dans une base de données **MySQL**
- Une classe conçue en PHP regroupe **toutes les fonctions d'accès aux données**
- **Une classe par entité** de la DB
 - Livre
 - Utilisateur
 - ...