



# Le Framework PHP ORMFramework

## Sommaire

- **Présentation**
- **Le langage de développement**
- **Les fonctionnalités**
- **Un framework modulaire**
- **La structure du framework**
- **Les frameworks du même type déjà en place**
- **Comparaison avec Symfony**
- **TP : Webservice météo**

## Présentation

Le développement de l'ORMFramework a commencé en 2015 lorsque j'ai eu le besoin de développer des Webservices MVC.

J'en ai développé plusieurs et je me suis aperçu qu'ils suivaient tous le même modèle.

J'ai donc réfléchi à développer un outil générique qui prendrait en compte tous mes besoins et qu'on pourrait dupliquer et modeler assez simplement selon mes futurs besoins.

## Le langage de programmation

Étant donné que le langage PHP est mon langage de prédilection et un langage dans lequel je développe tous mes sites web, ce framework a été développé en PHP.

Il est compatible de la version 5 jusqu'à la dernière version de PHP.

## Les fonctionnalités

ORM Framework comporte différentes fonctionnalités :

- Un ORM permettant de générer un webservice potentiellement fonctionnel en quelques secondes.
  - Des commandes permettant de lancer l'orm, de créer une nouvelle strate ( un ensemble de modèle, contrôleur, entité, et fichier de configuration ), de supprimer une ou plusieurs strates, activer ou désactiver un module, ...
  - Différents services utilisables tel qu'un service de templates, de traduction, un service MySql et PostgreSQL, et un service pour parser un fichier HTML ou XML.
  - La possibilité de développer des modules ou des services et de les ajouter au framework très simplement.
  - L'utilisation facultative mais préférée des Annotations.
- 
- Il y a d'autres fonctionnalités que je ne cite pas dans le diaporama car elles ne sont pas encore fonctionnelles

## Un framework modulaire

ORMFramework est présenté comme une sorte de kit où il y a un certain nombre de modules qui font partis des fondations, des modules fournis et des possibilités infinies de nouveaux modules : Libre à votre imagination.

La structure d'un module :

- Un fichier 'autoload.php'
- votre code avec autant de fichiers et de répertoires que vous voulez.

Pour que votre module soit pris en compte par le framework :

- ouvrez un terminal sur votre linux ou votre mac, mettez vous à la racine du répertoire du framework et tapez :
- `php ormframework.php add module`

Il y a aussi la possibilité de récupérer un framework existant et de l'ajouter à l'ORMFramework en tant que module et de wrapper son utilisation sous forme de service pour avoir une syntaxe uniforme.

## La structure du framework

Le framework est structuré en 4 parties :

- Les fichiers `index.php` et `.htaccess` et `autoload.php` qui sont à la racine.
- Le fichier `ormframework.php` est à la racine et qui sert à exécuter les différentes commandes du framework.
- Un répertoire `Core` qui contient le code fondamental du framework et le code des différents modules.
- Un répertoire `Custom` qui, comme son nom l'indique, contient le code du projet écrit par l'utilisateur ou le code généré par l'ORM.

## Les frameworks de même type déjà en place

Parmi les frameworks PHP existants les plus connus, nous avons :

- `Symfony`
- `Zend Framework`
- `CodeIgniter`

Je vais plus particulièrement m'attarder sur les ressemblances et différences entre `ORMFramework` et `Symfony`

## Comparaison avec `Symfony`

`Symfony` a bien des qualités mais c'est un framework qui est aussi lourd en utilisation qu'en poids.

Un problème que j'ai surmonté grâce à la modularité et à la simplicité des syntaxes.

Je vais faire la comparaison avec les fonctionnalités que j'ai cité un peu plus haut.

<u>Fonctionnalités</u>	<u>ORM Framework</u>	<u>Symfony</u>
<u>ORM</u>	<u>OUI</u>	<u>OUI</u>
<u>Commandes</u>	<u>OUI</u>	<u>OUI</u>
<u>Services</u>	<u>OUI</u>	<u>OUI</u>
<u>Développement de services personnalisés</u>	<u>OUI</u>	<u>OUI</u>
<u>Annotations</u>	<u>OUI</u>	<u>OUI</u>
<u>Système modulaire</u>	<u>OUI</u>	<u>OUI</u>

On voit que toutes les fonctionnalités présentées de l'ORMFramework sont présentes dans `Symfony` mais dans `ORMFramework`, elles sont présentes soit sous une forme allégée soit sous une forme moins lourde à customiser.

- Certaines fonctionnalités ne portent pas le même nom dans `symfony` ou ne se présentent pas sous la même forme.

Exemples:

- Les “modules” dans ORMFramework sont appelés “Composants” dans Symfony.
- Les “Annotations” dans ORMFramework sont présentées sous forme de classe avec des méthodes statiques et dans Symfony sous forme de commentaires.

## TP

Pour illustrer tout ça j'ai prévu un petit TP.

Le TP consiste à générer un webservice de météo.

-

Les prérequis sont :

- php 5.6 +
- mysql
- Linux ( Ubuntu / Debian - idéal )
- git
- composer
- wget

Pour commencer récupérez le zip suivant

dézippez le fichier sur votre serveur, placez vous à l'endroit où à été dézipé le fichier puis lancez la commande :

- php demo\_meteo 0.0.1 test.php demo\_meteo

puis tapez sur entré.

le paramètre est le nom du fichier où va être installer le framework.

une fois que le framework est installé, installez la base de donnée qui se trouve dans le répertoire "Custom/Sql" -> meteo\_test\_ormframework.sql sur votre serveur.

-

Vous pourrez ensuite accéder à la documentation : en local ou en production

-

ensuite, pour ajouter une connection en base de donnés que vous pourrez réutiliser dans le code, tapez la commande :

- php ormframework.php add\_dbconnect

et suivez les instructions.

Le framework est en mode production par default c'est à dire que certaines commandes ne seront pas executables comme par exemple "destroy project" qui comme son nom l'indique, détruit l'application.

-

Pour changer de mode, tapez la commande suivante :

- php ormframework.php change\_mode dev / prod

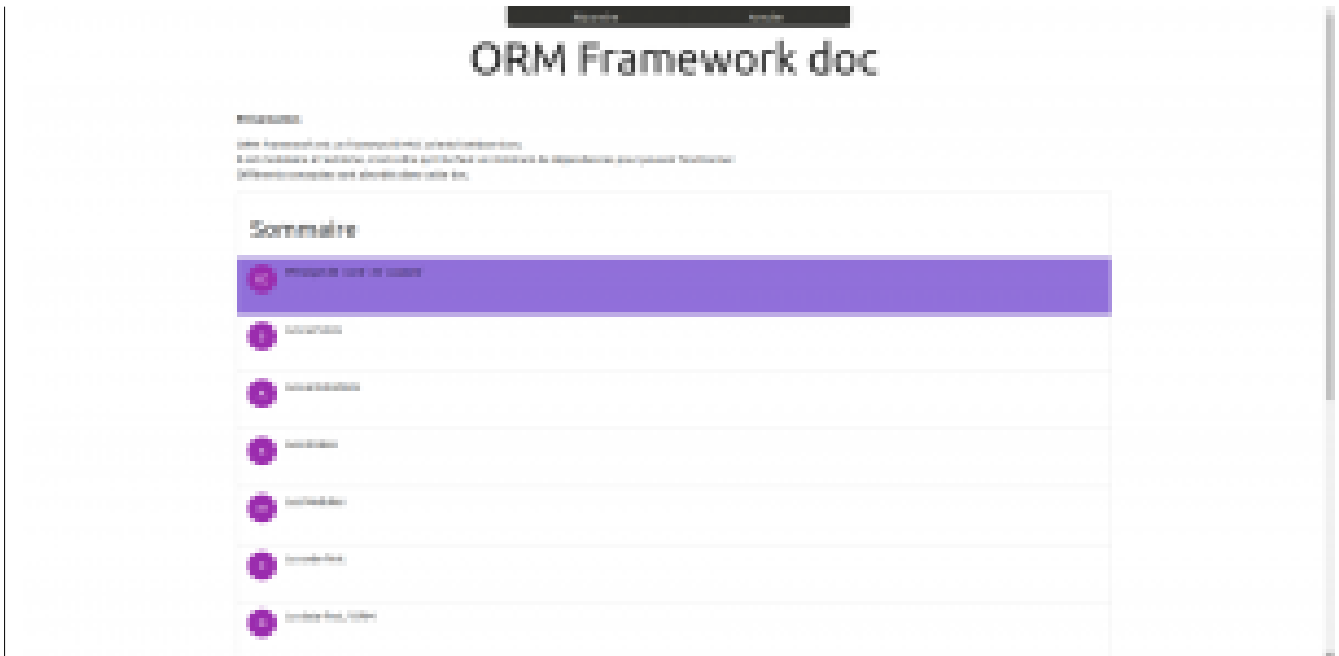
-

Autres particularités :

Le webservice en lui même est exploitable en lignes de commandes en tapant la commande suivante :

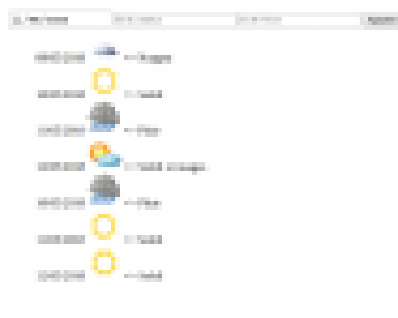
- `php index.php dev/app <controller> <methode> <parametre 1> <parametre 2> ...`

## Rendu



Si vous tapez <url du site>/doc

### Application de démo sur le thème de la météo



Si vous tapez <url du site>

