

# SYMFONY: 1-Installation COMPOSER et PHP

## Introduction #

Composer est un outil de gestion des dépendances en PHP. Il vous permet de déclarer les bibliothèques dont votre projet dépend et il les gère (installe / met à jour) pour vous.

## Gestion des dépendances #

Composer n'est **pas** un gestionnaire de packages au même sens que Yum ou Apt. Oui, il traite des "packages" ou des bibliothèques, mais il les gère par projet, en les installant dans un répertoire (par exemple `vendor`) à l'intérieur de votre projet. Par défaut, il n'installe rien du monde. Il s'agit donc d'un gestionnaire de dépendances. Il prend cependant en charge un projet "global" pour plus de commodité via la commande [globale](#).

Cette idée n'est pas nouvelle et Composer est fortement inspiré du [npm](#) de node et du [bundler](#) ruby.

Supposer (suppose):

1. Vous avez un projet qui dépend d'un certain nombre de bibliothèques.
2. Certaines de ces bibliothèques dépendent d'autres bibliothèques.

Compositeur (composer):

1. Vous permet de déclarer les bibliothèques dont vous dépendez.
2. Découvre quelles versions de quels packages peuvent et doivent être installés, et les installe (ce qui signifie qu'il les télécharge dans votre projet).

Composer est un outil pour gérer les dépendances en PHP (gestionnaire de librairies). Les dépendances, dans un projet, ce sont toutes les bibliothèques dont votre projet dépend pour fonctionner. Par exemple, si votre projet utilise la bibliothèque SwiftMailer pour envoyer des e-mails, il « dépend » donc de SwiftMailer. Autrement dit, SwiftMailer est une dépendance dans votre projet.

**Composer permet d'ajouter des librairies au sein de son projet, quand nous avons besoin d'une librairie, composer va la télécharger, de bien la ranger au sein de notre projet, donc de pouvoir l'utiliser très rapidement. ([getcomposer.org](http://getcomposer.org))**

Composer a donc pour objectif de vous aider à gérer toutes vos dépendances. En effet, il y a plusieurs problématiques lorsqu'on utilise des bibliothèques externes :

- Ces bibliothèques sont mises à jour. Il vous faut donc les mettre à jour une à une pour vous assurer de corriger les bogues de chacune d'entre elles.
- Ces bibliothèques peuvent elles-mêmes dépendre d'autres bibliothèques. En effet, si une de vos bibliothèques dépend d'autres bibliothèques, cela vous oblige à gérer l'ensemble de ces dépendances (installation, mises à jour, etc.).
- Ces bibliothèques ont chacune leur paramètres d'autoload, et vous devez gérer leur autoload pour chacune d'entre elles.

Composer va nous aider dans chacune de ces tâches.

## Installer Composer

---

### Sous Windows

Si vous êtes sous Windows, la méthode la plus facile est d'installer Composer via son exécutable d'installation, comme n'importe quel autre logiciel (mais n'oubliez pas que Composer est un outil fait 100% en PHP 😊).

Je vous invite à le télécharger depuis ce lien : [getcomposer.org/Composer-Setup.exe](https://getcomposer.org/Composer-Setup.exe). Suivez les instructions, en quelques clics vous aurez installé Composer !

Cette méthode fonctionne également avec Windows en fait, exécutez simplement la commande suivante :

```
php -r  
"eval('?'>'.file_get_contents('http://getcomposer.org/installer  
' ));"
```

Cette commande télécharge et exécute le fichier PHP suivant : [getcomposer.org/installer](https://getcomposer.org/installer). Vous pouvez aller le voir, ce n'est pas Composer en lui-même mais son installateur. Il fait quelques vérifications (version de PHP, etc.), puis télécharge effectivement Composer dans le fichier `composer.phar`.

### Dans tous les cas

Vous pouvez déjà exécuter Composer pour vérifier que tout est OK :

```
C:\wamp\www>php composer.phar --version  
Composer version 1.6.3 2018-01-31 16:28:17
```

Si vous avez installé Composer avec l'installateur Windows, ou si vous l'avez défini globalement, vous pouvez l'exécuter simplement via `composer --version`, sans devoir exécuter spécifiquement le PHAR. N'hésitez pas à mettre à jour Composer lui-même de temps en temps. Il faut pour cela utiliser la commande `self-update` de Composer, comme suit :

```
C:\wamp\www>php composer.phar self-update  
Updating to version 1.7.2 (stable channel).  
Downloading (100%)  
Use composer self-update --rollback to return to version 1.6.3
```

Et voilà ! Mon Composer est à jour.

## Vérifier l'installation de PHP en console

---

1. Allez dans les paramètres système avancés  
: *Démarrer > Panneau de configuration > Système et sécurité > Système > Paramètres système avancés* ;
2. Cliquez sur le bouton `Variables d'environnement...` ;
3. Regardez dans la section `Variables utilisateurs` ;
4. Trouvez l'entrée `Path` et sélectionnez-la ;

5. Cliquez sur Modifier ;
6. Cliquez sur Nouveau puis entrez votre répertoire PHP à la fin. C'est le répertoire dans lequel se trouve le fichier php.exe . Dans mon cas par exemple, je mets : C:\wamp\bin\php\php7.1.0 ;
7. Confirmez en cliquant sur OK. Vous devez ensuite redémarrer l'invite de commandes pour prendre en compte les changements.