

# Symfony Partie 0

## Installation

### Description du thème

Ce document présente l'installation des outils nécessaires à la réalisation d'une suite de TP au sein du Framework Symfony.

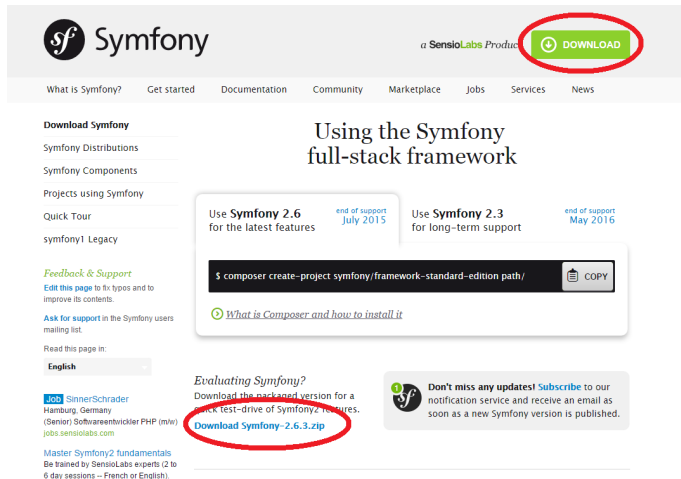
Propriétés	Description
<b>Intitulé long</b>	Initiation à Symfony par une suite de TP – Partie 0 : installation
<b>Formation concernée</b>	BTS Services informatiques aux organisations
<b>Matière</b>	SLAM 4
<b>Présentation</b>	Comprendre les concepts de base de Symfony
<b>Transversalité</b>	SLAM 5
<b>Notions</b>	<p>D4.1 - Conception et réalisation d'une solution applicative</p> <p>Savoir-faire :</p> <ul style="list-style-type: none"><li>• Programmer au sein d'un Framework</li></ul> <p>Savoir :</p> <ul style="list-style-type: none"><li>• Caractéristiques d'un Framework</li><li>• Persistance et couche d'accès aux données, technologies et techniques associées</li><li>• Architectures applicatives : concepts avancés</li></ul>
<b>Pré-requis</b>	<p>Programmation objet</p> <p>Pattern MVC</p> <p>Pattern DAO</p> <p>Serveur web installé</p>
<b>Outils</b>	Un environnement de développement pour le web
<b>Mots-clés</b>	Symfony, Framework, PHP
<b>Durée</b>	30min
<b>Auteur(es)</b>	Luc Frébourg
<b>Relectures</b>	Gaëlle Castel
<b>Version</b>	v 1.0
<b>Date de publication</b>	Avril 2015

L'environnement de travail de cette série de TP repose sur un serveur web installé sous Windows.

## Travail à faire :

1. Récupérer le zip et le décompresser dans le répertoire `www/symfony` de votre serveur web

`http://symfony.com/`



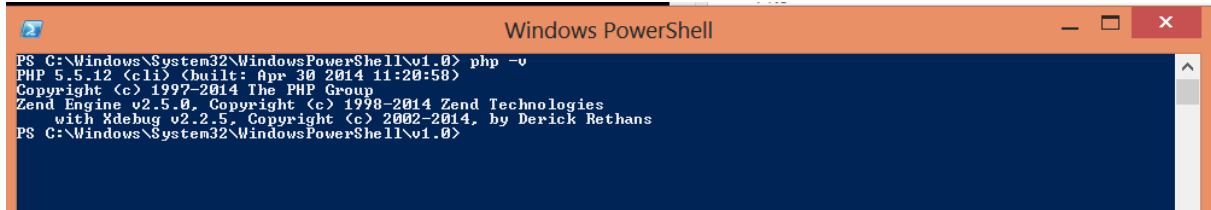
2. Après avoir démarré le serveur web, tester la configuration de Symfony sur `localhost/symfony/web/config.php`

Suivre les recommandations. Pour ma part, je n'active pas le cache lors du développement.

En cas de difficultés, pour activer opcache sous Wamp, remplacer la section [opcache] de `php.ini` par :

```
zend_extension="c:\ - votre chemin - \php\ext\php_opcache.dll"
opcache.enable=1
opcache.use_cwd=0
opcache.validate_timestamps=0 ; file stat
opcache.revalidate_freq=2 ; seconds
opcache.revalidate_path=0
opcache.save_comments=1
opcache.load_comments=1
opcache.dups_fix=0
opcache.inherited_hack=0
opcache.log_verbosity_level=1
opcache.memory_consumption=128
opcache.interned_strings_buffer=8
opcache.max_accelerated_files=4000
opcache.enable_file_override=1
opcache.max_wasted_percentage=5 ; percent waste until restart
opcache.consistency_checks=0
opcache.force_restart_timeout=180
opcache.blacklist_filename=paypal.php
opcache.fast_shutdown=1
opcache.optimization_level=0xffffffff
opcache.enable_slow_optimizations=1
opcache.max_file_size=0
```

### 3. Ajouter PHP au Path de Windows. Tester sous PowerShell la commande PHP -v

A screenshot of a Windows PowerShell window. The title bar is orange and says "Windows PowerShell". The background is dark blue. The text is white. It shows the command prompt at "PS C:\Windows\System32\WindowsPowerShell\v1.0>". The user has entered "php -v". The output shows "PHP 5.5.12 (cli) (built: Apr 30 2014 11:20:58) Copyright (c) 1997-2014 The PHP Group Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies with Xdebug v2.2.5, Copyright (c) 2002-2014, by Derick Rethans". The prompt returns to "PS C:\Windows\System32\WindowsPowerShell\v1.0>".

```
PS C:\Windows\System32\WindowsPowerShell\v1.0> php -v
PHP 5.5.12 (cli) (built: Apr 30 2014 11:20:58)
Copyright (c) 1997-2014 The PHP Group
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies
with Xdebug v2.2.5, Copyright (c) 2002-2014, by Derick Rethans
PS C:\Windows\System32\WindowsPowerShell\v1.0>
```

### 4. Commandes utiles :

*Vider le cache de Symfony :*

```
php app/console cache:clear --env=dev
```

*Raccourci Shell :*

Cette ligne permet de ne plus écrire php app/console

```
wamp\www\symfony>php app/console --shell
```