

Introduction

Symfony 5





Vetixy



Adrien MORIN

Ancien ESGI

CEO Vetixy
CEO Suchweb



> 3 500 000 000 DL

Octobre 2019

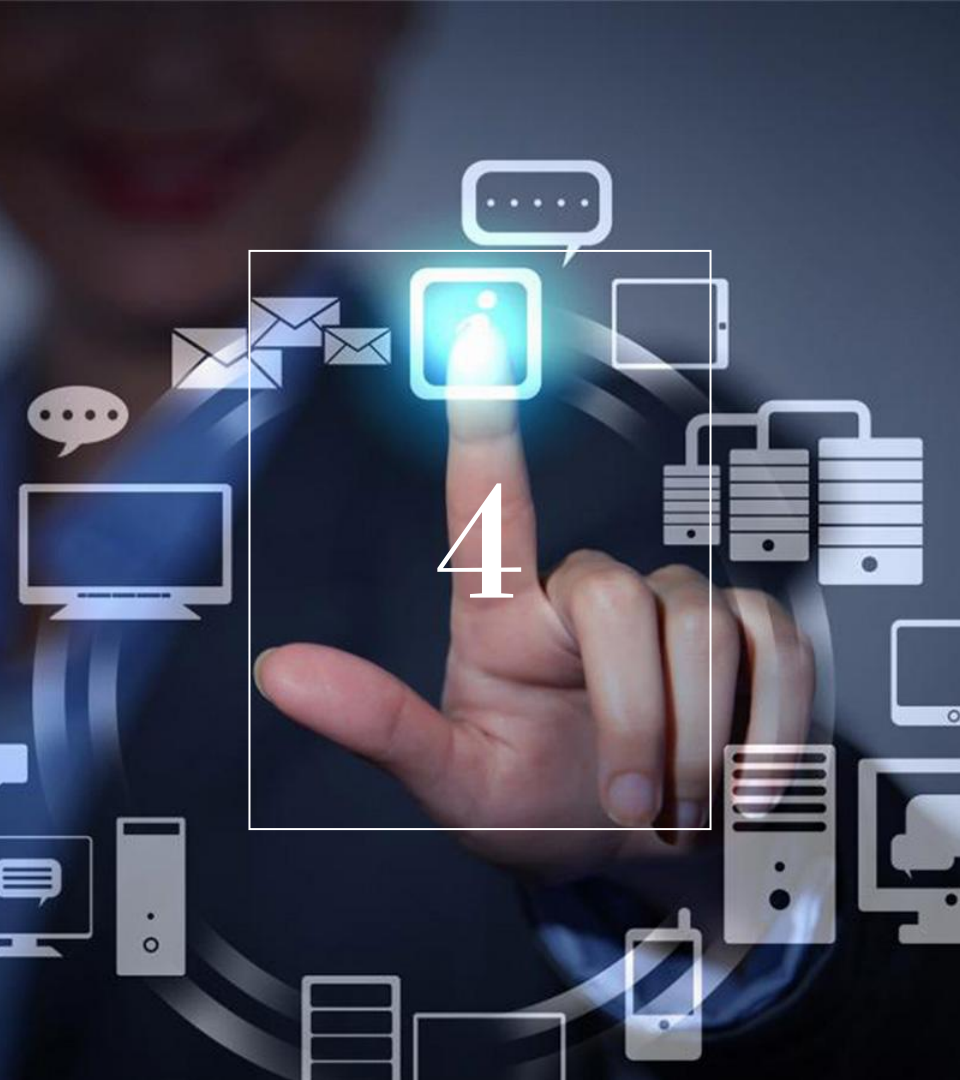
15 ans déjà

18 octobre 2005

> 2 200

Nombre de contributeurs Symfony

3



IDEOLOGIE

Framework

C'est quoi un Framework ?

Communauté

Une communauté digne de ce nom ! Puissante et à l'écoute !

La documentation

Votre nouveau livre pour vous endormir 😊

Best Practices

Une aide précieuse pour travailler en équipe : les best practices !

Liens utiles :

<https://symfony.com/doc/current/index.html>

https://symfony.com/doc/current/best_practices/index.html



CONCEPT

Commandes

Une aide précieuse aussi bien pour le développement que pour exécuter des actions serveur via php.

Flex

Processus d'installation simplifié de bundle.

Les bundles sont installés par des "recipes" (recette).

Les environnements

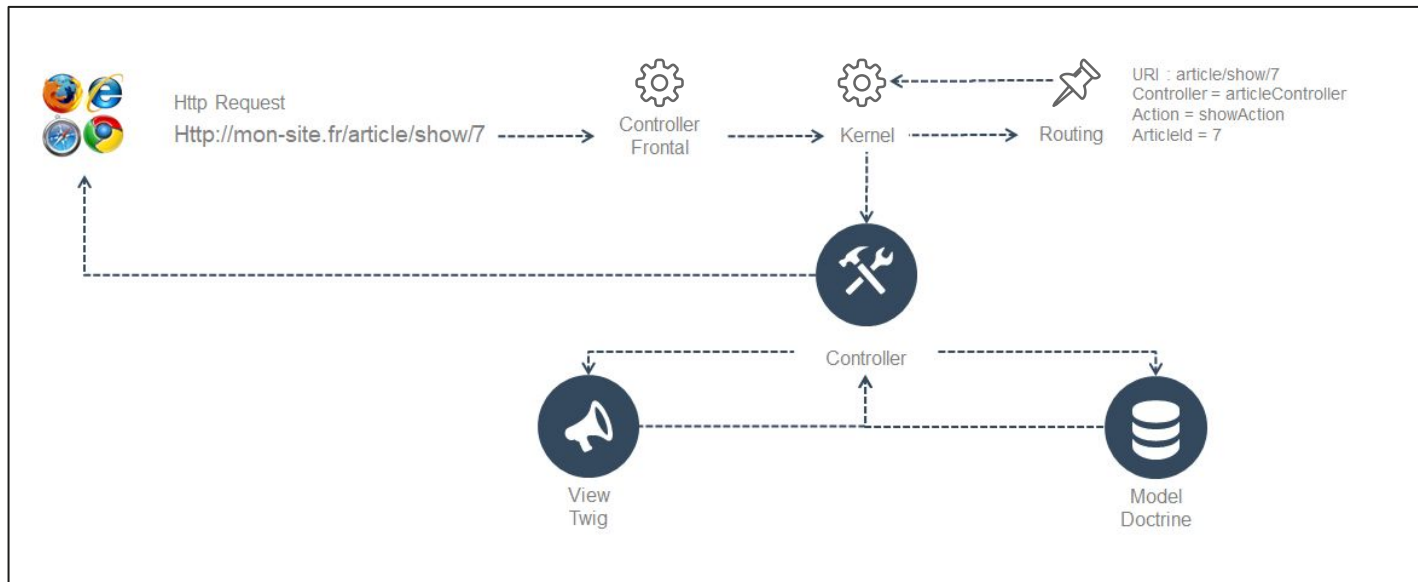
Un environnement de DEV pour travailler, avec des outils très pratiques.

Un environnement de PROD sécurisé pour la production.



Ne pas laisser l'environnement de DEV en PROD !!

CHEMIN D'UNE REQUÊTE HTTP DANS LE MODÈLE MVC DE SYMFONY





Composer

Composer est un **gestionnaire de dépendance** qui permet de définir différentes dépendances pour votre projet.

L'ensemble des packages disponibles via composer sont listés sur le site Packagist : <https://packagist.org/>

Composer & Symfony

Via composer, vous pouvez installer le "skeleton" de Symfony.

Pour ce faire, il suffit de taper la commande suivante :
composer create-project symfony/website-skeleton NomProjet

Composer utilisera alors le package présent sur github :
<https://github.com/symfony/website-skeleton>



Flex (<https://symfony.sh>)

Une **recette** (Recipe) est utilisée par le moteur Flex de Symfony pour installer **des bundles**.

Exemples :

- `recipes/symfony/console/3.3`
- `recipes/symfony/framework-bundle/3.3`

Utilisation : `composer require symfony/console`

Un **package** est utilisé par le moteur Flex de Symfony pour installer un **groupement de recette**.

Exemple :

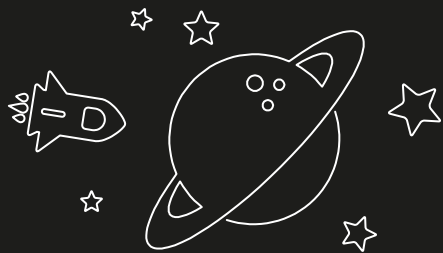
- `orm-pack`
- `api-pack`

Utilisation :

- **`composer require orm-pack`**
- **`composer unpack orm-pack | composer require orm-pack --unpack`**

Deux groupes de recettes / packages existent :

- Les officiels
- Les contribs (dites non officiel)
→ dans ce cas, il est nécessaire d'autoriser l'installation de ses recettes dans la configuration de composer sur le projet.



INSTALLATION

9



Quelques informations sur les fichiers importants

file.yaml

- Utiliser principalement pour l'écriture de fichiers de configurations ou de variables
- Permet de simplifier la saisie
- Convertit automatiquement la saisie en tableau array
- ATTENTION : La syntaxe est sensible en yaml

composer.json

- Liste l'ensemble des recettes / packages / bundles installés

symfony.lock

- Permet **l'historisation** de l'ensemble des recettes / packages / bundles installés.
- **Simplifie** la **mise à jours** des recettes / packages / bundles
- **Évite l'installation en double** des recettes / packages / bundles
- **Simplifie le déploiement**

bundles.php

- Liste l'ensemble des bundles installés

.env / .env.dist

- Liste l'ensemble des variables d'environnement (clé -> valeur)

Le système de route

Format de route disponible par Symfony : yml, annotation, php, xml

Route simple :

blog_index:
path: /blog/
defaults: { _controller: App:Blog:index }

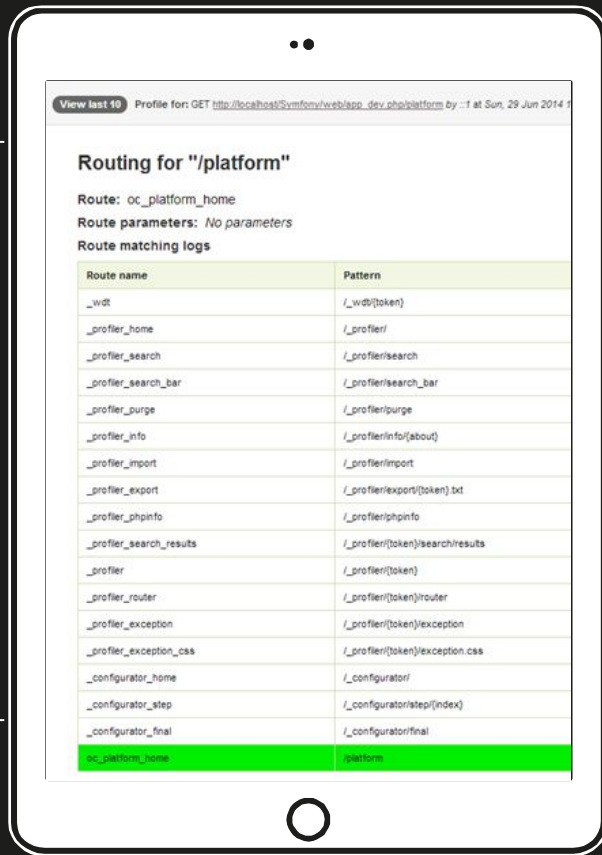
Route dynamique :

blog_show:
path: /blog/{slug}
defaults: { _controller: App:Blog:show }

Route avancés :

blog_show_format:
path: /blog/{slug}.{format}
defaults: { _controller: App:Blog:showFormat, format: html }
requirements:
slug: \w{15}
format: html|pdf

11





Des questions



12

CONTROLLER & TWIG

A vos machines !



13