

Symfony Introduction

AYMEN SELLAOUTI

C'est quoi symfony ?

« Symfony is a set of PHP Components, a Web Application framework, a Philosophy, and a Community — all working together in harmony. » [site Officiel Symfony]

- ✓ *Ensemble de composants PHP*
- ✓ *Framework pour les applications web*
 - ✓ *Basée sur des composants*
 - ✓ *Structuration du code*
 - ✓ *Maintenabilité*
- ✓ *Une philosophie*
 - ✓ *Les bonnes pratiques*
 - ✓ *standardisation*
- ✓ *Une très grande communauté*

FrameWork : Cadre De Travail (Boite à outils)

Ensemble de composants servant à créer :

- Fondation
- Architecture

Pourquoi utiliser un Framework

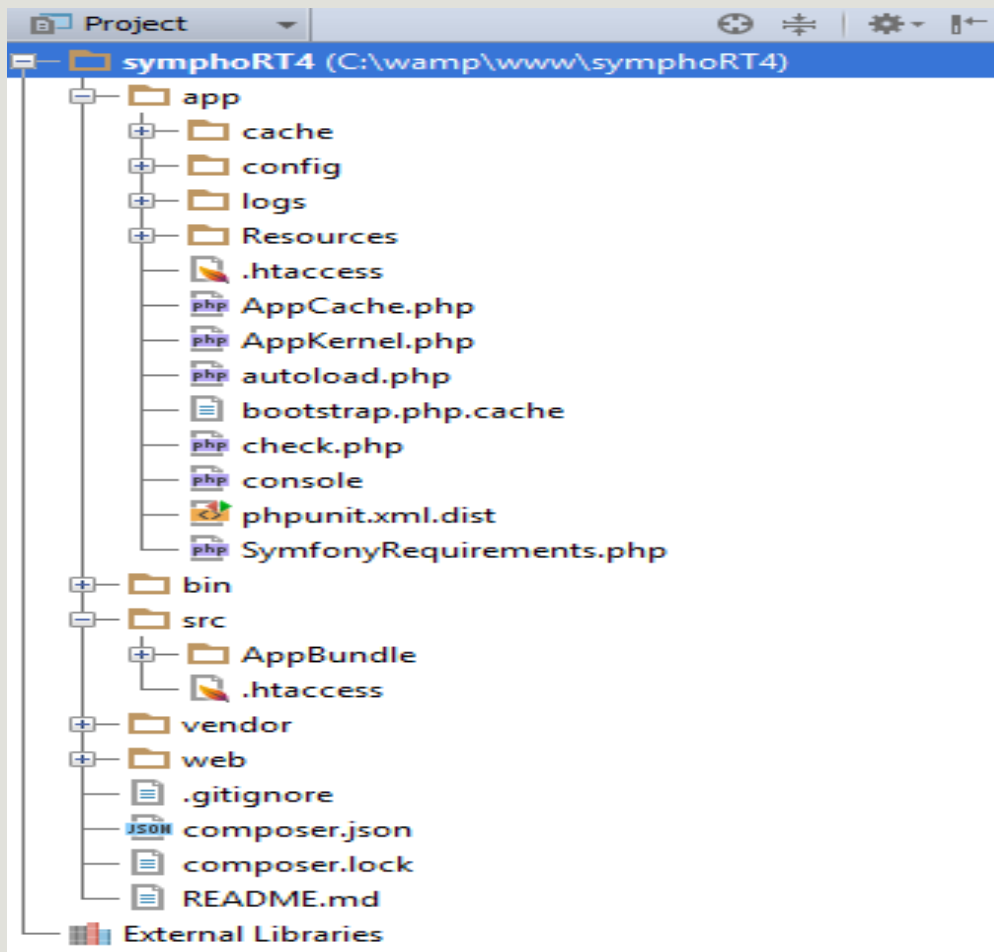
Productivité : ensemble de composants déjà prêt à l'emploi

Propreté du code => maintenabilité et évolutivité

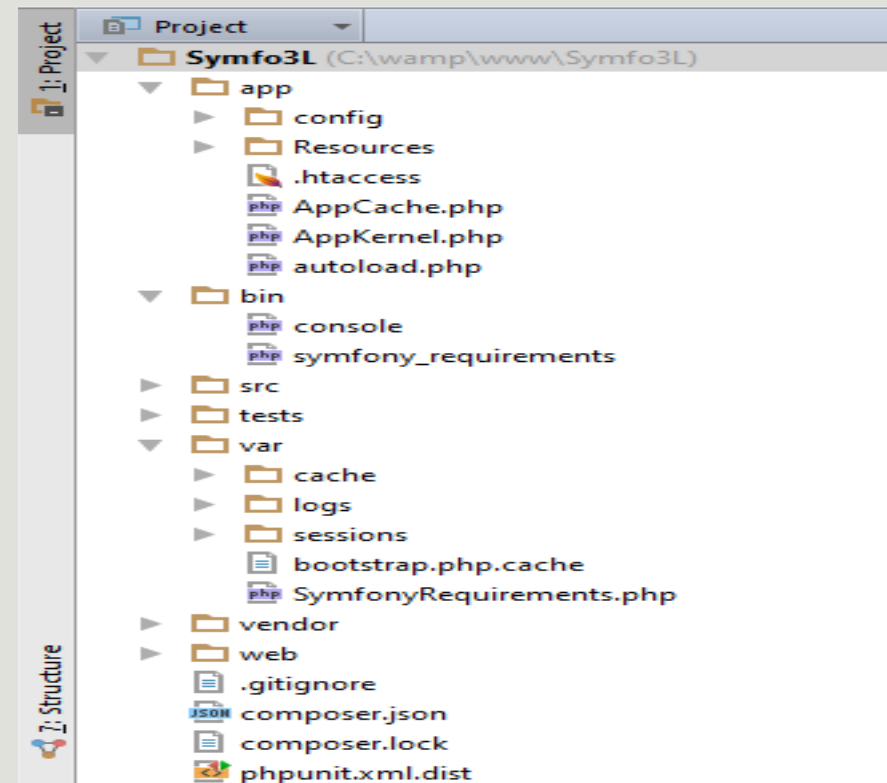
Basée sur une architecture => facilite le travail en équipe

Communauté et documentation

Architecture d'un projet Symfony



- **app/** : La configuration de l'application,
- **src/** : Le code PHP du projet,
- **vendor/** : Les bibliothèques tierces,
- **web/** : Le répertoire Web racine.



Symfony 3

Identifier les différences entre symfony2.8 et 3

Le contrôleur frontal (1)

Il joue le rôle de dispatcheur :

- Intercepte les requêtes
- Appelle le noyau de symfony (AppKernel.php)
- Le noyau prépare la réponse à rendre

Le contrôleur frontal (2)

Le contrôleur principal s'occupe de gérer les requêtes, mais cela signifie plus que simplement déterminer une action à exécuter. En fait, il exécute le code qui est commun à chaque action, soit les étapes suivantes :

Définir les constantes.

Déterminer les chemins des bibliothèques Symfony.

Charger et initialiser les classes du coeur du framework.

Charger la configuration.

Décoder la requête URL afin d'identifier les actions à effectuer et les paramètres de la requête.

Si l'action n'existe pas, faire une redirection sur l'action 404 error.

Activer les filtres (par exemple, si les requêtes nécessitent une authentification).

Exécuter les filtres une première fois.

Exécuter l'action et générer la présentation.

Exécuter les filtres une seconde fois.

Renvoyer la réponse.

Architecture MVC

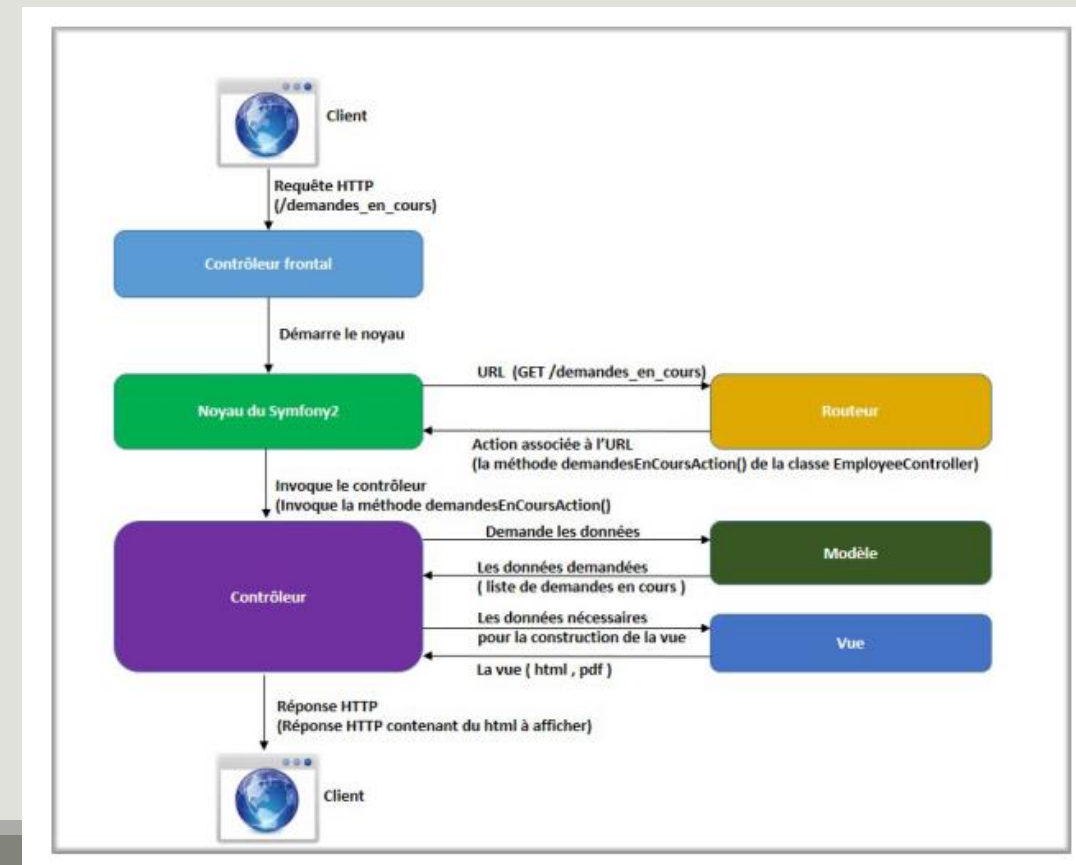
Symfony n'est pas une architecture MVC (Model View Controller).

SensioLab définit Symfony comme un HTTP

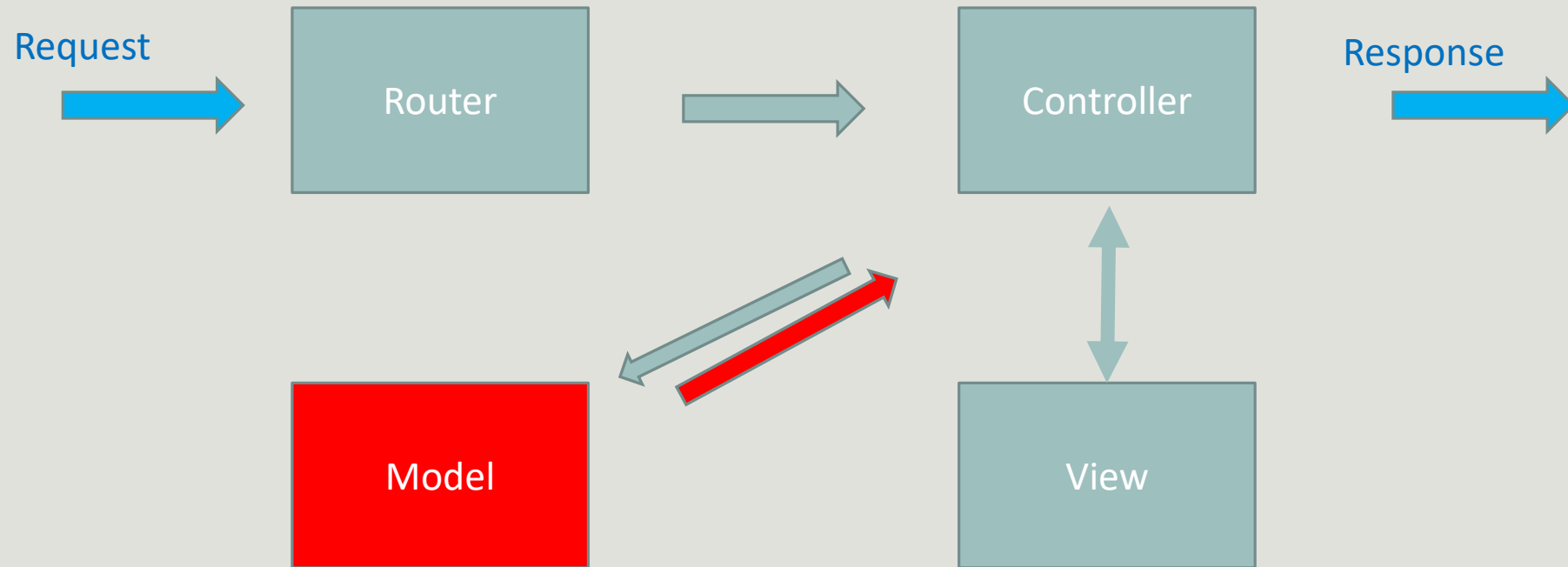
Framework basé sur l'échange HTTP Request/Response.

Cependant on peut l'implémenter.

La raison pour laquelle Symfony n'est pas défini
comme telle est que la couche Model n'est pas fournie

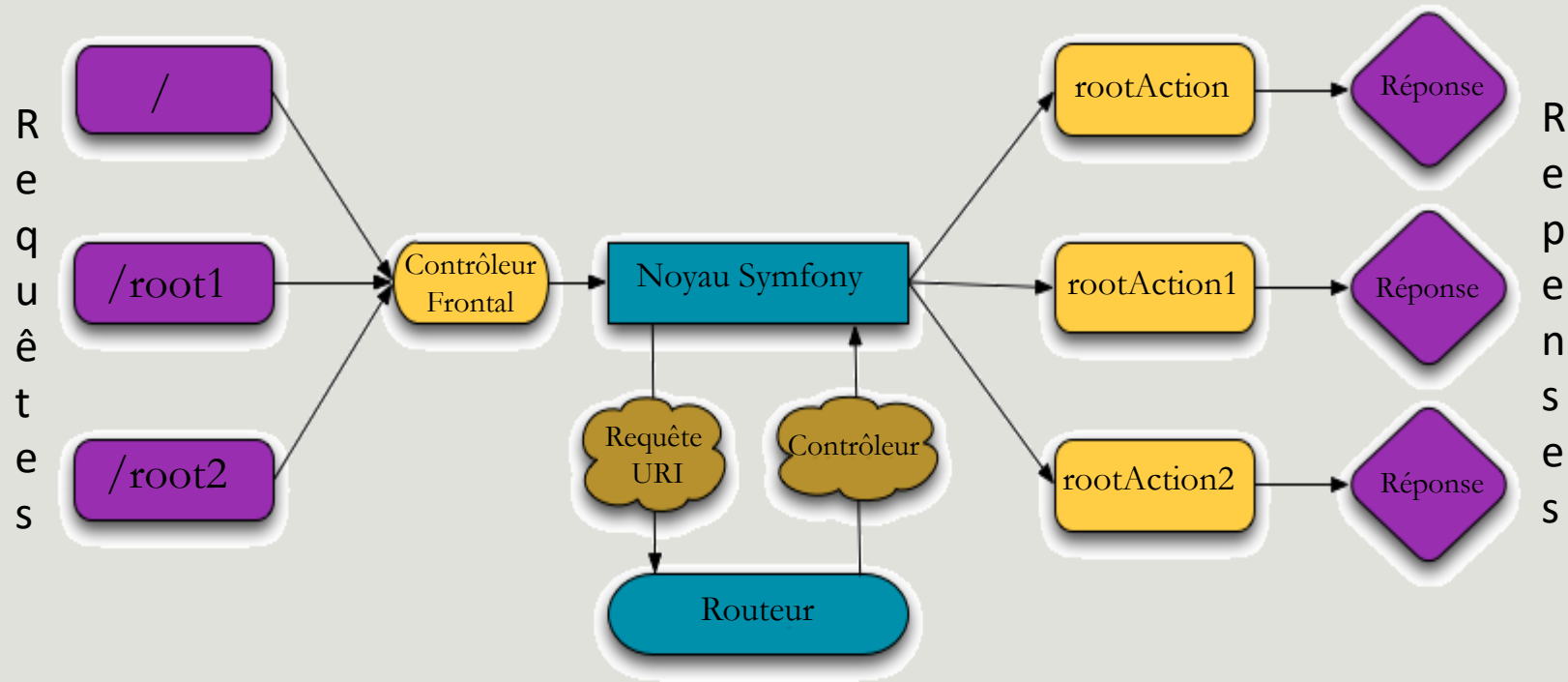


Architecture MVC



Cette partie est indépendante de Symfony vous pouvez utiliser ce que vous voulez Doctrine par exemple

Traitement d'une requête au sein de Symfony



La décomposition en Bundle (1)

Un bundle peut être défini comme étant un :

- Une brique logicielle,
- un module fonctionnel
- un morceau d'application.

Exemple : Site internet sur les Jeux Vidéos

Nous pouvons avoir les bundles suivants :

- Forum
- Newsletter
- Actualité...

La décomposition en Bundle (2)

Un Bundle peut être

- Indépendant et réutilisable
- Dépendant d'autres bundles

Un Bundle permet de découper l'application en fonctionnalités. Il a une architecture bien définie.

Un Bundle peut être utilisé dans plusieurs applications.

Structure d'un Bundle

Controller : C'est le dossier qui contient les contrôleurs du Bundle.

Entity : Dossier Qui contient les entités du Bundle

Form : Dossier contenant la définition des formulaires du Bundle

Test : Dossier pour les test unitaires du Bundle

Ressources : Dossier contenant les fichiers nécessaires à votre bundle à savoir :

- les fichiers de configurations dans le sous dossier **config**
- Les vues du Bundle dans le sous dossier **views**
- Les fichiers Css et Js dans le sous dossier **public**

...

Les Bundles les plus connus de la communauté

[FOSUserBundle](#) : Gestion des utilisateurs

[SonataAdminBundle](#) : Espace d'administration

[KnpMenuBundle](#) : Gestion des menus

[KnpPaginatorBundle](#) : Pagination

<http://knpbundles.com/> ; <http://symfony.com/blog/the-30-most-useful-symfony-bundles-and-making-them-even-better>

<https://packagist.org>

Installation de Symfony (1)

Il existe [plusieurs](http://symfony.com/fr/doc/current/book/installation.html) méthodes afin d'installer symfony2
(<http://symfony.com/fr/doc/current/book/installation.html>)

Pré requis :

Serveur Web contenant PHP (Apache).

Pour vérifier la bonne mise en place des pré requis utiliser la commande suivante :

```
php app/check.php
```

<http://symfony.com/doc/current/reference/requirements.html>;

Installation de Symfony (2)

Méthode 1

Télécharger la version souhaitée (<http://symfony.com/download>)

Méthode 2

Télécharger et installer composer (<https://getcomposer.org/download/>)

Ouvrir la ligne de commande et placer vous sur le dossier d'installation

Tapez la commande suivante : **composer create-project symfony/framework-standard-edition path/ "3.4.*"**

Méthode 3

Installer le symfony installer avec la commande suivante :

c:\> php -r "readfile('http://symfony.com/installer');" > symfony

Copier le fichier dans le dossier père de votre dossier d'installation

Taper la commande suivante : **php symfony new nomProjet**

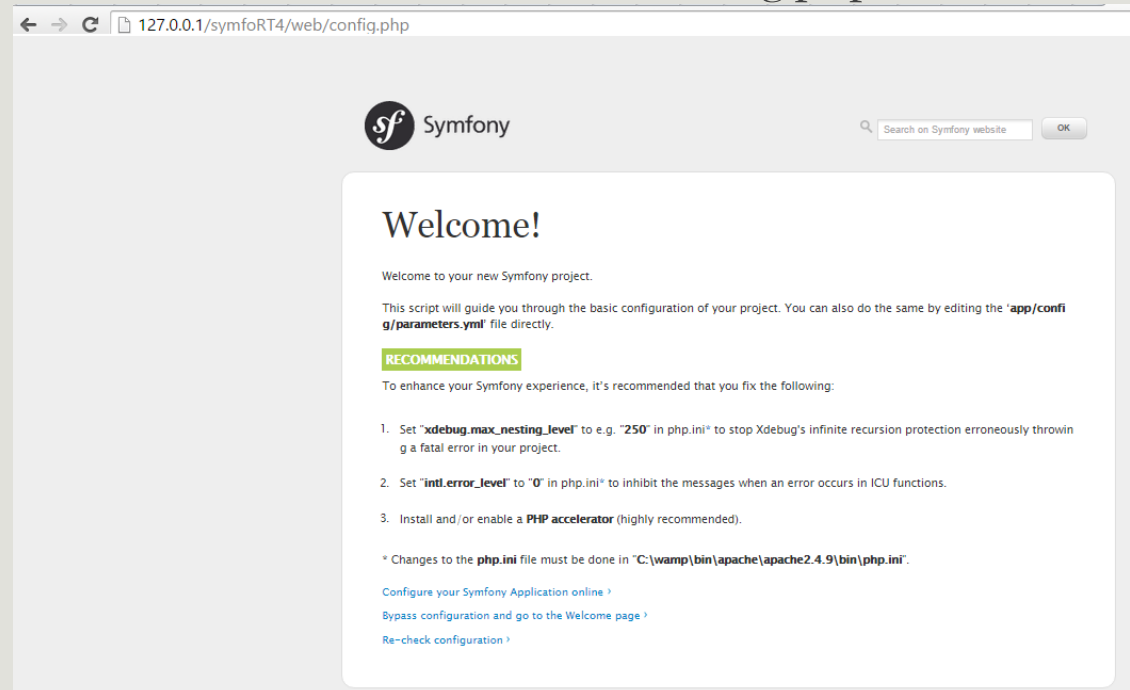
Pour voir les différentes options tapez php symfony et vous aurez le help de l'installer.

Configuration de votre projet

Afin de configurer votre projet il y a deux solutions :

Solution 1 : `app/config/parameters.yml`

Solution 2 : `http://127.0.0.1/nomDossier/web/config.php`



Génération du premier Bundle

2 Méthodes peuvent générer un Bundle

- Méthode manuelle (non recommandée)
- Méthode automatique avec la console

commande : **php bin/console generate:bundle**

Propriétés de la commande :

namespace : Espace de nom il peut avoir n'importe quel nom qui se termine par Bundle les bonnes pratiques nous apprennent qu'il est préférable de le décomposer en trois parties

- Namespace racine qui représente généralement l'organisme (e.g. nom d'un particulier, d'une entreprise)
- Nom du Bundle
- Le suffixe Bundle

Exemple : MyVendorName/TestBundle

Génération du premier Bundle (2)

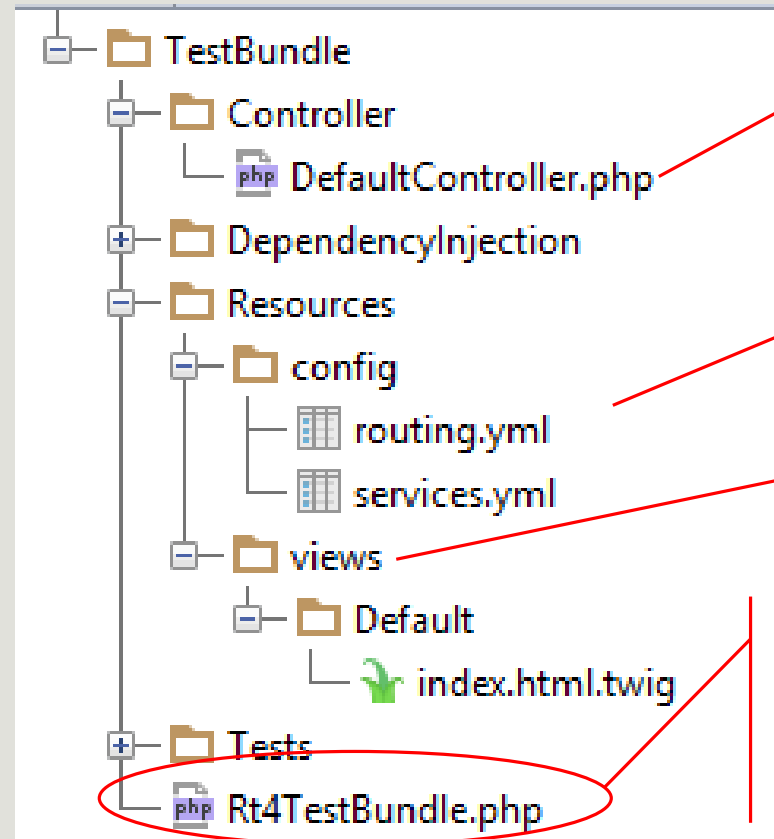
Propriétés de la commande :

Nom du bundle : Le nom est automatiquement suggéré en utilisant le namespace sans le /

Destination : le répertoire de destination par défaut c'est un dossier portant le nom du bundle sous le répertoire source

Format de configuration : Représente la forme avec laquelle Symfony va configurer le Bundle. Les plus utilisées sont YML et Annotation. Nous utilisons comme départ YML.

Génération du premier Bundle (3)



Les contrôleurs du bundle

Les routes du bundle

Les vues du bundle

Le seul fichier obligatoire pour un bundle il est donc nécessaire de le créer lors d'une création manuelle d'un bundle. Il va juste hériter de la classe bundle

Enregistrement du Bundle

Afin de charger les bundles d'un projet, Symfony enregistre ces derniers dans son noyau.


Le fichier AppKernel se trouve dans le dossier app

L'enregistrement du Bundle se fait automatiquement avec la console.

Pour la création manuelle du Bundle il faut l'ajouter manuellement dans le tableau de bundle de la méthode registerBundles() du appKernel

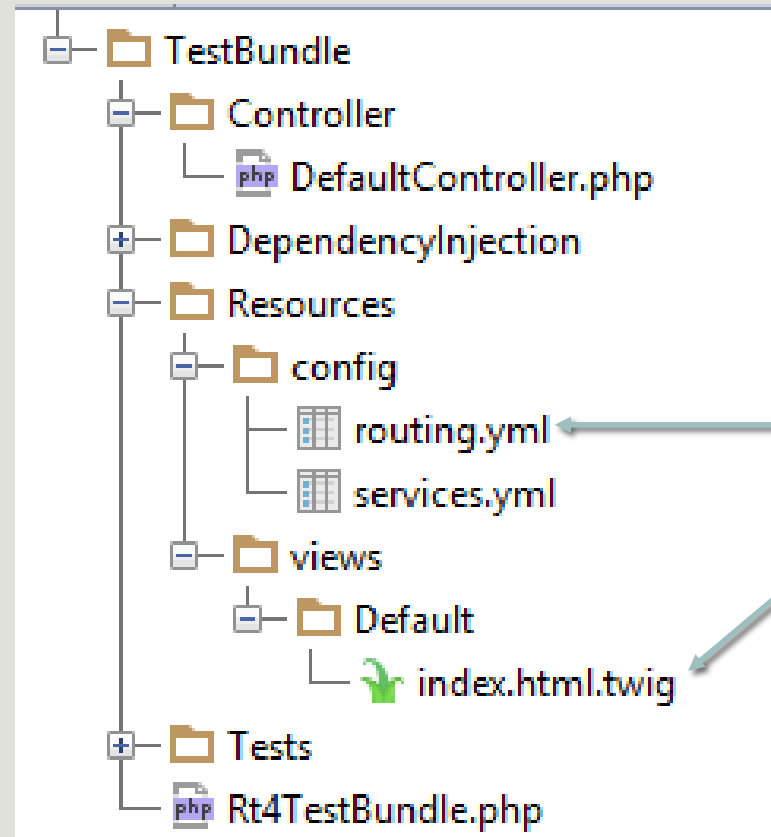
Ajout du bundle test

```
class AppKernel extends Kernel
{
    public function registerBundles()
    {
        $bundles = array(
            new Symfony\Bundle\FrameworkBundle\FrameworkBundle(),
            new Symfony\Bundle\SecurityBundle\SecurityBundle(),
            new Symfony\Bundle\TwigBundle\TwigBundle(),
            new Symfony\Bundle\MonologBundle\MonologBundle(),
            new Symfony\Bundle\SwiftmailerBundle\SwiftmailerBundle(),
            new Symfony\Bundle\AsseticBundle\AsseticBundle(),
            new Doctrine\Bundle\DoctrineBundle\DoctrineBundle(),
            new Sensio\Bundle\FrameworkExtraBundle\SensioFrameworkExtraBundle(),
            new AppBundle\AppBundle(),
            new Rt4\TestBundle\Rt4TestBundle(),
            new Rt4\AsBundle\Rt4AsBundle(),
        );
    }
}
```



Enregistrement des roots du Bundle (1)

Afin d'accéder aux différentes vues d'un bundle nous devons enregistrer les roots qui y correspondent.

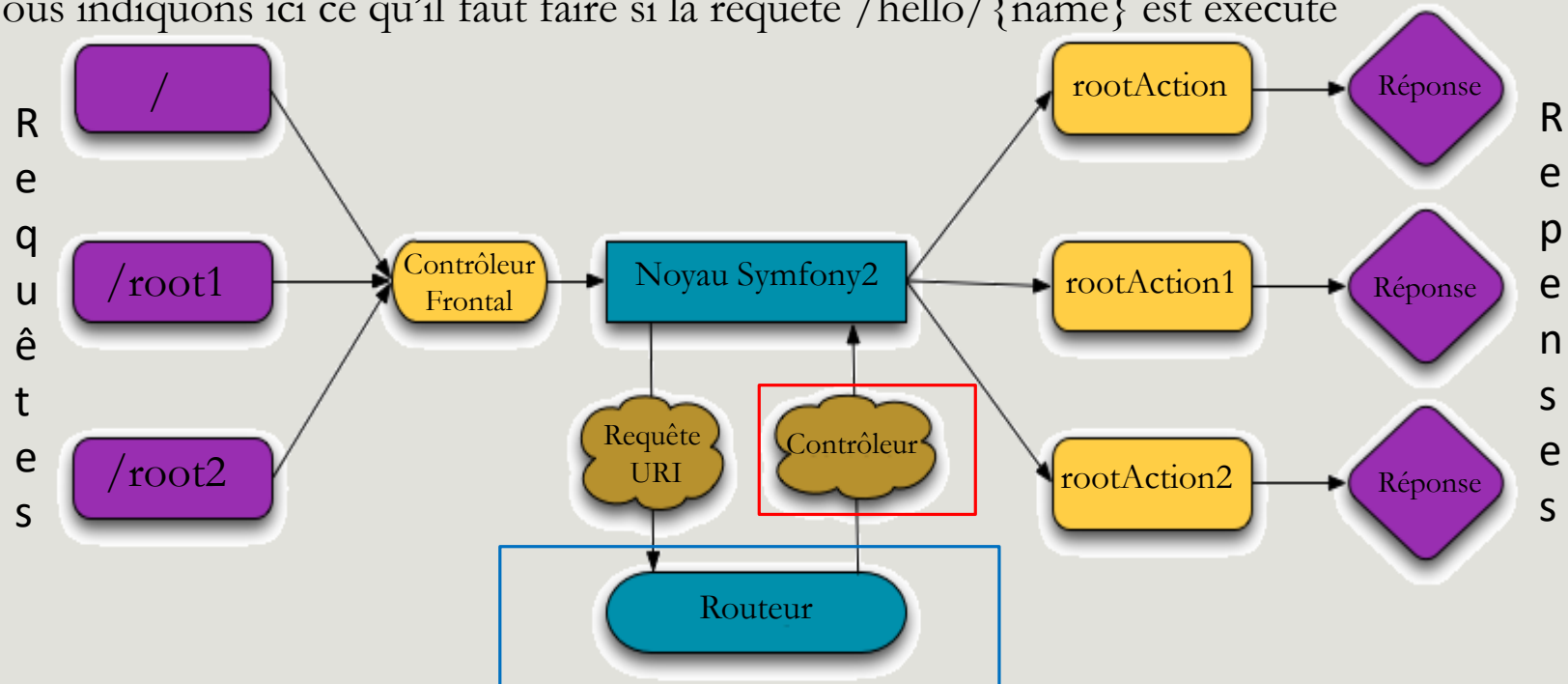


Pour accéder à l'index.html.twig, il faut définir la root dans le fichier routing.yml

Enregistrement des roots du Bundle (2)

```
Rt4_test_homepage:  
  path:      /hello/{name}  
  defaults: { _controller: Rt4TestBundle:Default:index }
```

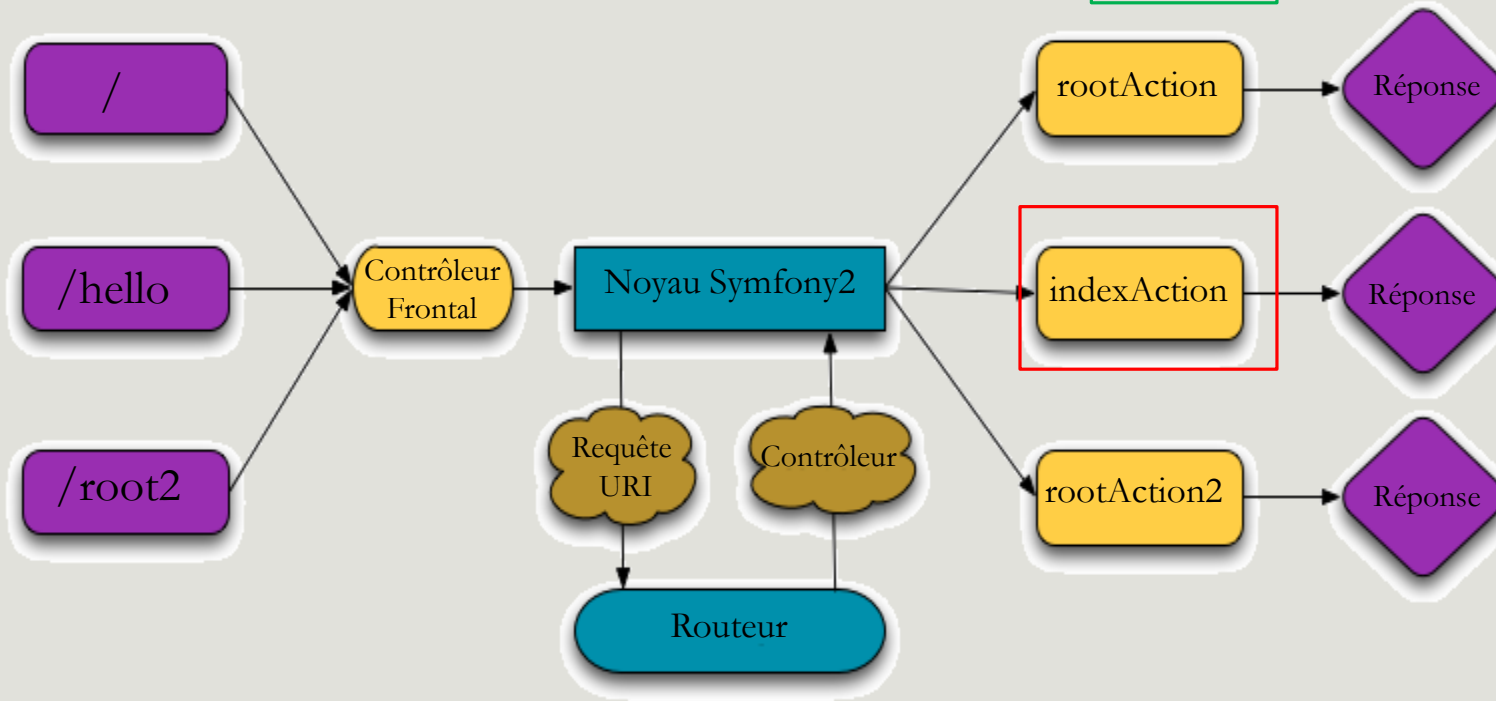
Nous indiquons ici ce qu'il faut faire si la requête `/hello/{name}` est exécuté



Enregistrement des roots du Bundle (3)

```
rt4_test_homepage:  
  path:      /hello/{name}  
  defaults: { _controller: Rt4TestBundle:Default:index }
```

R
e
q
u
ê
t
e
s



```
app\...routing.yml x  Rt4TestBundle.php x  Resources\...routing.yml x  Controller\DefaultController.php x  
\\Rt4TestBundle\\Controller\\DefaultController\\indexAction  
<?php  
namespace Rt4\\TestBundle\\Controller;  
  
use Symfony\\Bundle\\FrameworkBundle\\Controller\\Controller;  
  
class DefaultController extends Controller  
{  
    public function indexAction($name)  
    {  
        return $this->render('Rt4TestBundle:Default:index.html.twig', array('name' => $name));  
    }  
}
```