Zend Framework : Premiers pas avec Zend_Application

par AIT YAHIA Iidir (Mon site perso) (Blog)

Date de publication : 18 juillet 2009

Dernière mise à jour :

Avec l'avènement de la version 1.8 de Zend_Framework, la configuration du bootstraping peut paraitre à première vue déroutante pour les novices et même pour les confirmés qui ont l'habitude de travailler avec les versions précédentes du Framework. Dans ce tutoriel nous allons essayer de vous montrer comment configurer une application en MVC Avec Zend_Application. Vous allez voir que c'est claire et plus simple que ce que nous faisions d'habitude.

I - Introduction....3
I-A - Pré-requis....3
I-B - Présentation....3
I-C - Pourquoi utiliser Zend_Application ?....3
II - Mise en uvre d'une application basique....3
II-A - Création de la structure de votre application....3
II-B - Configuration de notre application....4
II-B-1 - Fichier de configuration....4
II-B-2 - Index.php....6
II-B-3 - Fichier .htaccess....7
II-B-4 - Bootstrap....7
III - Mise en uvre d'une application modulaire....8
III-A - Fichier de configuration....9
III-B - Bootstrap....10
III-C - les Controlleurs d'actions....11
IV - Ajouter une ressource....12
V - Conclusion....14

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

I - Introduction

I-A - Pré-requis

Cet article vise un publique ayant de très bonnes connaissances en PHP, le concept MVC et ayant déjà utilisé Zend Framework auparavant.
Ce tutoriel a été réalisé avec la version 1.8.4 du Framework et ne fonctionnera avec aucune des versions antérieures à la version 1.8.

I-B - Présentation

Zend_ Application est l'un des nouveaux composants introduit dans la version 1.8 de Zend Framework, il nous permet de faciliter et de standardiser la mise en place du bootstraping ainsi que la configuration de l'environnement PHP.

Avec Zend_Application l'autoloading est mis de l'avant et le bootstrap se présente sous une forme modulaire ou sont initialisées toutes les ressources réutilisables telles que les vues, les connexions aux bases de données, les layout...etc.

l'autoloading plus en détail **Atelier Zend Framework : Autochargement de classes et de composants**

I-C - Pourquoi utiliser Zend_Application ?

Une application MVC peut très rapidement devenir complexe et faire intervenir énormément d'objets, chacun est configurable d'une manière qui lui est propre, chacun est dépendant éventuellement d'autres objets .... Un casse tête est alors apparu : comment créer, configurer, lier et mémoriser tous ces objets de manière pérenne, et propre ?
Zend_Application répond à ces problématiques en proposant une manière simple et efficace de configurer tout l'environnement MVC

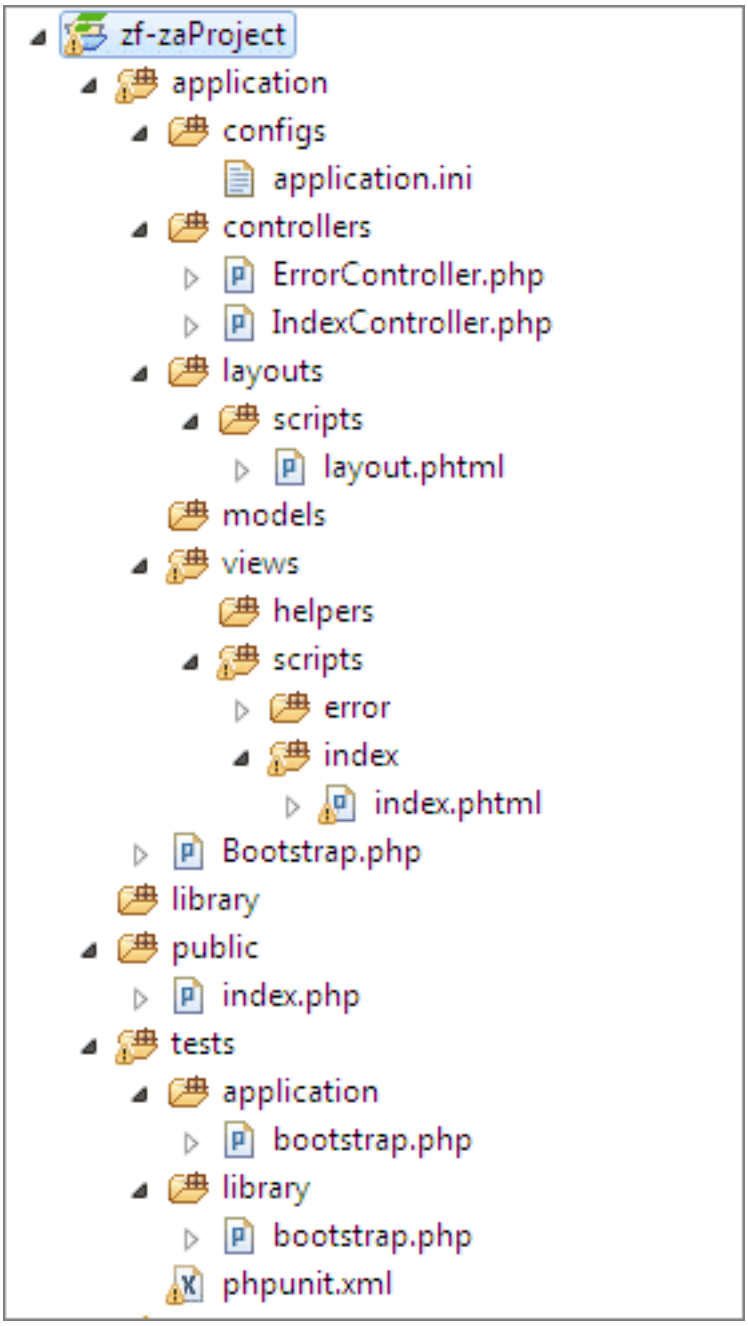
II - Mise en uvre d'une application basique

II-A - Création de la structure de votre application

Pour commencer nous devrions créer une nouvelle application Zend Framework avec la structure conventionnelle semblable à celle présenté dans l'image ci-dessous. Nous pouvons faire cela manuellement, mais il est plus judicieux d'utiliser Zend_Tool qui va faire le plus gros du travail à votre place.
Pour de plus amples détails sur la création d'applications avec Zend_Tool vous pouvez consulter cet article.

Créer une application basée sur Zend Framework avec Zend_Tool

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.



Le dossier public contient aussi le fichier htaccess qui n'apparait pas dans l'arborescence.

II-B - Configuration de notre application

Nous allons maintenant nous intéresser aux fichiers importants pour notre configuration.

II-B-1 - Fichier de configuration

Le fichier auquel nous devons accorder une attention et particulière est sans doute notre fichier de configuration application.ini, c'est dans ce fichier que seront initialisés toutes les ressources et l'environnement PHP.
Remarque que vous avons plusieurs sections dans notre **application.ini**, celles-ci nous permettent d'avoir des configurations distinctes pour chaque mode d'exécution de notre application (production, développement, test...etc.).

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/configs/application.ini

```
[production]
# initilisation du report d'erreurs pour le mode production
phpSettings.display_startup_errors = 0
phpSettings.display_errors = 0

#definition de la timezone
phpsettings.date.timezone = "Africa/Algiers"

# Include path
includePaths.library = APPLICATION_PATH "/../library"

# Bootstrap
bootstrap.path = APPLICATION_PATH "/Bootstrap.php"
bootstrap.class = "Bootstrap"

# Front controller
 resources.frontController.controllerDirectory = APPLICATION_PATH "/controllers"

# Layout
resources.layout.layout = "layout"
resources.layout.layoutPath = APPLICATION_PATH "/layouts/scripts"

# Views
resources.view.encoding = "UTF-8"

# connexion à une base de données
resources.db.adapter = "pdo_mysql"
resources.db.params.host = "localhost"
resources.db.params.username = "aityahia"
resources.db.params.password = "monpasse"
resources.db.params.dbname = "zf-zaProject"
resources.db.isDefaultTableAdapter = true


[staging : production]


[testing : production]
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1

[development : production]
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1

# base de données test
resources.db.params.dbname = "zf-zaProject-test"
```

Remarque que vous avons plusieurs sections dans notre **application.ini**, celles-ci nous permettent d'avoir des configurations distinctes pour chaque mode d'exécution de notre application (production, développement, test...etc.). Dans notre fichier de configuration, tout ce qui vient après le mot ressources est considéré en tant que telle.
L'option phpSettings nous permet de définir les directives du **php.ini** Lorsque cette option est définie Zend_Application fait appel à la fonction php **ini_set** , la liste de toutes les directives définissables est disponible sur le site officiel de php **Liste des directives du php.ini**
La définition des directives se fait de la manière suivante :

```
phpSettings.nom_directive = valeur
```

Comme vous pouvez le voir nous avons pratiquement tout initialisé dans notre fichier de configuration, nous avons même défini une base de données spécifique pour les essais en cours de développement.

Il nous est tout à fait possible d'utiliser un autre format de fichier de configuration .xml, .inc, .php des exemples sont disponibles dans la documentation officiel. La piste du fichier XML est à écarter si vous utilisez dans votre configuration

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

*des constantes définies avec la méthode **define** (comme c'est dans notre cas avec la constante APPLICATION_PATH), pour cause, Zend_Application utilise en interne Zend_Config_Xml pour la lecture du XML, hors que celui-ci ne prend pas en charge la lecture des constantes d'environnement. Ce problème est valable du moins pour le moment car d'après mes lectures cette restriction sera levée dans la version 1.9 du Framework.*

II-B-2 - Index.php

Dans le fichier index.php nous allons définir les chemins et les constantes d'exécutions dont nous avons besoin, invoquer Zend_Application et en dérnier lieu nous lançons l'application.

public/index.php

```
<?php

// Défini le chemin du repértoire de l'application
defined('APPLICATION_PATH')
    || define('APPLICATION_PATH', realpath(dirname(__FILE__) . '/../application'));

// Défini l'environnement de l'application
defined('APPLICATION_ENV')
    || define('APPLICATION_ENV',
 (getenv('APPLICATION_ENV') ? getenv('APPLICATION_ENV') : 'production'));

// garantir que library/ est bien dans l'include path
set_include_path(implode(PATH_SEPARATOR, array(
    realpath(APPLICATION_PATH . '/../library'),
    get_include_path(),
)));

/** Zend_Application */
require_once 'Zend/Application.php';

// Creation de l'application, bootstrap et exécution
$application = new Zend_Application(
    APPLICATION_ENV,
    APPLICATION_PATH . '/configs/application.ini'
);
$application->bootstrap()
            ->run();
```

Analysons du plus près l'instanciation de Zend_Application

```
// Creation de l'application, bootstrap et exécution
$application = new Zend_Application(
    APPLICATION_ENV,
    APPLICATION_PATH . '/configs/application.ini'
);
$application->bootstrap()
            ->run();
```

Dans cette partie de code nous invoquons Zend_Application en lui indiquant qu'il doit tenir compte de la configuration présente dans notre fichier de configuration, mais il vous est tout à fait possible de lui passer en paramètre un tableau définissant les options de configuration, comme on peut le voir dans cet l'exemple.

```
$application = new Zend_Application(APPLICATION_ENV, array(
    'pluginPaths' => array(
        'Application_Resources' => APPLICATION_PATH . '/resources/',
    ),
    'resources' => array(
        'FrontController' => array(
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

```
            'controllerDirectory' => APPLICATION_PATH . '/controllers',
        ),
    ),
));
```

Il est fortement recommandé d'utilisé le fichier de configuration pour ne pas d'encombrer l'instanciation de votre application qui risque de devenir incompréhensible.
Il est à noter que dans les deux cas Zend_Application récupère la configuration sous forme de tableau (Zend_Config_Ini est utilisé en interne pour la lecture de votre fichier configuration).

II-B-3 - Fichier .htaccess

Notons bien que nous avons fait appel à la variable d'environnement **APPLICATION_ENV**, celle-ci peut être définie dans notre fichier htaccess ou dans la configuration apache, c'est elle qui défini le mode d'exécution de votre application.

public/.htaccess

```
SetEnv APPLICATION_ENV development

RewriteEngine On
RewriteCond %{REQUEST_FILENAME} -s [OR]
RewriteCond %{REQUEST_FILENAME} -l [OR]
RewriteCond %{REQUEST_FILENAME} -d
RewriteRule ^.*$ - [NC,L]
RewriteRule ^.*$ index.php [NC,L]
```

II-B-4 - Bootstrap

Dans notre Bootstrap nous allons initialiser la vue de notre Layout, dans cet exemple nous avons utilisé Zend_View pour sa définition, mais il nous est possible de nous en passer.
Nous allons aussi initialiser notre adaptateur de base de données qui sera utilisable dans tous nos modèles et contrôleurs

application/bootstrap.php

```
class Bootstrap extends Zend_Application_Bootstrap_Bootstrap {

    //initilisation de l'autoloader
 protected function _initAutoload() {
  $moduleLoader = new Zend_Application_Module_Autoloader (
  array ('namespace' => '', 'basePath' => APPLICATION_PATH ) );
  return $moduleLoader;
 }

 protected function _initView() {
  // Initialisation de la vue
  $view = new Zend_View ( );
  $view->doctype ( 'XHTML1_STRICT' );
  $view->headTitle ( 'Mon application avec Zend_Application' );

  // Add it to the ViewRenderer
  $viewRenderer = Zend_Controller_Action_HelperBroker::getStaticHelper ( 'ViewRenderer' );
  $viewRenderer->setView ( $view );

  // Return it, so that it can be stored by the bootstrap
  return $view;
 }

 protected function _initDb() {
  $db = $this->getPluginResource ( 'db' )->getDbAdapter ();
  $db->setFetchMode ( Zend_Db::FETCH_OBJ );
  Zend_Db_Table_Abstract::setDefaultAdapter ( $db );
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/bootstrap.php

```
    return $db;
  }

 }
```

soit notre fichier layout.phtml

application/layouts/scripts/layout.phtml

```
<?php echo $this->doctype() ?>
<html>
<head>
    <?php
echo $this->headTitle();
      echo $this->headLink();
      echo $this->headStyle();
      echo $this->headScript();
?>
</head>
<body>
    <?php echo $this->layout()->content ?>
</body>
</html>
```

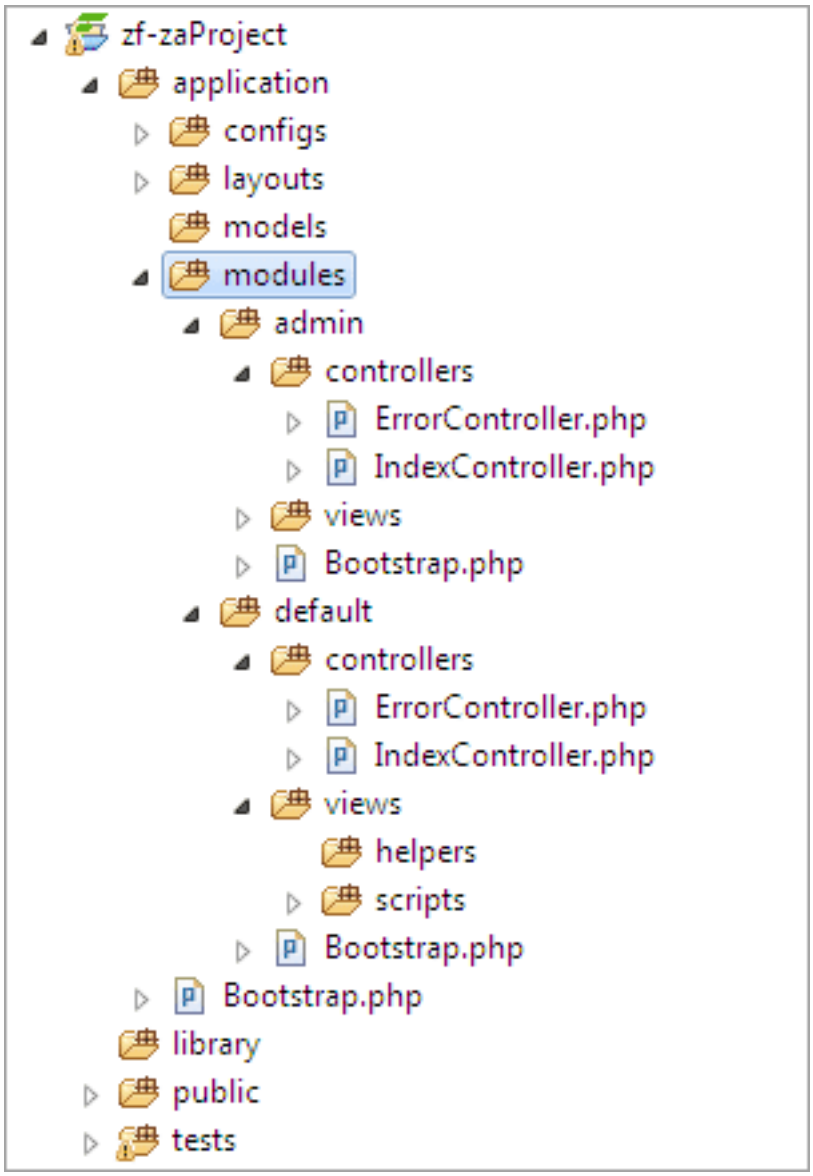
Nous avons terminé avec la configuration de notre application et elle est dès à présent fonctionelle et nous pouvons nous attaquer au développement de nos contrôleurs,modèles et vues...etc.

III - Mise en uvre d'une application modulaire

Pour la mise en uvre d'une application modulaire, la configuration se corse un petit peu, mais soyez sans crainte pas au point de vous étourdir, c'est pour cet effet que nous allons étudier ce cas en particulier.

En premier lieu, nous allons remanier notre structure en ajoutant deux modules (default et admin). Dans notre exemple le Layout et les modèles sont commun à nos modules, notre nouvelle structure devra ressembler à ceci.

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.



III-A - Fichier de configuration

Pour notre fichier de configuration, nous allons apporter quelques modifications au contrôleur frontal (Front Controller).

```
# Front controller
resources.frontController.baseUrl = "/zf-zaProject/public"
resources.frontController.controllerDirectory.default = APPLICATION_PATH "/modules/default/controllers"
resources.frontController.controllerDirectory.admin   = APPLICATION_PATH "/modules/admin/controllers"


#initialisation des modules
resources.frontController.moduleDirectory       = APPLICATION_PATH "/modules"
resources.frontController.defaultModule         = "default"
resources.modules = ""
```

Après modification notre fichier de configuration tel qu'il devrait être.

application/configs/application.ini

```
[production]

# initilisation du report d'erreurs pour le mode production
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/configs/application.ini

```
phpSettings.display_startup_errors = 0
phpSettings.display_errors = 0

#definition de la timezone
phpsettings.date.timezone = "Africa/Algiers"


# Include path
includePaths.library = APPLICATION_PATH "/../library"

# Bootstrap
bootstrap.path = APPLICATION_PATH "/Bootstrap.php"
bootstrap.class = "Bootstrap"


# Front controller
resources.frontController.baseUrl = "/zf-zaProject/public"
resources.frontController.controllerDirectory.default = APPLICATION_PATH "/modules/default/controllers"
resources.frontController.controllerDirectory.admin   = APPLICATION_PATH "/modules/admin/controllers"


#initialisation des modules
resources.frontController.moduleDirectory       = APPLICATION_PATH "/modules"
resources.frontController.defaultModule         = "default"
resources.modules = ""

# Layout
resources.layout.layout = "layout"
resources.layout.layoutPath = APPLICATION_PATH "/layouts/scripts"

# Views
resources.view.encoding = "UTF-8"


# connexion à une base de données
resources.db.adapter = "pdo_mysql"
resources.db.params.host = "localhost"
resources.db.params.username = "aityahia"
resources.db.params.password = "monpasse"
resources.db.params.dbname = "zf-zaProject"
resources.db.isDefaultTableAdapter = true


[staging : production]


[testing : production]
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1

[development : production]
# initilisation du report d'erreurs pour le mode developpement
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1
#base de données test

resources.db.params.dbname = "zf-zaProject-test"
```

III-B - Bootstrap

Le Bootstrap de notre application reste inchangé par contre nous devrons définir les classes des Bootstrap de nos modules, celles-ci étendent la classe Zend_Application_Module_Bootstrap, notez bien la différence avec celle de notre application qui étend Zend_Application_Bootstrap_Bootstrap.

application/modules/default/bootstrap.php

```
class Bootstrap extends Zend_Application_Module_Bootstrap{
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/modules/default/bootstrap.php

```
}
```

A l'exception de la classe Bootstrap du module par défaut toutes les autres devront respecter une certaine règle de nommage (**Nommodule_** Bootstrap) dans notre cas **Admin_**Bootstrap. Cette règle peut être appliquée sans effet de la classe Bootstrap du module par défaut.

application/modules/admin/bootstrap.php

```
class Admin_Bootstrap extends Zend_Application_Module_Bootstrap {

}
```

Il peut paraitre évident, que si nous initialisons une ressource au niveau du Bootstrap de l'un de nos modules, celle-ci est faite uniquement quand le module en question est invoqué; Détrompé vous car ce n'est pas le cas, les méthodes et plugins y afférent sont exécuté bien avant que le contrôleur frontal n'expédie la requête.

III-C - les Controlleurs d'actions

Pour pouvoir tester que notre application modulaire fonctionne, nous allons définir les Contrôleurs d'actions index de nos modules.

application/modules/default/IndexController.php

```
class IndexController extends Zend_Controller_Action
{

    public function init()
    {
       $this->view->headTitle ( 'Mon application avec Zend_Application (Utilsateur par défaut)' );
    }

    public function indexAction()
    {
     $this->_response->appendBody('Bievenue');

    }


}
```

Comme pour les classes Bootstrap des modules, les contrôleurs d'actions obéissent à la même règle de nommage. Dans notre exemple **Admin_**IndexController.

application/modules/admin/IndexController.php

```
class Admin_IndexController extends Zend_Controller_Action
{

    public function init()
    {

     $this->view->headTitle ( 'Mon application avec Zend_Application (Administrateur)' );
    }

    public function indexAction()
    {
       $this->_response->appendBody('Bievenue Admin');

    }


}
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

Notre application modulaire est à présent opérationnelle, essayons maintenant de tester que tout fonctionne en tapant ces URLs **http://localhost/zf-zaProject/public** et **http://localhost/zf-zaProject/public/admin** dans le navigateur.

IV - Ajouter une ressource

Les ressources que nous avons ajoutées jusque-là sont un peu particulières car elles ont des plugins prédéfinis dans Zend_Application et elles répondent à un schéma de configuration bien déterminé. Les ressources dont il est question sont le FrontController, Layout ,Db etc. Vous pouvez consulter la liste complète avec tous les paramètres de configuration dans la documentation officielle de Zend_Application.
Vous allez vous dire pourquoi ce n'est pas le cas des autres ressources ? Imaginez qu'on avait fait la même chose avec les autres ressources, il serait vraiment très difficile de se souvenir et d'obéir aux différents schémas de configuration vu la quantité innombrable de ressources disponibles donc les concepteurs de Zend Framework nous ont laissés le soin de définir nous même à notre guise la configuration de nos ressources. Rajouter à cela les ressources dont le plugin est préfinit sont considérées comme étant incontournables de l'élaboration dune application MVC.
Si vous avez l'habitude dans le passé d'enregistrer tous les objets, méthodes et ressources que vous initialisés dans votre bootstrap avec Zend_Registry pour leurs réutilisation.
Avec Zend_Application ce n'est plus nécessaire car il fait office de registre est enregistre systématiquement toutes les méthodes et ressources que vous avez définies dans votre bootstrap. Nous allons voir dans les exemples ci-dessous comment il vous possible de les invoquer.
Pour un exemple d'ajout de ressource, nous allons doter notre application d'un Log qui va écrire vers la console de firebug en mode développement et vers un fichier en mode production.
Pour procéder nous allons rajouter deux dossiers (ressources et logs) à la base de notre application, l'un va accueillir les plugins et l'autre notre log.
Nous allons maintenant revenir sur notre fichier **application.ini** pour initialiser les paramètres de notre log.

application/configs/application.ini

```
[production]
...
...
...
#Ajout du path vers nos plugins
pluginpaths.Application_Resources = APPLICATION_PATH "/Resources"


# initilisation du log
resources.logs.writer.type = stream
resources.logs.writer.file = APPLICATION_PATH "/logs/application.log"

...
...
...

[development : production]
resources.logs.writer.type = firebug
```

Définissons notre plugin **logs.php** que nous allons placer dans le dossier que nous avons prévu a cet effet.Lorsque nous ajoutons une ressource à notre fichier de configuration (excepté les ressources préfinis), Zend_Application s'attend toujours à trouver une classe qui étend **Zend_Application_Resource_ResourceAbstract** dans le dossier plugins dont nous avons définis le path précédemment. La fonction **init** doit être implicitement déclarée sinon une exception sera déclenchée.

application/resources/logs.php

```
class Application_Resources_Logs extends Zend_Application_Resource_ResourceAbstract
{

 public function init()
 {
  // récupère les paramètres de notre ressource
  $options = $this->getOptions();

  // récupération type de notre log (stream ou firebug)
  $type = $options['writer']['type'];
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/resources/logs.php

```
  //defini le writer de notre log selon le type.
  switch ( $type)
  {
   case 'stream':
    $writer = new Zend_Log_Writer_Stream($options['writer']['file']);


    break;
   case 'firebug':

    $writer = new Zend_Log_Writer_Firebug();

    break;
   default:
    $writer = NULL;
  }

  if (isset($writer)) {

   $logger = new Zend_Log($writer);

   return $logger;
  }

 }




}
```

Ce qui il ya lieu de retenir dans notre logs, est certainement la méthode $this->getOptions() qui récupère les paramètres de notre logs .

- **$this** fait référence à notre ressource .
- **getOptions()** récupère les paramètres sous forme d'un tableau, la profondeur d'accès a une option est déterminé par celle définit dans la configuration après le nom de la ressource.

Pour y voir plus claire, si par exemple dans votre fichier de configuration vous avez une ressource définit comme ceci.

```
resources.nomressource.niveau1.niveau2.niveau3 =   test
```

Soit $option est le résultat de la méthode getOptions(), l'option peut être récupérée comme ceci.

```
$options[niveau1][ niveau2][ niveau3]  == 'test' ; //  revnoit  vrai
```

Quant il s'agit d'un plugin de ressource la fonction **init** doit impérativement renvoyer une instance de l'objet, Zend_Log dans notre cas. Pour que Zend_Application l'inscrit dans son registre.

```
$logger = new Zend_Log($writer);

   return $logger;
```

Afin de tester si notre log fonctionne bien nous allons rajouter quelques instructions d'écriture dans nos contrôleurs.

application/modules/default/IndexController.php

```
class IndexController extends Zend_Controller_Action
{


 public function init()
```


Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

application/modules/default/IndexController.php

```
 {
  $this->view->headTitle ( 'Mon application avec Zend_Application (Utilsateur par défaut)' );
 }

 public function indexAction()
 {
  $this->_response->appendBody('Bienvenue');

  $bootstrap = $this->getInvokeArg('bootstrap');

  if ($bootstrap->hasResource('logs'))
  {
      $logs = $bootstrap->getResource('logs');

      if (isset($logs)){

           $logs->info('une information');
           $logs->warn('un avertissement');
           $logs->err('une erreur');
      }
  }

 }


}
```

Voici une autre méthode d'accéder à nos ressources, vous pouvez utiliser à votre convenance l'une ou l'autre.

application/modules/admin/IndexController.php

```
class Admin_IndexController extends Zend_Controller_Action
{

 public function init()
 {
  /* Initialize action controller here */
  $this->view->headTitle ( 'Mon application avec Zend_Application (Administrateur)' );
 }

 public function indexAction()
 {
  $this->_response->appendBody('Bienvenue Admin');


  $bootstrap = $this->getInvokeArg('bootstrap');
     $conteneur  = $bootstrap->getContainer();

  if (isset($conteneur->logs)) {

   $logs = $conteneur>logs;

   $logs->info('une information');
   $logs->warn('un avertissement');
   $logs->err('une erreur');

  }
 }
}
```

V - Conclusion

Personnellement, j'apprécie beaucoup l'utilité de Zend_Application et je trouve que, c'est une très forte valeur ajoutée au Framework, car il apporte une certaine subtilité à la configuration des applications.

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.

Il faut savoir aussi, que nous n'avons évoqué dans ce tutoriel n'est qu'une infime partie de ce qu'on peut faire avec Zend_Application, alors n'hésitez pas à faire un tour dans la documentation officielle pour voir toutes les possibilités que nous offre ce composant.

Zend_Application : Documentation officielle

Les sources présentées sur cette pages sont libre de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © - AIT YAHIA Idir. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérets. Droits de diffusion permanents accordés à developpez LLC.