

# ZF2 For ZF1 Developers

Gary Hockin

~~5th December 2012~~

26th February 2013

**Who?**



<http://blog.hock.in>

**@GeeH**

**Spabby** in #zftalk (freenode)

**What?**

**Why?**

**THERE WILL BE CODE!**

# Getting Started



# **Namespaces**

# ZF1

```
class FeaturedController extends Zend_Controller_Action
{
    public function indexAction()
    {
        $logfile = date('dmY') . '.log';
        $logger = new Zend_Log();
        $writer = new Zend_Log_Writer_Stream($logfile);
        $logger->addWriter($writer);
    }
}
```

# ZF2

```
namespace Application\Controller;

use Zend\Mvc\Controller\AbstractActionController;
use Zend\Log\Logger;
use Zend\Log\Writer\Stream as Writer;

class FeaturedController extends AbstractActionController;
{
    public function indexAction()
    {
        $logfile = date('dmY') . '.log';
        $logger = new Logger();
        $writer = new Writer($logfile);
        $logger->addWriter($writer);
    }
}
```

**Goodbye...**

Long\_Underscore\_Seperated\_Class

**Hello...**

Nice\Snappy\Namespaced\Class

# Modules

# **Bootstrapping**

# **Service Manager**

```
namespace Application;

use Zend\Mvc\Controller\AbstractActionController;
use Application\Service\MyService;

class MyController extends AbstractActionController
{
    protected $myService;

    public function __construct(MyService $myService)
    {
        $this->myService = $myService;
    }

    public function indexAction()
    {
        return array(
            'myViewVar' => $myService->getMyViewVar()
        );
    }
}
```



```
use Application\Controller\MyController;

public function getServiceConfig ()
{
    return array(
        'factories' => array(
            'myService' => function(ServiceManager $sm) {
                $myService = new \Application\Service\MyService ();
                $myService->setSomeProperty( true);
                return $myService;
            },
        ),
    );
}

public function getControllerConfig ()
{
    return array(
        'factories' => array(
            'MyController' => function(ServiceLocator $sl) {
                $myService = $sl->getServiceManager() ->get('myService');
                $myController = new MyController($myService);
                return $myService;
            },
        ),
    );
}
```

# Dependency Injector

# **Event Manager**

# Module.php

```
public function init(ModuleManager $moduleManager)
{
    $moduleManager->getEventManager()->attach(
        'loadModules.pre',
        function(\Zend\ModuleManager\ModuleEvent $e) {
            // do some pre module loading code here
        }
    );
}
```

# Module.php

```
public function onBootstrap (ModuleManager $moduleManager)
{
    $moduleManager->getEventManager()->attach(
        'myUser.Logout',
        function(\Zend\ModuleManager\ ModuleEvent $e) {
            // cleanup some cache/temp files here
        }
    );
}

/**
 * MyService class
 * Presume you have passed the EM into $eventManager via SM or Di
 */
public function logout()
{
    \\ do Logout code
    $this->eventManager->trigger('myUser.Logout');
}
```

# Routing

# ZF1

```
;Match route /product/1289 to ProductController DetailsAction with the  
parameter ProductId set to 1289
```

```
resources.router.routes.product.route = "/product/:ProductId"  
resources.router.routes.product.defaults.controller = "product"  
resources.router.routes.product.defaults.action = "details"
```

# ZF2

```
return array(
    'router' => array(
        'routes' => array(
            'category' => array(
                'type' => 'Segment',
                'options' => array(
                    'route' => '/product/:ProductId',
                    'constraints' => array(
                        'ProductId' => '[0-9]*'
                    ),
                    'defaults' => array(
                        '__NAMESPACE__' => 'Application\Controller',
                        'controller' => 'Product',
                        'action' => 'details',
                    ),
                ),
            ),
        ),
    ),
    ...
);
```



# **Zend View**

# **Zend View**

## View Helpers

# ZF1

## application.ini

```
resources.view.helperPath.My_View_Helper = "My/View/Helper"
```

## My/View/Helper/Bacon.php

```
class My_View_Helper_Bacon extends Zend_View_Helper_Abstract
{
    public function bacon()
    {
        return 'Bacon';
    }
}
```

# ZF2

## Module.php

```
public function getViewHelperConfig ()
{
    return array(
        'invokables' => array(
            'bacon' => 'My\View\Helper\Bacon'
        )
    );
}
```

## My\View\Helper\Bacon.php

```
namespace My\View\Helper

class Bacon extends \Zend\View\Helper\AbstractHelper
{
    public function __invoke ()
    {
        return 'bacon';
    }
}
```

**Zend Db**

**Zend Db Adapter  
&  
Zend Db TableGateway**

# ZF1

## application.ini

```
resources.db.adapter = "PDO_MYSQL"  
resources.db.isdefaulttableadapter = true  
resources.db.params.dbname = "mydb"  
resources.db.params.username = "root"  
resources.db.params.password = "password"  
resources.db.params.host = "localhost"
```

## Application\Model\Table\User.php

```
class Application_Model_Table_User extends Zend_Db_Table_Abstract  
{  
    protected $_name = 'user';  
  
    public function getUser($id)  
    {  
        return $this->find($id);  
    }  
}
```

# ZF2

## config/autoload/database.local.php

```
return array(  
    'service_manager' => array(  
        'factories' => array(  
            'Adapter' => function ($sm) {  
                return new Zend\Db\Adapter\Adapter(array(  
                    'driver'      => 'pdo',  
                    'dsn'        => 'mysql:dbname=db;host=127.0.0.1',  
                    'database'   => 'db',  
                    'username'   => 'user',  
                    'password'   => 'bacon',  
                    'hostname'   => '127.0.0.1',  
                ));  
            },  
        ),  
    ),  
);
```



# ZF2

## Application\Module.php

```
namespace Application;
use Zend\Db\TableGateway\TableGateway;

class Module
{
    public function getServiceConfig()
    {
        return array(
            'factories' => array(
                'UserTable' => function ($serviceManager) {
                    $adapter = $serviceManager->get('Adapter');
                    return new TableGateway('user', $adapter);
                },
            ),
        );
    }
}
```

# ZF2

## Application\Controller\IndexController.php

```
namespace Application\Controller;

class IndexController extends AbstractActionController
{
    public function indexAction()
    {
        $userTable = $this->getServiceLocator()->get('UserTable');
        return new ViewModel(array('user' => $userTable->select(
            array('id' => '12345'))
        ));
    }
}
```

# **Zend Db TableGateway & Object Hydration**

# **Zend Db TableGateway & Object Hydration**

`http://hock.in/WIP1qa`

# **Zend Form**

# Zend Form

<http://hock.in/11SCIvU>

# Questions?

<http://blog.hock.in>

**@GeeH**

**Spabby** in #zftalk (freenode)

Slides and Feedback:

**<https://joind.in/8247>**

# Want to get a deeper knowledge of ZF2

---

Check out our Zend Framework 2 Fundamentals class!

## Next dates:

- March 25, 13-15:00 CET
- April 22, 4-6pm PDT
- May 13, 9-11am PDT / 18-20:00 CET
- June 17, 14-16:00 CET



<http://www.zend.com/services/training/course-catalog/zend-framework-2>