TYPO3.Flow - Feature # 29258

Status:	Needs Feedback	**Priority:**	Should have
Author:	Fernando Arconada	**Category:**	
Created:	2011-08-26	**Assigned To:**	
Updated:	2012-03-14	**Due date:**	
PHP Version:			
Has patch:	No		
Complexity:			
Subject:	Provide a way to override classes by environment		

Description

In FLOW3 you can define several environments in your configuration. But, how could you override a class (ie. a controller) depending of the environment?.

If you always can clone your server you dont need environments, even in configuration.

I'm thinking in something like

```
MyPackage
 Classes
  mycontroller.php
  ValidationEnvironment
    mycontroller.php
```

The class is only overrided if it is defined for this environment
Cause the feature could decrease the performance of the framework it could be enabled/disabled by a config option

History

#1 - 2011-08-29 21:42 - Karsten Dambekalns

- Project changed from TYPO3 Flow Base Distribution to TYPO3.Flow

#2 - 2011-08-31 15:27 - Christopher Hlubek

I would recommend to use configuration settings for that (which can be context dependent). If this is not enough, the Objects.yaml can override interfaces but not concrete classes per context (without performance losses).

#3 - 2011-08-31 15:35 - Fernando Arconada

As yu said Objects.yaml only override interfaces,
For example you are doing A/B testing and for example you need to have an additional variable in your view. how to do that? conditions inside the action? It doesnt looks very clean. It is easy with template views, for example:

```
protected function initializeView(\TYPO3\FLOW3\MVC\View\ViewInterface $view)    {
        /**         * @var TYPO3\FLOW3\MVC\Web\Request
         */
        $requestArguments = $this->controllerContext->getRequest()->getArguments();
        if(isset($requestArguments['bversion']) && $requestArguments['bversion']=='1' && $view instanceof \TYPO3\Fluid\View\TemplateView ) {
            $packageName = $this->controllerContext->getRequest()->getControllerPackageKey();
            $view->setTemplateRootPath("/var/www/html/bversion/$packageName/Templates");
            $view->setLayoutRootPath("/var/www/html/bversion/$packageName/Layouts");
```

```
  }
}
```

but this is only at initializeView, such kind of conditions in every action....

#4 - 2012-03-14 15:07 - Christian Müller

- Status changed from New to Needs Feedback

- Has patch set to No

I don't really see the need for this. Debugging on a live machine is bad all the time. Maybe we could introduce a context condition viewhelper so you can change output by context...

#5 - 2012-03-14 19:29 - Fernando Arconada

not only debugging, maybe you need to introduce a new feature in a live system (you could change the environment in a dynamic way with apache). Some changes need to be aproved in staging before going production.... continuous deployment in life system is better with this feature, for example yahoo use to do it using a lot of conditionals but could be more clean with environments