Zend_Config_Ini with variable expansion - Douglas Hammond

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

Zend Framework: Zend_Config_Ini_Expandable Component Proposal

Proposed Component Name	Zend_Config_Ini_Expandable
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Config_Ini_Expandable
Proposers	Douglas Hammond
Zend Liaison	TBD
Revision	1.0 - 1 January 2008: Initial Draft. (wiki revision: 6)

Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

1. Overview

Extend Zend_Config_Ini to support variable expansion.

2. References

3. Component Requirements, Constraints, and Acceptance Criteria

All values in ini file to be expanded from previously defined values.

- This component **will** expand variables defined previously in the ini file.
- This component **will** expand variables passed as array to constructor.

4. Dependencies on Other Framework Components

- Zend_Config_Ini

5. Theory of Operation

The component is instantiated with a mind-link that ...

6. Milestones / Tasks

- Milestone 1: Look at current Zend_Config_Ini to see how to add functionality [DONE]
- Milestone 2: Working prototype checked into the incubator supporting use cases #1
- Milestone 4: Unit tests exist, work, and are checked into SVN.
- Milestone 5: Initial documentation.

7. Class Index

- Zend_Config_Ini_Expandable

8. Use Cases

UC-01

```
config.ini
; variables passed in by application environment startup
; ${application.rootDir}, ${application.configDir}, ${application.libraryDir},
; ${application.dataDir}

layout.layoutPath = "${application.rootDir}/application/layouts"


test.php

$rootDir = realpath(dirname(__FILE__) . DIRECTORY_SEPARATOR . '..');
$configDir = realpath($rootDir . DIRECTORY_SEPARATOR . 'config');
$libraryDir = realpath($rootDir . DIRECTORY_SEPARATOR . 'library');
$dataDir = realpath($rootDir . DIRECTORY_SEPARATOR . 'data');

$config = new Zend_Config_Ini_Expandable(
    $configDir . DIRECTORY_SEPARATOR . 'config.ini',
    $configSection,
    true,
    array( 'application' =>
        array(
            'rootDir' => $rootDir,
            'configDir' => $configDir,
            'libraryDir' => $libraryDir,
            'dataDir' => $dataDir
        )
    )
);

echo $config->application->layoutPath;
```

9. Class Skeletons

```
class Zend_Config_Ini_Expandable extends Zend_Config_Ini
{

    public function __construct($filename, $section = null, $options = false, array $array = array())
    {
        parent::__construct($filename, $section, $options);
        $this->merge(new Zend_Config($array, true));
        $this->_postProcessValues($this);
    }

    protected function _postProcessValues($config)
    {
        foreach ($config as $key => $value) {
            if (is_a($value, "Zend_Config")) {
                $this->_postProcessValues($value);
            } else {
                $config->__set($key, preg_replace_callback("|\\\${(.*)}|",
                        array($this, '_postProcessValuesCallback'), $value));
            }
        }
    }

    protected function _postProcessValuesCallback($matches)
    {
        if (isset($matches[1])) {
            $path = explode($this->_nestSeparator, $matches[1]);

            $result = $this;
            foreach($path as $value) {
                if(!isset($result->$value)) {
                    return $matches[0];
                }
                $result = $result->$value;
            }
            return $result;
        }
        return $matches[0];
    }
}
```

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>