Integrating Zend_Config Proposal (Please Remove this Proposal)

<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_DB Component Proposal

Proposed Component Name	Zend_DB
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_DB
Proposers	Daniel Schwiperich
Revision	1.1 - 1 August 2006: Updated from community comments. (wiki revision: 12)

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

(First) Get Zend_DB.::factory with a Zend_Config object working

2. References

*

3. Component Requirements, Constraints, and Acceptance Criteria

*

4. Dependencies on Other Framework Components

- Zend_DB
- Zend_Config

5. Theory of Operation

Zend_DB::factory only accepts an array as param.
To get compatible to Zend_Config it has also to accept an object of type Zend_Config
I've changed therefore the factorymethod a little bit, that it'll accept a Zend_Config object
and turns the object into an array. If the object contains other types than Strings an exception will be thrown.
See the example below...

6. Milestones / Tasks

zone: Missing {zone-data:milestones}

7. Class Index

- Zend_DB
- Zend_Config

8. Use Cases

UC-01

An example config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
 <development>
  <webhost>localhost</webhost>
  <database>
   <dbtype>PDO_MYSQL</dbtype>
   <host>localhost</host>
   <username>root</username>
   <password>xxx</password>
   <dbname>xxxx</dbname>
  </database>
 </development>
</config>
```

The DB-Init in index.php

```
//...
    $config = new
Zend_Config(Zend_Config_Xml::load(CONFIG.DIRECTORY_SEPARATOR.'config.xml','development'));
    $db = Zend_Db::factory($config->database->dbtype, $config->database);
//...
```

9. Class Skeletons

The updated DB_Zend::factory method. Changes between the //----- comments

```
static public function factory($adapterName, $config = array())
{
        if (!is_string($adapterName) or !strlen($adapterName)) {
            throw new Zend_Db_Exception('Adapter name must be specified in a string');
        }
//-----------------------------------------------------------------------------------------------
        if (!is_array($config) && !is_a($config, 'Zend_Config')) {
            throw new Zend_Db_Exception('Configuration must be an array or a Zend_Config-object');
        }

        //Converts Zend_Config object to an array
        if(is_a($config, 'Zend_Config')){
            $tmp = array();
            foreach($config as $attribute => $val){
                if(is_string($val)){
                    $tmp[$attribute] = $val;
                }
                else throw new Zend_Db_Exception('Your Zend_Config object must only contains values of
the type String. Found a:'.gettype($val));
            }
            $config = $tmp;
            $tmp = null;
        }
//-----------------------------------------------------------------------------------------------
        $adapterName = strtolower($adapterName); // normalize input
        if (substr($adapterName, 0, 3) === 'pdo') {
            if ($adapterName{3} !== '_') // this check will be removed, and underscores will be
required in 0.2
                trigger_error("Use of adapter name '$adapterName' is deprecated. Underscores will be
required in 0.2 release. Please use 'PDO_".(strtoupper(substr($adapterName,3))).."'.", E_USER_WARNING);
            $adapterName = 'Zend_Db_Adapter_Pdo_' . ucfirst(ltrim(substr($adapterName, 3),'_'));
        } else {
            $adapterName = 'Zend_Db_Adapter_' .
                           str_replace(' ',
                                       '_' ,
                                       ucwords(str_replace('_', ' ', $adapterName)));
        }

        Zend::loadClass($adapterName);

        return new $adapterName($config);
}
```

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