Zend_Session_SaveHandler_Doctrine - Matthew Lurz

<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_Session_SaveHandler_Doctrine Component Proposal

Proposed Component Name	Zend_Session_SaveHandler_Doctrine
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Session_SaveHandler_Doctrine
Proposers	Matthew Lurz
Zend Liaison	TBD
Revision	0.1 - 3 June 2009: Initial Draft. (wiki revision: 4)

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

Zend_Session_SaveHandler_Doctrine is a session save handler modeled after Zend_Session_SaveHandler_DbTable for the Doctrine OR/M.

A working prototype is available here

2. References

- Doctrine_Table
- Zend_Session_SaveHandler_DbTable

3. Component Requirements, Constraints, and Acceptance Criteria

- This component will require a pre-configured Doctrine environment
- This component will write session data to a Doctrine_Table

4. Dependencies on Other Framework Components

- Zend_Session_SaveHandler_Interface

5. Theory of Operation

The constructor of Zend_Session_SaveHandler_Doctrine receives an array or Zend_Config object specifying the name of the table to write to as well as primary key and column data.

6. Milestones / Tasks

- Milestone 1: design notes will be published here
- Milestone 2: Working prototype checked into the incubator supporting use cases #1, #2, ...
- Milestone 3: Working prototype checked into the incubator supporting use cases #3 and #4.
- Milestone 4: Unit tests exist, work, and are checked into SVN.
- Milestone 5: Initial documentation exists.

7. Class Index

- Zend_Session_SaveHandler_Doctrine

8. Use Cases

UC-01

Write session data to a table

Generate model class from Yaml

```
Session:
    columns:
        session_id:
            type: string(32)
            fixed: true
            primary: true
        modified:
            type: integer
            notnull: true
        lifetime:
            type: integer
            notnull: true
        data:
            type: string
            notnull: true
```

...Or, create model class manually

```
abstract class BaseSession extends Doctrine_Record
{
    public function setTableDefinition()
    {
        $this->setTableName('session');
        $this->hasColumn('id', 'string', 32, array('type' => 'string', 'fixed' => 1,
'primary' => true, 'length' => '32'));
        $this->hasColumn('modified', 'integer', null, array('type' => 'integer'));
        $this->hasColumn('lifetime', 'integer', null, array('type' => 'integer'));
        $this->hasColumn('data', 'string', null, array('type' => 'string'));
    }
}
```

```
$config = array(
    'tableName'         => 'Session',
    'dataColumn'        => 'data',
    'lifetimeColumn'    => 'lifetime',
    'modifiedColumn'    => 'modified',
    'primaryKeyColumn'  => 'id',
);

Zend_Session::setSaveHandler(new Zend_Session_SaveHandler_Doctrine($config));
Zend_Session::start();
```

9. Class Skeletons

```
class Zend_Session_SaveHandler_Doctrine implements Zend_Session_SaveHandler_Interface
{
```

```
    const DATA_COLUMN           = 'dataColumn';
    const LIFETIME_COLUMN       = 'lifetimeColumn';
    const MODIFIED_COLUMN       = 'modifiedColumn';
    const PRIMARY_KEY_COLUMN    = 'primaryKeyColumn';

    const LIFETIME              = 'lifetime';
    const OVERRIDE_LIFETIME     = 'overrideLifetime';
    const TABLE_NAME            = 'tableName';

    /**
     * @var string
     */
    protected $_dataColumn = null;

    /**
     * @var string
     */
    protected $_lifetimeColumn = null;

    /**
     * @var string
     */
    protected $_modifiedColumn = null;

    /**
     * @var string
     */
    protected $_primaryKeyColumn = null;

    /**
     * @var int
     */
    protected $_lifetime = false;

    /**
     * @var boolean
     */
    protected $_overrideLifetime = false;

    /**
     * @var string
     */
    protected $_sessionName = null;

    /**
     * @var string
     */
    protected $_sessionSavePath = null;

    /**
     * @var string
     */
    protected $_tableName = null;

    /**
     * Constructor
     *
     * @param  Zend_Config|array $config
     * @return void
```

```
     * @throws Zend_Session_SaveHandler_Exception
     */
    public function __construct($config)
    {
    }

    /**
     * Destructor
     *
     * @return void
     */
    public function __destruct()
    {
    }

    /**
     * Set session lifetime and optional whether or not the lifetime of an
     * existing session should be overridden
     *
     * $lifetime === false resets lifetime to session.gc_maxlifetime
     *
     * @param int $lifetime
     * @param boolean $overrideLifetime (optional)
     * @return Zend_Session_SaveHandler_Doctrine
     * @throws Zend_Session_SaveHandler_Exception
     */
    public function setLifetime($lifetime, $overrideLifetime = null)
    {
    }

    /**
     * Retrieve session lifetime
     *
     * @return int
     */
    public function getLifetime()
    {
    }

    /**
     * Set whether or not the lifetime of an existing session should be
     * overridden
     *
     * @param boolean $overrideLifetime
     * @return Zend_Session_SaveHandler_Doctrine
     */
    public function setOverrideLifetime($overrideLifetime)
    {
    }

    /**
     * Retrieve whether or not the lifetime of an existing session should be
     * overridden
     *
     * @return boolean
     */
    public function getOverrideLifetime()
    {
    }
```

```
    /**
     * Set primary key column
     *
     * @param string|array $key
     * @return Zend_Session_SaveHandler_Doctrine
     * @throws Zend_Session_SaveHandler_Exception
     */
    public function setPrimaryKeyColumn($key = 'id')
    {
    }

    /**
     * Retrieve primary key column
     *
     * @return array
     */
    public function getPrimaryKeyColumn()
    {
    }

    /**
     * Set session table name
     *
     * @param string $name
     * @return Zend_Session_SaveHandler_Doctrine
     */
    public function setTableName($name = 'Session')
    {
    }

    /**
     * Retrieve session table name
     *
     * @return string
     */
    public function getTableName()
    {
    }

    /**
     * Open Session
     *
     * @param string $savePath
     * @param string $name
     * @return boolean
     */
    public function open($savePath, $name)
    {
    }

    /**
     * Close session
     *
     * @return boolean
     */
    public function close()
    {
    }
```

```
    /**
     * Read session data
     *
     * @param string $id Session identifier
     * @return string Session data
     */
    public function read($id)
    {
    }

    /**
     * Write session data
     *
     * @param string $id
     * @param string $data
     * @return boolean
     */
    public function write($id, $data)
    {
    }

    /**
     * Destroy session
     *
     * @param string $id
     * @return boolean
     */
    public function destroy($id)
    {
    }

    /**
     * Garbage Collection
     *
     * @param int $maxlifetime
     * @return true
     */
    public function gc($maxlifetime)
    {
    }

    /**
     * Retrieve session lifetime
     *
     * @param Doctrine_Record $record
     * @return int
     */
    protected function _getLifetime(Doctrine_Record $record)
    {
    }

    /**
     * Retrieve session expiration time
     *
     * @param Doctrine_Record $record
     * @return int
     */
    protected function _getExpirationTime(Doctrine_Record $record)
```

{

```
        }
    }
```

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