Zend_Gearman

<ac:macro ac:name="info"><ac:parameter ac:name="title">Zend_Gearman</ac:parameter></ac:macro>

<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_Gearman Component Proposal

Proposed Component Name	Zend_Gearman
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Gearman
Proposers	Mike Willbanks
Zend Liaison	TBD
Revision	1.0 - 15 June 2011: Initial Draft. (wiki revision: 3)

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\Gearman is a component for interfacing with the Gearman server. The functionality allows for different adapters (PECL: Gearman, PEAR: Net_Gearman and a Mock class for unit testing) to interface as a client and a worker.

2. References

- PECL Gearman
- PEAR Net_Gearman (Github)
- Initial ZF 1 Version

3. Component Requirements, Constraints, and Acceptance Criteria

Most requirements take the form of "foo will do ...." or "foo will not support ...", although different words and sentence structure might be used. Adding functionality to your proposal is requirements creep (bad), unless listed below. Discuss major changes with your team first, and then open a "feature improvement" issue against this component.

- This component **will** include an abstract factory for client and workers.
- This component **will only** interface with existing client / workers (PECL, PEAR) and a new one Mock.
- This component **will** utilize a new directory called "workers" inside of application and modules.
- This component **will** attempt to mimic or create similar behaviors between adapters
- This component **will not** handle process monitoring.
- This component **will** make use of a new bootstrap resource and application.ini configuration.
- This component **will** support usage of multiple serialization techniques (json, php) for workloads.

4. Dependencies on Other Framework Components

Zend\Application - for initializing and bootstrapping "workers" directory.
Zend\Controller\Router - for routing the request if using the "workers" directory.

5. Theory of Operation

The component will provide a unified interface to utilizing Gearman, it will also make use of bootstrapping to utilize existing resources.

6. Milestones / Tasks

- Milestone 1: Finish proposal and architectural description.
- Milestone 2: Working prototype, excluding dependencies on ZF2 MVC.
- Milestone 3: Unit tests exist, work, and are checked into github.
- Milestone 4: Working prototype updated for dependencies on ZF2 MVC.
- Milestone 5: Initial documentation exists.

7. Class Index

- Zend\Gearman
- Zend\Gearman\Task
- Zend\Gearman\Client
- Zend\Gearman\Client\Pecl
- Zend\Gearman\Client\Pear
- Zend\Gearman\Client\Mock
- Zend\Gearman\Worker
- Zend\Gearman\Worker\Pecl
- Zend\Gearman\Worker\Pear
- Zend\Gearman\Worker\Mock

8. Use Cases

UC-01

Submitting a Gearman Task to Gearman

```
$task = new \Zend\Gearman\Task()->setFunction('doStuff')
                  ->setWorkload('something')
                  ->setTaskType(Task::DO_NORMAL_BACKGROUND);

$client = \Zend\Gearman\Client::factory('pecl');
$client->addServer('127.0.0.1', 4730);
$client->runTask($task);
```

9. Class Skeletons

```
// will be filled in later today
```

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