

## Blog Export: Rasmus' Toys Page, <http://toys.lerdorf.com/>

Thursday, September 24, 2009

### Playing with Gearman

This was written in September 2009 when the current version of Gearman was 0.9. Thanks to Eric Day for answering my dumb questions along the way.

To get started, install Gearman. I am on Debian, so this is what I installed:

```
% apt-get install gearman gearman-job-server gearman-tools libgearman1 libgearman-dev libdrizzle-dev
```

Enable Gearman in /etc/default/gearman-server

Set up Gearman to use MySQL for its persistent queue store in /etc/default/gearman-job-server

```
PARAMS="-q libdrizzle --libdrizzle-host=127.0.0.1 --libdrizzle-user=gearman \  
--libdrizzle-password=your_pw --libdrizzle-db=gearman \  
--libdrizzle-table=gearman_queue --libdrizzle-mysql"
```

```
% mysqladmin create gearman
```

```
% mysql  
mysql> create USER gearman@localhost identified by 'your_pw';  
mysql> GRANT ALL on gearman.* to gearman@localhost;
```

\*\* Careful, if you are running MySQL using --old-passwords this won't work with libdrizzle. You will need to get the 41-char password hash with a little snippet of PHP that does the double sha1 encoding:

```
% php -r "echo '*'.strtoupper(sha1(sha1('your_pw',true)));"
```

```
% mysql  
mysql> UPDATE mysql.user set Password='above_output' where User='gearman';
```

```
% mysqladmin flush-privileges
```

Then start the server.

```
% /etc/init.d/gearman-job-server start
```

Check to make sure gearmand is running. If it isn't, check for errors in /var/log/gearman-job-server/gearman.log

Also note that your password will be visible with 'ps' with this setup. The Gearman guys will be addressing this in gearmand-0.10

Next, let's get the gearman PHP extension installed. Grab it from svn:

```
% svn co http://svn.php.net/repository/pecl/gearman/trunk gearman  
% cd gearman  
% phpize  
% ./configure --with-php-config=/usr/local/bin/php-config
```

(If you have autoconf problems, apt-get install autoconf2.59 and set your

## Blog Export: Rasmus' Toys Page, <http://toys.lerdorf.com/>

PHP\_AUTOCONF env variable to "autoconf2.59")

% make install

edit your php.ini file and add: extension=gearman.so

Now we can add a worker. A worker is something that will process a Gearman request. We can write it in almost any language, but here is one in PHP (worker.php):

```
#!/usr/local/bin/php
```

Posted by Rasmus at 14:57

For all other readers who find this post interesting, but have - like me - no idea what you are talking about:

<http://gearman.org/>

Gearman provides a generic application framework to farm out work to other machines or processes that are better suited to do the work. It allows you to do work in parallel, to load balance processing, and to call functions between languages.

Anonymous on Sep 24 2009, 15:43

Yeah, I guess I could have explained that.

Anonymous on Sep 24 2009, 16:28

This is cool

Great job there. A lot of new perspectives for us. With memcache, it will be a killing. Persistence and multi-threading in PHP. Very cool

S.L.

Anonymous on Sep 24 2009, 22:51

Great article! Thank you for all the missing pieces. :)

Anonymous on Sep 25 2009, 10:23

Just wondering about the multi threading capacity of the worker library. Let say I have a worker.php script, and all tasks are submitted using addTaskBackground. If I run a single instance of the worker script, so does the worker library automatically fork the child processes to handle multi jobs? Or I have to handle the multi threading myself using some kind of pcntl?

I personally prefer Beanstalkd than Gearman due to its priority and delay capacity, but i have to handle the job execution manually. Gearman can auto execute a job via registered function, but not sure about the multi threading.

Thanks

Anonymous on Dec 23 2009, 01:32

I don't think there is any multi-threading capacity in the worker library. You launch as many workers as you think you will need and if you have more concurrent jobs than that, they will be queued up waiting for an available worker. Adding an external mechanism that could monitor the queue and potentially launch more workers doesn't seem like a very difficult thing to do. Someone has probably done it.

Anonymous on Dec 23 2009, 01:55

```
aptitude show libgearman1
```

```
No current or candidate version found for libgearman1
```

```
Package: libgearman1
```

```
State: not a real package
```

FYI libgearman1 doesn't seem to exist and will throw errors if you try to install the package.

Anonymous on Jan 18 2010, 07:26

Right, since writing this libgearman1 has been replaced by libgearman2 and then libgearman3.

Anonymous on Jan 18 2010, 07:44

You should update it for future reference :)

Anonymous on Jan 22 2010, 09:41

## **Blog Export: Rasmus' Toys Page, <http://toys.lerdorf.com/>**

@Van, No automatic multi threading or forking as such if you use a single worker.php. But as Rasmus said, I am sure there are ways and someone has already done it probably.  
Anonymous on May 16 2010, 05:44