



Hidden gears of your application



Sergej Kurakin

Problem

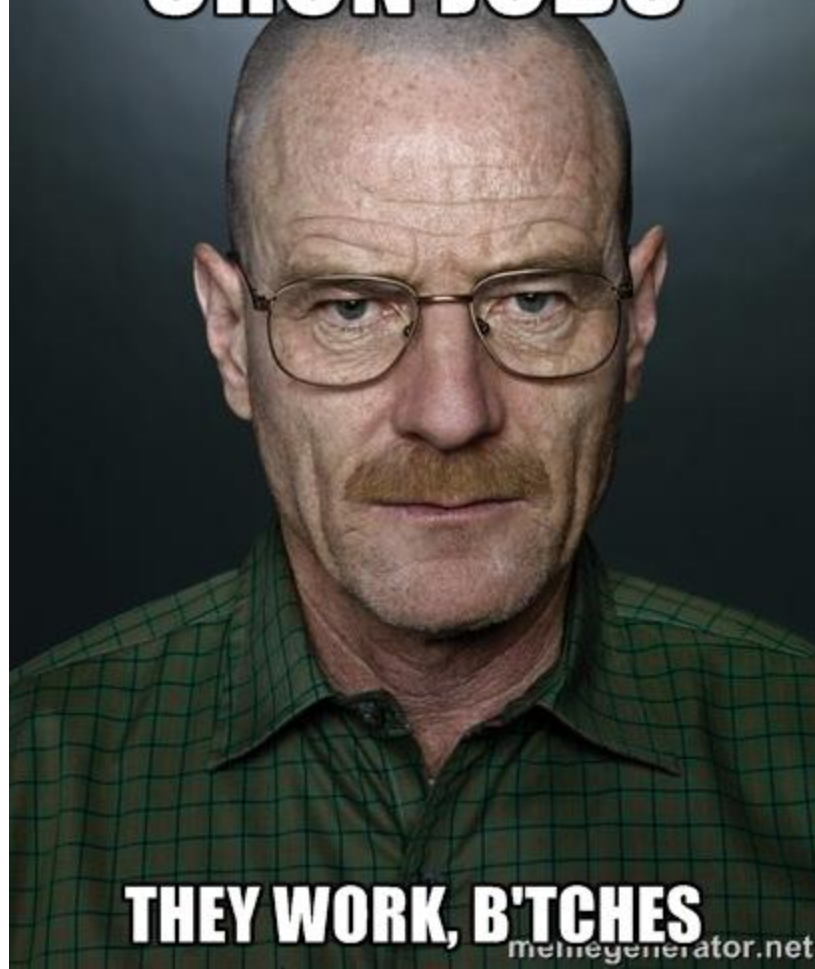
- Need for quick response
- Need for many updates
- Need for different jobs done
- Need for task to be done as different user on server side
- Near real-time job start
- Load distribution



Job Queue

- You put job to queue
- Worker takes the job and makes it done

CRON JOBS



THEY WORK, B'TCHES

memegenerator.net

Job Queue using Crons

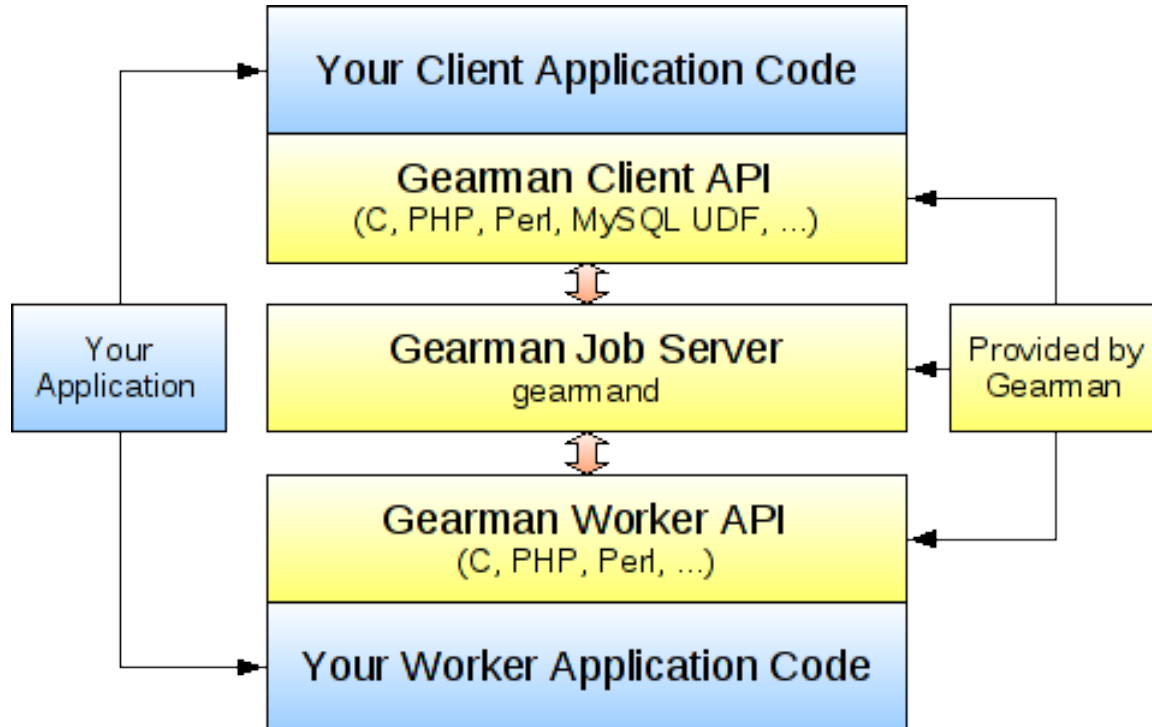
- Many different implementations
- Perfect for small scale
- Available on many systems/servers
- Crons are limited to running once per minute
- Harder to distribute load

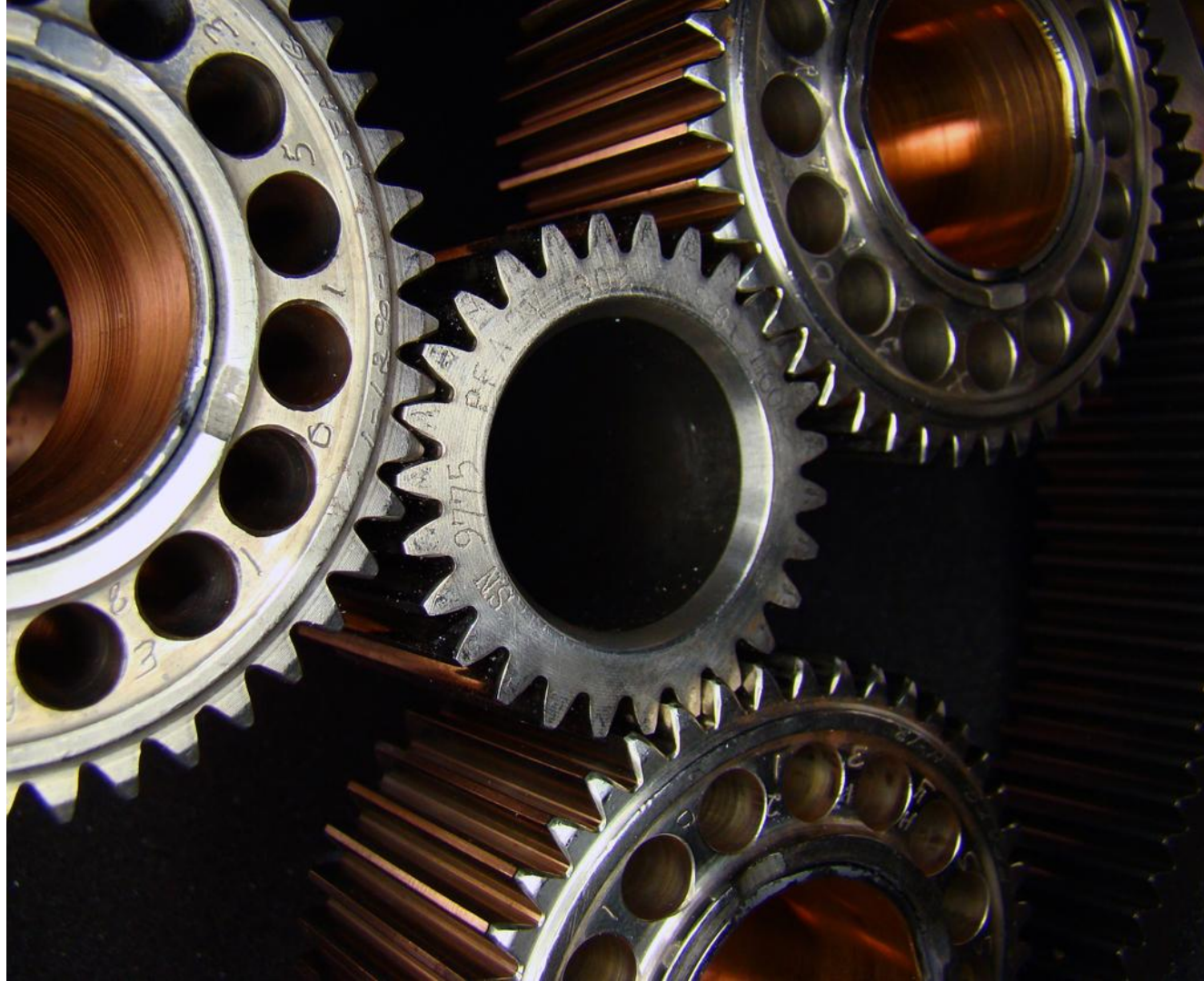


Gearman Job Server

- Job Queue
- <http://gearman.org/>
- C/C++
- Multi-language
- Scalable and Fault Tolerant
- Huge message size (up to 4 gig)

Gearman Stack





Gearman Job Types

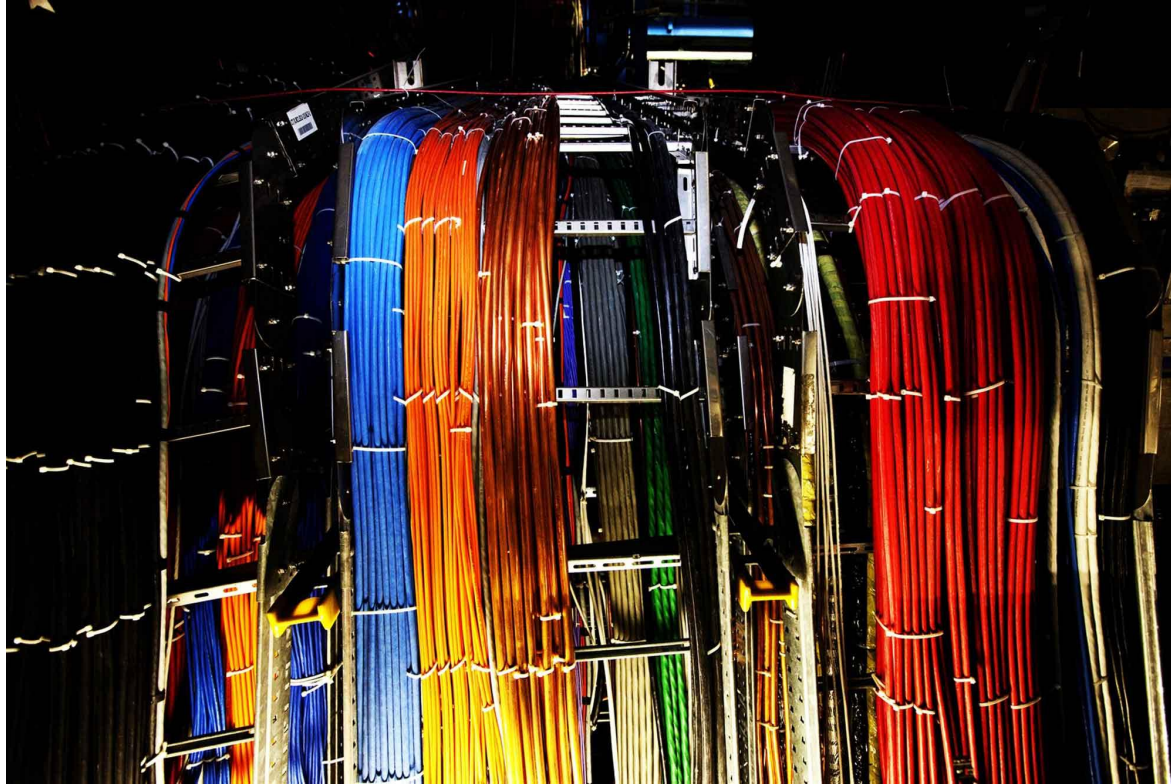
Normal Job

- Run Job
- Return Result

Background Job

- Run Job in Background
- No Return of Result

Gearman Parallel Tasks



Gearman Supported Languages

- C
- Perl
- NodeJS
- PHP
- Python
- Java
- C#/.NET
- Ruby
- Go
- Lisp

Job Priority

- Low
- Normal
- High

Gearman Worker Example

```
<?php
```

```
// Reverse Worker Code
```

```
$worker = new GearmanWorker();
```

```
$worker->addServer();
```

```
$worker->addFunction("reverse", function ($job) {
```

```
    return strrev($job->workload());
```

```
});
```

```
while ($worker->work());
```


Gearman Client Example

```
<?php
```

```
// Reverse Client Code
```

```
$client = new GearmanClient();
```

```
$client->addServer();
```

```
print $client->do("reverse", "Hello World!");
```

Gearman Client Example

```
<?php
```

```
// Reverse Client Code
```

```
$client = new GearmanClient();
```

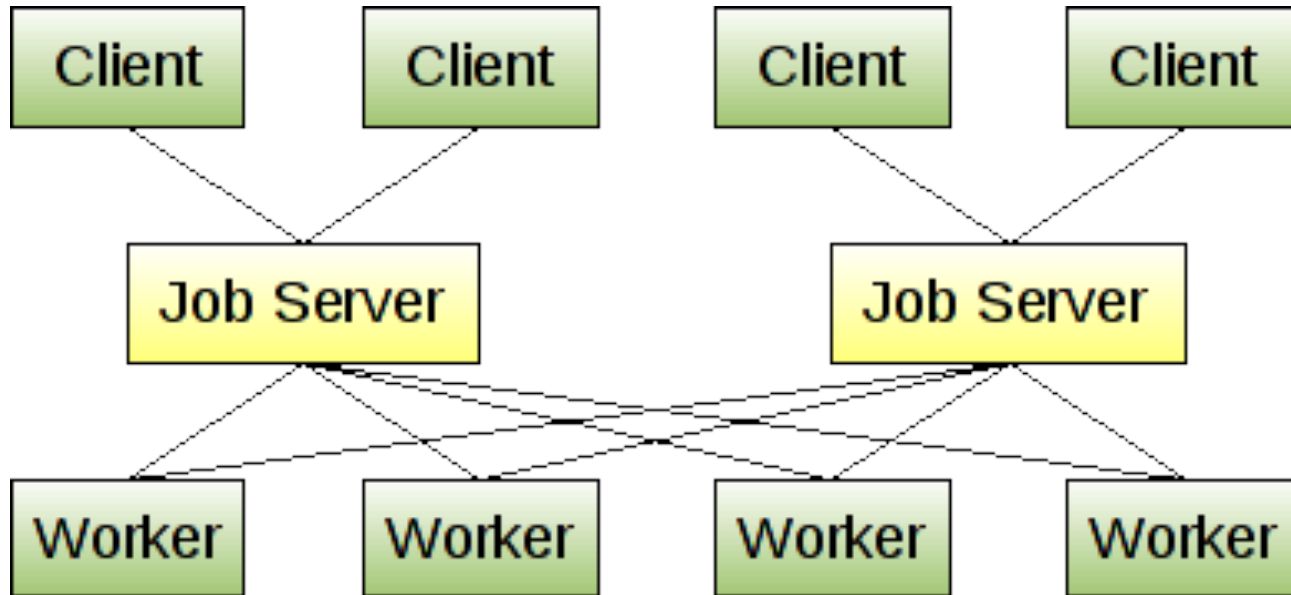
```
$client->addServer();
```

```
$client->doBackground("reverse", "Hello World!");
```

Running Worker in Background

- CLI
- screen / tmux
- supervisord - <http://supervisord.org/>
- daemontools - <http://cr.yp.to/daemontools.html>
- daemon - [http://en.wikipedia.org/wiki/Daemon_\(computing\)](http://en.wikipedia.org/wiki/Daemon_(computing))

High Availability



Persisted Job Queue

- libmysqlclient
- libdrizzle
- libmemcached
- libsqlite3
- Postgresql



Security

- No security at all in Gearman
- Protect by yourself: 127.0.0.1, Firewalls
- Ask your Sysadmins for help: Firewalls, Secure Tunnels
- Do some crazy APIs

Running Example



Usage Examples

- Image Resize
- Search Index updates
- Email sending
- External API calls
- Small tasks that can be done in background

Alternatives

- RabbitMQ - <http://www.rabbitmq.com/>
- Redis - <http://redis.io/>
- Amazon SQS - <http://aws.amazon.com/sqs/>
- IronMQ - <http://www.iron.io/mq>
- Beanstalkd - <http://kr.github.io/beanstalkd/>
- [JFGI](#)

Links

- <http://gearman.org/>
- <http://gearman.org/manual/>
- <http://gearman.org/getting-started/>
- <http://php.net/manual/en/book.gearman.php>
- <https://github.com/sergej-kurakin/gearman-demo>

Questions?



Sergej Kurakin

Work Email: sergej.kurakin@nfq.it

Personal Email: sergej@kurakin.info

<https://www.linkedin.com/in/sergejkurakin>

Special thanks to authors of all pictures used in this presentation.