



PRZETWARZANIE ROZPROSZONE ZADAŃ

Mariusz Gil
4Developers 2012, Poznań

W DWÓCH
SŁOWACH

AGENDA

**DLACZEGO WARTO
O TYM POROZMAWIAĆ?**

KLIK



REQUEST



PRZETWARZANIE DANYCH W PHP



RENDEROWANIE TEMPLATE



ODPOWIEDŹ

JEDNA ISTOTNA SPRAWA



IM SZYBCIEJ, TYM LEPIEJ

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

5.6s



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

5.6s



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ

REFRESH

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

7.2s



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ

REFRESH

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

9.8s



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ

REFRESH

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

13.3s



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ

REFRESH

REFRESH

REFRESH

ROZWIAZANIE?

KLIK



REQUEST

10ms



PRZETWARZANIE DANYCH W PHP

254ms



RENDEROWANIE TEMPLATE

98ms



ODPOWIEDŹ



DELEGACJA ZADANIA
DO ZEWNĘTRZNEGO
WORKERA

REQUEST

10ms

1. REQUEST

2. TOKEN ZADANIA

PRZETWARZANIE DANYCH W PHP

254ms

RENDEROWANIE TEMPLATE

98ms

PRZETWARZANIE
DANYCH

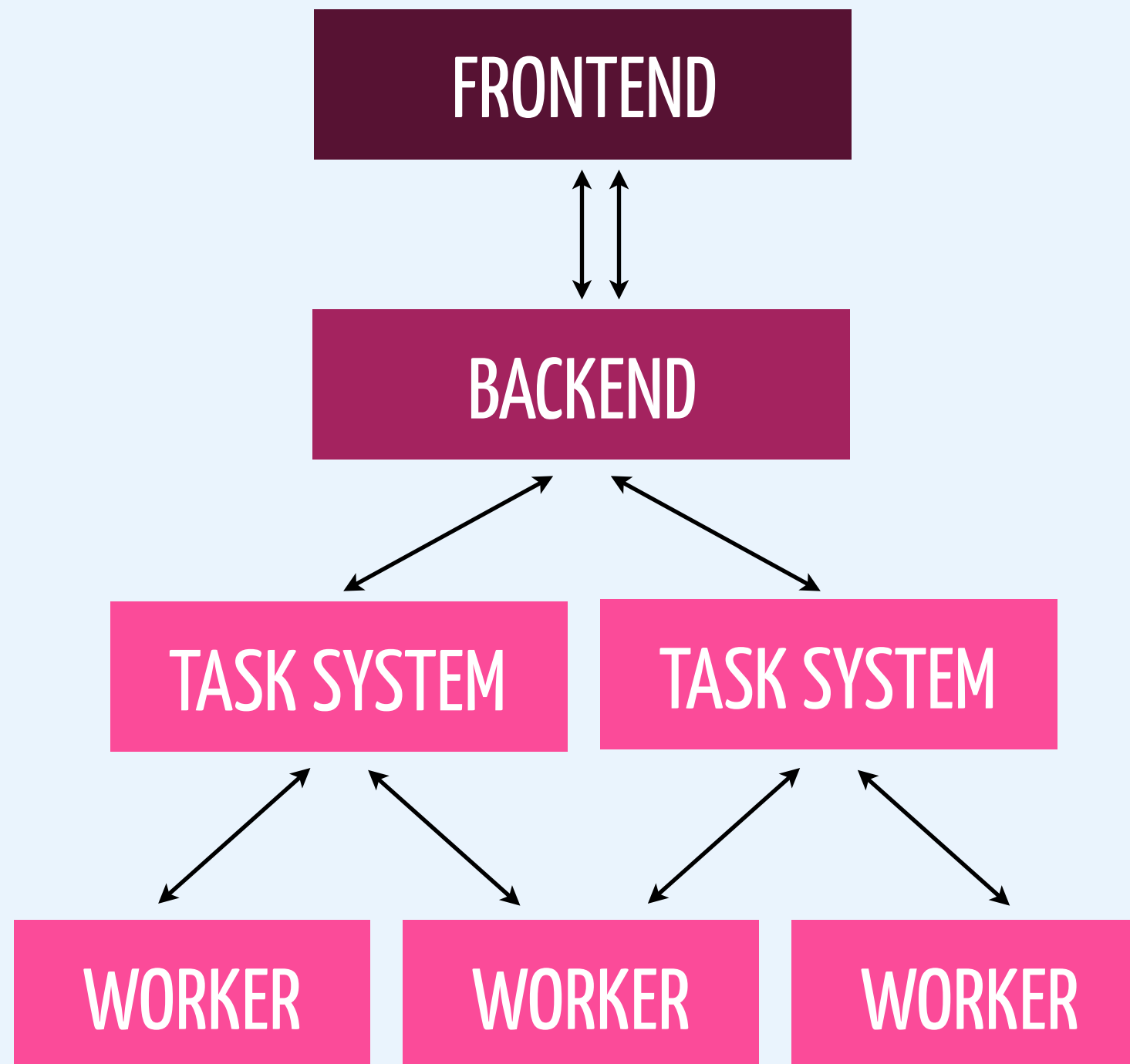
ODPOWIEDŹ

3. TOKEN ZADANIA

4. WYNIK ZADANIA

PREZENTACJA DANYCH Z WORKERA

(np. z wykorzystaniem wywołania ajax)





WSZYSTKO WYGLĄDA PROSTO
ALE ISTOTNE SĄ SZCZEGÓŁY

PROSTA IMPLEMENTACJA

WPISUJE SIĘ W PEWIEN SCHEMAT



AUTORSKI **FRAMEWORK PHP**
AUTORSKI **SYSTEM TEMPLATES**
AUTORSKI **CMS**

PROSTY WORKER CRON & PHP

ROZWIĄZANIE 1

PRZYKŁADOWA TABELA

ID	TASK_NAME	CREATED_AT	DATA	STATUS
1	CONTACT_EMAIL	2012-01-01 00:00:00	{"email":"john@doe.com"}	PROCESSING
2	INVITATION_EMAIL	2012-01-01 00:00:01	{"email":"john@doe.com"}	PROCESSING
3	CONTACT_EMAIL	2012-01-01 00:00:01	{"email":"john@doe.com"}	WAITING
4	ACCOUNT_DELETE	2012-01-01 00:00:02	{"id": 34532, "with_opts": true}	WAITING
5	INVITATION_EMAIL	2012-01-01 00:00:03	{"email":"john@doe.com"}	WAITING
6	IMAGE_RESIZE	2012-01-01 00:00:04	{"id": 322, "thumbs": true}	WAITING

DEFINICJA W CRON-TAB

```
* * * * * php -f /var/www/project/workers/simple_worker.php
```

FLOW

OZNACZ ZADANIE DO PRZETWORZENIA

WYKONAJ ZADANIE

USUN ZADANIE Z TABELI

ZAKONCZ PRACE

CZY SA DANE W TABELI?



PROBLEM

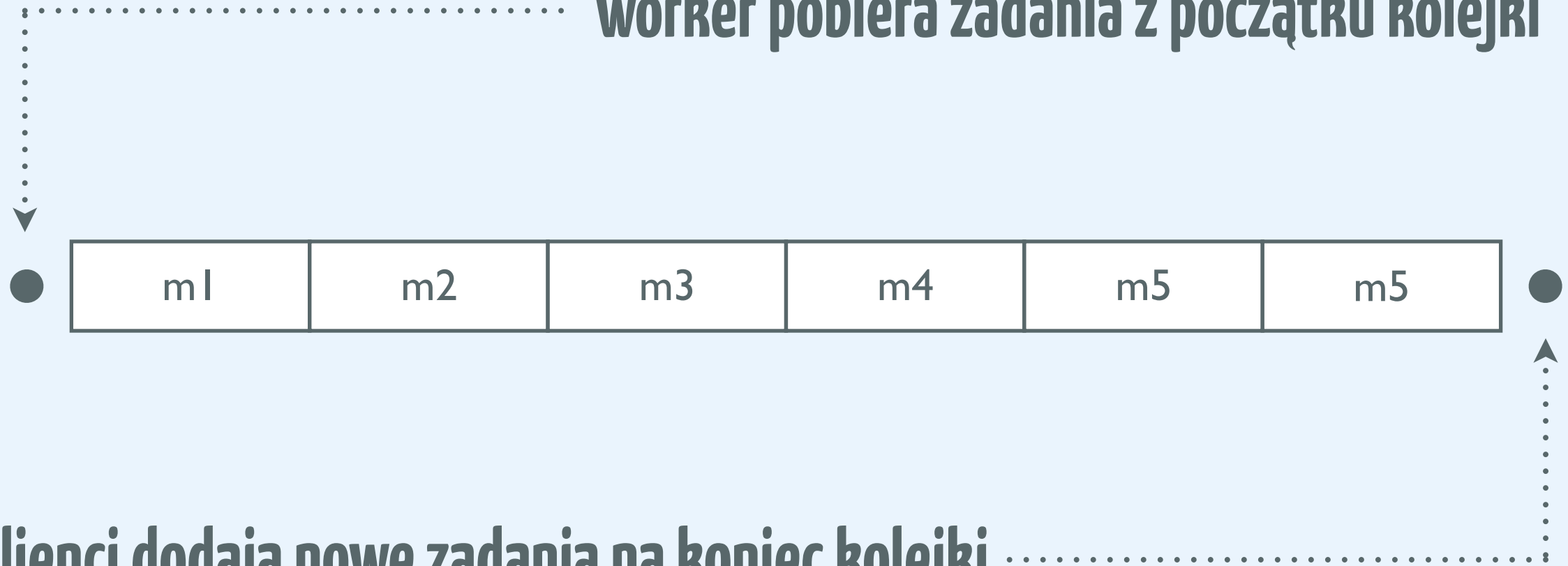
OBSŁUGA SYTUACJI KRYTYCZNYCH
WZNAWIANIE BŁĘDNYCH ZADAN
CZESTOTLIWOSC WYWOŁAŃ

MEMCACHEQ

PONAD MEMCACHEDB

ROZWIĄZANIE 2

worker pobiera zadania z początku kolejki



klienci dodają nowe zadania na koniec kolejki

KLIENT

DODANIE WPISU DO KOLEJKI

WORKER

PRZETWORZENIE WPISU Z KOLEJKI

```
<?php
/* connect to memcached server */
$memcache = memcache_connect('memcacheq_host', 21201);

/* append a message to queue */
memcache_set($memcache, 'queue_1', 'message 1', 0, 0);

/* append a message to queue */
memcache_set($memcache, 'queue_1', 'message 2', 0, 0);

/* append a message to queue */
memcache_set($memcache, 'queue_2', 'message 1', 0, 0);

memcache_close($memcache);
?>
```

.....

```
<?php
/* connect to memcached server */
$memcache = memcache_connect('memcacheq_host', 21201);

/* consume a message from queue */
memcache_get($memcache, 'queue_1');

/* consume a message from queue */
memcache_get($memcache, 'queue_1');

/* consume a message from queue */
memcache_get($memcache, 'queue_2');

memcache_close($memcache_obj);
?>
```

BEANSTALKD

WORK QUEUE

ROZWIĄZANIE 3

HISTORIA

BEANSTALKD POWSTAŁ NA
POTRZEBY APLIKACJI FACEBOOK
CAUSES, GDZIE MIAŁ SKRÓCIC CZAS
ODPOWIEDZI SERWERA

WORKER

PRZETWORZENIE ZADANIA

KLIENT

REJESTRACJA ZADANIA

```
<?php
// register Pheanstalk class loader
require_once('pheanstalk_init.php');

$pheanstalk = new Pheanstalk('127.0.0.1');

$job = $pheanstalk
    ->watch('testtube')
    ->ignore('default')
    ->reserve();

echo $job->getData();

$pheanstalk->delete($job);
```

.....

```
<?php
// register Pheanstalk class loader
require_once('pheanstalk_init.php');

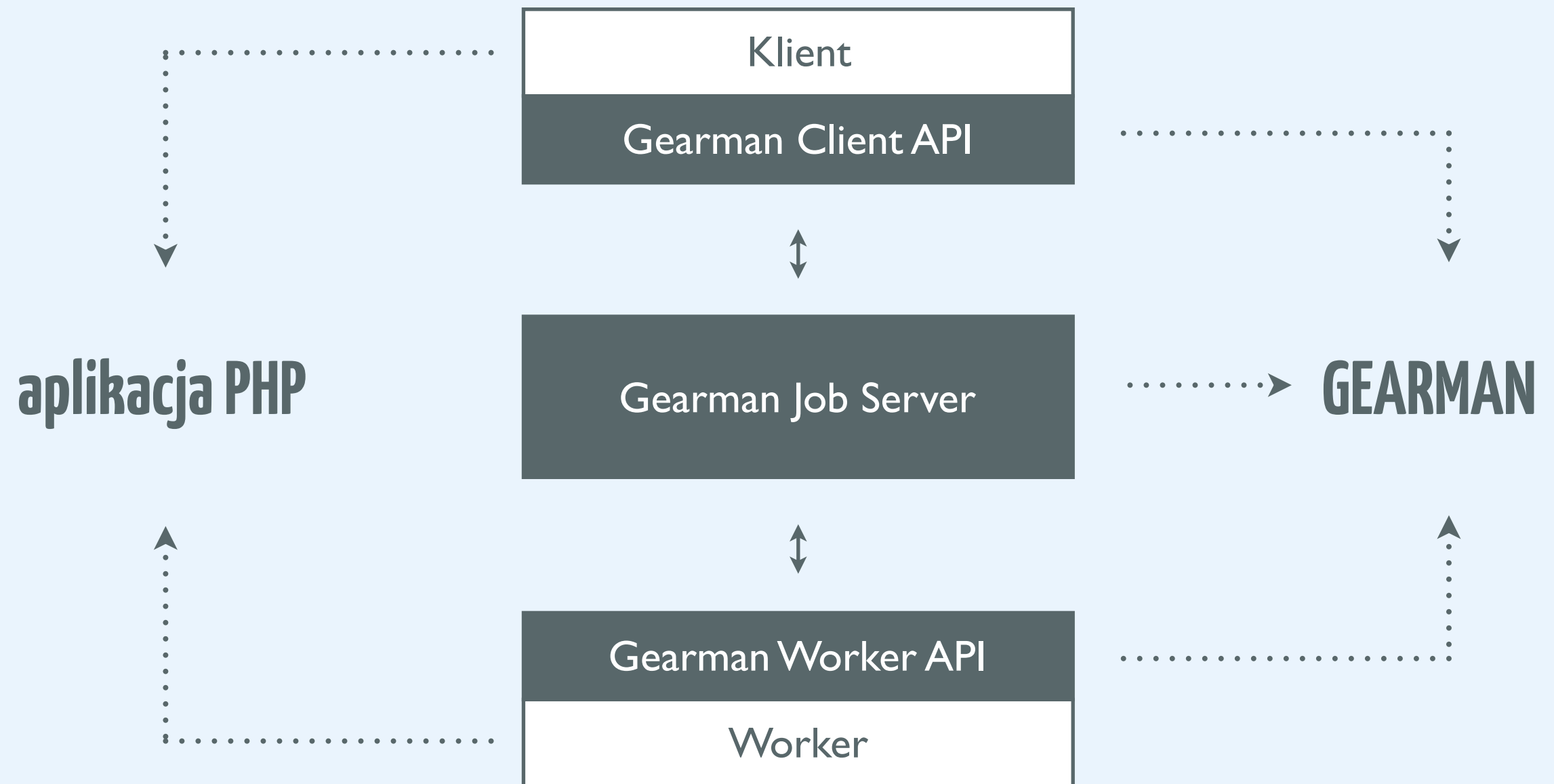
$pheanstalk = new Pheanstalk('127.0.0.1');

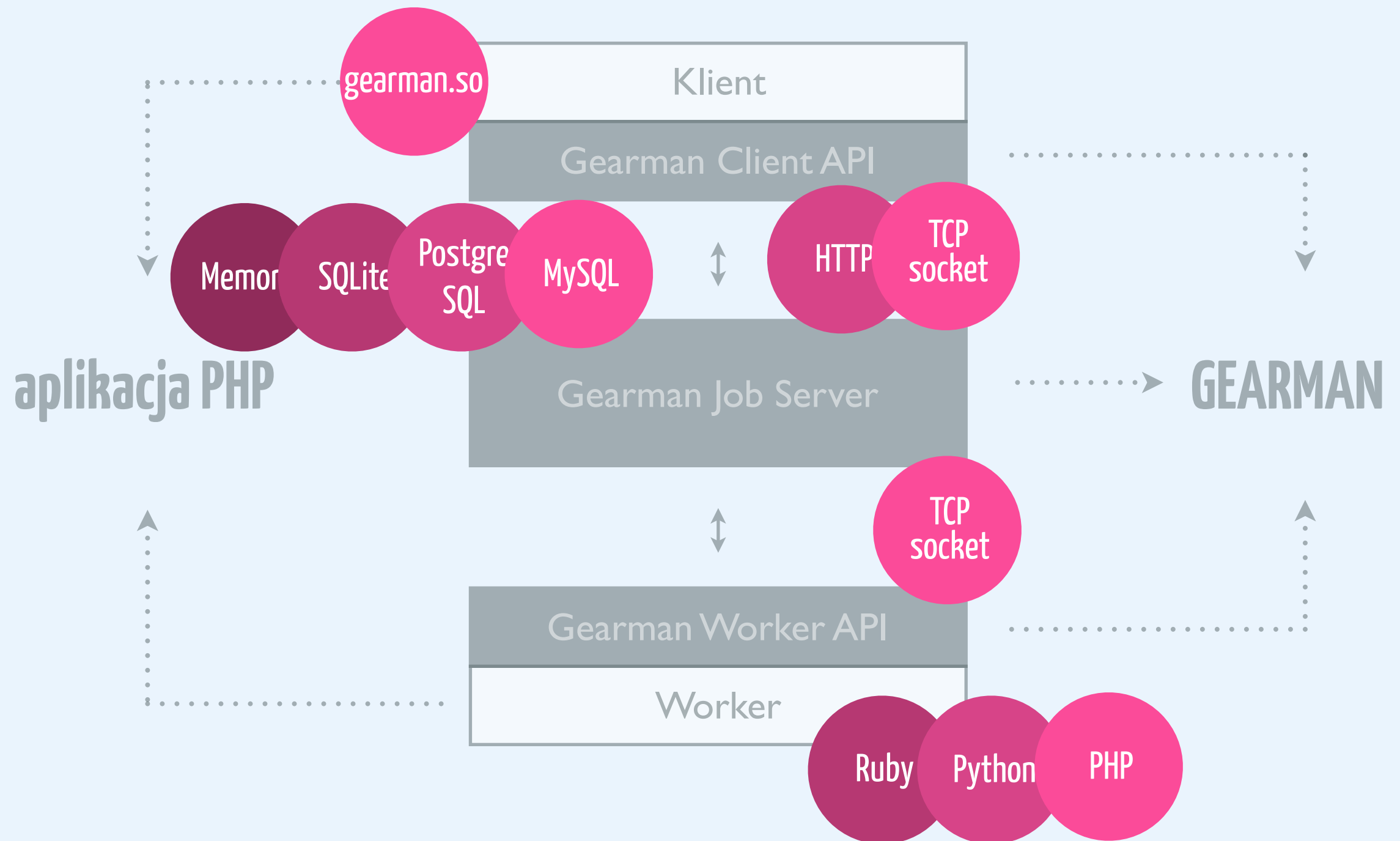
$pheanstalk
    ->useTube('testtube')
    ->put("job payload goes here\n");
```

GEARMAND

TASK MANAGEMENT FRAMEWORK

ROZWIĄZANIE 4





WORKER

PRZETWORZENIE ZADANIA

```
<?php
$worker= new GearmanWorker();

$worker->addServer();
$worker->addFunction("reverse", "my_reverse_function");

while ($worker->work());

function my_reverse_function($job)
{
    return strrev($job->workload());
}
?>
```

.....

KLIENT

REJESTRACJA ZADANIA

```
<?php

$client= new GearmanClient();
$client->addServer();

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


2000 WORKERÓW

INSTAGRAM CASE STUDY

AMAZON SQS

KOLEJKA ZADAŃ W CHMURZE

ROZWIĄZANIE 5

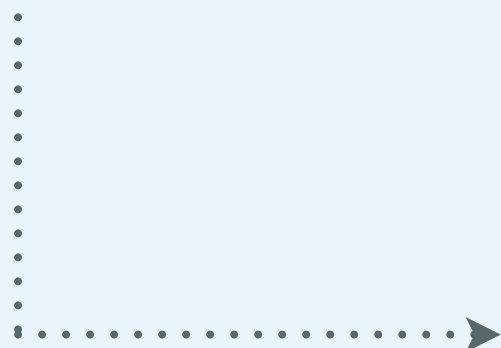
TANIE

BEZPIECZNE

SKALOWALNE

PROSTE

NIEZAWODNE*



\$0.01 za 10,000 requestów do Amazon SQS



PROSTYCH API

CreateQueue
SendMessage
ReceiveMessage
ChangeMessageVisibility
DeleteMessage

WORKER

PRZETWORZENIE ZADANIA

KLIENT

REJESTRACJA ZADANIA

```
<?php
require_once ( 'Amazon/SQS/Model/ReceiveMessageRequest.php' );

$request = new Amazon_SQS_Model_ReceiveMessageRequest();
$request->setQueueUrl( 'queue URL from CreateQueue call' );

invokeReceiveMessage( $service, $request );
```

.....

```
<?php
require_once ( 'Amazon/SQS/Model/SendMessageRequest.php' );

$request = new Amazon_SQS_Model_SendMessageRequest();
$request->setQueueUrl( 'queue URL from CreateQueue call' );
$request->setMessageBody( 'This is my message text.' );

invokeSendMessage( $service, $request );
```

RESQUE

PHP PORT

ROZWIĄZANIE 6



github

SOCIAL CODE HOSTING
WORKERS SYSTEM

BUDOWANIE TARBALLI
BUDOWANIE RUBYGEMS
URUCHAMIANIE WEB-HOOKS
USUWANIE UŻYTKOWNIKÓW
AKTUALIZACJA INDEXÓW
...i WIELE INNYCH

DEFINICJA

KLASY ZADANIA

KLIENT

REJESTRACJA ZADANIA

```
<?php
class My_Job
{
    public function setUp()
    {
        // ... Set up environment for this job
    }

    public function perform()
    {
        // .. Run job
    }

    public function tearDown()
    {
        // ... Remove environment for this job
    }
}
```

.....

```
<?php
require_once 'lib/Resque.php';

// Required if redis is located elsewhere
Resque::setBackend('localhost:6379');

$args = array(
    'name' => 'Chris'
);

Resque::enqueue('default', 'My_Job', $args);
```


STANY ZADAŃ

ŚLEDZENIE PRZETWARZANIA

```
<?php
$token = Resque::enqueue('default', 'My_Job', $args, true);

echo $token;

$status = new Resque_Job_Status($token);

echo $status->get();

switch ($status->get()) {
    case Resque_Job_Status::STATUS_WAITING:
        // ...
        break;
    case Resque_Job_Status::STATUS_RUNNING:
        // ...
        break;
    case Resque_Job_Status::STATUS_FAILED:
        // ...
        break;
    case Resque_Job_Status::STATUS_COMPLETE:
        // ...
        break;
    default:
        // ...
}
```

HOOKS

REJESTRACJA OBSŁUGI EVENTÓW

```
<?php
Resque_Event::listen(
    'onFailure',
    function($exception, $resqueJob) {
        echo sprintf(
            "\n Join has failed with exception %s \n\n",
            $exception->getMessage()
        );
    }
);

Resque_Event::listen(
    'beforeFork',
    function($resqueWorker) {
        // ...
    }
);

Resque_Event::listen(
    'beforePerform',
    function($resqueJob) {
        // ...
    }
);

Resque_Event::listen(
    'afterEnqueue',
    function($class, $arguments) {
        // ...
    }
);
```

FRONT-END WWW

OUT OF THE BOX

The image displays two overlapping browser windows showing the Resque web interface. The background window shows the 'Queues' overview, and the foreground window shows the 'Pending jobs on low' queue.

Background Window: Resque Overview

URL: internal.github.com/resque/overview

Navigation: Overview (selected), Working, Failed, Queues, Workers, Stats

Queues

The list below contains all the registered queues with the pending on the queue.

Name	Jobs
archive	55
critical	1
high	0
low	10
mail	21
failed	109

63 of 124 Workers Working

The list below contains all workers which are currently working.

Where
fe3.rs.github.com:9934
fe3.rs.github.com:10753
fe3.rs.github.com:10588
fe2.rs.github.com:10328
fe3.rs.github.com:11152

Foreground Window: Pending jobs on low

URL: <http://aux1.rs.github.com:9292/resque/queues/low>

Navigation: Overview, Working, Failed, Queues (selected), Workers, Stats

Filters: [critical](#) [page](#) [high](#) [low](#) (selected) [archive](#)

Pending jobs on low

Showing 0 to 20 of 9 jobs

Class	Args
GitHub::Jobs::ClearNetworkGraphCaches	[326476]
GitHub::Jobs::UpdateServerInfo	[326476]
Graphite	["4momsonline", "4moms"]
GitHub::Jobs::ClearNetworkGraphCaches	[193131]
GitHub::Jobs::UpdateServerInfo	[193131]
GitHub::Jobs::ClearNetworkGraphCaches	[50179]
Graphite	["postage", "theworkinggroup"]
GitHub::Jobs::UpdateServerInfo	[50179]
Graphite	["zendesk", "zendesk"]

Powered by [Resque](#) v1.1.0
Connected to Redis on db1:6379

Bottom window snippet:

Priority	Class	Args	Status
LOW	GitHub::Jobs::UpdateNetworkGraph		just now
LOW	GitHub::Jobs::UpdateNetworkGraph		just now

memcachedb.org/memcacheq
gearman.info
gearman.org
gearmanhq.com
kr.github.com/beanstalkd
github.com/pda/pheanstalk
aws.amazon.com/sqs
github.com/chrisboulton/php-resque
q4m.github.com
www.rabbitmq.com
www.amqp.org
activemq.apache.org

LINKI NA PÓZNIEJ

DEMO

http://tiny.cc/4dev_djp



PYTANIA?

.....

e: mariusz.gil@scaleup.pl
t: @mariuszgil