Gearman

OSCON 2009

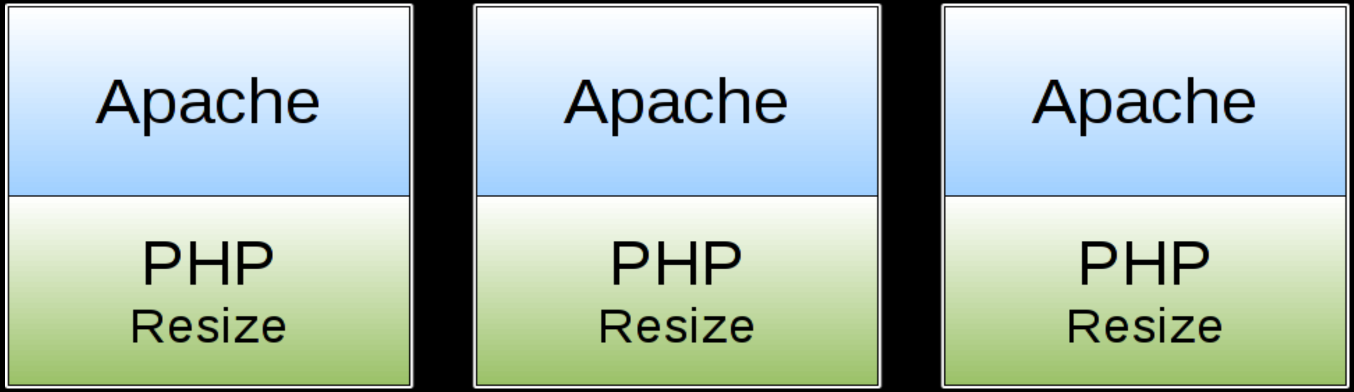
Eric Day – Sun Microsystems
http://oddments.org/

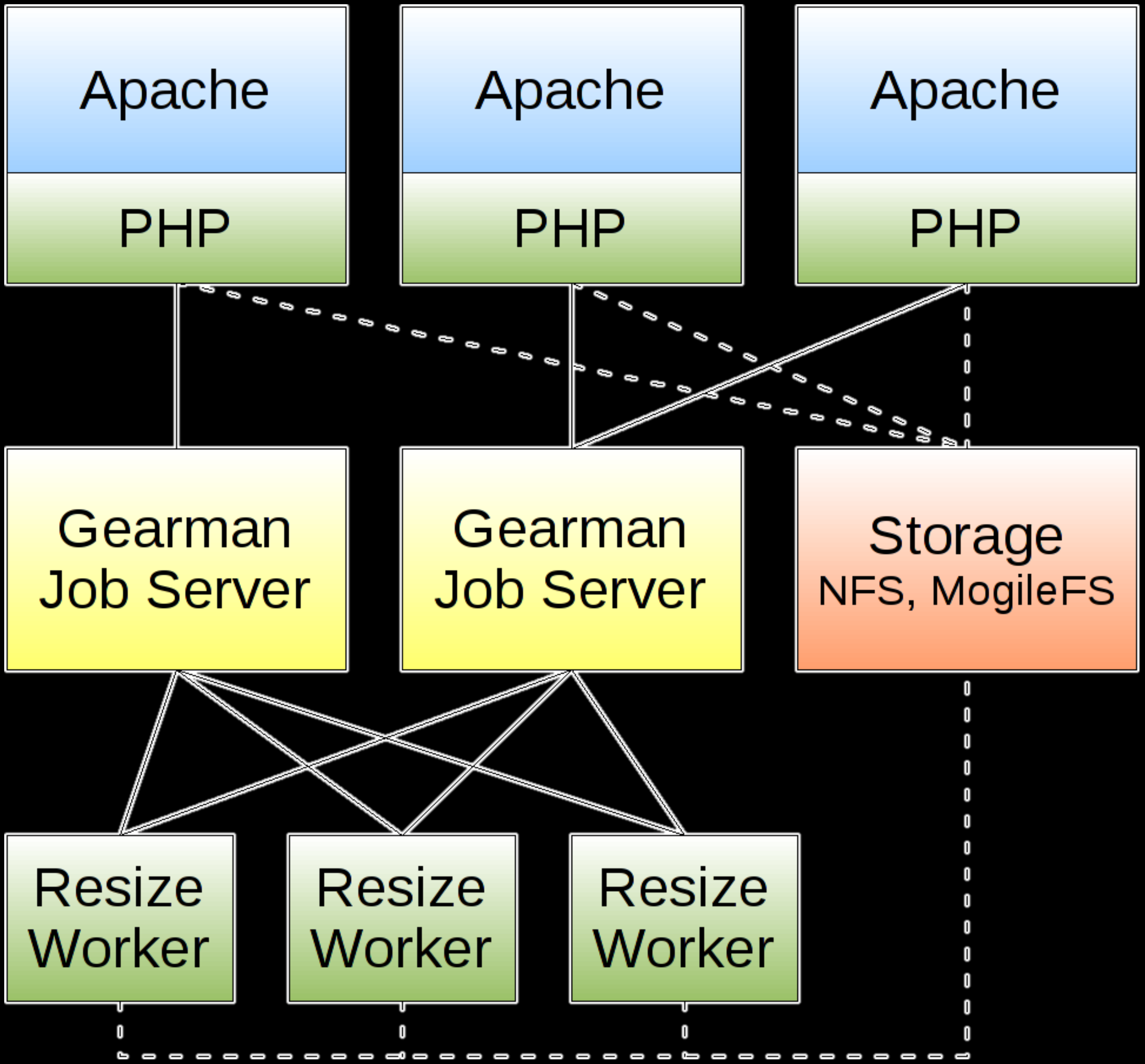
Brian Aker – Sun Microsystems
http://krow.net/

Gearman Overview

- History
- Basics
- Example
- Job Server
- Map/Reduce
- Log Analysis
- Asynchronous Queues
- Narada
- Roadmap

IM IN UR COMPUTER
EATIN UR BYTES
COMPAQ


Apache
PHP
Resize
Apache
PHP
Resize
Apache
PHP
Resize


Apache
PHP
Apache
PHP
Apache
PHP
Gearman Job Server
Gearman Job Server
Storage
NFS, MogileFS
Resize Worker
Resize Worker
Resize Worker

“The way I like to think of Gearman is as a massively distributed, massively fault tolerant fork mechanism.”

- Joe Stump, Digg

History

- Danga – Brad Fitzpatrick & company
  - Related to memcached, MogileFS, ...
- Anagram for “manager”
  - Gearman, like managers, assign the tasks but do none of the real work themselves
- Digg: 45+ servers, 400K jobs/day
- Yahoo: 60+ servers, 6M jobs/day
- LiveJournal, SixApart, DealNews, xing.com, ...

Recent Development

- Rewrite in C
- New language APIs
  - PHP, Perl, Java, Drizzle, MySQL, PostgreSQL
- Command line tool
- Protocol Additions
- Multi-threaded (50k jobs/second)
- Persistent queues
- Pluggable protocol

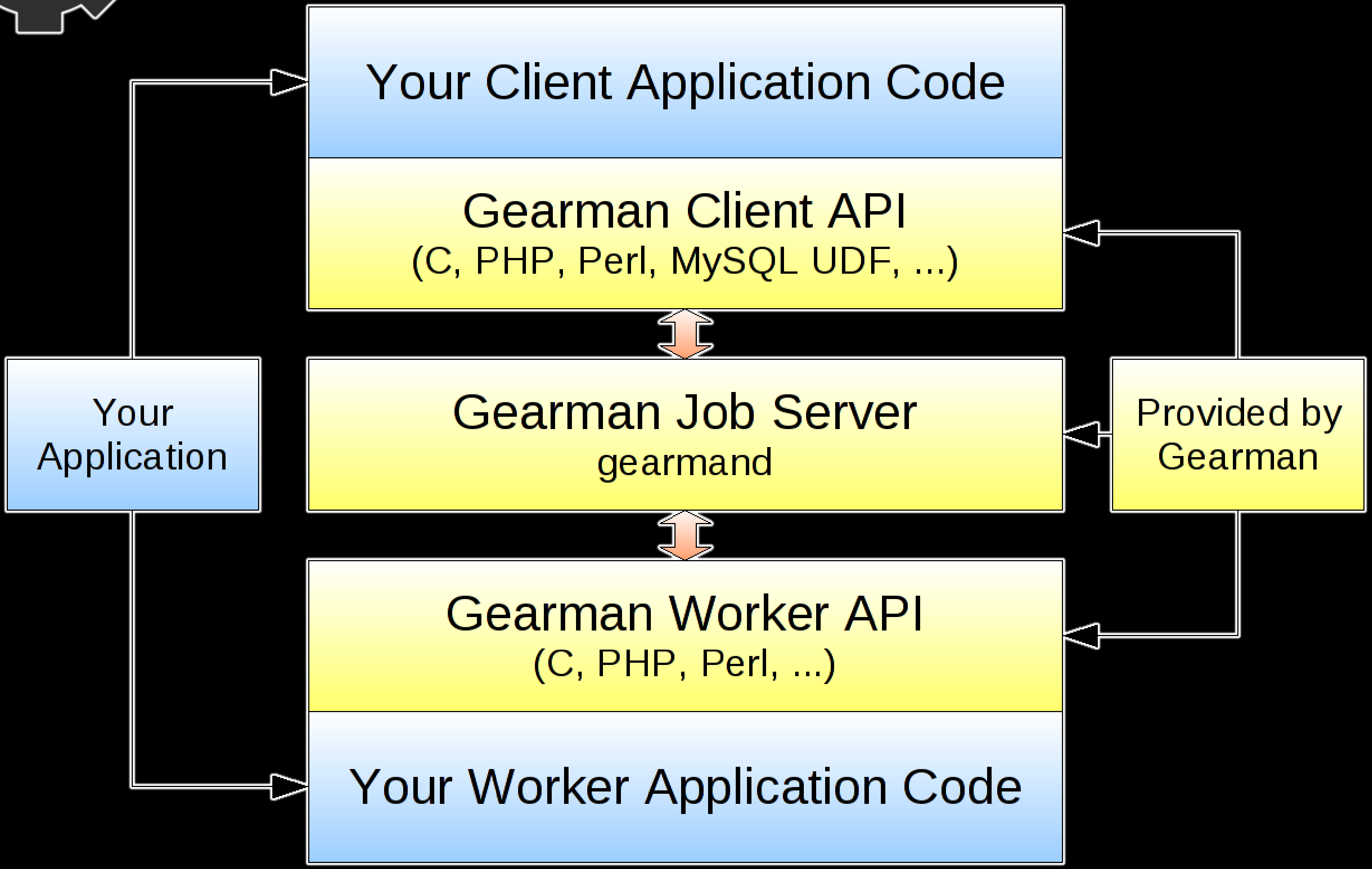
Features

- Open Source (mostly BSD)
- Simple & Fast
- Multi-language
  - Mix clients and workers from different APIs
- Flexible Application Design
  - Not restricted to a single distributed model
- Embeddable
  - Small & lightweight for applications of all sizes
- No Single Point of Failure

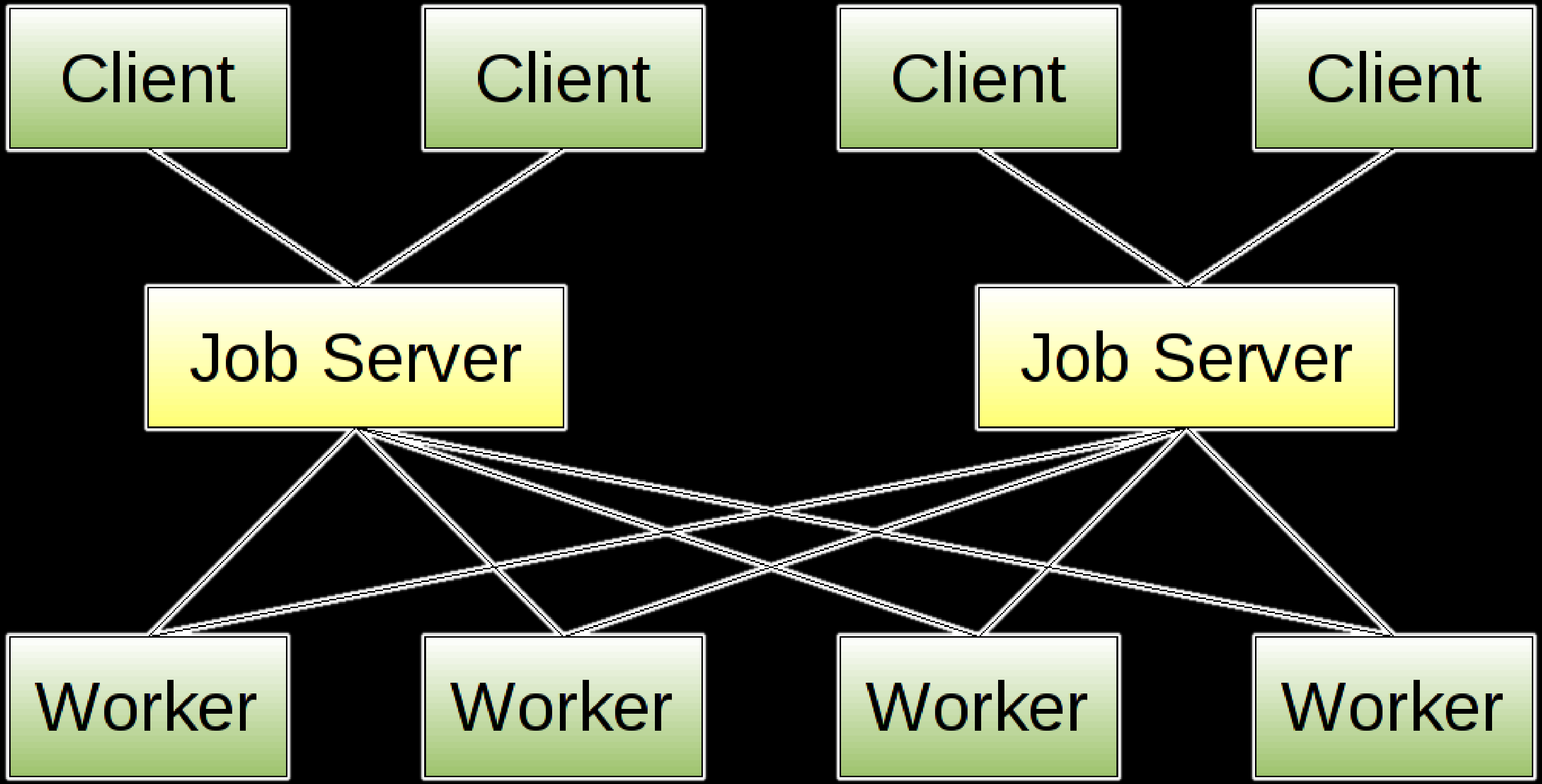
Basics

- Gearman provides a distributed application framework
- Uses TCP port 4730 (was port 7003)
- **Client** – Create jobs to be run and send them to a job server
- **Worker** – Register with a job server and grab jobs to run
- **Job Server** – Coordinate the assignment from clients to workers, handle restarts

Gearman Stack



No Single Point of Failure



Hello World

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

```
$worker= new GearmanWorker();
$worker->addServer();
$worker->addFunction("reverse", "my_reverse_function");
while ($worker->work());

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

Hello World

```
shell$ gearmand -d

shell$ php worker.php &
[1] 17510

shell$ php client.php
!dlroW olleH
```

How Is This Useful?

- Provides a distributed nervous system
- Natural load balancing
  - Workers are notified and ask for work, not forced
- Multi-language integration
- Distribute processing
  - Possibly closer to data
- Synchronous and asynchronous queues

Back to the Kittens

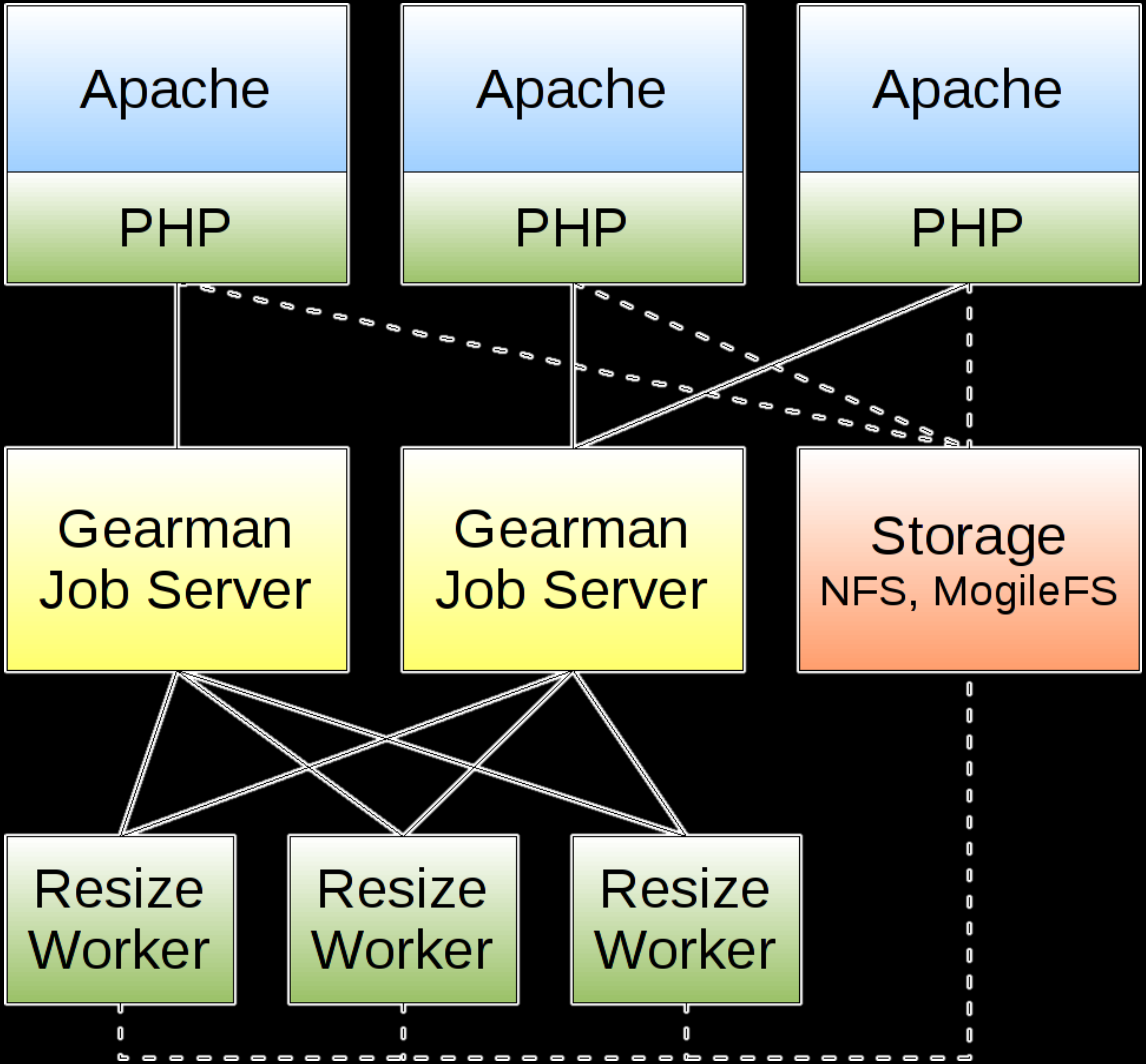

Apache
PHP
Apache
PHP
Apache
PHP
Gearman Job Server
Gearman Job Server
Storage
NFS, MogileFS
Resize Worker
Resize Worker
Resize Worker

Image Resize Worker

```
$worker= new GearmanWorker();
$worker->addServer();
$worker->addFunction("resize", "my_resize_function");
while ($worker->work());

function my_resize_function($job)
{
  $thumb = new Imagick();
  $thumb->readImageBlob($job->workload());
  $thumb->scaleImage(200, 150);
  return $thumb->getImageBlob();
}
```

Image Resize Worker

```
shell$ gearmand -d

shell$ php resize.php &
[1] 17524

shell$ gearman -f resize < large.jpg > thumb.jpg

shell$ ls -sh large.jpg thumb.jpg
3.0M large.jpg   32K thumb.jpg
```

Command Line Tool

- gearman
  - Included in C server and library package
  - Command line and shell script interface
- Client mode
  - ls | gearman -f function
  - gearman -f function < file
  - gearman -f function "some data"
- Worker mode
  - gearman -w -f function -- wc -l
  - gearman -w -f function ./script.sh

Command Line Tool

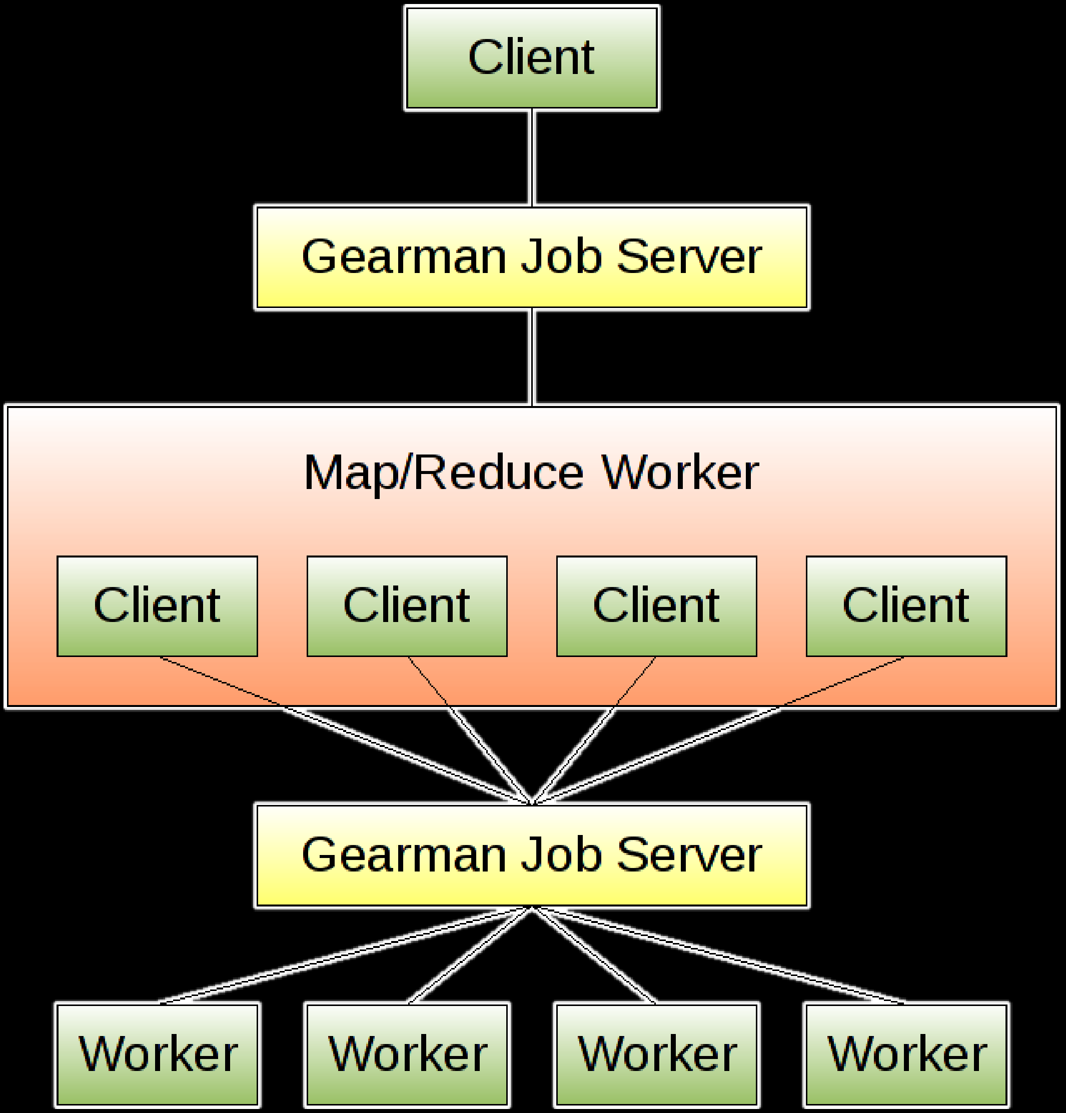
```
shell$ gearmand -d

shell$ gearman -w -f test -- grep lib &
[1] 17524

shell$ ls / | gearman -f test
lib
lib32
lib64
```

Applications

Map/Reduce


Client
Gearman Job Server
Map/Reduce Worker
Client
Client
Client
Client
Gearman Job Server
Worker
Worker
Worker
Worker

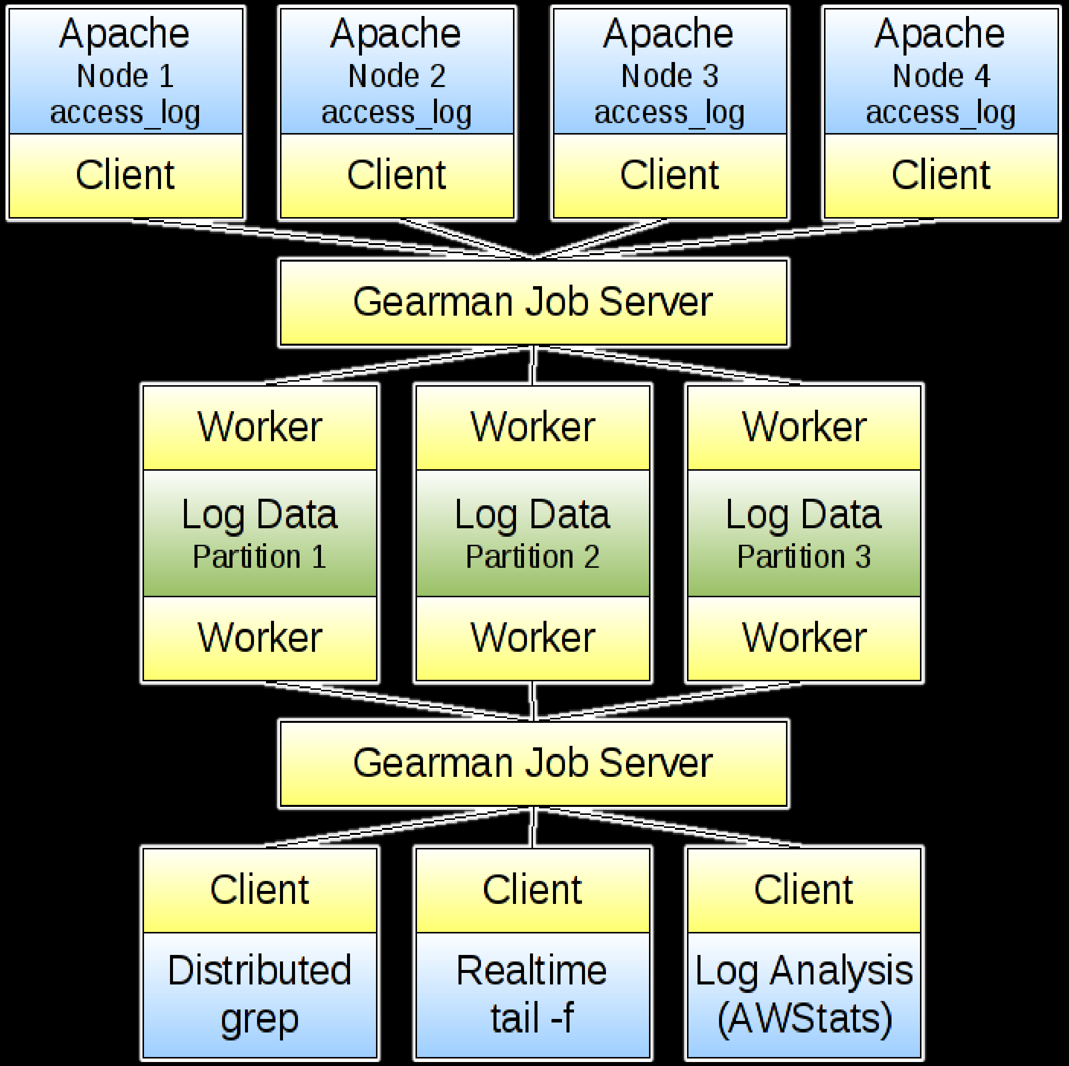
Log Processing

- Bring Map/Reduce to Apache logs
- Get log storage off Apache nodes
- Push processing to log storage nodes
- Combine data in some meaningful way
  - Summary
  - Distributed merge-sort algorithms

Log Processing

- Collection
  - tail -f access_log | gearman -n -f logger
  - CustomLog "| gearman -n -f logger" common
  - Write a Gearman Apache logging module
- Processing
  - Distributed/parallel grep
  - Log Analysis (AWStats, Webalizer, ...)
  - Custom data mining & click analysis

Log Processing



Asynchronous Queues

- Background Tasks
- They help you scale
- Distributed data storage
  - Eventually consistent data models
  - Choose “AP” in “CAP”
    - Consistency
    - Availability
    - Partitions (tolerance to network partitions)
  - Make eventual consistency work
  - Conflict resolution if needed

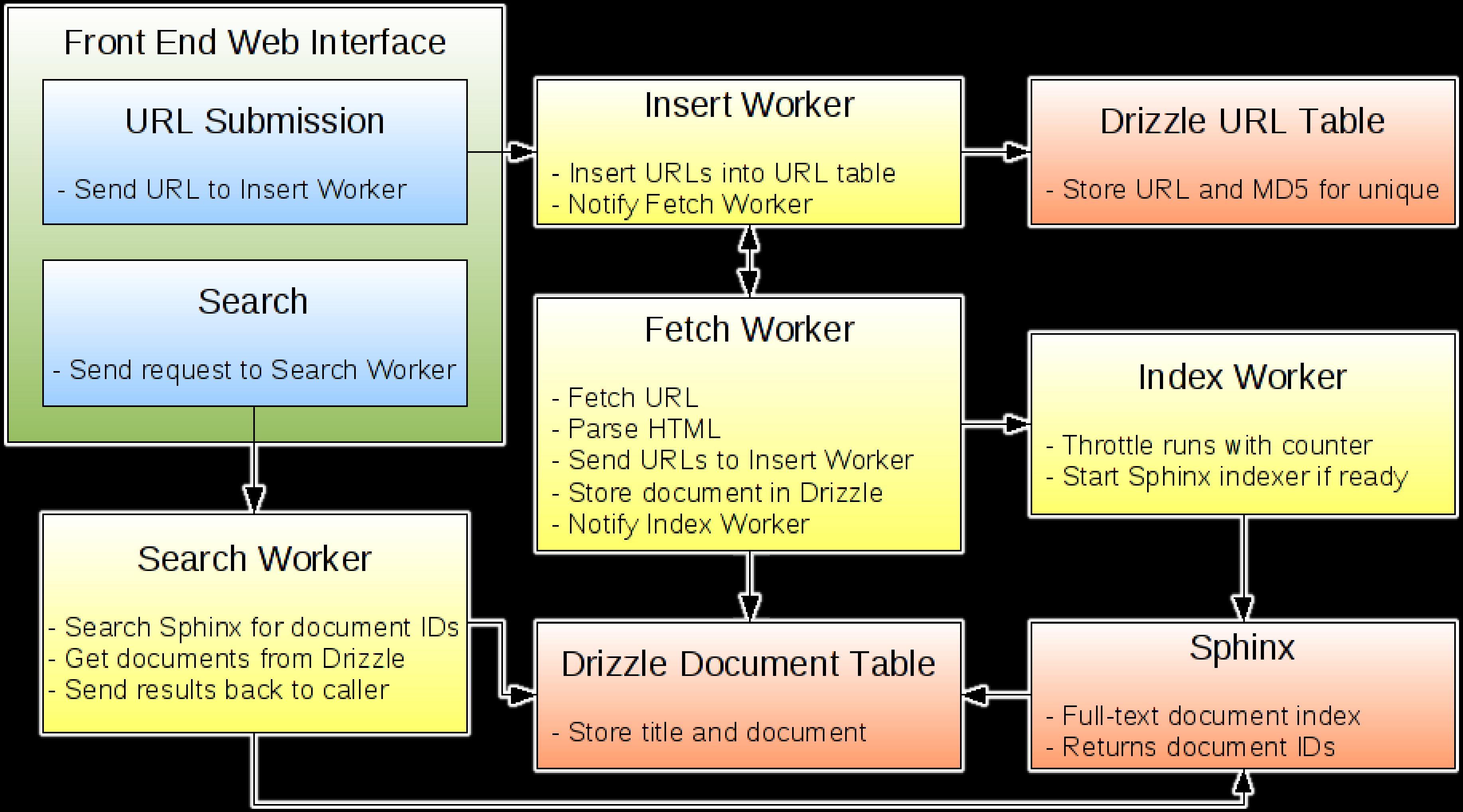
Asynchronous Queues

- Not everything needs immediate action
  - E-Mail notifications
  - Tweets
  - Certain types of database updates
  - RSS aggregation
  - Search indexing
- Allows for batch operations

Narada

- Example in Patrick Galbraith's book
- Custom search engine
- Perl, PHP, and Java implementations
- Asynchronous queues
- Drizzle or MySQL
- Optionally use memcached
- Easy to integrate into existing projects
- https://launchpad.net/narada

Narada



Front End Web Interface
URL Submission
- Send URL to Insert Worker
Search
- Send request to Search Worker
Insert Worker
- Insert URLs into URL table
- Notify Fetch Worker
Drizzle URL Table
- Store URL and MD5 for unique
Fetch Worker
- Fetch URL
- Parse HTML
- Send URLs to Insert Worker
- Store document in Drizzle
- Notify Index Worker
Index Worker
- Throttle runs with counter
- Start Sphinx indexer if ready
Search Worker
- Search Sphinx for document IDs
- Get documents from Drizzle
- Send results back to caller
Drizzle Document Table
- Store title and document
Sphinx
- Full-text document index
- Returns document IDs

Other Applications

- MogileFS
- Distributed e-mail storage
- Gearman Monitor Project
  - Configuration management (elastic)
  - Statistics gathering
  - Monitoring
  - Modular (integrate existing tools)
- What will you build?

What's Next?

- More protocol and ~~queue modules~~
- TLS, SASL, multi-tenancy
- Replication/subscription/job relay
- Job result cache (think memcached)
- Improved statistics gathering and reporting
- Event notification hooks
- Monitor service

Get involved

- http://gearman.org/
- #gearman on irc.freenode.net
- http://groups.google.com/group/gearman
- Gearman @ OSCON
  - Birds of a Feather (BoF) – Tonight @ 7PM
  - Expo Hall Booth