

heroku

Introducing **Procfile** – a better way to define and scale web apps



Oren Teich
OSCON 2011

Demo app

A **running** web app is a
collection of **processes**

Create a new process (running a program):

```
$ ./memcached -vv  
<4 server listening  
<5 server listening  
...
```

(...and stopping/restarting it with signals such as
SIGHUP, SIGINT, SIGKILL etc)

Manage processes in production systems with upstart, launchd, etc:

```
/etc/init/memcached.conf
```

```
description "Memcached"  
exec /usr/bin/memcached >> /var/log/memcached.log  
start on runlevel [345]  
respawn
```

```
$ start memcached
```

A **running** web app is a
collection of **processes**

A typical Rails app

```
bundle exec thin start  
bundle exec rake jobs:work
```

This is language agnostic. Same thing in Django:

```
python manage.py runserver  
celeryd --loglevel=INFO
```

We need a **declarative** way to define these processes, so they can be **easily managed in unison**

Procfile!

Format:

```
<process type>: <command to run it>
```

Example app:

```
web: bundle exec thin start -p $PORT -e $RACK_ENV
```

```
worker: bundle exec rake jobs:work
```

```
clock: bundle exec clockwork clock.rb
```

Local Development

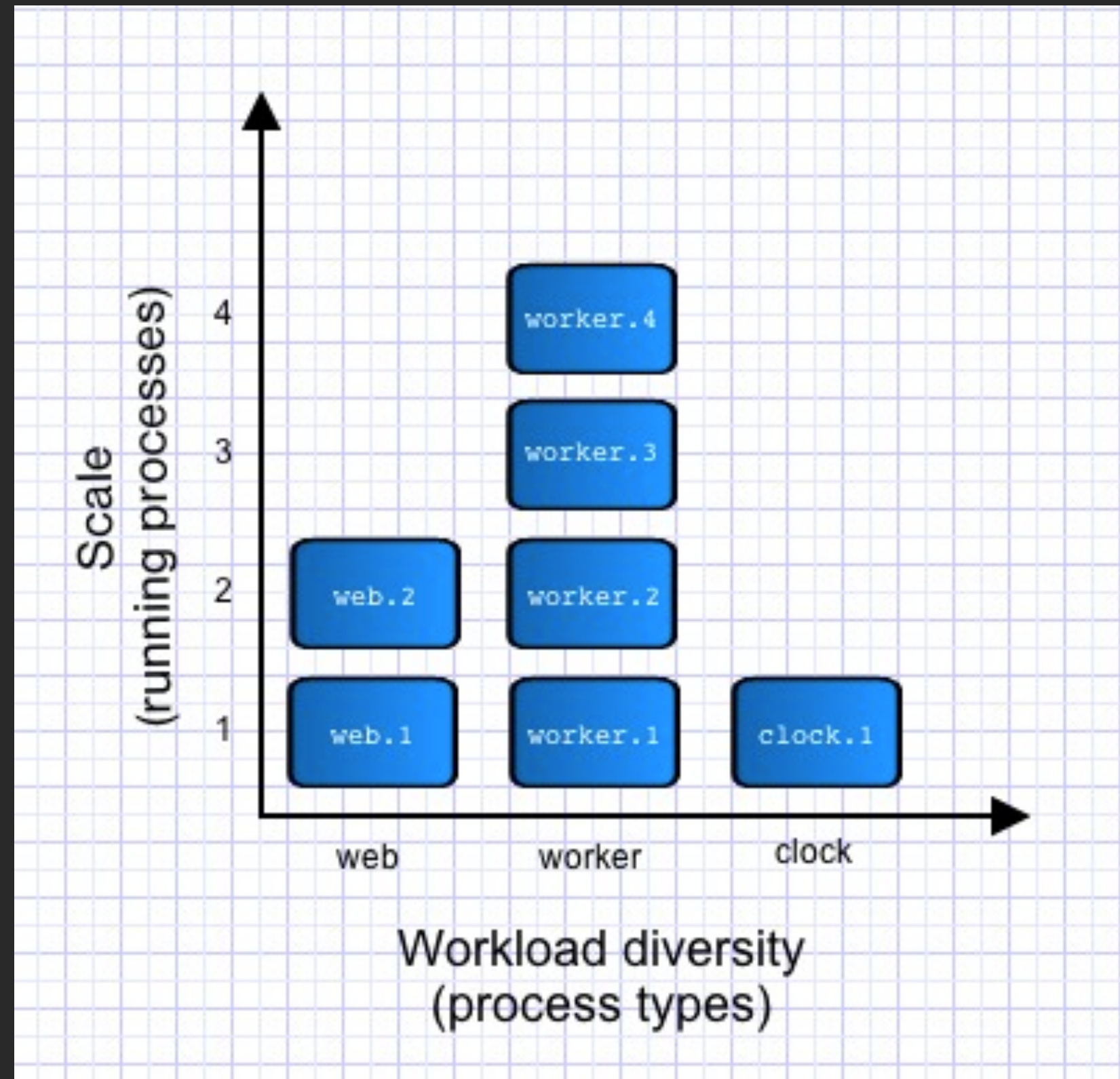
github.com/ddollar/foreman

```
$ foreman start
20:36:47 web.1      | started with pid 62628
20:36:47 worker.1   | started with pid 62629
20:36:47 clock.1    | started with pid 62632
20:36:50 clock.1    | Starting clock for 1 events: [ print.time ]
20:36:50 clock.1    | Triggering print.time
20:36:50 clock.1    | clock process is running
20:36:50 worker.1   | (in /Users/oren/code/pusher-example)
20:36:50 web.1      | >> Using rack adapter
20:36:51 web.1      | >> Thin web server (v1.2.11 20:36:51 web.1      | >>
Listening on 0.0.0.0:5000, CTRL+C to stop
```

Deploying with Foreman

```
$ foreman export upstart ~/init
[foreman export] writing: /Users/oren/init/pusher-example.conf
[foreman export] writing: /Users/oren/init/pusher-example-web.conf
[foreman export] writing: /Users/oren/init/pusher-example-web-1.conf
[foreman export] writing: /Users/oren/init/pusher-example-worker.conf
[foreman export] writing: /Users/oren/init/pusher-example-worker-1.conf
[foreman export] writing: /Users/oren/init/pusher-example-clock.conf
[foreman export] writing: /Users/oren/init/pusher-example-clock-1.conf
```

Scaling



Concurrency with Foreman

```
$ foreman start -c web=4 worker=2
11:20:33 web.1      | started with pid 80707
11:20:33 web.2      | started with pid 80709
11:20:33 web.3      | started with pid 80710
11:20:33 web.4      | started with pid 80713
11:20:33 worker.1   | started with pid 80714
11:20:33 clock.1    | started with pid 80717
```



Built in Procfile support:

- Create processes on deploy
- Automatic process management
- OS-level process isolation w LXC
- Easy scale at runtime & on-demand

Demo Heroku



Thanks!

@teich

