My name is Trinidad

David Calavera
@calavera
Farmhouse conf!
Why?
Rails
<plugin>
  <artifactId>exec-maven-plugin</artifactId>
  <executions>
    <execution>
      <id>create-mock-web-descriptor</id>
      <phase>compile</phase>
    </execution>
  </executions>
</plugin>
class UsersController < ApplicationController
  include_spring_bean :users_service

  def authorized?(user, action)
    @users_service.authorized?(user, action)
  end
end
class User < HibernateModel
  def self.find(id)
    hibernate.find_by_id(HUser.class, id)
  end
end
development

Mongrel

staging / production

Tomcat
development

Jetty-Rails

staging / production

Tomcat
what about tomcat?
development

Tomcat-Rails

staging / production

Tomcat
gem install trinidad

cd my_rails_application
ejruby -S trinidad
cd my_threadsafe_rails_application
jruby -S trinidad -t
cd my_rack_application
jruby -S trinidad -r
cd my_application

jruby -S trinidad
conventions

lib/ spring.jar
    hibernate.jar
    junit.jar
conventions

classes/  Context.class
         FooBar.class
         BarBaz.class
conventions

config/  web.xml
        trinidad.yml
secure layer

---

```plaintext
ssl:
  port: 3443
  keystore: keystore.p12
...
```
multiple apps

---

web_apps:
  app1:
    context_path: /foo
  app2:
    rackup: config.ru
multiple apps

$ jruby -S trinidad --apps webapps

---

apps_base: webapps
extensible
gem install trinidad_FOO_extension

---

technologies:

  FOO:

gem install

trinidad_FOO_extension

trinidad --load FOO
module Trinidad::Extensions
  class F00ServerExtension < ServerExtension
    def configure(tomcat)
      ...
    end
  end
end
module Trinidad::Extensions
  class FOOWebAppExtension < WebAppExtension
    def configure(tomcat, app_context)
      ...
    end
  end
end
module Trinidad::Extensions
  class F000OptionsExtension < OptionsExtension
    def configure(parser, default_options)
      ...
    end
  end
end
daemon
daemon

---

extensions:
daemon:
pid_file: tmp/trinidad.pid
jvm_args: -Xmx20Gb
<table>
<thead>
<tr>
<th>/neo4j-rails-example</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>jruby.compat.version =&gt; 1.8.7</td>
<td>jruby.min.runtimes =&gt; 1</td>
</tr>
<tr>
<td>jruby.initial.runtimes =&gt; 1</td>
<td>public.root =&gt; /public</td>
</tr>
<tr>
<td>jruby.max.runtimes =&gt; 1</td>
<td>rails.env =&gt; production</td>
</tr>
<tr>
<td>rails.root =&gt; /</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>/rails3_test</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>jruby.compat.version =&gt; 1.8.7</td>
<td>jruby.min.runtimes =&gt; 1</td>
</tr>
<tr>
<td>jruby.initial.runtimes =&gt; 1</td>
<td>public.root =&gt; /public</td>
</tr>
<tr>
<td>jruby.max.runtimes =&gt; 1</td>
<td>rails.env =&gt; production</td>
</tr>
<tr>
<td>rails.root =&gt; /</td>
<td></td>
</tr>
</tbody>
</table>
sandbox

$ curl http://example.com/sandbox/apps.xml

<applications>
  <application>
    <name>default</name>
    <link href=".../default/start.xml" rel="start" />
    <link href=".../default/stop.xml" rel="stop" />
    <link href=".../default/restart.xml" rel="restart" />
  </application>
</applications>
opts = {
    :context_path => '/sandbox',
    :web_app_dir => 'sandbox',
    :public => 'app/public',
    :rackup => 'config.ru'
}
sandbox howto

```ruby
web_app = Trinidad::RackupWebApp.new({}, opts)

app_ctx = tomcat.add_webapp(
  opts[:context_path], [:web_app_dir])

app_ctx.add_lifecycle_listener(
  Trinidad::Lifecycle::Default.new(web_app))
```
hot deploy
hot deploy

---

monitor: tmp/restart.txt
“hi David, quick question about hot deploy... “
“... it would be great if we could have real zero down time “
Trinidad Zero
class ScheduledJob < TrinidadScheduler.Cron
  "0/5 * * * * ?"
  def run
    _logger.info "I'm printed every 5 seconds"
  end
end
class ScheduledJob < TrinidadScheduler.Simple
    :start => Time.now, :end => Time.now + 240

    def run
        _logger.info "I’m printed every 5 seconds"
    end
end
TrinidadScheduler.run_later do # 3 seconds
  _logger.info "I’m inside this block"
end

TrinidadScheduler.run_later(:delay => 20) do
  _logger.info "I’m inside this block"
end
lifecycle

... n m

evolution
module Trinidad::Lifecycle::WebApp
class MyListener
  ...
  def lifecycle_event(event)
    if BEFORE_START_EVENT == event.type
      # do something before the app starts
lifecycle

when Lifecycle::START_EVENT
when Lifecycle::BEFORE_START_EVENT
when Lifecycle::AFTER_START_EVENT
when Lifecycle::STOP_EVENT
when Lifecycle::BEFORE_STOP_EVENT
when Lifecycle::AFTER_STOP_EVENT
when Lifecycle::PERIODIC_EVENT
Queues

The list below contains all the registered queues with the number of jobs currently in the queue. Select a queue from above to view all jobs in the queue.

<table>
<thead>
<tr>
<th>Name</th>
<th>Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>failed</td>
<td>0</td>
</tr>
</tbody>
</table>

0 of 1 Workers Working

The list below contains all workers which are currently running a job.

<table>
<thead>
<tr>
<th>Worker</th>
<th>Queue</th>
</tr>
</thead>
</table>

Nothing is happening right now.

Powered by Resque v1.15.0
Connected to Redis namespace resque on redis://127.0.0.1:6379/0
resque

```yaml
---

extensions:
  resque:
    queues: critical, high, low
    redis_host: example.com:6379
```
HELLO IT

DID YOU TRY TURNING IT OFF AND ON AGAIN?
Try Trinidad

Try how your Rails application runs on Trinidad without hassle.

This service runs on a Small Amazon EC2 instance from a temporal directory, so don't expect the best performance ever. Your application will disappear as soon as I restart the instance.

This service only clones public repositories using http or git read-only urls.

It doesn't provide any database, so make sure your application uses sqlite3 or any other embedded database.

If your application uses Bundler the service installs the required gems automatically.

1. Use the form to deploy your application or add this Post-Receive URL on Github:

   http://trytrinidad.com/sandbox/deploy?deploy_token=trytrinidad

2. Check the status of your application in the console.
next
what’s next?
better rack integration?
sandbox features?
clustering?
github.com/trinidad
thank you!
credits

http://www.flickr.com/photos/rooreynolds/243811287/
http://www.sxc.hu/photo/1192923
http://www.flickr.com/photos/streetfly_jz/2752896812
http://www.flickr.com/photos/58376723@N07/5359060059/
http://www.flickr.com/photos/cosacutro/4820222363/
http://www.flickr.com/photos/darek-zon/2457168503/
http://www.flickr.com/photos/veganstraightedge/5697792356
http://www.flickr.com/photos/veganstraightedge/5696662117
http://www.flickr.com/photos/drumminhands/4757641653/