Developing portable PaaS applications

@jclouds
First Things First

Thanks!

@jclouds
Agenda?
Agenda

• PaaS in a very small nutshell
• Introducing TweetStore
• Making TweetStore portable
• Integration Testing
• Continuous Integration & Deployment
• Portable PaaS: make sense (yet)?
PaaS in a Nutshell
PaaS in a Nutshell
PaaS in a Nutshell

@jclouds
PaaS in a Nutshell

- “Application-scoped” deliverable
- Platform handles scaling, monitoring, deployment
- App service catalogue
- Multi-language eventually
- Pluggable eventually
- Hybrid eventually
PaaS in a Nutshell

Access via library

- Messaging
- Scheduling
- Delayed Jobs
- Persistence
- Deployment

Access via console or command line API

- Runtime Status
- Monitoring
- Billing

@jclouds
Introducing TweetStore

@jclouds
Introducing TweetStore

@jclouds

A multi-cloud library, right?
jclouds
Introducing TweetStore

• jclouds demo web application
• Guice & Spring versions
• Originally written for Google App Engine
• Now also on RUN@cloud, OpenShift Express, Cloud Foundry
• In progress: Heroku Cedar, DotCloud, Amazon Elastic Beanstalk

@jclouds
# Introducing TweetStore

## Tweets in Clouds

<table>
<thead>
<tr>
<th>Tweet ID</th>
<th>Who Said it</th>
<th>Tweet</th>
</tr>
</thead>
<tbody>
<tr>
<td>163585760287662080</td>
<td>Pay4PlayAnalyst</td>
<td>The #IaaS Daily is out! <a href="http://t.co/bvUWZbi3">http://t.co/bvUWZbi3</a>? Top stories today via @markwilkinz @cjdailey @jclouds</td>
</tr>
<tr>
<td>163585760287662080</td>
<td>Pay4PlayAnalyst</td>
<td>The #IaaS Daily is out! <a href="http://t.co/bvUWZbi3">http://t.co/bvUWZbi3</a>? Top stories today via @markwilkinz @cjdailey @jclouds</td>
</tr>
<tr>
<td>163828289783610368</td>
<td>jonounabroad</td>
<td>@DrMiaow oh. People are meant to be building on @jclouds I've not used it. Believe it offers cool options.</td>
</tr>
<tr>
<td>163828289783610368</td>
<td>jonounabroad</td>
<td>@DrMiaow oh. People are meant to be building on @jclouds I've not used it. Believe it offers cool options.</td>
</tr>
<tr>
<td>163828509507190785</td>
<td>michaelneale</td>
<td>@jonounabroad @DrMiaow @jclouds for a reasonable subset, yes! AWS has become an almost dominating cloud provider, but there are many others out there now.</td>
</tr>
<tr>
<td>163828509507190785</td>
<td>michaelneale</td>
<td>@jonounabroad @DrMiaow @jclouds for a reasonable subset, yes! AWS has become an almost dominating cloud provider, but there are many others out there now.</td>
</tr>
<tr>
<td>163977261953906568</td>
<td>Rackspace</td>
<td>@jclouds Please email details to help @rackspace.com and we’ll look into that for you!</td>
</tr>
<tr>
<td>163977261953906568</td>
<td>Rackspace</td>
<td>@jclouds Please email details to help @rackspace.com and we’ll look into that for you!</td>
</tr>
<tr>
<td>164287448707379202</td>
<td>jQueryJavaBot</td>
<td>RT @m_holmqvist RT @jclouds: meet us at #fokus and learn how to deploy to the swedish #cloud</td>
</tr>
<tr>
<td>164287448707379202</td>
<td>jQueryJavaBot</td>
<td>RT @m_holmqvist RT @jclouds: meet us at #fokus and learn how to deploy to the swedish #cloud</td>
</tr>
<tr>
<td>1643121273639424</td>
<td>reillyusa</td>
<td>The Christian Reilly Daily is out! <a href="http://t.co/infhWY3f">http://t.co/infhWY3f</a>? Top stories today via @jordaanirke @pirospas @mattstine @mikebrady</td>
</tr>
</tbody>
</table>
Introducing TweetStore
Introducing TweetStore

• Using “in-application code” for persistence
• No messaging
Making TweetStore Portable

@jclouds
Making TweetStore Portable

- Different app services available
- How to separate from business logic?
Making TweetStore Portable

- Different app services available
- How to separate from business logic?
- DI layer as boundary
  - “native” for some PaaSes
- Supporting Guice & Spring
Making TweetStore Portable

- Container
- Servlets
- Controllers

@jclouds
Making TweetStore Portable

@jclouds

Now with PlatformServicesInitializer!
Making TweetStore Portable

1. Listener retrieves information from platform and instantiates \textit{PlatformServices}

2. “Application” DI layer uses \textit{PlatformServices} as a factory
Making TweetStore Portable

1. Listener retrieves information from platform and instantiates `PlatformServices`
2. “Application” DI layer uses `PlatformServices` as a factory
   - Essentially “platform” DI layer
   - Keeps platform knowledge isolated
Making TweetStore Portable

• Essentially “platform” DI layer
• Keeps platform knowledge isolated
• TODO: Create service abstraction?
• TODO: Spring namespace?
• TODO: Guice module?
Integration Testing

- One option: different app instance in the PaaS
- Cost? Access? Debugging?
- Would like to test offline
- Would like to test from build
- Would like to test from IDE
Integration Testing

• Use provided “local” emulators/servers
• Create your own emulator
• Use local test image
Integration Testing

- Use provided “local” emulators/servers
- Create your own emulator
- Use local test image
- Test support varies widely
- Arquillian PaaS containers would be nice!

@jclouds
Integration Testing

- Use provided “local” emulators/servers
- Create your own emulator
- Use local test image
- Test support varies widely
- Arquillian PaaS containers would be nice!
  hint hint ;-)

@jclouds
Continuous Integration & Deployment
Continuous Integration & Deployment

@jclouds
Continuous Integration & Deployment

- Relatively easy with deployment-via-SDK
- Maven plugins make it even nicer
- Trickier with the “git push” model if you want your source elsewhere
- Can handle with build-store-checkout-commit
Continuous Integration & Deployment

- Release management/change control still open questions
- Use deploy and post-deploy hooks?
- Pluggable PaaS service?
Portable PaaS: make sense (yet)?

• For PaaS apps generally
  • You’ll probably write it differently when writing for PaaS
  • Good for greenfield
• For existing apps, see if you can just run
Portable PaaS: make sense (yet)?

- For PaaS apps generally
  - You’ll probably write it differently when writing for PaaS
  - Good for greenfield
  - For existing apps, see if you can just run
  - “Porting to PaaS”? Hm.
Portable PaaS: make sense (yet)?

• *Portable* PaaS apps?
  • Want to avoid lock-in
  • Need multiple *instances*, not necessarily multiple platforms
  • PaaS services will standardize over time
Portable PaaS: make sense (yet)?

- **Portable** PaaS apps?
- Want to avoid lock-in
- Need multiple *instances*, not necessarily multiple platforms
- PaaS services will standardize over time
- Probably not yet.

@jclouds
More Info

- [http://www.jclouds.org](http://www.jclouds.org)
- [http://github.com/jclouds/jclouds](http://github.com/jclouds/jclouds)
- [https://github.com/jclouds/jclouds/tree/master/demos/tweetstore](https://github.com/jclouds/jclouds/tree/master/demos/tweetstore)
Come Hang Out!

  Citerus AB, Barnhusgatan 16, 111 23, 17:30
• jclouds@googlegroups.com
• @jclouds
• IRC #jclouds on freenode