springone 2GX
DALLAS
2014

Experiences using Grails in a Microservice Architecture

Jeff Beck

@beckje01

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

springone 2GX

Who Am I

- Jeff Beck
- TechLead at ReachLocal
- @beckje01 everywhere

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

What I mean by Microservice

- Single concern
- Deployable in isolation

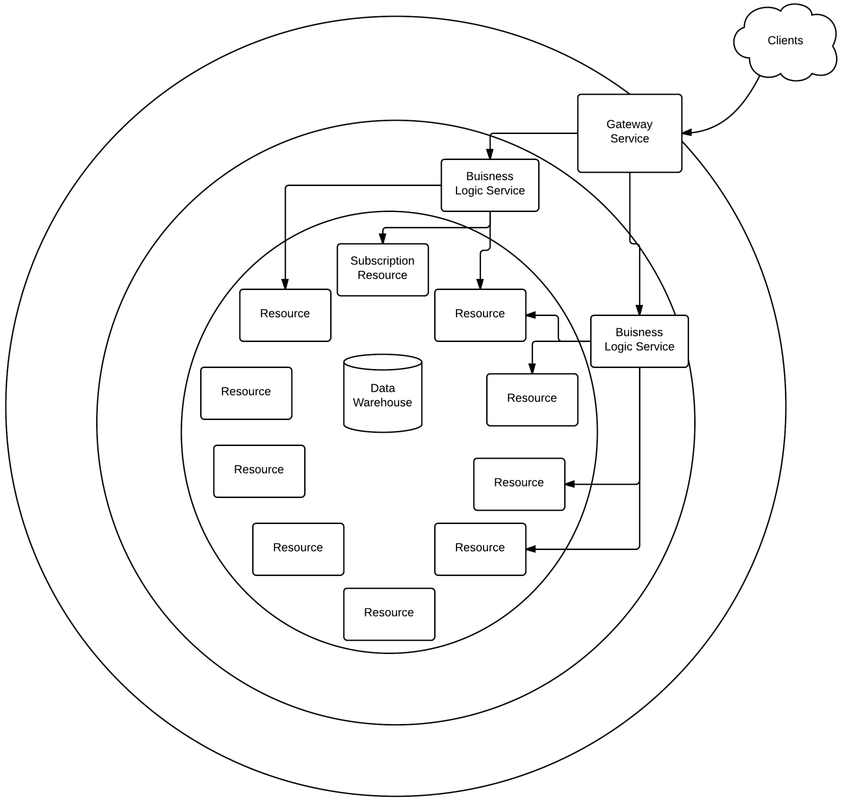
Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Our Architecture

Over the course of acquisitions and expanding the products we have come out with a polyglot architecture including Java, Groovy, Ruby, PHP, Node, and PERL. In order to take advantage of our existing talent and software, we have started down the micro-service path.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Our Architecture



Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Our Architecture

- Layered Microservices
- Asynchronous Broadcast Event System
- REST calls can be either Synchronous or Asynchronous

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Where Grails Fits

- REST Resources
- GORM

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Where Grails Doesn't Fit

- Batch Jobs
- Lacking any web endpoint

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Example Service

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Subscription Service

Acts as the source of truth about subscriptions.

- Exposes the Subscription REST Resource
- Emanates Events

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Requirements

- Application Monitoring
- Security
- Server Setup
- Builds
- Deployments

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Building Our Service

All our Grails micro services start out the same way.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Project Setup

- Grails create-app
- Change Java Version to 1.7
- Add Internal Nexus

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Project Setup - Plugins

- Add our normal plugins
    - code-coverage
    - console
    - cucumber
- Remove the extra standard plugins
    - scaffolding
    - asset-pipeline
    - jquery

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Project Setup - Plugins Config

- Set up cucumber plugin
- Configure GORM

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Build and Repo Setup

- Commit everything as the starting point
- Add the new CI job
- Bring up a dev server

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Customizing Our Service

We add scenarios, using cucumber allows us to keep our testing more DRY.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

```
Scenario: Get nonexistent subscription by ID
 Given I am a valid api client
 And A valid subscription ID which does not match any subscription
 When I request a subscription by ID
 Then I get a 404 response
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

```
Scenario: Get existing subscription by ID
 Given I am a valid api client
 And A valid subscription ID which matches a subscription
 When I request a subscription by ID
 Then I get a 200 response
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

```
Given(~'^A valid subscription ID which matches a subscription$') { ->
    //Have a valid subscription which exposes an ID
}
Given(~'^A valid subscription ID which does not match any
subscription$') { ->
    //Have an ID which is valid but no matching subscription
}
When(~'^I request a subscription by ID$') { ->
    //Do actual request
}
Then(~'^I get a (\\d+) response$') { int statusCode ->
    if (response.statusCode != statusCode) {
        println response.asString()
    }

    assert response.statusCode == statusCode
}
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

What We Reused

- Health Checks
- Security
- Server Setup
- Deployments
- SI Components

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Health Checks

To support reusable monitoring we expose a health check in a known way that attempts to be both human readable and programmatically useful.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Example 200 Status

```
{
  "dependencies": {
    "database":"OK",
    "file-access":"WARN"
  }
}
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

What Was Exposed

Via a Grails plugin we share

- Common Grails focused health check implementations
- Controller that supports the expected output

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Sharing Outside Grails

Via a JAR

- Common health check implementations
- A registry of health checks

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Security

We do server to server authentication with a token. So checking the Authorization header the plugin authenticates a client.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Our Plugin

- Expects a known GORM object that has an ID which is the token.
- Uses a static list of resource names to secure
- Intended to be as light weight as possible

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Experiences With Our Plugin

- Moving away from our custom implementation towards a SpringSecurity based Grails plugin
- Was opt in security which was easy to miss a controller
- Intently lacked roles which we have found a use-case for now

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

How We Shared The Plugins

We use an internal Nexus repo, and release plugins to that.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Experiences

- Supports Versioning
- Dependency resolution works the way the rest of Grails does
- Changes don't reload

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Guide for Internal Plugins

- Tend to adding features allowing customization.
- Each plugin is a project it needs CI, CodeNarc, etc
- Use Versioning
- CI pushing out SNAPSHOT versions is very helpful

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Server Setup

We use puppet to automate our server setup. Using classes we share default setup for a Tomcat server.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

init.pp

```
class apps_subscription_api (
  $heap_min = '256m',
  $heap_max = '1024m',
  $permgen_size = '1024m'
 ){
  class { 'standard_tomcat7_web_server':
  minimum_heap => $heap_min,
  maximum_heap => $heap_max,
  permgen_size => $permgen_size,
  }

  include apps_subscription_api::config

  base::nagios::hostgroup { "rsubscription_api_servers": }
  base::nagios::hostparam { "_healthuri": value => '/health' }
 }
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

config.pp

```
class apps_subscription_api::config {

     file { '/rl/path/configs/subscription-config.groovy':
      ensure => present,
      owner => tomcat7,
      group      => tomcat7,
      mode       => '0400',
      content    => template('apps_subscription_api/subscription-
config.groovy.erb'),
      require    => File['/rl/path/configs']
      }
    }
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Deployments

We automate our deployments via custom bash scripts kicked off by bamboo deployments.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Changes per Service

- Server List
- Artifact Name
- Deployment permissions

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Spring Integration Components

Our spring integration apps tend to need the same filters and transformers for our Events. Grails apps are not the only users of these components.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Shared via a Jar

- Exposes a Pojo that represents our internal idea of an Event
- De-duplication filter
- An Event deserializer that supports our Builder
- Built to include the least dependencies

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Logging

As a single request can spread out across the graph of microservices it is helpful to have some way to correlating all the work back together.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Correlation ID

We use Log4J's MDC to log a correlation ID throughout the requests life in Grails.

```
MDC.put('correlationId', ", CorrelationId=${correlationId}")
```

The Grails Filter

```
new EnhancedPatternLayout(conversionPattern: '
%d{ISO8601}%d{ z}{GMT+0} %-5p [%t] %c{2}(:%L) - %m
%X{correlationId}%n')
```

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Correlation ID Pitfalls

- LB calls without Correlation ID generating noisy logs
- Passing further on to other systems transparently is tricky without some standards for http clients etc
- Passing the correlation id around explicitly is messy

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Log Aggregation

We use Splunk for log aggregation for all applications.
Allowing a query across many apps for a single correlation id.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Splunk Correlation ID and Transactions

With Splunk we can use our correlationId as a transaction key, so we see log messages grouped together in spunk

```
index=my_api | transaction correlationId keepevicted=true
```

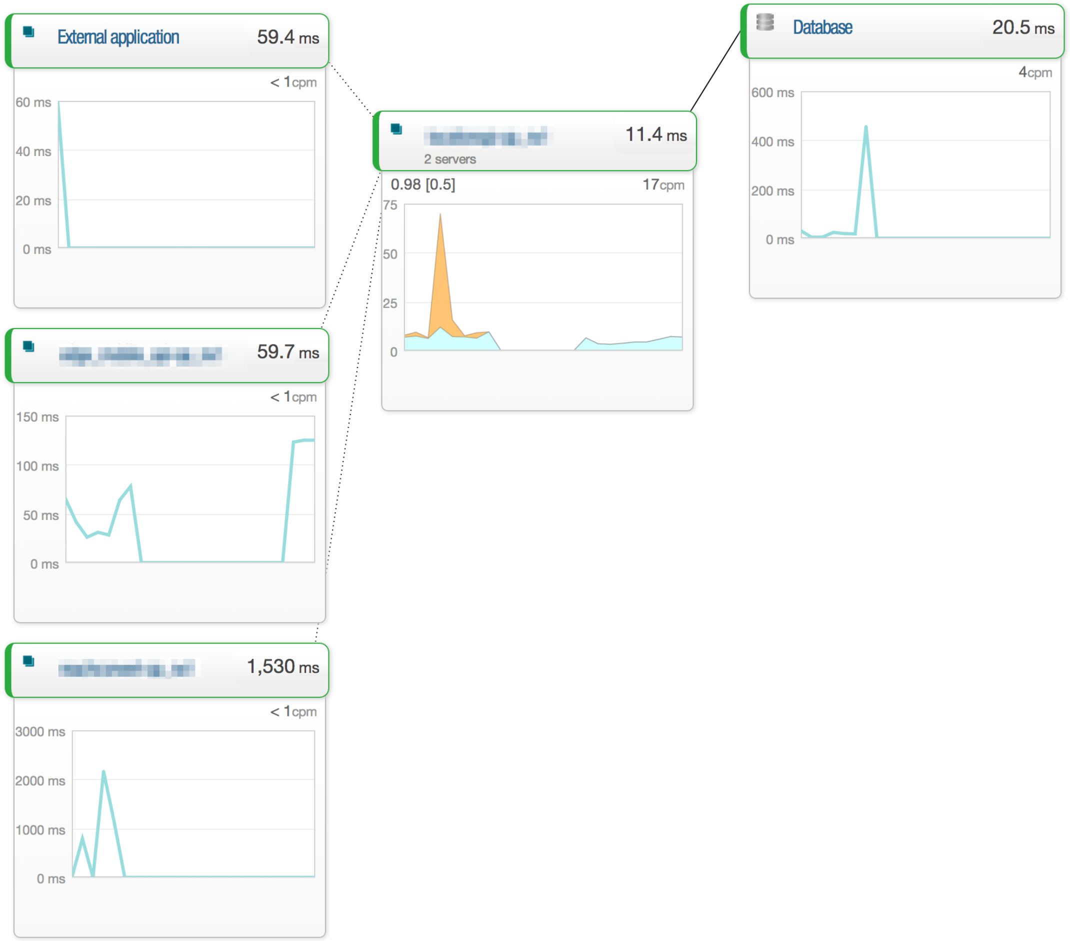
Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Centralized Monitoring

- Nagios
- NewRelic
- Starting to use dashboards of health checks

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

NewRelic Map



Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

NewRelic + Grails

Grails Service

```groovy
import com.newrelic.api.agent.Trace

class SubscriptionService {

  @Trace
  def save(Subscription subscription) {
    //Work Here
  }
```

newrelic.yml

```yaml
#enable_custom_tracing is used to allow @Trace on methods
 enable_custom_tracing: true
```

Blog Post

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Compensation

Account for failures across the system building a convergent system.

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Compensation

- Spring Batch for jobs and batching up the work
- POST to a REST(ful) endpoint that will do the work
- Reuse the same code/libraries to do the compensation work that would have done the work in real time

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Pitfalls

- Poor communications between teams gets worse
- Coordinating Releases
- Development to spec vs finished service
- Leaving in -SNAPSHOT dependencies
- Poor monitoring in lower than production environments
- Lacking log aggregation in lower environments

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Grails Microservice Checklist

- ☐ Low overhead to start new projects
- ☐ Internal maven repo
- ☐ Automated deployments
- ☐ Server configuration management
- ☐ Good team to team communication

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Open Questions

- Correct size of Microservices
- Pure REST vs RESTful
- Security: Centralized vs Decentralized

Unless otherwise indicated, these slides are © 2013-2014 Pivotal Software, Inc. and licensed under a
Creative Commons Attribution-NonCommercial license: http://creativecommons.org/licenses/by-nc/3.0/

Learn More. Stay Connected

Thinking about Microservices? Make sure to complete the checklist first!

@beckje01

@springcentral | spring.io/video