Grails 4

What’s New and What To Know
About Me

1. Principal Software Engineer at OCI
2. Grails Engineering Team since 2015
3. Original Micronaut Engineer
4. @Schlogen
Agenda

1. What’s New
2. Upgrade Challenges
3. Micronaut Integration
History

Upgrade started October 2017

Then mostly delayed until late 2018

Milestone 1 released in February 2019

Milestone 2 released in March 2019 (yesterday)
What’s New

1. Java 8 Minimum
2. Groovy 2.5
3. Spring Boot 2.1
4. Spring 5
5. GORM 7 / Hibernate 5.3+
6. Gradle 5
7. Micronaut Integration
Upgrading

Grails 2 -> Grails 3 != Grails 3 -> Grails 4
Upgrading

1. Most deprecations have been removed
2. Some package restructuring
3. No major breaking API changes
4. Most plugins should just work
5. Breaking changes in Spring, Hibernate, Groovy
Notable:

1. Many configuration changes
2. Embedded container API
3. Endpoints Changes

Spring Changes


Nothing that should impact the average Grails application
GORM / Hibernate Changes

http://gorm.grails.org/7.0.x/hibernate/manual/index.html#upgradeNotes

Notable:
1. All operations now require a transaction
2. Proxy behavior has changed
3. No more REST client
Previously the author variable would be a direct Author instance
Impacts domains with inheritance (instanceof)
Groovy Changes

http://groovy-lang.org/releasenotes/groovy-2.5.html

Notable:
1. No more “all” jar
2. New annotations
3. Date extensions require a new dependency
4. JDK 11 warnings not resolved
Plugins

Plugins should just work unless…

They use an API that has been changed or removed

GrailsDomainClass

Spring Boot Embedded Server
Gradle Changes

https://docs.gradle.org/current/userguide/upgrading_version_4.html

Gradle 3 no longer supported, 4 not officially supported

Many breaking changes from 3 to 5
Micronaut Integration

- Micronaut a Foundational Library for building applications of any type
- Focuses on Small Memory Footprint and Speed
- Eliminates Reflection, Runtime Proxies and Runtime Analysis
Micronaut Integration

Grails 4 still using the Servlet API
Grails components conversion
Many existing Micronaut features usable today
• Kafka / RabbitMQ
• HTTP Client
• Datastore Configurations
Micronaut or Grails?

• Consider Building Configurations instead of Plugins
• Work with Micronaut, Spring (with `micronaut-spring`) and Grails
• Plugins only work with Grails
• … although some things only possible with Plugins (Views, taglibs etc.)
Why Micronaut?

• No reflection, no runtime proxies, no runtime byte code generation etc.
• Eliminating these leads to reduced memory consumption and faster startup
• Increasingly important for Microservices, Serverless, IoT, any low-memory footprint environment
Micronaut Configurations

- Configuration with @ConfigurationProperties
- Beans with @Singleton, @Factory etc.
- Conditional Behavior with @Requires
- Customization with @Replaces
```java
@ConfigurationProperties("example")
class ExampleConfiguration {
    String name = "someDefaultValue"
}

def context = ApplicationContext.run("example.name":"Demo")
FooConfiguration config = context.getBean(FooConfiguration)
assert config.name == "Demo"
```
Conditional Beans

```java
@Requires(property="example.enabled")
@Requires(beans=DataSource)
@Requires(missingBeans=Example)
@Singleton
class DefaultExampleBean implements Example {
    ...
}

def context = ApplicationContext.run("example.enabled":"true")
Example example = context.getBean(Example)
```
Overriding Beans

```java
@Replaces(DefaultExample)
@Singleton
class AlternativeExample implements Example {
  ...
}
```

- Types match much
- Beans annotated with `@Infrastructure` not replaceable
Micronaut HTTP Client

```java
@Client("https://api.github.com")
@Header(name="User-Agent", value="micronaut-client")
interface GithubClient {
    @Get("/repos/{+slug}")
    Info getInfo(String slug)
}
```

- Blocking or Non-Blocking (RxJava, Reactor or Future)
- Reflection / Runtime Proxy Free
Micronaut Messaging

- Micronaut Supports Message-Driven Applications
- Declarative Clients for Kafka & RabbitMQ
- Use @RabbitListener for RabbitMQ
- Use @KafkaListener for Kafka
- Planned Support For Other Messaging Systems
In Summary

• Upgrading very different compared to 2 -> 3
• Micronaut Provides an Awesome Foundation
• Building Blocks to Create Libraries, Configurations and Clients
• Most Micronaut Features Available in Grails
• Build Micronaut Libraries not Plugins
Questions?

CONNECT WITH US

1+ (314) 579-0066
@objectcomputing
objectcomputing.com