

SMARTER TESTING WITH SPOCK

PETER NIEDERWIESER
PRINCIPAL ENGINEER, GRADLEWARE

WHAT WE'LL TALK ABOUT

- ✿ What is Spock?
- ✿ State Based Testing
- ✿ Data Driven Testing
- ✿ Interaction Based Testing
- ✿ Spock Extensions
- ✿ More Cool Stuff

WHAT IS SPOCK?

SPOCK IS...

- ✱ A developer testing framework...
- ✱ for Groovy and Java applications...
- ✱ based on Groovy...
- ✱ fully compatible with JUnit...
- ✱ but going beyond!

SPOCK CAN...

- ✱ Reduce the lines of test code
- ✱ Make tests more readable
- ✱ Turn tests into specifications
- ✱ Be extended in powerful ways
- ✱ Bring back the fun to testing!

GETTING STARTED

- ✿ Homepage
<http://spockframework.org>
- ✿ Source Code
<http://github.spockframework.org/spock>
- ✿ Documentation
<http://docs.spockframework.org/en/latest>
- ✿ Spock Web Console
<http://webconsole.spockframework.org>
- ✿ Spock Example Project
<http://github.spockframework.org/spock-example>
- ✿ Slides and Code for this Presentation
<http://github.spockframework.org/smarter-testing-with-spock>

WHO'S USING SPOCK?



WHO'S USING SPOCK? (2)



STATE BASED TESTING

STATE BASED TESTING

- ✱ Classical Unit Testing

- ✱ Arrange

- ✱ Act

- ✱ Assert

- ✱ Given-When-Then

STATE BASED TESTING

- ✱ Classical Unit Testing

- ✱ Arrange

- ✱ Act

- ✱ Assert

- ✱ Given-When-Then

Show me the code!

RECAP: STATE BASED TESTING

- ✱ Blocks

setup: cleanup: expect: given: when: then:
where: and:

- ✱ Fixture Methods

setup() cleanup() setupSpec() cleanupSpec()

- ✱ Instance and @Shared fields

- ✱ old() and thrown()

DATA DRIVEN TESTING

DATA DRIVEN TESTING

- ✱ Test the same behavior...

- ✱ ...with varying data!

DATA DRIVEN TESTING

- ✱ Test the same behavior...

- ✱ ...with varying data!

Show me the code!

RECAP: DATA DRIVEN TESTING

- ✱ where: block
- ✱ Data tables
- ✱ External data sources
- ✱ @Unroll

INTERACTION BASED TESTING

INTERACTION BASED TESTING

- ✱ How objects communicate
- ✱ Mocking frameworks simplify the job
- ✱ Spock has its own mocking framework

INTERACTION BASED TESTING

- ✱ How objects communicate
- ✱ Mocking frameworks simplify the job
- ✱ Spock has its own mocking framework

Show me the code!

RECAP: INTERACTION BASED TESTING

✿ Creating

```
def sub = Mock(Subscriber)  
Subscriber sub = Mock()
```

✿ Mocking

```
1 * sub.receive("msg")  
(1..3) * sub.receive(_)  
(1.._) * sub.receive(_ as String)  
1 * sub.receive(!null)  
1 * sub.receive({it.contains("m")})  
1 * _./rec.*/"msg")
```


RECAP: INTERACTION BASED TESTING (2)

☼ Stubbing

```
// now returns status code
```

```
String receive(String msg) { ... }
```

```
sub.receive(_) >> "ok"
```

```
sub.receive(_) >>> ["ok", "ok", "fail"]
```

```
sub.receive(_) >> { msg -> msg.size() > 3 ? "ok" :  
"fail" }
```

☼ Mocking and Stubbing

```
3 * sub.receive(_) >>> ["ok", "ok", "fail"]
```

☼ Impressing your friends

```
(_.._) * _._(*_) >> _
```

NEW IN 0.7

WHAT'S TO COME?

NEW IN 0.7 (1)

- ✻ Group conditions/interactions w/ same target

```
with(person) {  
    name == "Fred"  
    age == 30  
    sex == "male"  
}
```

```
with(service) {  
    1 * start()  
    1 * stop()  
}
```

- ✻ Better mocking failure messages

NEW IN 0.7 (2)

- ☼ Stubs

```
def person = Stub(Person)
```

- ☼ Spies

```
def person = Spy(Person, constructorArgs: ["Fred", 30])
```

- ☼ Groovy Mocks

```
def person = GroovyMock(person) // GroovyStub, GroovySpy
```

- ☼ Declare interactions upfront

```
def person = Stub(Person) {  
    1 * sleep()  
    sing() >> "tra-la-la"  
}
```

- ☼ More at: http://docs.spockframework.org/en/latest/new_and_noteworthy.html

SPOCK EXTENSIONS

SPOCK EXTENSIONS

- ☼ Listeners

- ☼ Interceptors

- ☼ Annotation-Driven Extensions

- ☼ Global Extensions

BUILT-IN EXTENSIONS

- ✿ @Ignore
- ✿ @IgnoreRest
- ✿ @FailsWith
- ✿ @Timeout
- ✿ @AutoCleanup
- ✿ @Stepwise
- ✿ @RevertMetaClass
- ✿ @Rule

BUILT-IN EXTENSIONS

✱ @Ignore

✱ @IgnoreRest

✱ @FailsWith

✱ @Timeout

✱ @AutoCleanup

✱ @Stepwise

✱ @RevertMetaClass

✱ @Rule

Show me the code!

EXTERNAL EXTENSIONS

- ✱ spock-grails
- ✱ spock-spring
- ✱ spock-guice
- ✱ spock-tapestry
- ✱ spock-unitils
- ✱ spock-griffon
- ✱ spock-arquillian
- ✱ spock-extensions <http://github.com/robletcher/spock-extensions>

GRAILS EXTENSION

- ✱ <http://grails.org/plugin/spock>
- ✱ <http://github.spockframework.org/spock-grails>
- ✱ `grails install plugin spock 0.6`
- ✱ `grails test-app`
- ✱ `grails test-app integration:spock 'C*'`

GRAILS EXTENSION (2)

```
@TestFor(BookController)
@Mock(Book)
class BookControllerTests extends Specification {
    void "saving a book"() {
        params.title = "The Stand"
        params.pages = "500"

        when:
            controller.save()

        then:
            response.redirectedUrl == '/book/show/1'
            Book.count() == 1
    }
}
```

SPRING EXTENSION

```
@ContextConfiguration(locations = ["my-app-context.xml"])
class InjectionExamples extends Specification {
    @Autowired
    Service1 injectedByType

    @Resource
    Service1 injectedByName

    @Autowired
    ApplicationContext context
}
```

MORE COOL STUFF

CONFIGURING SPOCK

- ☼ ~/.spock/SpockConfig.groovy, or on class path, or with -Dspock.configuration

```
runner {  
    filterStackTrace false  
    include Fast  
    exclude Slow  
    optimizeRunOrder true  
}
```

```
@Fast  
class MyFastSpec extends Specification {  
    def "I'm fast as hell!"() { expect: true }  
  
    @Slow  
    def "sorry, can't keep up..."() {  
        expect: false  
    }  
}
```

TOOLING

- ✱ Eclipse, IDEA
- ✱ Ant, Maven, Gradle
- ✱ Jenkins, Bamboo, TeamCity
- ✱ Spock runs everywhere JUnit and Groovy run!

SPOCK UNDER THE HOOD

✱ You write...

```
a = 1; b = 2; c = 4
expect: sum(a, b) == c
```

✱ Spock generates...

```
def rec = new ValueRecorder()
SpockRuntime.verifyCondition(
    rec.record(rec.record(sum(rec.record(a),
        rec.record(b)) == rec.record(c)))
```

✱ You see...

```
sum(a, b) == c
|   |   |   |   |
3   1   2   |   4
           false
```


Q&A

- ✻ Homepage
<http://spockframework.org>
- ✻ Source Code
<http://github.spockframework.org/spock>
- ✻ Documentation
<http://docs.spockframework.org/en/latest>
- ✻ Spock Web Console
<http://webconsole.spockframework.org>
- ✻ Spock Example Project
<http://github.spockframework.org/spock-example>
- ✻ Slides and Code for this Presentation
<http://github.spockframework.org/smarter-testing-with-spock>