2GM Town Hall Meeting

Presented by:
James Kleeh, Puneet Behl, Paul King, Graeme Rocher, and Jen Wiese
Object Computing, Inc. (OCI)
12140 Woodcrest Exec. Dr., Ste. 300
Saint Louis, MO 63141 USA

© 2021 All Rights Reserved

No part of this publication may be photocopied or reproduced in any form without written permission from OCI. Nor shall the OCI logo or copyright information be removed from this publication. No part of this publication may be stored in a retrieval system, transmitted by any means, recorded or otherwise, without written permission from OCI.

Limits of Liability and Disclaimer of Warranty
While every precaution has been taken in preparing this material, including research, development and testing, OCI assumes no responsibility for errors or omissions. No liability is assumed by OCI for any damages resulting from the use of this information.
Agenda

- Introductions
- Birthday Celebration
- Exciting Announcements
- State of Grails
  - Foundation Updates
- State of Micronaut
  - Foundation Updates
- State of Groovy
- Community Q&A
Happy Birthday Grails!
Meet the Grails New Mascot!

Name The New Grails Knight!

The Knight doesn't have a name yet, and it's up to our community to come up with one! Provide your best suggestion for a chance to win a one-of-a-kind Grails swag bag (and bragging rights, of course).

2GM Teams Are Hiring!

- OSS Engineer working with our Product Development teams

- Software Engineer consulting and working on projects using Grails/Micronaut

- Apply today!
Grails® framework Updates

- Release Process Updates
- End Of Life Schedule
- Recent Releases
- Grails 5 Update
- Road Ahead
Grails - Release Updates

- Moved most CI builds to Github Actions
- Adoption of Semantic versioning
- New Release Cadence
- Grails 5
Grails - End of Life (EOL) Schedule

- Grails 2 will be EOL after June 30th, 2021.
- Grails 3 will be EOL after September 30th, 2021.
Grails - Recent Releases

- Grails 5.0.0.M1
- Grails 4.0.7
- Grails 3.3.12
Grails 5.0 Update

- Apache Groovy 3
- Gradle 6
- Spring 5.3 and Spring Boot 2.4
- Spock 2.0
- Grails 5 GA
Grails...The Road Ahead

- JFrog - Service End for JCenter and Bintray
- Making Grails GSP more modular and independent
- More Documentation and Blogs around how to leverage benefits of using Microanut in the Grails application
• Grails Foundation is a not-for-profit organization that exists to support and collectively lead the open source Grails® project.

• Grails Foundation is supported by a Technology Advisory Board that ensures the Framework continues to reflect and serve its diverse and growing user community.

• Grails Foundation serves to:
  ○ Ensure technical innovation and advancement of Grails as a free and open public use software development framework for a growing global community
  ○ Evangelize and promote the Grails framework as a leading technology in the JVM space
  ○ Build and support an ecosystem of complementary documentation, functionality, and services

• For more information on becoming a contributing member and for organizations interested in nominating a representative to the Grails Foundation Technical Advisory Board, please contact us at grails.org/foundation.
Grails Foundation Sponsorships

● As a not-for-profit organization, the Grails Foundation relies on the financial support from our amazing community of generous sponsors.

● Corporate and Community Sponsorships

● Plus limited edition swag!
Let’s Talk Micronaut

- Micronaut Roadmap Update
- Recent Releases
Micronaut Timeline

● 2.2.0 Released Nov 2020
  ○ Major improvements to the build plugins
  ○ Launch features (push to Github, Gradle Kotlin)
  ○ MQTT integration
  ○ R2DBC (preview)

● 2.3.0 Released Jan 2021
  ○ JMS integration
  ○ Latest GraalVM
  ○ R2DBC release
  ○ Locale Resolution

● 3.0.0 Coming Summer 2021
Micronaut has many contributors across organizations and individuals. Some repositories are community driven

- Over 275 unique contributors to the core repository alone
- 4.7k stars on GitHub
- Over 1800 people in our Gitter community
- Over 900 questions in stackoverflow
Micronaut Launch

Almost 35,000 applications created since the launch at the start of May 2020!
What’s Next?

- Major enhancements and improvements to Micronaut Data including document and graph datastores, and more
- Support for Jakarta annotations
- Validation improvements
- Injection by generics, qualifier improvements
- More integrations with cloud provider solutions
- Improved Kubernetes Support
- Support for more authentication options in the security module
- Integrations with more messaging solutions
- Numerous improvements to existing features and improved performance
Micronaut Foundation is a not-for-profit organization that exists to support and collectively lead the open source Micronaut® project.

Micronaut Foundation is supported by a Technology Advisory Board that ensures the Framework continues to reflect and serve its diverse and growing user community.

Micronaut Foundation serves to:

- Ensure technical innovation and advancement of Micronaut as a free and open public use software development framework for a growing global community.
- Evangelize and promote the Micronaut framework as a leading technology in the JVM space.
- Build and support an ecosystem of complementary documentation, functionality, and services.

For more information on becoming a contributing member and for organizations interested in nominating a representative to the Micronaut Foundation Technical Advisory Board, please contact us at foundation@micronaut.io.
Micronaut Foundation Sponsorship

- As a not-for-profit organization, the Micronaut Foundation relies on the financial support from our amazing community of generous sponsors.

- Corporate sponsorships
  - [https://micronaut.io/foundation/sponsorship.html](https://micronaut.io/foundation/sponsorship.html)

- Community sponsorships
  - [https://micronaut.io/foundation/community-sponsorship.html](https://micronaut.io/foundation/community-sponsorship.html)

- Plus limited edition swag!
Groovy Updates

- Groovy Stats 2020
- Groovy Roadmap
- Latest Releases
- Groovy 4
Remembering a Groovy Star

I’m so sad to hear that the wonderful @russel_winder passed away.

Those in the @ApacheGroovy community will remember his contributions to the language, to GPar, and more.

I always loved his humour. Thoroughly enjoyed our conversations. I will miss him a lot 😢

Roy van Rijn  @roywannijn  •  Jan 24
Reposting to @aaimiray and @russel_winder
RIP @russel_winder, loved the interactions we had, great passion and sense of humor.

Mark Handley  @MarkHandley  •  Jan 24
Very sad! Russel taught me software engineering back in 1986. Always came out of his lectures with notes, but thoroughly inspired. Great teacher, inspiration and genuinely nice person. RIP.

Hubert Klein Ikkink  @mrhaki  •  Jan 24
Reposting to @aaimiray and @russel_winder
Sad news indeed. Loved his talks and conversations at conferences. A great person. ❤️

Ralf D. Müller  @RalfDMueller  •  Jan 24
Reposting to @gglaforg @russel_winder and @ApacheGroovy
Sad to read this but happy that I had the chance to meet him at conferences. His work and memory will still remain in the code of so many tools. Rest in Peace, Russel.

Hans Dockter  @Hans_d  •  Jan 25
Reposting to @Devoxx and @russel_winder
This is very sad news. At the very beginning of Gradle, I had a lot of interactions with Russel and he energized our project a lot. He was a true original and will be missed. RIP!
Groovy Stats 2020

- Groovy 3 general availability (a year ago)
- 21 releases (211 since 2003)
- ~350M downloads (~740M since 2012)
Groovy Roadmap

- Groovy 2.4  (Essentially EOL but critical fixes possible, JDK6+)
- Groovy 2.5  (Important fixes only, JDK7+)
- Groovy 3  (Recommended version JDK8+)
- Groovy 4  (Current focus)

- Release Notes for Groovy 4
  - [http://groovy-lang.org/releases.html](http://groovy-lang.org/releases.html)
Latest releases (2020/Dec/03)

- 2.4.21
- 2.5.14
- 3.0.7
- 4.0.0-alpha-2

- All included CVE fix related to temp directories
- New releases expected shortly
  - Groovy 4 moving to beta/RC stages soon
Groovy 4: Structural Changes

Maven coordinate change

org.codehaus.groovy → org.apache.groovy

Legacy package removal

groovy-xml:
groovy.util (includes XmlParser & XmlSlurper) → groovy.xml

Module changes for groovy-all

groovy-testng: was included, now optional
groovy-yaml: was optional, now included
Groovy 4: Consolidation

Old parser removal

- Antlr2
- Antlr4

Classic bytecode generation removal

- Classic
- Indy
Reminder

Groovy’s static nature:

- Supports extensible type checking
  - Make type checker weaker or stronger depending on the context
  - This feature is used but custom type checkers are not commonly shared
Groovy 4: Built-in Type Checkers (incubating)

```groovy
import groovy.transform.TypeChecked

@TypeChecked(extensions = 'groovy.typecheckers.RegexChecker')
def whenIs2020Over() {
    def newYearsEve = '2020-12-31'
    def matcher = newYearsEve =~ /\d{4}-\d{1,2}-\d{1,2}/  // PatternSyntaxException
}
```

1 compilation error:
[Static type checking] - Bad regex: Unclosed group near index 26
`\d{4}`-`\d{1,2}`-`\d{1,2}`
at line: 6, column: 19
Groovy 4: Built-in Type Checkers

```groovy
// missing closing repetition quantifier brace
// missing closing group bracket
Pattern.compile(/?/) // dangling meta character '?' (Java Longhand)

'foobar' =~ /[o]{2/ // missing closing repetition quantifier brace
'foobar' =~ /(foo/ // missing closing group bracket
Pattern.matches(/?/, 'foo') // dangling meta character '?' (Java Longhand)

def m = 'foobar' =~ /(...)(...)/
assert m[0][1] == 'foo' // okay
assert m[0][3] // type error: only two groups in regex

Pattern p = Pattern.compile('(...)(...)
Matcher m = p.matcher('foobar')
assert m.find()
assert m.group(1) == 'foo' // okay
assert m.group(3) // type error: only two groups in regex
```
Reminder

Groovy’s Macro Methods:

- **Support easier AST transform creation via macros**
- **Transformation doesn’t rely on an annotation in the source code**
- **This feature is used but custom macro methods are not commonly shared**
Groovy 4: Built-in Macro Methods (incubating)

def num = 42
def list = [1, 2, 3]
def range = 0..5
def string = 'foo'

println NV(num, list, range, string)

num=42, list=[1, 2, 3], range=[0, 1, 2, 3, 4, 5], string=foo

println NVI(range)

range=0..5

println NVD(range)

range=<groovy.lang.IntRange@14 from=0 to=5 reverse=false inclusive=true modCount=0>
Groovy 4: Built-in Macro Methods (incubating)

def num = 42
def list = [1,2,3]
def range = 0..5
def string = 'foo'

println NV(num, list, range, string)
println NVI(range)
println NVD(range)

num=42, list=[1, 2, 3], range=[0, 1, 2, 3, 4, 5], string=foo
range=0..5
range=<groovy.lang.IntRange@14 from=0 to=5 reverse=false inclusive=true modCount=0>

println "num=" + num.toString() + ...
println "range=" + range.inspect()
println "range=" + range.dump()
import org.apache.groovy.util.JavaShell

def opts = ['--enable-preview', '--release', '14']
def src = 'record Coord(int x, int y) {}'

Class coordClass = new JavaShell().compile('Coord', opts, src)

assert coordClass.newInstance(5, 10).toString() == 'Coord[x=5, y=10]'
import groovy.contracts.*

@Invariant({ speed >= 0 })
class Rocket {
    int speed = 0
    boolean started = false

    @Requires({ !started })
    def start() { started = true }

    @Requires({ started })
    @Ensures({ old.speed < speed })
    def accelerate(inc) { speed += inc }
}

def r = new Rocket()
// r.accelerate(5) // PreconditionViolation
r.start()
r.accelerate(5)
Groovy 4: @POJO (incubating)

```groovy
@CompileStatic
@POJO
@Canonical(includeNames = true)
class Point {
    Integer x, y
}

@CompileStatic
@POJO
class PointList {
    @Delegate List<Point> points
}

Predicate<Point> xNeqY = p -> p.getX() != p.getY();

Point p13 = new Point(1, 3);
List<Point> pts = List.of(p13, new Point(2, 2), new Point(3, 1));
PointList list = new PointList();
list.setPoints(pts);

System.out.println(list.size());
System.out.println(list.contains(p13));
list.forEach(System.out::println);

long count = list.stream().filter(xNeqY).collect(counting());
System.out.println(count);
```

Groovy jar not needed on classpath - currently only in simple cases like this one
Groovy 4.0: @RecordType (incubating)

```groovy
@groovy.transform.RecordType
class Cyclist {
    String firstName
    String lastName
}

def richie = new Cyclist('Richie', 'Porte')
```

Produces a class that:
- is implicitly final
- has a private final field `firstName` with an accessor method `firstName()`; ditto for `lastName`
- has a default `Cyclist(String, String)` constructor
- has a default `serialVersionUID` of 0L
- has implicit `toString()`, `equals()` and `hashCode()` methods

```groovy
record Cyclist(String firstName, String lastName) { }  // possible future syntax
```
Groovy 2.5: AST Transforms: @Immutable becomes meta-annotation

```groovy
@Immutable
class Point {
    int x, y
}
```

```groovy
@ToString(includeSuperProperties = true, cache = true)
@EqualsAndHashCode(cache = true)
@ImmutableBase
@ImmutableOptions
@propertyOptions(propertyHandler = ImmutablePropertyHandler)
@TupleConstructor(defaults = false)
@MapConstructor(noArg = true, includeSuperProperties = true, includeFields = true)
class Point {
    int x, y
}
```
Groovy 4.0: AST Transforms: @RecordType meta-annotation (incubating)

```groovy
@RecordType
class Point {
    int x, y
}
```

```groovy
@RecordBase
@ToString(cache = true, includeNames = true)
@EqualsAndHashCode(cache = true, useCanEqual = false)
@ImmutableOptions
@PropertyOptions(propertyHandler = ImmutablePropertyHandler)
@TupleConstructor(defaults = false)
@MapConstructor
@KnownImmutable
@POJO
class Point {
    int x, y
}
```
Groovy 4.0: Language integrated query
(incubating)

DSL for querying aggregations

def nums = [1, 2, 3, 3, 4, 7]
GQ {
    from n in nums
    groupby ((0 == n % 2) as isEven)
    select isEven, count(), max(n), avg(n)
}
Groovy 4.0: Language integrated query

def nums = [1, 2, 3, 3, 4, 7]
GQ {
    from n in nums
    groupby ((0 == n % 2) as isEven)
    select isEven, count(), max(n), avg(n)
}
import static org.apache.groovy.ginq.provider.collection.runtime.Queryable.from
import org.apache.groovy.ginq.provider.collection.runtime.NamedRecord

def nums = [1, 2, 3, 3, 4, 7]
assert [[false, 4, 7, 3.5G], [true, 2, 4, 3.0G]] ==
    from(nums).groupBy(e -> new NamedRecord([e % 2 == 0], ['isEven'], ['n'])).sourceRecord(e)
    .select((e, q) -> Tuple.tuple(
        e.v1.isEven,
        e.v2.count(),
        e.v2.max(n -> n),
        e.v2.avg(n -> new BigDecimal(n)))).toList()
Groovy 4.0: Language integrated query

Supports nesting

```groovy
assert [[1, 1], [2, 4], [3, 9]] == GQ {
    from v in (from n in [1, 2, 3]
        select n, n ** 2 as nSquared
    )
    select v.n, v.nSquared
}.toList()
```
Groovy 4.0: Language integrated query

Supports aggregate functions

```groovy
def nums = [1, 2, 3, 3, 4, 7]
GQ {
    from n in nums
    groupby ((0 == n % 2) as isEven)
    select isEven, count(), max(n), avg(n)
}
```

Also: min, sum, median, stdev, stdevp, var, varp, agg
import groovy.json.JsonSlurper

def json = new JsonSlurper().parseText('''
{
  "fruits": [
    {"name": "Orange", "price": 11},
    {"name": "Apple", "price": 6},
    {"name": "Banana", "price": 4},
    {"name": "Mango", "price": 29},
    {"name": "Durian", "price": 32}
  ]
}
''')

def expected = [['Mango', 29], ['Orange', 11], ['Apple', 6], ['Banana', 4]]

assert expected == GQ {
  from f in json.fruits
  where f.price < 32
  orderby f.price in desc
  select f.name, f.price
}.toList()
Groovy Roadmap: under investigation

- Switch expressions
- Additional destructuring/pattern matching within switch case expressions
- Smarter type checking: non-null, pure
- Module definitions in Groovy

- Release Notes for Groovy 4
  - [http://groovy-lang.org/releases.html](http://groovy-lang.org/releases.html)
MICRONAUT COMMUNITY RESOURCES

- @micronautfw
- github.im/micronautfw
- micronaut.io/launch
- docs.micronaut.io
- micronaut.io/learn
- guides.micronaut.io
- micronaut.io/blog
- micronaut.io/faq
- micronaut.io/foundation
- github.com/micronaut-projects/micronaut-core
- objectcomputing.com/products/micronaut/solutions
- objectcomputing.com/products/micronaut
- objectcomputing.com/resources/events
GRAILS & GROOVY COMMUNITY RESOURCES

- @grailsframework
- Grails.org
- github.com/grails
- Grails-slack.cfapps.io
- grails.org/blog/index.html
- grails.org/foundation
- objectcomputing.com/products/grails/solutions
- objectcomputing.com/products/grails
- @ApacheGroovy
- groovy-lang.org
- github.com/apache/groovy
- groovy-community.slack.com
- Apache.org
Q&A
THANK YOU!
LET’S STAY CONNECTED

✉️ 2GM@objectcomputing.com

🐦 @micronautfw

🐦 @grailsframework

🐦 @ApacheGroovy