

# Getting Started with Micronaut Data



Michael P. Redlich



# Who's Mike?

- Bachelor of Science, Computer Science
- “Petrochemical Research Organization”
- Java Queue News Editor, InfoQ
- Leadership Council, Jakarta EE Ambassadors
- Amateur Computer Group of New Jersey

# Objectives

- What is Micronaut?
- What is Micronaut Data?
- Get Started Building an Application
  - Live Demo (yea!)
- Micronaut Resources

# What is Micronaut?

*“A modern, JVM-based, full-stack framework for building modular, easily testable microservice and serverless applications.”*

Micronaut website: <https://micronaut.io>

# What is Micronaut?

- A full-featured, full-stack JVM-based lightweight application framework for creating microservice-based, cloud-native and serverless applications that can be written in Java, Groovy and Kotlin
- Created by Graeme Rocher, Principal Software Engineer at Object Computing, Inc.

# What is Micronaut?

- First introduced at the Greach Conference in March 2018
- Designed from the ground up for micro services and serverless applications
- Current version: 1.3.2
- Based on Ahead of Time (AoT) compilation

# What is Micronaut Data?

“A database access toolkit that uses Ahead-of-Time (AoT) compilation to pre-compute queries for repository interfaces that are then executed by a thin, lightweight runtime layer.”

Micronaut website: <https://micronaut.io>

# What is Micronaut Data?

- Originally known as Micronaut Predator
- Inspired by GORM and Spring Data
- Current version: 1.0.0



# Why Micronaut Data?

- No runtime model
- No query translation
- No reflection or runtime proxies
- Type safety

# Databases Supported

- H2
- MySQL 5.5+
- Oracle 12c+
- PostgreSQL 9.5+
- SQLServer 2012+

# Defining Repositories

```
@Repository  
interface BeerRepository  
    extends CrudRepository<Beer, Long> {  
    Beer find(String name);  
}
```

# Query Computation

```
@Inject  
BeerRepository beerRepository;
```

```
@Inject  
BeanContext beanContext;
```

```
@Test  
void testAnnotationMetadata() {  
    String query = beanContext.getBeanDefinition(BeerRepository.class)  
        .getRequiredMethod("find", String.class)  
        .getAnnotationMetadata()  
        .stringValue(Query.class)  
        .orElse(null);  
  
    assertEquals("SELECT...", query);  
}
```

# Let's Get Started...



# Generating an Application

```
mn create-app myapp --features data-jdbc
```

```
mn create-app myapp --features data-hibernate-jpa
```

Demo Time...

# Micronaut Resources

- <https://micronaut.io>
- <https://micronaut-projects.github.io/micronaut-data/latest/guide>
- <https://objectcomputing.com/products/micronaut/resources#on-demand>



# Micronaut Resources

- <https://infoq.com/news/2018/10/the-road-to-micronaut-1.0>
- <https://infoq.com/news/2018/12/micronaut-for-spring-graalvm>
- <https://infoq.com/news/2019/05/micronaut-1.1-cloud-native>

# Contact Info

`mike@redlich.net`

`@mpredli`

`redlich.net/portfolio`

`github.com/mpredli01`

# Thanks!

