

Testing a Spring Service with Spring Data and a NoSQL Back-end

INTRODUCTION



Steven Haines

PRINCIPAL SOFTWARE ARCHITECT

@geekcap www.geekcap.com



Overview



**Overview of Globomatics
Review Service**

Testing the Controller

Testing the Service

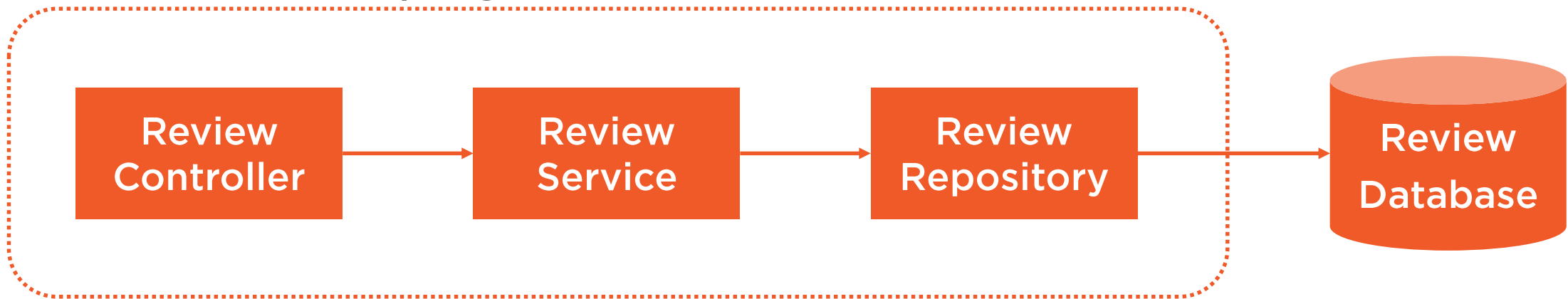
Testing the Repository

- Simple
- More Elegant



Globomatics Review Service

Spring Boot Microservice



Review

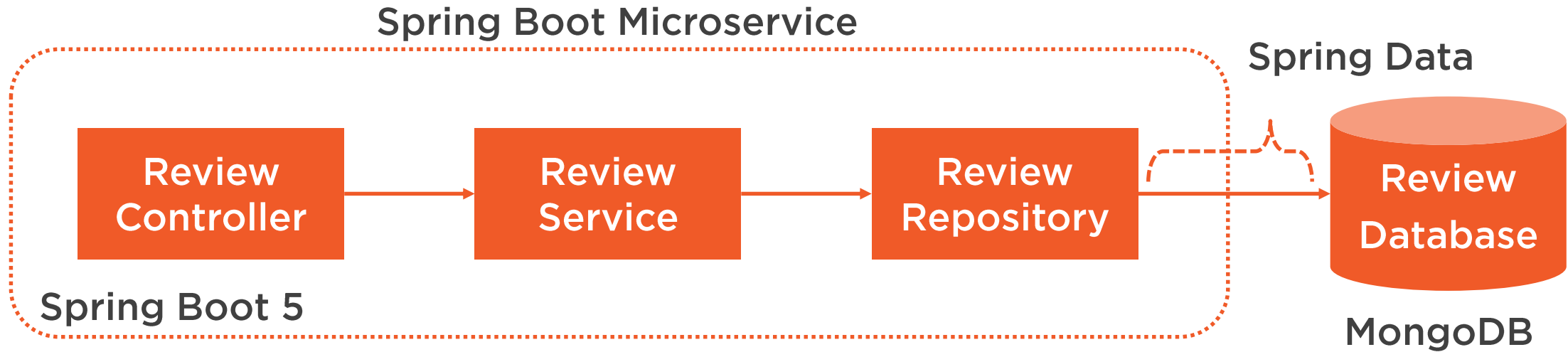
ID	Primary Key
Product ID	Product ID
Version	Review Object Version
Entries	Reviews for this Product

Review Entry

Username	Reviewer
Date	Review Date
Review	Content of the Review



Review Service Technology Stack



Spring Boot 5

RestController

Autowiring, Convention
over Configuration

MongoDB

Embedded Mongo

Use at runtime for
simplicity

Spring Data

Define Interface

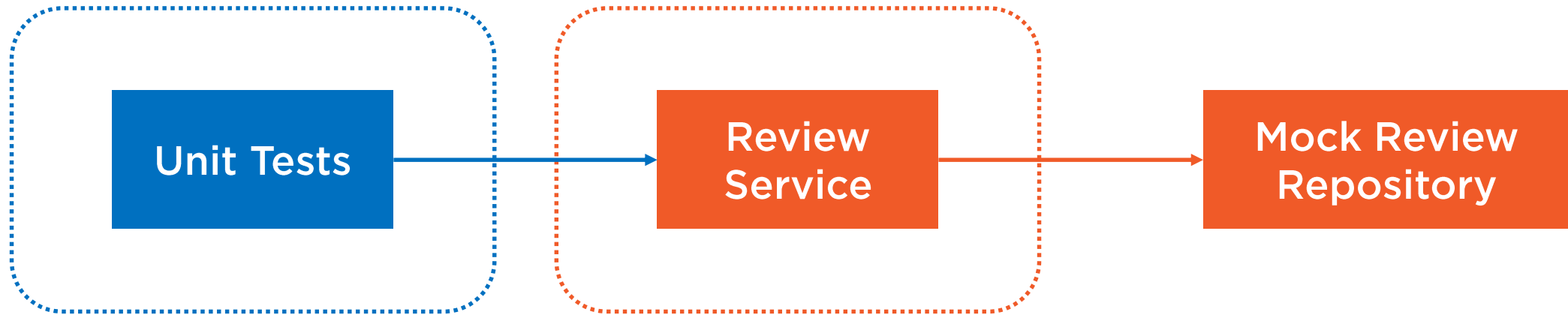
Generated
Implementation



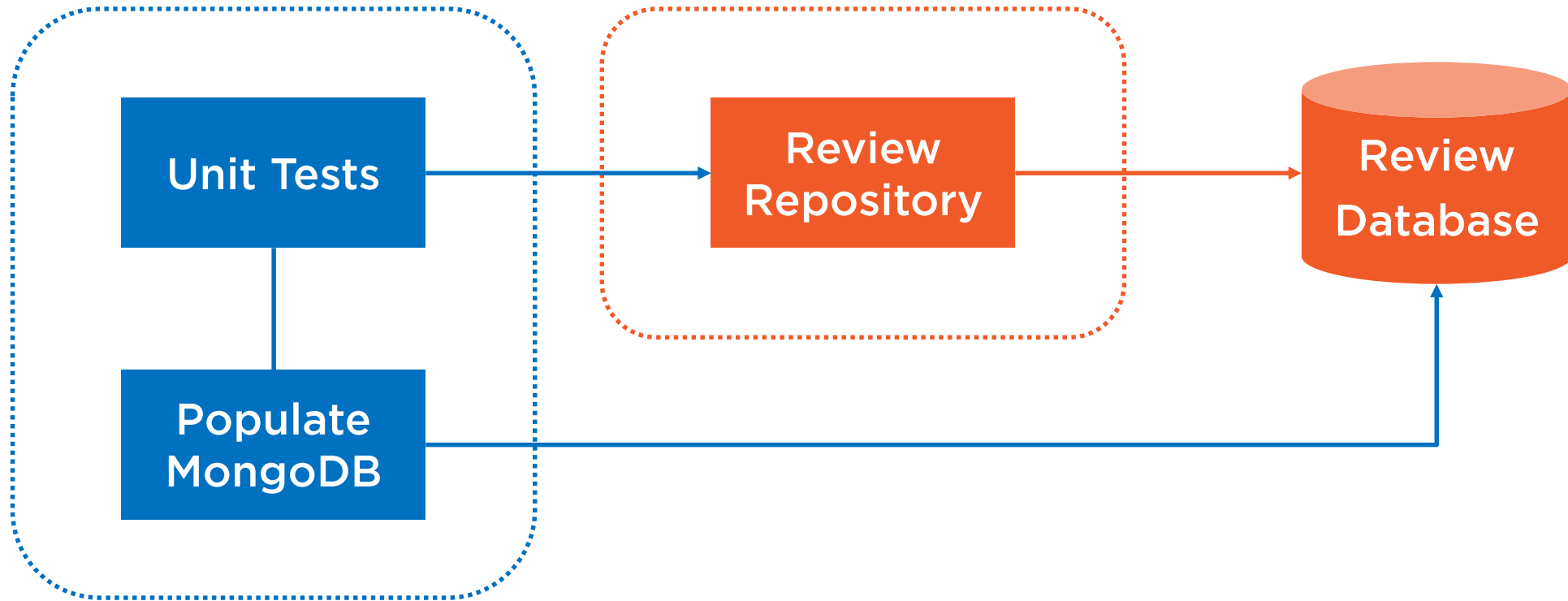
Strategy: Testing the Controller



Strategy: Testing the Service



Strategy: Testing the Repository



Demo



ReviewController walkthrough



Demo



ReviewControllerTest walkthrough



Demo



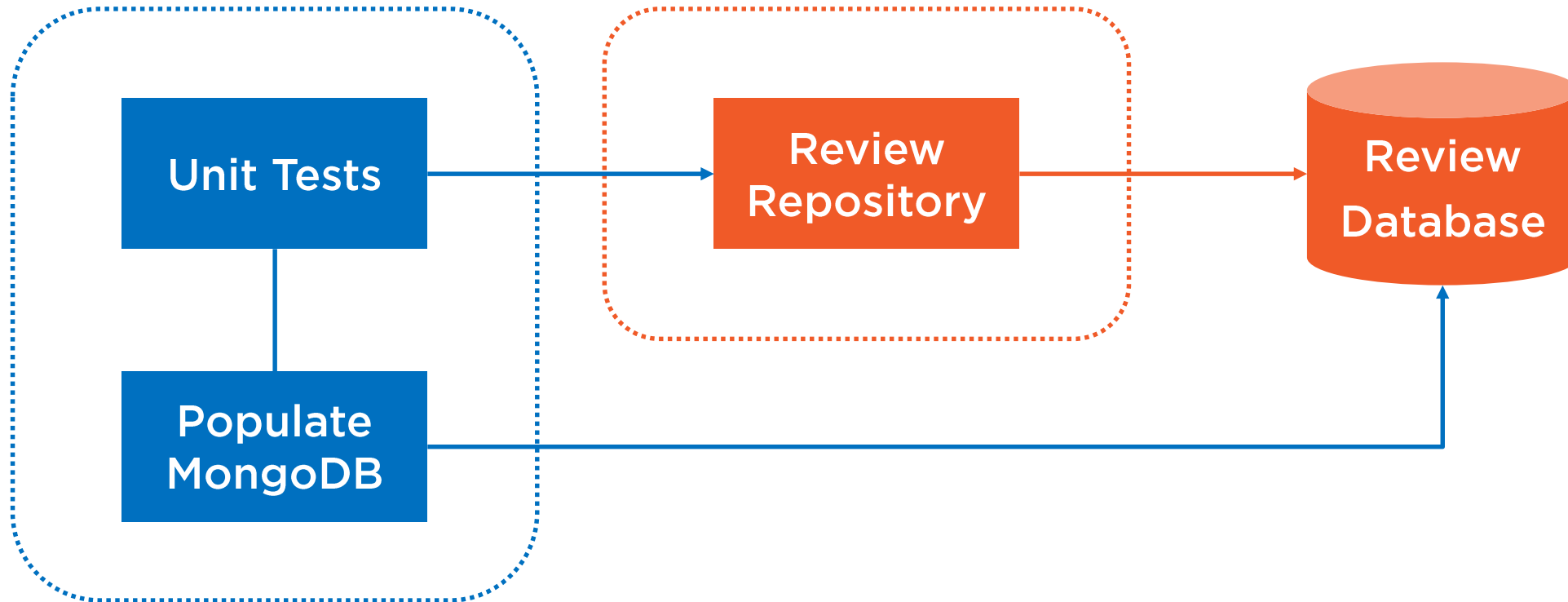
ReviewServiceTest walkthrough



Testing a MongoDB Repository



Strategy: Testing the Repository



```
@DataMongoTest
class ReviewRepositoryTest {

    @Autowired
    private MongoTemplate mongoTemplate;

    @Autowired
    private ReviewRepository repository;

    ...
}
```

◀ Loads Embedded Mongo

◀ Autowires a MongoTemplate

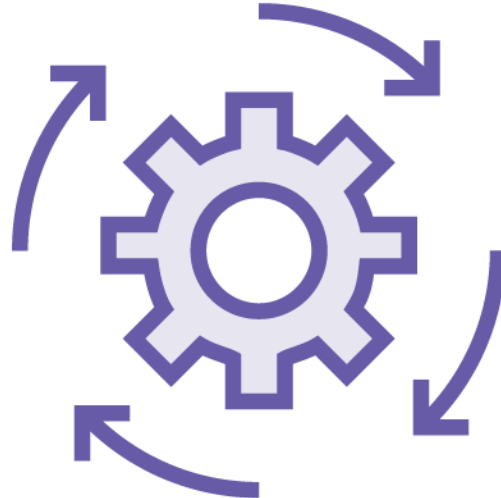
◀ Autowires our repository



MongoDB Test Lifecycle



@BeforeEach
Pre-populate data
from a JSON file



@Test
Run tests against
known data



@AfterEach
Delete pre-populated
Data

sample.json

```
[ {  
    "id": "1",  
    "productId": 1,  
    "version": 1,  
    "entries": [{  
        "username": "user1",  
        "date": "...",  
        "review": "My review"}  
    ]  
}]
```

```
/src/test/resources  
    /data/sample.json
```

@BeforeEach

@AfterEach

◀ Array of sample reviews

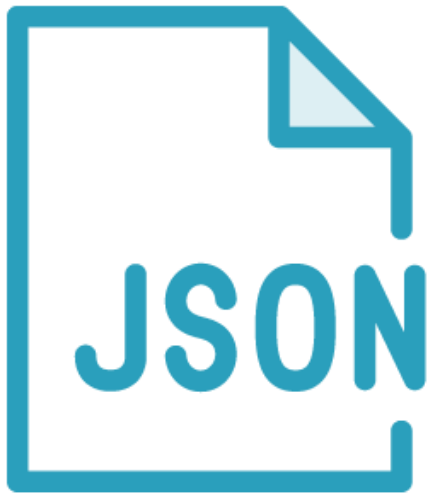
◀ Stored in
src/test/resources/data

◀ Populate Data

◀ Drop Data



Importing Data from JSON to MongoDB



Jackson

Map JSON to Objects



MongoTemplate

Save Objects to MongoDB




```
// @BeforeEach – Pre-populate MongoDB  
Review[] objects = mapper.readValue(SAMPLE_JSON, Review[].class);  
Arrays.stream(objects).forEach(mongoTemplate::save);  
  
// @AfterEach – Clean-up MongoDB  
mongoTemplate.dropCollection("Reviews");
```

Jackson and MongoTemplate

Load the SAMPLE_JSON file into an Array of Review objects

Save each Review to MongoDB

Clean-up after each test



Demo



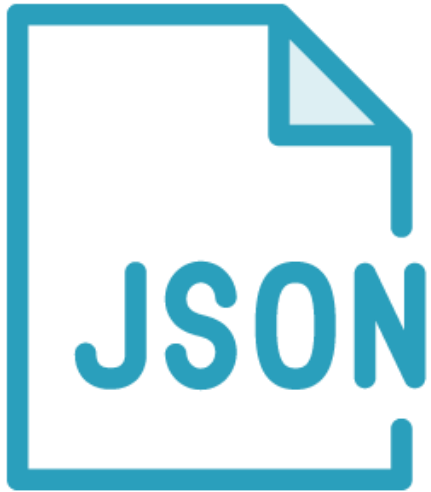
ReviewRepositoryTest walkthrough



A More Elegant Solution (Not for the faint of heart)



Current Limitations



JSON MongoDB Loading

The code to load a JSON file into MongoDB is in each test



Single JSON File

All Tests use the same JSON file



Custom JUnit 5 Extension



BeforeEachCallback

Load arbitrary JSON file into
MongoDB



AfterEachCallback

Clean up the collection

MongoSpringExtension.java

```
public class MongoSpringExtension
    implements BeforeEachCallback,
               AfterEachCallback {

    public void
    beforeEach(ExtensionContext
                context) {

        ...
    }

    public void
    afterEach(ExtensionContext
              context) {

        ...
    }
}
```

- ◀ **MongoSpringExtension implements BeforeEachCallback and AfterEachCallback**
- ◀ **Called before each test method**
- ◀ **Called after each test method**



ReviewRepositoryTestMoreElegant.java

```
@DataMongoTest
@ExtendWith(MongoSpringExtension.class)
class ReviewRepositoryTestMoreElegant {

    @Autowired
    private MongoTemplate mongoTemplate;
    public MongoTemplate getMongoTemplate()
    {
        return mongoTemplate;
    }

    @Test
    @MongoDataFile(value = "sample.json",
                   classType = Review.class,
                   collectionName = "Reviews")
    void testFindAll() { ... }
}
```

◀ Extend with
MongoSpringExtension

◀ Returns the autowired
MongoTemplate

◀ New MongoDataFile annotation



Demo



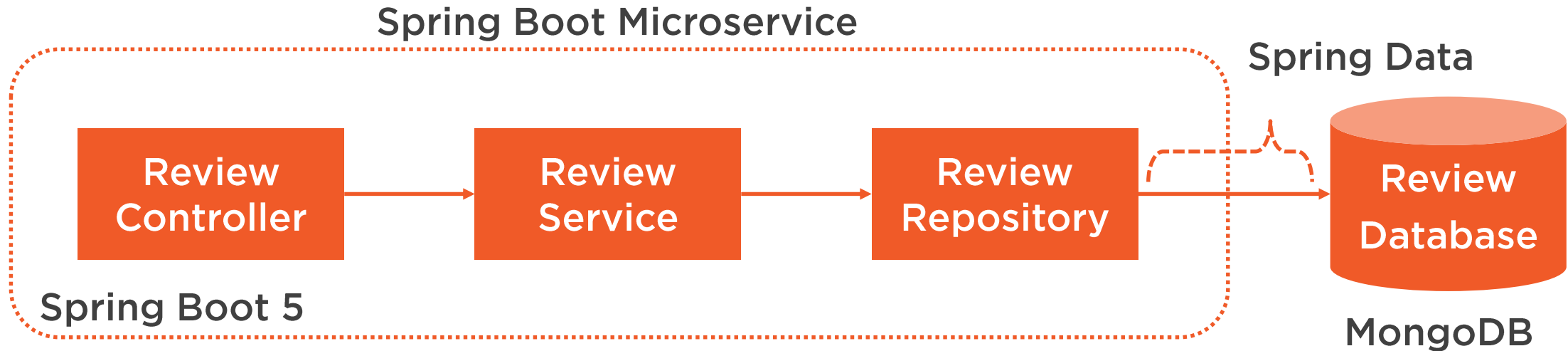
Building a more elegant solution



Summary



Review Service Technology Stack



Spring Boot 5

RestController

Autowiring, Convention
over Configuration

MongoDB

Embedded Mongo

Use at runtime for
simplicity

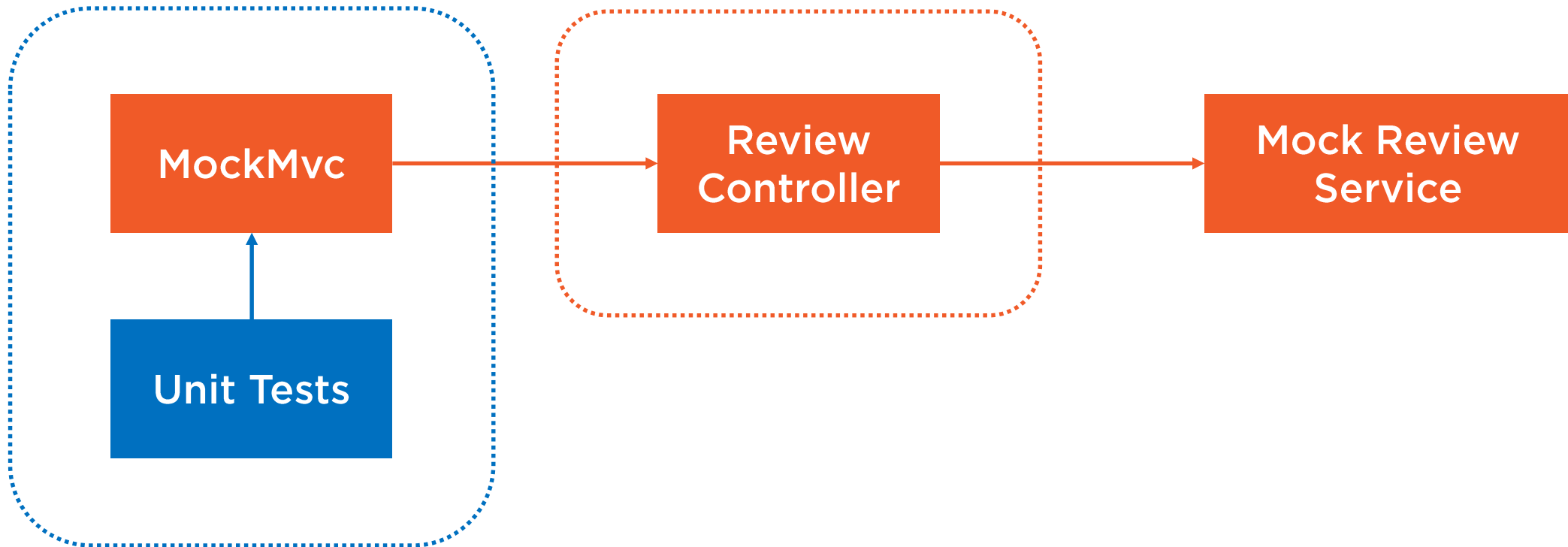
Spring Data

Define Interface

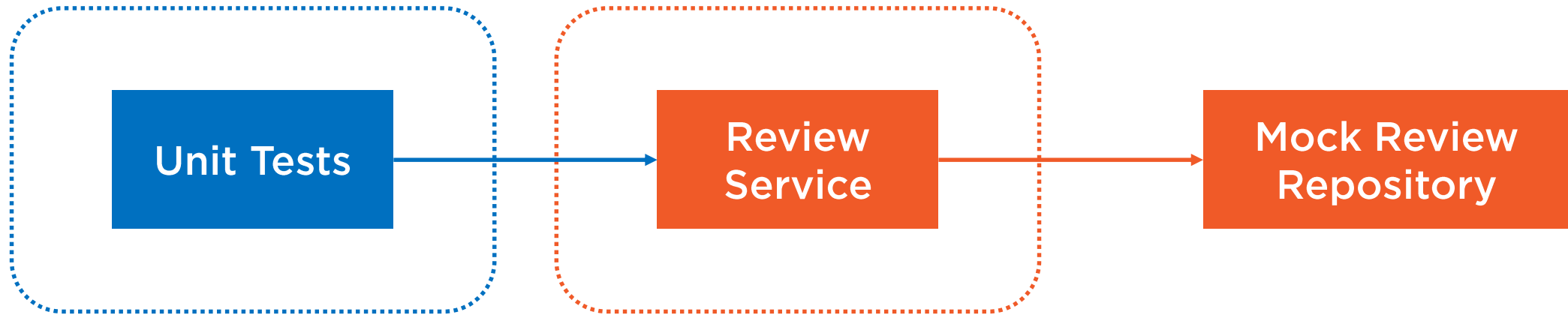
Generated
Implementation



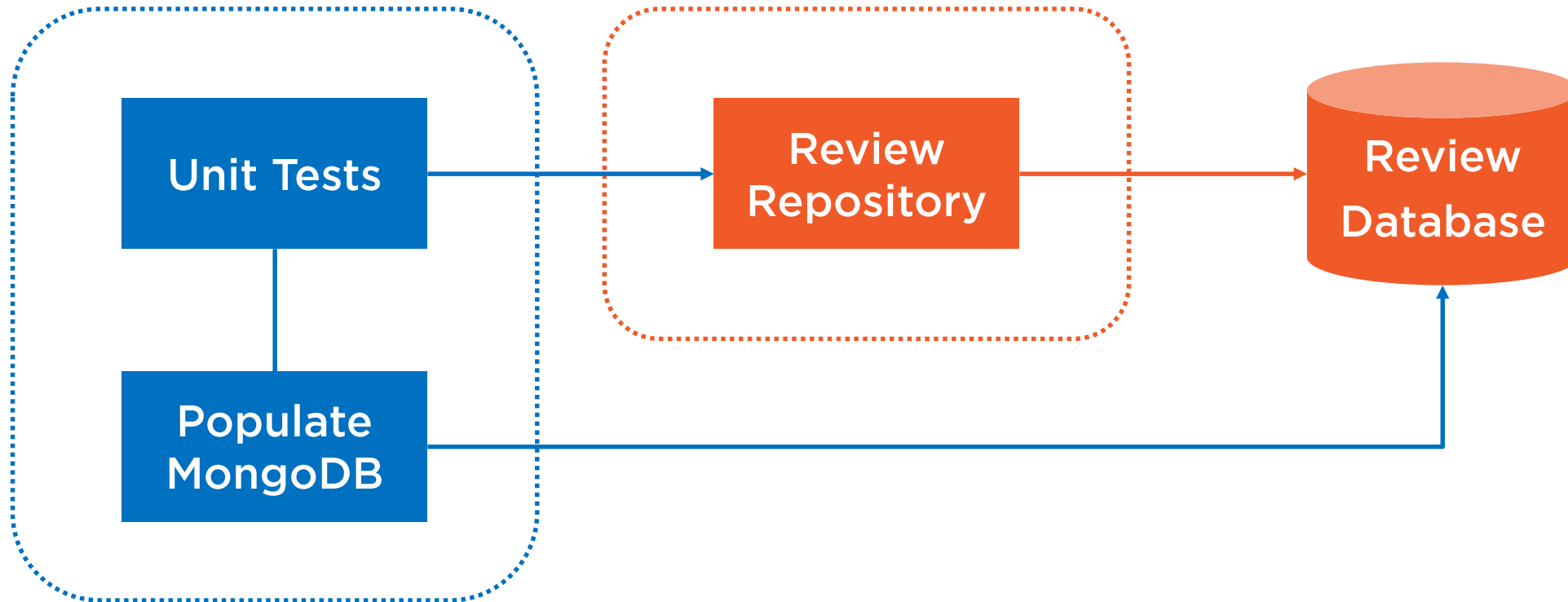
Strategy: Testing the Controller



Strategy: Testing the Service



Strategy: Testing the Repository



Summary



Advanced Controller Tests with MockMvc
Testing Repositories that use MongoDB

