Spring Cloud Function

Write once execute anywhere!

Oleg Zhurakousky

Twitter: @z_oleg

Github: @olegz

Who am I?

- Intro
  - Name: Oleg Zhurakousky
  - Project Lead: Spring Cloud Function & Spring Cloud Stream
  - With Spring organisation since 2008

Segment 1

Functions, why do I care?

Pulse check - thumbs up if. . .

- You are using Spring/SpringBoot today?

Pulse check - thumbs up if. . .

- You are using Spring Cloud Function today?

Pulse check - thumbs up if. . .

- You are using Spring Cloud Stream today?

Pulse check - thumbs up if. . .

- You are using Project Reactor (Flux/Mono) today?

Functions - why do I care?

- What are we talking about?

```
@FunctionalInterface
public interface Supplier<T> {
    T get();
}

@FunctionalInterface
public interface Function<T, R> {
    R apply(T t);
    . . .
}

@FunctionalInterface
public interface Consumer<T> {
    void accept(T t);
}
```

Functions - why do I care?

Demo

Functions - Why?

- Simplicity
- Consistency
- Extensibility
- Portability

Functions - Why?

```
@FunctionalInterface
public interface MessageSource<T> {
     @Nullable
     Message<T> receive();
}

@FunctionalInterface
public interface MessageHandler {
     void handleMessage(Message<?> message) throws MessagingException;
}

@FunctionalInterface
public interface Callable<V> {
    V call() throws Exception;
}
```

Functions - Why?

```
@FunctionalInterface
public interface MessageSource<T> extends Supplier<Message<T>> {
     @Nullable
     Message<T> get();
}

@FunctionalInterface
public interface MessageHandler extends Consumer<Message<T>>{
     void accept(Message<T> message) throws MessagingException;
}

@FunctionalInterface
public interface Callable<V> extends Supplier<V> {
    V get() throws Exception;
}
```

Functions - Why?

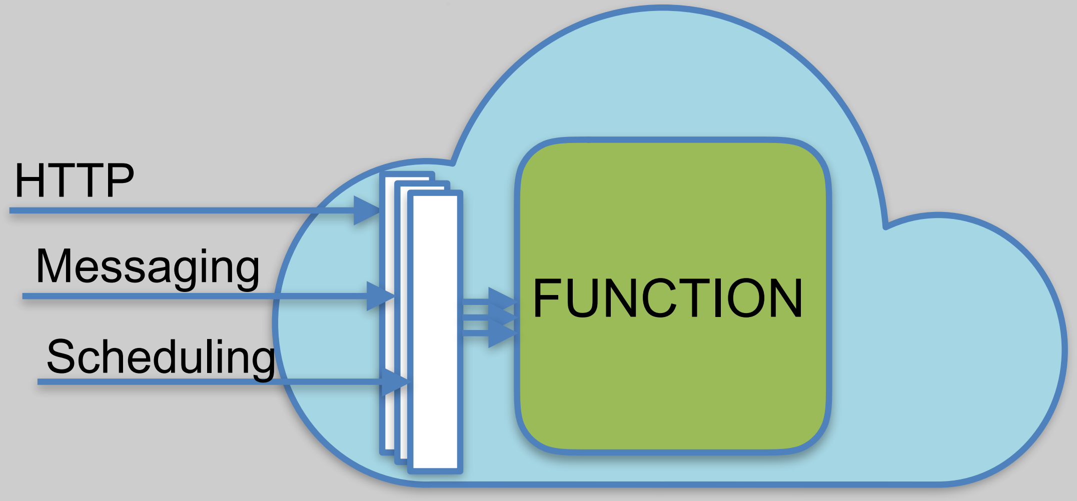
- Contract
- Pattern

Functions - why do I care?

- What can you NOT do with functions?
  - Supplier<O>
  - Function<I, O>
  - Consumer<I>

Functions - why do I care?

Activation and invocation facade



Spring Cloud Function - what and how?

- The goals:
  - Promote implementation of business logic via Functions
  - Uniformed and portable programming model
  - Integration with server less platforms
    - Amazon AWS
    - Microsoft Azure
    - others…

Spring Cloud Function - what and how?

Demo

Segment 1 - summary

- Functions are simple, expressive extensible and portable.
- Most if not all requirements could be expressed with functions
- Still needs activation and invocation facade
- Spring Cloud Function - a facade to address limitation and add additional features.

Segment 1 - the end!

Questions?

Segment 1 - the end!

BREAK

Segment 2

Spring Cloud Function - under the hood.

Spring Cloud Function - under the hood?

- Features:
  - Transparent Type Conversion
  - Function Composition
  - POJO functions (if it looks/smells like a function it must be a function)
  - Reactive support
  - Arity - functions with multiple inputs/outputs

Spring Cloud Function - under the hood?

- Features (cont):
  - Deployment of packaged functions (JARs or exploded archives)
    - Boot configuration
    - Simple Spring configurations
    - Simple non-Spring packages

Spring Cloud Function - features?

- Features (cont):
    - Function routing
    - Web Support
    - Message - first class citizen

Spring Cloud Function - under the hood?

- Core strategies
  - Function Catalog
  - Function Registry
  - Function Registration

Spring Cloud Function - under the hood?

- Function Registration:
  - Container to store meta information about the function:
    - Target function
    - Input/Output type
    - Name(s)
    - Additional properties
  - Can be used for manual function registration

Spring Cloud Function - under the hood?

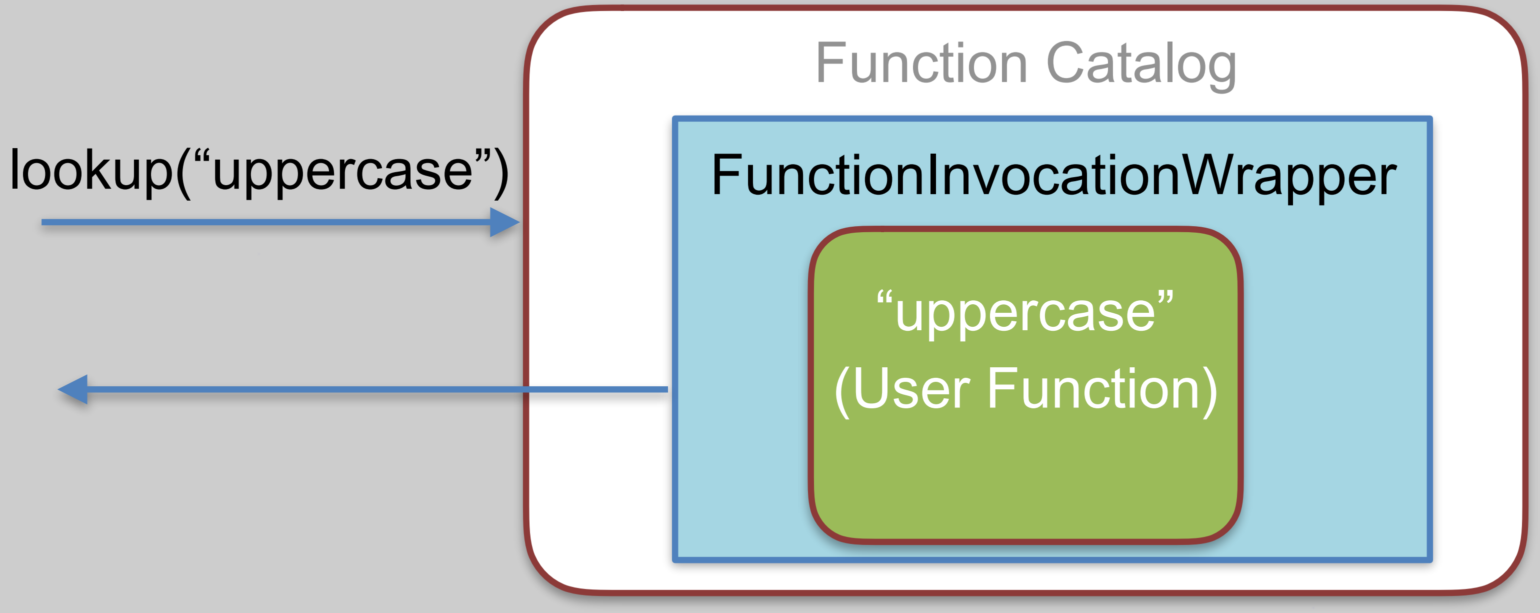
- Function Registry:
  - Registers functions with FunctionCatalog
  - It can instrument and decorate functions with additional features:
    - Type conversion
    - Composition
    - etc…
  - Creates FunctionRegistration for each function and registers it

Spring Cloud Function - under the hood?

- Function Catalog:
  - Function repository
  - An accessor to FunctionRegistry

Spring Cloud Function - under the hood?

- Function Catalog (cont):



Spring Cloud Function - under the hood?

DEMOS

Spring Cloud Function - under the hood?

- Transparent type conversion - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- Reactive function support - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- Function composition - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- Function routing - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- Web Support - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- Function deployment - demo

Pulse check - thumbs up or down

Does it make sense?

Spring Cloud Function - under the hood?

- POJO Functions - demo

Pulse check - thumbs up or down

Does it make sense?

Segment 2 - summary

- TBD

Segment 2 - the end!

Questions?

Segment 2 - the end!

BREAK

Segment 3

Spring Cloud Function

and serverless platforms

Spring Cloud Function - AWS Lambda

- Amazon AWS provides several strategies for Request Handlers
  - RequestHandler
  - RequestStreamHandler

```
public interface RequestHandler<I, O> {
    public O handleRequest(I input, Context context);
}
```

What is the problem?

Pearson

Spring Cloud Function - AWS Lambda

- Simple or not you need to know how to implement one.
- Your implementation also becomes AWS specific (what about portability)

Spring Cloud Function - AWS Lambda

- FunctionInvoker
  - Implementation of RequestStreamHandler
  - You only need to know it's fully qualified name and only if you are the deployer/ administrator.

Spring Cloud Function - AWS Lambda

Demo

Spring Cloud Function - AWS Lambda

Questions?

Spring Cloud Function - Microsoft Azure

- Annotation based model

Spring Cloud Function - Microsoft Azure

Demo

Pearson

Spring Cloud Function - Microsoft Azure

Questions?

Segment 3 - summary

- Spring Cloud Functions server less layer provides necessary abstractions to either simplify or completely decouple implementation details from the specifics of the underlying platform

Segment 3 - the end!

Questions?

Segment 3 - the end!

BREAK

Segment 4

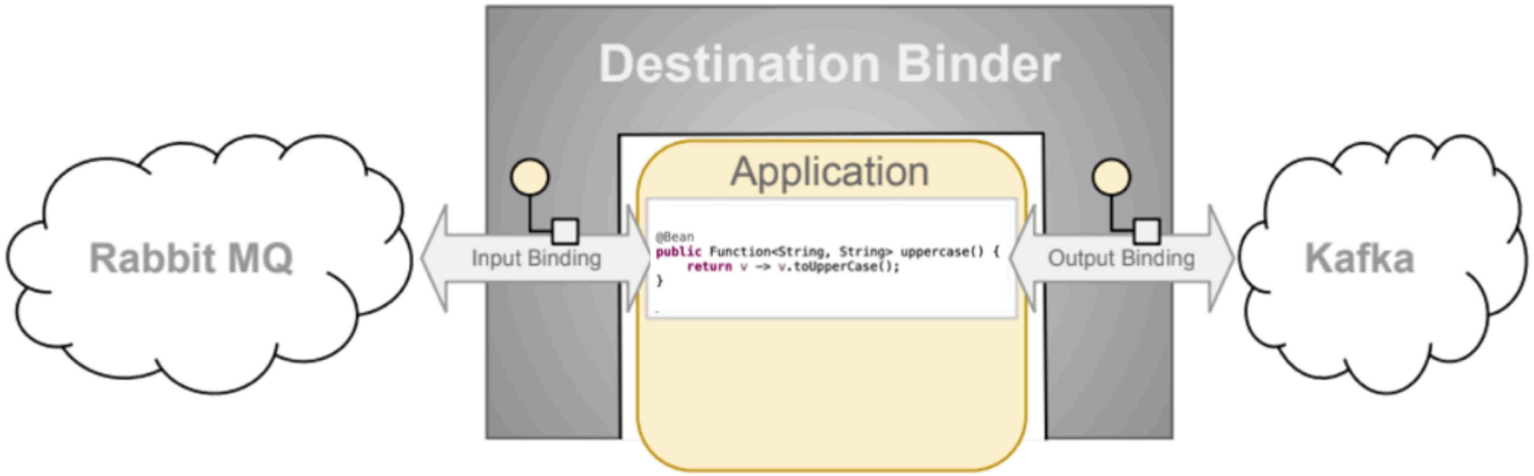
Streaming with Spring Cloud Function

Streaming with Spring Cloud Function?

- What is Spring Cloud Stream?
    - Framework to build highly scalable event-driven and/or streaming microservices.
    - Provides boot-driven integration with Messaging Brokers using Destination Binders
    - Leverages native features of brokers while also providing a workarounds for not supported features.
        - Partitioning and Consumer Groups
        - Message Headers
        - Destination provisioning

Streaming with Spring Cloud Function?

- Spring Cloud Stream



```java
@Bean
public Function<String, String> uppercase() {
    return v -> v.toUpperCase();
}
```

Streaming with Spring Cloud Function?

- Function Binding
  - Naming convention
  - Configuration

Streaming with Spring Cloud Function?

- Multiple Function binding

Streaming with Spring Cloud Function?

- Using function as sources
  - Imperative Supplier
  - Reactive Supplier
  - Reactive Supplier with finite stream

Streaming with Spring Cloud Function?

- Event routing with Routing Function
    - TO function
    - FROM function

Streaming with Spring Cloud Function?

- Advanced reactive streaming with functions
  - Multiple inputs
  - Multiple Outputs

Streaming with Spring Cloud Function?

Demo

Segment 4 - the end!

Questions?

The End!

THANK YOU!