Spring Cloud Functions

write once run anywhere!

SpringOne 2020

Oleg Zhurakousky

Spring Engineering

Spring Cloud Function - agenda

- Scope
- Deep(ish) dive into Spring Cloud Function
- Run anywhere part

Spring Cloud Function - scope

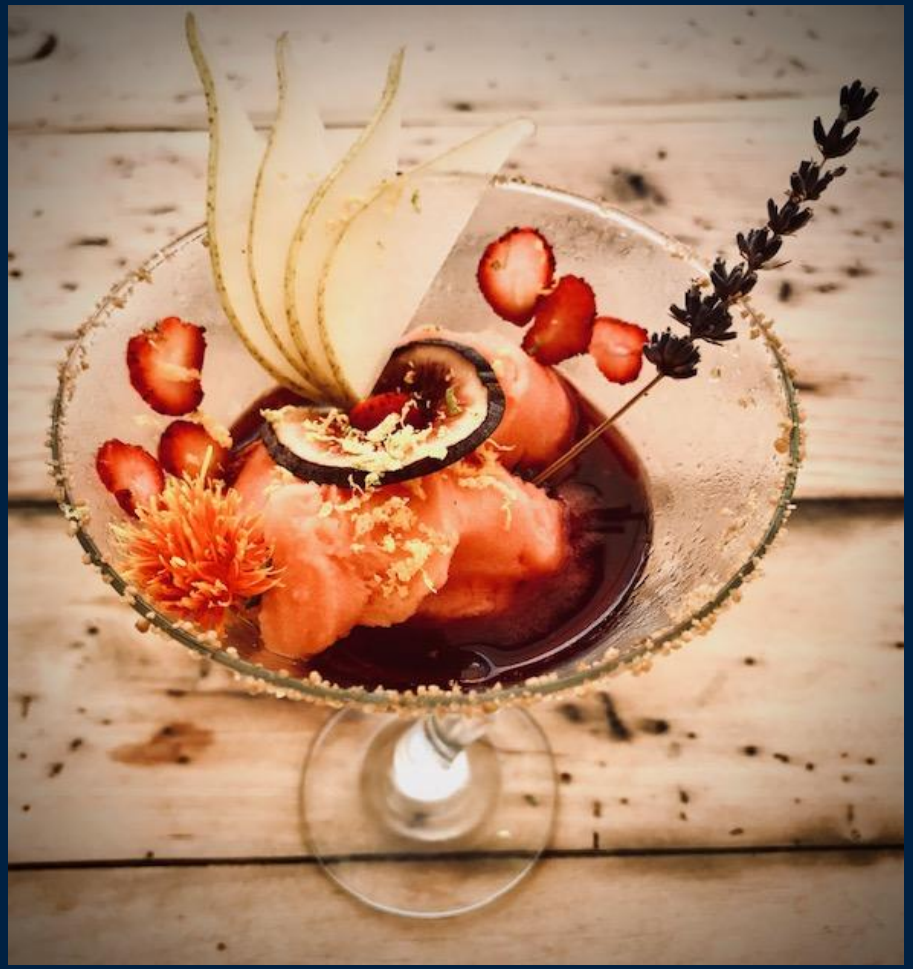
```
java.util.function.Supplier<?>

java.util.function.Function<?,?>

java.util.function.Consumer<?>

POJO function
```

Spring Cloud Function

- Promote implementation of business logic via java Functions
- Uniformed and portable programming model
  - *Transparent Type Conversion*
  - *Function Composition*
  - *POJO functions (if it looks/smells like a function it must be a function)*
  - *Reactive support*
  - *Deployment of packaged functions*
  - *Arity - functions with multiple inputs/outputs*
- Integration with serverless platforms
  - *Amazon AWS*
  - *Google Cloud Functions*
  - *Microsoft Azure*

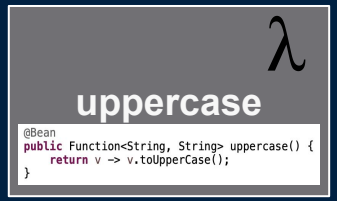
<Demo>

Why Spring Cloud Function



Why Functions?

FUNCTION



Why Functions?

FUNCTION

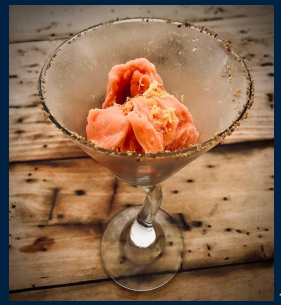
λ

uppercase

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

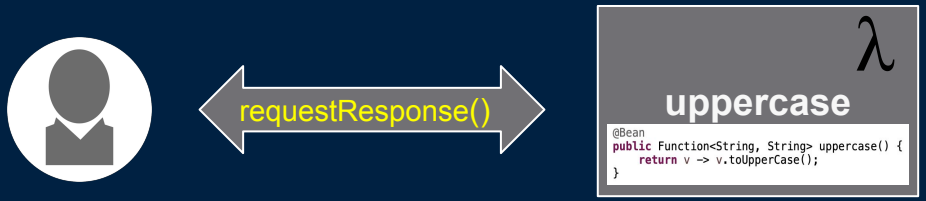
Why Functions?

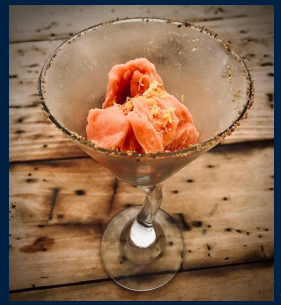
HTTP



Why Functions?

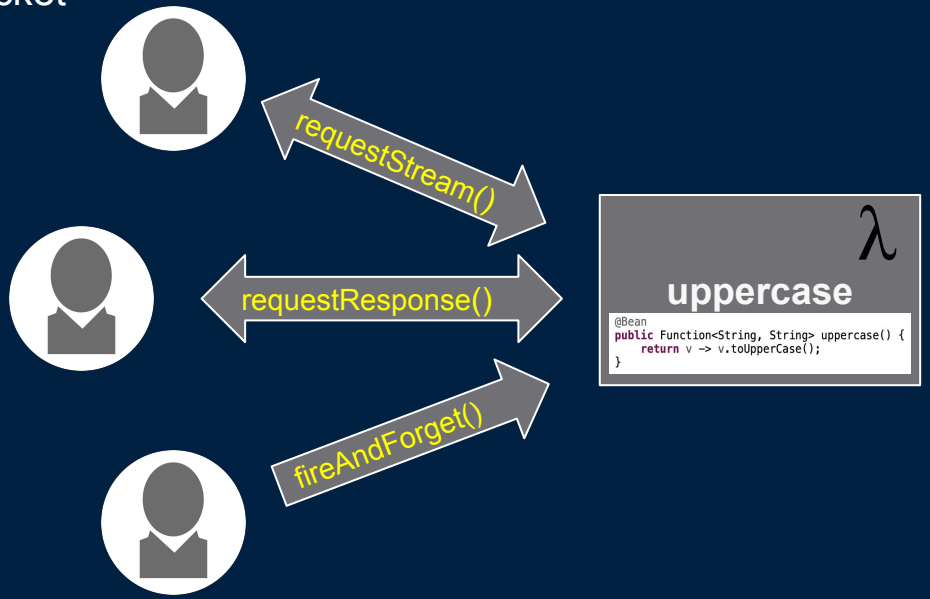
RSocket

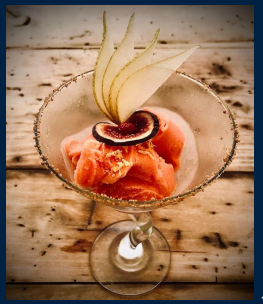




Why Functions?

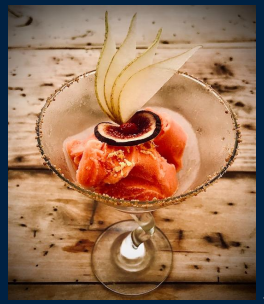
RSocket





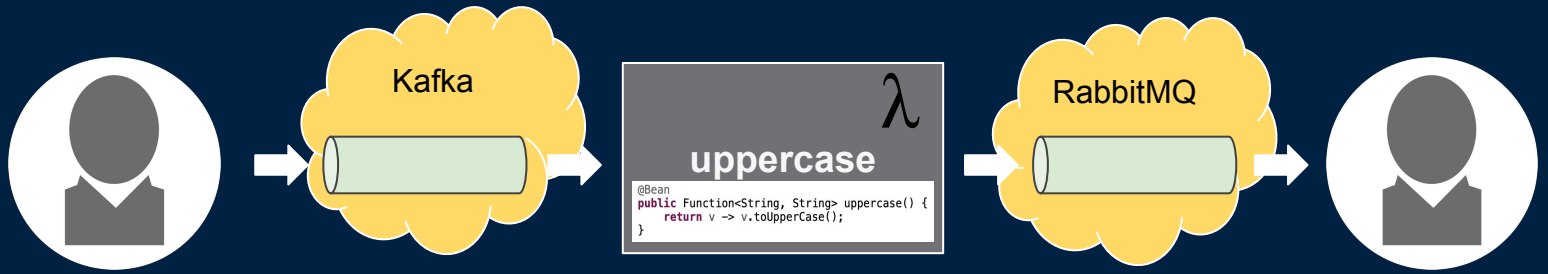
Why Functions?

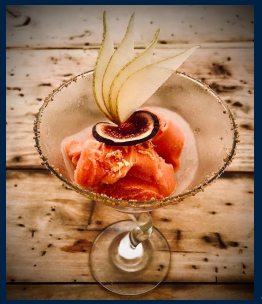
Serverless platforms



Why Functions?

Messaging



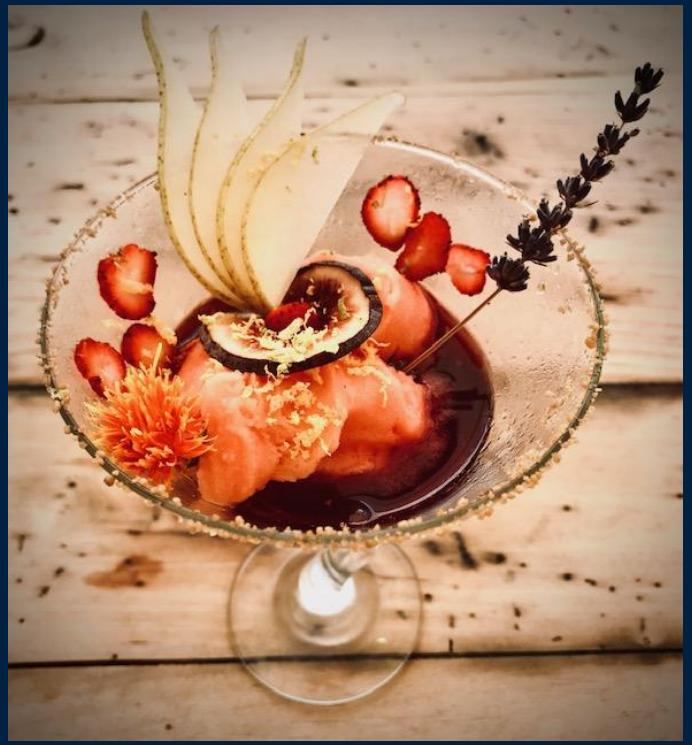


Why Functions?

λ

uppercase

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



<Demo>

Summary!

- Functions are simple, expressive, extensible and portable
- Most if not all requirements could be expressed with functions.
- Testing your functional application becomes very simple - you only need to test the function itself
- And list of benefits goes on. . .


RECEIPT
No. 367168
$
DOLLARS
Thank You!
DATE
FROM
FOR RENT
FOR
CASH
CHECK
MONEY ORDER
BY
ACCT.
PAID
DUE