Spring Cloud Stream From Zero To Hero

Qaiser Abbasi, codecentric AG

Agenda

- Part 1: Demystify The New World
- Part 2: Immerse Yourself
- Part 3: Lab Time!

http://cloud.spring.io/spring-cloud-stream/

Spring Cloud Stream





Spring Cloud Stream is a framework for building message-driven microservices. Spring Cloud Stream builds upon Spring Boot to create DevOps friendly microservice applications and Spring Integration to provide connectivity to message brokers. Spring Cloud Stream provides an opinionated configuration of message brokers, introducing the concepts of persistent pub/sub semantics, consumer groups and partitions across several middleware vendors. This opinionated configuration provides the basis to create stream processing applications.

By adding @EnableBinding to your main application, you get immediate connectivity to a message broker and by adding @StreamListener to a method, you will receive events for stream processing.

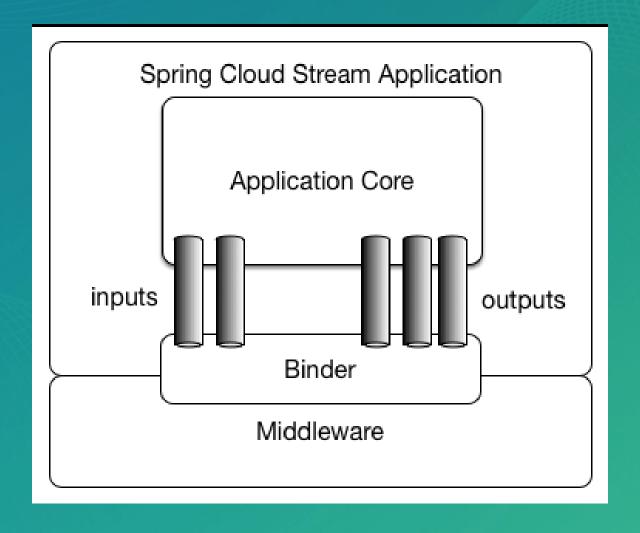
QUICK START

Buzzword Bingo Quick Context!

Spring Cloud Stream

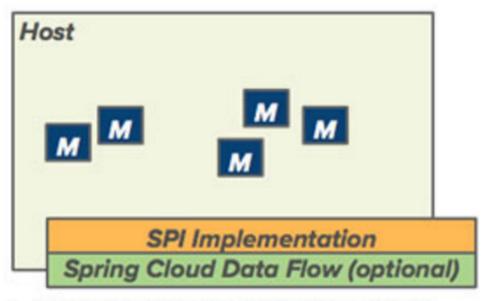
Messaging microservices with Redis, Rabbit or Kafka. Simple declarative model to send and receive messages in a Spring Cloud app.

10.000ft View



Conceptually

Spring Cloud Stream



- Modules are executable Boot apps
- Easier to use Spring Cloud features
- Easier to test, consistent dev experience
- Portable and "cloud-native"

Spring Cloud Stream - Quo Vadis?

From Modules to Microservices

Admin

Modules

Container

Spring Cloud Data Flow

- REST API, Shell, UI
- Module Deployer SPI:
 - Singlenode
 - Lattice
 - YARN
 - Cloud Foundry

Spring Cloud Stream Modules Spring Cloud Task Modules

- OOTB modules
- Executable Boot Apps

Spring Cloud Stream

Spring Cloud Task

- Boot for Spring Integration and Batch
- Auto-Configuration, Bindings



Unless otherwise indicated, these slides are O 2013-2015 Fivoral Software, Inc. and Rocased under a Creative Commons Attribution-NonCommercial Idease: http://icreativecommons.org/licenses/by-no/3.0/



Real World Code - Walkthrough

Hands-On "Twitter Streaming"

Process Twitter Messages In Real-Time

Ingredients

- Project Setup
- Kafka Binder Configuration
- Programming Model
- JSON Processing
- Stream Testing

Recipe

- 1. Source: Ingest Live Twitter Stream
- 2. Processor: Transform Into Domain Message
- 3. Sink: Log Incoming Messages

GOTO https://goo.gl/qq3Zr7;

Feedback

Any questions left?

Anyway: Tks for your patience;)

The End

Copyright 2016

⊘ codecentric