

# Java 微服务实践

## Spring Cloud 消息总线

小马哥



# Java 微服务实战系列课堂

- 讲师信息

小马哥，一线互联网技术专家，国内微服务技术客串讲师，目前主要负责微服务技术推广、架构设计、基础设施、迁移等。重点关注云计算、微服务以及软件架构等领域。从事十余年Java EE 开发，期间通过SUN Java (SCJP、SCWCD、SCBCD) 以及Oracle OCA等的认证。

- 微信/微博/SF: [mercyblitz](#)

- 课程详情: <http://sf.gg/n/1330000009887617>

- 课件资源: <https://github.com/mercyblitz/segmentfault-lessons/>

- JSR资源: <https://github.com/mercyblitz/jsr>





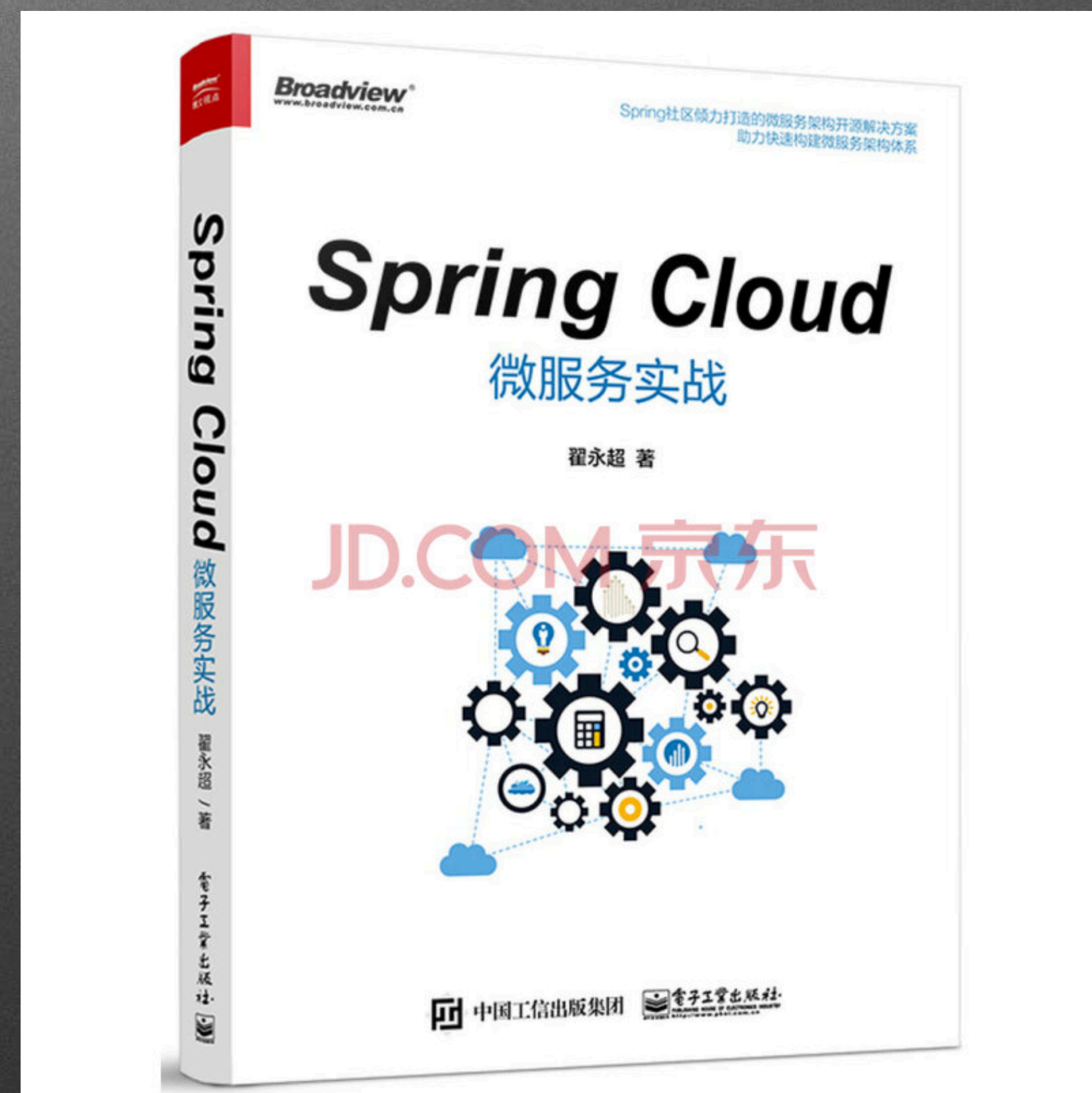
# 小马哥新书

- 《Spring Boot 编程思想》
  - 上市时间：预计 2018 年初
  - 特点：
    - 系统学习，拒绝浅尝辄止
    - 重视规范，了解发展趋势
    - 场景分析，掌握技术选型
    - 源码解读，理解设计思想
    - 实战演练，巩固学习成果



# 书籍推荐

- 《Spring Cloud 微服务实战》
- 作者：翟永超





# 主要议题

- 技术回顾
- Spring Cloud Bus
- Spring Cloud 整合
- 问答互动
- 下期预告



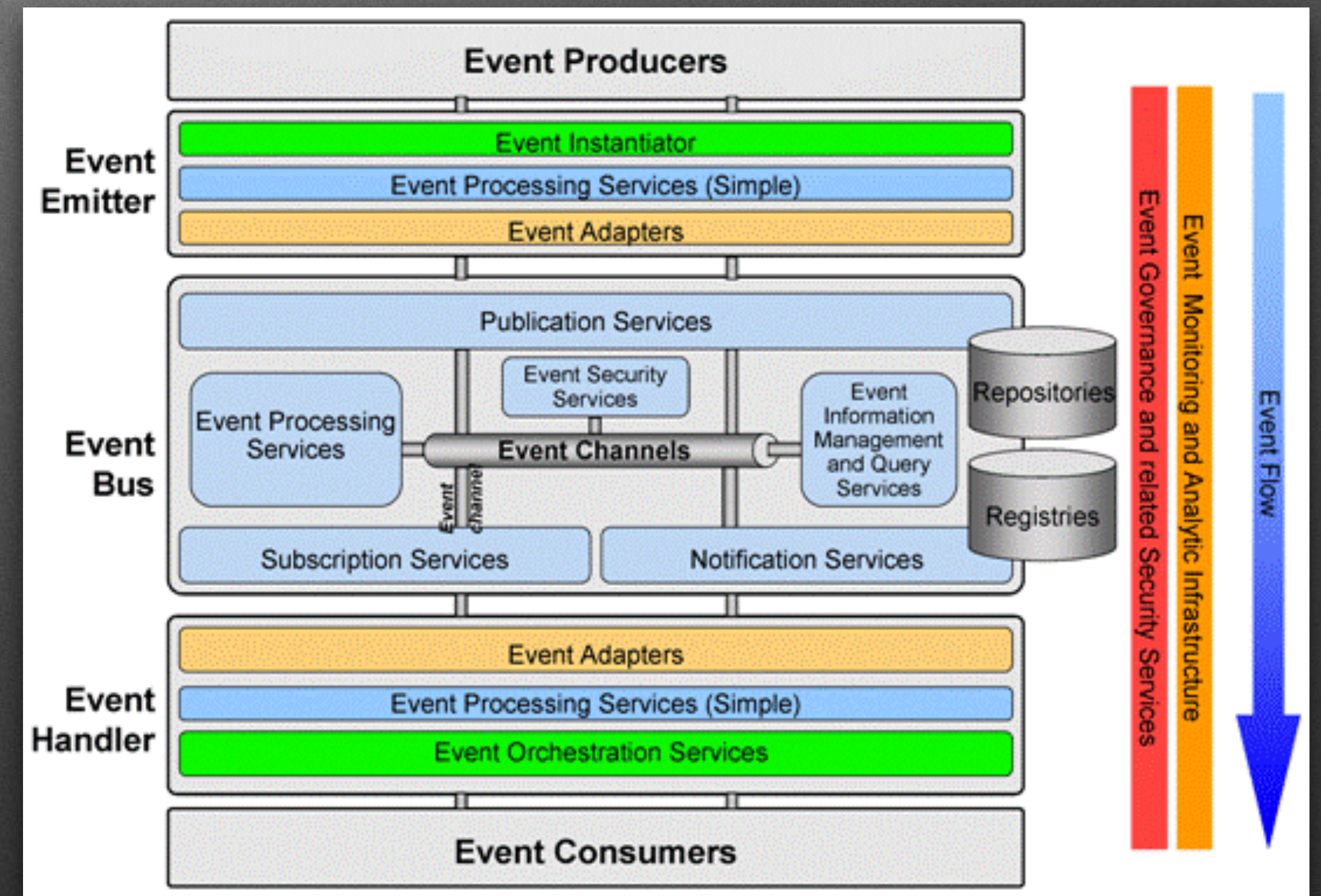
# 技术回顾



# 事件驱动架构 (EDA)

Event-driven architecture (EDA), is a software architecture pattern promoting the production, detection, consumption of, and reaction to events.

This architectural pattern may be applied by the design and implementation of applications and systems that transmit events among loosely coupled software components and services. An event-driven system typically consists of event emitters (or agents), event consumers (or sinks), and event channels.





# Spring 事件/监听器

- Spring 事件
  - ApplicationEvent
- Spring 监听器
  - ApplicationListener / @EventListener
- Spring 事件发布者
  - ApplicationEventPublisher



# Spring Cloud Stream

- 编程模型
  - 激活 : @EnableBinding
    - @Configuration
    - @EnableIntegration
  - Source
    - @Output
    - MessageChannel



# Spring Cloud Stream

- 编程模型
  - Sink
    - @Input
    - SubscribableChannel
    - @ServiceActivator
    - @StreamListener



# Spring Cloud Bus



# Spring Cloud Bus

- 使用场景

用于广播应用状态变更到分布式系统中的各个关联的节点。应用节点间不直接相互通讯，而通过消息总线来实现通知。

- 默认实现

- AMQP (Rabbit MQ)
- Kafka



# Spring Cloud Bus

- 激活总线
  - AMQP
    - spring-cloud-starter-bus-amqp
  - Kafka
    - spring-cloud-starter-bus-kafka
  - spring-cloud-bus



# Spring Cloud Bus

- 总线事件端点
  - 更新 Spring Environment
    - EnvironmentBusEndpoint: /bus/env
- 刷新应用配置
  - RefreshBusEndpoint: /bus/refresh



# Spring Cloud Bus

- 总线事件传播
  - 单点传播
    - Endpoint: `/bus/refresh?destination=${destination}:*`
  - 集群传播
    - Endpoint: `/bus/refresh?destination=${destination}:**`



# Spring Cloud Bus

- 总线事件跟踪
  - TraceEndpoint: /trace
- 总线内部事件
  - EnvironmentChangeRemoteApplicationEvent
  - RefreshRemoteApplicationEvent
  - AckRemoteApplicationEvent



# Spring Cloud Bus

- 自定义事件
  - 事件扩展
    - RemoteApplicationEvent
  - 事件扫描
    - @RemoteApplicationEventScan
- 事件序列化



# Spring Cloud Bus

- 源码分析
  - 自动装配
    - BusAutoConfiguration
  - 本地事件/监听
  - Stream 事件/监听
  - 事件匹配



# Spring Cloud 整合



# Spring Cloud 整合

- 整合消息总线：Spring Cloud Bus
- 整合消息服务：Spring Cloud Stream
- 整合服务调用：Feign
- 整合负载均衡：Netflix Ribbon
- 整合服务短路：Netflix Hystrix
- 整合服务发现：Netflix Eureka
- 整合配置服务器：Config Server
- 整合配置客户端：Config Client



# 问答互动



# 下期预告

Spring Cloud 分布式应用跟踪



谢谢观看