

Camel 4

O que esperar?

Sobre

Otavio R. Piske

- PSE @Red Hat
- PMC no projeto
Apache Camel

Online

- Github: [@orpiske](#)
- Twitter: [@otavio021](#)
- angusyoun@gmail.com

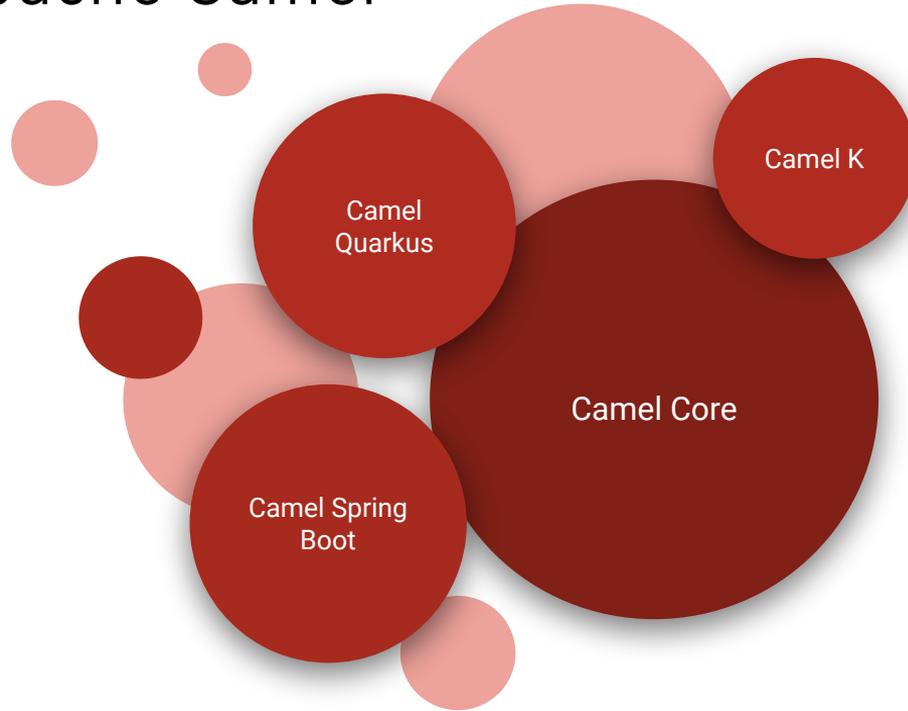
Real Life

- Brno - Rep. Checa

Agenda

- Overview
- Motivações e objetivos
- Mudanças por debaixo do capô
- Planejando seu upgrade

Projeto Apache Camel



Projeto Apache Camel

```
public class KafkaToSQSRoute extends RouteBuilder {  
  
    @Override  
    public void configure() throws Exception {  
        from("kafka:cloud-conf-days")  
            .to("aws-sqs://obrigado");  
    }  
}
```

Quer **aprender** mais? Palestra do Vinícius Ferraz no CCD:
Desenvolvendo e conectando aplicações com Camel e Quarkus.

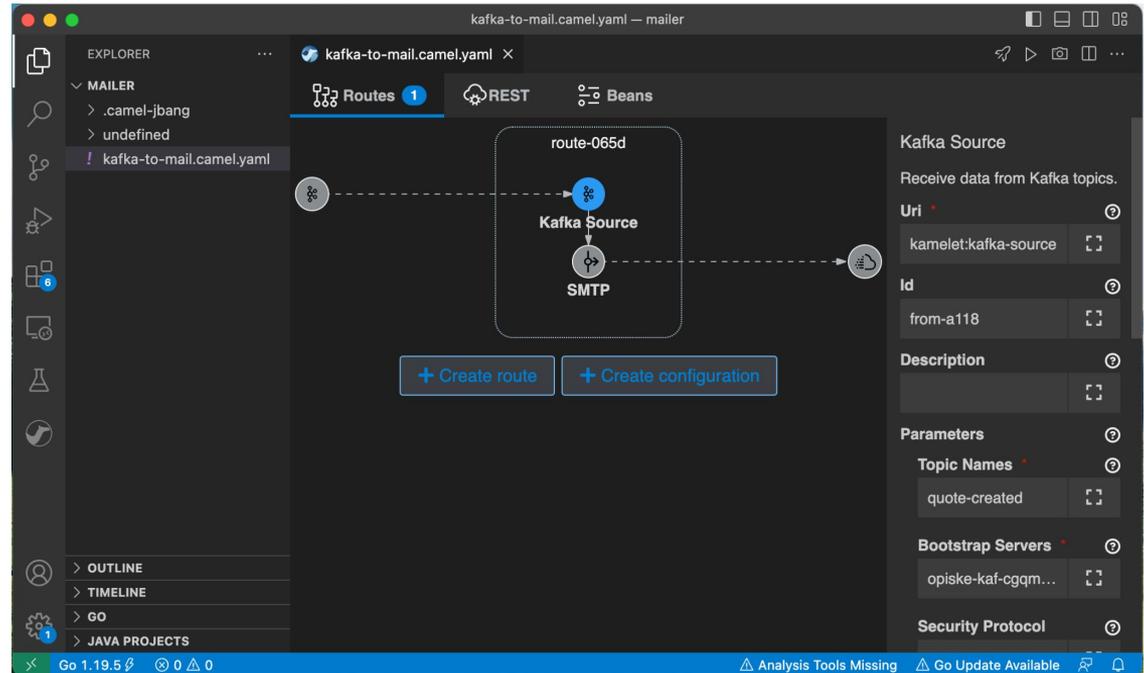
Camel JBang

- Rapidez para criar rotas
- Ideal para protótipos
- Simplifica a criação de novos projetos
- Provê funcionalidades para outros projetos

```
$ camel run my-integration.yaml
```

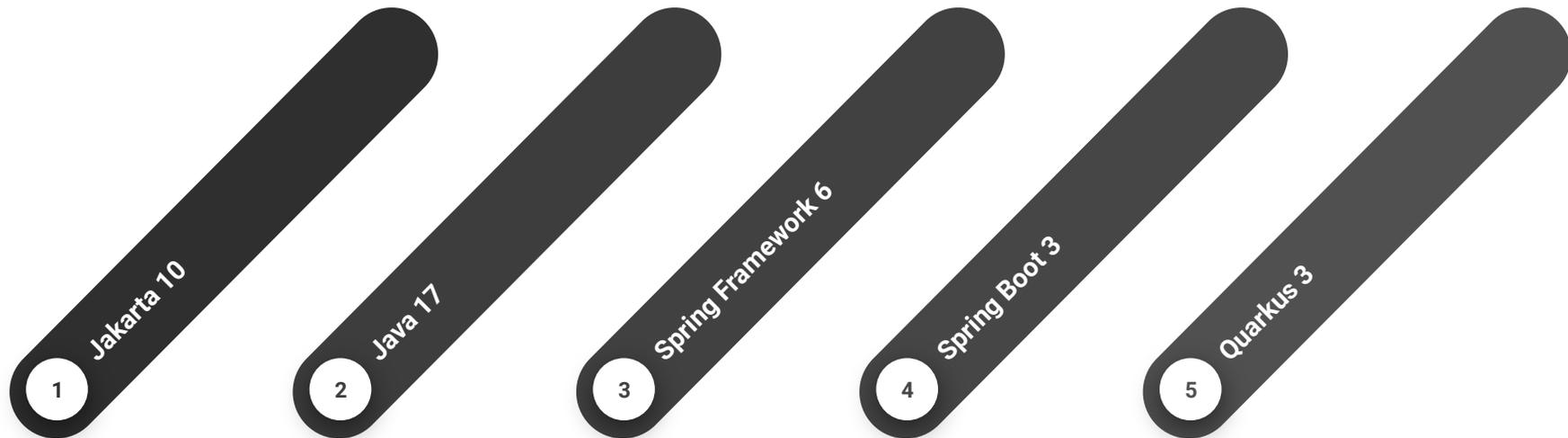
Camel Karavan

- Ferramenta visual para criar rotas
- Extensão para o VS Code



Motivações e Objetivos

Motivações para o Camel 4



Criar pilares para o futuro

- Suportar funcionalidades do Java 17
 - Java 11?
- Mirar no futuro

Simplificar o processo de build e release

- Camel Karaf (OSGI) como "melhor esforço"
- Desacoplar os releases do Camel Karaf do Camel Core

Reduzir o esforço de manutenção

- Limpezas internas
 - APIs internas depreciadas
- Remover o suporte a MDC
- Remover legado
 - Suporte a JUnit 4
- Remover componentes depreciados

Componentes depreciados

- Removidos temporariamente ou permanentemente
 - 33 componentes afetados
 - 20 tem alternativas fáceis (ex.: camel-activemq → camel-jms)
 - A grande maioria pouco utilizados

**Por debaixo do
capô**

Mudanças internas

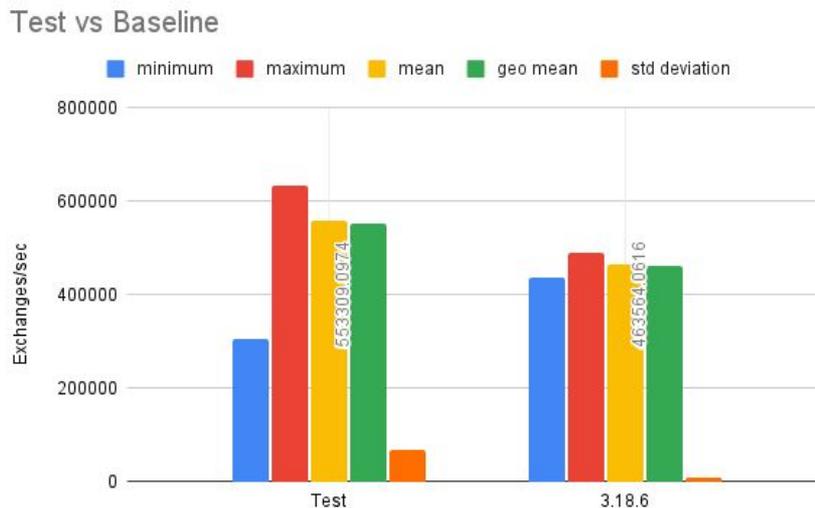
- Internal Plugin Manager
 - CamelContext
 - ExtendedCamelContext
- IntrospectionSupport de uso privado
- Limpeza e manutenção nos testes
 - Builds limpos em power e s390x
 - Extensão do JUnit para o CamelContext
- Software Bill of Materials (SBOM)

Melhorias de desempenho

- JDK-8180450
 - Para saber mais:
 - [Quarkus Runtime: A Peek Into JVM Internals](#)
 - [Seeing through hardware counters](#)
 - [Cracking the Scalability Wall](#)
- Otimizações
 - Seda (pronto)
 - Disruptor (pronto)
 - Core (in progress)
- Outras micro-otimizações

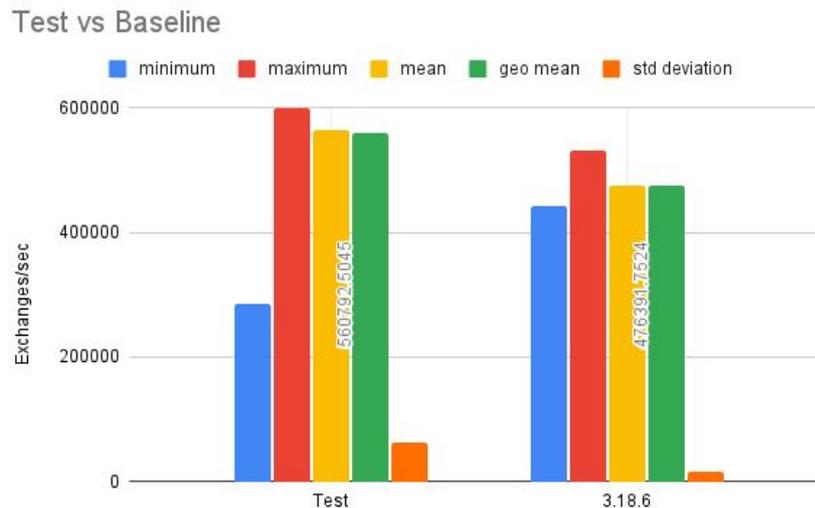
Melhorias de performance: SEDA

- Cenário com baixa contenção
 - 2 consumidores
 - 1 produtor
- Corpo do exchange com tipos variados
- Resultados
 - ~19% mais rápido



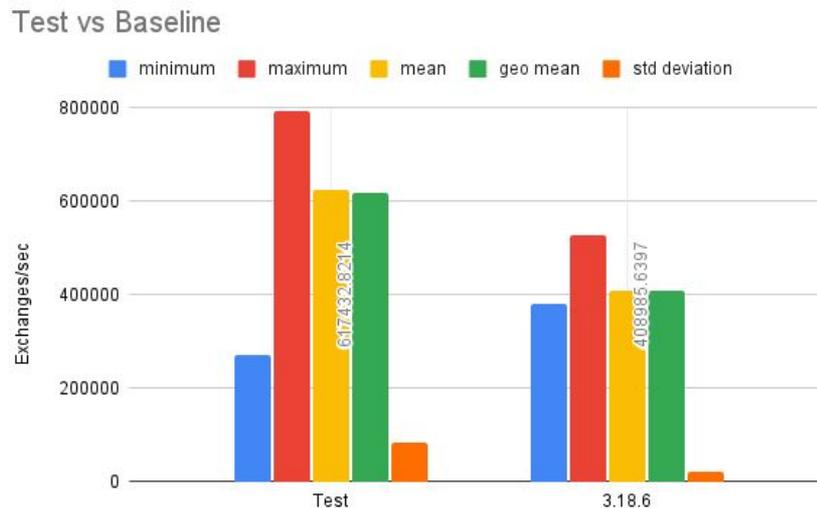
Melhorias de performance: SEDA

- Cenário com baixa contenção
 - 4 consumidores
 - 1 produtor
- Corpo do exchange com tipos variados
- Resultados
 - ~18% mais rápido



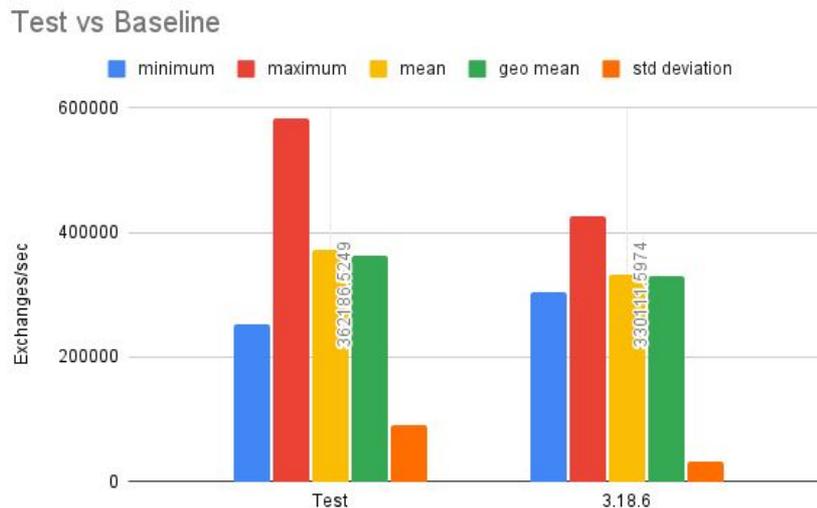
Melhorias de performance: SEDA

- Cenário com média contenção
 - 8 consumidores
 - 1 produtor
- Corpo do exchange com tipos variados
- Resultados
 - ~51% mais rápido



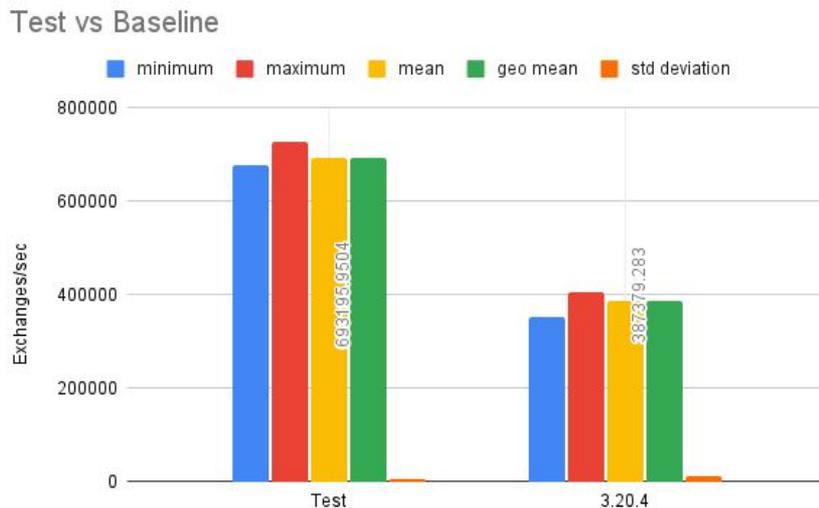
Melhorias de performance: SEDA

- Cenário com alta contenção
 - 16 consumidores
 - 1 produtor
- Corpo do exchange com tipos variados
- Resultados
 - ~10% mais rápido



Melhorias de performance: disruptor

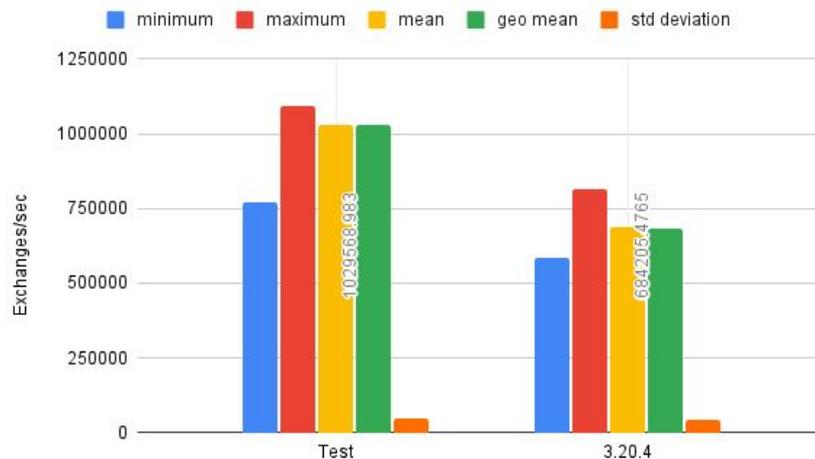
- Cenário com baixa contenção
 - 3 consumidores
 - 1 produtor
- Corpo do exchange com tipos variados
- Resultados
 - ~36% mais rápido



Melhorias de performance: disruptor

- Cenário com média contenção
 - 6 consumidores
 - 5 produtores
- Corpo do exchange com tipos variados
- Resultados
 - ~36% mais rápido

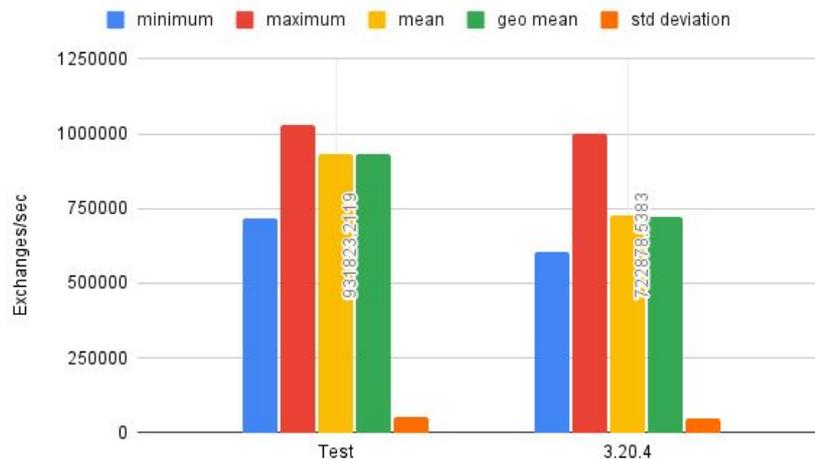
Test vs Baseline



Melhorias de performance: disruptor

- Cenário com alta contenção
 - 9 consumidores
 - 9 produtores
- Corpo do exchange com tipos variados
- Resultados
 - ~29% mais rápido

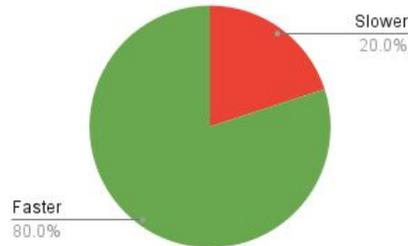
Test vs Baseline



Melhorias de performance: resumo

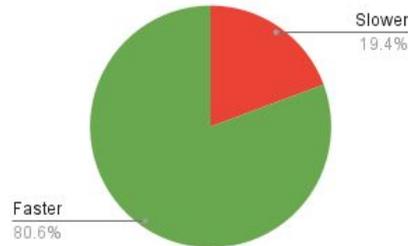
SEDA

- 4.x vs 3.20.4



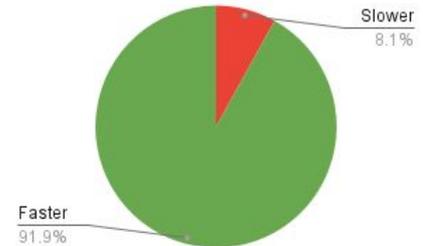
Disruptor

- 4.x vs 3.18.6



Disruptor

- 4.x vs 3.20.4



Planejando o upgrade

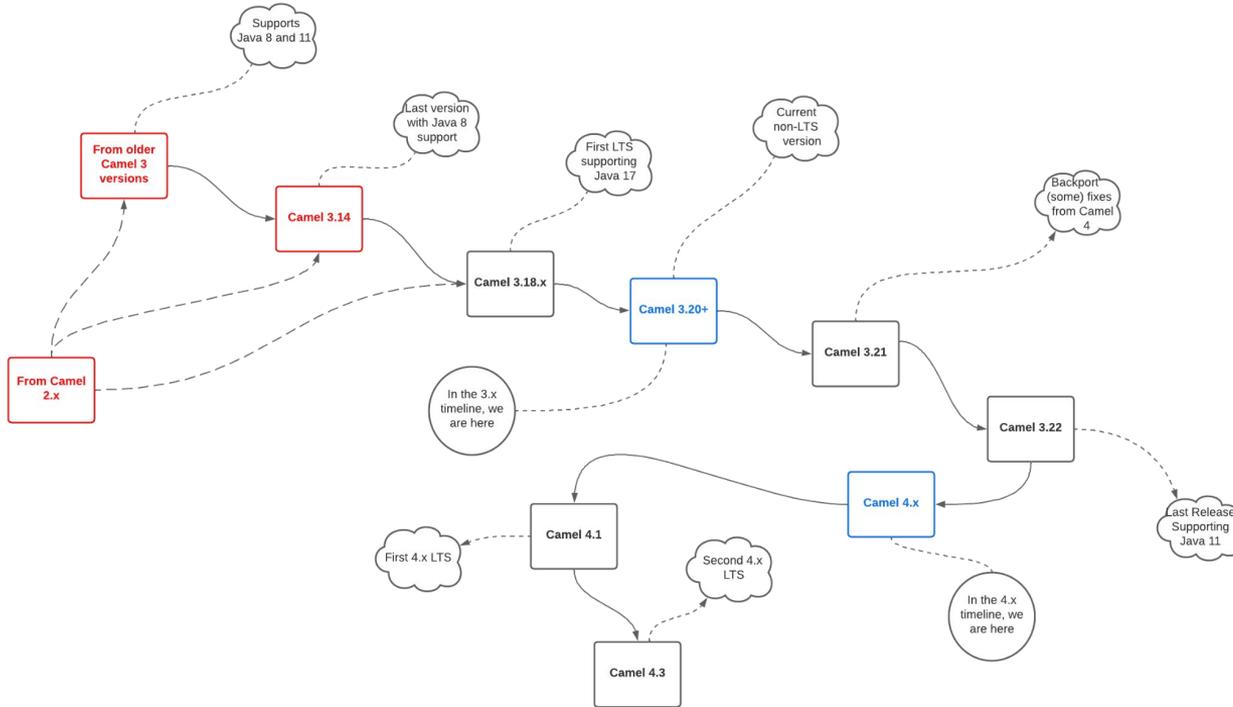
O que esperar do upgrade?

- Migração fácil do Camel 3 para o Camel 4
- Sem incompatibilidades na DSL

Planejando o upgrade

- Downstream / versões comerciais
 - Converse com seu fornecedor!
- Upstream
 - Reserve tempo para testar
 - Contribua com feedbacks para a comunidade
 - Compartilhe o seu conhecimento

O caminho para o Camel 4



Migrando do Camel 3

- Relativamente fácil se:
 - Estiver usando tecnologias/componentes mantidos
 - Estiver usando:
 - Camel Core
 - Camel Spring Boot
 - Camel Quarkus
 - Camel K
 - Uma versão LTS relativamente nova do 3.x

Migrando do Camel 3

- Levemente mais trabalhosa se:
 - Estiver usando uma versão LTS um pouco mais antiga (ex.: anterior a 3.14)
- Precisar  de altera o no c digo se:
 - Usar APIs internas (ExtendedCamelContext, InternalPlugins, etc)
 - Usar OSGI

Áreas com impacto para o usuário

- Logging
 - Slf4j-api foi atualizada para 2.x
- Exchange Patterns
 - InOptionalOut foi removido
- Ajustes na interface **MainListener**
 - Removido o método `configure()`.

Migrando do Camel 2

- Camel 3 foi modularizado
- Mudanças no gerenciamento de contextos
- Java 11
 - Já é antigo!
- Java 8

Comentários finais

- Planejamento é a chave pro sucesso
- Evite versões desatualizadas
- Reavalie padrões e tecnologias desatualizadas
- Compartilhe o conhecimento