Agile Integration with Camel

Otavio R. Piske

Bring the speed back to integration

About

- Otavio R. Piske
  - Twitter: @otavio021
  - E-mail: angusyoung@gmail.com
  - Github: https://github.com/orpiske
- My Work
  - Senior Software Engineer @ Red Hat
  - Committer @ Apache

Agenda

- Introduction
- Camel
- Camel K
- Kamelets
- Camel JBang
- Karavan
- Demo

Introduction

Development is fun ...

We are awesome!

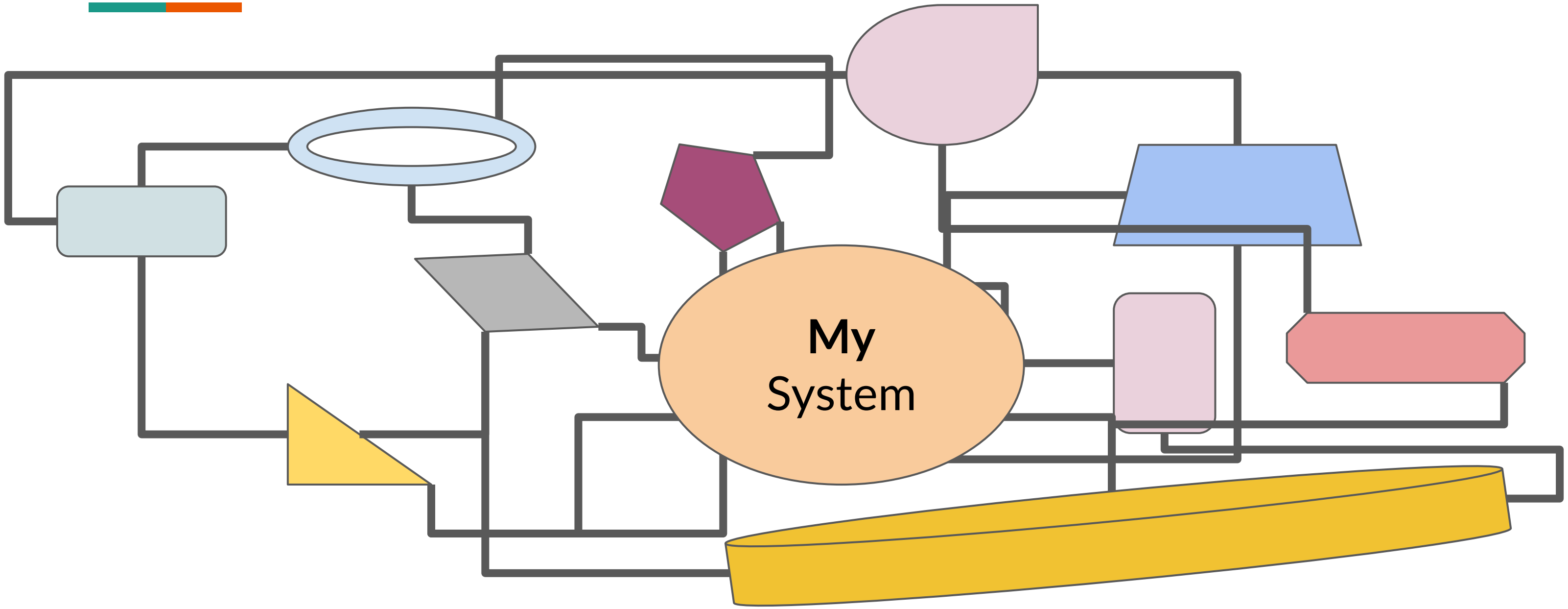
We love **microservices**!

Kubernetes is fun!

Let's use **serverless!**

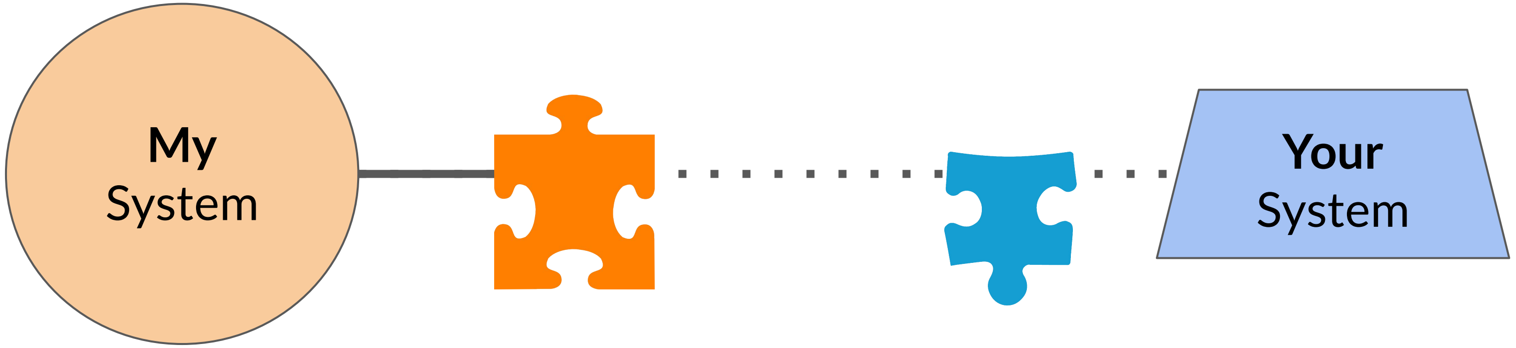
... when we have **freedom**!

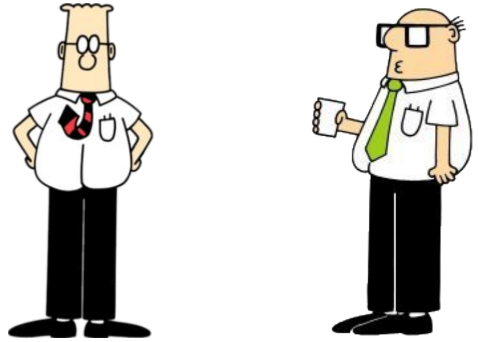
But soon you realize ...



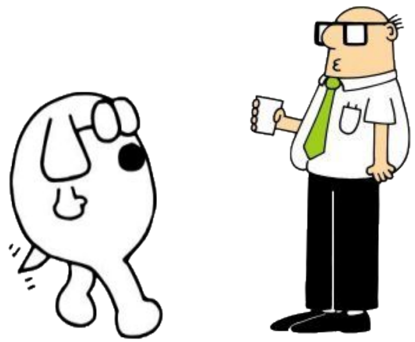
... that your system is **not alone**!

Even with only 2 systems ...



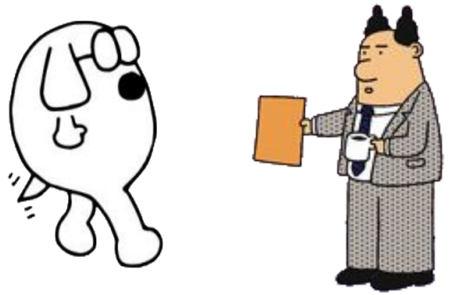


You have to solve a lot of problems



Communication

- Different **communication** models
- Heterogeneous **protocolos** or **messaging systems**
- Different **languages** ou **technologies**
- **Legacy systems** unable to change

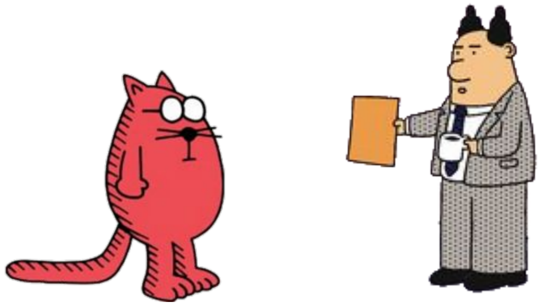


A "social contract"

- Different **business domains**
- Different **data access patterns**
- **Legal** requirements

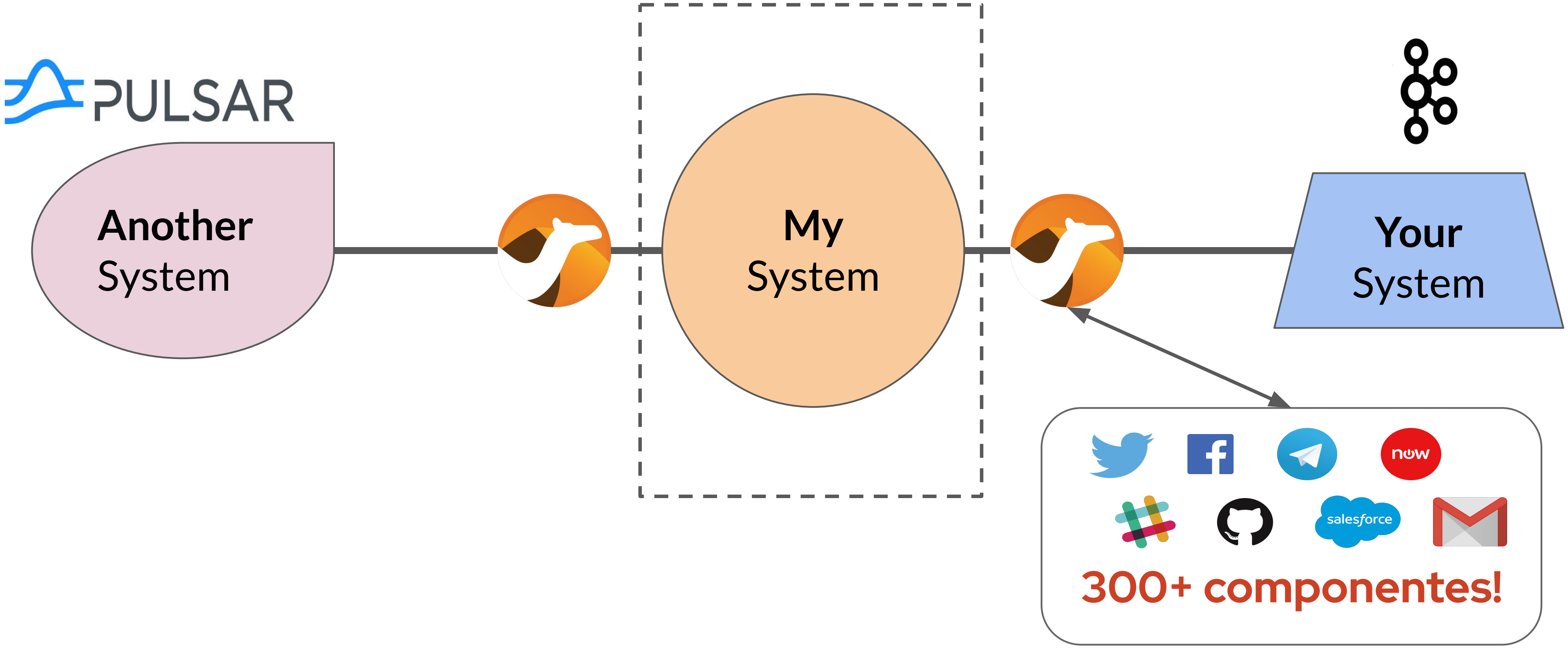
Manage the integration risks

- What is the system or network is **unavailable**?
- How can we guarantee **consistency**?
- How can we guarantee **availability**?



Apache Camel

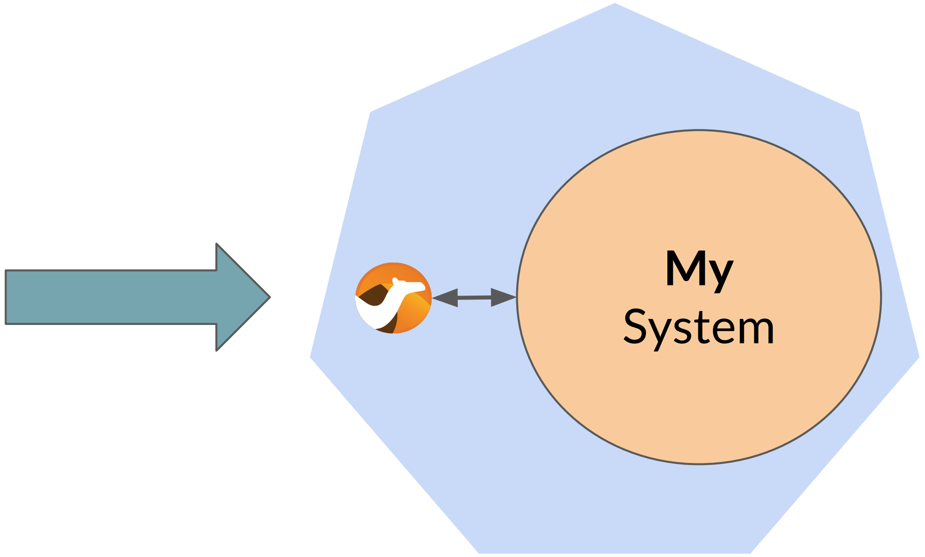
Apache Camel can set you free



Encapsulate complexity into an integration

Apache Camel DSL (***route.java***)

```
from("pulsar://company/nsx/topic1")
   .unmarshal().json()
   .transform().simple("${body[data]}")
   .to("rest:post:api1")
```



Apache Camel K

What is Apache Camel K?

CAMEL K

A **lightweight** integration platform, born on Kubernetes,
with serverless superpowers.

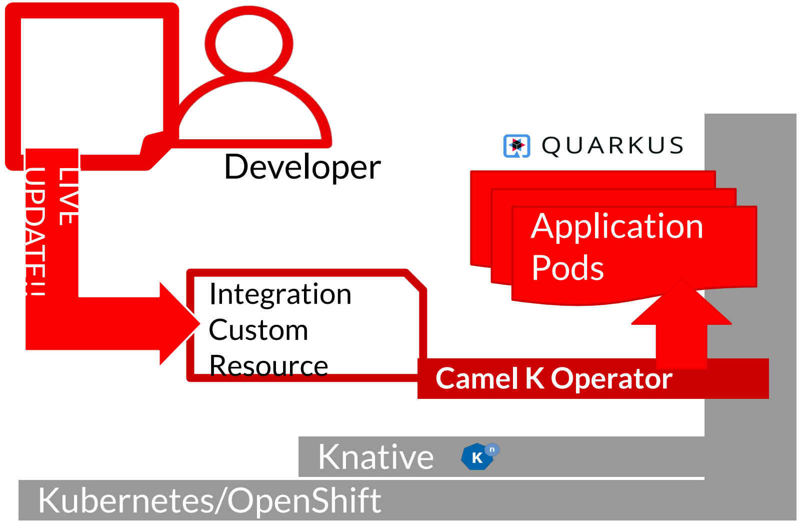
Tooling and ecosystem

- Work with Visual Studio Code or Eclipse Che
    - **Code completion** and **syntax highlighting**
    - Integration **lifecycle management**
    - Inspect **statuses and logs**
- **Open source** code
- **Actively** developed

Camel K architecture

Tailored for a **cloud-native development** experience:

- "Live coding" on the cloud
- Built-in dependency management
- Rapid deployment
- Highly customizable



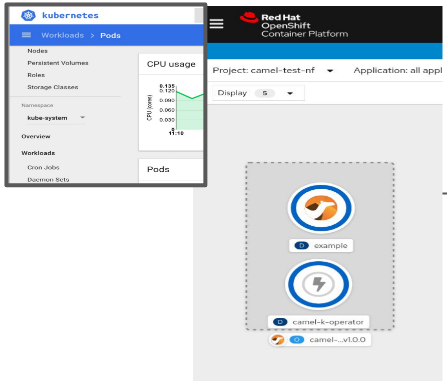
Camel K for developers

```
from("knative:channel/xxxx")
  .transform()...
  .to("kafka:topic")
```

1 **CREATE** an integration file

2 **EXECUTE** the CLI tool

```
$ kamel run integration.java
```



3 **RUN Serverless** on OpenShift or Kubernetes

Camel K and Knative

Camel K loves Knative

Not really the official logo ... but we like it

CAMEL K

A lightweight integration platform, born on Kubernetes, with **serverless superpowers**.

Knative

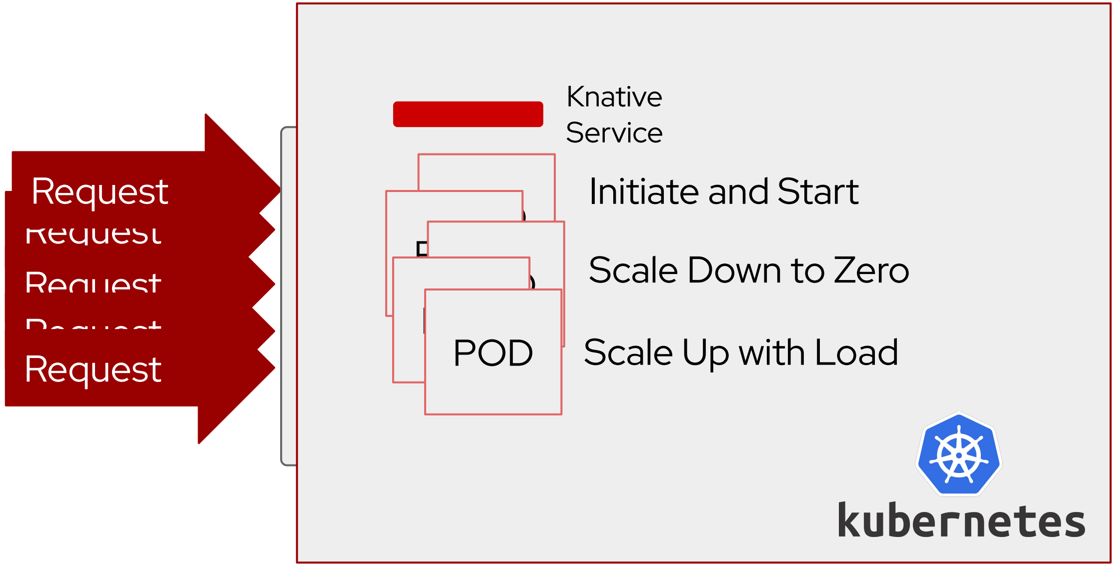
Knative Serving

Auto-scaling and scale-to-zero

Knative Eventing

Infrastructure for event-driven applications

Knative


Request
Request
Request
Request
Request
Knative Service
POD
Initiate and Start
Scale Down to Zero
Scale Up with Load
kubernetes

Kamelets

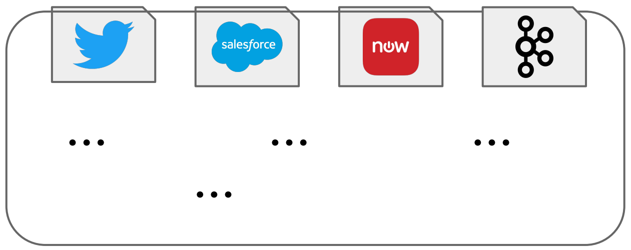
Kamelets

Means: **Kamel** route snipp**ets**

Aims to simplify writing and reusing integrations without writing the DSL.

Native support for the OpenShift Console starting with 4.7

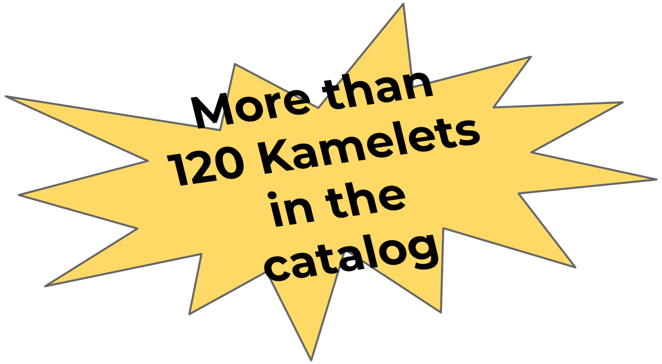
Kamelet Catalog (Kubernetes Objects)



Kamelets in details

Other use cases for Kamelets:

- Camel Core
- Camel JBang
- Visual Tools for Camel K development



1 - Choose the origin

2 - Set the parameters

```
query=quarkus
token=...
```

3 - Pick the destination

Camel JBang

Camel + JBang -> Camel JBang

- Run **self-contained** Java code
- No need to **build**
- No need to **package**

- **Exposes** Camel Core features
- Run simple integrations without **writing code**
- **Search** for components and DSLs in the documentation

Karavan Designer

Karavan

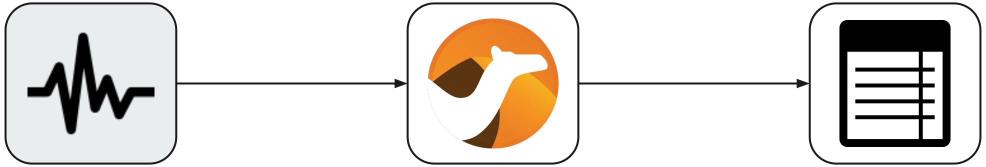
- A **visual** designer for integrations
- Runs on Visual Studio Code
- For the **citizen integrator**
- **Kamelets** integration
- Preliminary support for the integration DSL

Demo

Demo 1: Camel JBang basics

- Show help
- List components
- Search components
- Run simple routes

Demo 2: Earthquake source



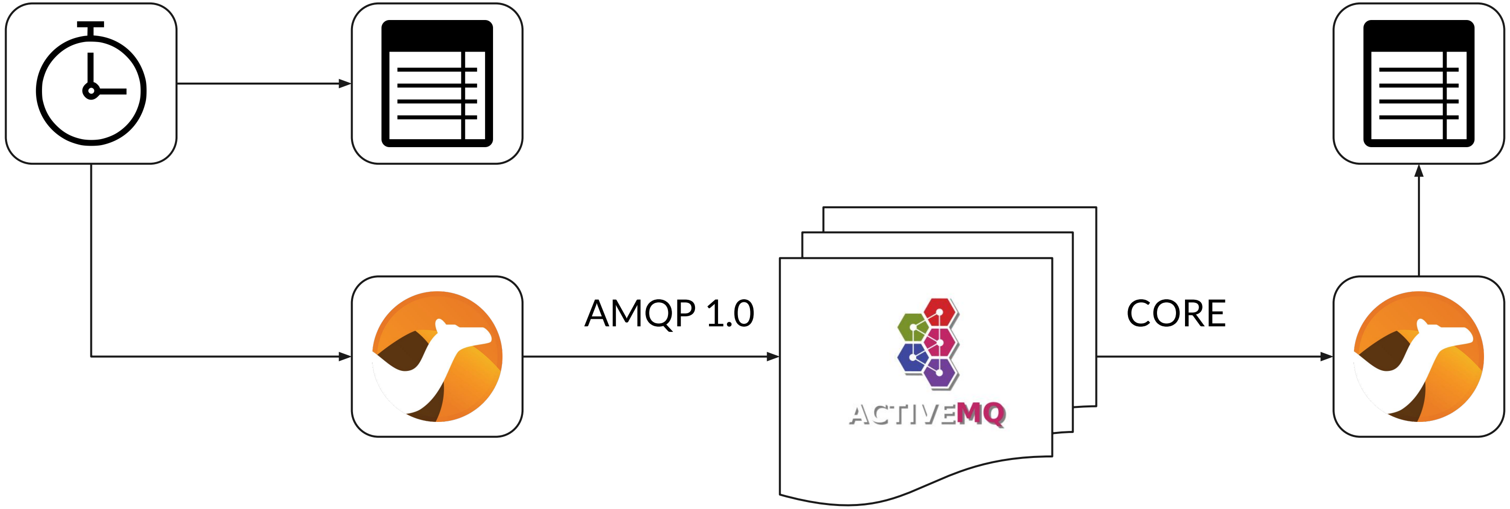
Demo 2: Earthquake source

```
{"type":"FeatureCollection","metadata":{"generated":1636120848000,"url":"
https://earthquake.usgs.gov/fdsnws/event/1/query?format=geojson&updatedaf
ter=2021-11-01T13%3A13%3A23.834&orderby=time-asc","title":"USGS
Earthquakes","status":200,"api":"1.12.3","count":1813},"features":[{"type
":"Feature","properties":{"mag":1,"place":"33 km SW of Alatna,
Alaska","time":1633528948582,"updated":1636063859[illegible]63,"tz":null,"url":"htt
ps://earthquake.usgs.gov/earthquakes/eventpage/ak021ctnadw9","detail":"ht
tps://earthquake.usgs.gov/fdsnws/event/1/query?eventid=ak021ctnadw9&forma
t=geojson","felt":null,"cdi":null,"mmi":null,"alert":null,"status":"revie
wed","tsunami":0,"sig":15,"net":"ak","code":"021ctnadw9","ids":",ak021ctn
adw9,","sources":",ak,","types":",origin,phase-data,","nst":null,"dmin":n
ull,"rms":0.97,"gap":null,"magType":"ml","type":"earthquake","title":"M
1.0 - 33 km SW of Alatna,
Alaska"},"geometry":{"type":"Point","coordinates":[-153.2547,66.3536,12.4
]},"id":"ak021ctnadw9"}
```

Demo 2: Earthquake source

```
- route:
      from: "kamelet:earthquake-source"
      steps:
      - unmarshal:
            json: {}
      - log: "Earthquake with magnitude ${body[properties][mag]} at
${body[properties][place]}"
```

Demo 3: Local broker



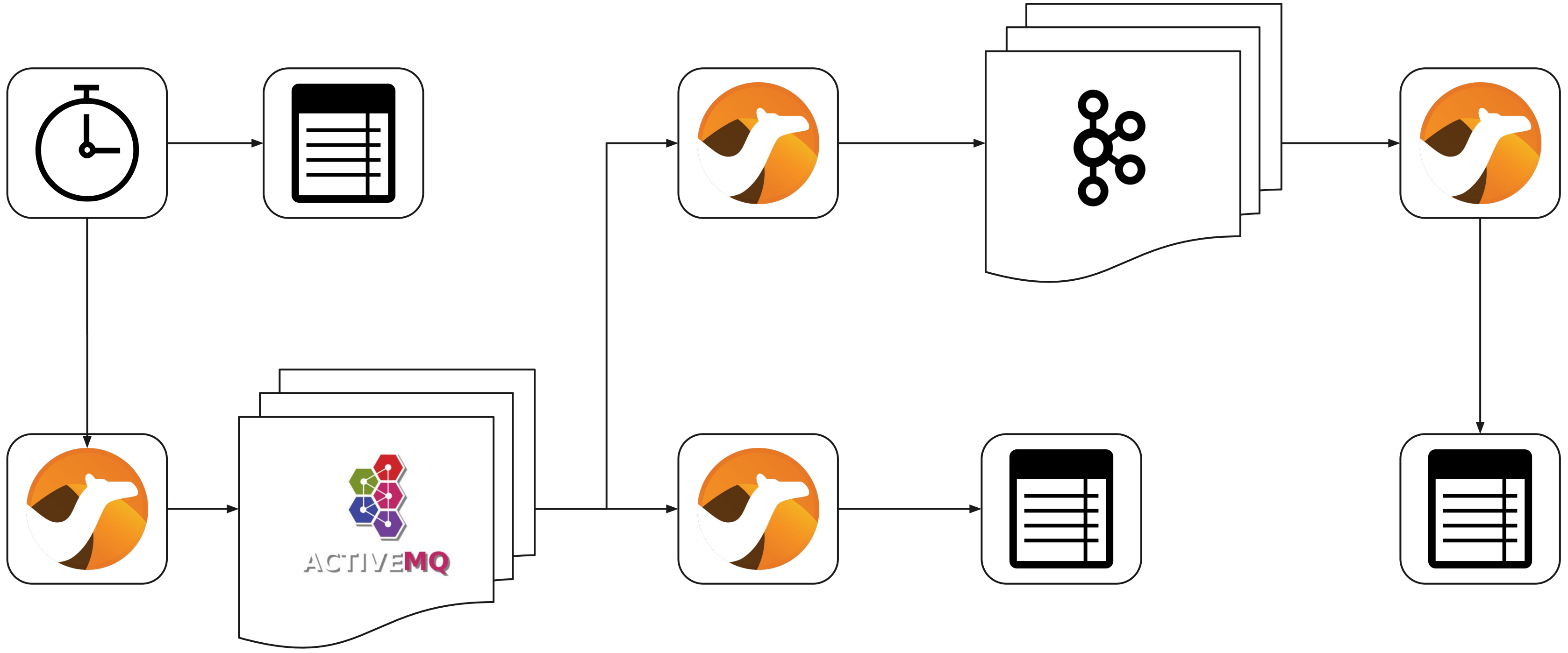
Demo 3: AMQP Sink

```
- route:
     from:
     uri: "kamelet:timer-source"
     parameters:
       period: 1000
        message: "Hello Camel JBang"
     steps:
       - log: "${body}"
       - to:
          uri: "kamelet:jms-amqp-10-sink"
          parameters:
             remoteURI: amqp://localhost:61616
             destinationName: test-queue
```

Demo 3: AMQP Source

```
- route:
      from:
      uri: "kamelet:jms-apache-artemis-source"
      parameters:
        destinationName: test-queue
         brokerURL: tcp://localhost:61616
      steps:
      - log: "${body}"
```

Demo 4: Local Broker with remote Kafka



Closing Comments

Thank you! Obrigado!

1. Use
   a. http://camel.apache.org
2. Participate
   a. https://camel.apache.org/community/
3. Contribute
   a. https://github.com/apache/camel
   b. https://github.com/apache/camel-k
   c. ...
4. Promote!