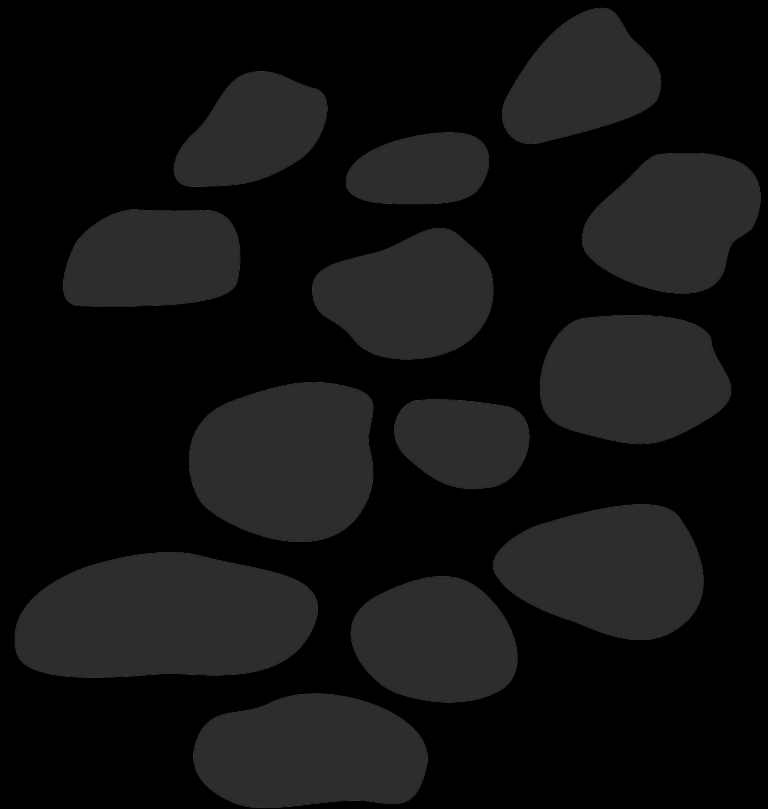


*gepard*ec

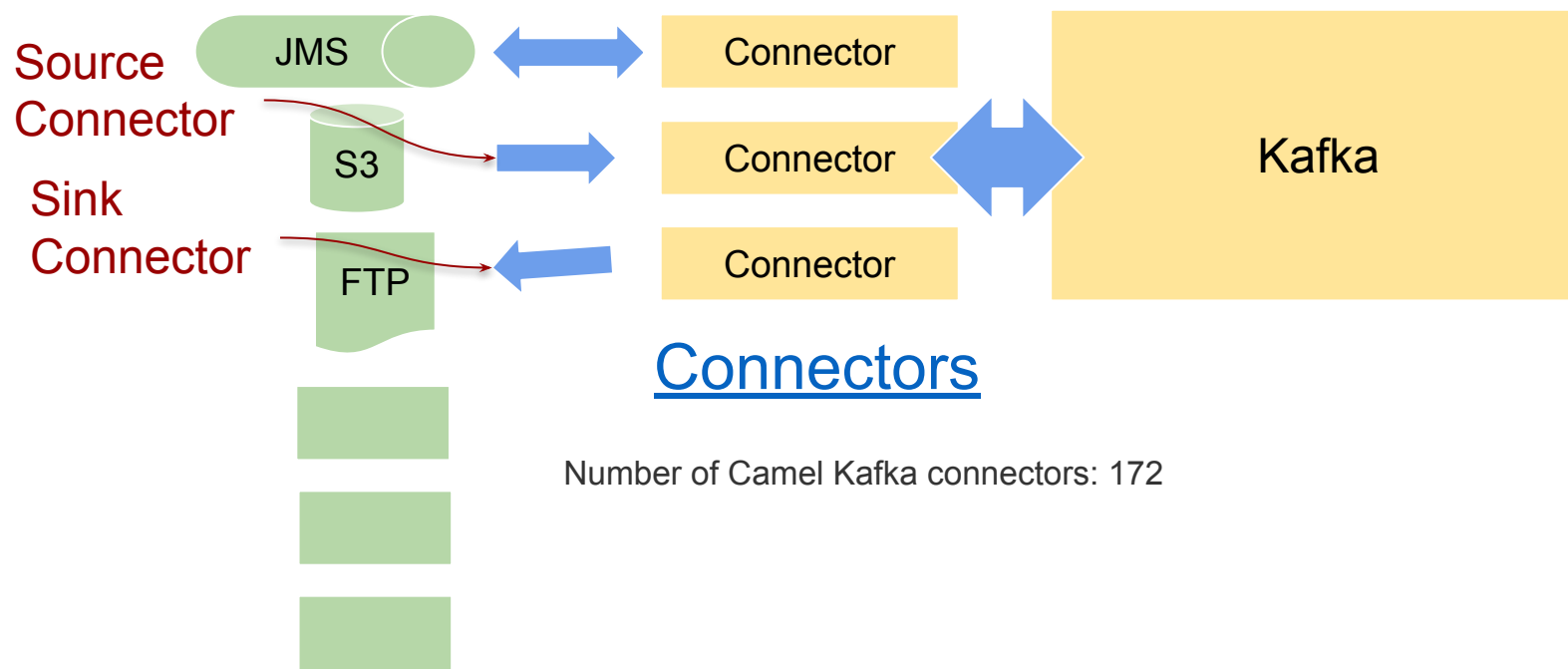
# Kafka Connect



# Why Kafka Connect?

// Exchange data with other systems and different formats

// Camel Kafka Connector allows using all Camel Components with Kafka Connect





# Configuration

// Is configured by a set of properties

// Internally a route is created with following steps

- from
- marshal
- aggregate
- idempotentConsumer
- removeHeaders
- errorHandler
- to

## AWS-S3 Example

```
name=CamelAWS2S3SourceConnector
connector.class=org.apache.camel.kafkaconnector.aws2s3.CamelAws2s3SourceConnector
key.converter=org.apache.kafka.connect.storage.StringConverter
value.converter=org.apache.kafka.connect.converters.ByteArrayConverter

camel.source.maxPollDuration=10000

topics=test1

camel.source.url=aws2-s3://camel-kafka-connector?autocloseBody=false

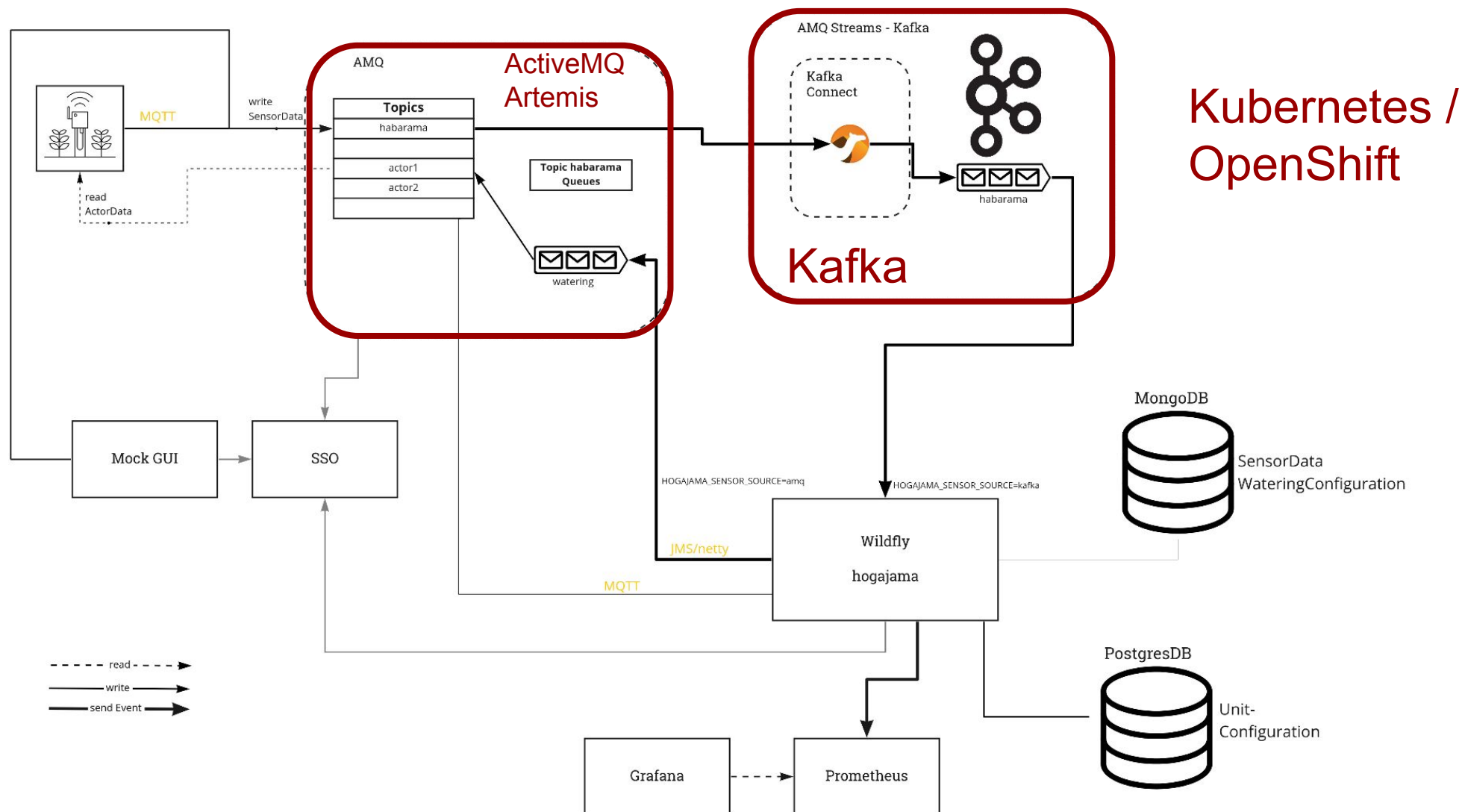
camel.component.aws2-s3.accessKey=xxxx
camel.component.aws2-s3.secretKey=yyyy
camel.component.aws2-s3.region=eu-west-1
```

## Plugins

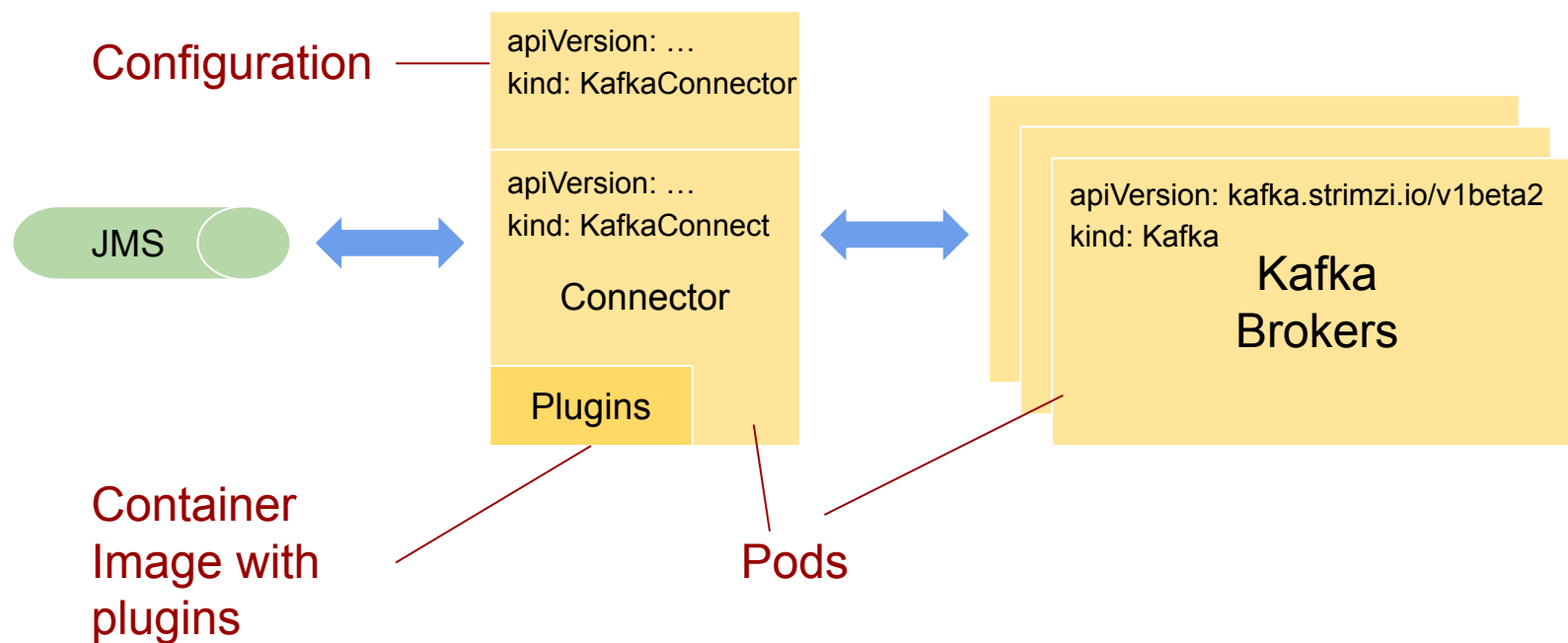
e.g:

camel-aws2-s3-kafka-connector-  
0.11.5-package.tar.gz

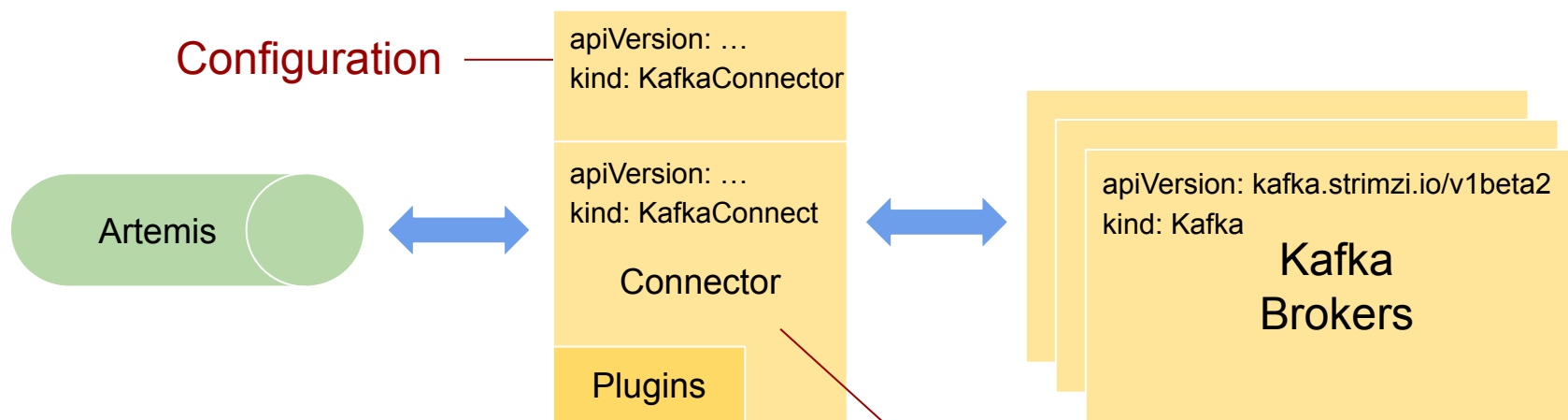
# Learning Friday Project - Hogarama



# Kafka on Kubernetes -> Strimzi



# Kafka on Kubernetes -> Strimzi



## see also

```
VERSION=$1
PLUGINS=("https://repo.maven.apache.org/maven2/org/apache/camel/kafkaconnector/camel-activemq-kafka-connector/1.0.0/camel-activemq-kafka-connector-1.0.0-package.tar.gz"
```

```
"https://repo.maven.apache.org/maven2/org/apache/camel/kafkaconnector/camel-sjms2-kafka-connector/1.0.0/camel-sjms2-kafka-connector-1.0.0-package.tar.gz")
```

```
...
# download the plugins
```

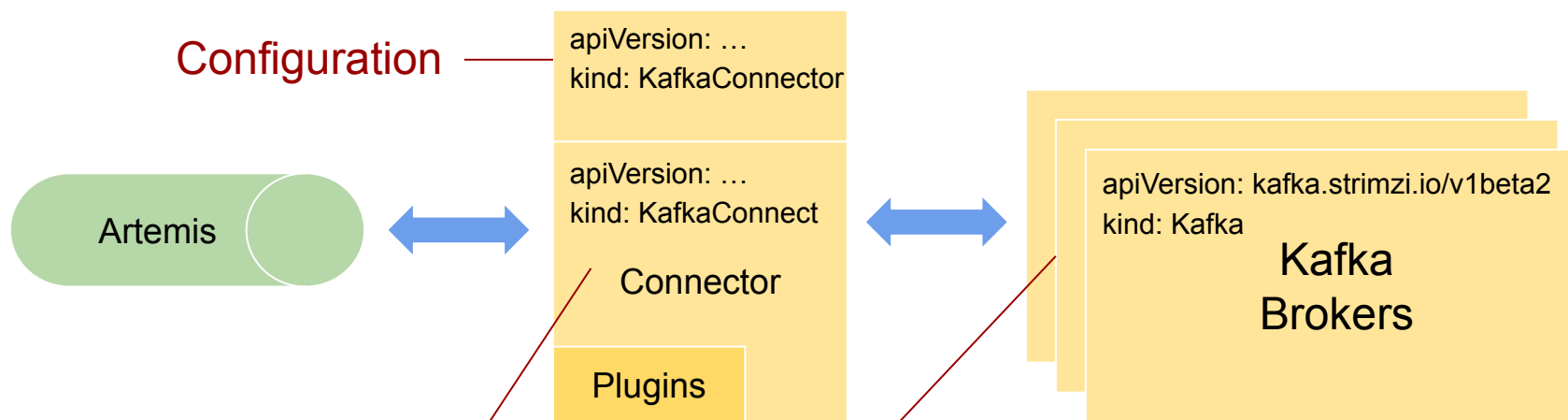
```
...
find my-plugins/ -maxdepth 1 -type f -exec tar -xzf {} --directory my-plugins/ \;
rm my-plugins/*.tar.gz
```

```
docker build -t gepardec/kafka-connect-cluster:${VERSION} .
```

```
FROM registry.redhat.io/amq7/amq-streams-kafka-26-rhel7:1.7.0-4
USER root:root
COPY ./my-plugins/ /opt/kafka/plugins/
USER kafka:kafka
```

## Dockerfile

# Kafka on Kubernetes -> Strimzi



[see also](#)

```
apiVersion: kafka.strimzi.io/v1beta2
kind: KafkaConnect
metadata:
  annotations:
    strimzi.io/use-connector-resources: 'true'
    play.gepaplexx.com/persist: "true"
  name: gepardec-connect-cluster
  labels:
    strimzi.io/cluster: gepardec-kafka-cluster
spec:
  bootstrapServers: 'gepardec-kafka-cluster-kafka-bootstrap:9093'
  image: 'gepardec/kafka-connect-cluster:1.1.0'
```

# Configuration

```
apiVersion: kafka.strimzi.io/v1beta2
```

```
kind: KafkaConnector
```

```
metadata:
```

```
...
```

```
spec:
```

```
class: org.apache.camel.kafkaconnector.sjms2.CamelSjms2SourceConnector
```

```
config:
```

```
value.converter: org.apache.kafka.connect.converters.ByteArrayConverter
```

```
topics: habarama
```

```
camel.component.sjms2.connectionFactory: org.apache.activemq.ActiveMQConnectionFactory
```

```
camel.component.sjms2.connectionFactory.brokerURL: 'tcp://amq.hogarama.svc.cluster.local:61616'
```

```
camel.source.path.destinationType: topic
```

```
camel.source.path.destinationName: habarama
```

```
cleanup.policy: compact
```

```
errors.log.enable: true
```

```
errors.log.include.messages: true
```

## Simple JMS 2

*sjms2:destinationType:destinationName*