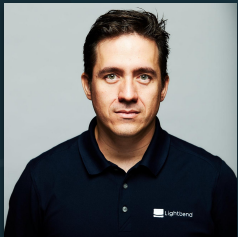


Kubernetes as a Streaming Data Platform

A Federated Operator Approach

Data Council - Barcelona, October 2nd, 2019



Gerard Maas
Principal Engineer, Lightbend, Inc.

@maasg





Gerard Maas

Principal Engineer



gerard.maas@lightbend.com



[@maasg](https://twitter.com/maasg)



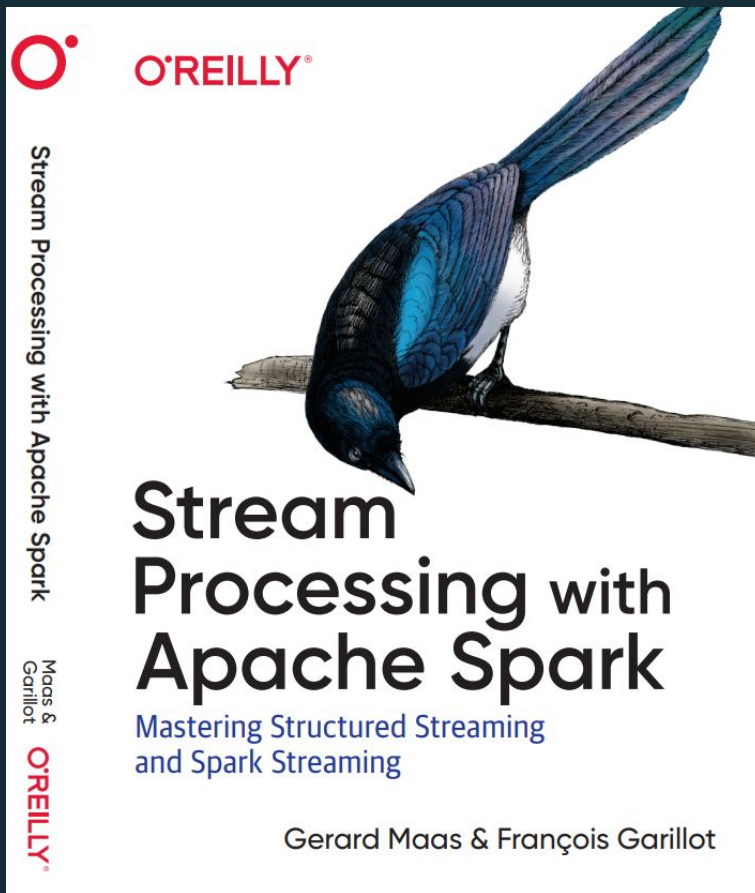
<https://github.com/maasg>



<https://www.linkedin.com/in/gerardmaas/>



<https://stackoverflow.com/users/764040/maasg>





Lightbend



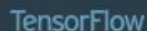
Reactive Microservices



Streaming Engines



Machine Learning



Others

Intelligent Management

Akka Management
SBR and Kubernetes Lease
Akka Multi-DC
Thread Starvation Detector
Config Checker

Monitoring

Lightbend Console
Lightbend Telemetry
Developer Sandbox

Integration

Alpakka Connectors
Telemetry APM Integrations
Akka Persistence Plugins

Security

Akka GDPR
Scala SCA Plugin For Fortify

Data Backplane



Storage Options

HDFS

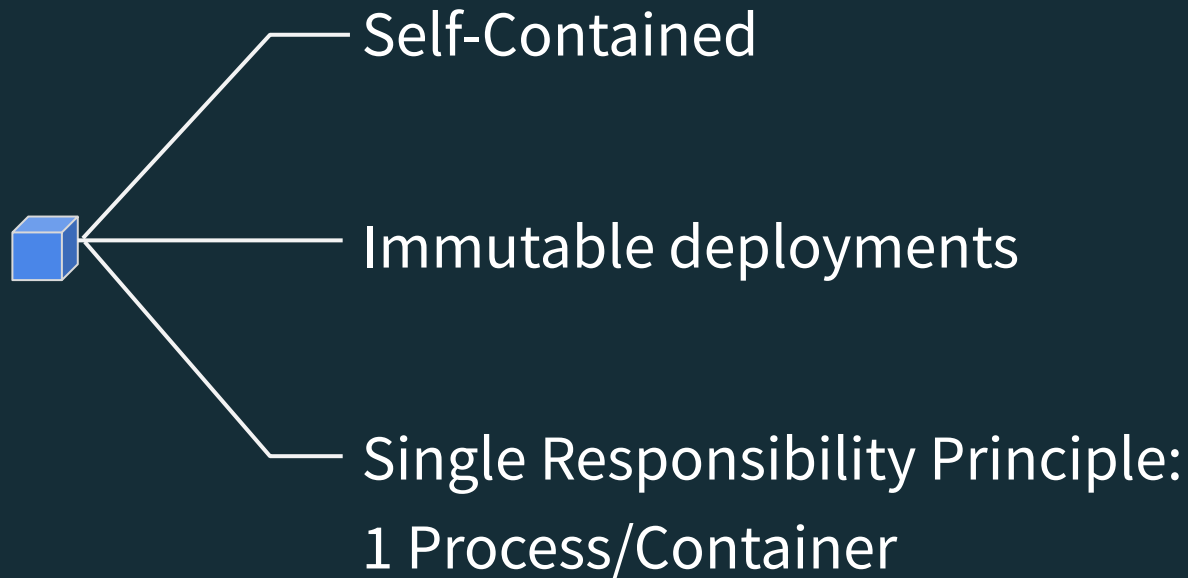
SQL, NoSQL

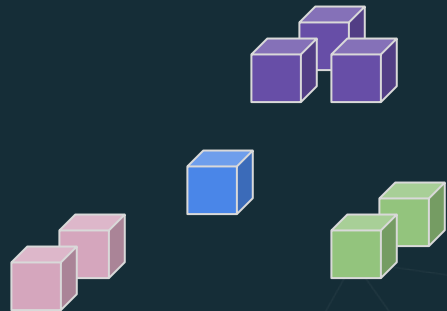
Cloud Storage (S3 etc)

Search













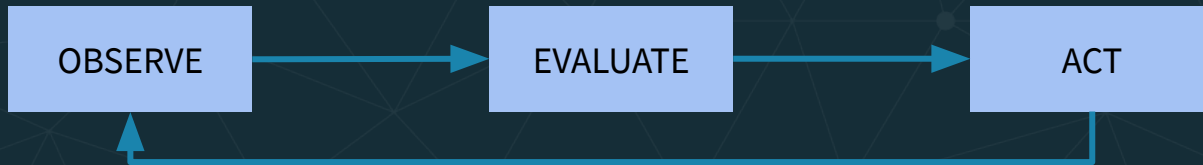




The Operator Pattern

The operator pattern is a way of packaging operational knowledge of an application and make it native to Kubernetes.

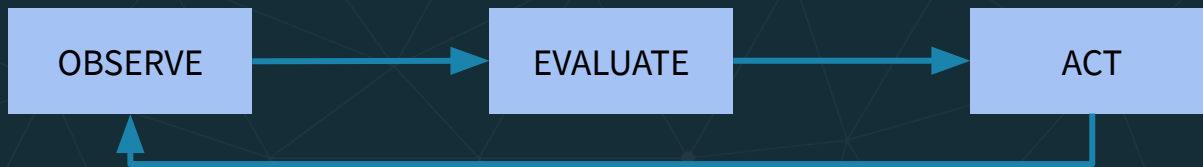
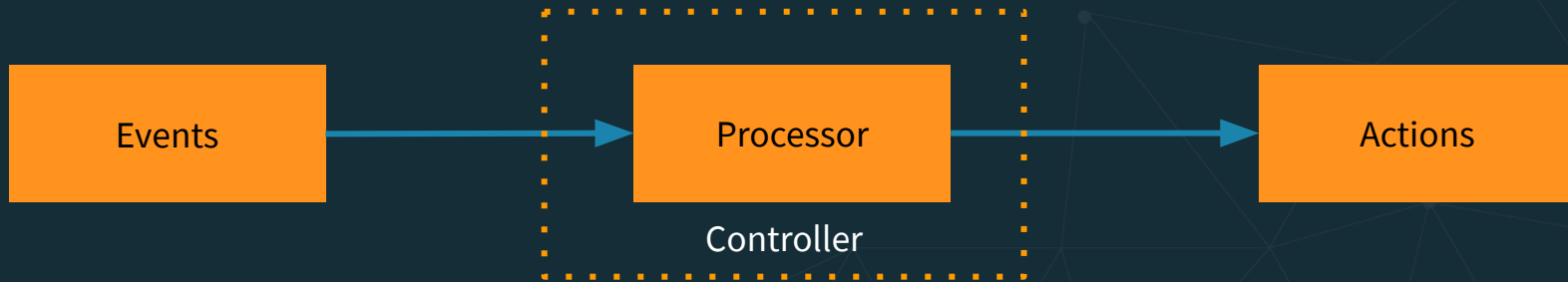
Builds on the concepts of controllers and resources.



What's An Operator?

An operator is an application-specific controller that extends the Kubernetes API to create, configure, and manage instances of complex stateful applications on behalf of a Kubernetes user.

Operator Function



Operator Event Loop

Akka Streams

```
runStream(  
  watch[PipelinesApplication.CR](client)  
    .alsoTo(eventsFlow)  
    .via(AppEvent.fromWatchEvent(logAttributes))  
    .via(TopologyMetrics.flow)  
    .via(AppEvent.toAction)  
    .via(executeActions(actionExecutor, logAttributes))  
    .toMat(Sink.ignore)(Keep.right),  
  "The actions stream completed unexpectedly, terminating.",  
  "The actions stream failed, terminating."  
)
```


Operators in the Wild



Comprehensive list: <https://github.com/operator-framework/awesome-operators>

Operator Definition

- Defines CustomResourceDefinitions (CRDs) to represent a custom resource.
- CRDs make custom features native citizens in Kubernetes.
- Custom Resources (CRs) streamlines the creation and management of the added functionality in a **declarative** way.

\$

```
$ kubectl get crds
```

```
$ kubectl get crds
```

NAME	CREATED AT
flinkapplications.flink.k8s.io	2019-09-20T20:10:00Z
kafkabridges.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkaconnects.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkaconnects2is.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkamirrormakers.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkas.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkatopics.kafka.strimzi.io	2019-09-14T14:42:10Z
kafkausers.kafka.strimzi.io	2019-09-14T14:42:10Z
pipelinesapplications.pipelines.lightbend.com	2019-09-14T14:42:38Z
scheduledsparkapplications.sparkoperator.k8s.io	2019-09-14T14:42:25Z
sparkapplications.sparkoperator.k8s.io	2019-09-14T14:42:24Z

```
$ kubectl get crds
NAME                                     CREATED AT
flinkapplications.flink.k8s.io          2019-09-20T20:10:00Z

kafkabridges.kafka.strimzi.io           2019-09-14T14:42:10Z
kafkaconnects.kafka.strimzi.io          2019-09-14T14:42:10Z
kafkaconnects2is.kafka.strimzi.io       2019-09-14T14:42:10Z
kafkamirrormakers.kafka.strimzi.io      2019-09-14T14:42:10Z
kafkas.kafka.strimzi.io                  2019-09-14T14:42:10Z
kafkatopics.kafka.strimzi.io          2019-09-14T14:42:10Z
kafkausers.kafka.strimzi.io             2019-09-14T14:42:10Z

pipelinesapplications.pipelines.lightbend.com 2019-09-14T14:42:38Z

scheduledsparkapplications.sparkoperator.k8s.io 2019-09-14T14:42:25Z
sparkapplications.sparkoperator.k8s.io         2019-09-14T14:42:24Z
```

```
$ kubectl get crd kafkatopics.kafka.strimzi.io -o YAML
```

```
$ kubectl get crd kafkatopics.kafka.strimzi.io -o YAML
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
```

```
metadata:
  creationTimestamp: "2019-09-14T14:42:10Z"
  generation: 1
  labels:
    app: strimzi
    chart: strimzi-kafka-operator-0.13.0
    component: kafkatopics.kafka.strimzi.io-crd
    heritage: Tiller
    release: pipelines-strimzi
  name: kafkatopics.kafka.strimzi.io
  resourceVersion: "39816972"
  selfLink: /apis/apiextensions.k8s.io/v1beta1/customresourcedefinitions/kafkatopics.kafka.strimzi.io
  uid: d58fb95b-d6fd-11e9-a782-02c9fae95360
spec:
  additionalPrinterColumns:
    - JSONPath: .spec.partitions
      description: The desired number of partitions in the topic
      name: Partitions
      type: integer
    - JSONPath: .spec.replicas
      description: The desired number of replicas of each partition
      name: Replication factor
      type: integer
  group: kafka.strimzi.io
  names:
    kind: KafkaTopic
    listKind: KafkaTopicList
    plural: kafkatopics
    shortNames:
    - kt
    singular: kafkatopic
  scope: Namespaced
  validation:
    openAPIV3Schema:
      properties:
        spec:
          properties:
            config:
              type: object
            partitions:
              minimum: 1
              type: integer
            replicas:
              maximum: 32767
              minimum: 1
              type: integer
            topicName:
              type: string
          required:
          - partitions
          - replicas
          type: object
  version: v1beta1
  versions:
    - name: v1beta1
      served: true
      storage: true
    - name: v1alpha1
      served: true
      storage: false
  status:
```

```
names:
  kind: KafkaTopic
  listKind: KafkaTopicList
  plural: kafkatopics
  shortNames:
  - kt
  singular: kafkatopic
```



```
$ kubectl get kafkatopics
```

```
$ kubectl get kafkatopics
```

NAME	PARTITIONS	REPLICATION FACTOR
call-record-aggregator.cdr-aggregator.out	53	2
call-record-aggregator.cdr-generator1.out	53	2
call-record-aggregator.cdr-generator2.out	53	2
call-record-aggregator.cdr-ingress.out	53	2
call-record-aggregator.cdr-validator.invalid	53	2
call-record-aggregator.cdr-validator.valid	53	2
call-record-aggregator.merge.out	53	2
Consumer-offsets---84e7a678d08f4bd226872e	50	3
mixed-sensors.akka-process.out	53	2
mixed-sensors.akka-process1.out	53	2
mixed-sensors.akka-process2.out	53	2
mixed-sensors.ingress.out	53	2
mixed-sensors.spark-process.out	53	2
mixed-sensors.spark-process1.out	53	2
mixed-sensors.spark-process2.out	53	2

```
$ kubectl get crd kafkatopics.kafka.strimzi.io -o YAML
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
```

```
metadata:
  creationTimestamp: "2019-09-14T14:42:10Z"
  generation: 1
  labels:
    app: strimzi
    chart: strimzi-kafka-operator-0.13.0
    component: kafkatopics.kafka.strimzi.io-crd
    heritage: Tiller
    release: pipelines-strimzi
  name: kafkatopics.kafka.strimzi.io
  resourceVersion: "39816972"
  selfLink: /apis/apiextensions.k8s.io/v1beta1/customresourcedefinitions/kafkatopics.kafka.strimzi.io
  uid: d58fb95b-d6fd-11e9-a782-02c9fae95368
spec:
  additionalPrinterColumns:
  - JSONPath: .spec.partitions
    description: The desired number of partitions in the topic
    name: Partitions
    type: integer
  - JSONPath: .spec.replicas
    description: The desired number of replicas of each partition
    name: Replication factor
    type: integer
  group: kafka.strimzi.io
  names:
    kind: KafkaTopic
    listKind: KafkaTopicList
    plural: kafkatopics
    shortNames:
    - kt
    singular: kafkatopic
  scope: Namespaced
  validation:
    openAPIV3Schema:
      properties:
        properties:
          spec:
            properties:
              config:
                type: object
              partitions:
                type: integer
                minimum: 1
                maximum: 32767
              replicas:
                type: integer
                minimum: 1
              topicName:
                type: string
            required:
            - partitions
            - replicas
            types: object
  version: v1beta1
  versions:
  - name: v1beta1
    served: true
    storage: true
  - name: v1alpha1
    served: true
    storage: false
  status:
```

spec:

- additionalPrinterColumns:
 - JSONPath: .spec.partitions
description: The desired number of partitions in the topic
name: Partitions
type: integer
 - JSONPath: .spec.replicas
description: The desired number of replicas of each partition
name: Replication factor
type: integer

```
$ cat users-topic.yaml
```

```
$ cat users-topic.yaml
apiVersion: kafka.strimzi.io/v1alpha1
kind: KafkaTopic
metadata:
  name: "spark.users"
  namespace: "lightbend"
  labels:
    strimzi.io/cluster: "pipelines-strimzi"
spec:
  topicName: "spark.users"
  partitions: 3
  replicas: 2
  config:
    retention.ms: 7200000
    segment.bytes: 1073741824
```

```
$ kubectl apply -f users-topic.yaml
```



```
$ kubectl apply -f users-topic.yaml
kafkatopic.kafka.strimzi.io/spark.users created
```

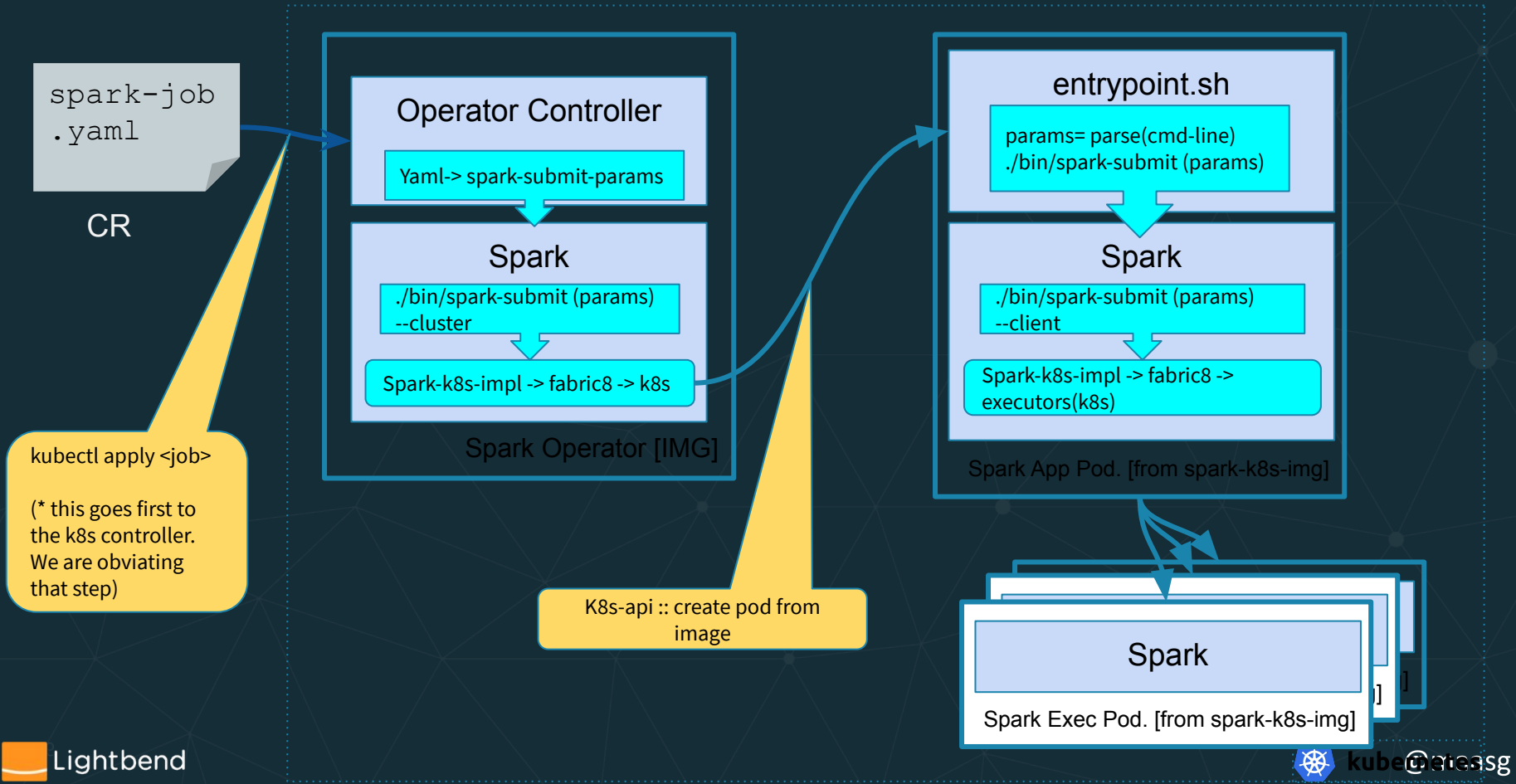


```
$ kubectl get kafkatopics
```

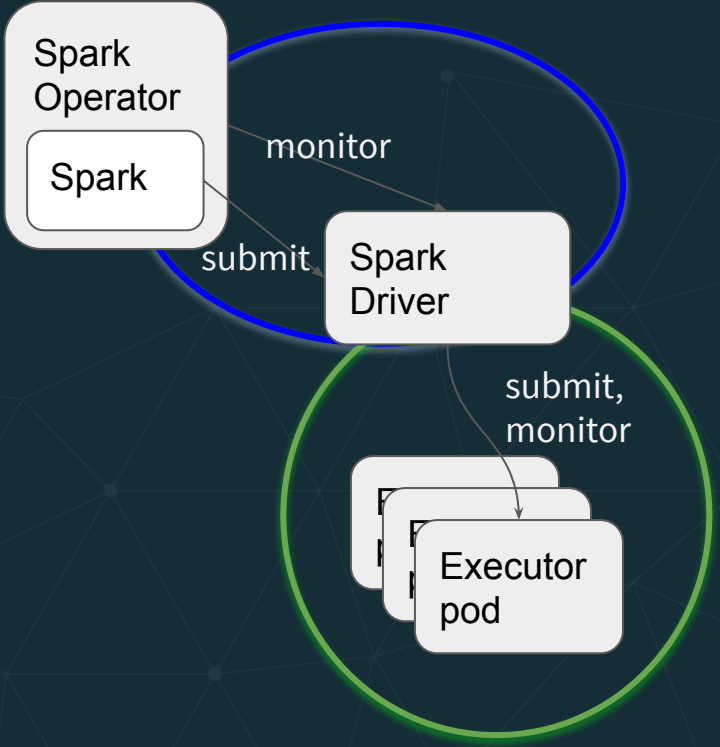
NAME	PARTITIONS	REPLICATION FACTOR
call-record-aggregator.cdr-aggregator.out	53	2
call-record-aggregator.cdr-generator1.out	53	2
call-record-aggregator.cdr-generator2.out	53	2
call-record-aggregator.cdr-ingress.out	53	2
call-record-aggregator.cdr-validator.invalid	53	2
call-record-aggregator.cdr-validator.valid	53	2
call-record-aggregator.merge.out	53	2
Consumer-offsets---84e7a678d08f4bd226872e	50	3
mixed-sensors.akka-process.out	53	2
mixed-sensors.akka-process1.out	53	2
mixed-sensors.akka-process2.out	53	2
mixed-sensors.ingress.out	53	2
mixed-sensors.spark-process.out	53	2
mixed-sensors.spark-process1.out	53	2
mixed-sensors.spark-process2.out	53	2
spark.users	3	2

Operator Federation

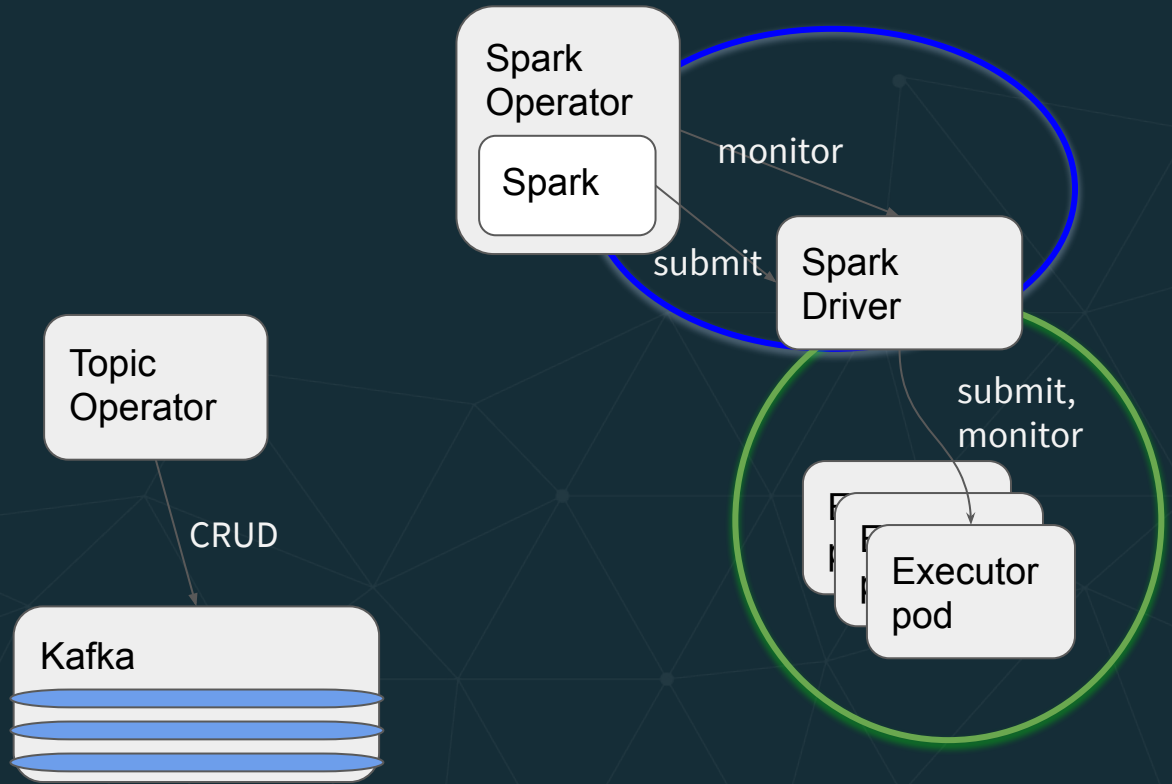
Example: Spark Operator



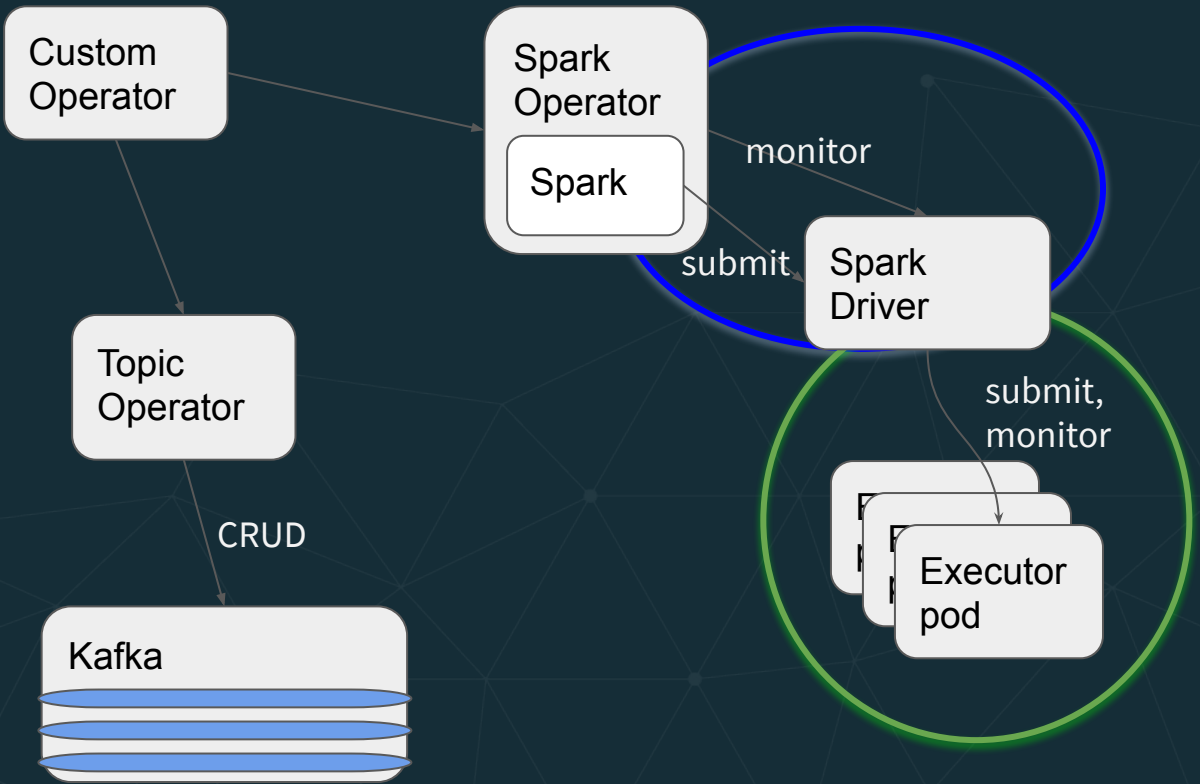
Operator Federation: Achieving Higher Levels of Abstraction



Operator Federation: Achieving Higher Levels of Abstraction

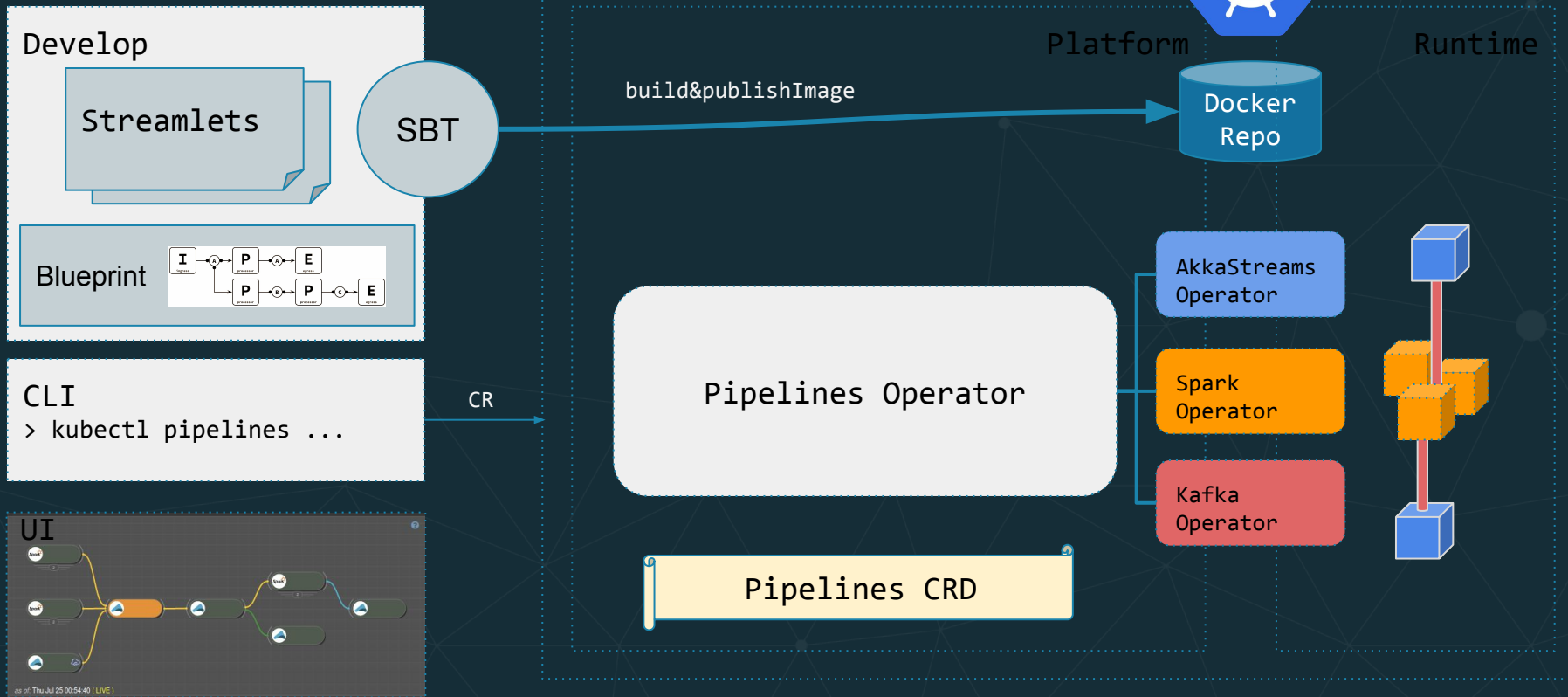


Operator Federation: Achieving Higher Levels of Abstraction

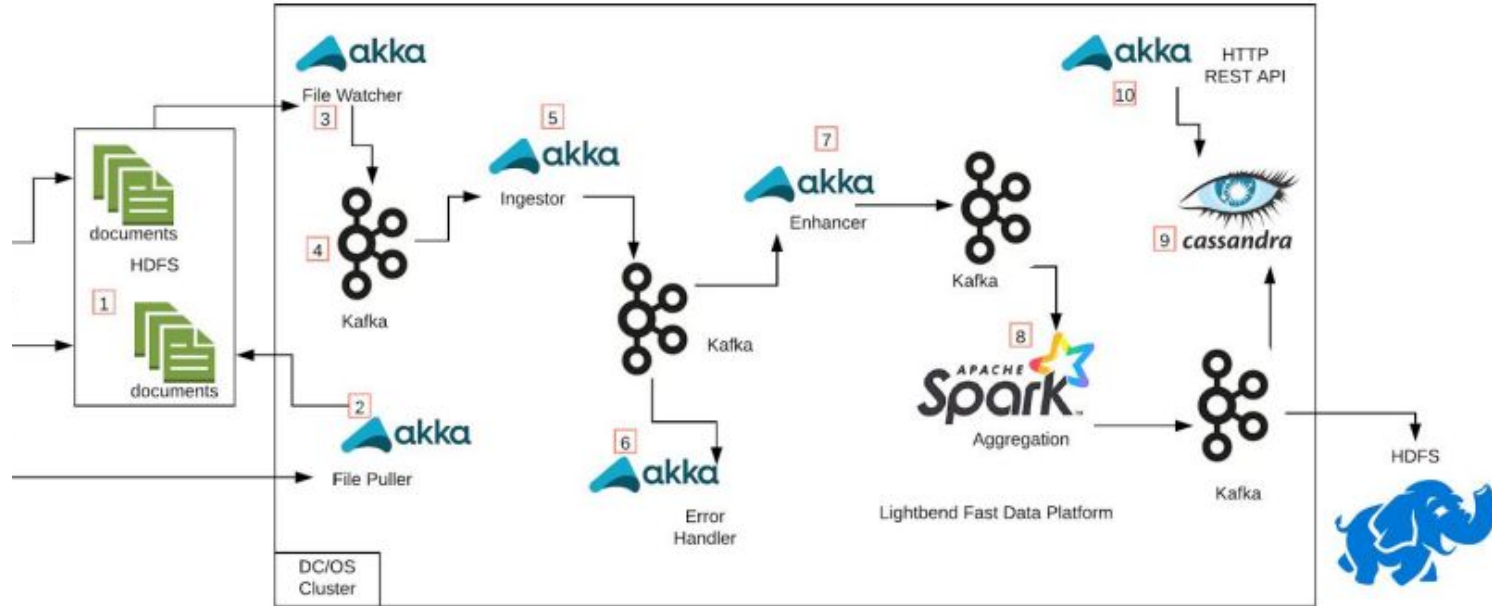


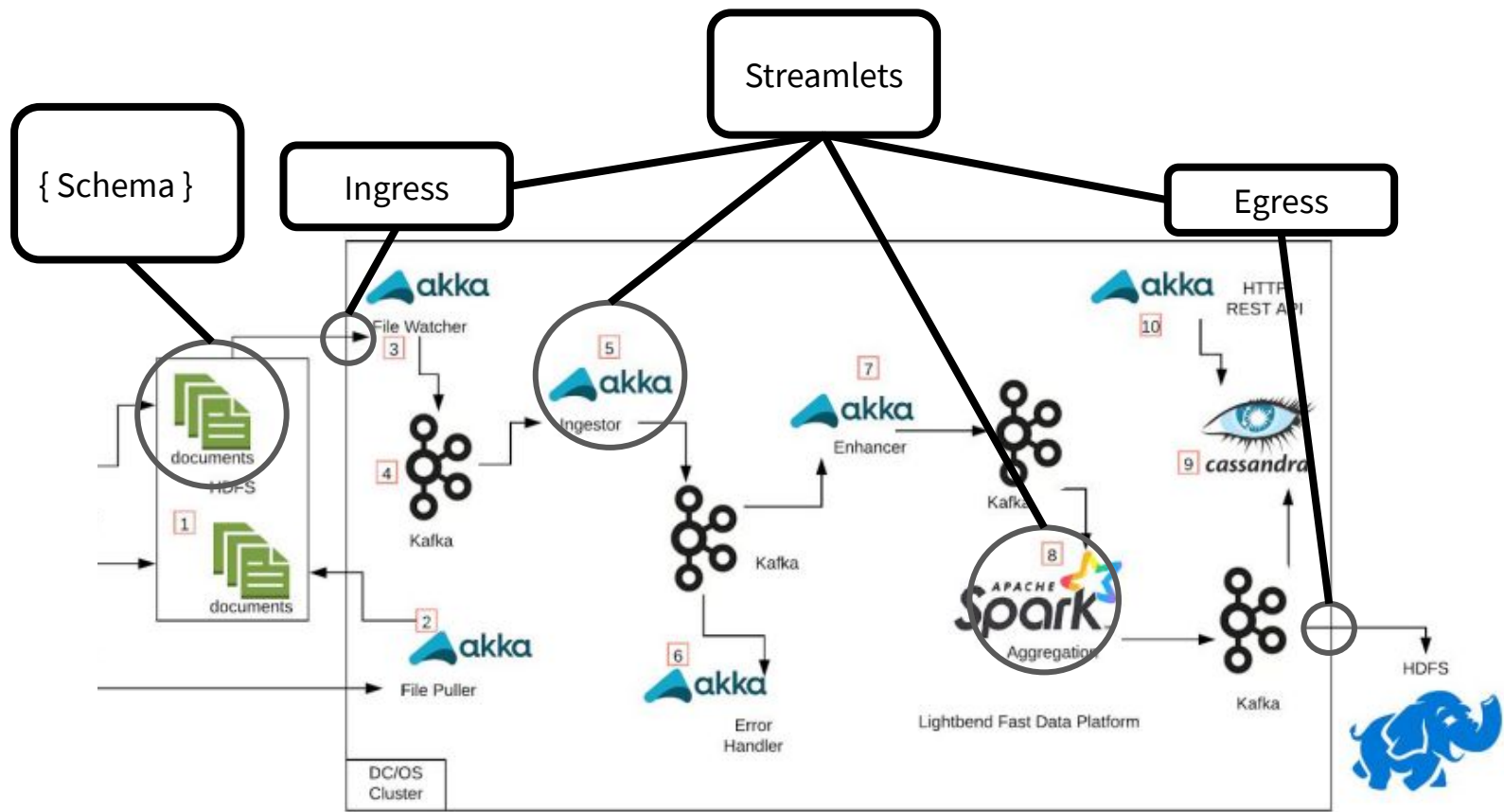
How Are We Using This Approach?

Pipelines Components



Example: Call Record Data Aggregation





```
call-record-aggregator$ tree -L1
```

```
call-record-aggregator$ tree -L1
```

```
.  
├── akka-cdr-ingestor  
├── akka-java-aggregation-output  
├── build.sbt  
├── call-record-pipeline  
├── datamodel  
└── spark-aggregation
```

```
blueprint.conf
```

```
...
```

```
connections {  
  cdr-generator1.out = [merge.in-0]  
  cdr-generator2.out = [merge.in-1]  
  cdr-ingress.out = [merge.in-2]  
  merge.out = [cdr-validator.in]  
  cdr-validator.valid = [cdr-aggregator.in]  
  cdr-aggregator.out = [console-egress.in]  
  cdr-validator.invalid = [error-egress.in]  
}
```

```
call-record-aggregator$ sbt buildAndPublish
```

```
call-record-aggregator$ sbt buildAndPublish
[info] Loading settings for project global-plugins from plugins.sbt ...
[info] Loading project definition from
/home/light/pipelines/pipelines-examples/call-record-aggregator/project
[info] Loading settings for project call-record-aggregator from
build.sbt,target-env.sbt ...
[info] Set current project to call-record-aggregator
[info] Updating datamodel...
...
[info] Sending build context to Docker daemon 180.7MB
[info] Step 1/12 : FROM
lightbend/pipelines-base:1.1.0-spark-2.4.3-flink-1.9.0-scala-2.12
...
[info] You can deploy the application to a Kubernetes cluster using the
following command:
[info] kubectl pipelines deploy
docker-registry-default.purplehat.lightbend.com/lightbend/call-record-aggregato
r:446-c5d6fb3
```

```
call-record-aggregator$ kubectl pipelines deploy  
docker-registry-default.purplehat.lightbend.com/lightbend/call-record-aggregator:446-c5d6fb3
```

```
call-record-aggregator$ kubectl pipelines deploy  
docker-registry-default.purplehat.lightbend.com/lightbend/call-record-aggregator:446-c5d6fb3
```

```
Default value '50' will be used for configuration parameter  
'cdr-generator2.records-per-second'
```

```
Default value '1 minute' will be used for configuration parameter  
'cdr-aggregator.group-by-window'
```

```
Default value '1 minute' will be used for configuration parameter  
'cdr-aggregator.watermark'
```

```
Default value '50' will be used for configuration parameter  
'cdr-generator1.records-per-second'
```

```
[Done] Deployment of application `call-record-aggregator` has started.
```

- Overview
- Applications >
- Builds >
- Resources >
- Storage
- Monitoring
- Catalog

Pods [Learn More](#)

Filter by label Add

Name	Status	Containers Ready	Container Restarts	Age
call-record-aggregator-cdr-ingress-79499485b9-ijvg2	🔄 Running	0/1	0	a few seconds
call-record-aggregator-merge-68d74b945c-q552p	🔄 Running	0/1	0	a few seconds
call-record-aggregator-error-egress-7bb8755b95-7pj45	🔄 Running	0/1	0	a few seconds
call-record-aggregator-console-egress-84857cb8fc-9v4qd	🔄 Running	0/1	0	a few seconds
call-record-aggregator-cdr-validator-6bddc76bbf-r7dkh	🔄 Running	0/1	0	a few seconds

- Overview
- Applications**
- Builds
- Resources
- Storage
- Monitoring
- Catalog

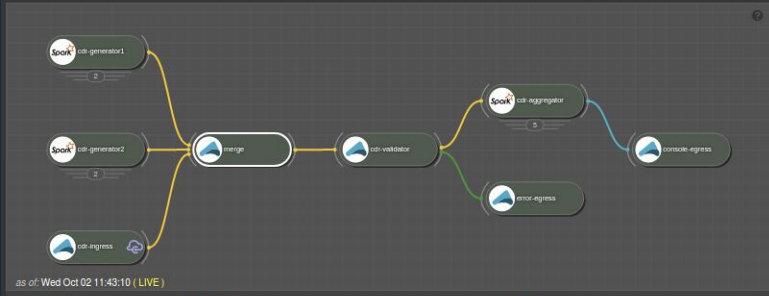
Name	Status	Containers Ready	Container Restarts	Age
call-record-aggregator-merge-68d74b945c-9bpw9	Running	1/1	0	a minute
call-record-aggregator-merge-68d74b945c-ql5b7	Running	1/1	0	a minute
call-record-aggregator-merge-68d74b945c-wrk4c	Running	1/1	0	a minute
call-record-aggregator-merge-68d74b945c-xs5x7	Running	1/1	0	a minute
call-record-aggregator-cdr-generator1-1569968510891-exec-1	Running	1/1	0	3 minutes
call-record-aggregator-cdr-generator1-1569968510891-exec-2	Running	1/1	0	3 minutes
call-record-aggregator-cdr-generator2-1569968510635-exec-2	Running	1/1	0	3 minutes
call-record-aggregator-cdr-generator2-1569968510635-exec-1	Running	1/1	0	3 minutes
call-record-aggregator-cdr-aggregator-1569968510893-exec-1	Running	1/1	0	3 minutes
call-record-aggregator-cdr-aggregator-1569968510893-exec-2	Running	1/1	0	3 minutes
call-record-aggregator-cdr-aggregator-driver	Running	1/1	0	4 minutes
call-record-aggregator-cdr-generator1-driver	Running	1/1	0	4 minutes
call-record-aggregator-cdr-generator2-driver	Running	1/1	0	4 minutes
call-record-aggregator-cdr-ingress-79499485b9-9zxq4	Running	1/1	0	4 minutes
call-record-aggregator-cdr-validator-6bddc76bbf-5s6cw	Running	1/1	0	4 minutes
call-record-aggregator-console-egress-84857cb8fc-x9vfm	Running	1/1	0	4 minutes
call-record-aggregator-error-egress-7bb8755b95-pfjzz	Running	1/1	0	4 minutes
call-record-aggregator-merge-68d74b945c-mbn2j	Running	1/1	0	4 minutes

```
call-record-aggregator$ kubectl pipelines scale cdr-aggregator 5
```

```
[Done] Streamlet cdr-aggregator in application call-record-aggregator is being scaled to 5 replicas.
```

Name	Status	Containers Ready	Container Restarts	Age
call-record-aggregator-cdr-aggregator-driver	⌛ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969844632-exec-2	✖ Terminating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969844632-exec-1	✖ Terminating	0/1	0	a few seconds
call-record-aggregator-cdr-generator1-1569968878788-exec-1	🔄 Running	1/1	0	16 minutes
call-record-aggregator-cdr-generator1-1569968878788-exec-2	🔄 Running	1/1	0	16 minutes
call-record-aggregator-cdr-generator2-1569968878594-exec-1	🔄 Running	1/1	0	16 minutes
call-record-aggregator-cdr-generator2-1569968878594-exec-2	🔄 Running	1/1	0	16 minutes
call-record-aggregator-cdr-generator2-driver	🔄 Running	1/1	0	17 minutes
call-record-aggregator-cdr-generator1-driver	🔄 Running	1/1	0	17 minutes
call-record-aggregator-cdr-ingress-79499485b9-jjvg2	🔄 Running	1/1	0	17 minutes
call-record-aggregator-merge-68d74b945c-qs52p	🔄 Running	1/1	0	17 minutes
call-record-aggregator-error-egress-7bb8755b95-7pj45	🔄 Running	1/1	0	17 minutes
call-record-aggregator-console-egress-84857cb8fc-9v4qd	🔄 Running	1/1	0	17 minutes
call-record-aggregator-cdr-validator-6bddc76bbf-r7dkh	🔄 Running	1/1	0	17 minutes

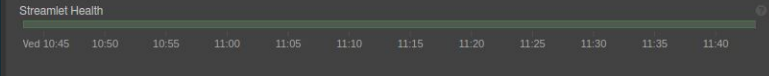
Name	Status	Containers Ready	Container Restarts	Age
call-record-aggregator-cdr-aggregator-1569969941862-exec-1	⌚ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969941862-exec-2	⌚ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969941862-exec-3	⌚ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969941862-exec-4	⌚ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-1569969941862-exec-5	⌚ Container Creating	0/1	0	a few seconds
call-record-aggregator-cdr-aggregator-driver	🔄 Running	1/1	0	a few seconds
call-record-aggregator-cdr-generator1-1569968878788-exec-1	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-generator1-1569968878788-exec-2	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-generator2-1569968878594-exec-1	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-generator2-1569968878594-exec-2	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-generator2-driver	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-generator1-driver	🔄 Running	1/1	0	18 minutes
call-record-aggregator-cdr-ingress-79499485b9-jjvg2	🔄 Running	1/1	0	18 minutes
call-record-aggregator-merge-68d74b945c-qs52p	🔄 Running	1/1	0	18 minutes
call-record-aggregator-error-egress-7bb8755b95-7pj45	🔄 Running	1/1	0	18 minutes
call-record-aggregator-console-egress-84857cb8fc-9v4qd	🔄 Running	1/1	0	18 minutes



as of: Wed Oct 02 11:43:10 (LIVE)

THEME Health health: ■ ■ ■ ■

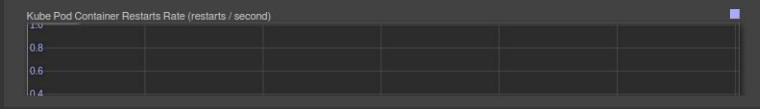
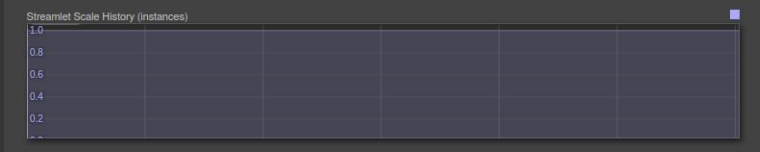
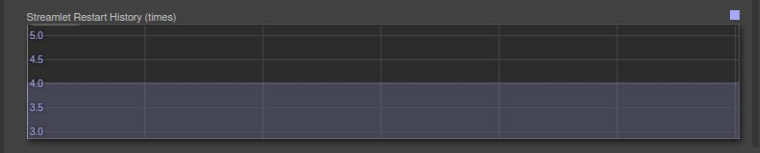
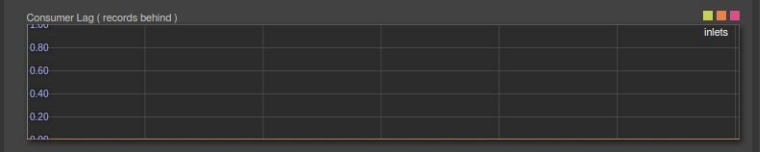
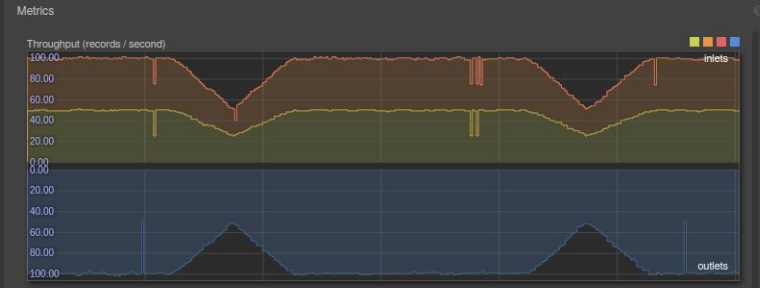
schema: ■ ■ ■



Selection: merge

Health Shape Stream Map

- Streamlet Monitors SORT BY First unhealthy
- kube_container_restarting
 - kube_pod_not_ready
 - pipelines_kafka_consumer_lag
 - pipelines_kafka_consumer_throughput
 - pipelines_kafka_producer_throughput
 - prometheus_target_down
 - scrape_time



Pipelines Design Principles

Blueprints Holistic view of the application

Schema-driven Provide consistency across components

sbt Assembles the pieces and generates meta-data

cli Hook into kubectl for K8S-native interactions

Operator Puts all the operational pieces together

Harnessing the power of existing Operators through a Custom Operator provides a scalable and composable way to transform Kubernetes into a **<your business>** platform.

Learn more

Kafka Operator (Strimzi)

webinar - <https://www.youtube.com/watch?v=rzHQvImn2XY>

demo - <https://www.youtube.com/watch?v=KEPB7iG5Fgc>

Website - <https://strimzi.io/>

Spark Operator

Video - <https://www.youtube.com/watch?v=SKXQwTItQf0>

Github: <https://github.com/GoogleCloudPlatform/spark-on-k8s-operator>

Pipelines

Blog - <https://www.lightbend.com/blog/pipelines>

lightbend.com

O'REILLY®

Compliments of
Lightbend

Designing Fast Data Application Architectures



Gerard Maas, Stavros Kontopoulos
& Sean Glover

\$>Ask (Questions)

Gerard Maas

Principal Engineer



gerard.maas@lightbend.com



[@maasg](https://twitter.com/maasg)



<https://github.com/maasg>



<https://www.linkedin.com/in/gerardmaas/>



<https://stackoverflow.com/users/764040/maasg>

