



# Semantix

## Kafka - Básico

Aula 3

# Quem sou eu?

Eu sou Rodrigo Augusto Rebouças.

Engenheiro de dados da Semantix  
Instrutor do Semantix Mentoring Academy

Você pode me encontrar em:  
[rodrigo.augusto@semantix.com.br](mailto:rodrigo.augusto@semantix.com.br)





# Control Center

Interface





- Interface gráfica para monitoramento e gerenciamento do Confluent
  - Monitoramento
    - Brokers
    - Tópicos
    - Grupos de consumidores
    - Fluxo de dados
  - Gerenciamento
    - Conectores
    - Cluster
    - Tópicos
  - Criação de alertas
  - Desenvolvimento em KSQL



# Interface gráfica

○ <http://localhost:9021/>

 CONFLUENT

 CO  
Cluster 1

## Clusters

1

 Healthy clusters

0

 Unhealthy clusters

**controlcenter.cluster**  
Running

**Overview**

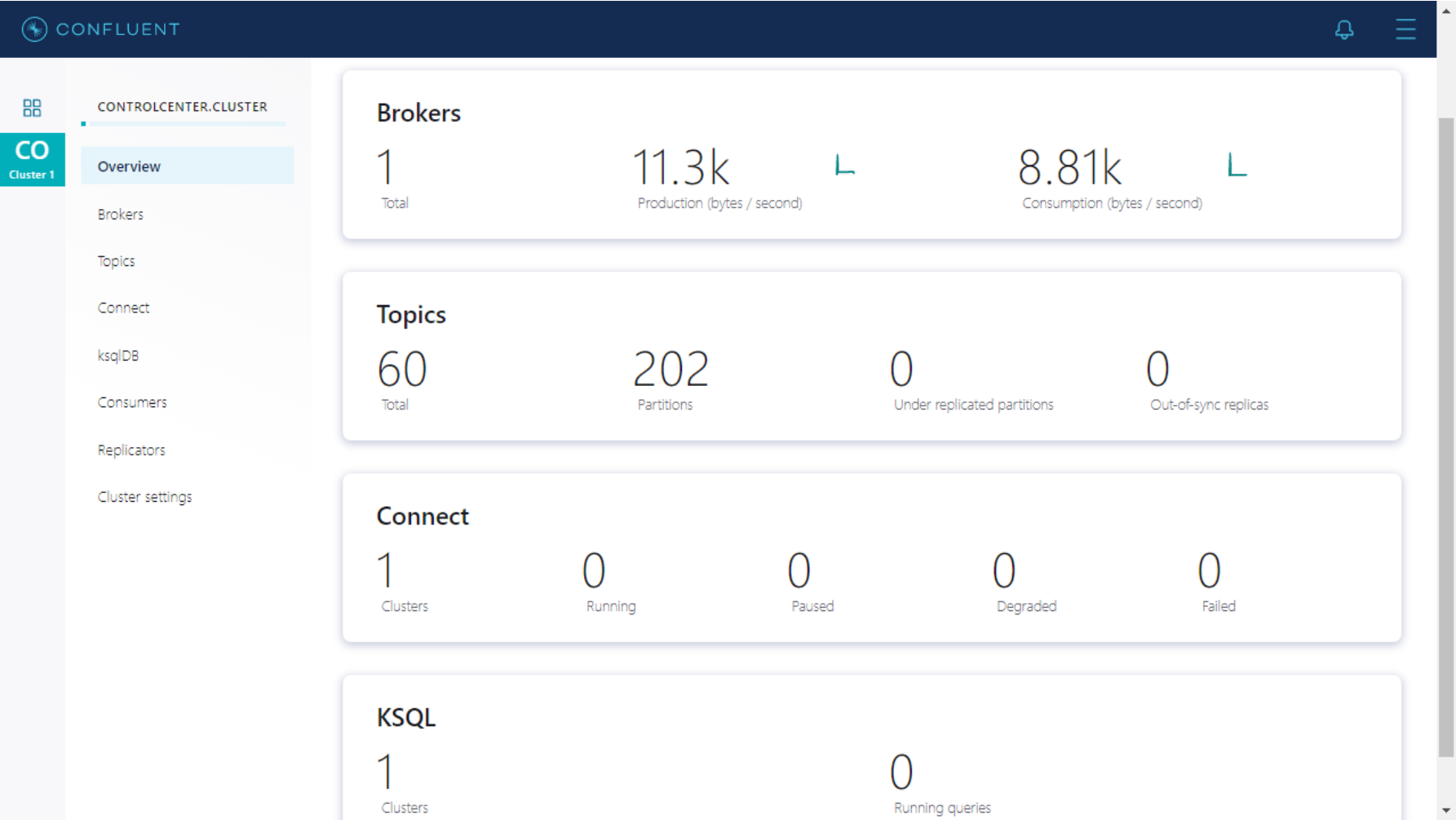
Brokers	1
Partitions	202
Topics	60
Production	10.3kB/s
Consumption	8.56kB/s

**Connected services**

KSQL clusters	1
Connect clusters	1

# Interface gráfica

○ Cluster/Overview





# Control Center

Criação de Tópicos

# Kafka Criação de tópicos

## ○ Confluent Community

- Kafka CLI

```
<path-confluent>/bin/kafka-topics --create --bootstrap-server  
localhost:9092 --replication-factor 1 --partitions 1 --topic users
```

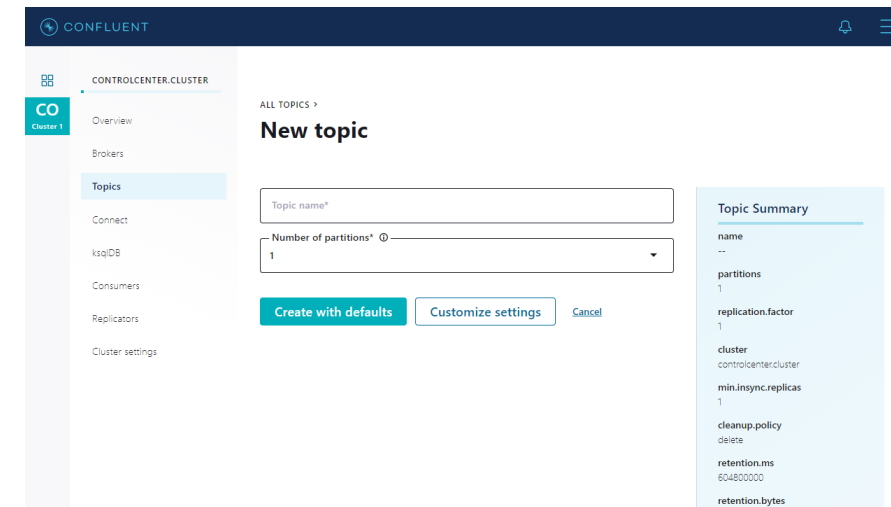
- Instalar Kafka Connect Datagen

```
<path-to-confluent>/bin/confluent-hub install \  
--no-prompt confluentinc/kafka-connect-datagen:latest
```

## ○ Confluent Platform

- Control Center

<http://localhost:9021/>



The screenshot shows the Confluent Control Center interface for creating a new topic. The left sidebar lists navigation options: Overview, Brokers, Topics (selected), Connect, ksqldb, Consumers, Replicators, and Cluster settings. The main panel is titled 'New topic' and contains a 'Topic name\*' input field, a 'Number of partitions\*' dropdown menu set to '1', and three buttons: 'Create with defaults' (highlighted in teal), 'Customize settings', and 'Cancel'. On the right, a 'Topic Summary' panel lists default configuration values: name (empty), partitions (1), replication.factor (1), cluster (controlcenter:cluster), min.insync.replicas (1), cleanup.policy (delete), retention.ms (604800000), and retention.bytes (empty).



# Configurações Tópicos


- Configuração simples
  - Nome
  - Partição
- Outras configurações
  - Availability
  - Cleanup policy
    - Política para limpar
      - Deletar ou Compactar
    - Execução
      - Tempo
      - Tamanho
  - Message size
    - Tamanho máximo da mensagem


# Configurações Tópicos

- Disponibilidade
  - Máxima
    - Fator de Replicação - 3
    - Mínimo de Replicas de sincronização - 1
  - Equilibrada
    - Fator de Replicação - 3
    - Mínimo de Replicas de sincronização – 2
  - Moderada
    - Fator de Replicação - 2
    - Mínimo de Replicas de sincronização – 1
  - Baixa (Não usar em produção)
    - Fator de Replicação - 1
    - Mínimo de Replicas de sincronização - 1

# Control Center - Criação de tópicos

- Cluster / Topics / + Add a topic

 CONFLUENT



CONTROLCENTER.CLUSTER

CO  
Cluster 1

Overview

Brokers

Topics

Connect

ksqlDB

Consumers

Replicators

Cluster settings

## All topics

☒ Hide internal topics

[+ Add a topic](#)

Topics	Availability			Throughput
Topic name	Under replicated partitions	Out of sync followers	Out of sync observers	Bytes/sec produ
<a href="#">default_ksql_processing_log</a>	0 of 1	0 of 1	0 of 0	--
<a href="#">docker-connect-configs</a>	0 of 1	0 of 1	0 of 0	0
<a href="#">docker-connect-offsets</a>	0 of 25	0 of 25	0 of 0	--
<a href="#">docker-connect-status</a>	0 of 5	0 of 5	0 of 0	--
<a href="#">msg-cli</a>	0 of 2	0 of 2	0 of 0	--
<a href="#">t-1</a>	0 of 3	0 of 3	0 of 0	--

localhost:9021/clusters/sch6\_aWYRPOnadY01\_-3oA/management/create-to...

# Control Center - Criação de tópicos

## ○ Tópico padrão

CONFLUENT

CO  
Cluster 1

Overview

Brokers

Topics

Connect

ksqlDB

Consumers

Replicators

Cluster settings

ALL TOPICS >

New topic

Topic name\*

Number of partitions\* ⓘ  
1

Create with defaults

Customize settings

[Cancel](#)

Topic Summary

name

--

partitions

1

replication.factor

1

cluster

controlcenter:cluster

min.insync.replicas

1

cleanup.policy

delete

retention.ms

604800000

retention.bytes

-1

max.message.bytes

1048588

# Control Center - Criação de tópicos

## ○ Tópico com outras configurações

### Availability

- ☒ **Balanced availability/durability** ⓘ  
This option provides a balance between availability and durability. It enforces high replication with strict sync requirements.
- 
- ☐ **Low availability (not recommended for production)** ⓘ  
This option does not provide replication. It is not recommended for production or critical data.
- 
- ☐ **Maximum Availability** ⓘ  
This option prioritizes topic availability for read and writes. It enforces high replication with loose replica sync requirements.
- 
- ☐ **Moderate availability** ⓘ  
This option provides basic redundancy. It enforces replication with minimal sync requirements.
- 
- ☐ **Custom availability settings**  
Manually set replication-factor and min.insync.replicas

replication\_factor

3



min\_insinc\_replicas

2



### Cleanup policy

Cleanup policy\* ⓘ

Delete



Time to retain data ⓘ

1 week



Max size on disk in GB ⓘ

Not set



### Message size

Maximum message size in bytes ⓘ

1048588 bytes

### Expert Settings

☐ Manually configure other topic configuration options

Save & create

Back



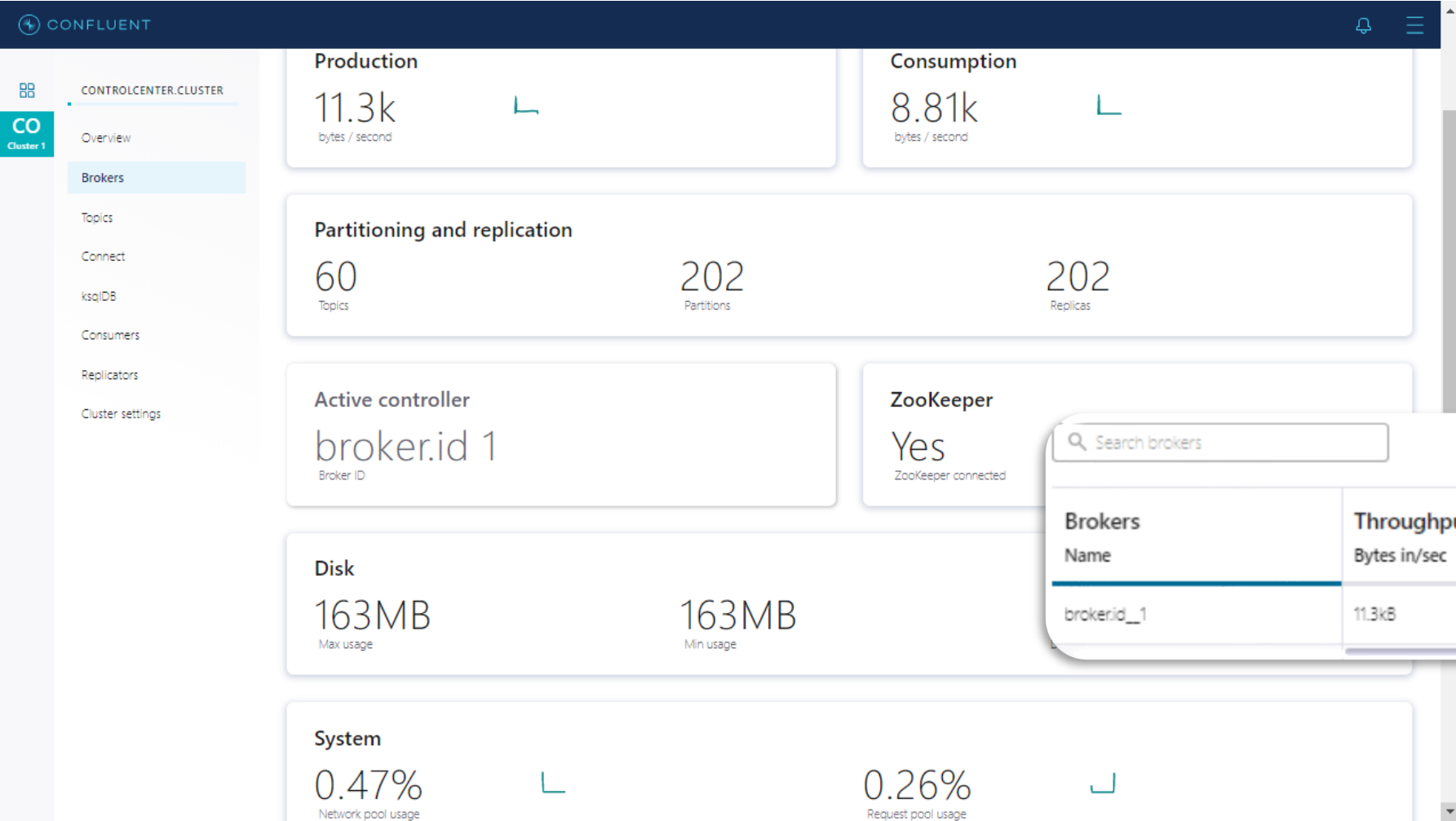


# Control Center

Configurações Cluster

# Configuração Cluster

## ○ Monitoramento Brokers



# Configuração Cluster

## ○ Cluster / Cluster settings / Kafka

CONFLUENT

Cluster 1

Cluster 1

CONTROLCENTER.CLUSTER

Overview

Brokers

Topics

Connect

ksqlDB

Consumers

Replicators

Cluster settings

Cluster settings

Kafka

Brokers

Cluster name\*

controlcenter.cluster

Hosts

broker:29092

Save

# Configuração Cluster

- Cluster / Cluster settings / Brokers

CONFLUENT

Cluster 1

Cluster 1

Overview

Brokers

Topics

Connect

ksqldb

Consumers

Replicators

Cluster settings

# Cluster settings

Kafka

Brokers

Hide settings set to default value

Edit Settings

Download

brokerid 1

## GENERAL

brokerid	1
----------	---

## GROUP MANAGEMENT

group.initial.rebalance.delay.ms	0
----------------------------------	---

## LISTENER

listeners	PLAINTEXT://0.0.0.0:29092,PLAINTEXT_HOST://0.0.0.0:29093
advertised.listeners	PLAINTEXT://broker-29092:29092,PLAINTEXT_HOST://broker-29093:29093

## LOG

log.dirs	/var/lib/kafka/data
----------	---------------------

# Exercícios Control Center

1. Criar um tópico com o nome msg-rapida com 4 partições, 1 replicação e deletar os dados após 5 minutos de uso.
2. Produzir e consumir 2 mensagens para o tópico msg-rapida
3. Qual o nome do cluster?
4. Quantos tópicos existem no cluster?
5. Quantas partições existem o tópico msg-cli?
6. Todas as réplicas estão sincronizadas no tópico msg-cli?
7. Qual a política de limpeza do tópico msg-cli?
8. Alterar a política de limpeza do tópico msg-cli para deletar depois de um ano.
9. Qual o diretório de armazenamento de logs do cluster?
10. Por padrão os dados são mantidos por quantos dias no Kafka?
11. Visualizar os gráficos de produção e consumo de dados do tópico msg-rapida.





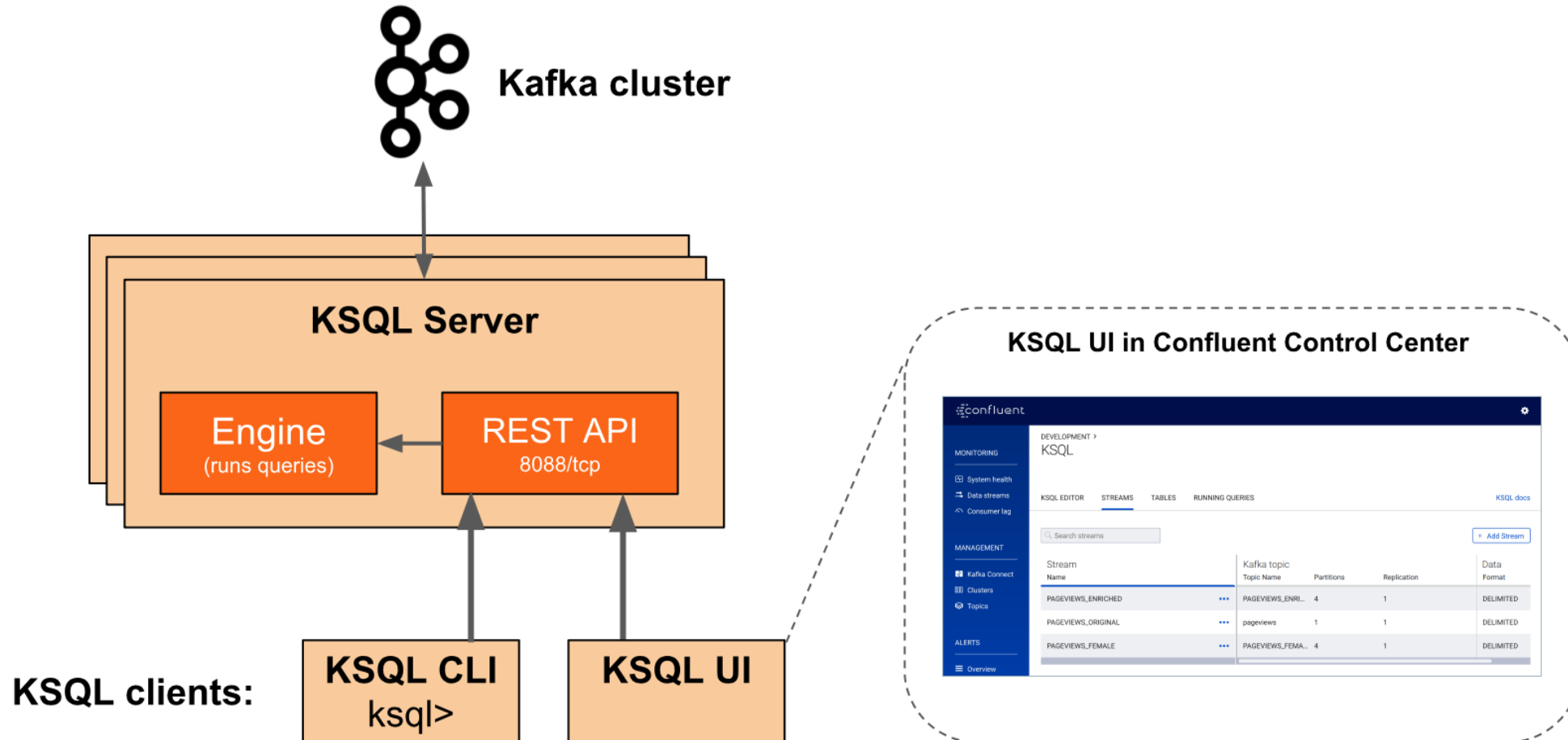
**KSQL**



# KSQL Conceitos

- Engine de Streaming SQL do Kafka
- Interface SQL interativa
  - Facilidade de uso
  - Focada para o processamento streaming no Kafka
  - Sem a necessidade de escrever código em Java ou Python
- Escalável
- Tolerante a falhas
- Tempo real
- Suporta várias operações de streaming
  - Filtragem de dados
  - Transformações
  - Agregações

# KSQL Arquitetura



## ○ Interface gráfica



## ○ Terminal

- `docker exec -it ksqldb-cli bash`
- `ksql`

```
=====
=                                     =
=  KSQL                             =
=                                     =
=  Streaming SQL Engine for Apache Kafka®  =
=====

Copyright 2017-2018 Confluent Inc.

CLI v5.2.2, Server v5.2.2 located at http://localhost:8088

Having trouble? Type 'help' (case-insensitive) for a rundown of how things work!

ksql> █
```

# Comandos Básicos

- Visualizar tópicos

- `ksql> list topics;`

- Mostrar conteúdo do tópico em tempo real

- `ksql> print "<nomeTopico>" <propriedades>;`

- Propriedades

- `from beginning;`

- `interval`

- `limit`

- Ex.:

`ksql> print "topic-produto" from beginning interval 5 limit 10;`





## KSQL Stream

- Sequencia de dados estruturados
- Características
  - Imutáveis
    - Apenas pode inserir dados, não atualizar ou excluir
  - Podem ser criados a partir de um tópico do Kafka ou derivados de um stream existente
    - Fornecer o formato dos valores armazenados no tópico
    - Não inferi o formato de dados do tópico
- Comando para visualizar Streams
  - `Ksql> list streams;`



# KSQL Stream

Criação - Comandos

# Criação Stream

## ○ Criar Stream

- `Ksql> create stream <nomeStream> (<campo> <tipo>, ..., <campo> <tipo>) with (Kafka_topic='<nomeTopic>', value_format='<formato>', KEY='<campoChave>', TIMESTAMP='<campoTimestamp> ...');`
- Formato
  - DELIMITED ( , CSV)
  - JSON
  - AVRO

# KSQL Tipos de dados

- BOOLEAN
- INTEGER ou INT
- BIGINT
- DOUBLE
- VARCHAR ou STRING
- Array
- Map
- Struct



# Exemplos Criação de Stream

- Tópico CSV
  - Informação do tópico cadastro
    - Rodrigo, São José dos Campos
    - Augusto, São Paulo
  - Criar tópico

```
Ksql> create stream cad_str_csv (nome varchar, cidade varchar) with (Kafka_topic='cadastro',  
value_format='delimited');
```

# Exemplos Criação de Stream

## ○ Tópico Json

- Informação do tópico cadastrojson
  - {"nome":"Rodrigo", "cidade":"São José dos campos"}
  - {"nome":"Rodrigo", "cidade":"São Paulo"}
- Criar tópico

```
Ksql> create stream cad_str_json (nome varchar, cidade varchar) with (Kafka_topic='cadastrojson',  
value_format='json');
```

# Exemplos Criação de Stream

- Alterar formato de serialização de CSV/Json para Avro
  - Criar um Stream no formato avro que cria um novo tópico com os dados do stream no formato csv /json
  - Ex. CSV para Avro:

```
Ksql> create stream cad_avro_csv with (kafka_topic='cadastro-avro', value_format='avro') as select * from cad_str;
```

- Ex. Json para Avro:

```
Ksql> create stream cad_str_json with (kafka_topic='cadastro-avro', value_format='avro') as select * from cad_str;
```



# KSQL Stream

Control Center

# Criação Stream

CONTROLCENTER.CLUSTER

Overview

Brokers

Topics

Connect

ksqlDB

Consumers

Replicators

Cluster settings

KSQLDB >

## ksqldb1

Editor

Flow

Streams

Tables

Running queries

Search streams

Stream

Name

KSQL\_PROCESSING\_L

## Create a ksqlDB Stream

Choose a topic that contains the data you want in your STREAM

Step 1 of 1

Search topics

☐ Show internal topics

default\_ksql\_processing\_log



docker-connect-configs



docker-connect-offsets



docker-connect-status



msg-cli



+ Add Stream

# Criação Stream

## Create a ksqlDB Stream

Choose a topic that contains the data you want in your STREAM

Step 2 of 2

ksqlDB Cluster

ksqldb1

Topic

msg-cli

STREAM name

MSG\_CLI

Query type

STREAM

Encoding

Key

Timestamp

Field(s) you'd like to include in your STREAM

Field name

FIELD\_0

Field type

VARCHAR

+ Add another field

Save STREAM

Back

- Type
  - Stream
  - Table
    - View de stream
- Encoding (Formato)
  - Delimited
  - Json
  - Avro
- Key
- Timestamp
- Field - type



An abstract graphic on the left side of the slide. It features two overlapping profiles of human heads facing each other. The profiles are filled with a blue, wavy, line-art pattern. Inside the profiles, there are faint, semi-transparent images of data visualizations, including bar charts and line graphs. The background is white.

## Operações com KSQL Stream



# Comandos Stream

SELECT select\_expr [, ...]

FROM from\_item

[ LEFT JOIN join\_table ON join\_criteria ]

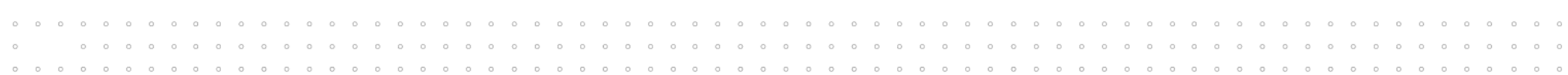
[ WINDOW window\_expression ]

[ WHERE condition ]

[ GROUP BY grouping\_expression ]

[ HAVING having\_expression ]

[ LIMIT count ];



# Comandos Stream - Visualização

- Visualizar conteúdo do Stream
  - Ksql> select <campo>,...<campo> from <nomeStream>
  - Ex.:  
Ksql> select nome from cad\_str;  
Ksql> select \* from cad\_str limit 10;
- Visualizar a estrutura do Stream
  - Ksql> describe <nomeStream>
  - Ksql> describe extended <nomeStream>


# Comandos Stream – Setar propriedades

- Setar propriedades
  - Ksql> set <propriedade> = <valor>
- Desfazer propriedades
  - Ksql> unset <propriedade> = <valor>
- Ex.
  - Setar para visualizar os dados desde o início
    - Ksql> set 'auto.offset.reset'='earliest';
  - Desfazer configuração
    - Ksql> unset 'auto.offset.reset'

Editor Flow Streams Tables Running queries

1 Example: SELECT field1, field2, field3 FROM mystream WHERE field1 = 'somevalue' EMI

[Add query properties](#)

auto.offset.reset = Latest 

[+ Add another field](#)

Run

# Comandos Stream - Inserção

## ○ Inserção

```
insert into <stream_name|table_name> (<column_name>, <...>) values (<value>, '<value>', <...>);
```

```
insert into stream_name
```

```
select select_expr [., ...]
```

```
from from_stream
```

```
[ where condition ]
```

```
[ partition by column_name ];
```

- Ex.:
- Ksql> insert into foo (rowtime, rowkey, key\_col, col\_a) values (1234, 'key', 'key', 'a');

# Comandos Stream - Exclusão

- Deletar uma Stream

- `ksql> drop stream <nomeStream>;`

- Deletar uma Stream e seu tópico

- `ksql> drop stream <nomeStream> delete topic;`
- `ksql> drop stream if exists <nomeStream> delete topic;`



## Agregações com KSQL

# Comandos Stream - Agregações

- agregações
  - count
  - max
  - min
  - sum
  - topk
  - topkdistinct



# Comandos Stream - Agregações

- Contar a quantidade de linhas de um campo Stream

- Ksql> count(\*) from <nomeStream>;
  - Erro, precisa fazer agregação
- Ksql> select <campo> count(\*) from <nomeStream> group by <campo>;
- Ex.:

```
Ksql> select cidade, count(*) from cad_str group by cidade;
```

# Comandos Stream - Agregações

- Contar a quantidade de linhas de todo o tópico
  - O count sempre precisa do group by
    - Criar um campo setado para 1 em todos os registros
  - Comando:

```
ksql> CREATE STREAM <novoStream> AS SELECT 1 AS unit FROM <nomeStreamParaContar>;  
select count(unit) from <novoStream> group by unit;
```

# Exercícios Stream

1. Criar o tópico msg-usuário-csv
2. Criar um produtor para enviar 3 mensagens contendo id e nome separados por virgula para o tópico msg-usuário-csv
3. Visualizar os dados do tópico msg-usuário-csv
4. Criar o Stream usuario\_csv para ler os dados do tópico msg-usuário-csv
5. Visualizar o Stream usuario\_csv
6. Visualizar apenas o nome do Stream usuario\_csv



# Semantix

## Obrigado!

Alguma pergunta?



Você pode me encontrar em:  
[rodrigo.augusto@semantix.com.br](mailto:rodrigo.augusto@semantix.com.br)

**GET SMARTER**