



# Semantix

## Kafka - Básico

Aula 1

Quem sou eu?

Eu sou Rodrigo Augusto Rebouças.

Engenheiro de dados da Semantix  
Instrutor do Semantix Academy

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





# Semantix

## Agenda

Introdução ao Kafka e a Confluent

Instalação

Confluent CLI

Control Center

KSQL

Schema Registry

Kafka Connect

Confluent Cloud



# Introdução

Kafka





- Plataforma de streaming distribuída
  - Open Source
  - Publicar e assinar streams de registros
  - Fluxos de registros
    - Processar
      - Tempo real
    - Armazenar
      - Tolerante a falhas



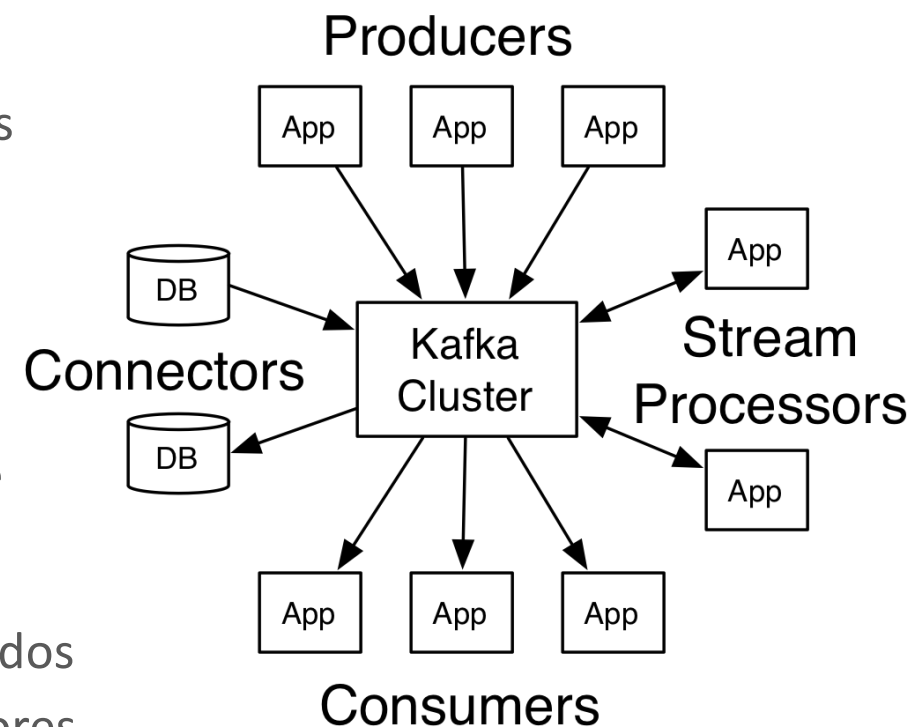
# Kafka História

- 2010
  - Originalmente desenvolvido pelo LinkedIn
    - Necessidade de integração massiva de dados
    - Conceito do Kafka surgido com Jay Kreps e sua equipe
- 2011
  - Liberado como um projeto open-source
- 2012
  - Apache Kafka



# Conceitos Kafka Cluster

- Kafka é desenvolvido em Scala e Java
- O Kafka é executado como um cluster em um ou mais servidores que podem abranger vários datacenters
- O cluster Kafka armazena fluxos de *registros* em categorias denominadas **tópicos**
- Cada registro consiste em uma chave, um valor e um registro de data e hora
- Apache Kafka é um sistema para gerenciamento de fluxos de dados em tempo real, gerados a partir de web sites, aplicações e sensores



# Principais API Kafka

## ○ Producer API

- Permite que um aplicativo publique um fluxo de registros em um ou mais tópicos do Kafka

## ○ Consumer API

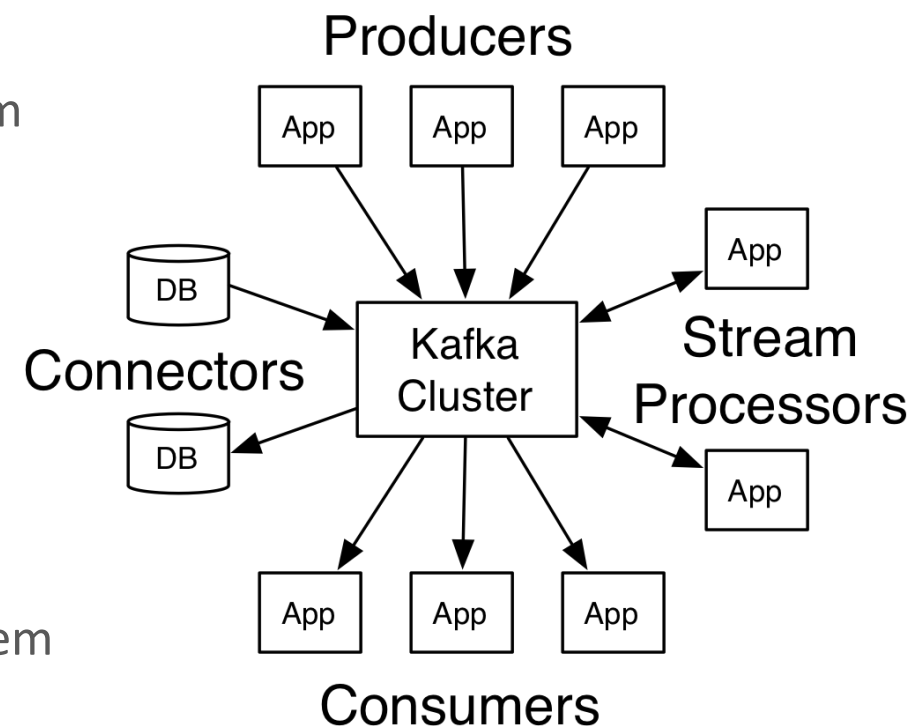
- Permite que um aplicativo assine um ou mais tópicos e processe o fluxo de registros produzidos para eles

## ○ Streams API

- Permite que um aplicativo transforme os fluxos de entrada em fluxos de saída

## ○ Connector API

- Permite criar e executar produtores ou consumidores reutilizáveis que conectam tópicos do Kafka a aplicativos ou sistemas de dados existentes







# Introdução

Confluent



# Confluent

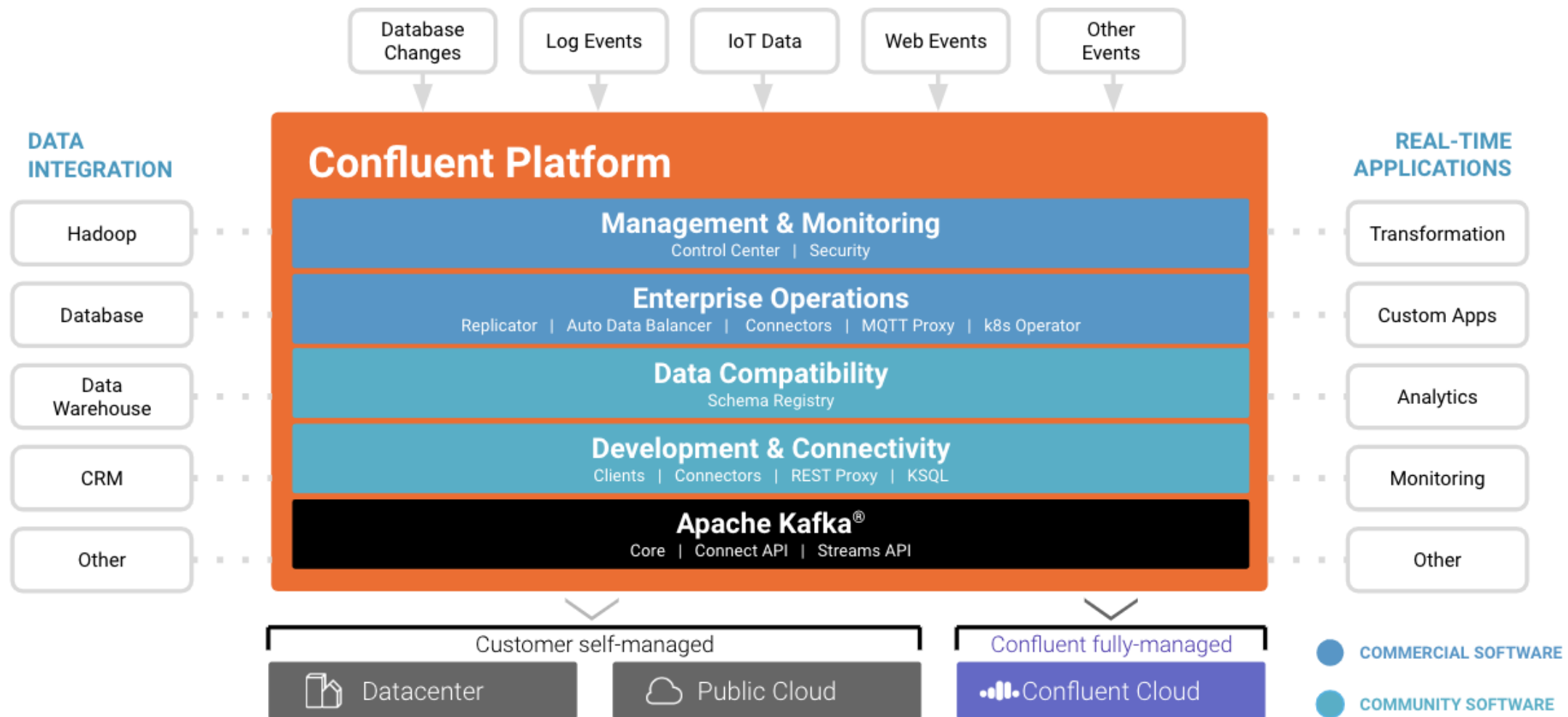
- Fundada em 2014 pelos criadores originais do Apache Kafka
  - Jay Kreps
- Plataforma de streaming
  - Uso corporativo
  - Infraestrutura de aplicativos e dados
- Fornece uma plataforma única
  - Eventos históricos
  - Tempo real



# Confluent

- Facilitar
  - Construção de pipelines de dados
  - Aplicativos de streaming em tempo real
  - Integração de dados de várias fontes e locais
- Simplificar
  - Conexão de fontes de dados ao Kafka
  - Criação de aplicativos com o Kafka
  - Infraestrutura Kafka
    - Proteção
    - Monitoramento
    - Gerenciamento

# Confluent - Plataforma





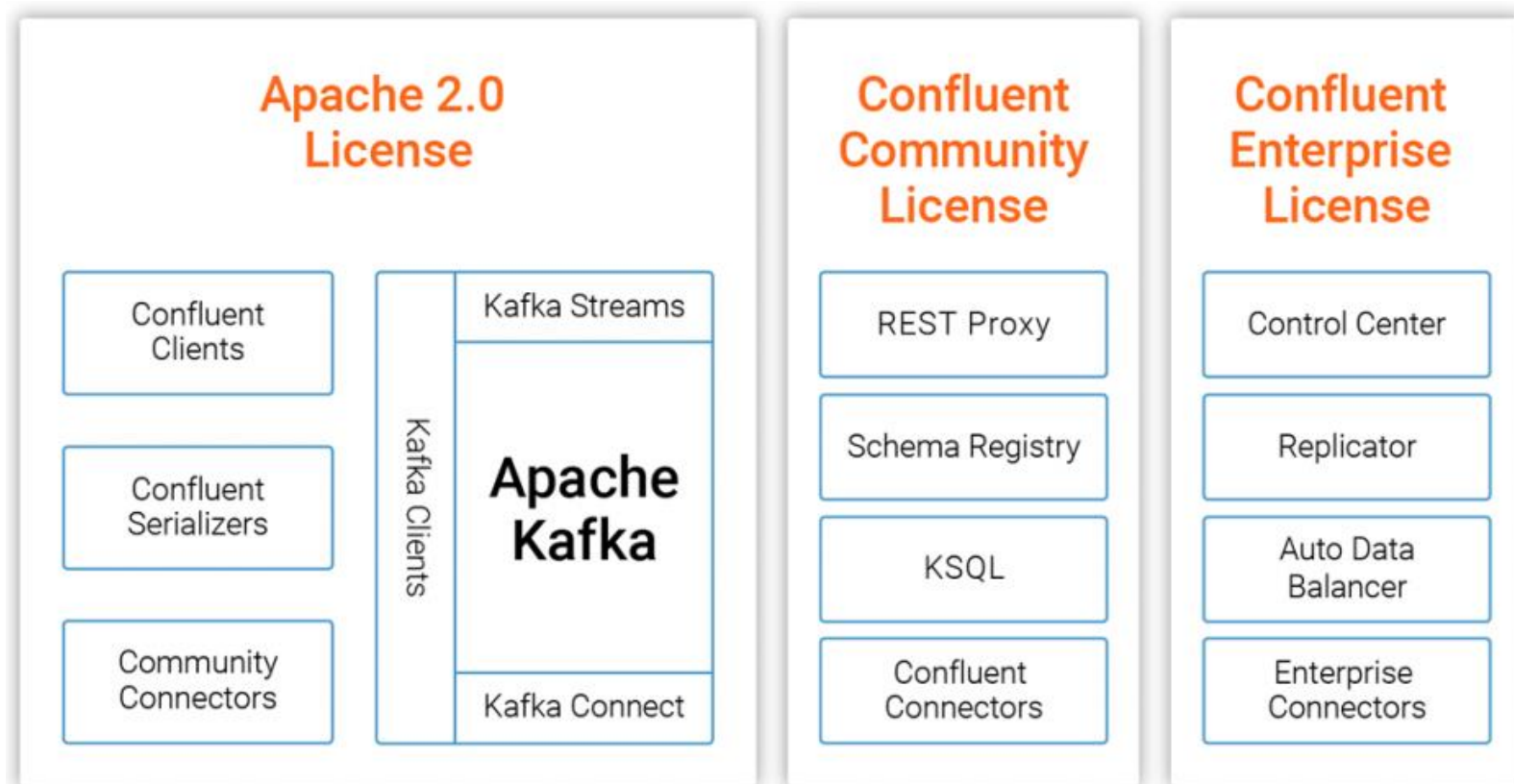
# Instalação

Licença e Versões



# Licença Confluent

- Licença Confluent Enterprise
  - Gratis
    - Single kafka broker
  - 30 dias gratuitos
    - Unlimited Kafka broker



- Versões Confluent
  - <https://docs.confluent.io/current/installation/versions-interoperability.html>



# Instalação Confluent

- Confluent Community
  - Docker
    - <https://docs.confluent.io/current/quickstart/cos-docker-quickstart.html>
  - Local
    - <https://docs.confluent.io/current/quickstart/cos-quickstart.html>
- Confluent Enterprise
  - Docker
    - <https://docs.confluent.io/current/quickstart/ce-docker-quickstart.html>
  - Local
    - <https://docs.confluent.io/current/quickstart/ce-quickstart.html>
- Cloud
  - <https://www.confluent.io/confluent-cloud>

# Versões Confluent

Confluent Platform	Release Date	Sem Suporte	Kafka	Scala	ZooKeeper	Java
1.0.0	02/2015	02/2017		2.10, 2.11		1.7.0_60, 1.8.0_60
2.0.x	12/2015	12/2017	0.9.0.x	2.10, 2.11		1.7.0_60, 1.8.0_60
3.0.x	05/2016	05/2018	0.10.0.x	2.10, 2.11	3.4.6	1.7.0_60, 1.8.0_60
3.1.x	11/2016	11/2018	0.10.1.x	2.10, 2.11	3.4.8	1.7.0_60, 1.8.0_60
3.2.x	03/2017	03/2019	0.10.2.x	2.10, 2.11	3.4.9	1.7.0_60, 1.8.0_60
3.3.x	08/2017	08/2019	0.11.0.x	2.11	3.4.10	1.7.0_60, 1.8.0_60
4.0.x	11/2017	11/2019	1.0.x	2.11	3.4.10	1.7.0_60, 1.8.0_60
4.1.x	04/2018	04/2020	1.1.x	2.11	3.4.10	1.7.0_60, 1.8.0_60
5.0.x	07/2018	07/2020	2.0.x	2.11	3.4.13	1.8.0_60
5.1.x	12/2018	12/2020	2.1.x	2.11	3.4.13	1.8.0_60
5.2.x	03/2019	03/2021	2.2.x	2.11, 2.12	3.4.13	1.8.0_60, 11.0_2
5.3.x	07/2019	07/2021	2.3.x	2.11, 2.12	3.4.14	1.8.0_60, 11.0_2
5.4.x	01/2020	01/2022	2.4.x	2.11, 2.12	3.5.8	1.8.0_202, 11.0_4
<b>5.5.x</b>	<b>02/2020</b>	<b>04/2022</b>	<b>2.5.x</b>	<b>2.11, 2.12</b>	<b>3.5.8</b>	<b>1.8.0_202, 11.0_4</b>
6.0.x	09/2020	09/2022	2.6.x	2.13	3.5.8	1.8.0_202, 11.0_4



## Instalação - Docker

# Referencia Imagens docker

- Docker Hub

- <https://hub.docker.com/u/confluentinc/>

- Referencia de imagens docker

- Link

- <https://docs.confluent.io/current/installation/docker/image-reference.html>

- Componentes

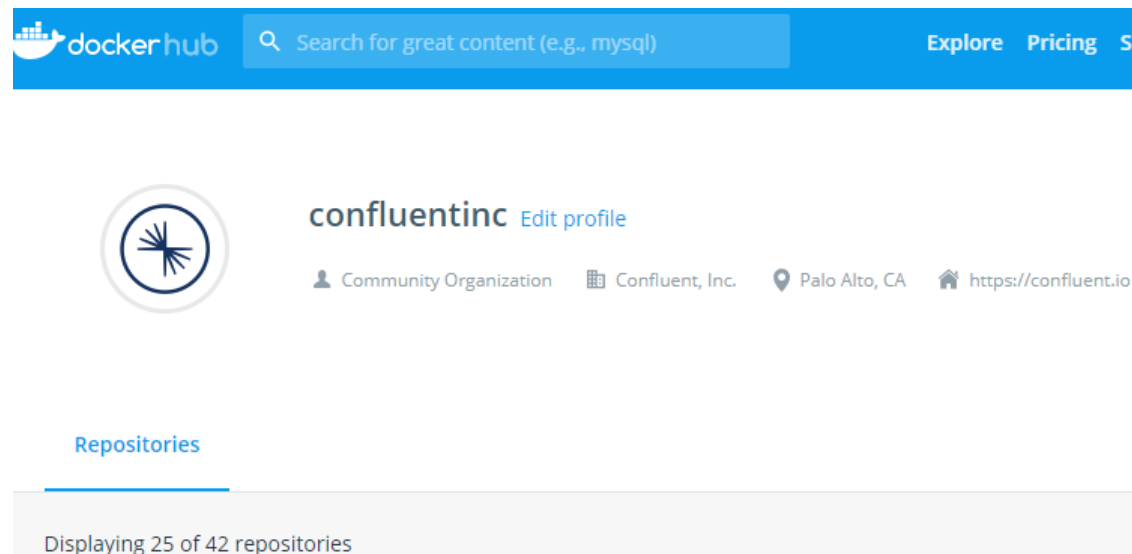
- Nome das imagens

- Licença

- Pacotes incluídos

- Repositório de GitHub

- Total do cluster ~ 8GB



# Preparação Ambiente

## ○ Instalação

- Docker: <https://docs.docker.com/get-docker/>
- Docker-compose: <https://docs.docker.com/compose/install/>
- SO
  - Windows
    - Docker Desktop (Hyper-V ou Hyper-V com WSL2)
    - Docker Toolbox (VirtualBox) (Instalacao\_Docker\_Windos\_10\_Home.pdf)
  - Linux - Seguir o passo a passo (PassosInstalacaoDockerLinux.txt)
  - Mac - Docker Desktop

## ○ Criar a seguinte estrutura de diretório:

kafka

docker-compose.yml

# Opções Docker Compose

- Iniciar todos os serviços
  - \$ docker-compose up -d
- Parar os serviços
  - \$ docker-compose stop
- Iniciar os serviços
  - \$ docker-compose start
- Término do treinamento
  - Matar os serviços
    - \$ docker-compose down
  - Apagar todos os volumes sem uso
    - \$ docker volume prune

- cat docker-compose.yml

```
services:
  zookeeper:
    image: confluentinc/cp-zookeeper:5.5.2
    hostname: zookeeper
    container_name: zookeeper
    ports:
      - "2181:2181"
    environment:
      ZOOKEEPER_CLIENT_PORT: 2181

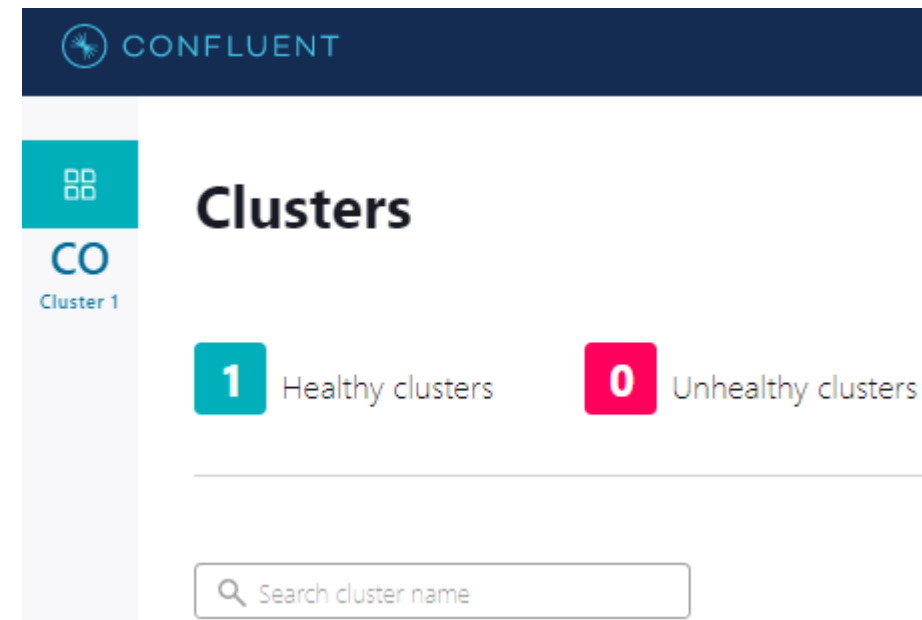
  broker:
    image: confluentinc/cp-server:5.5.2
    hostname: broker
    container_name: broker
    depends_on:
      - zookeeper
    ports:
      - "9092:9092"
      - "9101:9101"
    environment:
```



# Acessos Ambiente docker

- Visualizar os container
  - Ativos
    - \$ docker ps
  - Todos
    - \$ docker ps -a
- Executar comandos no container
  - \$ docker exec -it <container> <comando>
- Visualizar os logs
  - \$ docker logs <container>
- Enviar arquivos
  - \$ docker cp <diretório> <container>:/<diretório>

- Acessos de ferramentas
  - Server de Kafka
    - \$ docker exec -it broker bash
  - Confluent Control Center
    - <http://localhost:9021>



# Confluent Portas

## ○ Portas

Componentes	Portas Default
Apache Kafka brokers	9092
Confluent Control Center	9021
Kafka Connect REST API	8083
KSQL Server REST API	8088
REST Proxy	8082
Schema Registry REST API	8081
ZooKeeper	2181

# Exercícios Instalação

1. Criar a pasta kafka e inserir o arquivo docker-compose.yml da Guia Arquivos do treinamento
2. Instalação do docker e docker-compose
3. Inicializar o cluster Kafka através do docker-compose
4. Listas as imagens em execução
5. Visualizar o log dos serviços broker e zookeeper
6. Visualizar a interface do Confluent Control Center
  - Acesso: <http://localhost:9021/>



## Confluent CLI

# Confluent CLI

- Confluent **C**ommand **L**ine **I**nterface (CLI)
- Instalar, administrar a plataforma Confluent
- Aplicação para uso de desenvolvimento
  - Não adequado para ambiente de produção
- Execução
  - `<path-confluent>/bin/confluent <comandos>`
  - Exemplo
    - `<path-confluent>/bin/confluent list`

# Confluent CLI - Comandos

## ○ confluent <comando>

- consume <tópico>
- produce <tópico>
- config <conector>
- load <conector>
- unload <conector>
- Version <serviço>
- help <comando>
- list
- start <serviço>
- status <serviço>
- stop <serviço>
- top <serviço>
- log <serviço>
- current



# Exercícios Confluent CLI

1. Verificar versão do Confluent
2. Usar o comando de ajuda para ver as propriedades de status
3. Verificar o status do Zookeeper e do Kafka
4. Iniciar o Kafka
5. Verificar o status do do Zookeeper e do Kafka
6. Visualizar o log do Zookeeper e do Kafka
7. Verificar o consumo de CPU e RAM do Zookeeper e do Kafka
8. Verificar o status de todos os serviços
9. Visualisar todos os conectores



# Semantix

## Obrigado!

Alguma pergunta?



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

**GET SMARTER**