

Grupo

Apresentaremos o trabalho
sobre o framework e distributed
processing engine open source
Apache Flink



Gustavo Lucas



Sammuel Ramos

Conteúdo



1 O que é o Apache Flink

2 Aplicações do Flink

3 Empresas que utilizam

4 Diferenciais

5 Apache Flink na Cloud

6 Aplicação em execução

Apache Flink



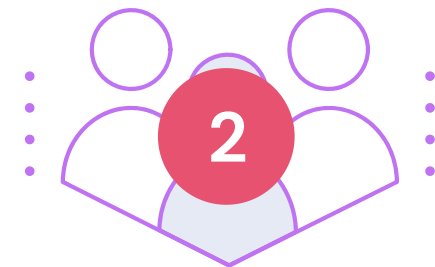
O que é?

- Plataforma open source para fluxo escalável e processamento em lote
- "Flink": palavra de origem alemã que significa rápido e veloz.
- O núcleo do Flink é um mecanismo de fluxo de dados de streaming distribuído
- Executa fluxos de dados em paralelo em clusters
- Fornece um back-end confiável para várias cargas de trabalho

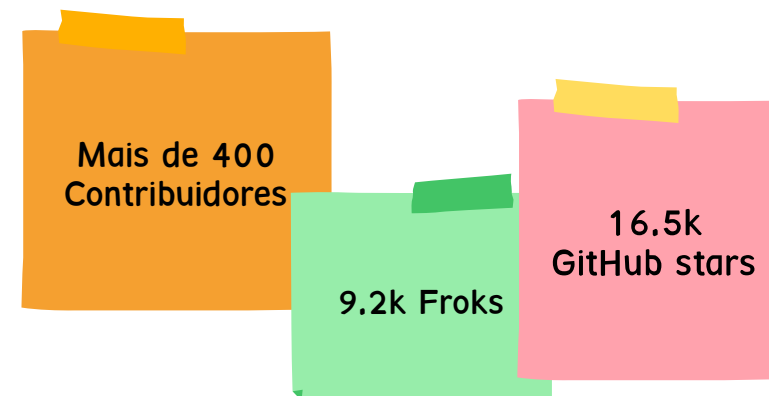
Sobre



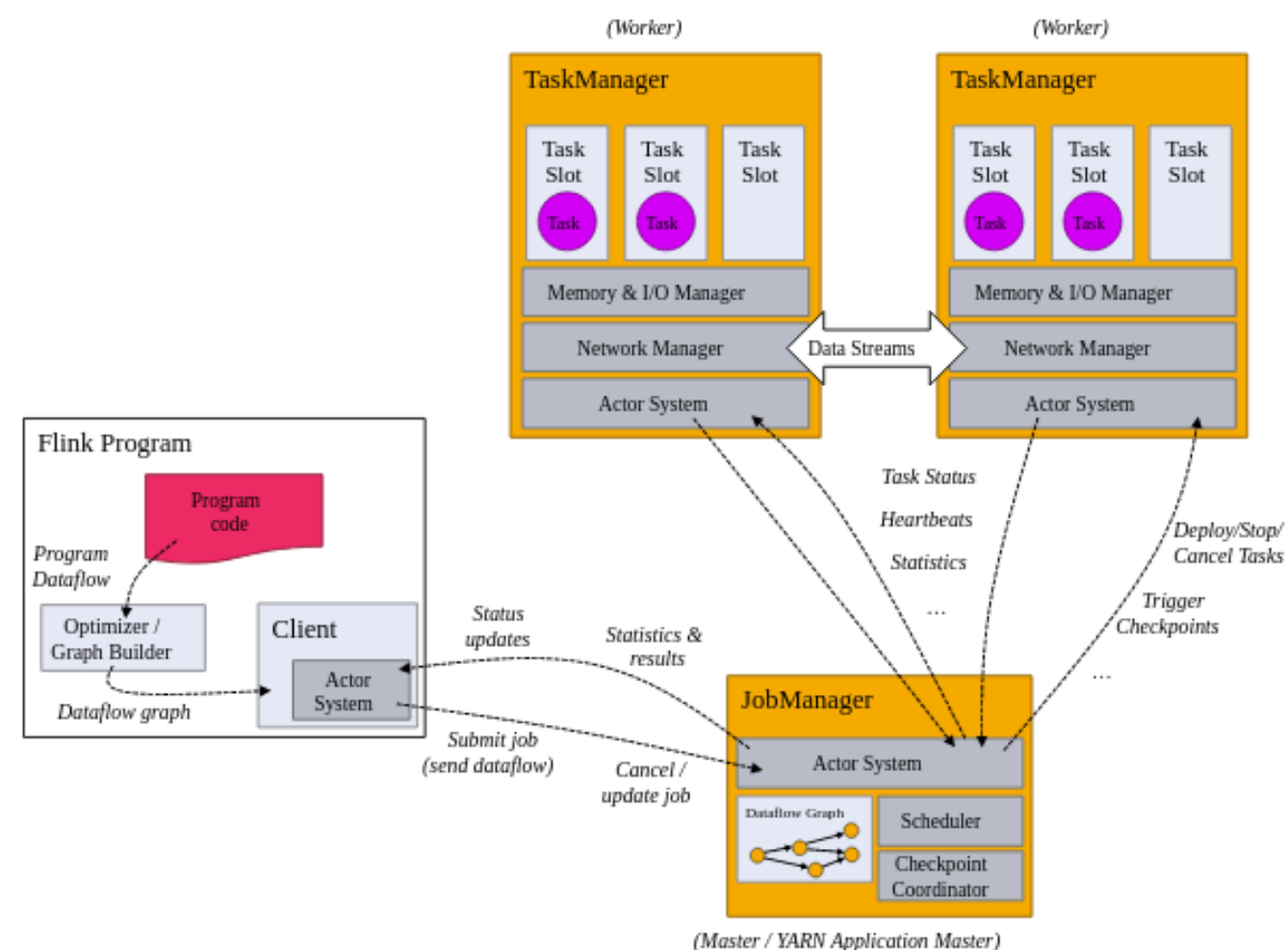
Linha do tempo



Comunidade



Arquitetura- Apache Flink



A execução do Flink consiste em dois tipos de processos: *Job Managers* e *Task Managers*

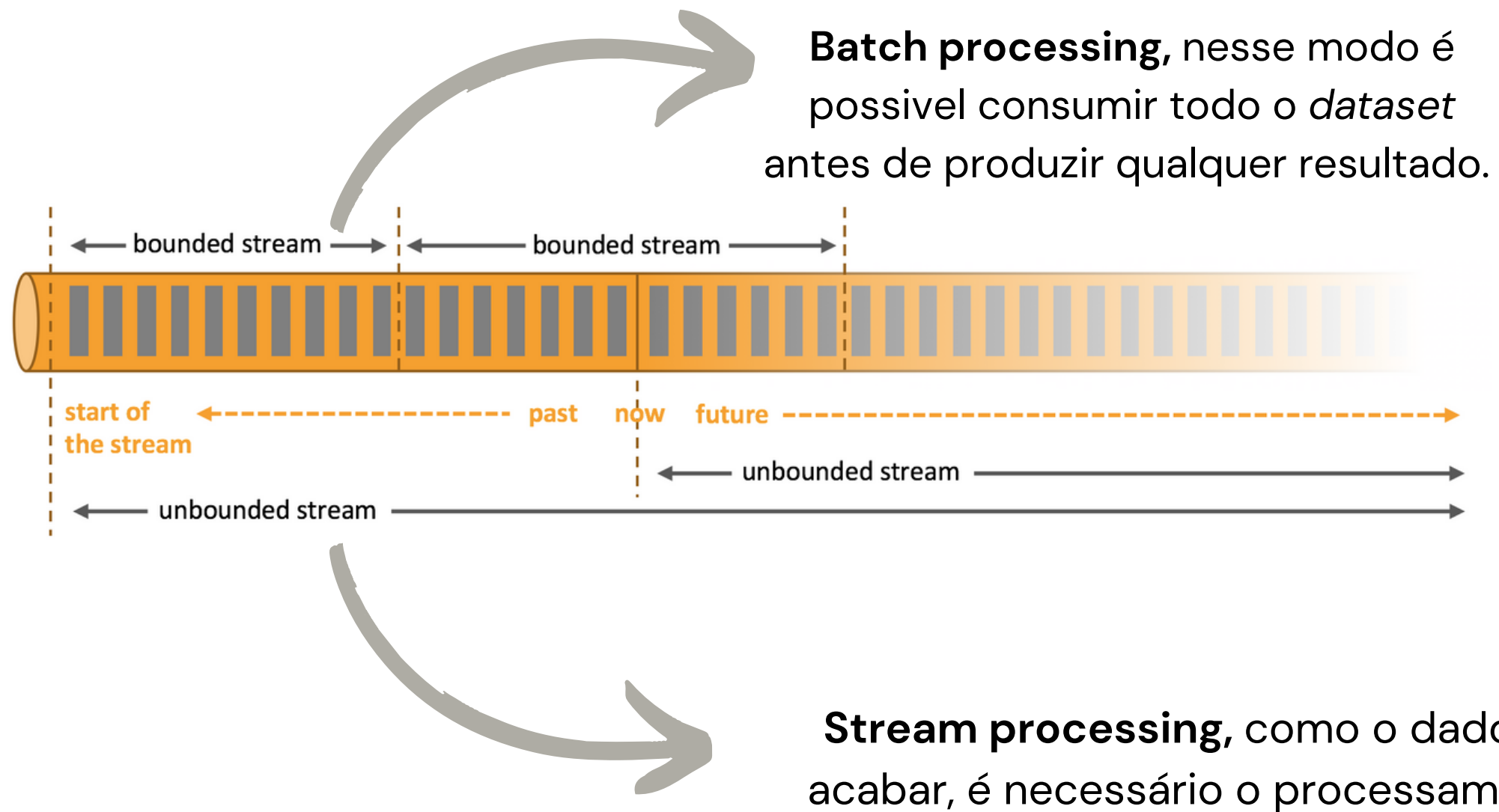
Job Manager, é responsável por coordenar a execução das aplicações do Flink.

- Agenda uma *Task*.
- Reage a *Tasks* finalizadas ou erros.
- Define *checkpoints* e coordena a recuperação de falhas

Task Manager, executa as *tasks* de um *dataflow*.

- Deve sempre haver um.
- Sua menor unidade é de agendamento de recurso é uma *Task Slot*.
- O número de *Task Slots* define o numero de *Tasks* concorrentes.
- Várias operações podem ser executadas em um *Task Slot*

Stream/Batch Process

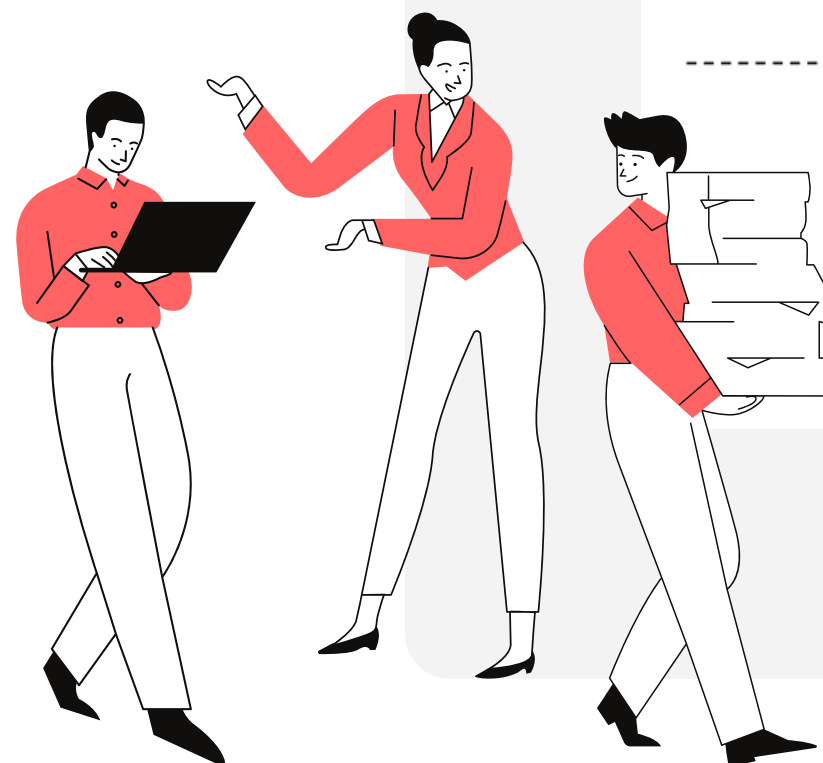


Quando é realizada a análise de dados podemos organizar o processamento em relação a *bounded* ou *unbounded Streams*, sendo essa uma escolha crucial para o projeto que deseja realizar.

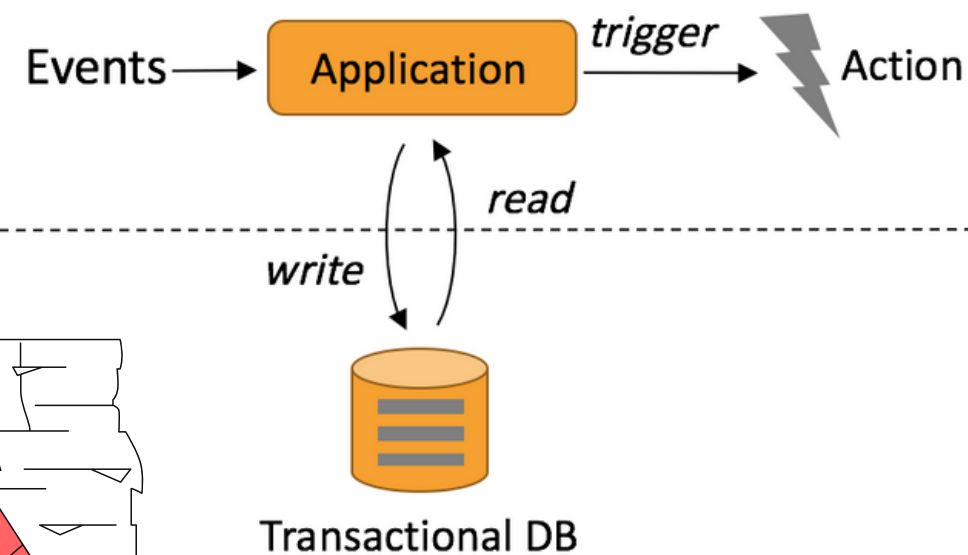
Aplicações – Event-driven

Traditional transactional application vs Flink

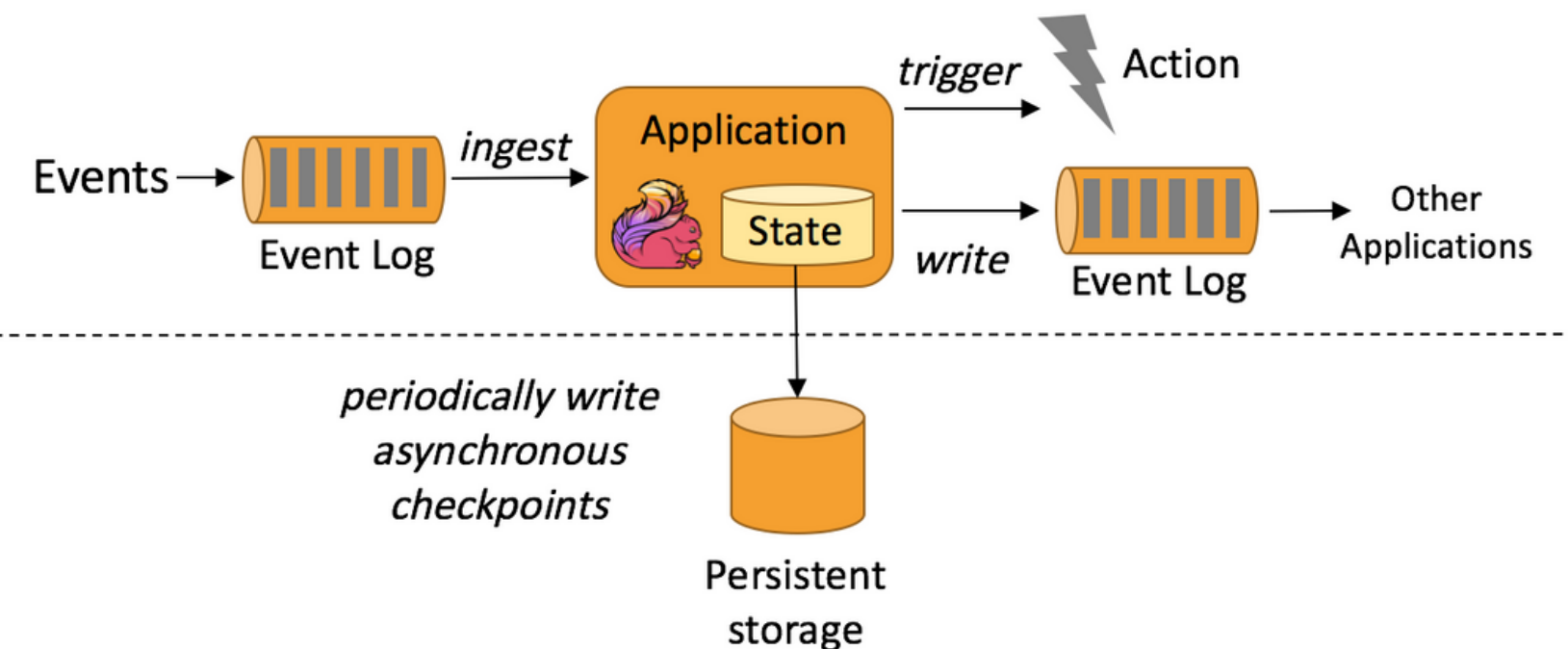
O Apache Flink opera aplicações orientados a eventos reagindo acionando calculos, atualizações



Traditional transactional application



Event-driven application

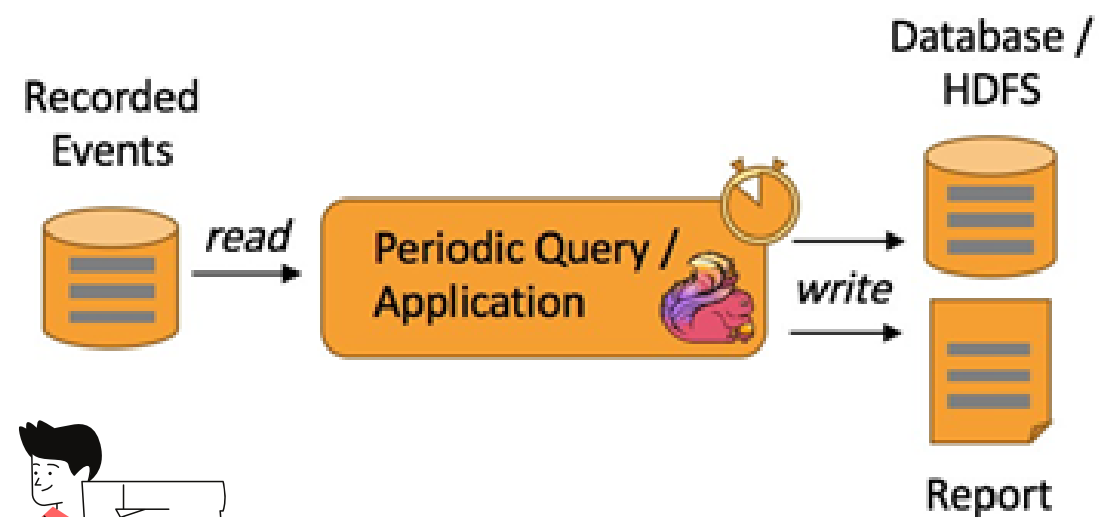


Aplicações – Data Analytics

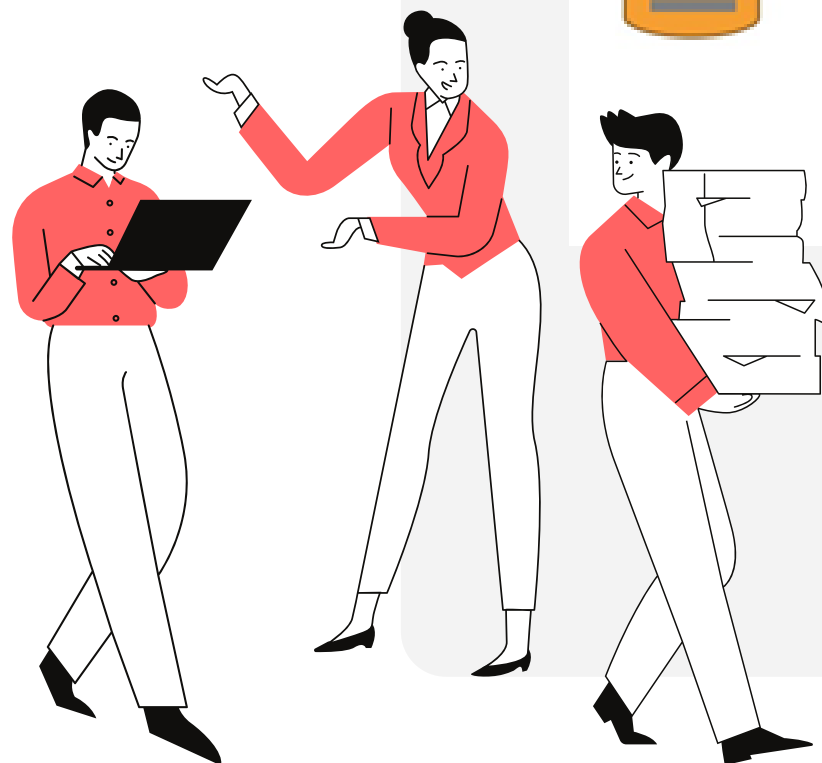
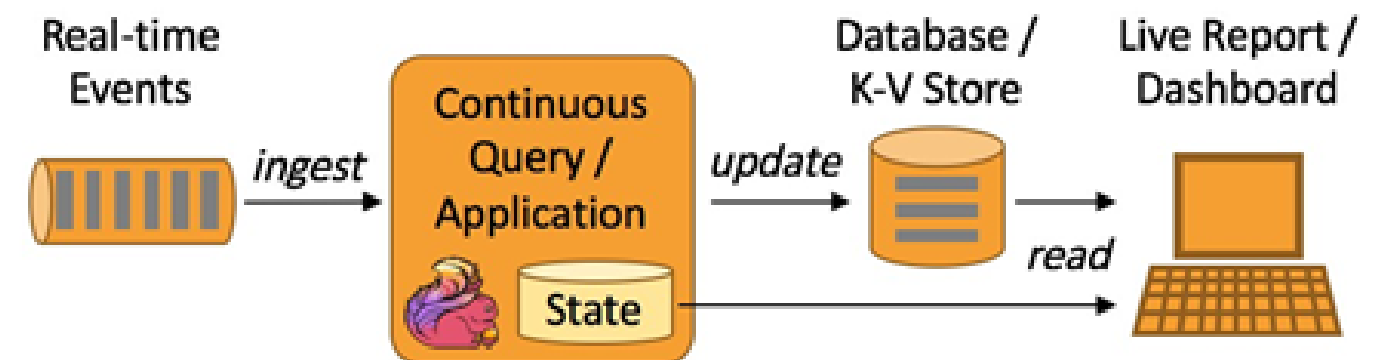
Stream ou Batch

O Apache Flink suporta *streaming processing*, bem como *Batch processing*

Batch analytics



Streaming analytics

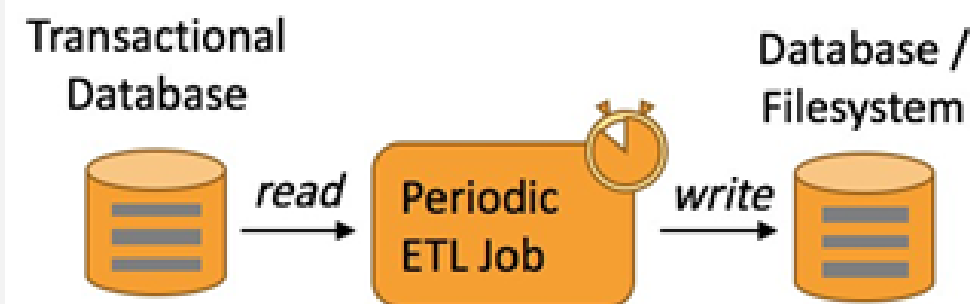


Aplicações – Data Pipeline

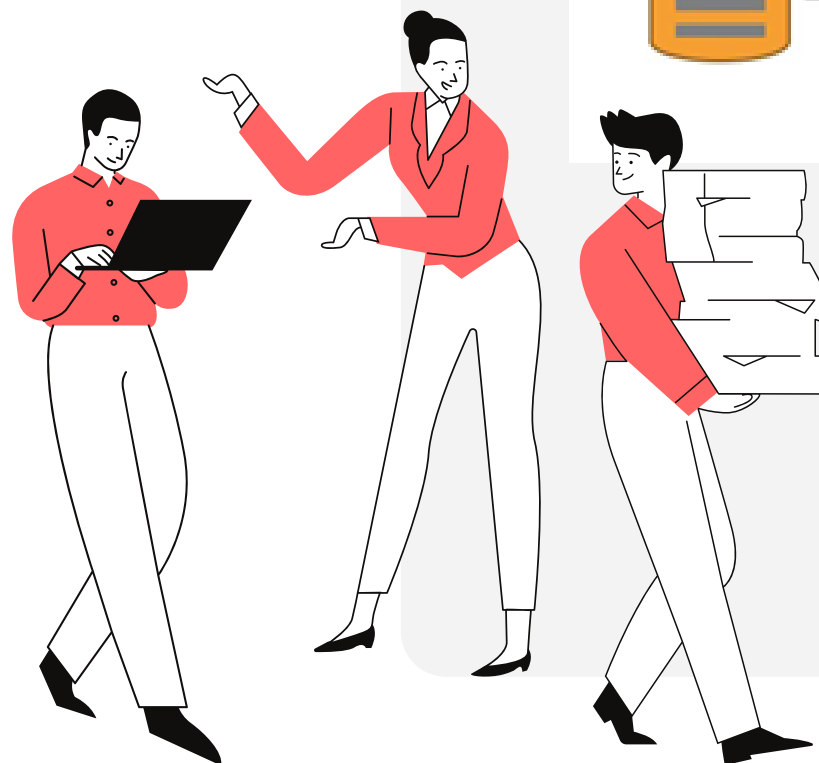
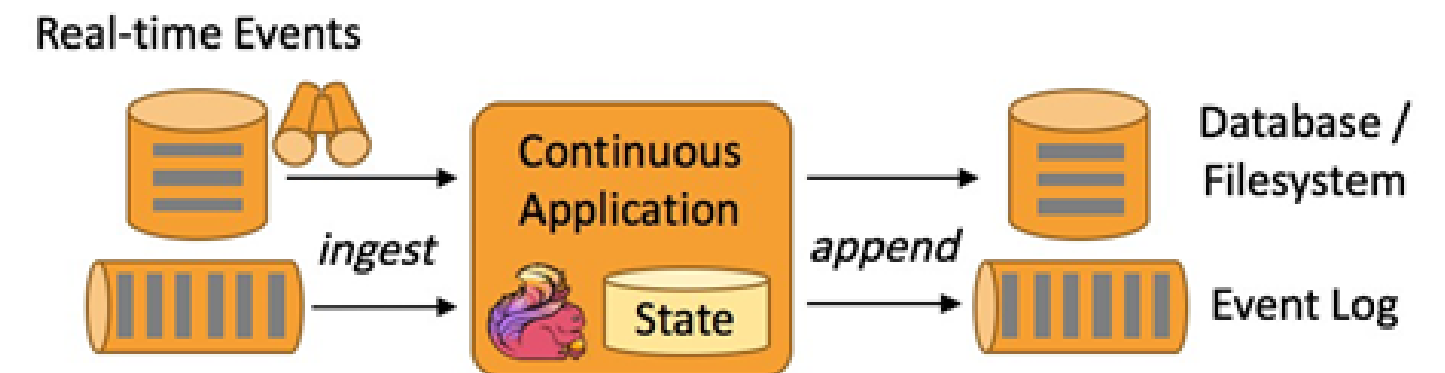
Extract-transform-load (ETL)

ETL X pipelines de dados contínuos.

Periodic ETL



Data Pipeline



DataStream API

```
DataStream<String> lines = env.addSource(new FlinkKafkaConsumer(...));
```

} Source

```
DataStream<Event> stream = lines.flatMap(line -> parse(line));
```

} Transformation

```
DataStream<Alert> stats = stream  
    .keyBy(event -> event.getId())  
    .process(new MyProcessFunction());
```

} Transformation

```
stats.addSink(new ElasticSearchSink());  
stats.addSink(new FlinkPulsarProducer());
```

} Sink

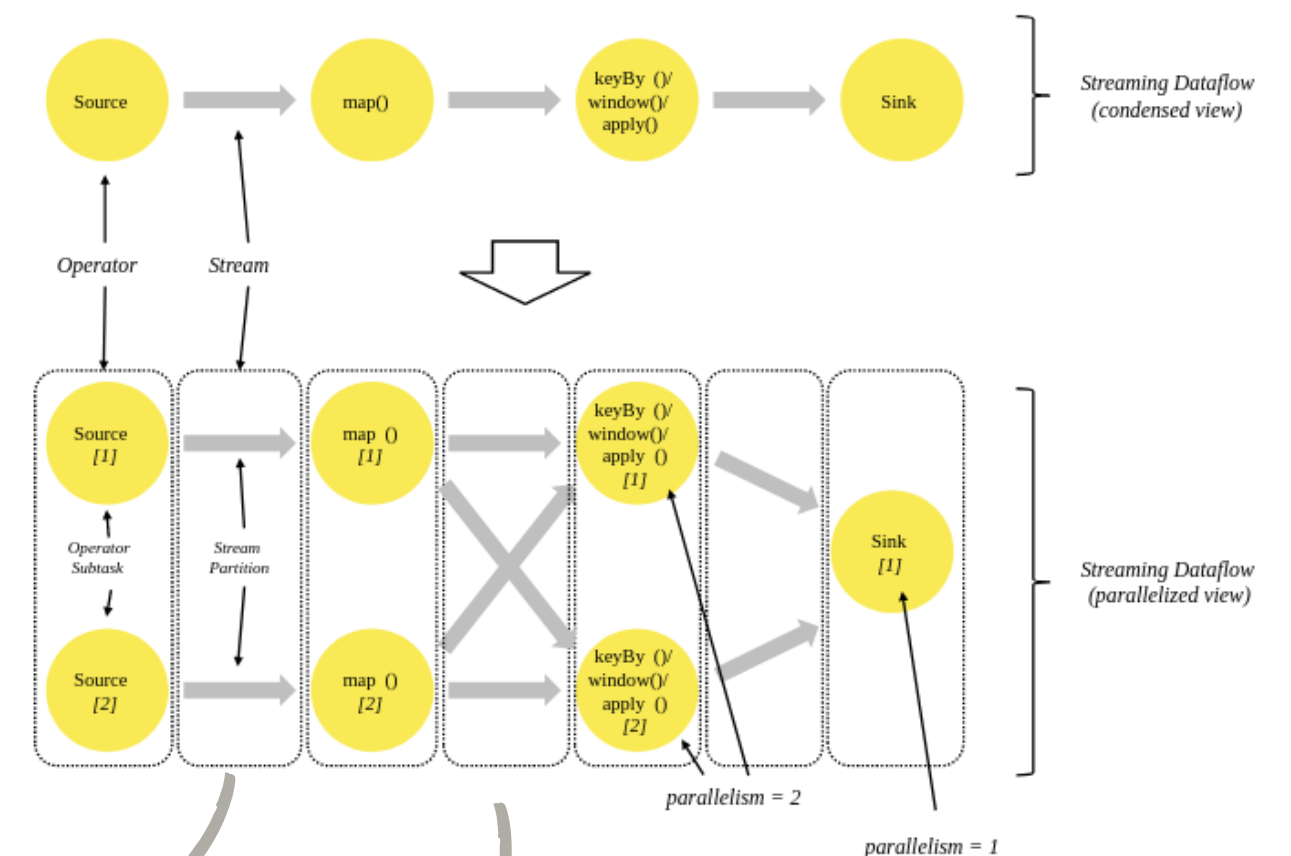
Parallel Dataflows

Programas em Flink são paralelos e distribuídos. Logo, durante a execução uma *stream* tem uma ou mais *streams partitions*, e cada operador possui uma ou mais **subtasks** que executam em paralelo.

Define o grau de paralelismo de um operador, podendo ser diferente para cada tipo de operador durante o *streaming dataflow*

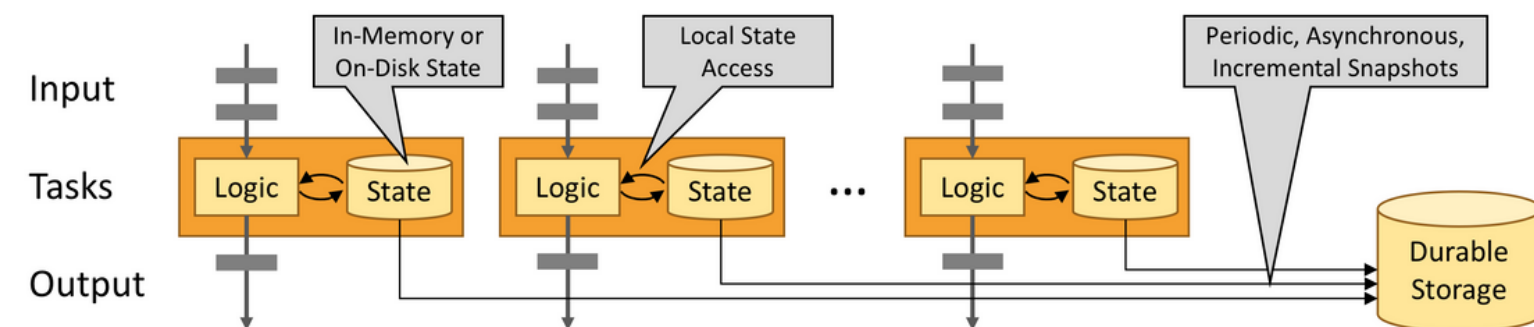
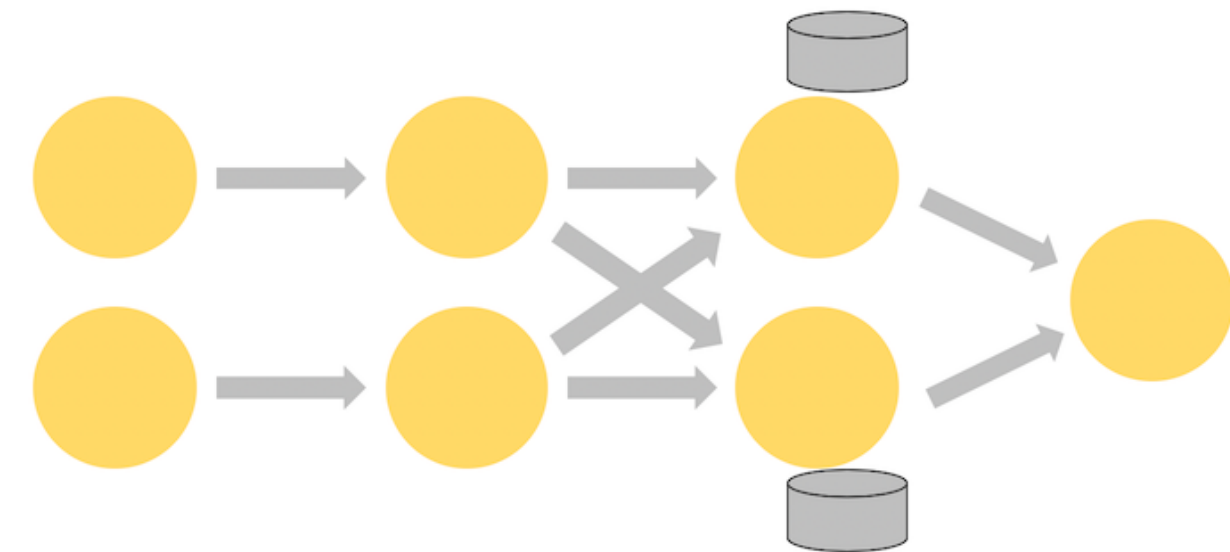
One - One, mantém a ordem e o particionamento dos dados

Redistributing, mantém a ordem dos dados somente entre o par de operadores (*send, receive*)



Stateful Stream Processing

As operações do Flink podem possuir estados, pra isso é necessário que o conjunto de operações em paralelo sejam um tipo de chave que é armazenada localmente em uma instância paralela.

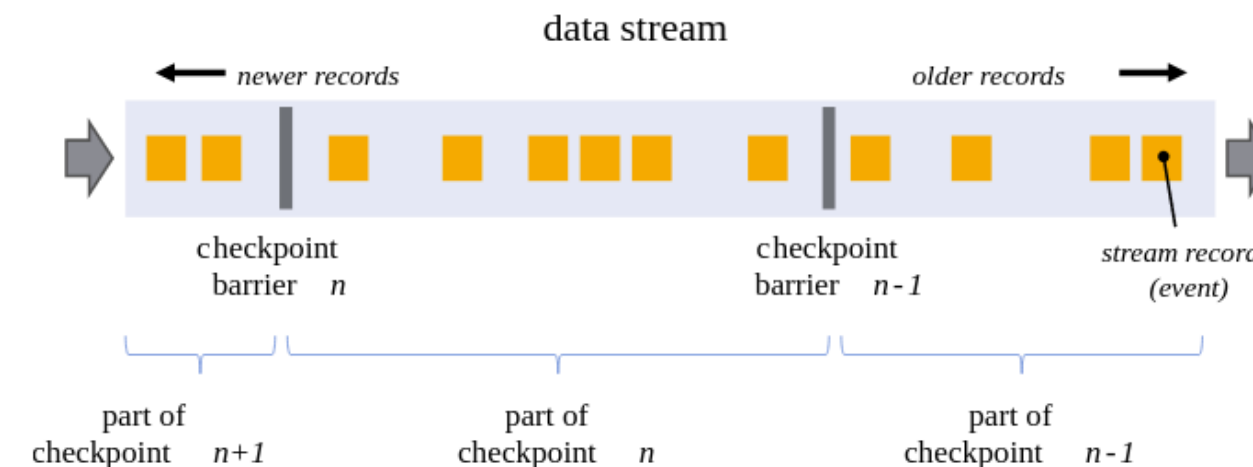


O *state* é sempre acessado localmente
=
Low Latency & High throughput

Fault Tolerance via State Snapshots

Flink faz uso de combinações de *state snapshots* e *stream replay* para ser tolerante a falhas

Quando ocorre uma falha as fontes de dados são restauradas, o estado é restaurado e o processo continua.





UBER
Athena X Streaming SQL Platform
Service



King, os criadores do Candy Crush Saga, usa o Flink para fornecer às equipes de ciência de dados um painel de análise em tempo real.

Quem utiliza?



100 jobs, 1000 nodes, TBs state metrics, analytics, real time ML Streaming SQL as a platform

Empresas que fazem uso da Apache Flink em seus serviços



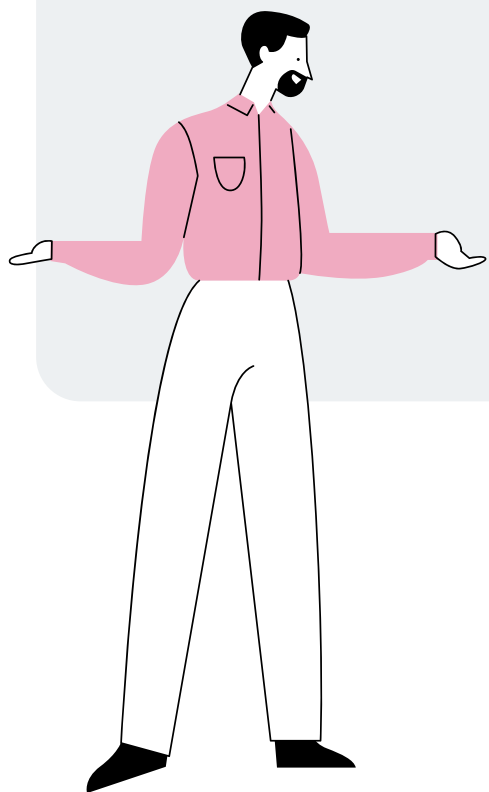
A Plataforma de Anonimização de Dados com certificação TÜV da Telefónica NEXT é fornecida pela Flink.

Diferenciais

É o que existe de mais moderno em computação distribuída e engines do ecossistema Hadoop

É a próxima geração para Big Data Analytics framework e não apenas uma alternativa para Hadoop MapReduce, mas também uma alternativa ao Apache Storm e Spark.

Trás uma visão única e filosofia de framework engine que é verdadeiramente híbrida (Real-Time Streaming + Batch), distribuída, suporta batch, streaming, queries relacionais, machine learning e processamento de gráficos.



Flink vs Spark vs Hadoop

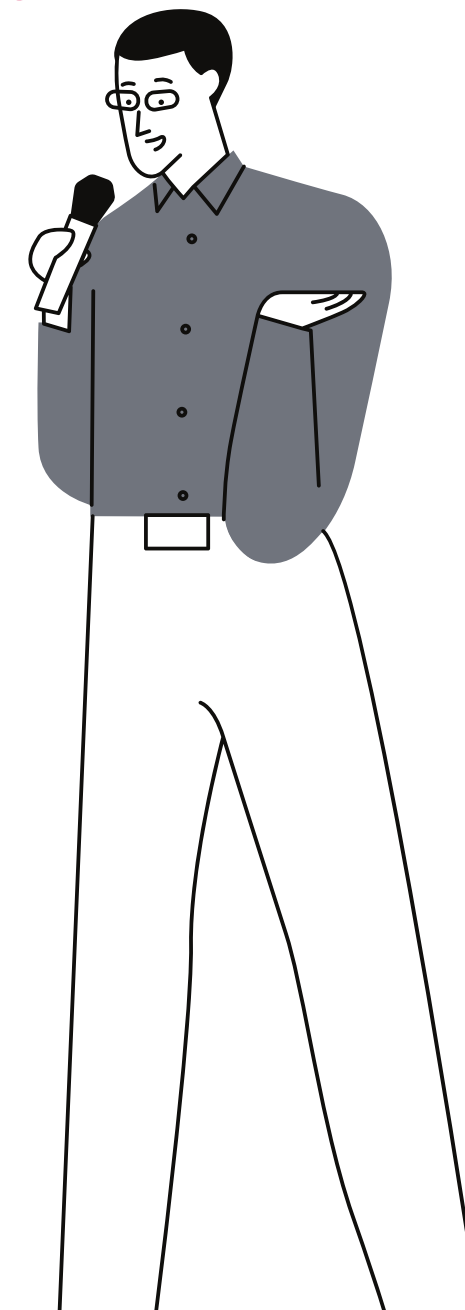
ANO DE ORIGEM
MECANISMO DE PROCESSAMENTO DE DADOS
VELOCIDADE DE PROCESSAMNETO
LINGUAGENS DE PROGRAMAÇÃO
TRANFERENCIA DE DADOS
GERENCIAMENTO DE MEMÓRIA
LATENCIA
OTIMIZAÇÃO

APACHE HADOOP	APACHE SPARK	APACHE FLINK
2008	2009	2009
LOTE	LOTE	CORRENTE
+ Lento que o Spark e Flink	100X + rapido que o Hadoop	Mais rapido
Java, C,C++, Ruby, Groovy, Perl, Python	Java, Scala, Python e R	Java, Scala e <i>Python (muito pouca documentação)</i>
Lote	Lote	Pipelined e Batch
Baseado em disco	JVM gerenciado	Gerenciamento ativo
Baixo	Médio	Baixo
Manual	Manual	Automatico



Flink na Cloud

Cloud

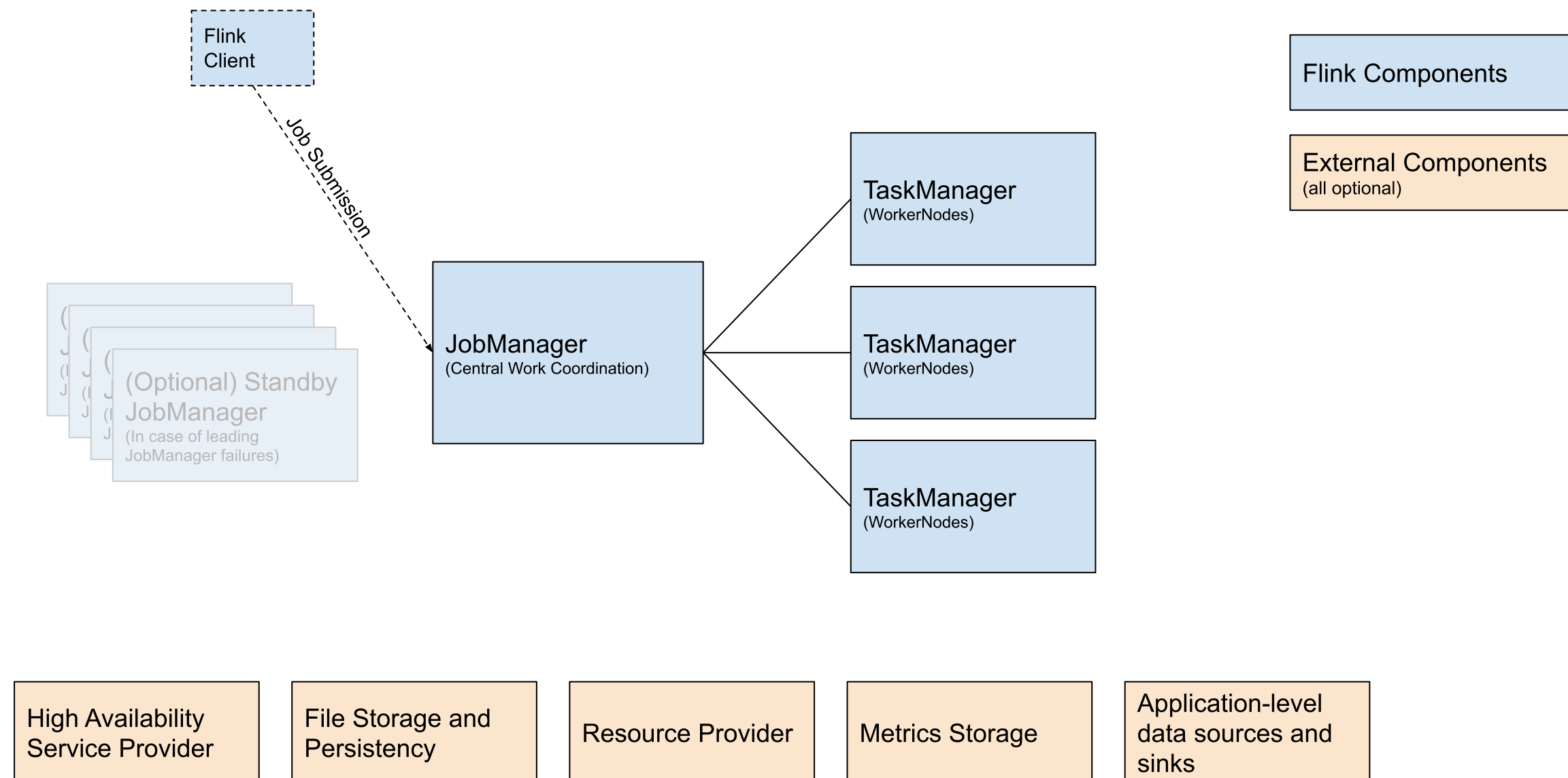


Deployment

O Flink é um *framework* versátil, suportando muitos cenários de implementação.

O cluster obtém o código da aplicação Flink, transforma em um JobGraph e o envia ao JobManager

O JobManager distribui o trabalho nos TaskManagers (onde os operadores reais estão em execução)



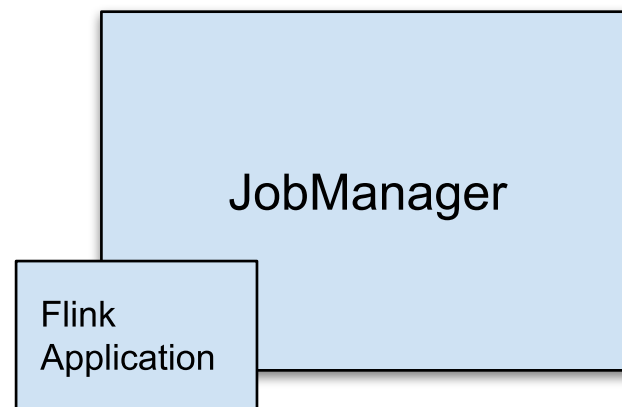
Deployment Modes

Flink executa aplicações de três maneiras distintas

Os modos diferem em:

- Ciclo de vida do cluster e as garantias de isolamento
- Se o *main()* é executado no ou no cluster

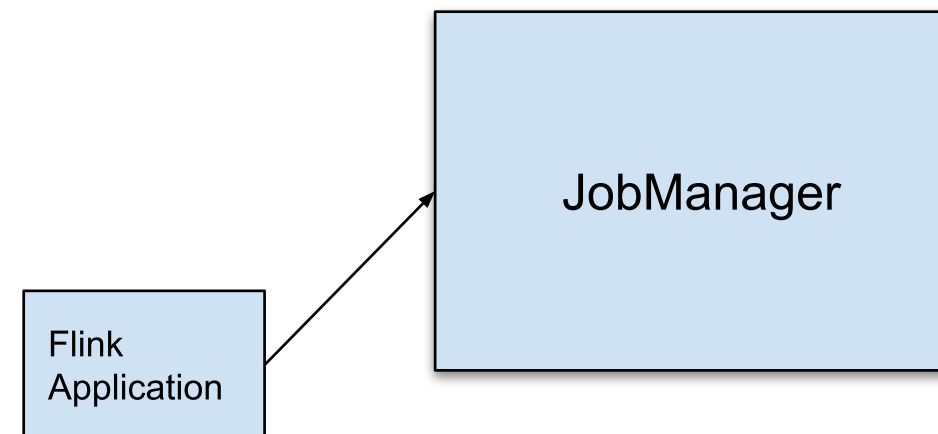
Application Mode



A dedicated JobManager is started for submitting the job. The JobManager will only execute this job, then exit.
The Flink Application runs on the JobManager.

O `main()` é executado no cluster e não no cliente, criando um cluster por aplicação enviada

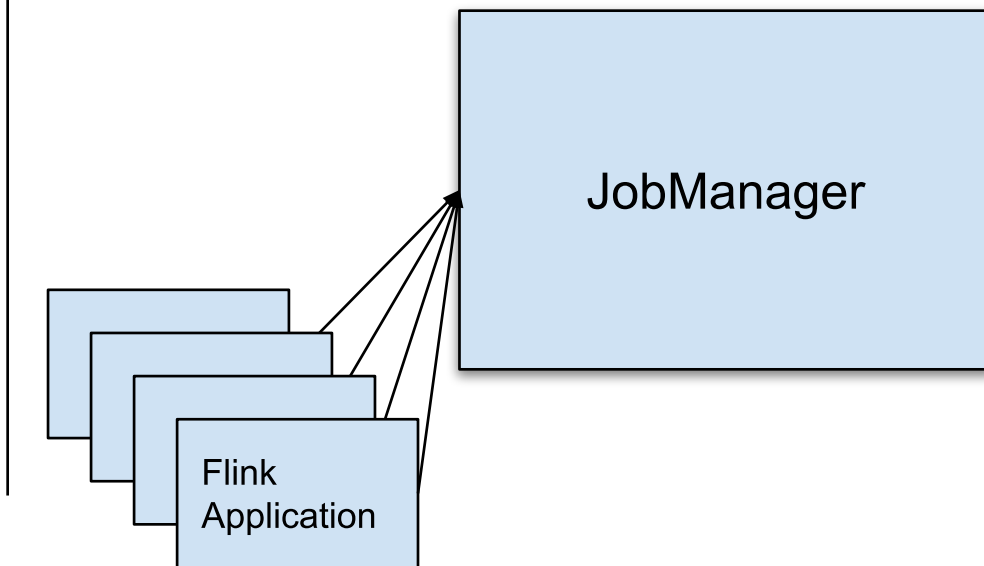
Per-Job Mode



A dedicated JobManager is started for submitting the job. The JobManager will only execute this job, then exit.
The Flink Application runs on the client submitting the per-job cluster

Melhores garantias de isolamento de recursos, utiliza a estrutura do provedor de recursos disponíveis (YARN, Kubernetes) para ativar um cluster para cada trabalho enviado

Session Mode

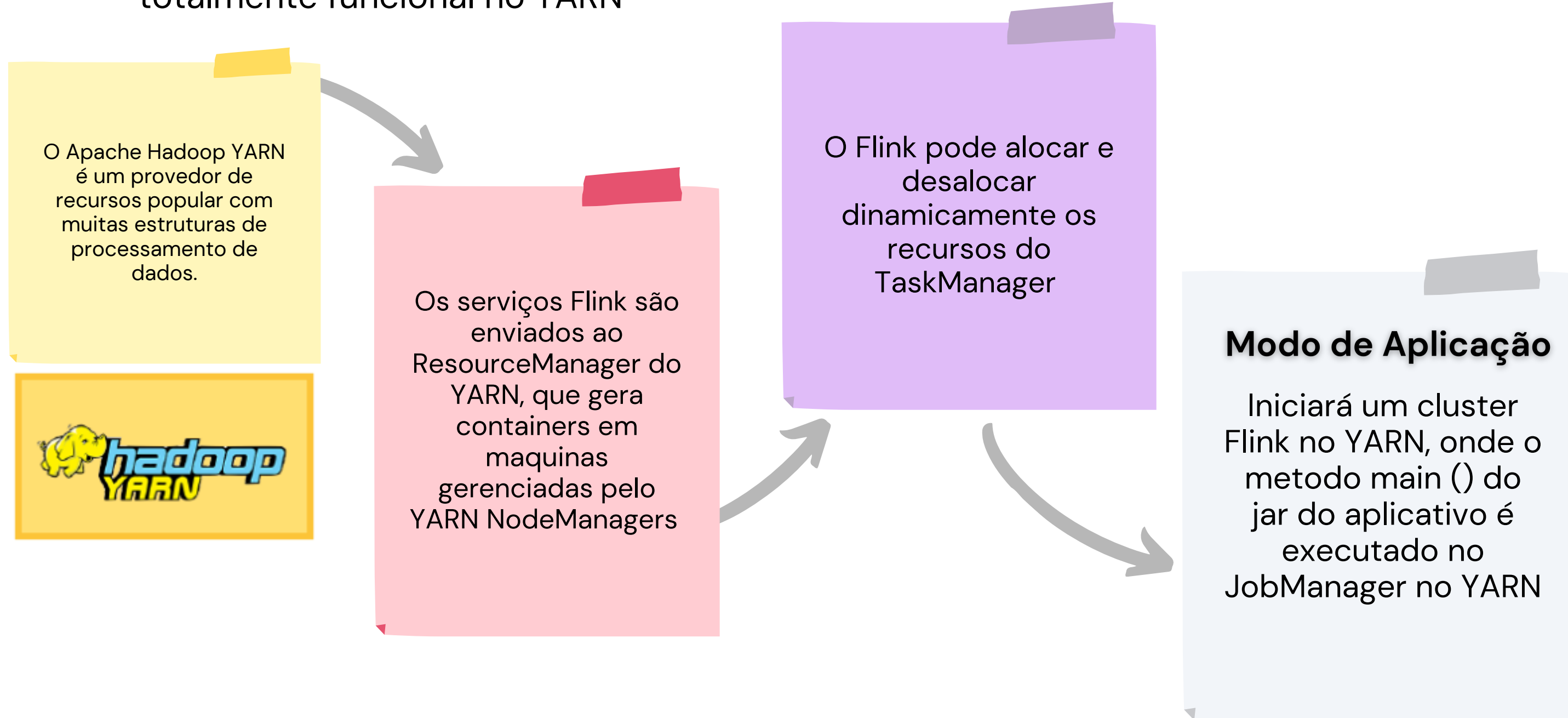


Multiple jobs share one JobManager.

Assume um cluster já em execução e usa os recursos desse cluster para executar qualquer aplicação enviada. As aplicações executadas no mesmo cluster usam e competem pelos mesmos recursos.

Apache Hadoop YARN

Configuração de um Flink Cluster
totalmente funcional no YARN



Chegamos ao fim



Obrigado!

