Fundamentos

Bootcamp Engenharia de Dados

Capítulo 1. Conceitos Fundamentais

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

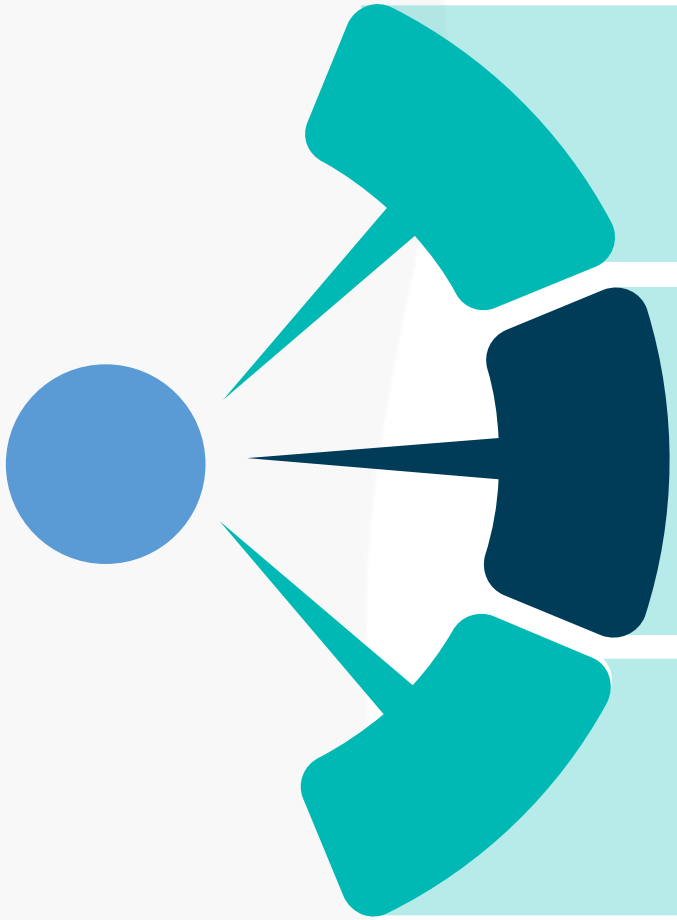
Aula 1.1. Dados, fontes de dados, Big Data, tipos de dados

Prof. Dr. Neylson Crepalde

Nesta aula

- Dados.
- Fontes de Dados.
- Big Data.


DADO
INFORMAÇÃO
CONHECIMENTO



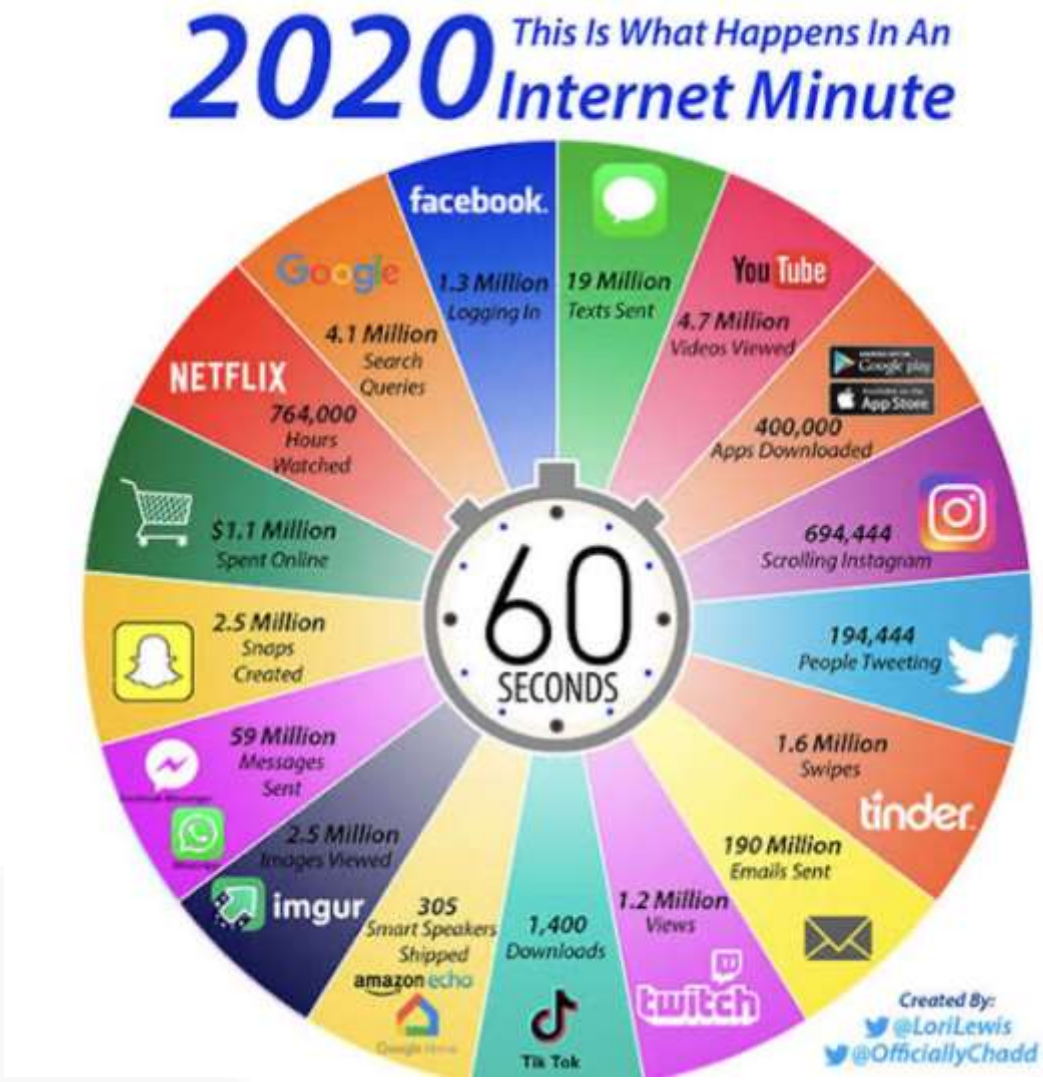
12.965

Lei 12.965

Lei 12.965 - Marco Civil da Internet

Big Data

- Volume.
- Velocidade.
- Variedade.


2020 This Is What Happens In An Internet Minute
facebook
1.3 Million
Logging In
19 Million
Texts Sent
YouTube
4.7 Million
Videos Viewed
400,000
Apps Downloaded
694,444
Scrolling Instagram
194,444
People Tweeting
1.6 Million
Swipes
tinder
190 Million
Emails Sent
1.2 Million
Views
twitch
1,400
Downloads
Tik Tok
305
Smart Speakers
Shipped
amazon echo
2.5 Million
Images Viewed
imgur
59 Million
Messages
Sent
2.5 Million
Snaps
Created
$1.1 Million
Spent Online
NETFLIX
764,000
Hours
Watched
Google
4.1 Million
Search
Queries
60
SECONDS
Created By:
@LoriLewis
@OfficiallyChadd

Fontes de dados

- Dados em tabela.
- Fotos.
- Vídeos.
- Textos curtos (tweets, comentários).
- Textos longos.
- Transações bancárias.
- Conexões na rede.
- Dados georreferenciados.

Dada essa grande variabilidade de dados, que ferramentas o(a) Engenheiro(a) de Dados precisa conhecer e mobilizar para processar dados de fontes tão diversas?

Repositório de dados capaz de armazenar formatos diversos (Data Lake).

Extração, Transformação e Carga (ETL).

Transformação = Limpeza, organização, estruturação, enriquecimento.

Repositório de dados analíticos (Data Warehouse).

Conclusão

Os dados em seu estado "cru" tem baixo potencial de agregar valor. Quando ele se encontra contextualizado se transforma em **informação** e, quando processado, em **conhecimento**.

O Big Data, seu grande volume, velocidade e variedade de dados e fontes exige do(a) Engenheiro(a) de Dados conhecimentos sólidos sobre Data Lake, ETL e DW.

Na próxima aula

01.

Visão geral do pipeline de Ciência de Dados.

02.

Coleta - Preparação – Armazenamento.

03.

Processamento - Análise – Visualização.

04.

Implantação.

Fundamentos

Bootcamp Engenharia de Dados

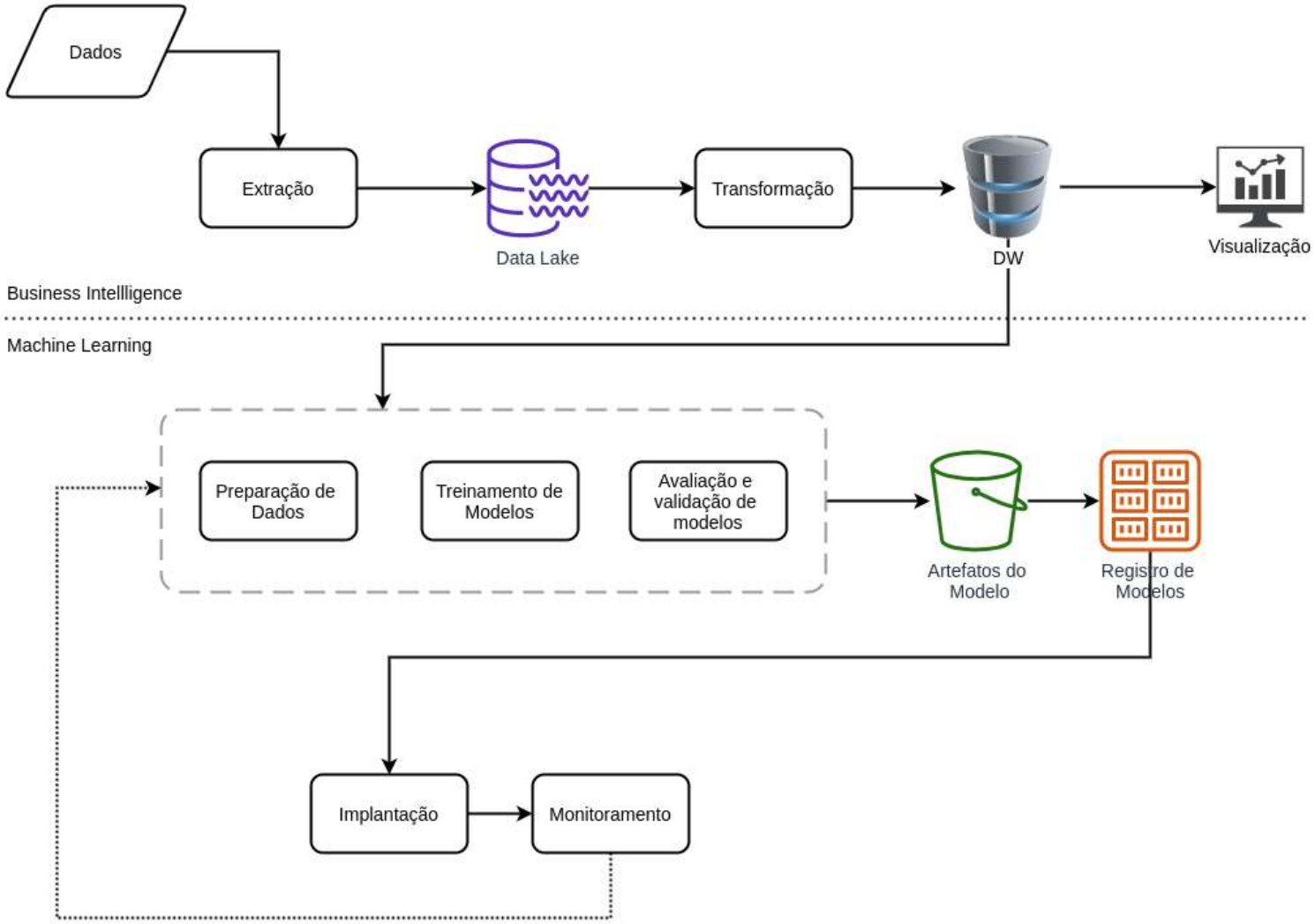
Aula 1.2. O Pipeline de Ciência de Dados

Prof. Dr. Neylson Crepalde

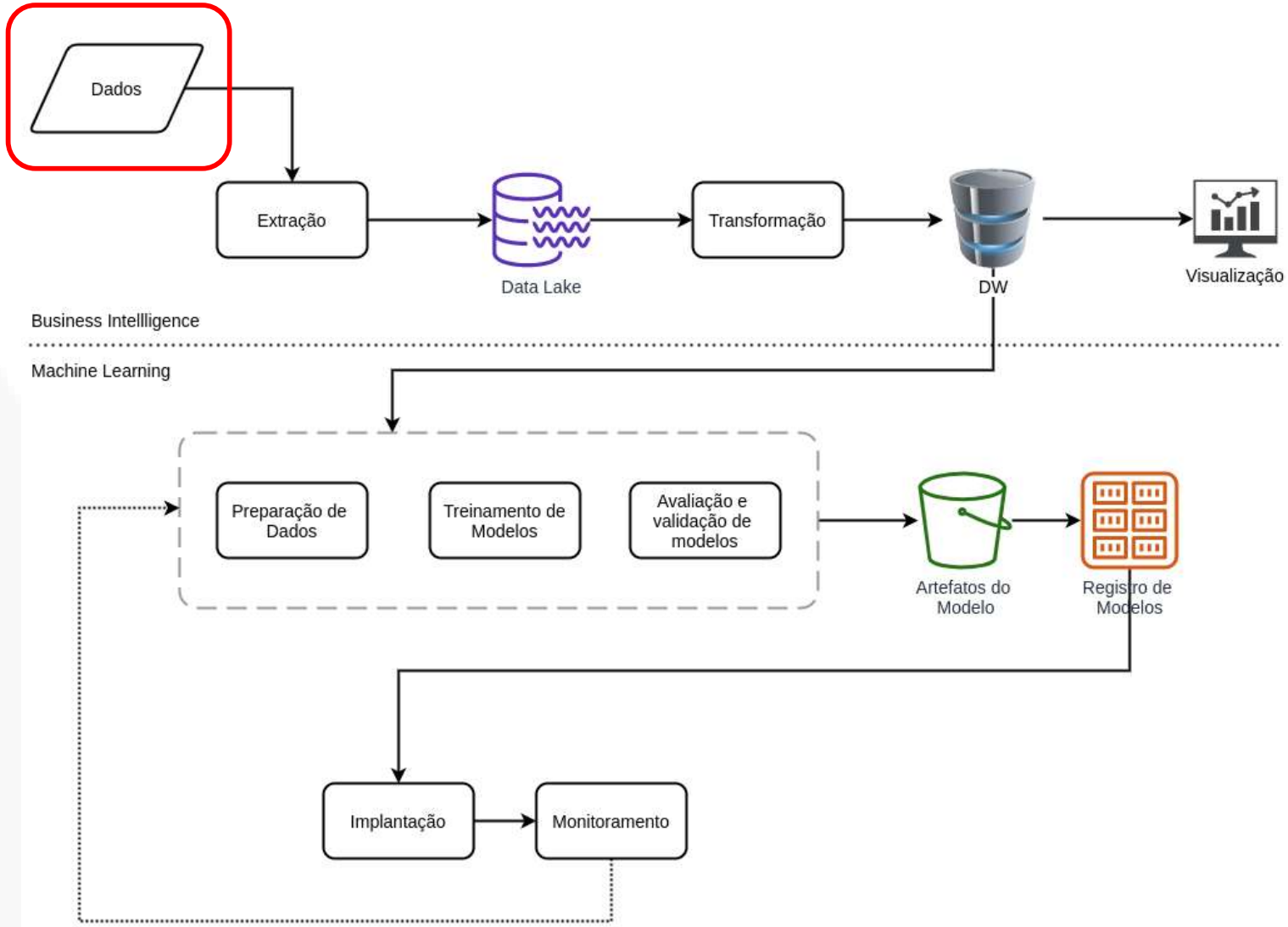
Nesta aula

- Pipeline de Ciência de Dados.
- Coleta, preparação e armazenamento.
- Processamento, análise e visualização.
- Implantação.
- Noções de DataOps/MLOps.

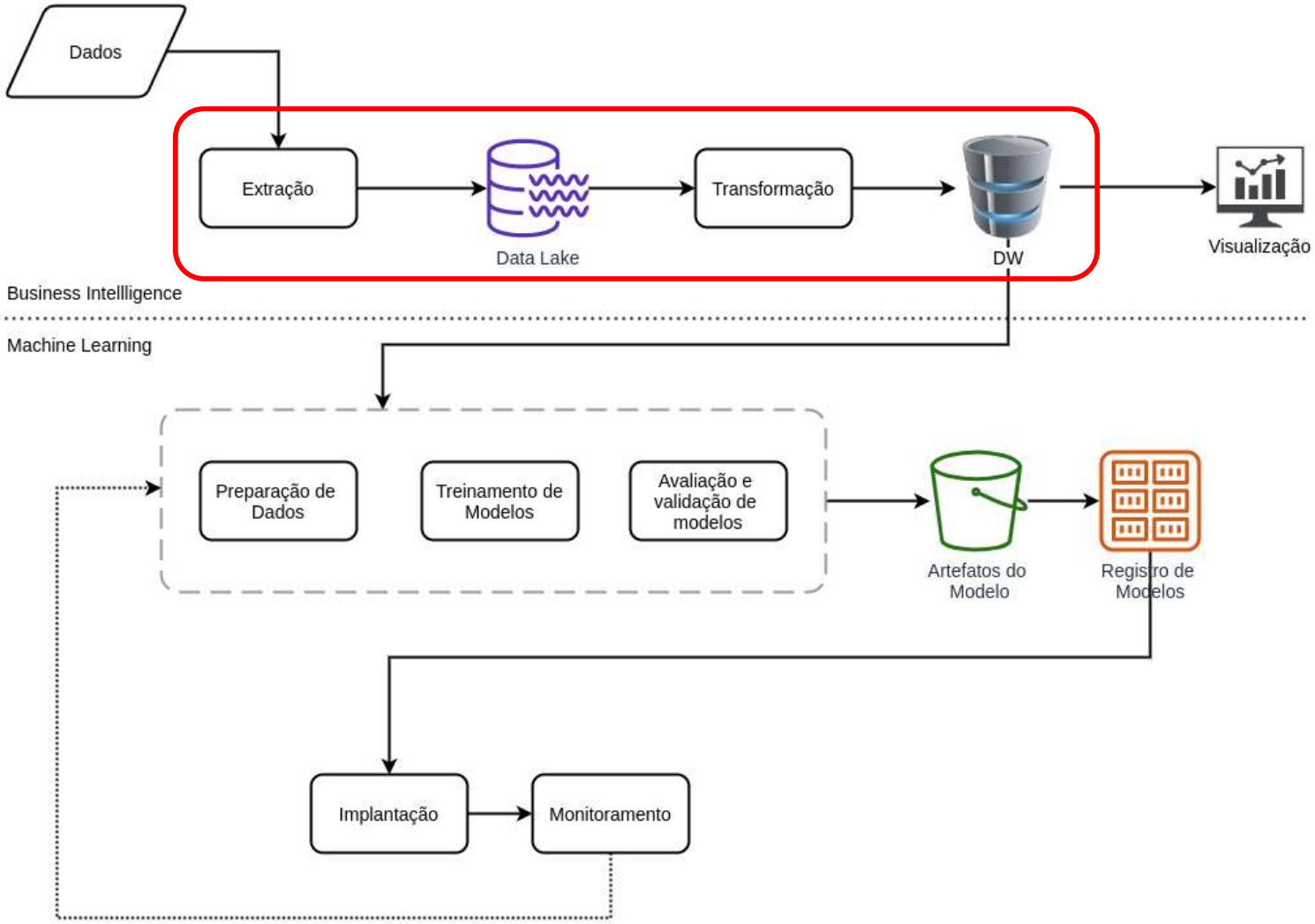
Pipeline de Ciência de Dados



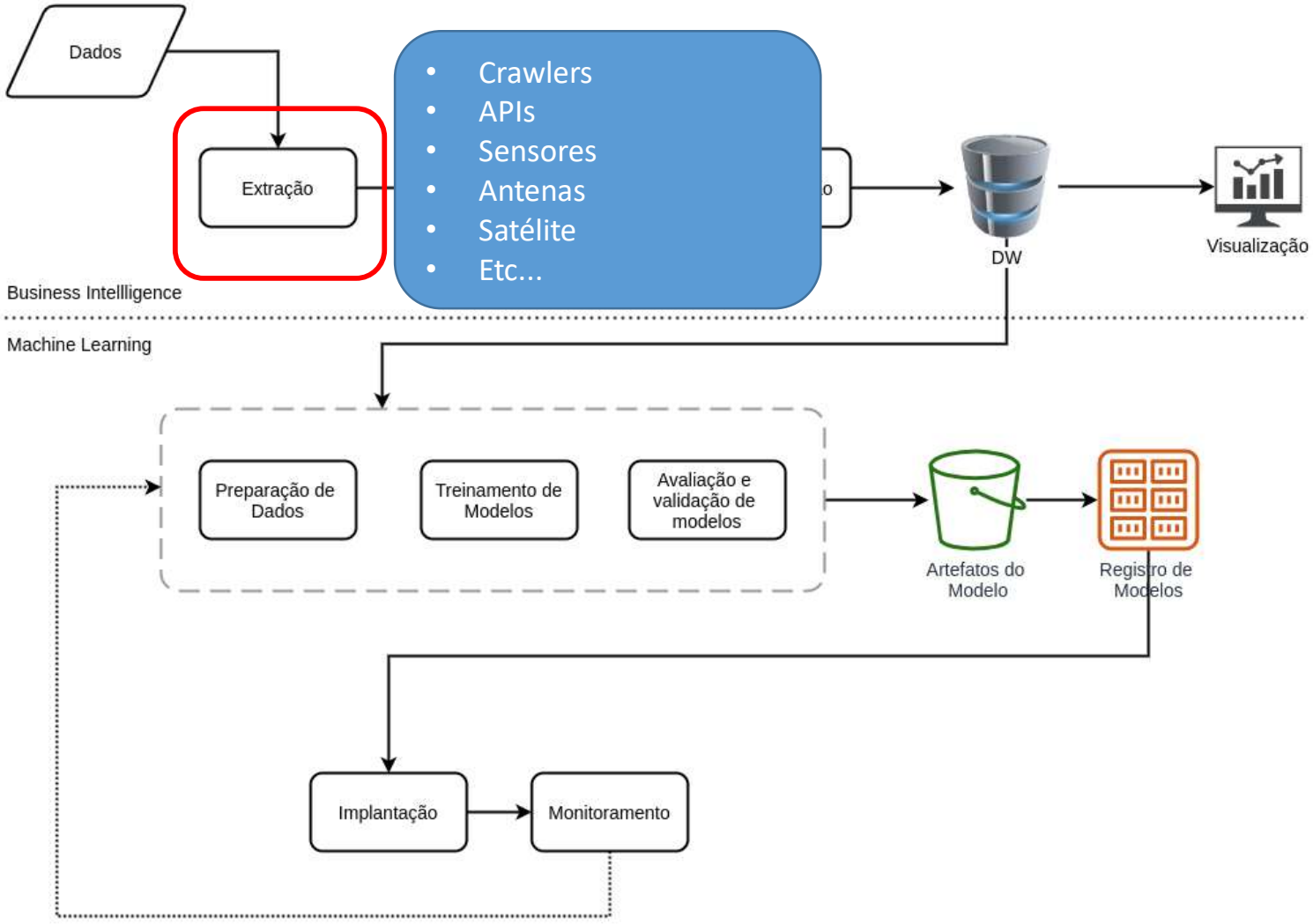
Pipeline de Ciência de Dados



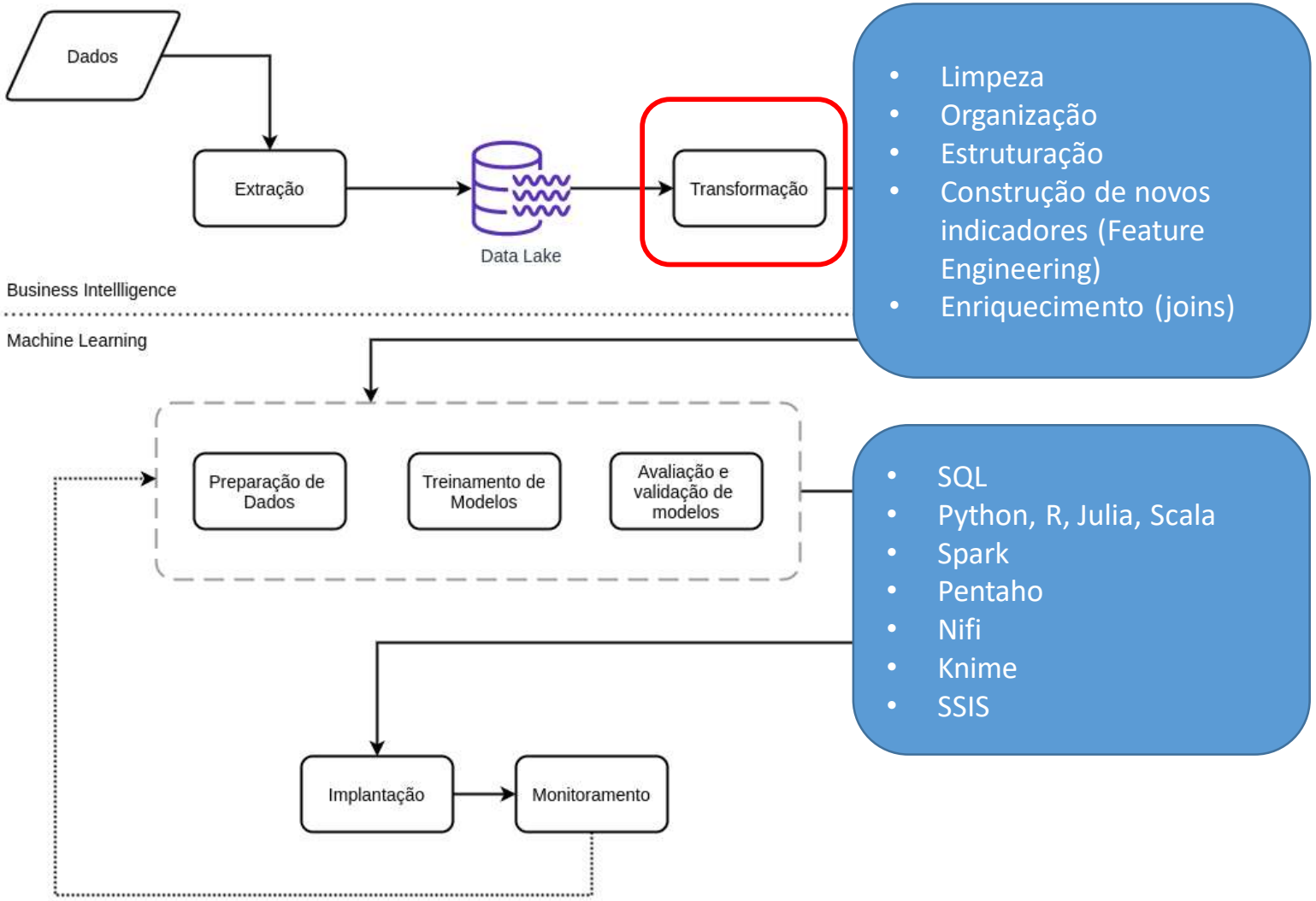
Pipeline de Ciência de Dados



Pipeline de Ciência de Dados



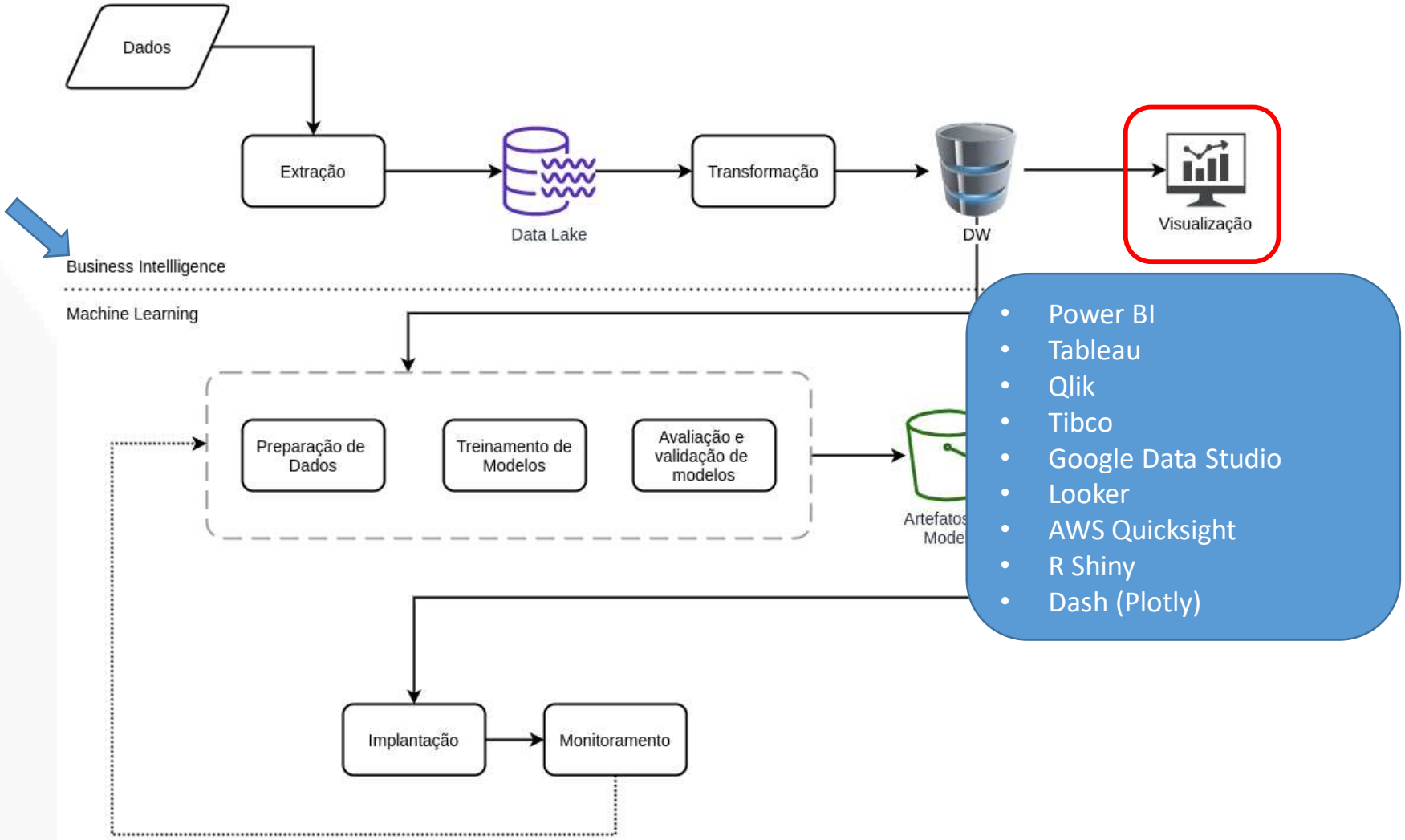
Pipeline de Ciência de Dados



- Limpeza
- Organização
- Estruturação
- Construção de novos indicadores (Feature Engineering)
- Enriquecimento (joins)

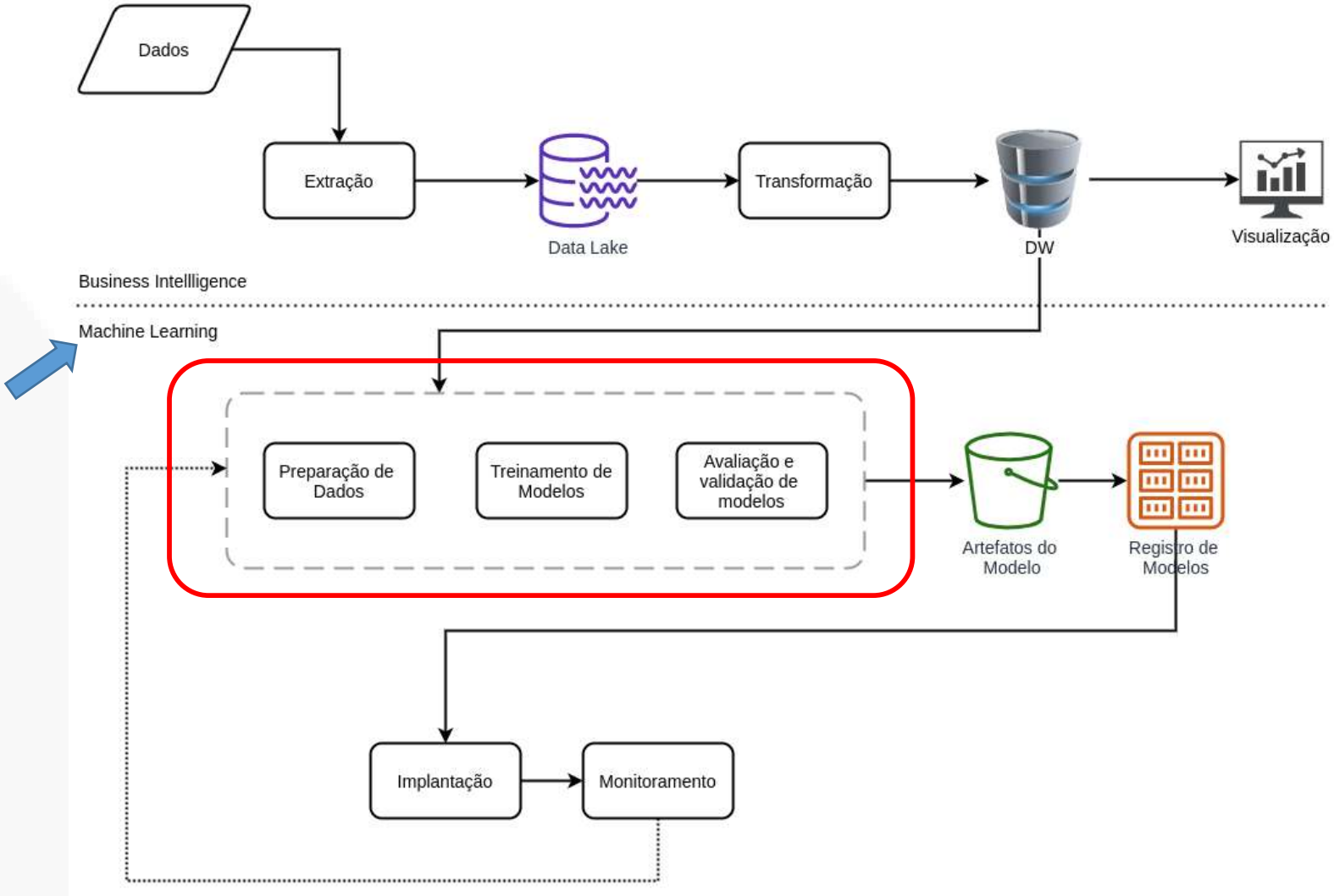
- SQL
- Python, R, Julia, Scala
- Spark
- Pentaho
- Nifi
- Knime
- SSIS

Pipeline de Ciência de Dados

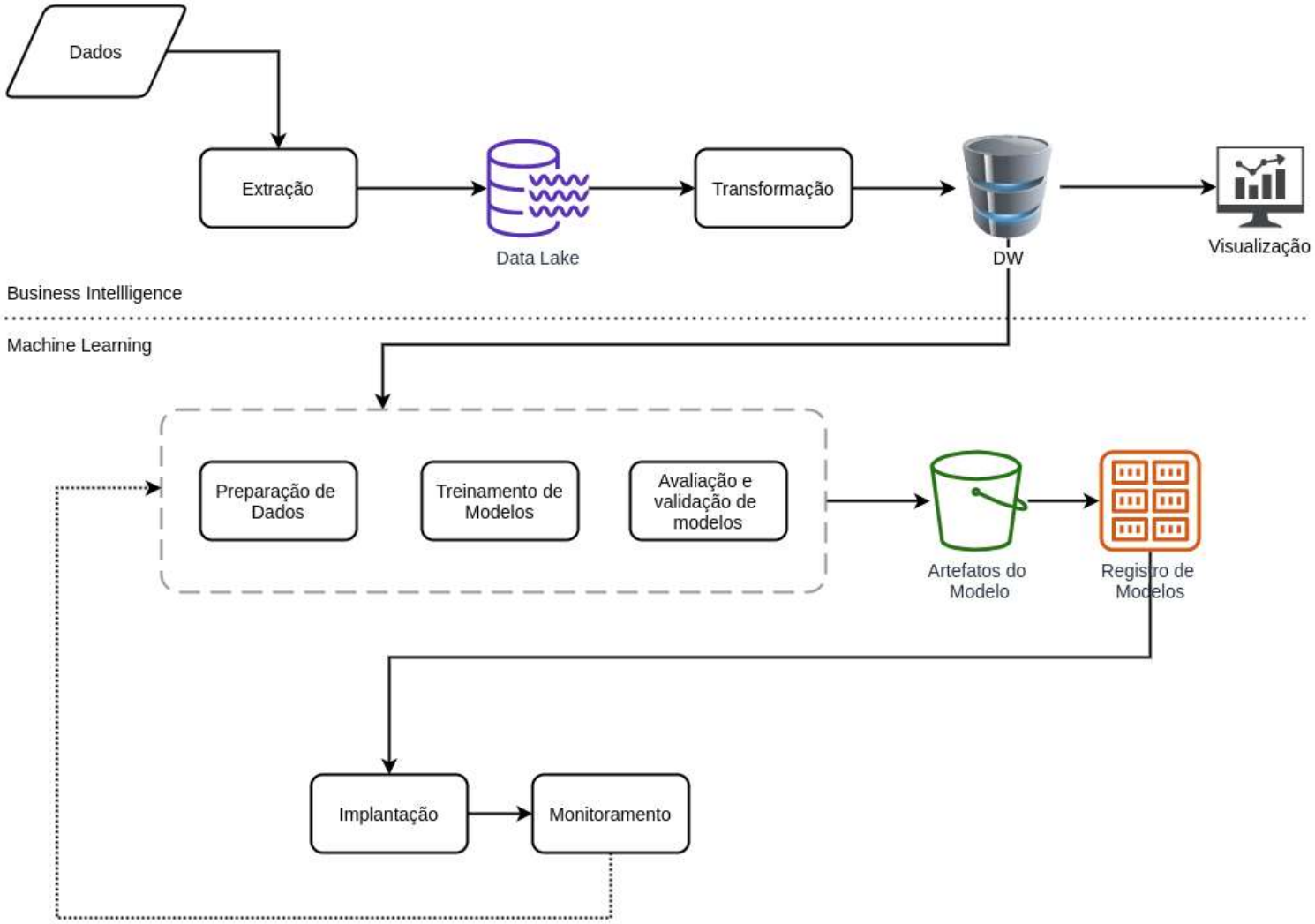


- Power BI
- Tableau
- Qlik
- Tibco
- Google Data Studio
- Looker
- AWS Quicksight
- R Shiny
- Dash (Plotly)

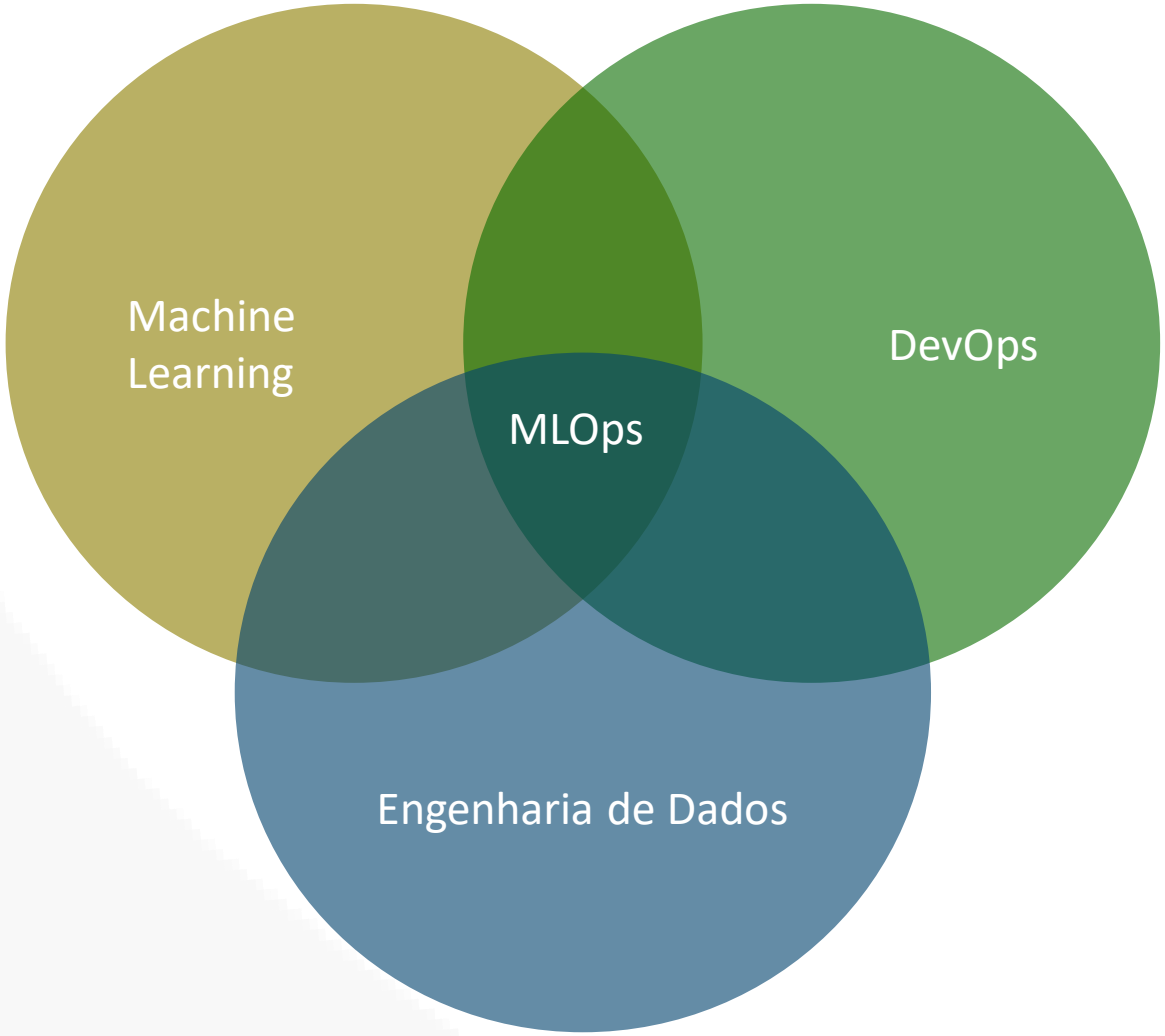
Pipeline de Ciência de Dados



Pipeline de Ciência de Dados

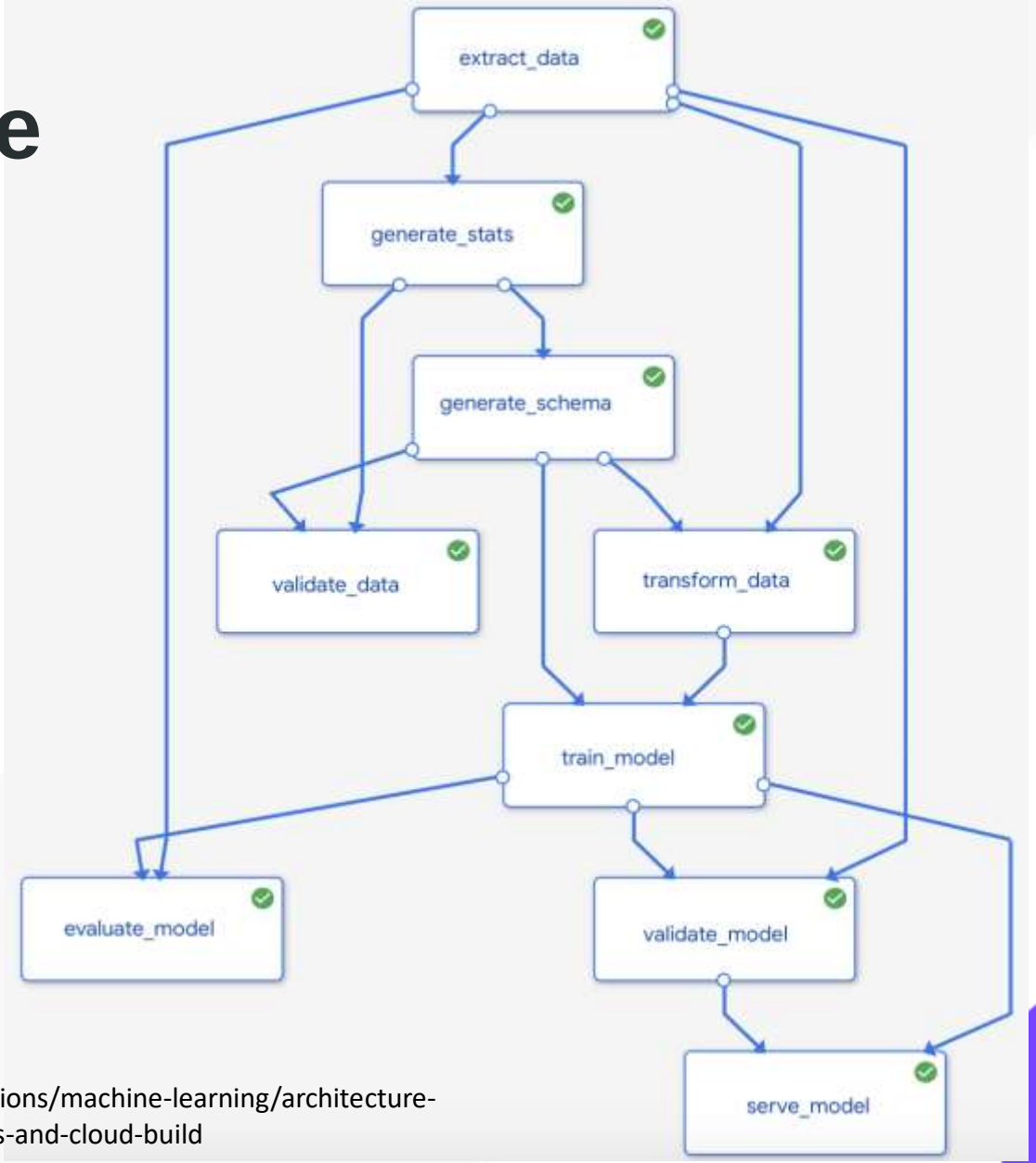


Pipeline de Ciência de Dados



Fonte: https://www.altexsoft.com/blog/mlops-methods-tools/

Pipeline



Fonte: https://cloud.google.com/solutions/machine-learning/architecture-for-mlops-using-tfx-kubeflow-pipelines-and-cloud-build

Conclusão

- ✓ O Pipeline de Ciência de Dados contempla, de maneira geral, o ETL inicial que pode ter dois destinos:

1. Dashboards de visualização
2. Pipelines de Machine Learning

- ✓ A área de DataOps nasceu como um esforço de automatização, maior celeridade, segurança, reprodutibilidade e governança dos processos de data analytics e foi extendido pelo MLOps.

Na próxima aula

01. Prática.

02. Extração de dados.

03. Educação.

04. ENADE.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 2. Extração de dados - Prática - ENADE - INEP - Ministério da Educação

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 2.1. Extração automatizada dos dados

Prof. Dr. Neylson Crepalde

Nesta aula

- Extração de dados – ENADE.

Na próxima aula

01.

Transformação de dados – Parte1

Fundamentos

Bootcamp Engenharia de Dados

Aula 2.2.1. Transformação de dados (Parte I)

Prof. Dr. Neylson Crepalde

Nesta aula

- Transformação de dados – ENADE – Parte 1.

Na próxima aula

01.

Transformação de dados – Parte 2

Fundamentos

Bootcamp Engenharia de Dados

Aula 2.2.2. Transformação de dados (Parte II)

Prof. Dr. Neylson Crepalde

Nesta aula

- Transformação de dados – ENADE – Parte 2.

Na próxima aula

01.

Extração de dados – Twitter.

02.

Configurar uma conta de Dev.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 3. Extração de dados - Prática - Twitter API

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 3.1. Configurando uma conta de DEV no Twitter

Prof. Dr. Neylson Crepalde

Nesta aula

- Configurando uma conta de Dev no Twitter.

Na próxima aula

01.

Criando um app.

02.

Adquirindo as chaves de acesso.

Fundamentos

Bootcamp Engenharia de Dados

Aula 3.2. Criando um app e pegando as chaves de acesso

Prof. Dr. Neylson Crepalde

Nesta aula

- Criar um app no ambiente de Dev.
- Obtendo as chaves de acesso.

Na próxima aula

01.

Construindo um crawler para streaming de tweets.

Fundamentos

Bootcamp Engenharia de Dados

Aula 3.3. Construindo um crawler para fazer streaming de tweets

Prof. Dr. Neylson Crepalde

Nesta aula

- Construindo um crawler para streaming de tweets.

Na próxima aula

01.

Transformação de dados do Twitter.

02.

Formato JSON.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 4. Transformação de dados - Prática - Organização e Tratamento dos dados

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 4.1. Entendendo o formato do tweet - JSON

Prof. Dr. Neylson Crepalde

Nesta aula

- Entendendo o formato JSON.

Na próxima aula

01.

Limpeza e organização de dados do Twitter – Parte 01.

Fundamentos

Bootcamp Engenharia de Dados

Aula 4.2.1. Limpeza e organização dos dados do Twitter (Parte I)

Prof. Dr. Neylson Crepalde

Nesta aula

- Limpeza e organização de dados do Twitter – Parte 1.

Na próxima aula

01.

Limpeza e organização de dados do Twitter – Parte 02.

Fundamentos

Bootcamp Engenharia de Dados

Aula 4.2.2. Limpeza e organização dos dados do Twitter (Parte II)

Prof. Dr. Neylson Crepalde

Nesta aula

- Limpeza e organização de dados do Twitter – Parte 2.

Na próxima aula

01.

Ingestão de dados do Twitter em uma tabela relacional.

Fundamentos

Bootcamp Engenharia de Dados

Aula 4.3. Ingestão de dados do Twitter em tabela relacional

Prof. Dr. Neylson Crepalde

Nesta aula

- Ingestão de dados do Twitter em uma tabela relacional.

Trabalho prático

Extração, Transformação e Análises de Dados do ENEM

Na próxima aula

01. Visão geral de soluções para ETL.

02. Introdução.

03. Soluções “Drag and Drop”.

04. Soluções com código.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 5. Soluções de ETL

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

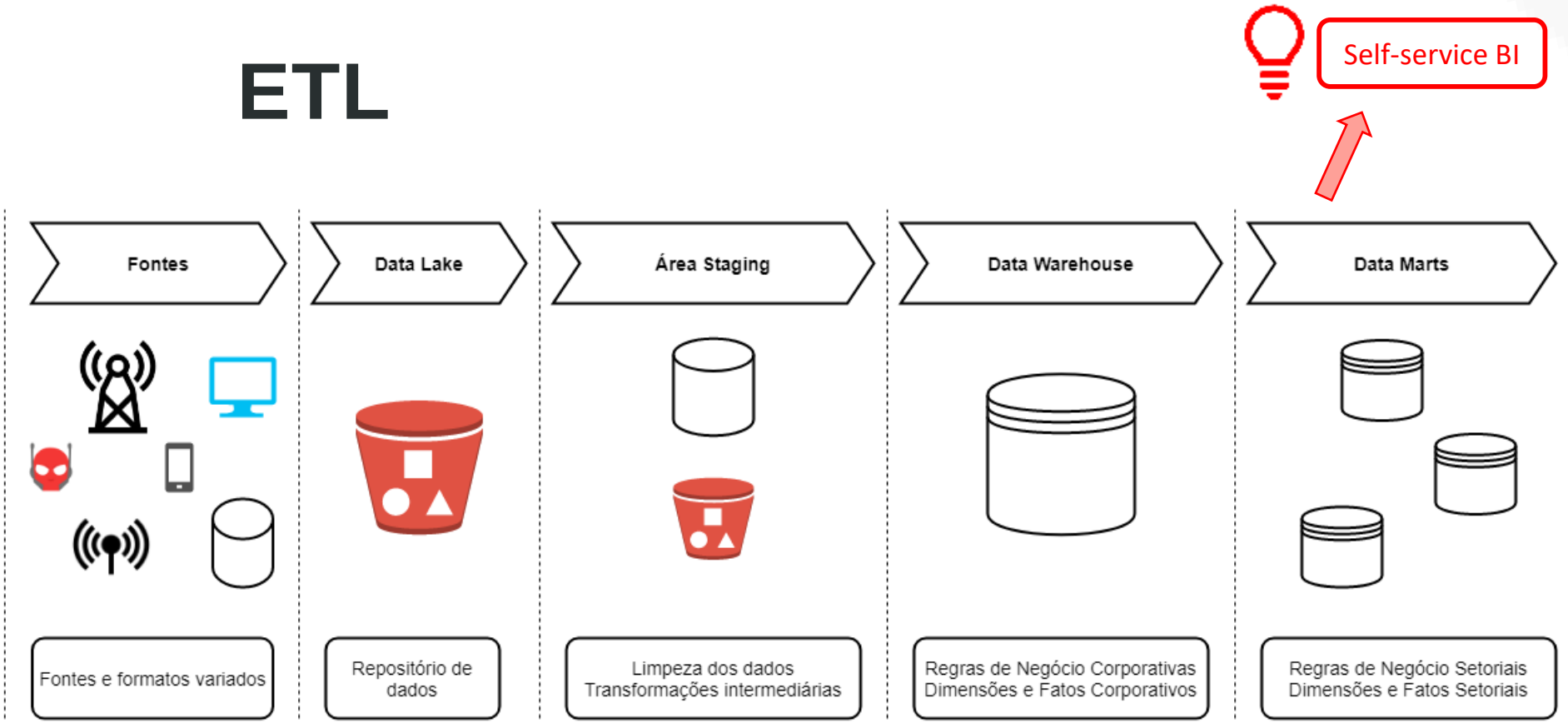
Aula 5.1. Introdução

Prof. Dr. Neylson Crepalde

Nesta aula

- Visão geral sobre ferramentas de ETL.
- Requisitos técnicos.
- Vantagens e desvantagens de cada tipo de ferramenta.

ETL



Adaptado de https://www.igti.com.br/blog/o-que-e-etl-bi/

Do que precisamos?

- Automatização.
- Consistência de execução de Jobs.
- Encadeamento sequencial ou paralelizado de tarefas.
- Possibilidade de conectar a diversas fontes tanto para extração quanto para entrega.
- *Scheduler* (programar execuções).
- Logs de execução.
- *Fail Safe* (possibilidade de recuperação em caso de falha).
- **Notificação** em caso de falha.

Tipos de ferramentas

- “Drag and Drop” – Interface de usuário.
- Soluções com código – Desenvolvimento do pipeline através de framework.


pentaho


APACHE
nifi


IGTI


Open for Innovation
KNIME


talend


Kubeflow


Apache
Airflow

Pachyderm

PREFECT

argo


luigi


APACHE
kafka

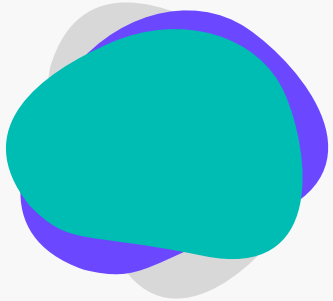
“Drag and Drop”

Facilidade de uso – quase nenhum código.

Podem ser amplamente usadas por parceiros nas áreas de negócio.

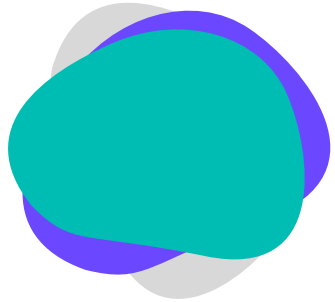
“Monte” o seu pipeline.

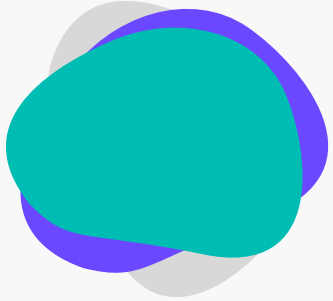
Soluções com código



Mais maleáveis e customizáveis.

São implantadas e escalam com maior facilidade.





São extensíveis.

Conclusão

- ✓ Boas ferramentas de ETL precisam nos dar a capacidade de automatizar o fluxo de dados, executá-los de maneira programada, consistente, segura contra falhas, agnóstica e com governança.

- ✓ Existem ferramentas do tipo “Drag and Drop” e do tipo “com código”. Ambas possuem vantagens e desvantagens dependendo do tipo de projeto em questão.

Na próxima aula

01.

Soluções Drag and Drop.

02.

Pentaho.

Fundamentos

Bootcamp Engenharia de Dados

Aula 5.2. Pentaho

Prof. Dr. Neylson Crepalde

Nesta aula

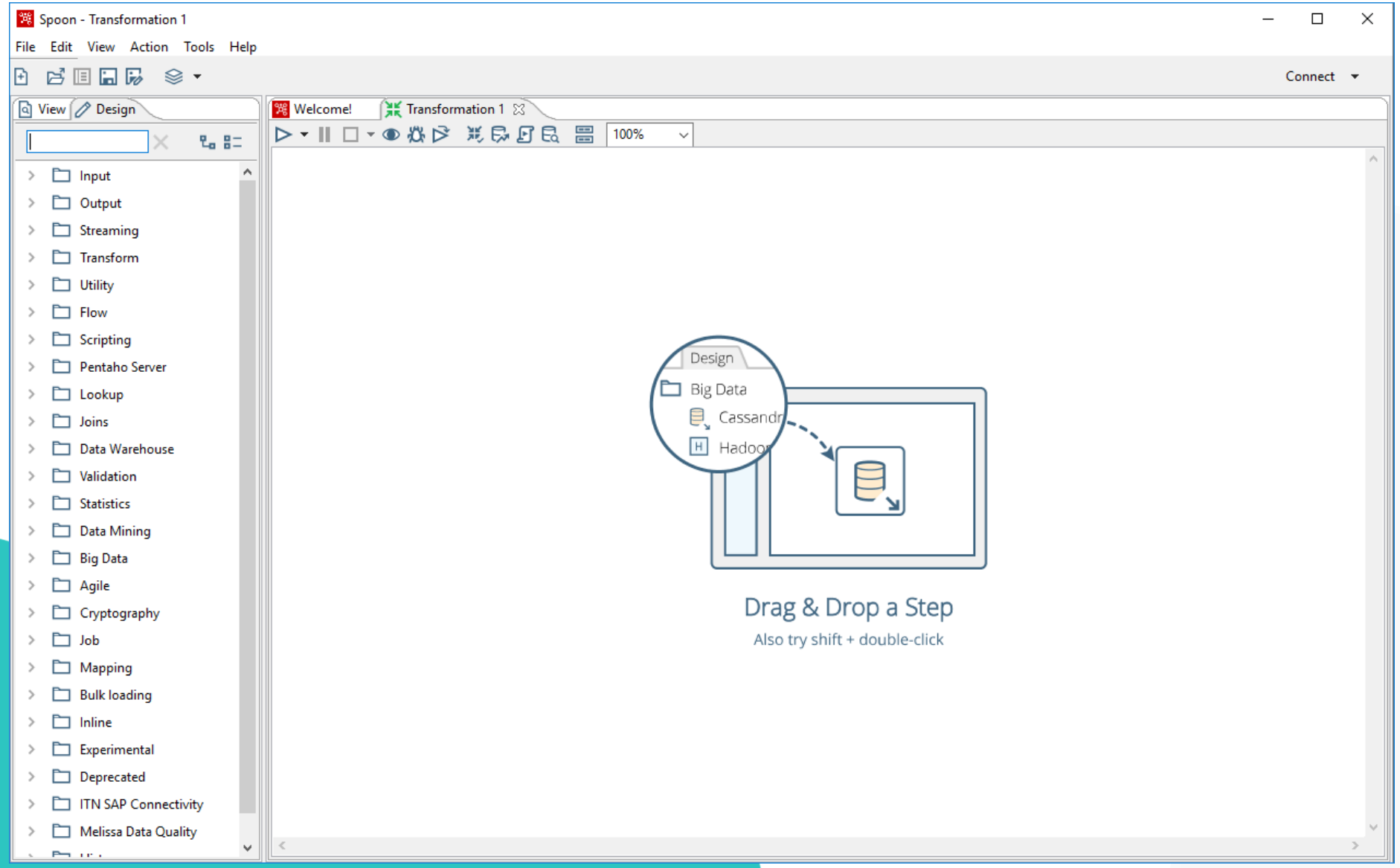
- Pentaho.


pentaho®

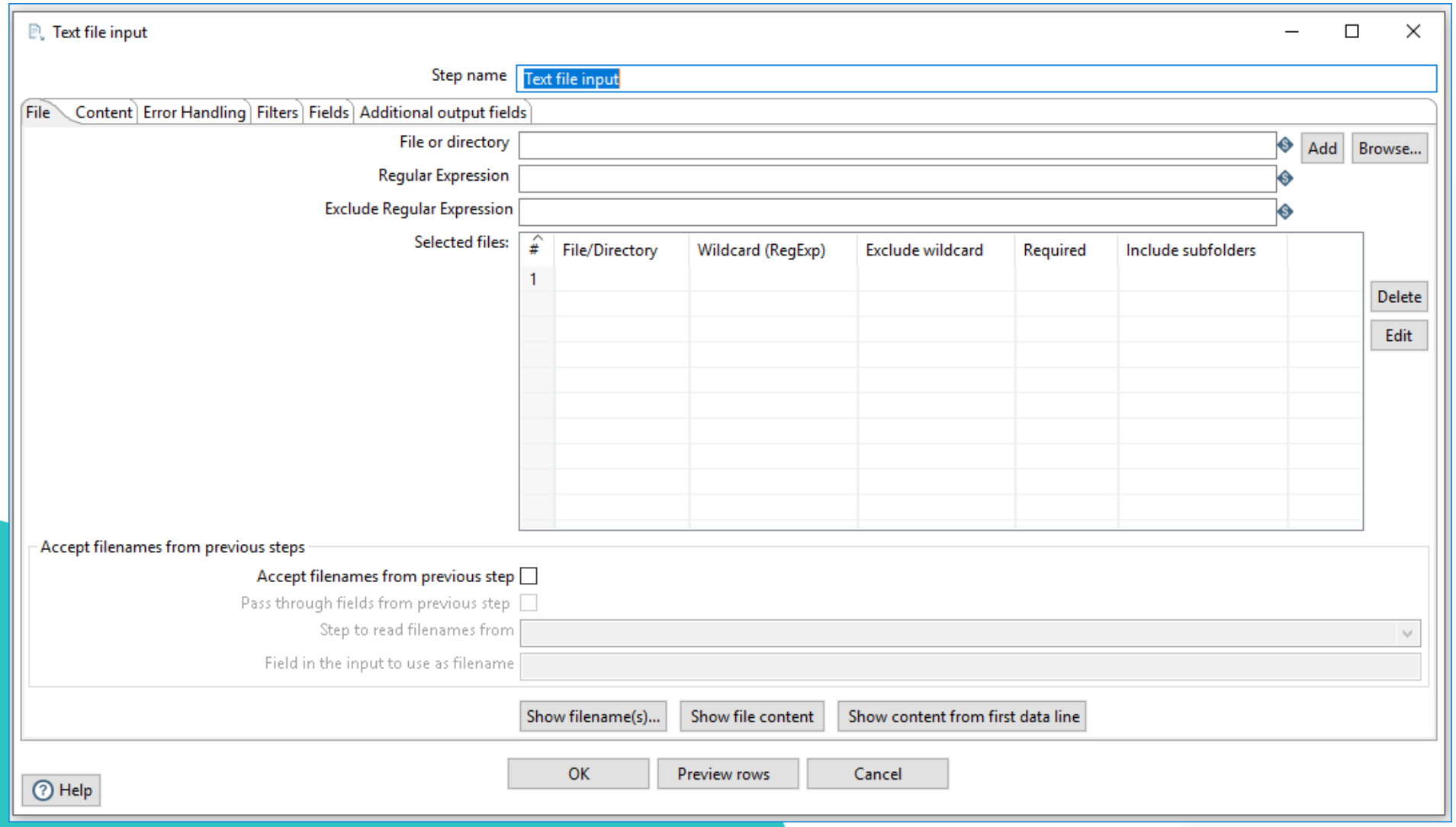
Pentaho Data Integration

- Ferramenta para integração de dados.
- Possui interface de usuário amigável e fácil de usar.
- Conectores com diversas fontes de dados.

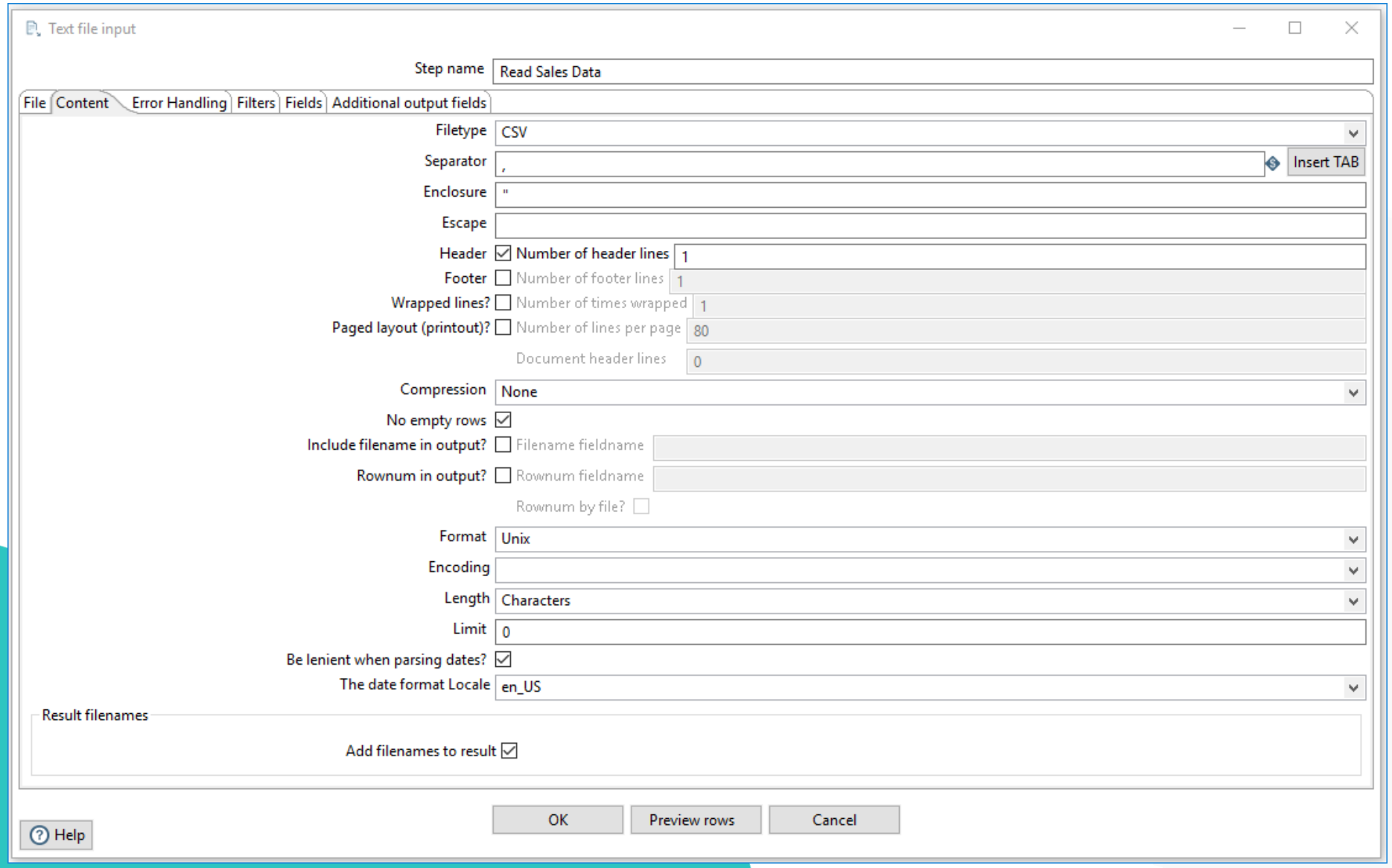
Tela inicial



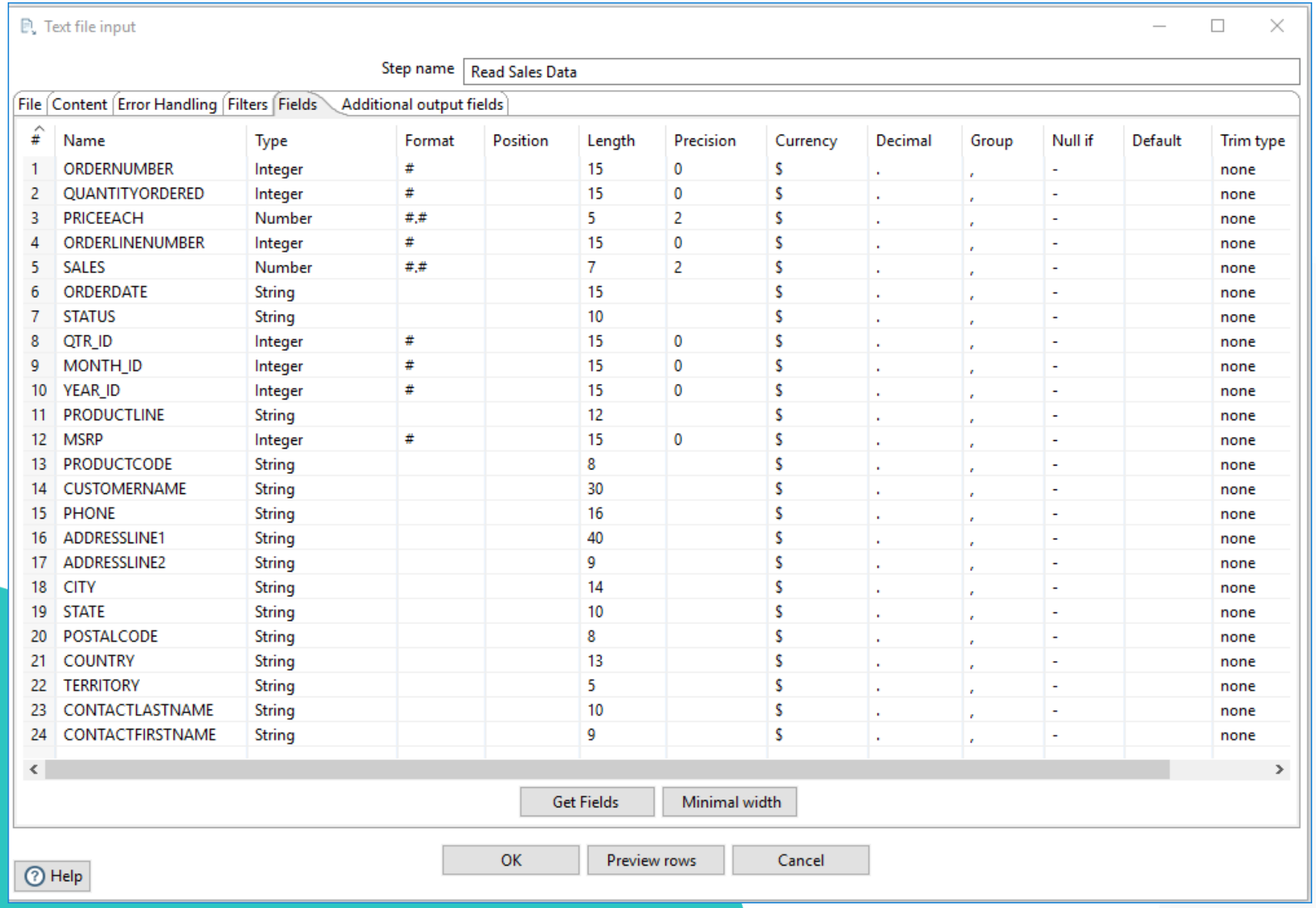
Input de arquivo de texto



Input de arquivo de texto

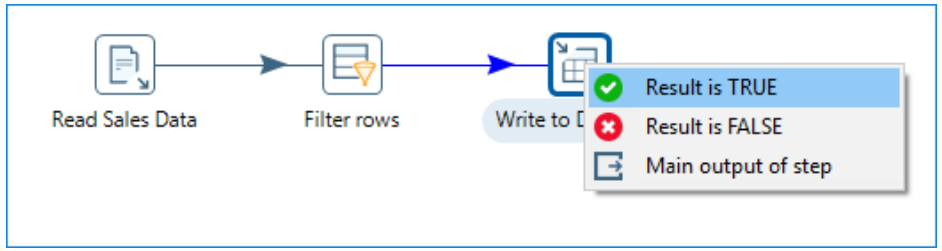


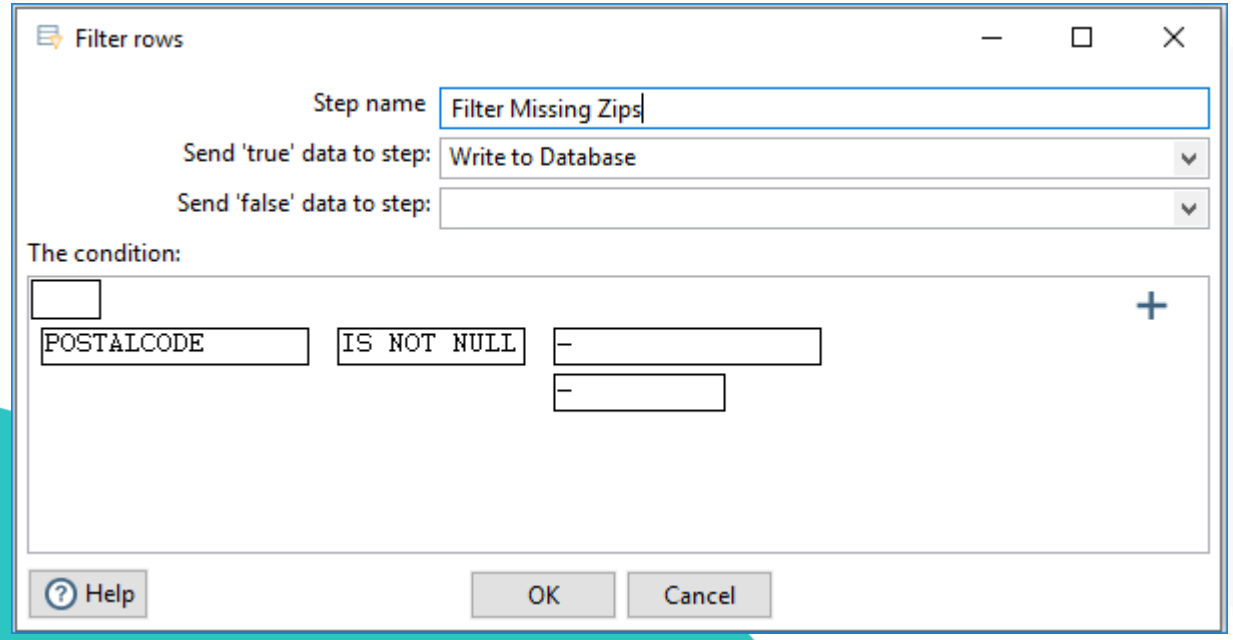
Input de arquivo de texto



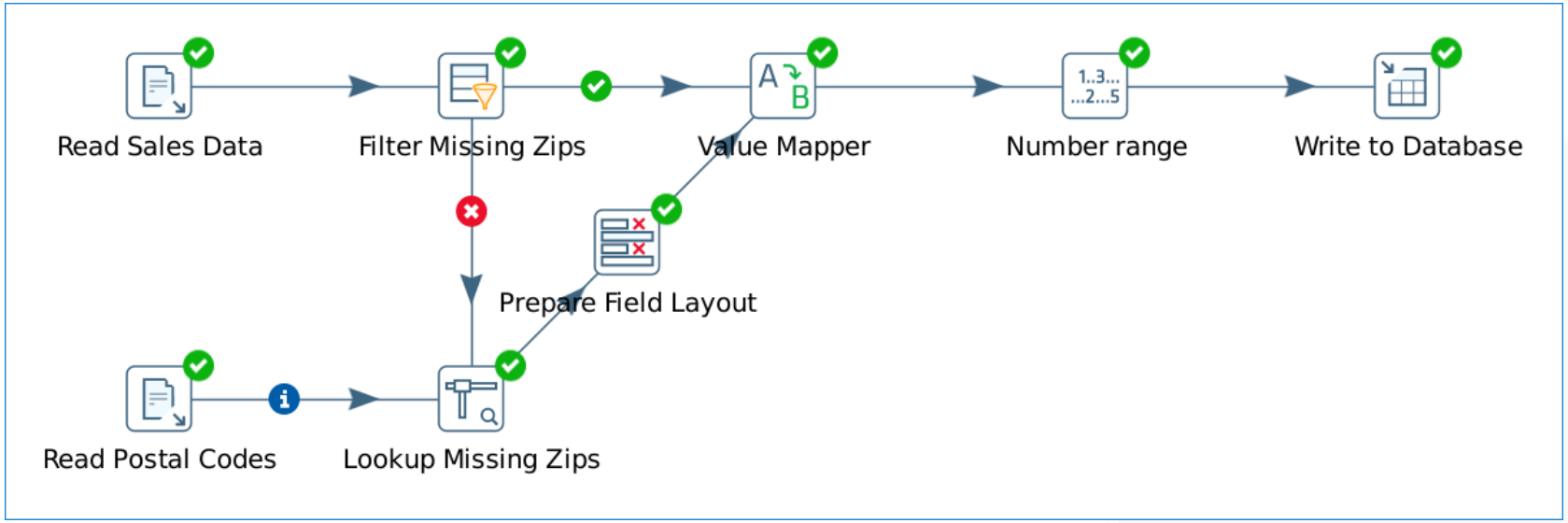
#	Name	Type	Format	Position	Length	Precision	Currency	Decimal	Group	Null if	Default	Trim type
1	ORDERNUMBER	Integer	#		15	0	$	.	,	-		none
2	QUANTITYORDERED	Integer	#		15	0	$	.	,	-		none
3	PRICEEACH	Number	#.#		5	2	$	.	,	-		none
4	ORDERLINENUMBER	Integer	#		15	0	$	.	,	-		none
5	SALES	Number	#.#		7	2	$	.	,	-		none
6	ORDERDATE	String			15		$	.	,	-		none
7	STATUS	String			10		$	.	,	-		none
8	QTR_ID	Integer	#		15	0	$	.	,	-		none
9	MONTH_ID	Integer	#		15	0	$	.	,	-		none
10	YEAR_ID	Integer	#		15	0	$	.	,	-		none
11	PRODUCTLINE	String			12		$	.	,	-		none
12	MSRP	Integer	#		15	0	$	.	,	-		none
13	PRODUCTCODE	String			8		$	.	,	-		none
14	CUSTOMERNAME	String			30		$	.	,	-		none
15	PHONE	String			16		$	.	,	-		none
16	ADDRESSLINE1	String			40		$	.	,	-		none
17	ADDRESSLINE2	String			9		$	.	,	-		none
18	CITY	String			14		$	.	,	-		none
19	STATE	String			10		$	.	,	-		none
20	POSTALCODE	String			8		$	.	,	-		none
21	COUNTRY	String			13		$	.	,	-		none
22	TERRITORY	String			5		$	.	,	-		none
23	CONTACTLASTNAME	String			10		$	.	,	-		none
24	CONTACTFIRSTNAME	String			9		$	.	,	-		none

Transformações

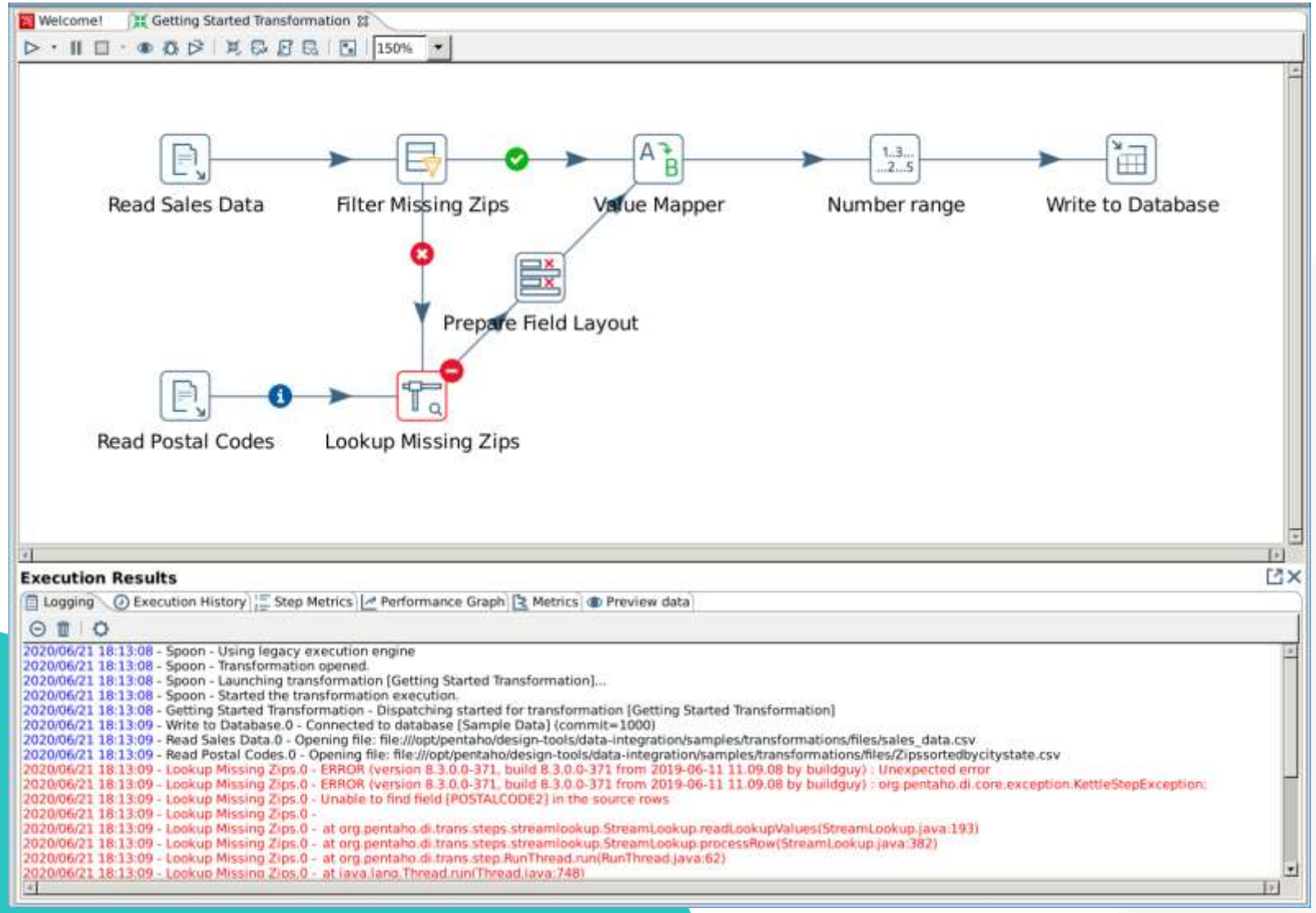




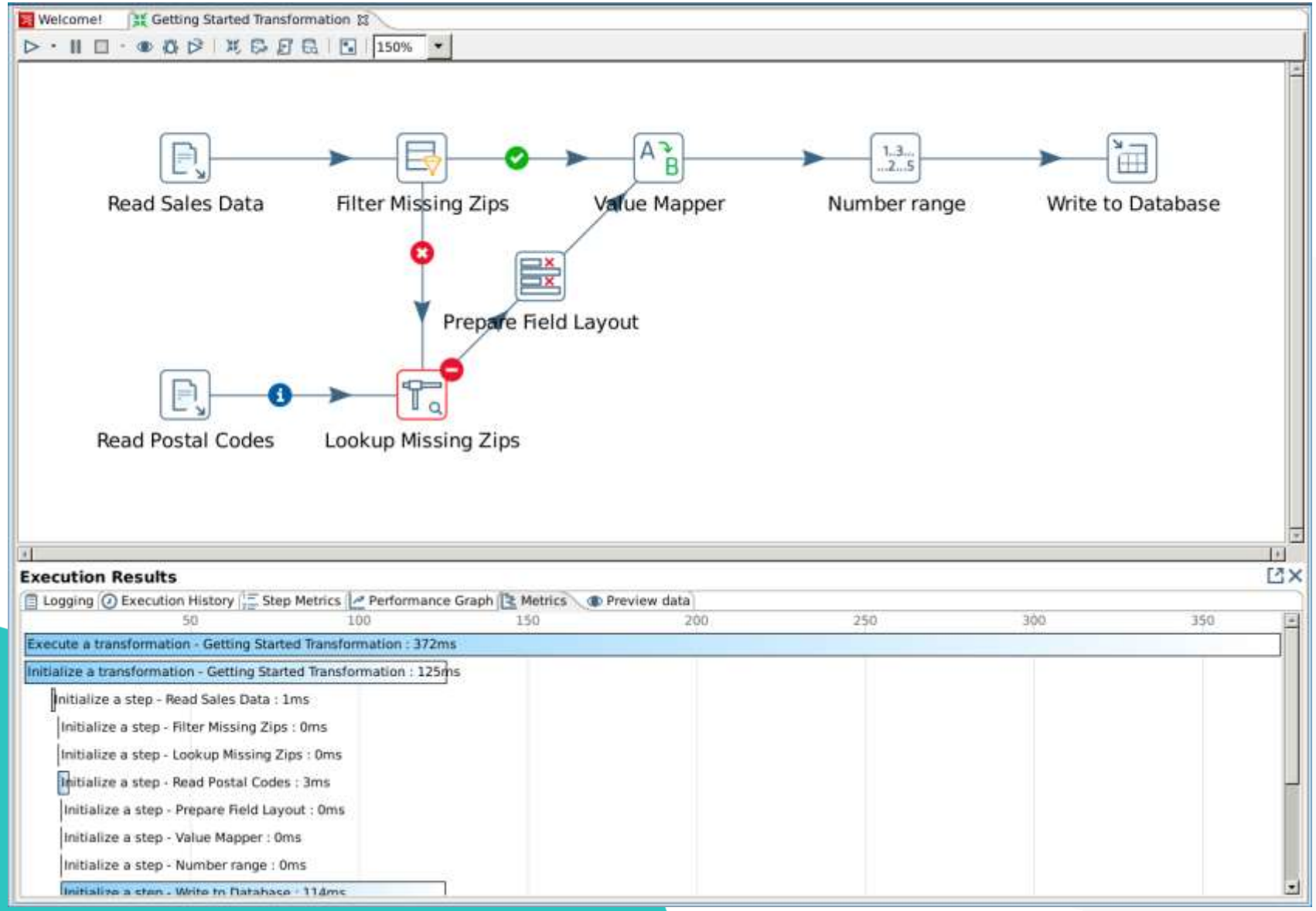
Transformações



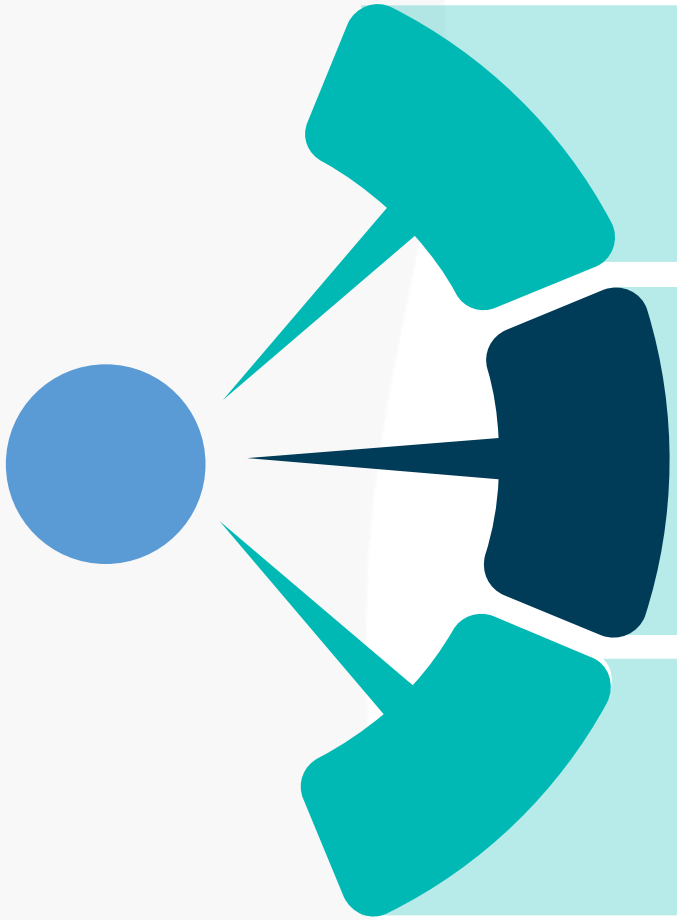
Logs


Welcome!
Getting Started Transformation
150%
Read Sales Data
Filter Missing Zips
Value Mapper
Number range
Write to Database
Prepare Field Layout
Read Postal Codes
Lookup Missing Zips
Execution Results
Logging
Execution History
Step Metrics
Performance Graph
Metrics
Preview data
2020/06/21 18:13:08 - Spoon - Using legacy execution engine
2020/06/21 18:13:08 - Spoon - Transformation opened.
2020/06/21 18:13:08 - Spoon - Launching transformation [Getting Started Transformation]...
2020/06/21 18:13:08 - Spoon - Started the transformation execution.
2020/06/21 18:13:08 - Getting Started Transformation - Dispatching started for transformation [Getting Started Transformation]
2020/06/21 18:13:09 - Write to Database.0 - Connected to database [Sample Data] (commit=1000)
2020/06/21 18:13:09 - Read Sales Data.0 - Opening file: file:///opt/pentaho/design-tools/data-integration/samples/transformations/files/sales_data.csv
2020/06/21 18:13:09 - Read Postal Codes.0 - Opening file: file:///opt/pentaho/design-tools/data-integration/samples/transformations/files/Zipssortedbycitystate.csv
2020/06/21 18:13:09 - Lookup Missing Zips.0 - ERROR (version 8.3.0.0-371, build 8.3.0.0-371 from 2019-06-11 11.09.08 by buildguy) : Unexpected error
2020/06/21 18:13:09 - Lookup Missing Zips.0 - ERROR (version 8.3.0.0-371, build 8.3.0.0-371 from 2019-06-11 11.09.08 by buildguy) : org.pentaho.di.core.exception.KettleStepException:
2020/06/21 18:13:09 - Lookup Missing Zips.0 - Unable to find field [POSTALCODE2] in the source rows
2020/06/21 18:13:09 - Lookup Missing Zips.0 -
2020/06/21 18:13:09 - Lookup Missing Zips.0 - at org.pentaho.di.trans.steps.streamlookup.StreamLookup.readLookupValues(StreamLookup.java:193)
2020/06/21 18:13:09 - Lookup Missing Zips.0 - at org.pentaho.di.trans.steps.streamlookup.StreamLookup.processRow(StreamLookup.java:382)
2020/06/21 18:13:09 - Lookup Missing Zips.0 - at org.pentaho.di.trans.step.RunThread.run(RunThread.java:62)
2020/06/21 18:13:09 - Lookup Missing Zips.0 - at java.lang.Thread.run(Thread.java:748)

Métricas


Welcome!
Getting Started Transformation
150%
Read Sales Data
Filter Missing Zips
Value Mapper
Number range
Write to Database
Prepare Field Layout
Read Postal Codes
Lookup Missing Zips
Execution Results
Logging
Execution History
Step Metrics
Performance Graph
Metrics
Preview data
50
100
150
200
250
300
350
Execute a transformation - Getting Started Transformation : 372ms
Initialize a transformation - Getting Started Transformation : 125ms
Initialize a step - Read Sales Data : 1ms
Initialize a step - Filter Missing Zips : 0ms
Initialize a step - Lookup Missing Zips : 0ms
Initialize a step - Read Postal Codes : 3ms
Initialize a step - Prepare Field Layout : 0ms
Initialize a step - Value Mapper : 0ms
Initialize a step - Number range : 0ms

Vantagens



Fácil uso – Interface de usuário simples e intuitiva.

Vasta gama de conectores à disposição.

Dispõe de logs, métricas de execução e *scheduler.*

Conclusão

- ✓ O Pentaho pode ser uma ótima ferramenta de ETL a ser utilizada por equipes com pouca experiência com programação ou que se encontram dentro de alguma área de negócio. Sua interface é amigável e intuitiva e ele possui excelente controle de logs e métricas de execução.

Na próxima aula

01.

Apache Nifi.

Fundamentos

Bootcamp Engenharia de Dados

Aula 5.3. Apache Nifi

Prof. Dr. Neylson Crepalde

Nesta aula

- Apache Nifi.

IGTI

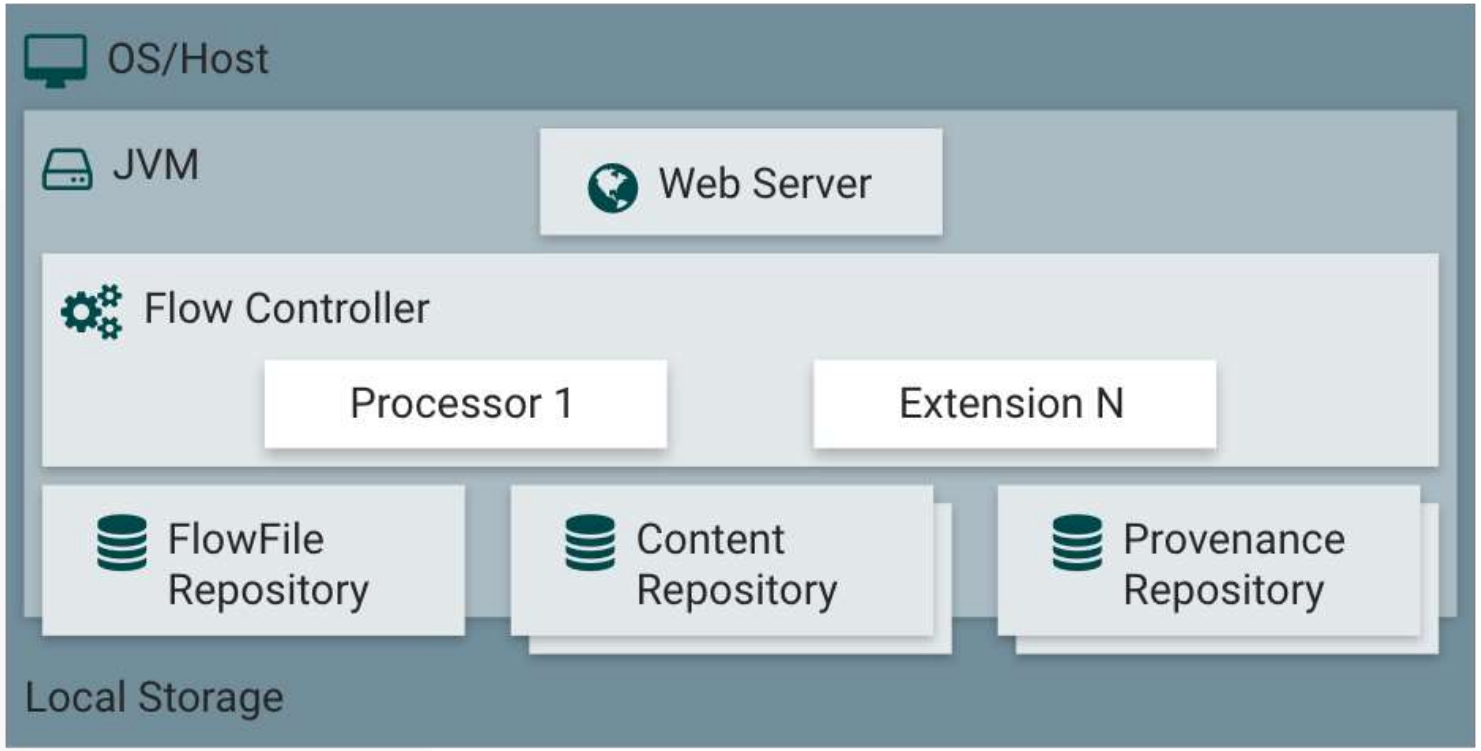
nifi

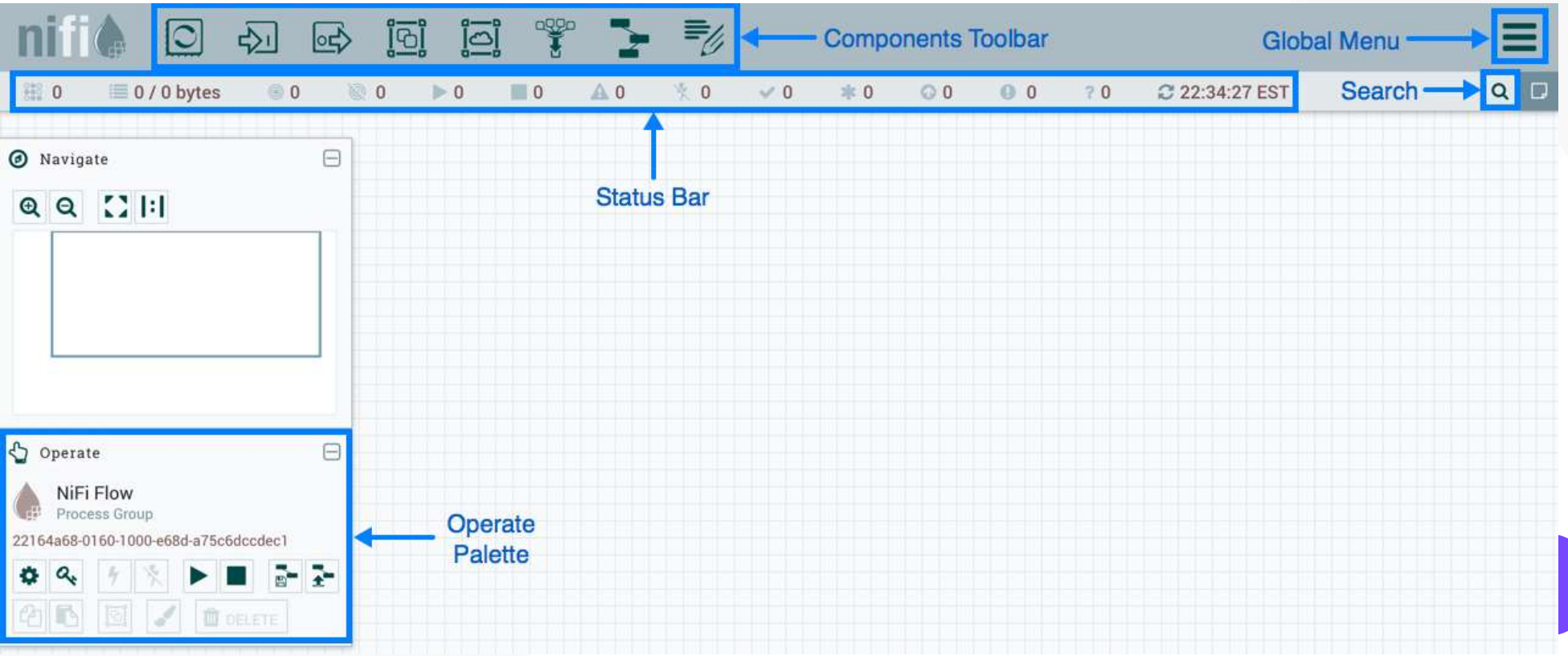
Apache Nifi

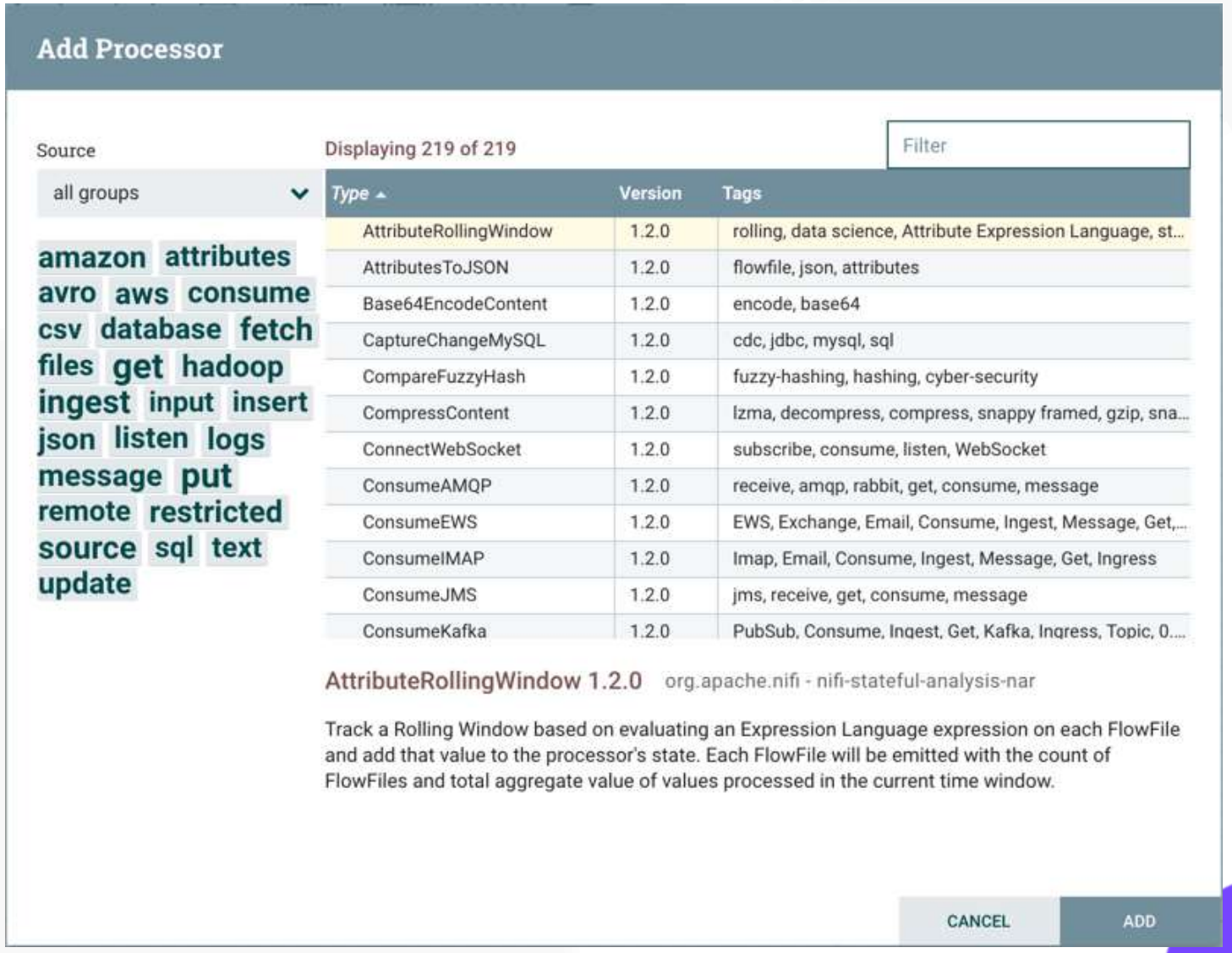
- Ferramenta de automatização de pipelines de dados.

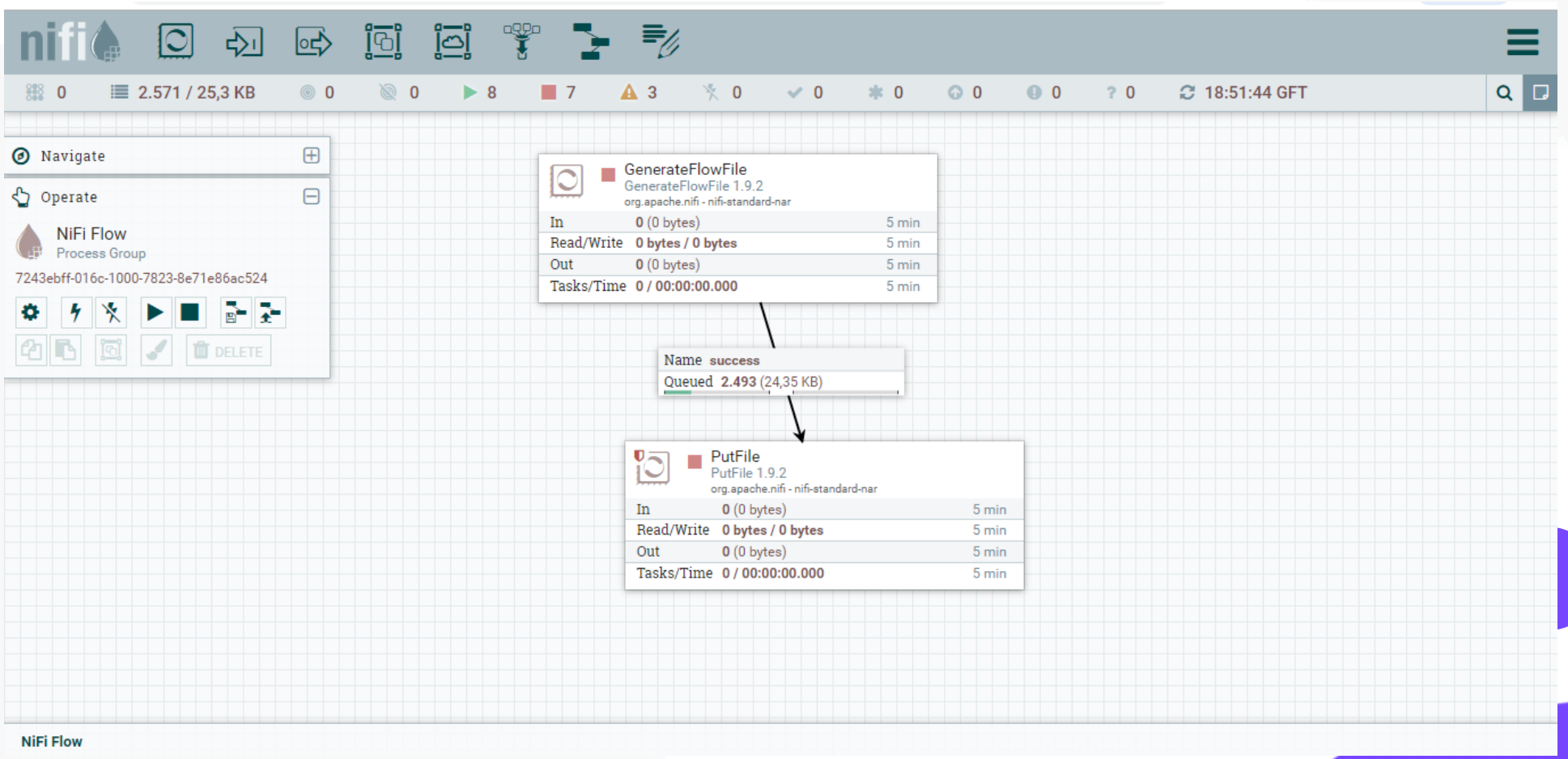
Alguns dos problemas que o Nifi se propõe a atacar:

- Falhas de sistema.
- Limitações insuficientes de dados (dados grandes demais, pequenos demais, rápidos demais, lentos demais...).
- Mudanças de comportamento.
- Compliance e segurança.
- Melhorias contínuas que só acontecem em PRODUÇÃO.


OS/Host
JVM
Web Server
Flow Controller
Processor 1
Extension N
FlowFile Repository
Content Repository
Provenance Repository
Local Storage


Components Toolbar
Global Menu
0
0 / 0 bytes
0
0
0
0
0
0
0
0
0
0
22:34:27 EST
Search
Status Bar
Navigate
Operate
NiFi Flow
Process Group
22164a68-0160-1000-e68d-a75c6dccdec1
DELETE
Operate
Palette

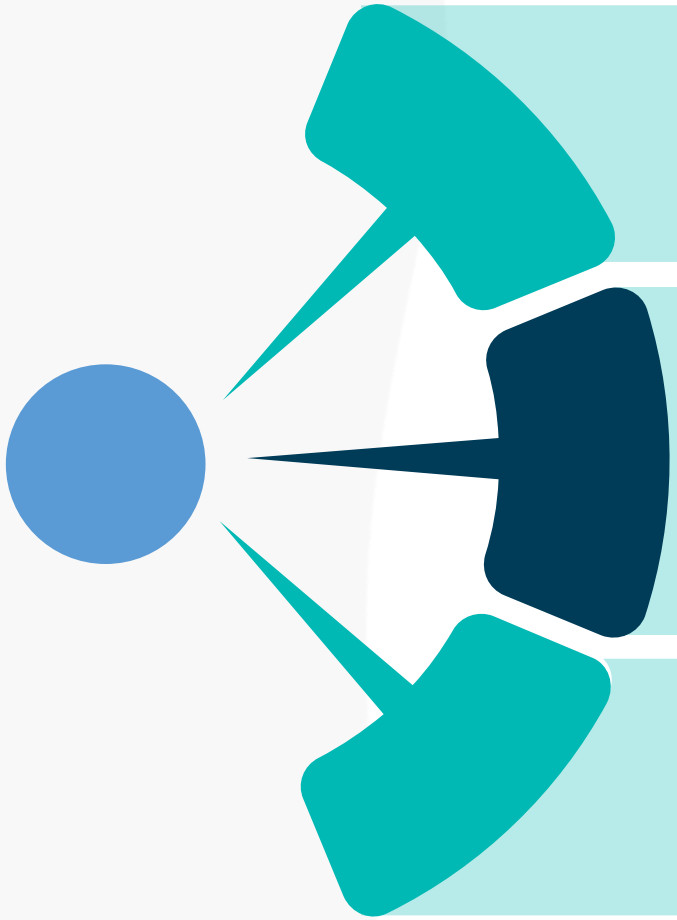

Add Processor
Source
all groups
amazon attributes avro aws consume csv database fetch files get hadoop ingest input insert json listen logs message put remote restricted source sql text update
Displaying 219 of 219
Filter
Type
Version
Tags
AttributeRollingWindow 1.2.0
AttributesToJSON 1.2.0
Base64EncodeContent 1.2.0
CaptureChangeMySQL 1.2.0
CompareFuzzyHash 1.2.0
CompressContent 1.2.0
ConnectWebSocket 1.2.0
ConsumeAMQP 1.2.0
ConsumeEWS 1.2.0
ConsumeIMAP 1.2.0
ConsumeJMS 1.2.0
ConsumeKafka 1.2.0
rolling, data science, Attribute Expression Language, st...
flowfile, json, attributes
encode, base64
cdc, jdbc, mysql, sql
fuzzy-hashing, hashing, cyber-security
lzma, decompress, compress, snappy framed, gzip, sna...
subscribe, consume, listen, WebSocket
receive, amqp, rabbit, get, consume, message
EWS, Exchange, Email, Consume, Ingest, Message, Get,...
Imap, Email, Consume, Ingest, Message, Get, Ingress
jms, receive, get, consume, message
PubSub, Consume, Ingest, Get, Kafka, Ingress, Topic, 0...
AttributeRollingWindow 1.2.0
org.apache.nifi - nifi-stateful-analysis-nar
Track a Rolling Window based on evaluating an Expression Language expression on each FlowFile and add that value to the processor's state. Each FlowFile will be emitted with the count of FlowFiles and total aggregate value of values processed in the current time window.
CANCEL
ADD


nifi
0
2.571 / 25,3 KB
0
0
8
7
3
0
0
0
0
0
0
18:51:44 GFT
Navigate
Operate
NiFi Flow
Process Group
7243ebff-016c-1000-7823-8e71e86ac524
DELETE
GenerateFlowFile
GenerateFlowFile 1.9.2
org.apache.nifi - nifi-standard-nar
In 0 (0 bytes) 5 min
Read/Write 0 bytes / 0 bytes 5 min
Out 0 (0 bytes) 5 min
Tasks/Time 0 / 00:00:00.000 5 min
Name success
Queued 2.493 (24,35 KB)
PutFile
PutFile 1.9.2
org.apache.nifi - nifi-standard-nar
In 0 (0 bytes) 5 min
Read/Write 0 bytes / 0 bytes 5 min
Out 0 (0 bytes) 5 min
Tasks/Time 0 / 00:00:00.000 5 min
NiFi Flow

Diversos processors

- CSV
- JSON
- SQL e NoSQL
- Databases diversas
- AWS
- Microsoft Azure
- Google Cloud
- Kafka
- Protocolos de fila diversos
- XPath
- Spark
- Ecossistema Hadoop

Vantagens



Fácil uso com interface gráfica.

Software Livre.

Segurança da aplicação e consistência de execuções.

Conclusão

- ✓ Embora não seja tão simples de instalar, o Apache Nifi pode ser uma excelente ferramenta de ETL sobretudo pela sua extensa gama de “*processors*” e sua interface gráfica intuitiva e simples de usar.

Na próxima aula

01.

Soluções com código.

02.

Apache AirFlow.

Fundamentos

Bootcamp Engenharia de Dados

Aula 5.4. Apache AirFlow

Prof. Dr. Neylson Crepalde

Nesta aula

- Apache AirFlow.

Apache
Airflow

Definições

O Apache AirFlow é uma plataforma para criar, *schedular* (programar execuções) e monitorar *workflows* – orquestrador.

O projeto AirFlow foi iniciado por Maxime Beauchemin no Airbnb em 2014. Em 2015 sua primeira versão era lançada.

Em 2016 o projeto entrou para o programa incubador da Fundação Apache e em 2019 foi anunciado como um projeto “Top Level”.

Princípios

Escalável – arquitetura modular que utiliza mensageria para orquestrar os workers.

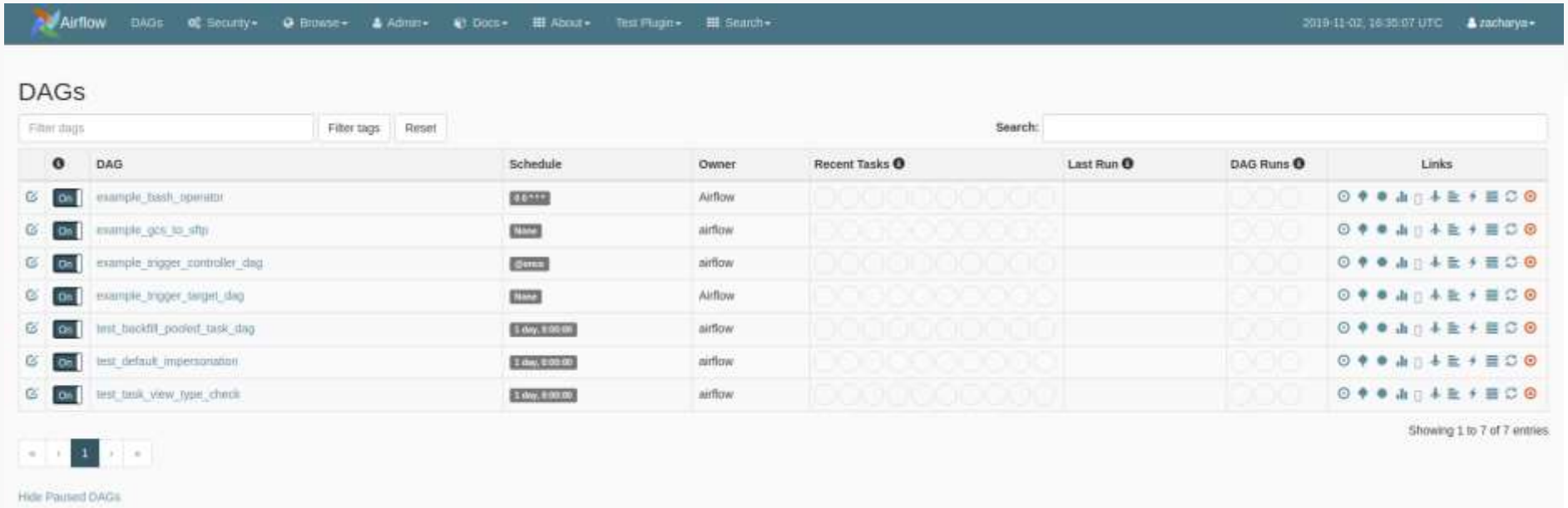
Dinâmico – as pipelines são definidas em Python permitindo geração dinâmica.

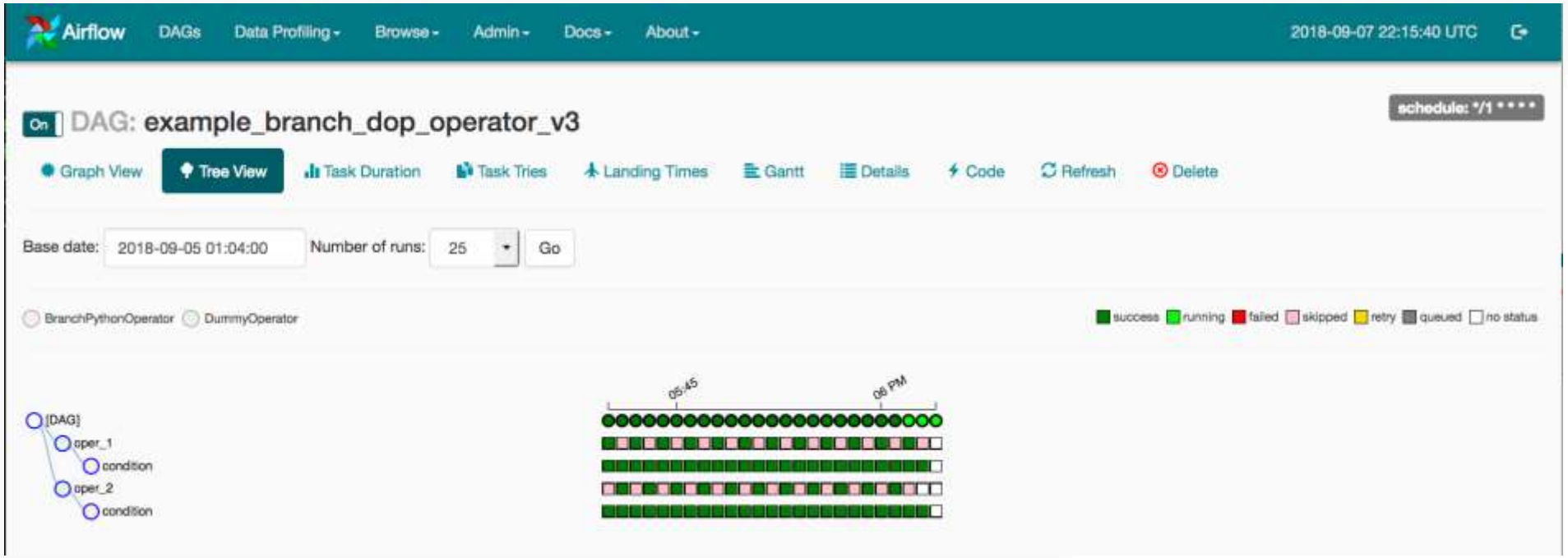
Extensível – permite fácil desenvolvimento de operadores e bibliotecas compatíveis.

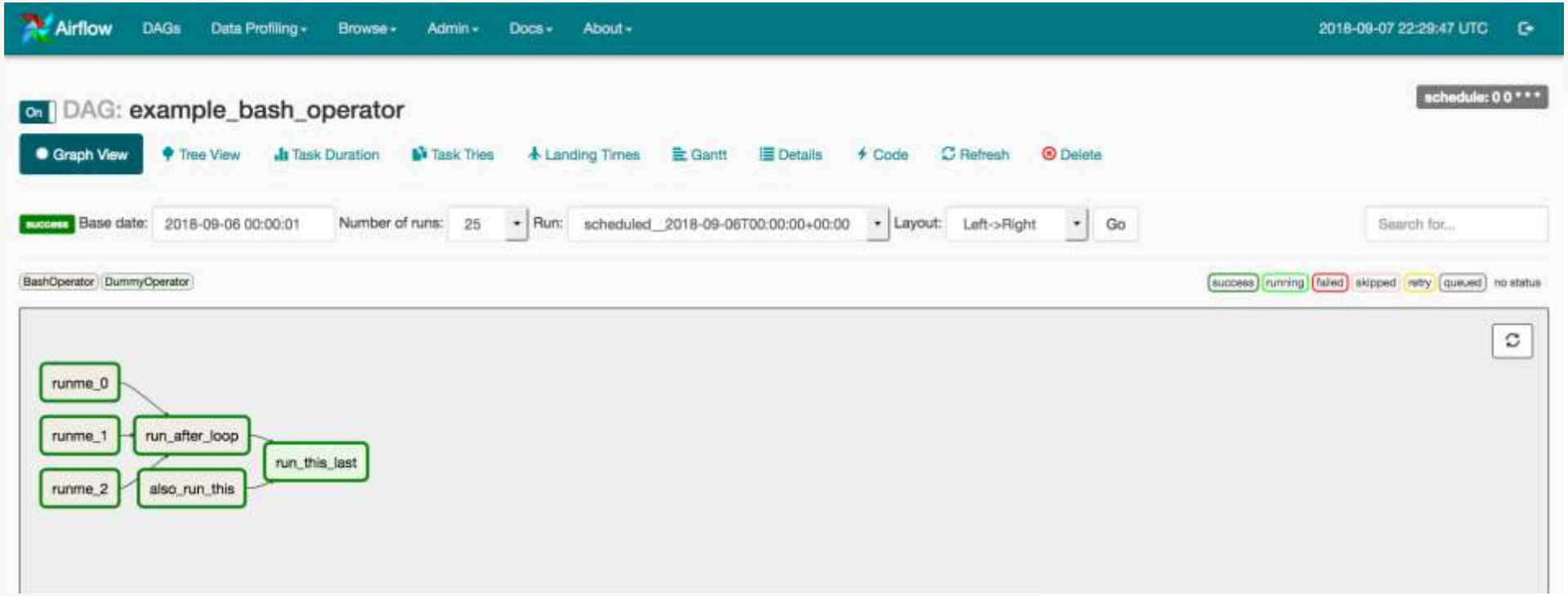
Elegante – pipelines *lean* e explícitas.

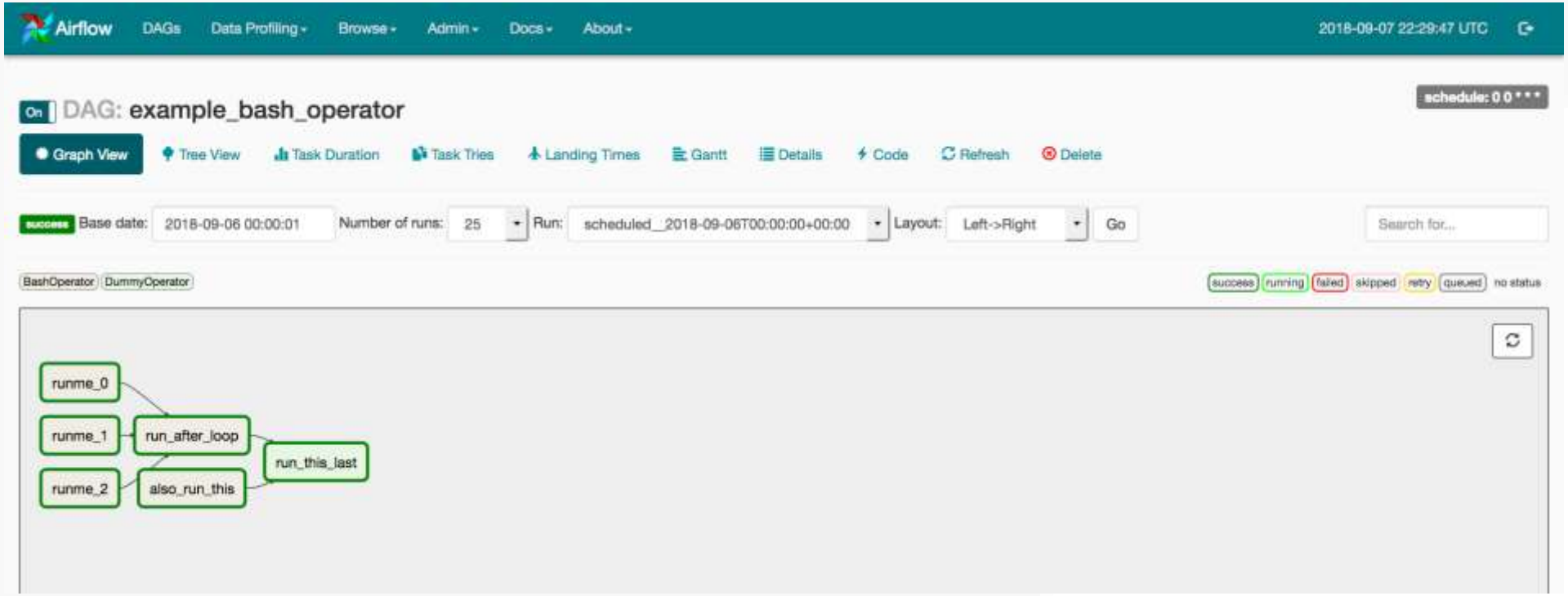
Features

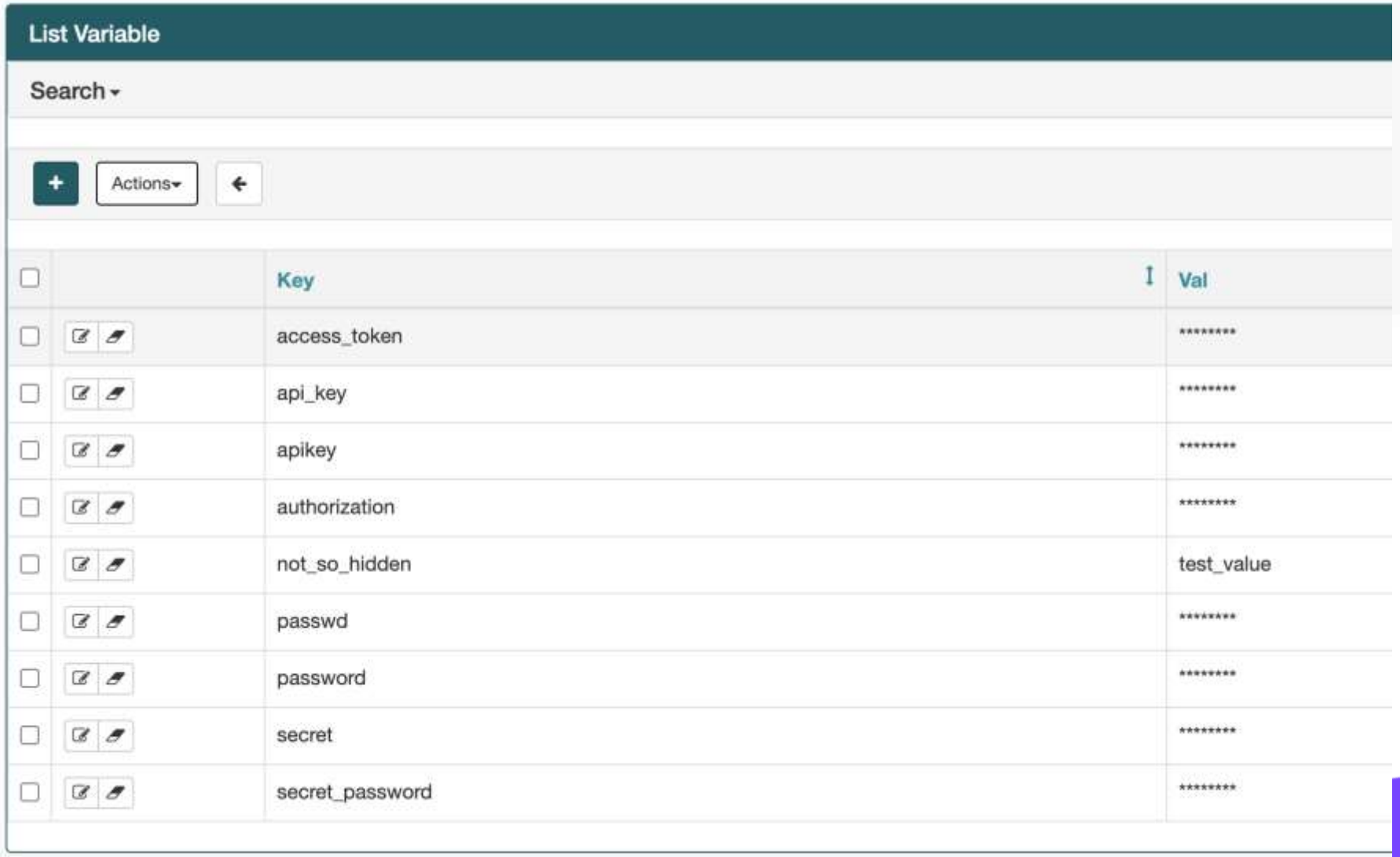
- Puro Python.
- Interface de usuário.
- Integrações robustas.
- Fácil de usar.
- Open Source.

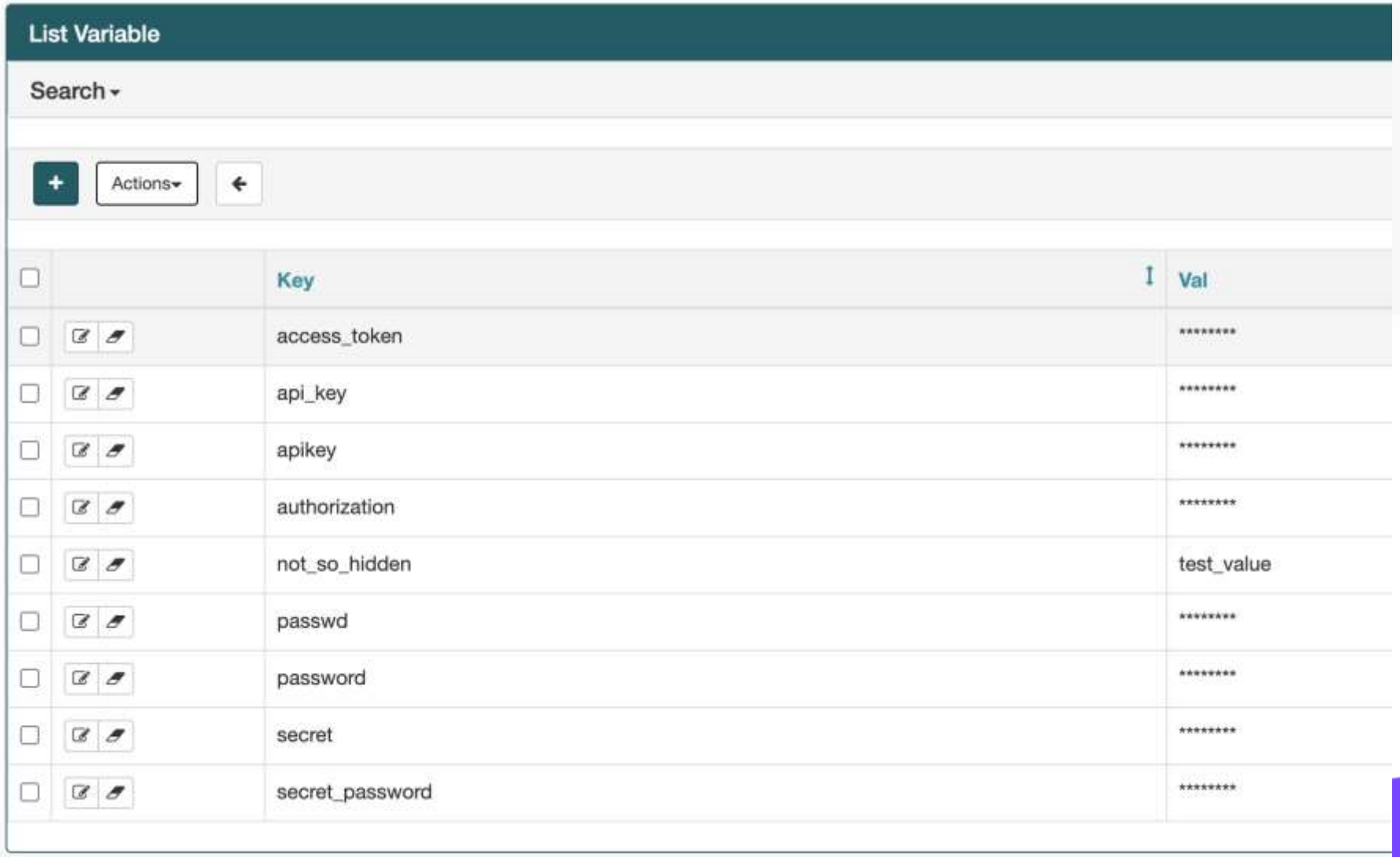

Airflow
DAGs
Security
Browse
Admin
Docs
About
Test Plugin
Search
2019-11-02, 16:35:07 UTC
zacharya
DAGs
Filter dags
Filter tags
Reset
Search:
DAG
Schedule
Owner
Recent Tasks
Last Run
DAG Runs
Links
example_bash_operator
example_gcs_to_sftp
example_trigger_controller_dag
example_trigger_target_dag
test_backfill_pooled_task_dag
test_default_impersonation
test_task_view_type_check
Airflow
airflow
Showing 1 to 7 of 7 entries
Hide Paused DAGs

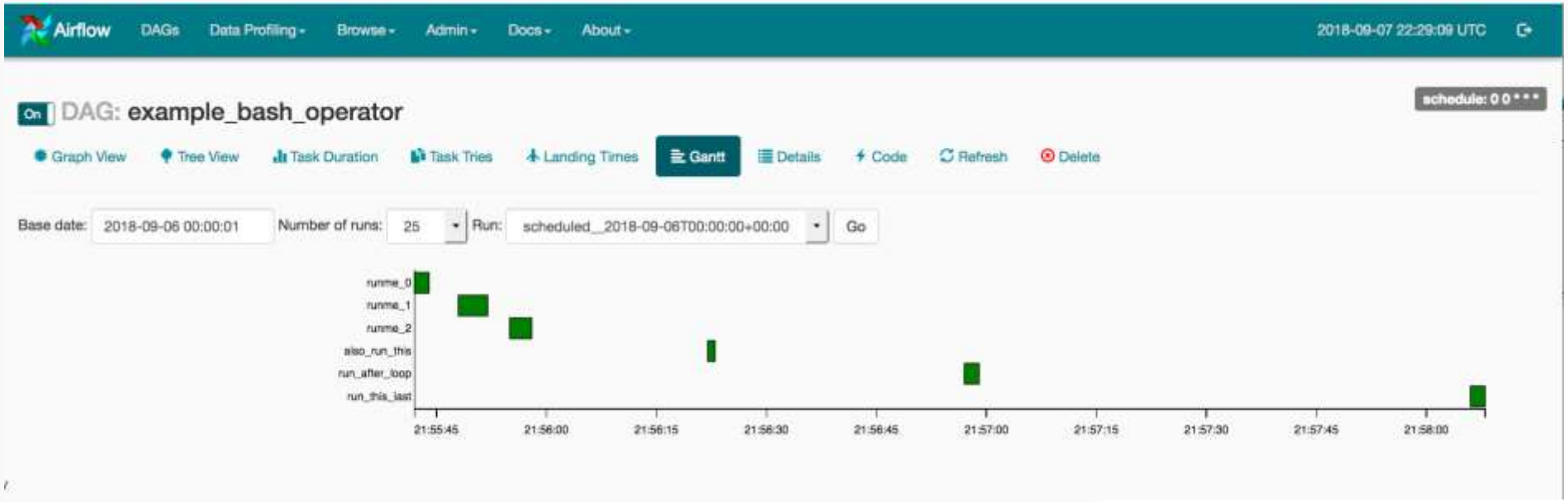

Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
2018-09-07 22:15:40 UTC
On
DAG: example_branch_dop_operator_v3
schedule: */1 * * * *
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
Base date:
2018-09-05 01:04:00
Number of runs:
25
Go
BranchPythonOperator
DummyOperator
success
running
failed
skipped
retry
queued
no status
05:45
06 PM
[DAG]
oper_1
condition
oper_2
condition

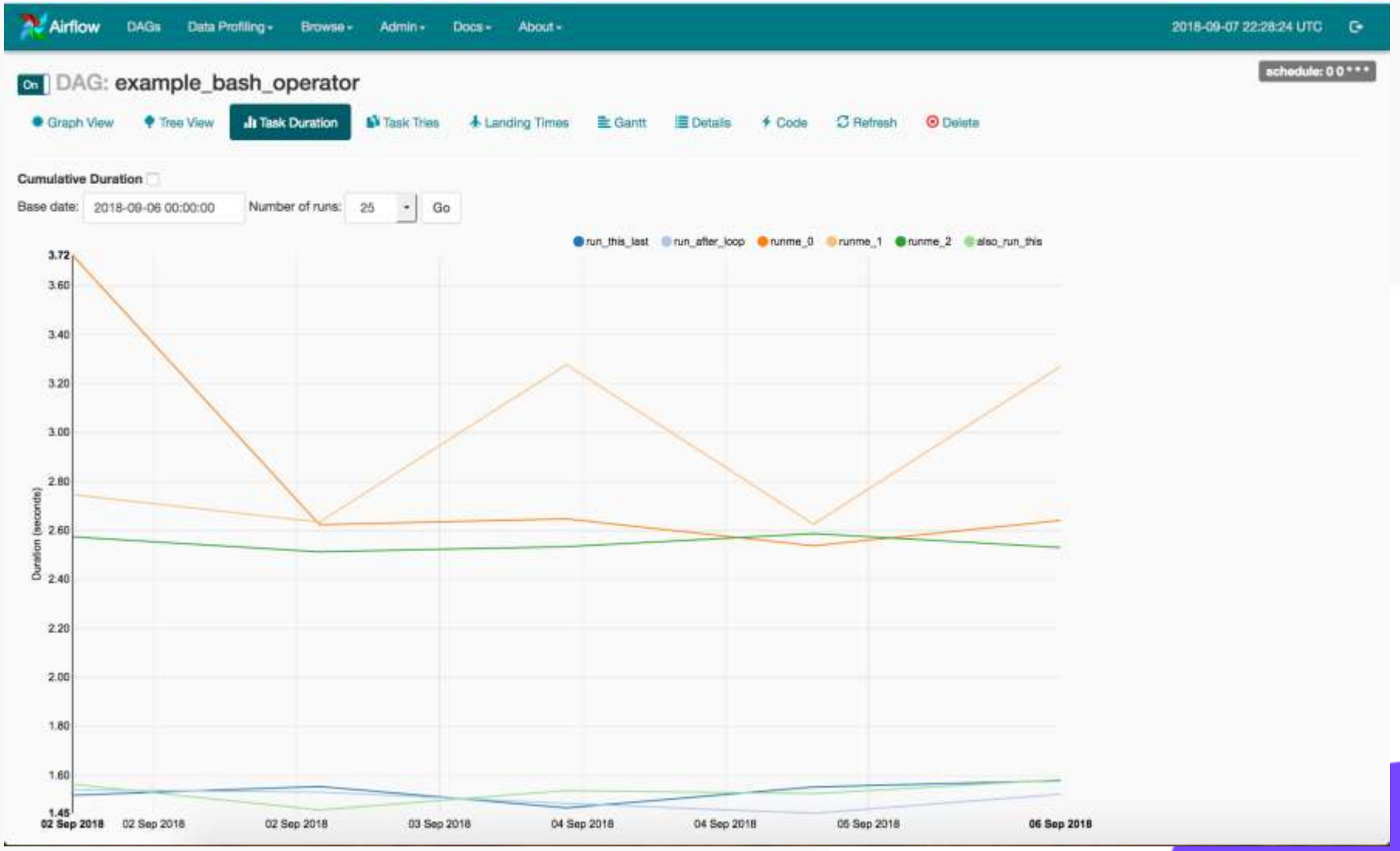

Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
2018-09-07 22:29:47 UTC
On
DAG: example_bash_operator
schedule: 0 0 * * *
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
success
Base date: 2018-09-06 00:00:01
Number of runs: 25
Run: scheduled__2018-09-06T00:00:00+00:00
Layout: Left->Right
Go
Search for...
BashOperator
DummyOperator
success
running
failed
skipped
retry
queued
no status
runme_0
runme_1
runme_2
run_after_loop
also_run_this
run_this_last

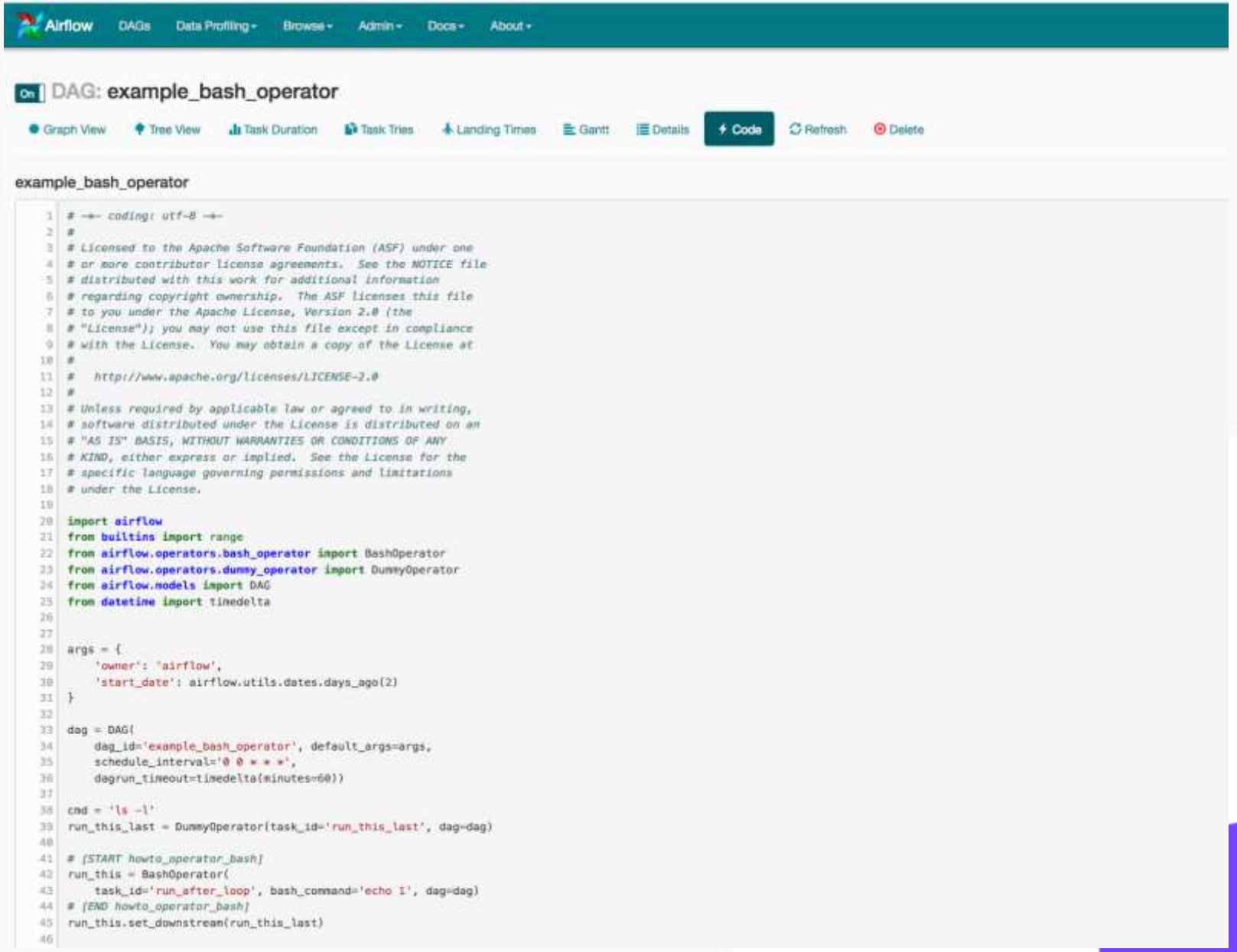

Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
2018-09-07 22:29:47 UTC
On
DAG: example_bash_operator
schedule: 0 0 * * *
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
success
Base date: 2018-09-06 00:00:01
Number of runs: 25
Run: scheduled__2018-09-06T00:00:00+00:00
Layout: Left->Right
Go
Search for...
BashOperator
DummyOperator
success
running
failed
skipped
retry
queued
no status
runme_0
runme_1
runme_2
run_after_loop
also_run_this
run_this_last


List Variable
Search
Actions
Key
Val
access_token
****
api_key
****
apikey
****
authorization
****
not_so_hidden
test_value
passwd
****
password
****
secret
****
secret_password
****


List Variable
Search
Actions
Key
Val
access_token
****
api_key
****
apikey
****
authorization
****
not_so_hidden
test_value
passwd
****
password
****
secret
****
secret_password
****


Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
2018-09-07 22:29:09 UTC
On
DAG: example_bash_operator
schedule: 0 0 * * *
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
Base date: 2018-09-06 00:00:01
Number of runs: 25
Run: scheduled__2018-09-06T00:00:00+00:00
Go
runme_0
runme_1
runme_2
also_run_this
run_after_loop
run_this_last
21:55:45
21:56:00
21:56:15
21:56:30
21:56:45
21:57:00
21:57:15
21:57:30
21:57:45
21:58:00


Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
2018-09-07 22:28:24 UTC
DAG: example_bash_operator
schedule: 0 0 * * *
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
Cumulative Duration
Base date: 2018-09-06 00:00:00
Number of runs: 25
Go
run_this_last
run_after_loop
runme_0
runme_1
runme_2
also_run_this
Duration (seconds)
3.72
3.60
3.40
3.20
3.00
2.80
2.60
2.40
2.20
2.00
1.80
1.60
1.45
02 Sep 2018
02 Sep 2018
02 Sep 2018
03 Sep 2018
04 Sep 2018
04 Sep 2018
05 Sep 2018
06 Sep 2018


Airflow
DAGs
Data Profiling
Browse
Admin
Docs
About
DAG: example_bash_operator
Graph View
Tree View
Task Duration
Task Tries
Landing Times
Gantt
Details
Code
Refresh
Delete
example_bash_operator
-*- coding: utf-8 -*-
#
Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at
#
http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing,
# software distributed under the License is distributed on an
# "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
# KIND, either express or implied. See the License for the
# specific language governing permissions and limitations
# under the License.
import airflow
from builtins import range
from airflow.operators.bash_operator import BashOperator
from airflow.operators.dummy_operator import DummyOperator
from airflow.models import DAG
from datetime import timedelta
args = {
'owner': 'airflow',
'start_date': airflow.utils.dates.days_ago(2)
}
dag = DAG(
dag_id='example_bash_operator', default_args=args,
schedule_interval='0 0 * * *',
dagrun_timeout=timedelta(minutes=60))
cmd = 'ls -l'
run_this_last = DummyOperator(task_id='run_this_last', dag=dag)
# [START howto_operator_bash]
run_this = BashOperator(
task_id='run_after_loop', bash_command='echo 1', dag=dag)
# [END howto_operator_bash]
run_this.set_downstream(run_this_last)

Conclusão

- ✓ AirFlow é uma das ferramentas mais utilizadas pelos times de Engenharia de Dados no mundo profissional para orquestrar os processos de ETL bem como pipelines de BI e Machine Learning.
- ✓ É uma ferramenta fácil de usar, escalável, e apresenta excelentes resultados em ambiente de produção.

Na próxima aula

01.

KubeFlow.

Fundamentos

Bootcamp Engenharia de Dados

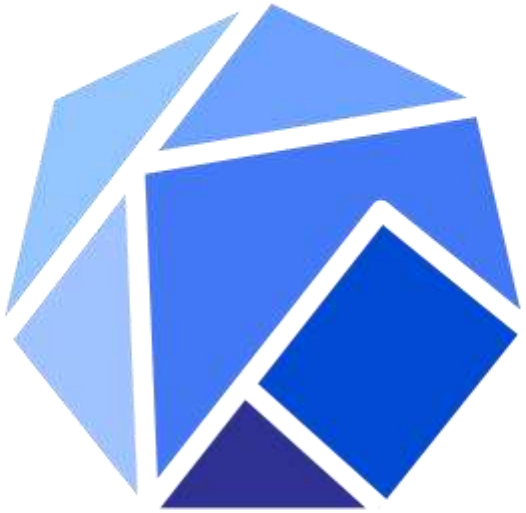
Aula 5.5. KubeFlow

Prof. Dr. Neylson Crepalde

Nesta aula

- KubeFlow.

IGTI



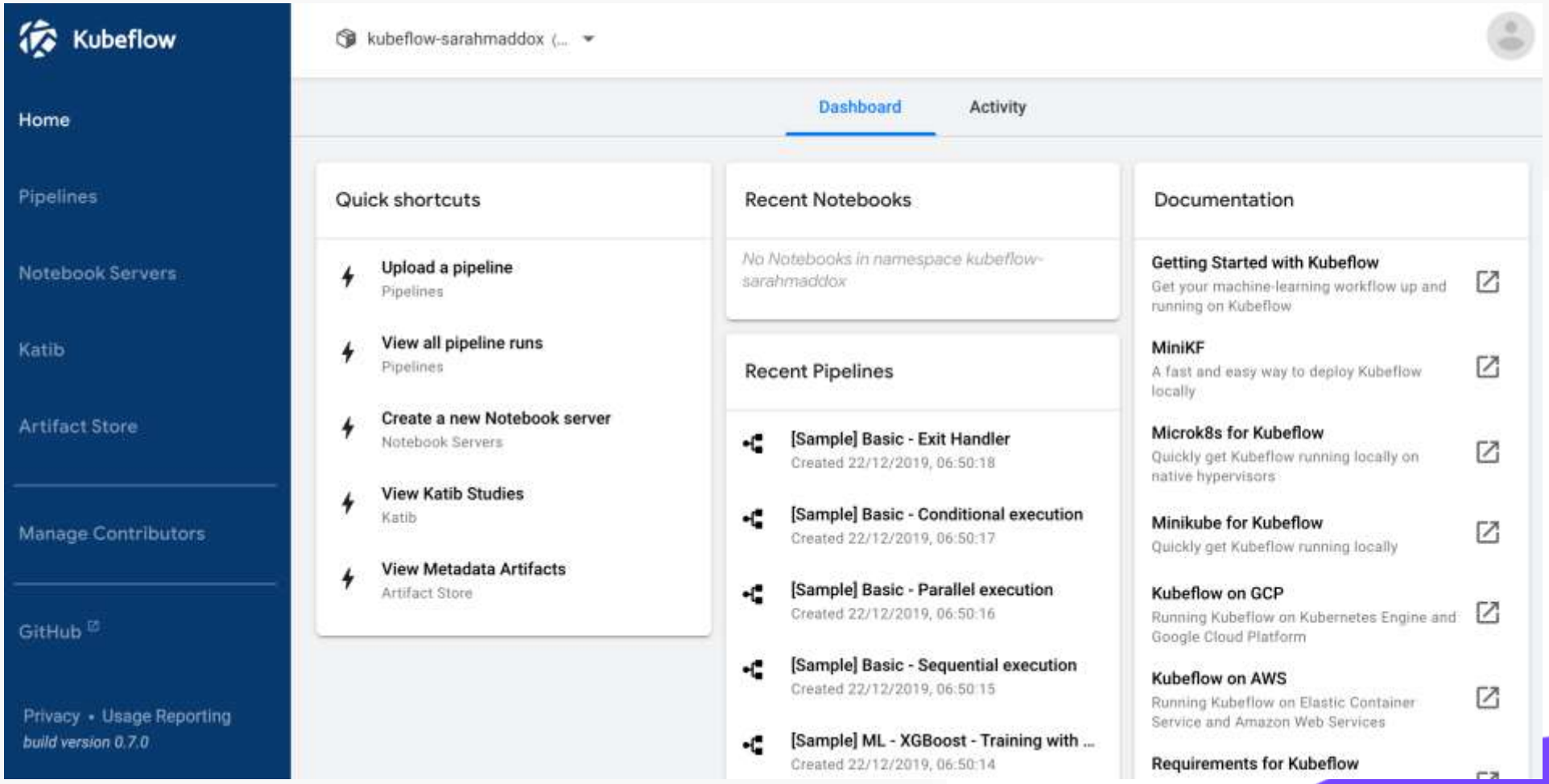
Kubeflow

Definições

- KubeFlow se dedica a Pipelines de Machine Learning escaláveis usando Kubernetes.
- Foi iniciado pelo Google em um projeto open source vinculado à biblioteca *TensorFlow*.
- Oferece grande parte do pipeline de Ciência de Dados.
- Encoraja o desenho de soluções de Ciência de Dados utilizando arquitetura de microsserviços (desacoplada).

Componentes

- Dashboard.
- Metadata (logs e monitoramento de execuções).
- Jupyter Notebooks.
- Fairing (biblioteca para automatização de treino e deploy de ML).
- Feature Store.
- Frameworks.
- Tuning de Hiperparâmetros.
- KF Pipelines.
- KF Serving.


Kubeflow
kubeflow-sarahmaddox (...
Home
Pipelines
Notebook Servers
Katib
Artifact Store
Manage Contributors
GitHub
Privacy • Usage Reporting
build version 0.7.0
Dashboard
Activity
Quick shortcuts
Upload a pipeline
Pipelines
View all pipeline runs
Pipelines
Create a new Notebook server
Notebook Servers
View Katib Studies
Katib
View Metadata Artifacts
Artifact Store
Recent Notebooks
No Notebooks in namespace kubeflow-sarahmaddox
Recent Pipelines
[Sample] Basic - Exit Handler
Created 22/12/2019, 06:50:18
[Sample] Basic - Conditional execution
Created 22/12/2019, 06:50:17
[Sample] Basic - Parallel execution
Created 22/12/2019, 06:50:16
[Sample] Basic - Sequential execution
Created 22/12/2019, 06:50:15
[Sample] ML - XGBoost - Training with ...
Created 22/12/2019, 06:50:14
Documentation
Getting Started with Kubeflow
Get your machine-learning workflow up and running on Kubeflow
MiniKF
A fast and easy way to deploy Kubeflow locally
Microk8s for Kubeflow
Quickly get Kubeflow running locally on native hypervisors
Minikube for Kubeflow
Quickly get Kubeflow running locally
Kubeflow on GCP
Running Kubeflow on Kubernetes Engine and Google Cloud Platform
Kubeflow on AWS
Running Kubeflow on Elastic Container Service and Amazon Web Services
Requirements for Kubeflow

Kubeflow anonymous

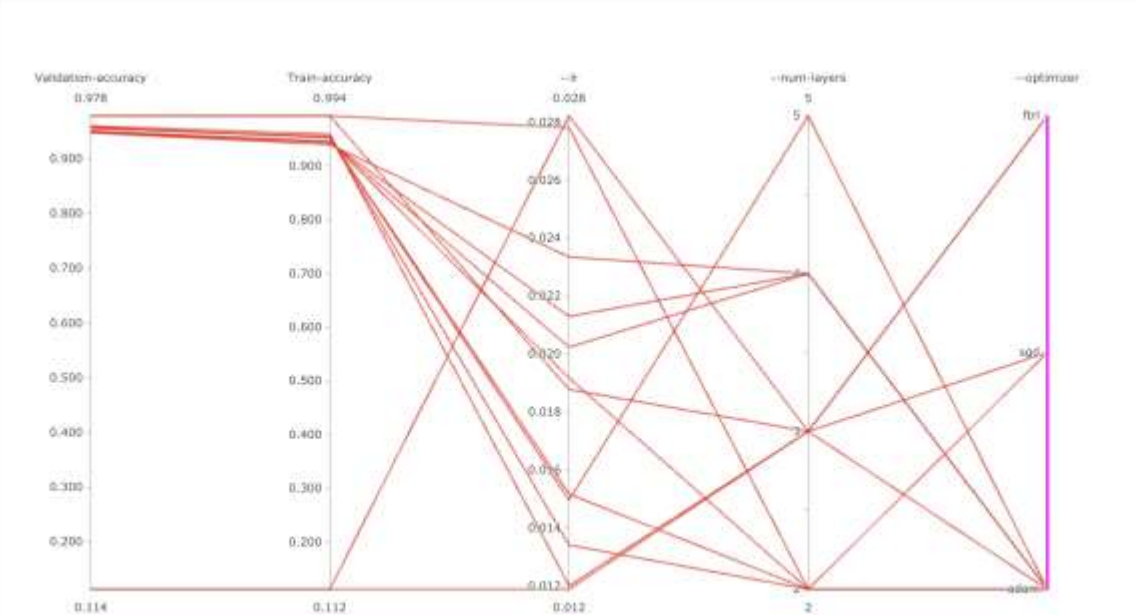
Katib

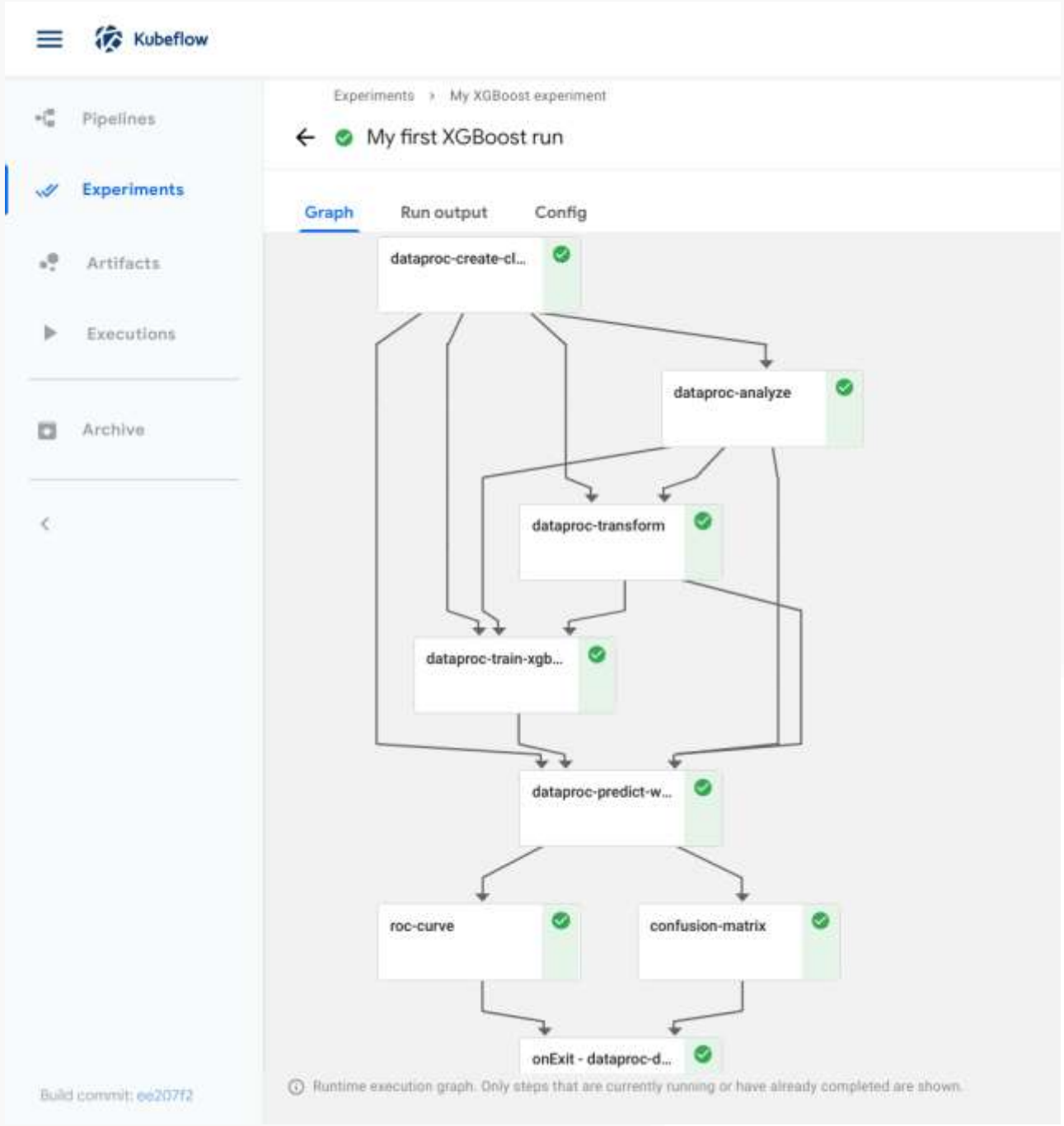
BACK

Experiment Name: random-example

Experiment Namespace: anonymous

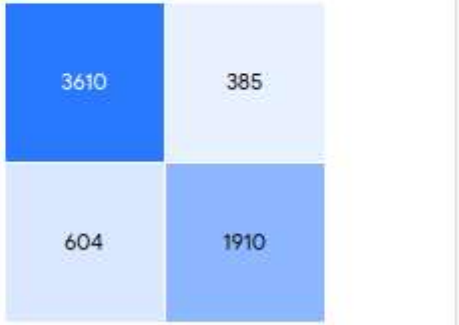
VIEW EXPERIMENT VIEW SUGGESTION




Kubeflow
Pipelines
Experiments
Artifacts
Executions
Archive
Build commit: ee207f2
Experiments > My XGBoost experiment
My first XGBoost run
Graph
Run output
Config
dataproc-create-cl...
dataproc-analyze
dataproc-transform
dataproc-train-xgb...
dataproc-predict-w...
roc-curve
confusion-matrix
onExit - dataproc-d...
Runtime execution graph. Only steps that are currently running or have already completed are shown.

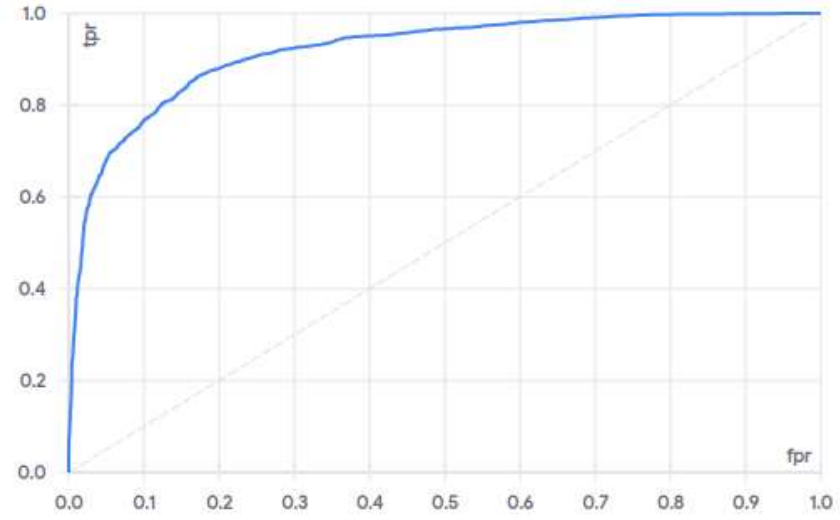
Artifacts Input/Output Logs

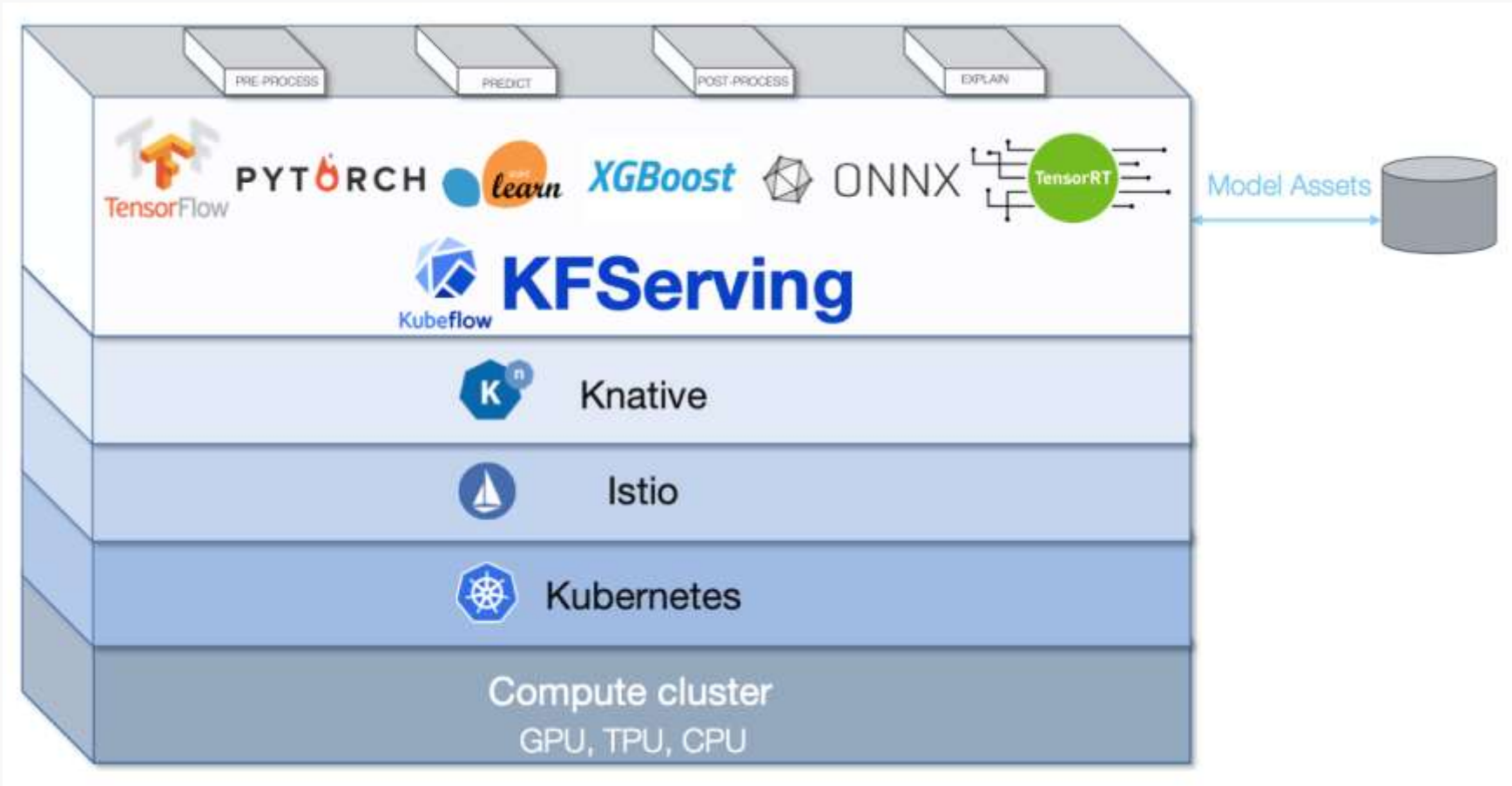
Confusion matrix



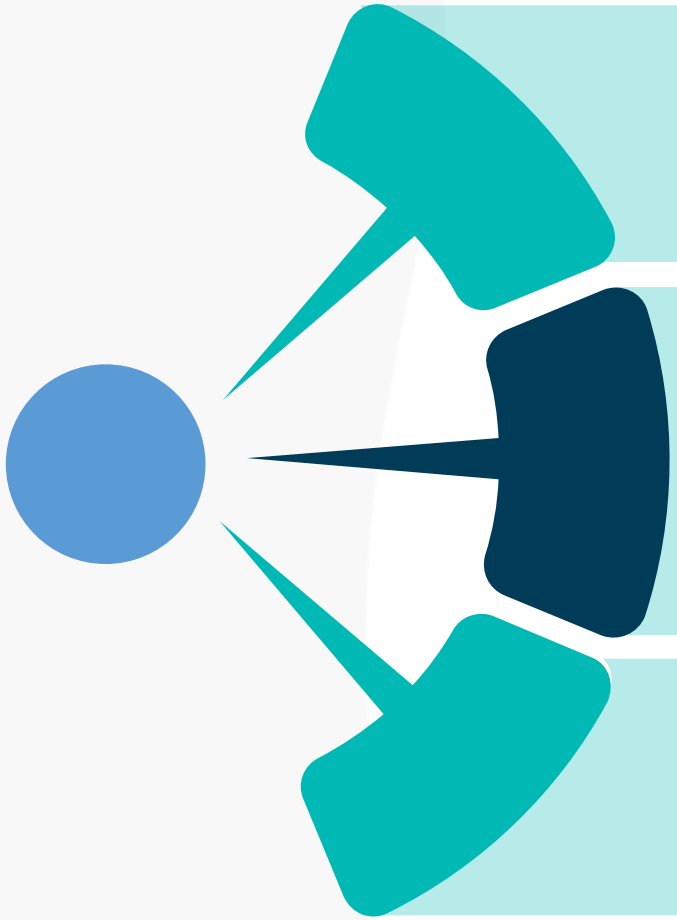
Artifacts Input/Output Logs

ROC Curve




PRE-PROCESS
PREDICT
POST-PROCESS
EXPLAIN
TensorFlow
PYTORCH
learn
XGBoost
ONNX
TensorRT
Model Assets
Kubeflow
KFServing
Knative
Istio
Kubernetes
Compute cluster
GPU, TPU, CPU

Vantagens



Utiliza tecnologia Kubernetes (escalável, alta disponibilidade, automatização).

Agrega todas as partes de uma equipe de analytics.

Pensado para ambientes de produção.

Conclusão

- ✓ KubeFlow é uma ferramenta que engloba quase todo o pipeline de Ciência de Dados. Pode ser utilizado tanto por Engenheiros de Dados quanto por Cientistas de Dados e Engenheiros de Machine Learning.
- ✓ KubeFlow utiliza o estado da arte em tecnologia de deploy de soluções escaláveis, com alta disponibilidade e automatizadas, o Kubernetes.
- ✓ Equipes com maior experiência com Kubernetes tendem a ter melhores resultados com a ferramenta.

Na próxima aula

01.

Prefect.

Fundamentos

Bootcamp Engenharia de Dados

Aula 5.6. Prefect

Prof. Dr. Neylson Crepalde

Nesta aula

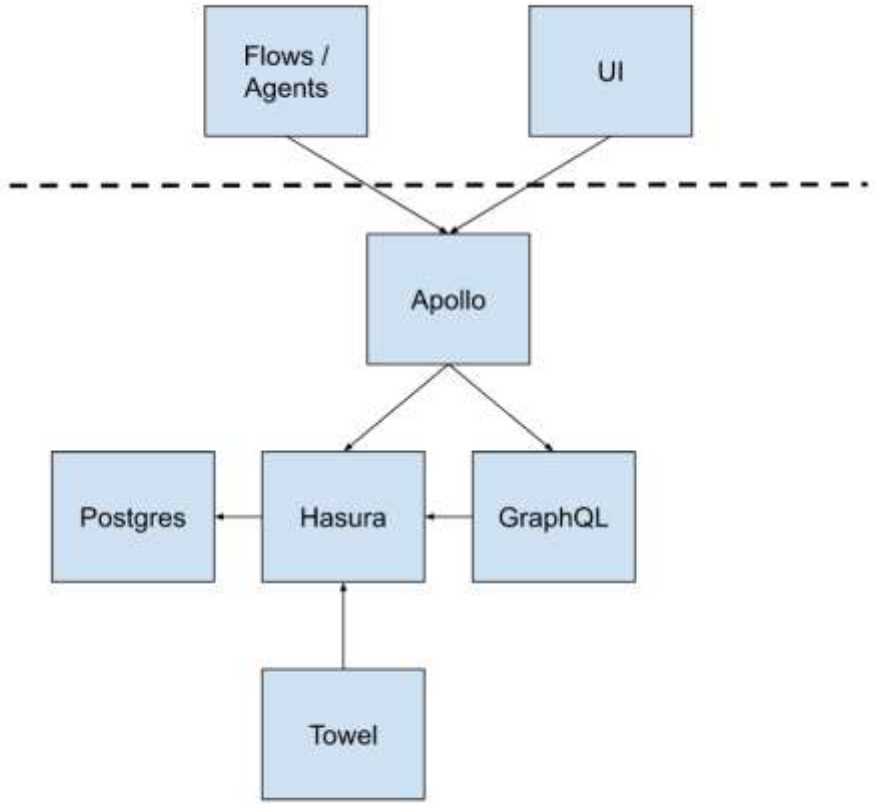
- Prefect.

PREFECT

Definições

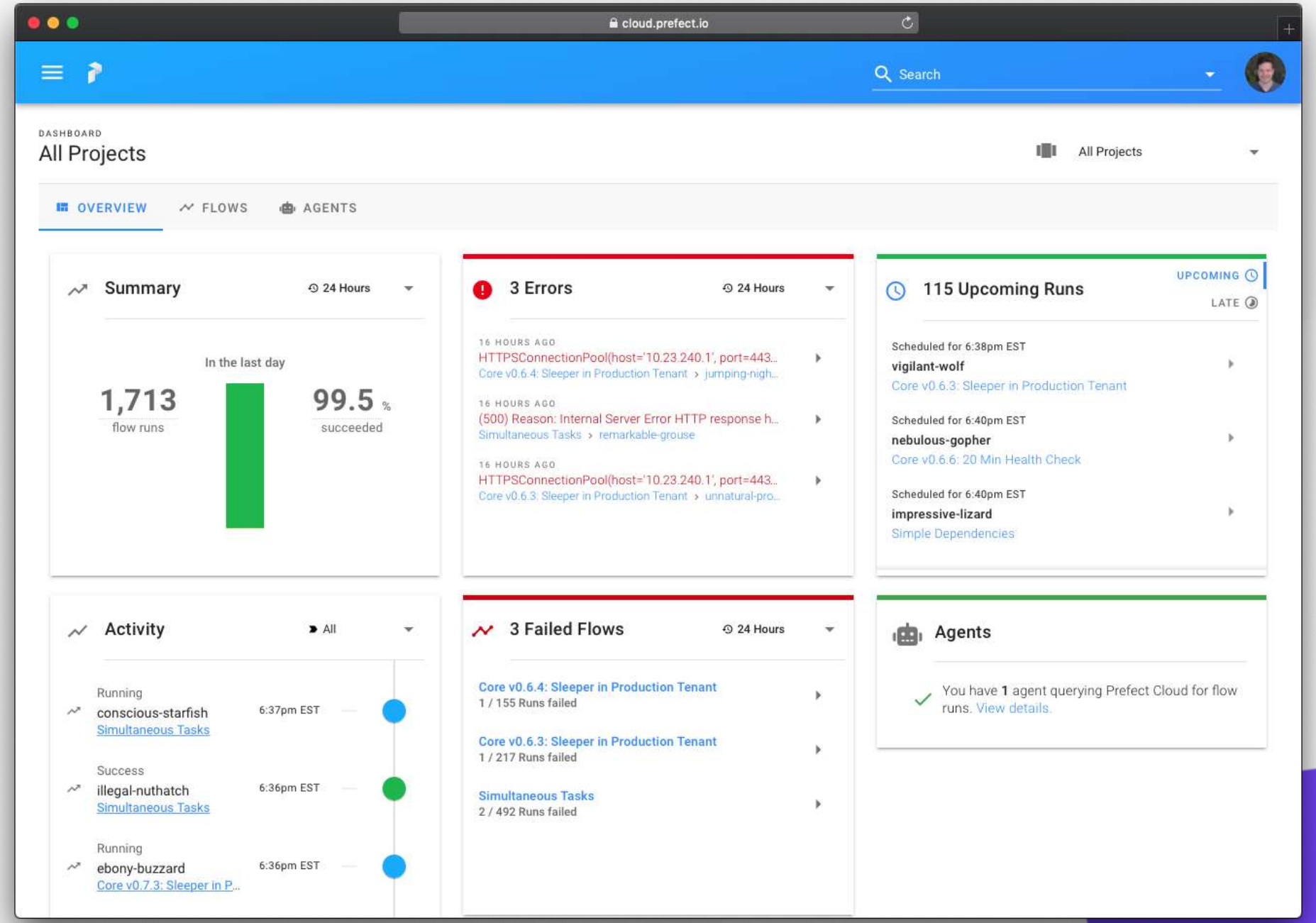
- Prefect se propõe a ser o jeito mais fácil para automatização de dataflow.
- Lógica muito parecida com o AirFlow – criação, schedule e monitoramento de Pipelines.
- O grande diferencial – além da biblioteca “Core” para orquestração de pipelines, Prefect oferece uma Cloud própria com vários níveis de assinatura (inclusive gratuito) para servir de backend para as execuções.

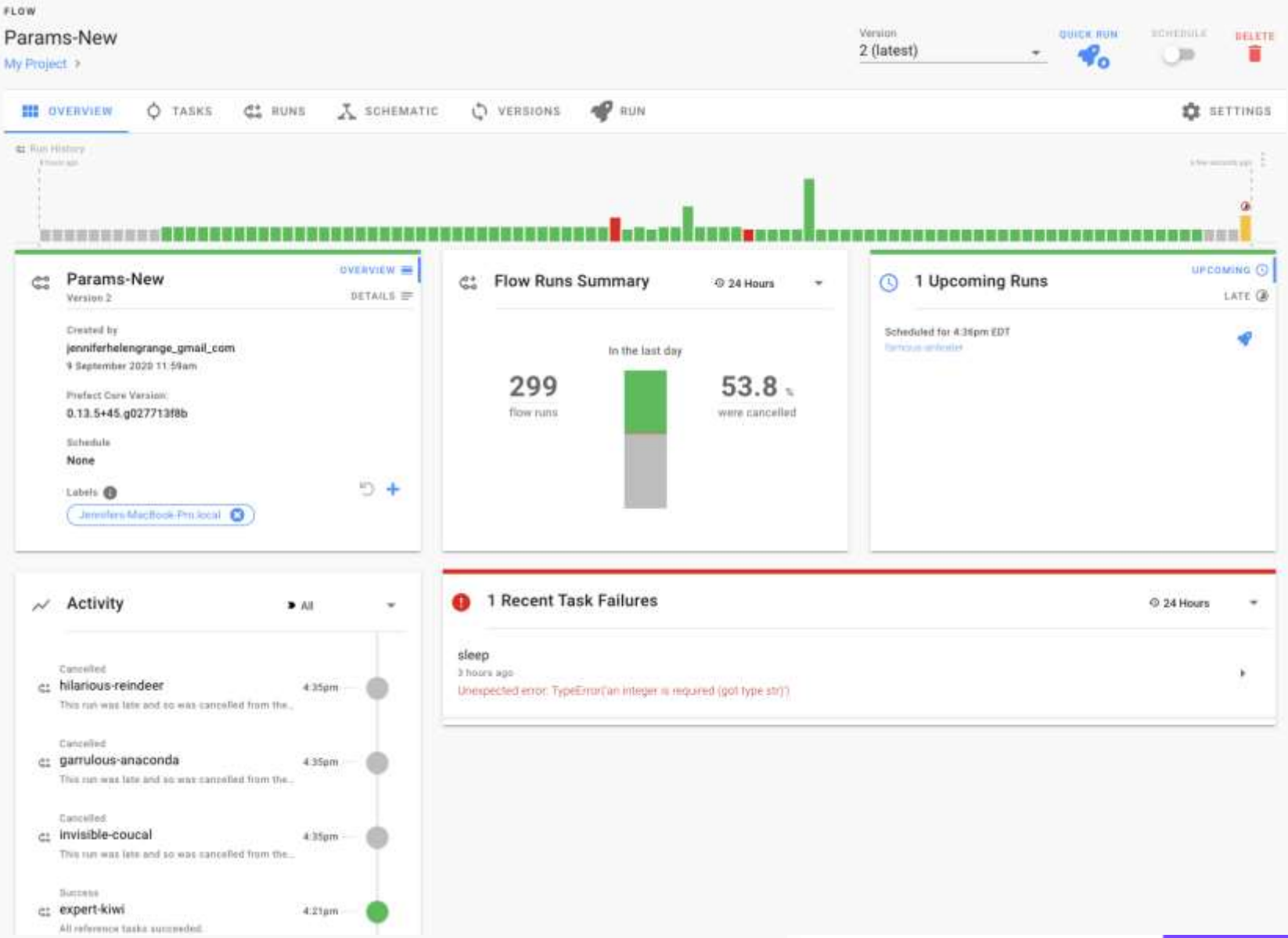
Arquitetura

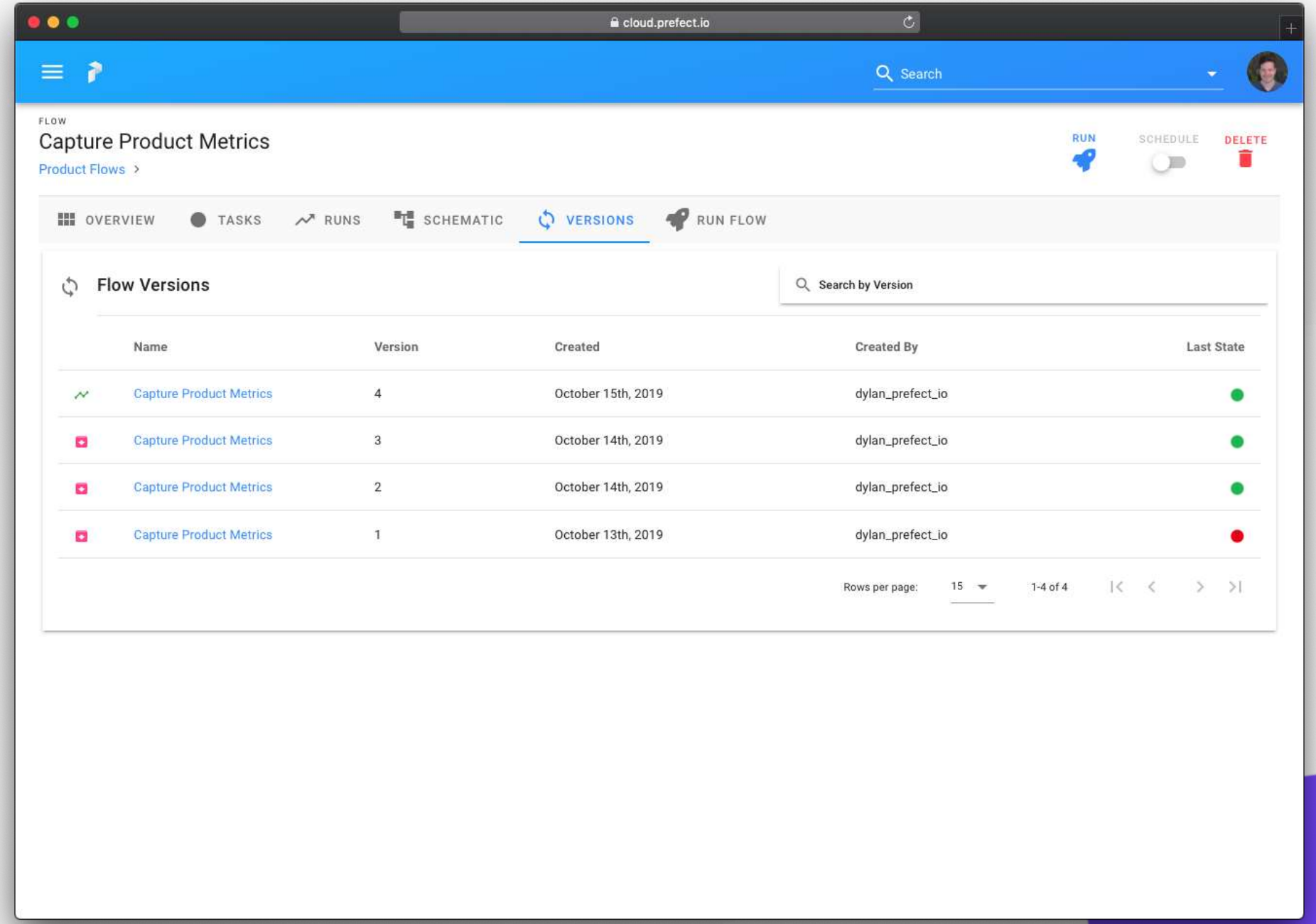


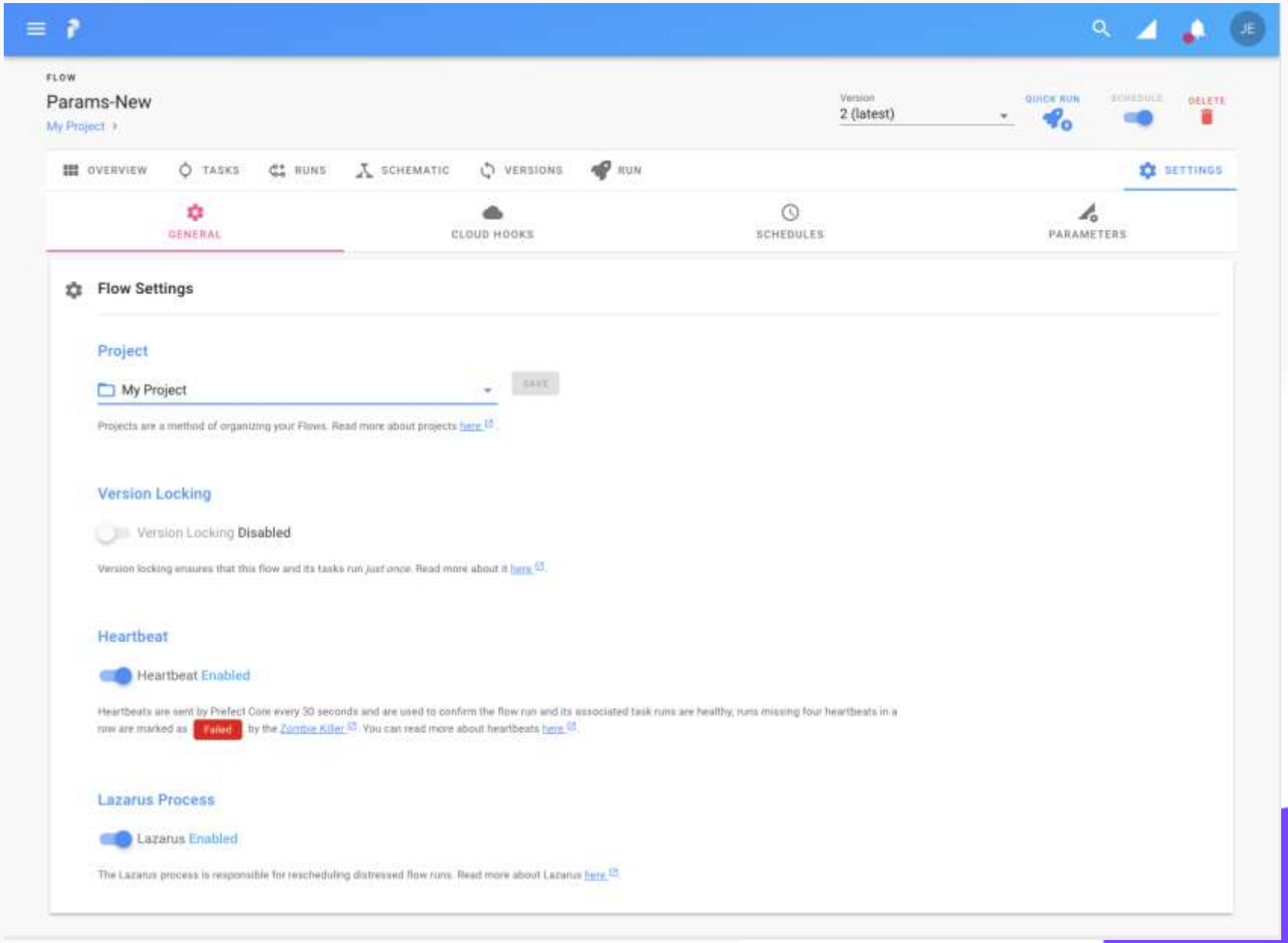
Arquitetura

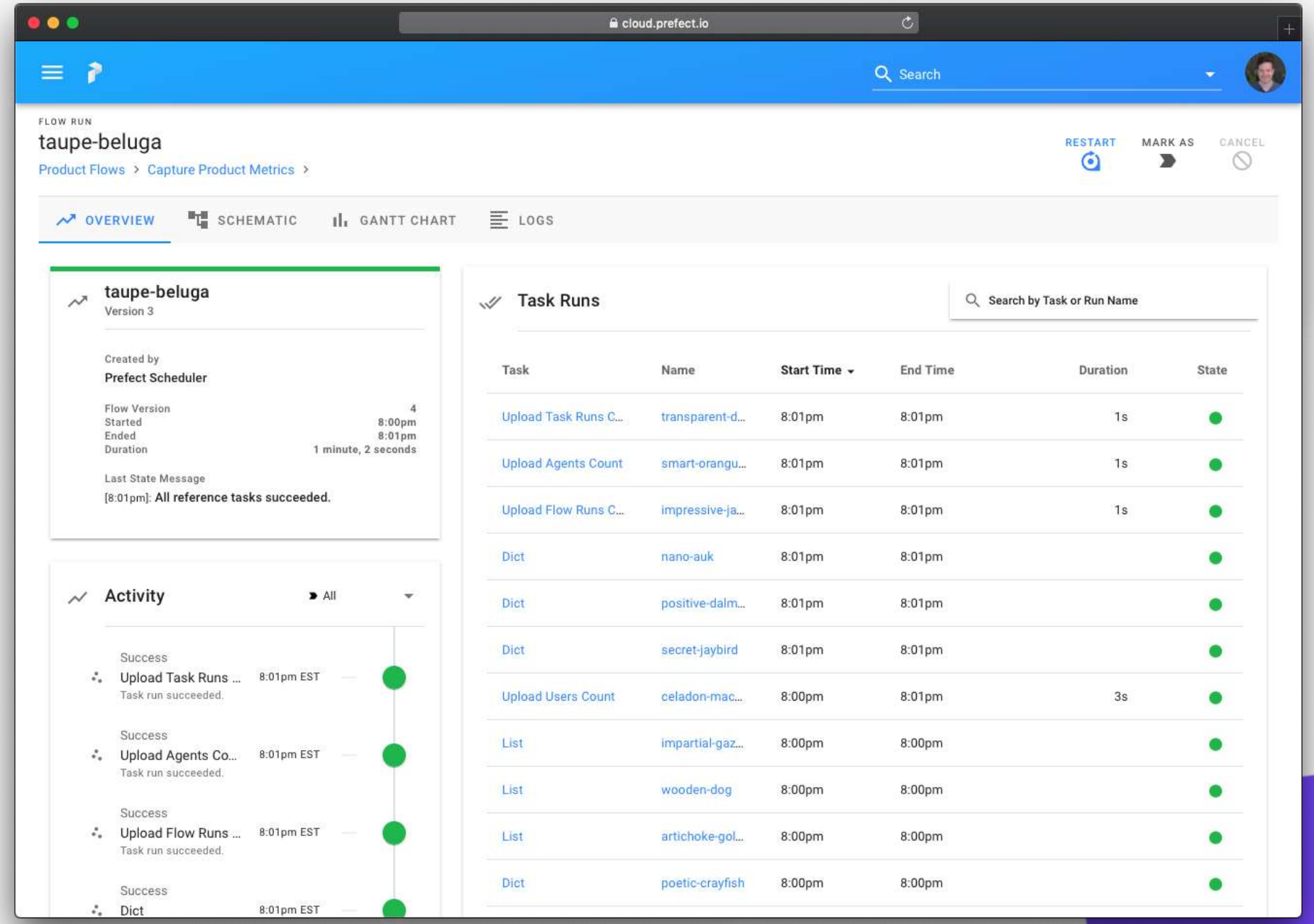
- UI: Interface do usuário.
- Apollo: Endpoint para interação com o servidor.
- PostgreSQL: camada de persistência dos dados.
- Hasura: GraphQL API para consultas aos metadados no PostgreSQL.
- GraphQL: As regras de negócio do servidor.
- Towel: Utilitários:
    - Scheduler: Agendamento de Execuções;
    - Zombie Killer: marca tarefas como falha;
    - Lazarus: reagenda execuções que mantém um estado não usual por um período.

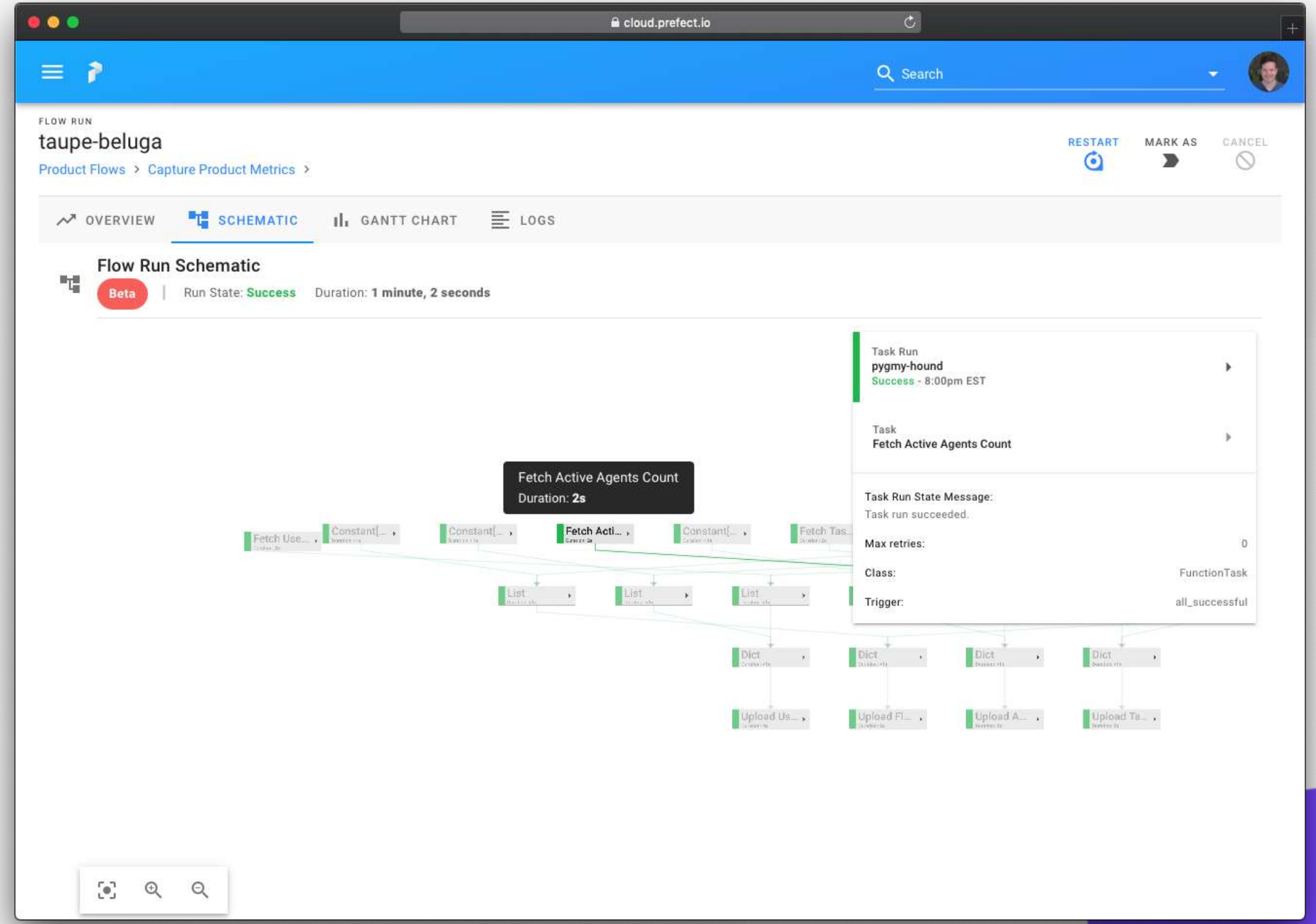

cloud.prefect.io
Search
DASHBOARD
All Projects
All Projects
OVERVIEW
FLOWS
AGENTS
Summary
24 Hours
In the last day
1,713
flow runs
99.5 %
succeeded
3 Errors
24 Hours
16 HOURS AGO
HTTPSConnectionPool(host='10.23.240.1', port=443...
Core v0.6.4: Sleeper in Production Tenant › jumping-nigh...
16 HOURS AGO
(500) Reason: Internal Server Error HTTP response h...
Simultaneous Tasks › remarkable-grouse
16 HOURS AGO
HTTPSConnectionPool(host='10.23.240.1', port=443...
Core v0.6.3: Sleeper in Production Tenant › unnatural-pro...
115 Upcoming Runs
UPCOMING
LATE
Scheduled for 6:38pm EST
vigilant-wolf
Core v0.6.3: Sleeper in Production Tenant
Scheduled for 6:40pm EST
nebulous-gopher
Core v0.6.6: 20 Min Health Check
Scheduled for 6:40pm EST
impressive-lizard
Simple Dependencies
Activity
All
Running
conscious-starfish
Simultaneous Tasks
6:37pm EST
Success
illegal-nuthatch
Simultaneous Tasks
6:36pm EST
Running
ebony-buzzard
Core v0.7.3: Sleeper in P...
6:36pm EST
3 Failed Flows
24 Hours
Core v0.6.4: Sleeper in Production Tenant
1 / 155 Runs failed
Core v0.6.3: Sleeper in Production Tenant
1 / 217 Runs failed
Simultaneous Tasks
2 / 492 Runs failed
Agents
You have 1 agent querying Prefect Cloud for flow runs. View details.

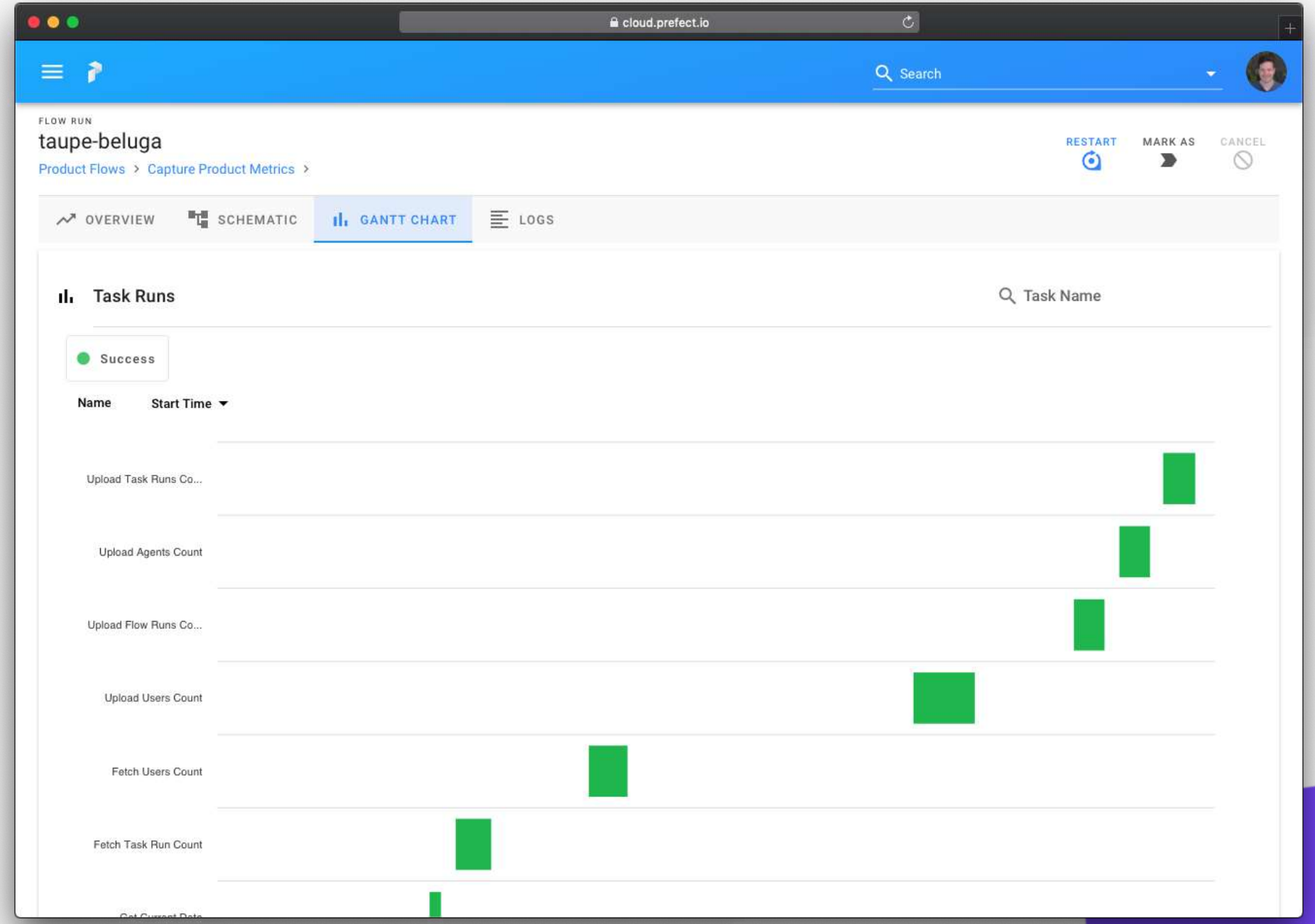

FLOW
Params-New
My Project
Version
2 (latest)
QUICK RUN
SCHEDULE
DELETE
OVERVIEW
TASKS
RUNS
SCHEMATIC
VERSIONS
RUN
SETTINGS
Run History
Params-New
Version 2
OVERVIEW
DETAILS
Created by
jenniferhelengrange_gmail_com
9 September 2020 11:59am
Prefect Core Version
0.13.5+45.g027713f8b
Schedule
None
Labels
Jennifers-MacBook-Pro.local
Flow Runs Summary
24 Hours
In the last day
299
flow runs
53.8
were cancelled
1 Upcoming Runs
UPCOMING
LATE
Scheduled for 4:36pm EDT
Activity
All
Cancelled
hilarious-reindeer
4:35pm
This run was late and so was cancelled from the...
Cancelled
garrulous-anaconda
4:35pm
This run was late and so was cancelled from the...
Cancelled
invisible-coucal
4:35pm
This run was late and so was cancelled from the...
Success
expert-kiwi
4:21pm
All reference tasks succeeded.
1 Recent Task Failures
24 Hours
sleep
3 hours ago
Unexpected error: TypeError('an integer is required (got type str)')

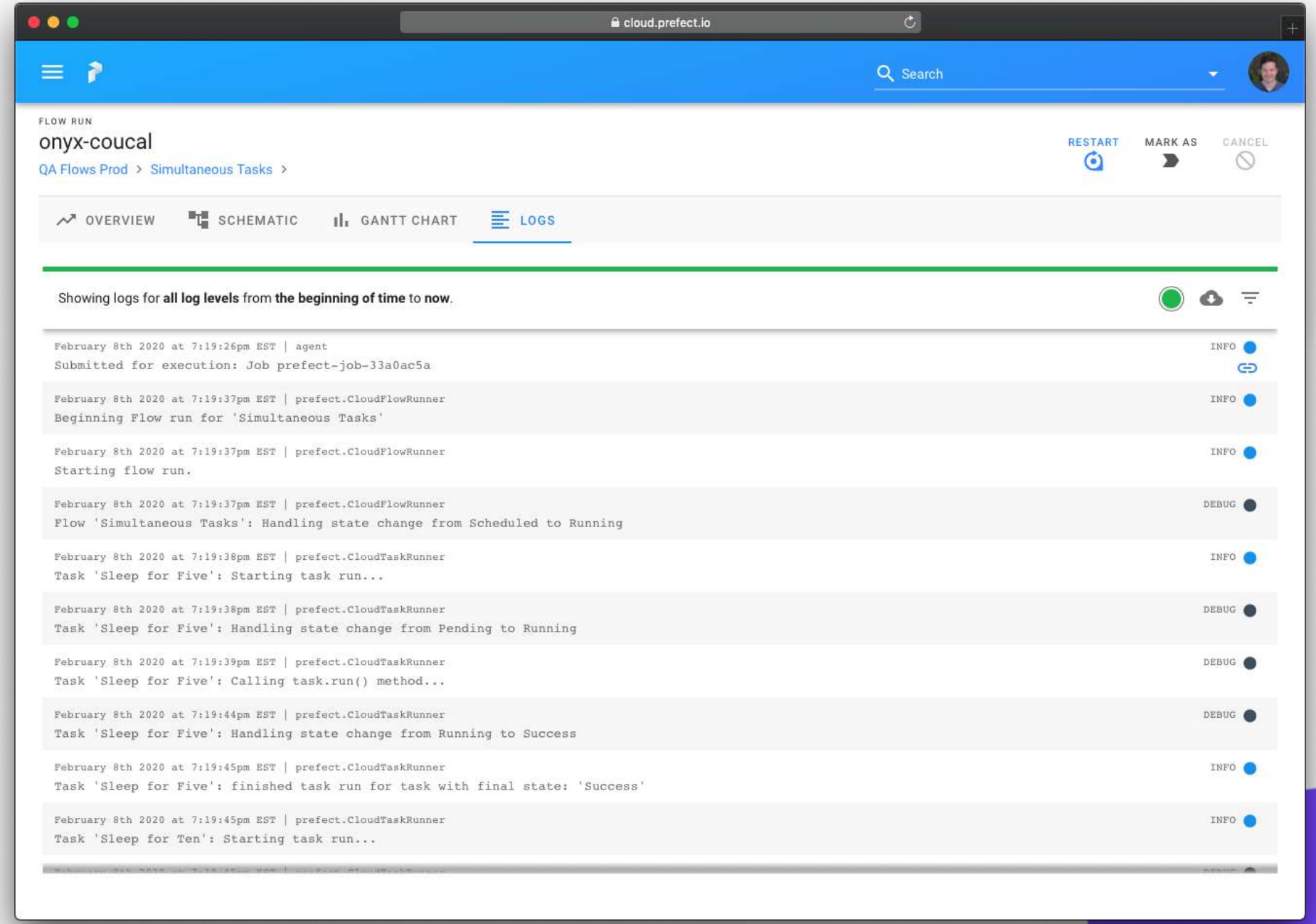

cloud.prefect.io
Search
FLOW
Capture Product Metrics
Product Flows >
RUN
SCHEDULE
DELETE
OVERVIEW
TASKS
RUNS
SCHEMATIC
VERSIONS
RUN FLOW
Flow Versions
Search by Version
Name
Version
Created
Created By
Last State
Capture Product Metrics
4
October 15th, 2019
dylan_prefect_io
Capture Product Metrics
3
October 14th, 2019
dylan_prefect_io
Capture Product Metrics
2
October 14th, 2019
dylan_prefect_io
Capture Product Metrics
1
October 13th, 2019
dylan_prefect_io
Rows per page:
15
1-4 of 4

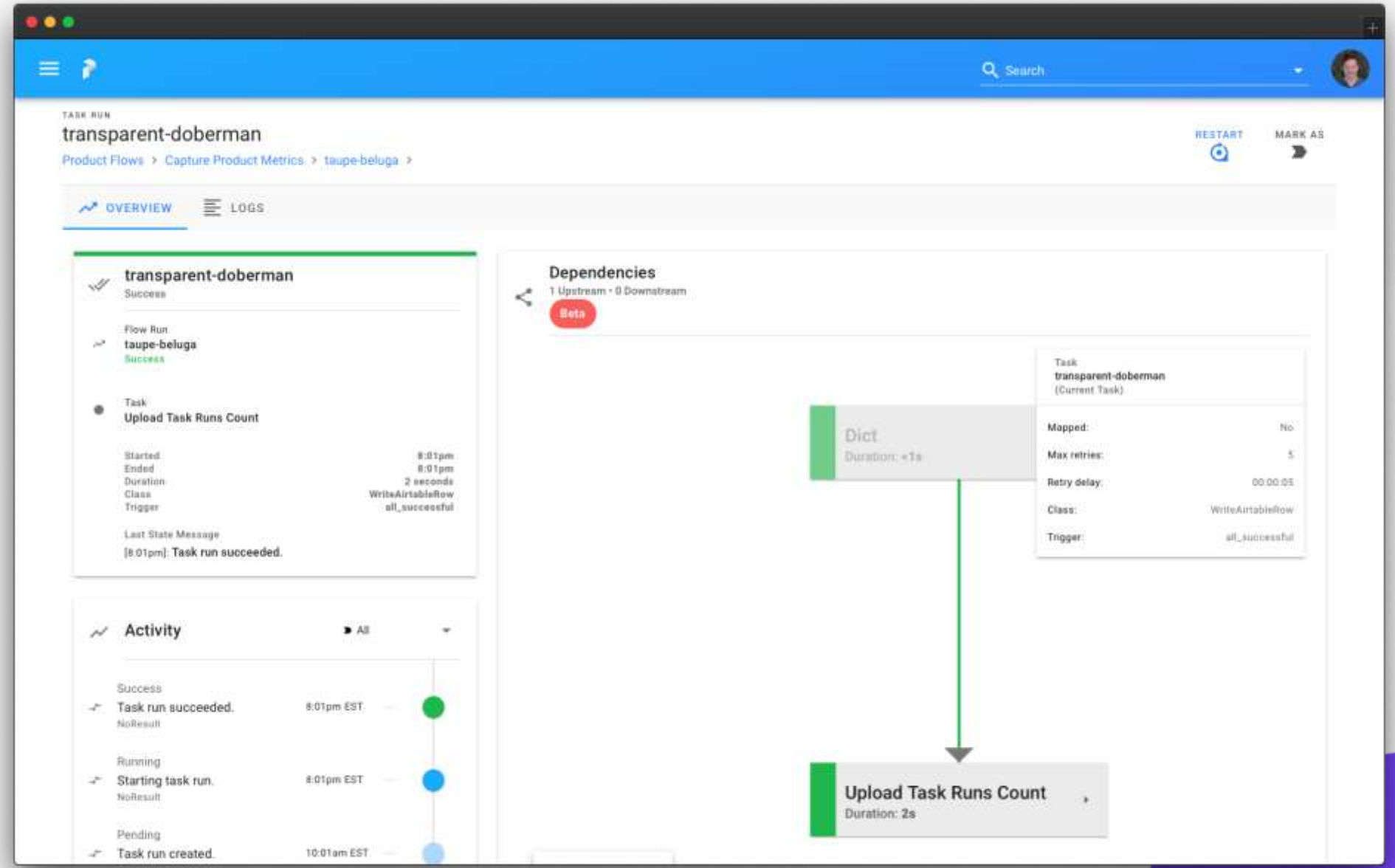

FLOW
Params-New
My Project
Version
2 (latest)
QUICK RUN
SCHEDULE
DELETE
OVERVIEW
TASKS
RUNS
SCHEMATIC
VERSIONS
RUN
SETTINGS
GENERAL
CLOUD HOOKS
SCHEDULES
PARAMETERS
Flow Settings
Project
My Project
SAVE
Projects are a method of organizing your Flows. Read more about projects here.
Version Locking
Version Locking Disabled
Version locking ensures that this flow and its tasks run just once. Read more about it here.
Heartbeat
Heartbeat Enabled
Heartbeats are sent by Prefect Core every 30 seconds and are used to confirm the flow run and its associated task runs are healthy; runs missing four heartbeats in a row are marked as Failed by the Zombie Killer. You can read more about heartbeats here.
Lazarus Process
Lazarus Enabled
The Lazarus process is responsible for rescheduling distressed flow runs. Read more about Lazarus here.

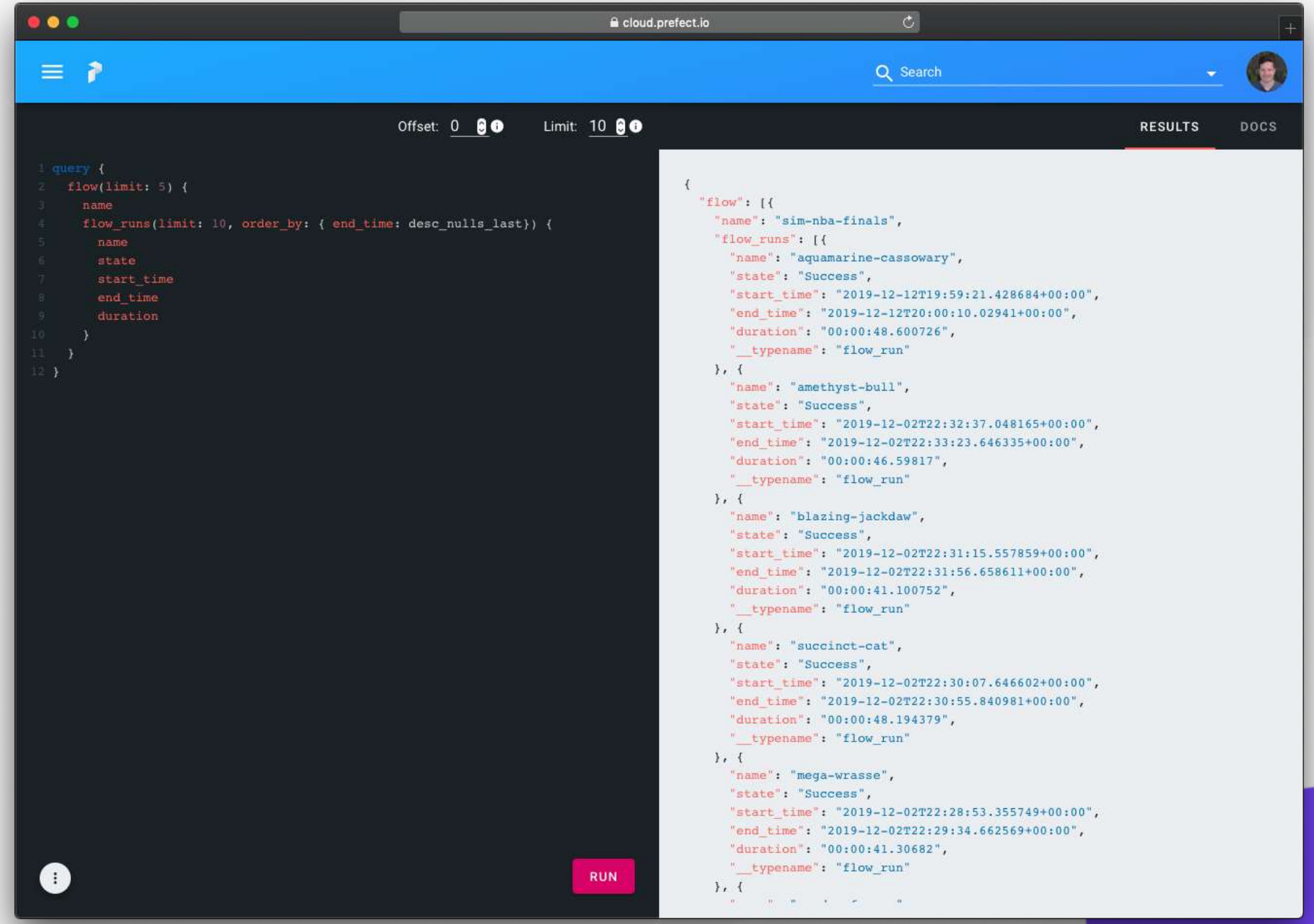

cloud.prefect.io
Search
FLOW RUN
taupe-beluga
Product Flows > Capture Product Metrics >
RESTART
MARK AS
CANCEL
OVERVIEW
SCHEMATIC
GANTT CHART
LOGS
taupe-beluga
Version 3
Created by
Prefect Scheduler
Flow Version 4
Started 8:00pm
Ended 8:01pm
Duration 1 minute, 2 seconds
Last State Message
[8:01pm]: All reference tasks succeeded.
Activity
All
Success
Upload Task Runs ... 8:01pm EST
Task run succeeded.
Success
Upload Agents Co... 8:01pm EST
Task run succeeded.
Success
Upload Flow Runs ... 8:01pm EST
Task run succeeded.
Success
Dict 8:01pm EST
Task Runs
Search by Task or Run Name
Task
Name
Start Time
End Time
Duration
State
Upload Task Runs C... transparent-d... 8:01pm 8:01pm 1s
Upload Agents Count smart-orangu... 8:01pm 8:01pm 1s
Upload Flow Runs C... impressive-ja... 8:01pm 8:01pm 1s
Dict nano-auk 8:01pm 8:01pm
Dict positive-dalm... 8:01pm 8:01pm
Dict secret-jaybird 8:01pm 8:01pm
Upload Users Count celadon-mac... 8:00pm 8:01pm 3s
List impartial-gaz... 8:00pm 8:00pm
List wooden-dog 8:00pm 8:00pm
List artichoke-gol... 8:00pm 8:00pm
Dict poetic-crayfish 8:00pm 8:00pm


cloud.prefect.io
Search
FLOW RUN
taupe-beluga
Product Flows > Capture Product Metrics >
RESTART
MARK AS
CANCEL
OVERVIEW
SCHEMATIC
GANTT CHART
LOGS
Flow Run Schematic
Beta
Run State: Success
Duration: 1 minute, 2 seconds
Task Run
pygmy-hound
Success - 8:00pm EST
Task
Fetch Active Agents Count
Fetch Active Agents Count
Duration: 2s
Task Run State Message:
Task run succeeded.
Max retries: 0
Class: FunctionTask
Trigger: all_successful


cloud.prefect.io
Search
FLOW RUN
taupe-beluga
Product Flows > Capture Product Metrics >
RESTART
MARK AS
CANCEL
OVERVIEW
SCHEMATIC
GANTT CHART
LOGS
Task Runs
Task Name
Success
Name
Start Time
Upload Task Runs Co...
Upload Agents Count
Upload Flow Runs Co...
Upload Users Count
Fetch Users Count
Fetch Task Run Count


cloud.prefect.io
Search
FLOW RUN
onyx-coucal
QA Flows Prod > Simultaneous Tasks >
RESTART
MARK AS
CANCEL
OVERVIEW
SCHEMATIC
GANTT CHART
LOGS
Showing logs for all log levels from the beginning of time to now.
February 8th 2020 at 7:19:26pm EST | agent
INFO
Submitted for execution: Job prefect-job-33a0ac5a
February 8th 2020 at 7:19:37pm EST | prefect.CloudFlowRunner
INFO
Beginning Flow run for 'Simultaneous Tasks'
February 8th 2020 at 7:19:37pm EST | prefect.CloudFlowRunner
INFO
Starting flow run.
February 8th 2020 at 7:19:37pm EST | prefect.CloudFlowRunner
DEBUG
Flow 'Simultaneous Tasks': Handling state change from Scheduled to Running
February 8th 2020 at 7:19:38pm EST | prefect.CloudTaskRunner
INFO
Task 'Sleep for Five': Starting task run...
February 8th 2020 at 7:19:38pm EST | prefect.CloudTaskRunner
DEBUG
Task 'Sleep for Five': Handling state change from Pending to Running
February 8th 2020 at 7:19:39pm EST | prefect.CloudTaskRunner
DEBUG
Task 'Sleep for Five': Calling task.run() method...
February 8th 2020 at 7:19:44pm EST | prefect.CloudTaskRunner
DEBUG
Task 'Sleep for Five': Handling state change from Running to Success
February 8th 2020 at 7:19:45pm EST | prefect.CloudTaskRunner
INFO
Task 'Sleep for Five': finished task run for task with final state: 'Success'
February 8th 2020 at 7:19:45pm EST | prefect.CloudTaskRunner
INFO
Task 'Sleep for Ten': Starting task run...


Search
TASK RUN
transparent-doberman
Product Flows > Capture Product Metrics > taupe-beluga >
RESTART
MARK AS
OVERVIEW
LOGS
transparent-doberman
Success
Flow Run
taupe-beluga
Success
Task
Upload Task Runs Count
Started 8:01pm
Ended 8:01pm
Duration 2 seconds
Class WriteAirtableRow
Trigger all_successful
Last State Message
[8:01pm]: Task run succeeded.
Activity
All
Success
Task run succeeded.
NoResult
8:01pm EST
Running
Starting task run.
NoResult
8:01pm EST
Pending
Task run created.
10:01am EST
Dependencies
1 Upstream • 0 Downstream
Beta
Task
transparent-doberman
(Current Task)
Dict
Duration: <1s
Mapped: No
Max retries: 5
Retry delay: 00:00:05
Class: WriteAirtableRow
Trigger: all_successful
Upload Task Runs Count
Duration: 2s


cloud.prefect.io
Search
Offset: 0
Limit: 10
RESULTS
DOCS
query {
flow(limit: 5) {
name
flow_runs(limit: 10, order_by: { end_time: desc_nulls_last}) {
name
state
start_time
end_time
duration
}
}
}
{
"flow": [{
"name": "sim-nba-finals",
"flow_runs": [{
"name": "aquamarine-cassowary",
"state": "Success",
"start_time": "2019-12-12T19:59:21.428684+00:00",
"end_time": "2019-12-12T20:00:10.02941+00:00",
"duration": "00:00:48.600726",
"__typename": "flow_run"
}, {
"name": "amethyst-bull",
"state": "Success",
"start_time": "2019-12-02T22:32:37.048165+00:00",
"end_time": "2019-12-02T22:33:23.646335+00:00",
"duration": "00:00:46.59817",
"__typename": "flow_run"
}, {
"name": "blazing-jackdaw",
"state": "Success",
"start_time": "2019-12-02T22:31:15.557859+00:00",
"end_time": "2019-12-02T22:31:56.658611+00:00",
"duration": "00:00:41.100752",
"__typename": "flow_run"
}, {
"name": "succinct-cat",
"state": "Success",
"start_time": "2019-12-02T22:30:07.646602+00:00",
"end_time": "2019-12-02T22:30:55.840981+00:00",
"duration": "00:00:48.194379",
"__typename": "flow_run"
}, {
"name": "mega-wrasse",
"state": "Success",
"start_time": "2019-12-02T22:28:53.355749+00:00",
"end_time": "2019-12-02T22:29:34.662569+00:00",
"duration": "00:00:41.30682",
"__typename": "flow_run"
}, {
RUN

Conclusão

- ✓ Prefect é uma ferramenta de Orquestração de Pipelines de dados moderna, robusta, simples de usar e possui um ótimo diferencial, a saber, uma cloud própria que pode ser usada de backend para as execuções dos Flows.
- ✓ Prefect tem sido amplamente utilizado por times de dados ao redor do mundo.
- ✓ Possui integração com Docker e Kubernetes para deploy em ambientes diferentes.

Na próxima aula

01.

Data Flow na Prática com AirFlow.

02.

Instalação do AirFlow.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 6. Data Flow na prática - AirFlow

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.1. Instalação do AirFlow

Prof. Dr. Neylson Crepalde

Nesta aula

- Instalação do AirFlow.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Airflow funcionando na nuvem.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.2. AirFlow rodando na nuvem

Prof. Dr. Neylson Crepalde

Nesta aula

- AirFlow funcionando na nuvem.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Tasks do AirFlow.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.3. Tasks do AirFlow

Prof. Dr. Neylson Crepalde

Nesta aula

- Tasks do AirFlow.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Programando execuções do Pipeline.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.4. Programando execuções do Pipeline

Prof. Dr. Neylson Crepalde

Nesta aula

- Programando execuções do Pipeline.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Condicionais.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.5. Condicionais

Prof. Dr. Neylson Crepalde

Nesta aula

- Condicionais.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Paralelismos.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.6. Paralelismos

Prof. Dr. Neylson Crepalde

Nesta aula

- Paralelismos.

Na próxima aula

01.

DataFlow na prática – AirFlow.

02.

Integrações para entrega.

Fundamentos

Bootcamp Engenharia de Dados

Aula 6.7. Integrações para entrega

Prof. Dr. Neylson Crepalde

Nesta aula

- Integrações para entrega.

Na próxima aula

01.

DataFlow na prática – Prefect.

02.

Configuração do ambiente Prefect na nuvem.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 7. Data Flow na prática - Prefect

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 7.1. Configuração do ambiente Prefect na nuvem

Prof. Dr. Neylson Crepalde

Nesta aula

- Configuração do ambiente Prefect na nuvem.

Na próxima aula

01.

DataFlow na prática – Prefect.

02.

Tasks do Prefect.

Fundamentos

Bootcamp Engenharia de Dados

Aula 7.2. Tasks do Prefect

Prof. Dr. Neylson Crepalde

Nesta aula

- Tasks do Prefect.

Na próxima aula

01.

DataFlow na prática – Prefect.

02.

Programação de execuções e analisando o dashboard.

Fundamentos

Bootcamp Engenharia de Dados

Aula 7.3. Programando execuções e analisando o dashboard

Prof. Dr. Neylson Crepalde

Nesta aula

- Programação de execuções e analisando o dashboard.

Na próxima aula

01.

DataFlow na prática – Prefect.

02.

Integrações para entrega.

Fundamentos

Bootcamp Engenharia de Dados

Aula 7.4. Integrações para entrega

Prof. Dr. Neylson Crepalde

Nesta aula

- Integrações para entrega.

Na próxima aula

01. Encerramento.

02. Resumo.

03. Outras ferramentas.

04. Próximos passos.

Fundamentos

Bootcamp Engenharia de Dados

Capítulo 8. Encerramento

Prof. Dr. Neylson Crepalde

Fundamentos

Bootcamp Engenharia de Dados

Aula 8.1. Resumo, outras ferramentas e próximos passos

Prof. Dr. Neylson Crepalde

Nesta aula

- ❑ Resumo.
- ❑ Outras ferramentas.
- ❑ Próximos passos.

Parabéns!

Você completou a primeira disciplina do Bootcamp de Engenharia de Dados!

Atenção com o DESAFIO.

Resumo

- Big Data, dados, fontes de dados.
- ETL.
- Técnicas de extração – Crawlers.
- Transformação – Limpeza, organização e estruturação de dados.
- Orquestração de Pipelines de Dados:
  - Drag and Drop;
  - Com código.
- Ferramentas “on premisses” e na nuvem.

Outras ferramentas

- Ferramentas nativas de provedores de serviços em nuvem – (AWS StepFunctions, Google Data Flow, Azure Logic Apps).
- Aprofundar nas diversas possibilidades de armazenamento (SQL, NoSQL, Grafos).

Próximos passos

Data Lake e Data Warehouse

Processamento distribuído (Hadoop, Spark)

Containers e Kubernetes

Arquitetura de Soluções

Obrigado!