

# Airflow

ETL Descomplicado

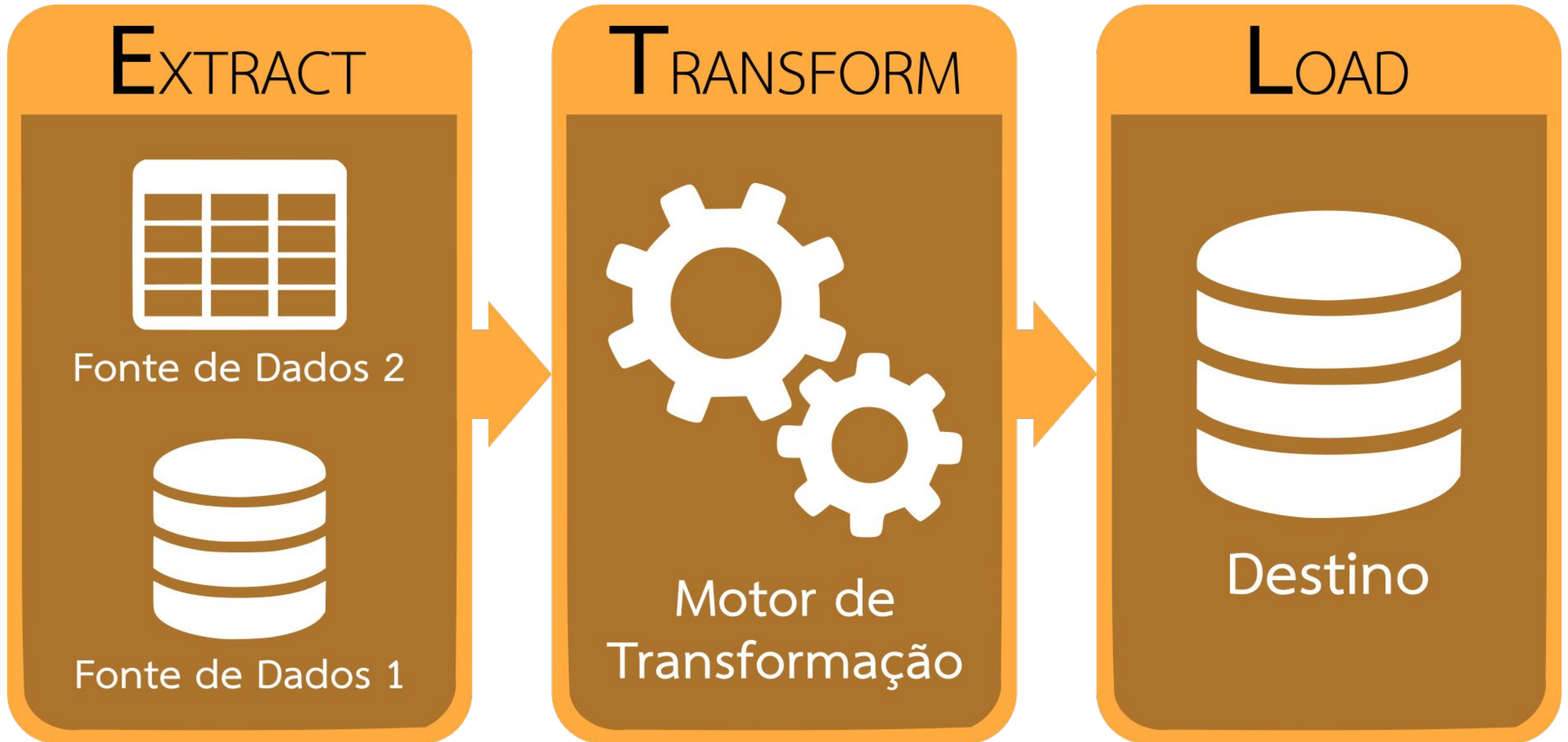
# Roteiro

---

- ETL
- Airflow
- Solução

# ETL - O que é?

---



- **Falhas**
- **Monitoramento**
- **Dependências**
- **Escalabilidade**

- **Falhas**

Tentar recuperar quantas vezes? Com que frequência?

- **Monitoramento**

- **Dependências**

- **Escalabilidade**

- **Falhas**

Tentar recuperar quantas vezes? Com que frequência?

- **Monitoramento**

Quando tempo o processo demora pra executar? O que falhou e o que deu certo?

- **Dependências**

- **Escalabilidade**

- **Falhas**

Tentar recuperar quantas vezes? Com que frequência?

- **Monitoramento**

Quando tempo o processo demora pra executar? O que falhou e o que deu certo?

- **Dependências**

Sincronizar tarefas em máquinas cron?

- **Escalabilidade**

- **Falhas**

Tentar recuperar quantas vezes? Com que frequência?

- **Monitoramento**

Quando tempo o processo demora pra executar? O que falhou e o que deu certo?

- **Dependências**

Sincronizar tarefas em máquinas cron?

- **Escalabilidade**

Falta *scheduler* centralizado



# Airflow - O que é?

---

Airflow é uma plataforma para **criar**, **agendar** e **monitorar** *workflows* (*pipelines*) programaticamente.



**ETL**  
**Machine Learning**  
**Execução de Scripts**

🕒 6,540 commits

🌿 8 branches

📦 114 releases

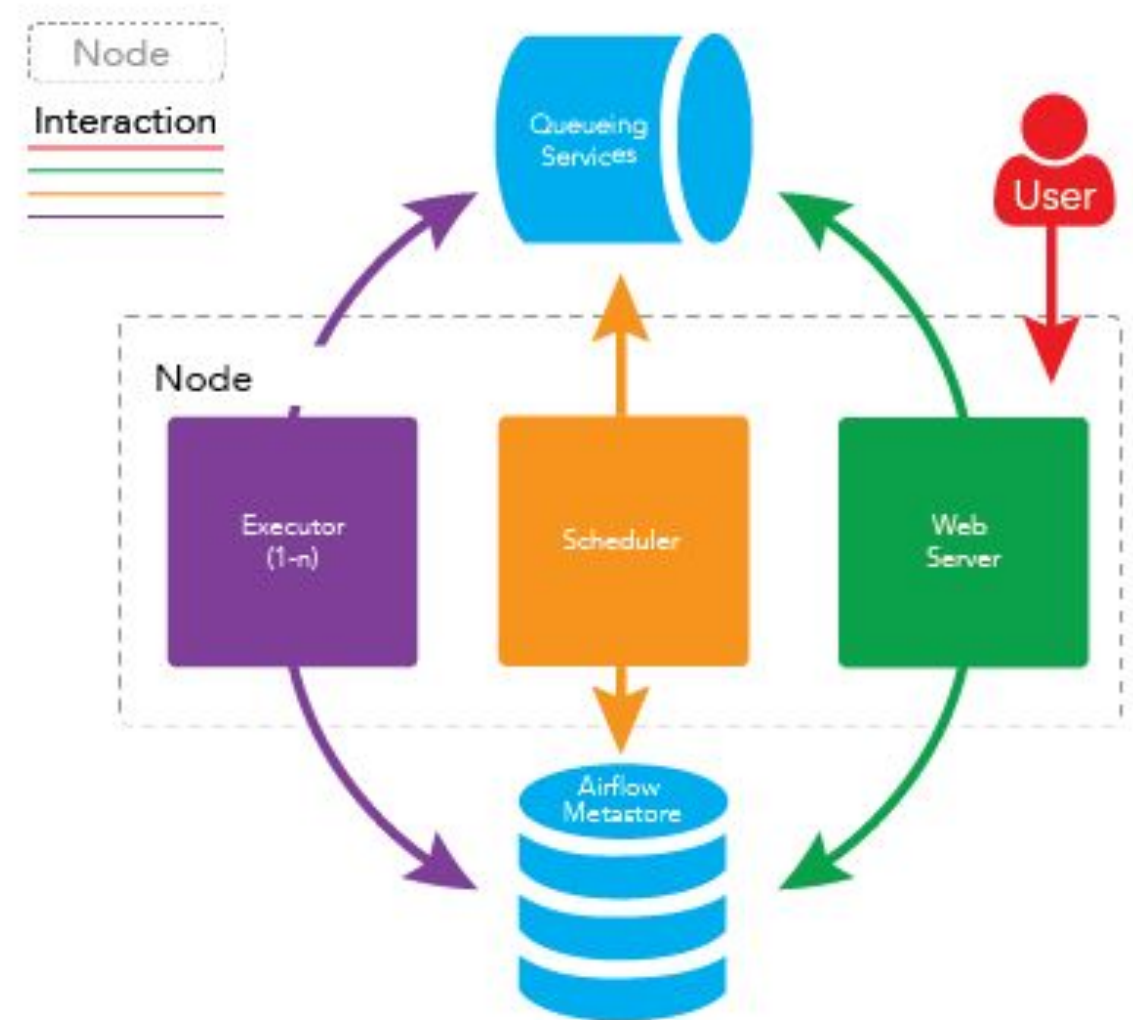
👤 839 contributors

📄 Apache-2.0

# Airflow - Componentes

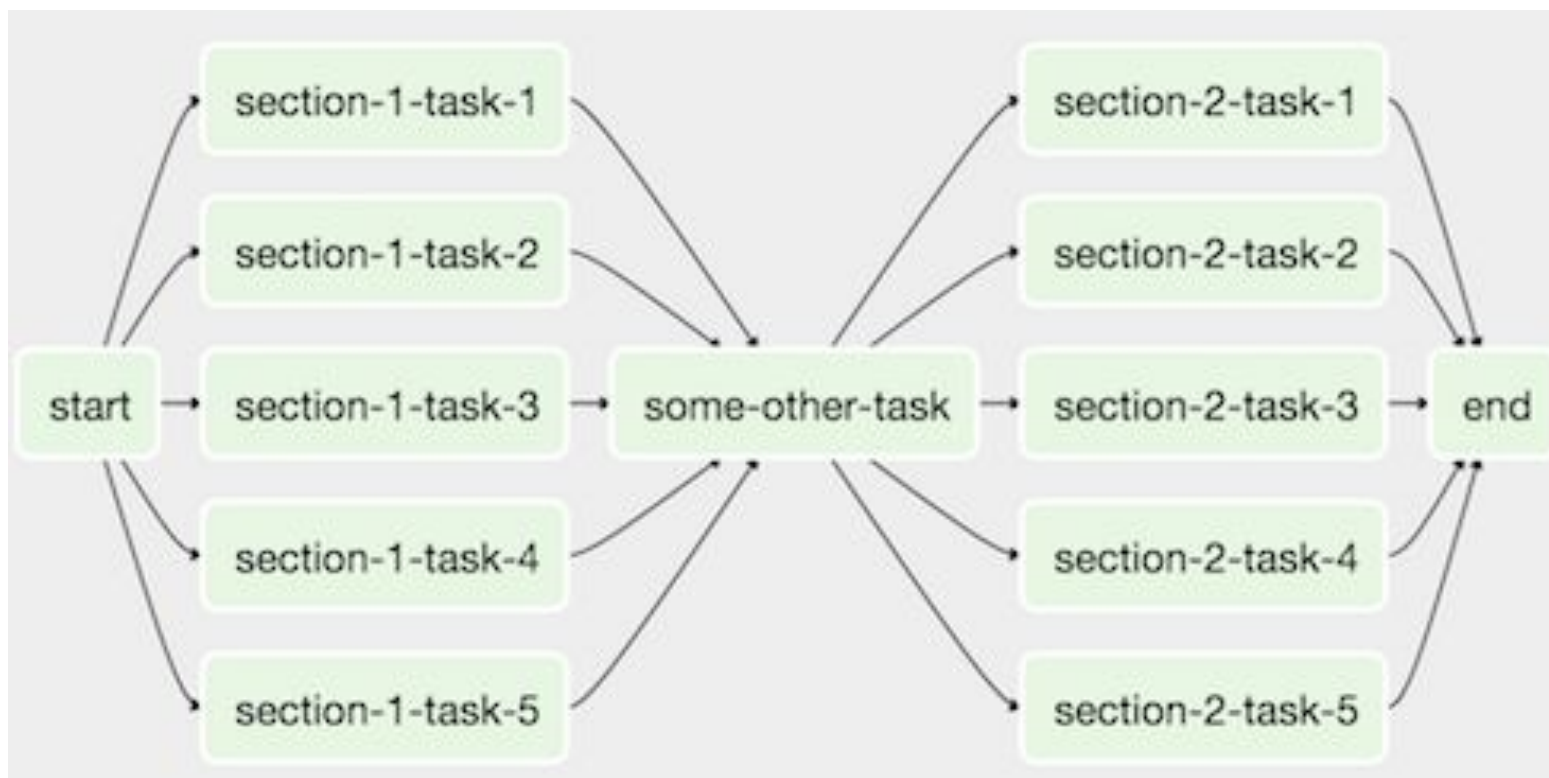
---

- **Webserver**
- **Scheduler**
- **Executor**
- **Metadata Database**



# Airflow - DAG

---



## Oficiais

- Microsoft Azure
- Amazon Web Services
- Databricks
- Google Cloud Platform
- Qubole


## Plugins

[github.com/airflow-plugins](https://github.com/airflow-plugins)

- Google Analytics
- Mongo
- Github
- REST Like API
- Google Sheets

...

# Airflow - Interface (DAGs)

 Airflow

DAGs


Data Profiling ▾

Browse ▾

Admin ▾

Docs ▾

About ▾

2018-09-07 22:14:10 UTC 

DAGs

Search: 

Showing 1 to 5 of 5 entries

«

<

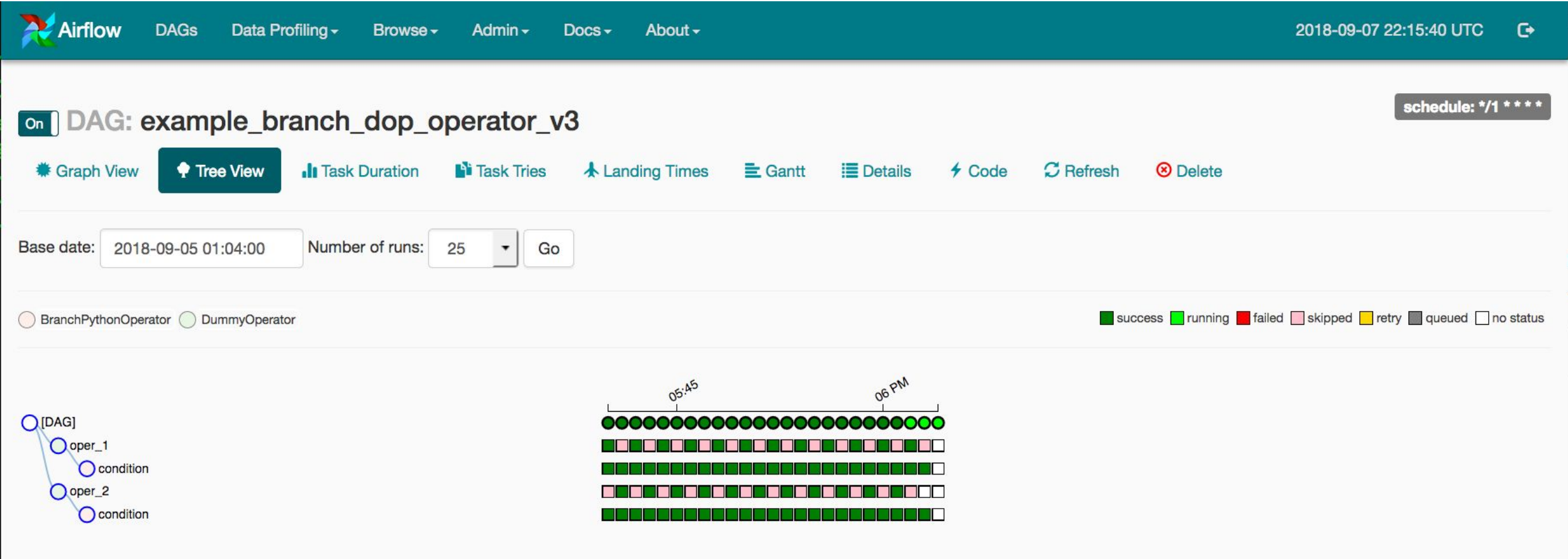
1

>


»

Show Paused DAGs

# Airflow - Interface (Árvore)



# Airflow - Interface (Grafo)

 Airflow

DAGsData ProfilingBrowseAdminDocsAbout

2018-09-07 22:29:47 UTC

On DAG: example\_bash\_operator

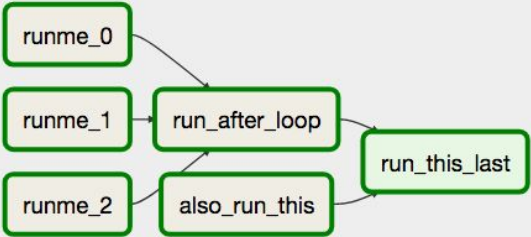
schedule: 0 0 \*\*\*

Graph ViewTree ViewTask DurationTask TriesLanding TimesGanttDetailsCodeRefreshDelete

successBase date: 2018-09-06 00:00:01Number of runs: 25Run: scheduled\_\_2018-09-06T00:00:00+00:00Layout: Left->RightGoSearch for...


BashOperatorDummyOperator

successrunningfailedskippedretryqueuedno status



```
graph LR; runme_0 --> run_after_loop; runme_1 --> run_after_loop; runme_2 --> also_run_this; run_after_loop --> run_this_last; also_run_this --> run_this_last;
```

# Airflow - Interface (Variáveis)

 Airflow

[DAGs](#)

[Data Profiling ▾](#)

[Browse ▾](#)

[Admin ▾](#)

[Docs ▾](#)

## Variables



















List (9)

Create

Add Filter ▾

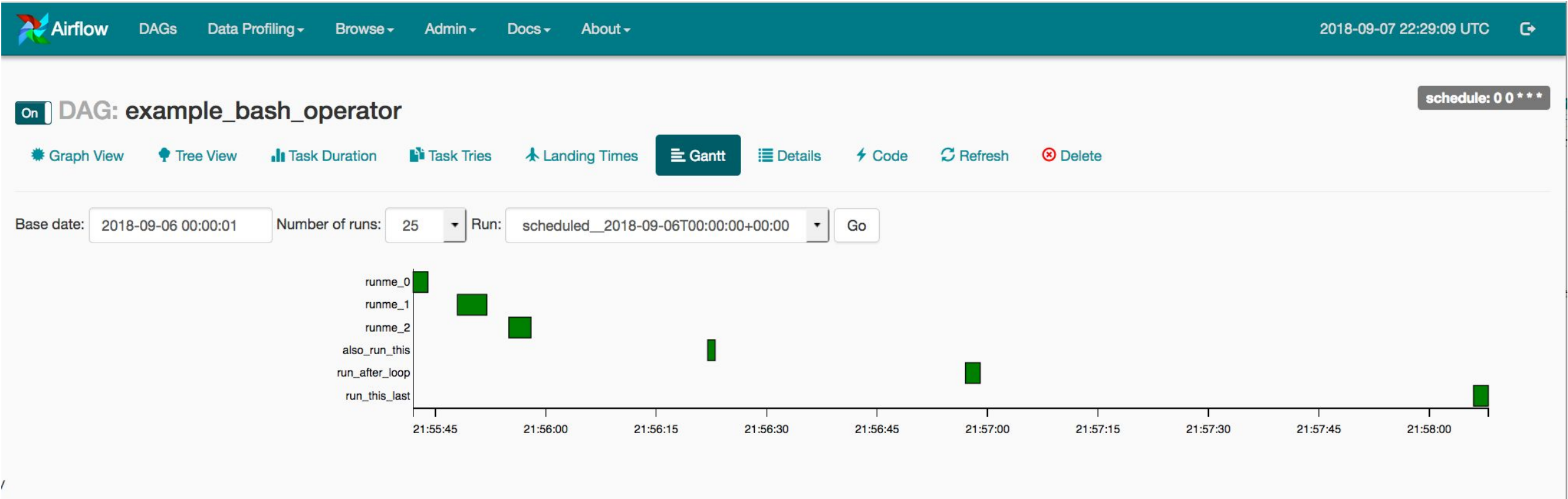
With selected ▾

Search

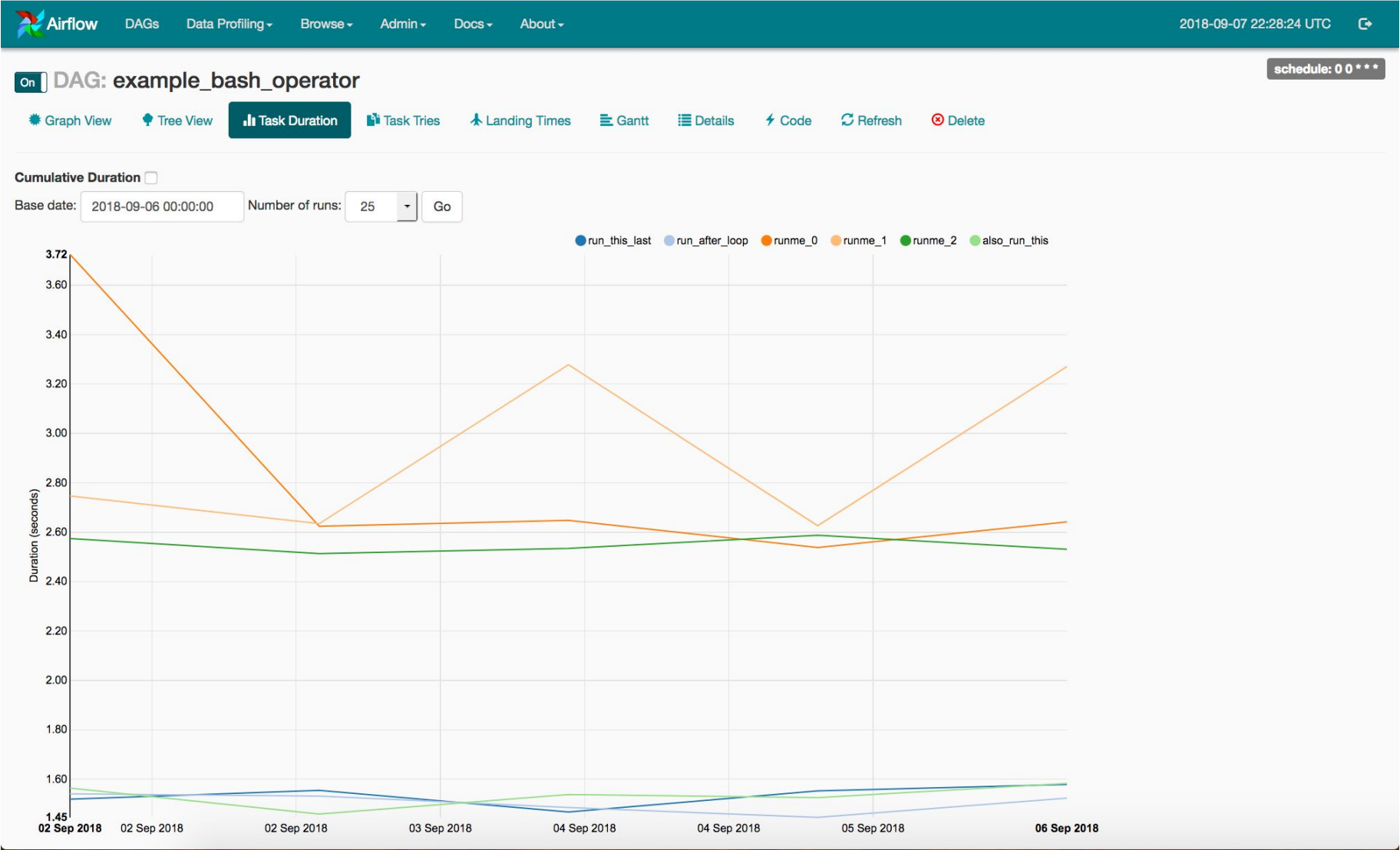
<input type="checkbox"/>		Key	Val
<input type="checkbox"/>	 	secret_password	*****
<input type="checkbox"/>	 	not_so_hidden	test value
<input type="checkbox"/>	 	secret	*****
<input type="checkbox"/>	 	password	*****
<input type="checkbox"/>	 	passwd	*****
<input type="checkbox"/>	 	api_key	*****
<input type="checkbox"/>	 	apikey	*****
<input type="checkbox"/>	 	authorization	*****
<input type="checkbox"/>	 	access_token	*****




# Airflow - Interface (Gráfico Gantt)



# Airflow - Interface (Duração das Tasks)



# Airflow - Interface (Código)

 Airflow

DAGs

Data Profiling ▾


Browse ▾

Admin ▾


Docs ▾


About ▾


2018-09-07 22:32:44 UTC





On **DAG: example\_bash\_operator** schedule: 0 0 \* \* \*


 Graph View


 Tree View


 Task Duration


 Task Tries


 Landing Times

 Gantt

 Details

 Code

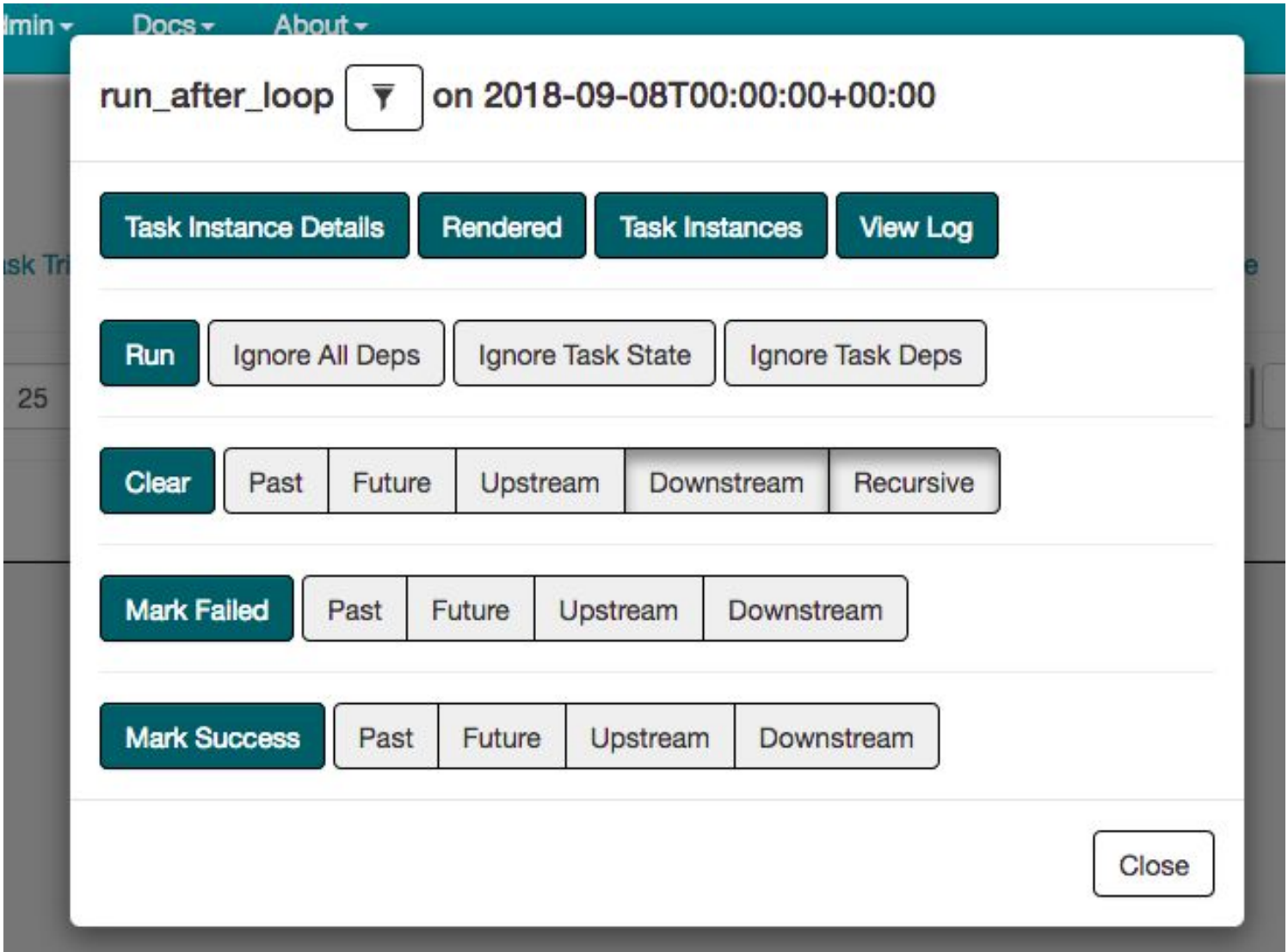
 Refresh

 Delete

example\_bash\_operator

```
1  # -*- coding: utf-8 -*-
2  #
3  # Licensed to the Apache Software Foundation (ASF) under one
4  # or more contributor license agreements. See the NOTICE file
5  # distributed with this work for additional information
6  # regarding copyright ownership. The ASF licenses this file
7  # to you under the Apache License, Version 2.0 (the
8  # "License"); you may not use this file except in compliance
9  # with the License. You may obtain a copy of the License at
10 #
11 # http://www.apache.org/licenses/LICENSE-2.0
12 #
13 # Unless required by applicable law or agreed to in writing,
14 # software distributed under the License is distributed on an
15 # "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
16 # KIND, either express or implied. See the License for the
17 # specific language governing permissions and limitations
18 # under the License.
19
20 import airflow
21 from builtins import range
22 from airflow.operators.bash_operator import BashOperator
23 from airflow.operators.dummy_operator import DummyOperator
24 from airflow.models import DAG
25 from datetime import timedelta
26
27
28 args = {
29     'owner': 'airflow',
30     'start_date': airflow.utils.dates.days_ago(2)
31 }
32
33 dag = DAG(
34     dag_id='example_bash_operator', default_args=args,
35     schedule_interval='0 0 * * *',
36     dagrun_timeout=timedelta(minutes=60))
37
38 cmd = 'ls -l'
39 run_this_last = DummyOperator(task_id='run_this_last', dag=dag)
40
41 # [START howto_operator_bash]
42 run_this = BashOperator(
43     task_id='run_after_loop', bash_command='echo 1', dag=dag)
44 # [END howto_operator_bash]
45 run_this.set_downstream(run_this_last)
46
```

# Airflow - Interface (Menu de Contexto da Task)

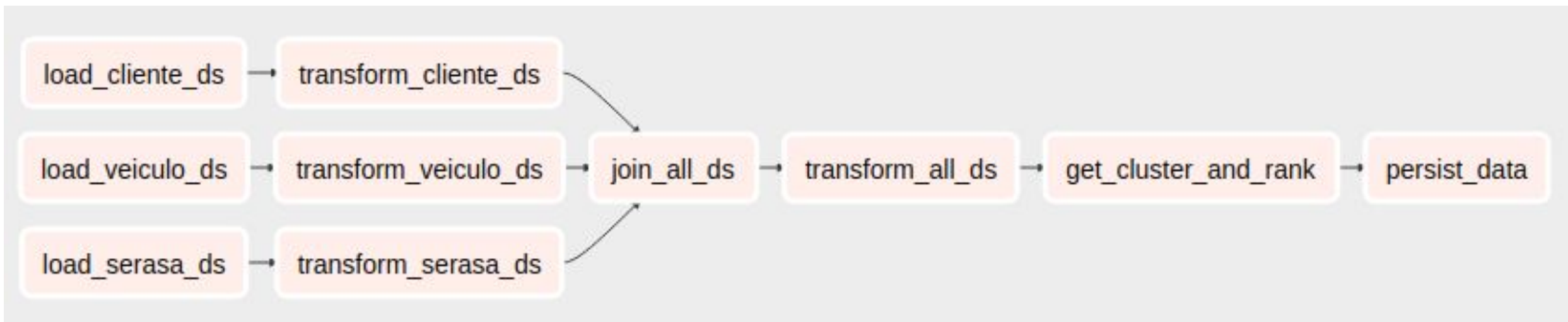


### Web Crawler

<https://www.infomoney.com.br/mercados/cambio>

# Airflow - DAG Rede Frota (Clusterização de Clientes)

---



## Airflow - “Contras”

---

- Implementado a nível de código;
- Realização de Testes;
- Não funciona para Streaming;

# Outras Ferramentas

---



**Luigi**



**Azkaban**



**Oozie**



**Spark Streaming**



**Data Factory**



**Cloud Composer**



**Glue**





**Podcast: Airflow in  
Practice with  
Chaim Turkel**

[softwareengineeringdaily.com](https://softwareengineeringdaily.com)



**Comunidade  
Brasileira de Ciência  
de Dados**

[datahackers.com.br](https://datahackers.com.br)



**Playlist  
Apache Airflow  
Tutorials**

Dúvidas?