Tempest

ACADEMY

Conference

2023

Pipelines de dados eficientes: transformando dados brutos em informações facilmente consumíveis

by José henrique Davino

Classificação da Informação: PÚBLICA
Autor da apresentação: José Henrique Davino

Tempest

:ACADEMY:

Conference

Quem sou?

- Um nordestino VIP - “Vindo do interior de Pernambuco”
- Na área de tecnologia desde 2003
- Mestrado em Ciência da Computação pela UFPE
- Desde 2017 atuando em projetos de Dados
- Na Tempest desde 2021
    - Senior Staff Data Engineer / Research Advisor

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Objetivo

Tempest
[ACADEMY]
Conference

Apresentar uma visão geral do que pode ser usado para lidar grandes volumes [muitas vezes complexos] de forma escalável e gerenciada.

Tempest

ACADEMY

Conference

01 Introdução

02 Benefícios e desafios quando se trabalha com dados

03 A natureza dos dados de cyber

04 O que o mercado oferece como solução para lidar com Big Data e o que escolhemos

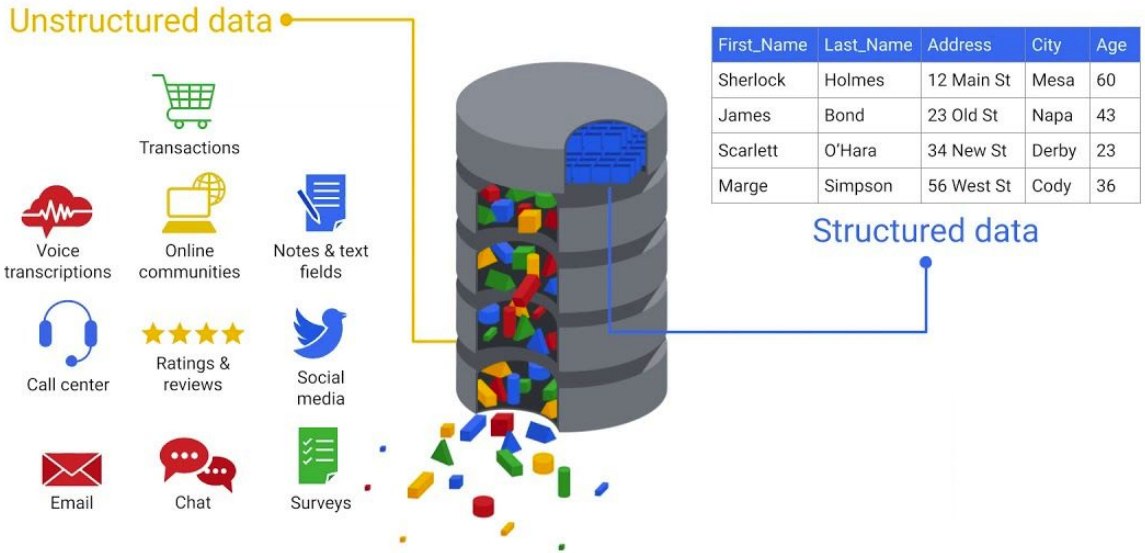
05 Considerações finais

O cenário atual

Tempest
ACADEMY
Conference

Um **volume enorme** de dados é **coletado** pelas empresas **todos os dias**, em **múltiplos canais** e em **diversos formatos**.

Em geral, as **empresas utilizam** diversos **softwares** para **obtê-los** e **mantê-los**, o que faz com que **se espalhem em silos**.



Tempest ACADEMY Conference

Empresas que buscam **organizar** e **integrar** seus **dados** para **auxiliar em decisões**, ampliam as chances de assertividade em suas estratégias de negócios.

Os benefícios são diversos

Tempest
[ACADEMY]
Conference

- Construir uma **fonte única da verdade**;
- **Comportamentos, tendências, padrões** e outras informações sobre nosso próprio negócio **podem ser descobertos**;
- Ser **guiados pelos dados** além do **conhecimento do especialista**;
- **Ampliar a assertividade** na criação de **novas estratégias** de negócios/serviços.

Desafios mais comuns

Tempest
[ACADEMY]
Conference

- As aplicações **fazem uso de tecnologias distintas**, inclusive de armazenamento;
- **Nem sempre dispõem de mecanismos** [APIs, por exemplo] **de acesso aos dados** para qualquer usuário;
- **A criação** de tais mecanismos **demandaria horas de desenvolvimento**;
- Dependendo da fonte, **pode não possuir um *esquema* definido**.

Dados de Cyber

Tempest
[ACADEMY]
Conference

Dados de Cyber

Tratar e disponibilizar dados dessa natureza é, muitas vezes, uma **atividade complexa** que **demanda tempo** e **recurso computacional**, principalmente quando o volume de dados é expressivo!

Quando **realizado de forma inadequada** o **ônus financeiro vem junto!** :(

Looker

Tempest
ACADEMY
Conference

Amundsen

Apache Beam

DataHub

Metabase

PREFECT

great expectations

tableau

Apache Airflow

SODA

elementary

dbt

Airbyte

python

rudderstack

Flink

APACHE Spark

Open Metadata

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Desafios e motivações que guiaram a escolha do conjunto de soluções

Tempest
[ACADEMY]
Conference

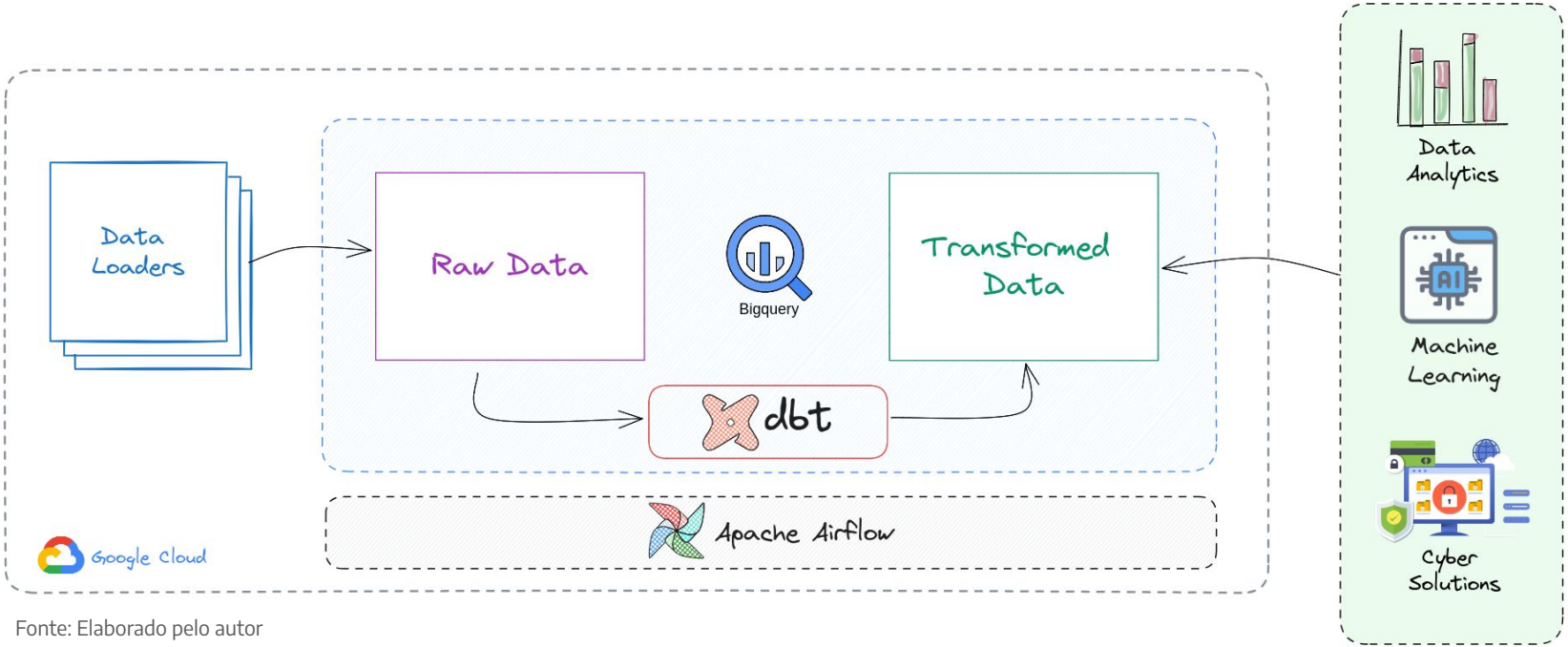
- Tornar os **dados disponíveis e úteis** para qualquer pessoa dentro de uma organização;
- **Processar** grandes volumes de dados **de forma recorrente**;
- **Curva de aprendizado** que fosse **a menor possível**;
- Que a **sintaxe de consulta** fosse **SQL**;
- Possibilidade **reuso de código** e **trabalho colaborativo**;
- **Algo utilizado** no mercado **por equipes de engenharia de dados**;
- **Menor custo financeiro possível**.

Tempest
[ACADEMY]
Conference

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Arquitetura de Dados - Visão simplificada

Tempest ACADEMY Conference



Fonte: Elaborado pelo autor

BigQuery

Tempest
ACADEMY
Conference

O BigQuery é um **serviço de armazenamento e análise de dados em nuvem**, totalmente **gerenciado pela Google Cloud** que **permite** às empresas **armazenar e analisar vastos volumes de dados** de maneira rápida e escalável.

Google BigQuery

O que é o DBT

Tempest
ACADEMY
Conference

É uma ferramenta de **transformação SQL** que viabiliza o **desenvolvimento de fluxos de trabalho analítico**, de forma **rápida** e **colaborativa**, seguindo **boas práticas de engenharia de software**, como: modularidade, reuso de código, versionamento de código, testes, entre outros.

dbt

DBT - Principais características

Tempest
ACADEMY
Conference

- SQL + Jinja
- Sources, Models, Macros
- Testes
- Documentação
- Snapshots, Seeds, Packages *

dbt

Como as coisas eram sem o DBT?

Tempest
[ACADEMY]
Conference

Simples e velho SQL

```sql
select
    order_id,
    sum(case when payment_method = 'bank_transfer' then amount end) as bank_transfer_amount,
    sum(case when payment_method = 'credit_card' then amount end) as credit_card_amount,
    sum(case when payment_method = 'gift_card' then amount end) as gift_card_amount,
    sum(amount) as total_amount
from app_data.payments
group by 1
```

Fonte: https://docs.getdbt.com/docs/build/jinja-macros

dbt

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Como são agora!

Tempest
[ACADEMY]
Conference

SQL + linguagem de modelagem Jinja.

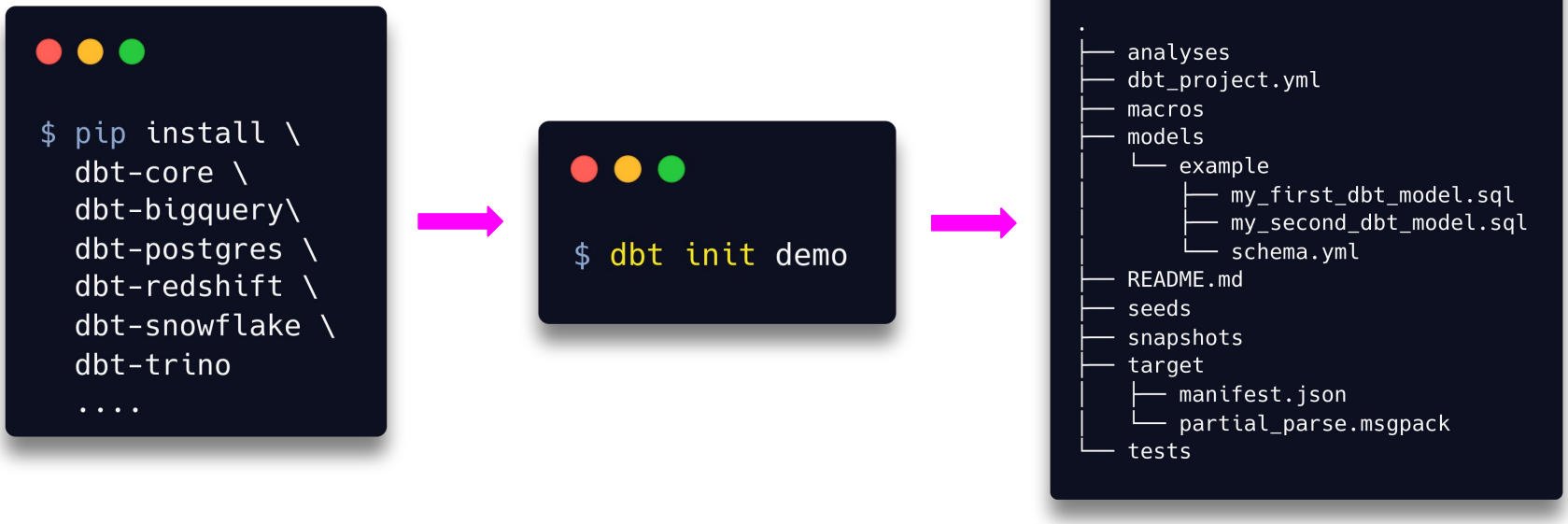
```
{% set payment_methods = ["bank_transfer", "credit_card", "gift_card"] %}

select
    order_id,
    {% for payment_method in payment_methods %}
    sum(case when payment_method = '{{payment_method}}' then amount end) as {{payment_method}}_amount,
    {% endfor %}
    sum(amount) as total_amount
from app_data.payments
group by 1
```

Fonte: https://docs.getdbt.com/docs/build/jinja-macros

dbt

Instalando e criando um projeto

Tempest
[ACADEMY]
Conference



Fonte: https://docs.getdbt.com/docs/core/pip-install

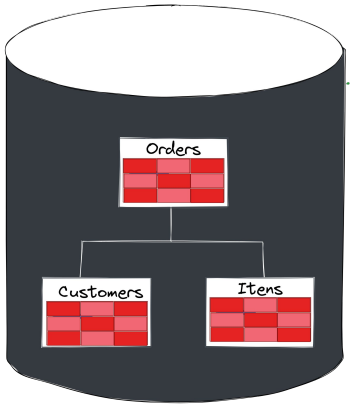
dbt

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Sources

Tempest
ACADEMY
Conference

```yaml
version: 2

sources:
  - name: sales
    tables:
      - name: orders
      - name: customers

  - name: billing
    tables:
      - name: payments
```



- Definições escritas em arquivo **.yml**;
- Usado para mapear entidades existentes (tabelas, views) em um base de dados.

dbt

Fonte: https://docs.getdbt.com/docs/build/sources

Sources

Tempest
ACADEMY
Conference

Sources podem ser referenciados a partir de um **model** usando a sintaxe jinja

model_a.sql

```
select
  ...

from {{ source('sales', 'orders') }}
```

model_a.sql

```
select
  ...

from sales.orders
```

Fonte: https://docs.getdbt.com/docs/build/sources

dbt

Models

Tempest
[ACADEMY]
Conference

- São ***queries*** construídas em um **arquivo .sql**
  - Seus **resultados são persistidos** no banco de dados **em forma de tabelas ou views**;
  - Cada **Model** deve conter apenas uma declaração SELECT;
  - O **nome do arquivo será o nome da entidade criada no banco de dados** (tabela/view).

model_a.sql

```sql
with custumer_orders as (
  SELECT
    custumer_id,
    MAX(order_datetime) as most_recent_order,
    COUNT(order_id) as number_of_orders
  FROM orders
)

SELECT
 c.custumer_id,
 c.first_name,
 c.last_name,
 co.most_recent_order,
 COALESCE(co.number_of_orders,0) as number_of_orders

FROM custumers c

LEFT JOIN custumer_orders co
ON co.custumer_id = c.custumer_id
```

Fonte: https://docs.getdbt.com/docs/build/models

Models

Tempest
ACADEMY
Conference

Podem ser referenciados a partir de outros **models** usando a sintaxe jinja + a função **ref()**

model_a.sql

```
select
  ...

from {{ ref('model_a') }}
```

model_a.sql

```
select
  ...

from sales.orders
```

Fonte: https://docs.getdbt.com/docs/build/models

dbt

Macros

Tempest ACADEMY Conference

- Trechos de **códigos reutilizáveis** definidos em um **arquivo.sql** dentro do diretório **macros**;
- São análogas a “funções” em outras linguagens de programação;
- Ajudam a manter o código mais limpo e mais fácil de entender.

```
{% macro generate_where(field_name, initial_date, final_date) %}
       {{ field_name }} >= DATE(initial_date)
   AND {{ field_name }} <= DATE(final_date)

  {% if target.name == 'dev' %}
    LIMIT 10
  {%- endif -%}

{% endmacro %}
```

model_a.sql

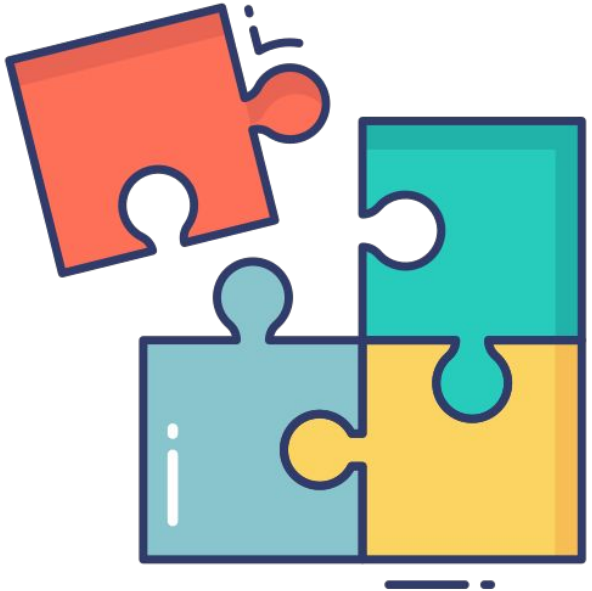
```
select
  ...
from {{ ref('model_a') }}

{{ generate_where("created_at", "2023-07-01", "2023-07-10") }}
```

Fonte: https://docs.getdbt.com/docs/build/jinja-macros

Modularidade

Tempest
ACADEMY
Conference

A abordagem modular do dbt permite dividir um pipeline complexo em partes menores.



Modularidade

Tempest
[ACADEMY]
Conference

Pode ser difícil manter e testar um **Model** contendo várias transformações de dados.

complex_model.sql

```
WITH stations AS (
  SELECT station_id, council_district,
    CASE
    WHEN property_type IN ('parkland', 'sidewalk') THEN 'free_parking'
    ELSE property_type
    END AS property_type,
  FROM `bigquery-public-data.austin_bikeshare.bikeshare_stations`
  WHERE
    property_type IN ('parkland','sidewalk','paid_parking')
),
trips AS (
  SELECT start_station_id
  FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
  WHERE
    start_station_id is NOT NULL
)
SELECT
  stations.property_type,
  COUNT(*) AS trips,
FROM trips
JOIN stations ON trips.start_station_id = stations.station_id
GROUP BY
  stations.property_type
```

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Modularidade

Tempest
ACADEMY
Conference

complex_model.sql

```sql
WITH stations AS (

  select * from {{ref('model_stations')}}
),
trips AS (

  select * from {{ref('model_trips')}}
)
SELECT
  stations.property_type,
  COUNT(*) AS trips,
FROM trips
JOIN stations ON trips.start_station_id = stations.station_id
GROUP BY
  stations.property_type
```

model_stations.sql

```
...
```

model_trips.sql

```
...
```

dbt

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Modularidade

Viabiliza o reuso

Tempest ACADEMY Conference

model_stations.sql

```sql
SELECT station_id, council_district,
    CASE
    WHEN property_type IN (...) THEN 'free_parking'
    ELSE property_type
    END AS property_type,
  FROM `bigquery-public-data.austin_bikeshare.bikeshare_stations`
  WHERE
    property_type IN (...)
```

stations_per_district.sql

```sql
SELECT
  property_type,
  council_district,
  count(*) total_station
FROM
  {{ref('model_stations')}}
GROUP BY
  property_type,
  council_district
```

dbt

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Modularidade

Viabiliza o trabalho em equipe simultaneamente

Tempest
ACADEMY
Conference

models/model_stations.sql

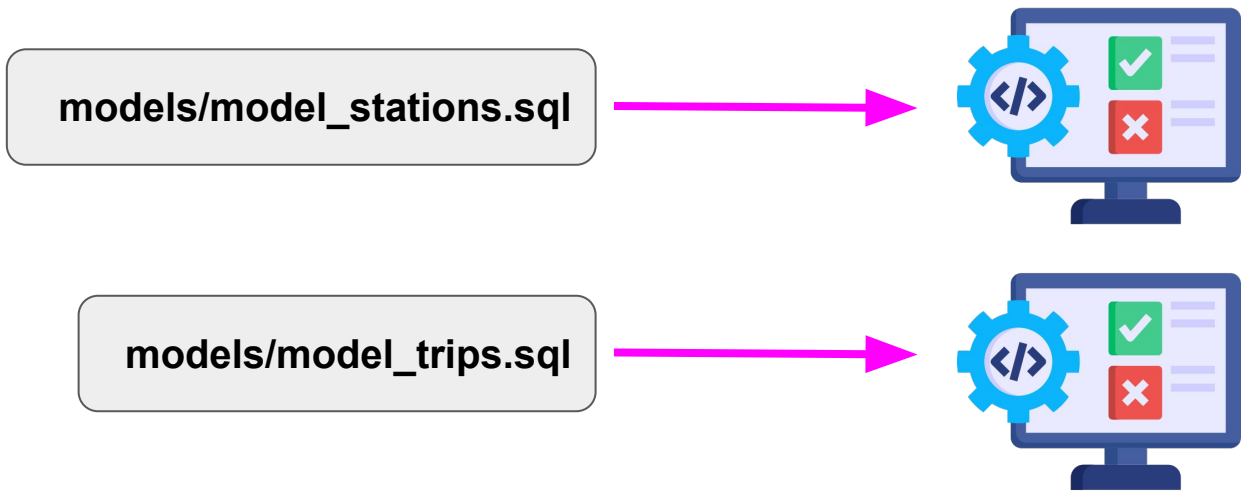
models/model_trips.sql

dbt

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Modularidade

Possibilita testar e validar um **Model** de forma independente



Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Dependência

Onde a mágica acontece

complex_model.sql

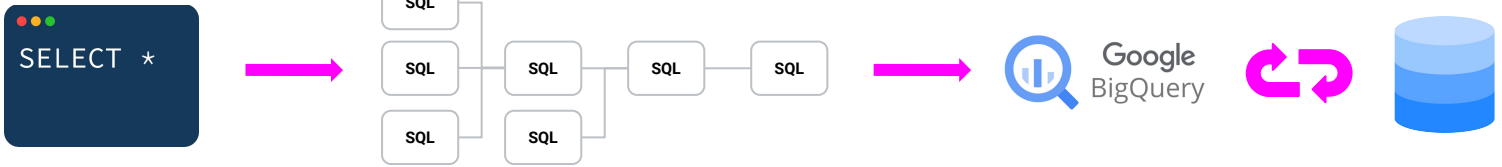
```
WITH stations AS (

  select * from {{ref('model_stations')}}
),
trips AS (

  select * from {{ref('model_trips')}}
)
SELECT
  stations.property_type,
  COUNT(*) AS trips,
FROM trips
JOIN stations ON trips.start_station_id = stations.station_id
GROUP BY
  stations.property_type
```

model_stations.sql

...

model_trips.sql

...



Compilação em SQL

Ocorre em tempo de execução. Resolvendo dependências e checando erros.

Executa no mecanismo de processamento distribuído

Todas as transformações ocorrem diretamente no **BigQuery**, rodando comandos SQL.

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

Testes

Permite checar a corretude de um Model

```yaml
version: 2

models:
  - name: stg_stations
    columns:
    - name: station_id
      tests:
        - not_null  # Should not contain null values.
        - unique    # Should be unique
    - name: property_type
      tests:
        # Should be either 'free_parking' or 'paid_parking'
        - accepted_values:
            values: ['free_parking', 'paid_parking']
```

Ajudam a rastrear alterações nos dados ao longo do tempo e garantem que a lógica de transformação permaneça correta à medida que os dados subjacentes mudam!

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

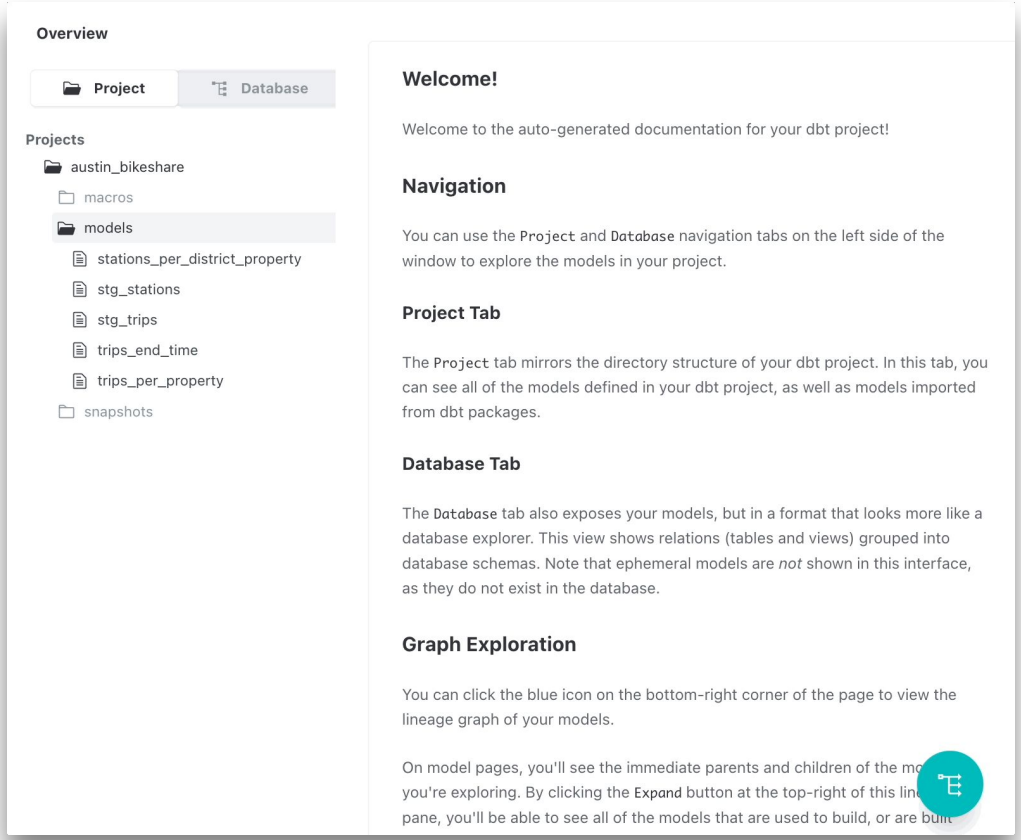
DBT - Documentação

Tempest
[ACADEMY]
Conference

É possível documentar os modelos e compartilhar a documentação com a equipe.

```yaml
version: 2

models:
  - name: stg_stations
    descriotion: A model station data
    columns:
    - name: station_id
      description: A unique identifier for each station
      tests:
        - not_null  # Should not contain null values.
        - unique    # Should be unique
    - name: property_type
      tests:
        # Should be either 'free_parking' or 'paid_parking'
        - accepted_values:
            values: ['free_parking', 'paid_parking']
```

DBT - Documentação



A documentação pode ser acessada via interface web:

1. *dbt docs generate* para gerar a documentação do projeto;
2. *dbt docs serve* para iniciar um host local com a documentação;
3. *http://localhost:8080* via navegador para acessar o conteúdo.

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino

DBT - Como executá-lo na prática?

Tempest
[ACADEMY]
Conference

Command line interface (CLI)

- **dbt compile** - Compila (mas não executa) os modelos em um projeto;
- **dbt run** - Executa os modelos definidos em um projeto;
- **dbt build** - Constrói e testa os resources [models, testes, etc];
- **dbt test** - Executa os testes.

Ferramentas de orquestração (airflow).

- **DbtRunOperator**
- **DbtTestOperator**

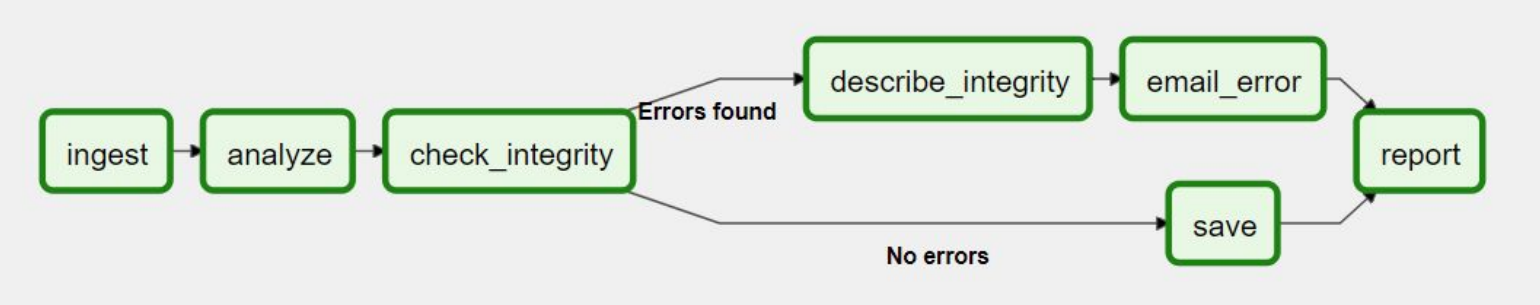
Fonte: https://docs.getdbt.com/reference/dbt-commands

dbt

O que é o Apache Airflow?

Tempest ACADEMY Conference

É uma plataforma de código aberto para **desenvolver, agendar e monitorar fluxos de trabalho,** baseados em lote (batch). Usa o conceito de **Gráficos acíclicos direcionados** (Directed Acyclic Graphs, DAG) para gerenciar o fluxo da execução de um conjunto de tarefas/ações.



Apache Airflow

Por que Apache Airflow?

Tempest
[ACADEMY]
Conference

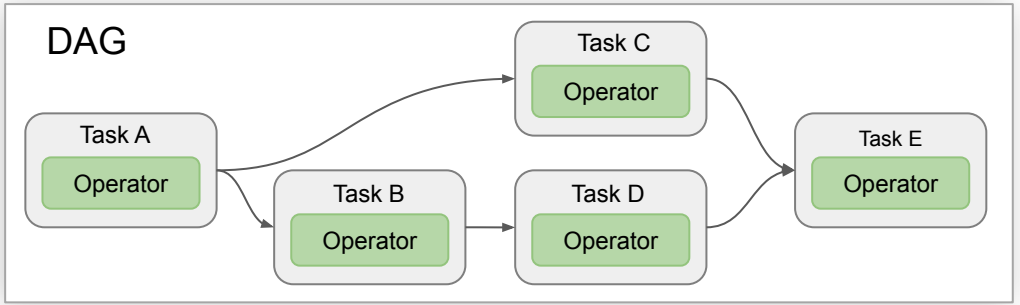
- **Fácil de operar** - Básico de Python é o único requisito;
- **Lista de Operadores -** Suporta uma infinidade de serviços (cloud, infraestrutura, banco e outros);
- **Extensível -** Permite criar operadores customizados para uma necessidade específica;
- Segue a ideia de "**dividir para conquistar**"

Apache Airflow

Airflow - Características Principais

Tempest ACADEMY Conference

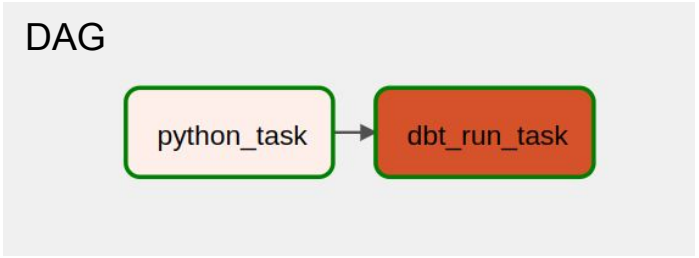
- **DAG** - Espécie de “grafo” que pode possuir caminhos paralelos vindos de um ponto de origem.
- **Task** - Determina como e quando executar um **Operator** dentro do contexto de uma DAG.
- **Operator** - É um tipo de “ template” que nos permite executar uma determinada atividade.
    - **PythonOperator**: executa código Python
    - **DbtRunOperator**: executa o comando “dbt run”
    - **PostgresOperator**: realiza consultas SQL no PostgreSQL



Exemplo de DAG com DBT

Tempest
ACADEMY
Conference

```python
from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow_dbt.operators.dbt_operator import DbtRunOperator
from datetime import datetime

def print_message():
    print("Hello Tempest")

with DAG(
    dag_id="dag_exemplo",
    start_date=datetime(2023, 2, 1),
    schedule_interval = "0 * * * *",  #@daily
) as dag:

    task1 = PythonOperator(
        task_id='python_task',
        python_callable=print_message,
    )

    task2 = DbtRunOperator(
        task_id='dbt_run_task',
        select='my_model_name'
    )

    task1 >> task2  #execution order
```



Apache Airflow

Orquestrando a execução

Tempest ACADEMY Conference

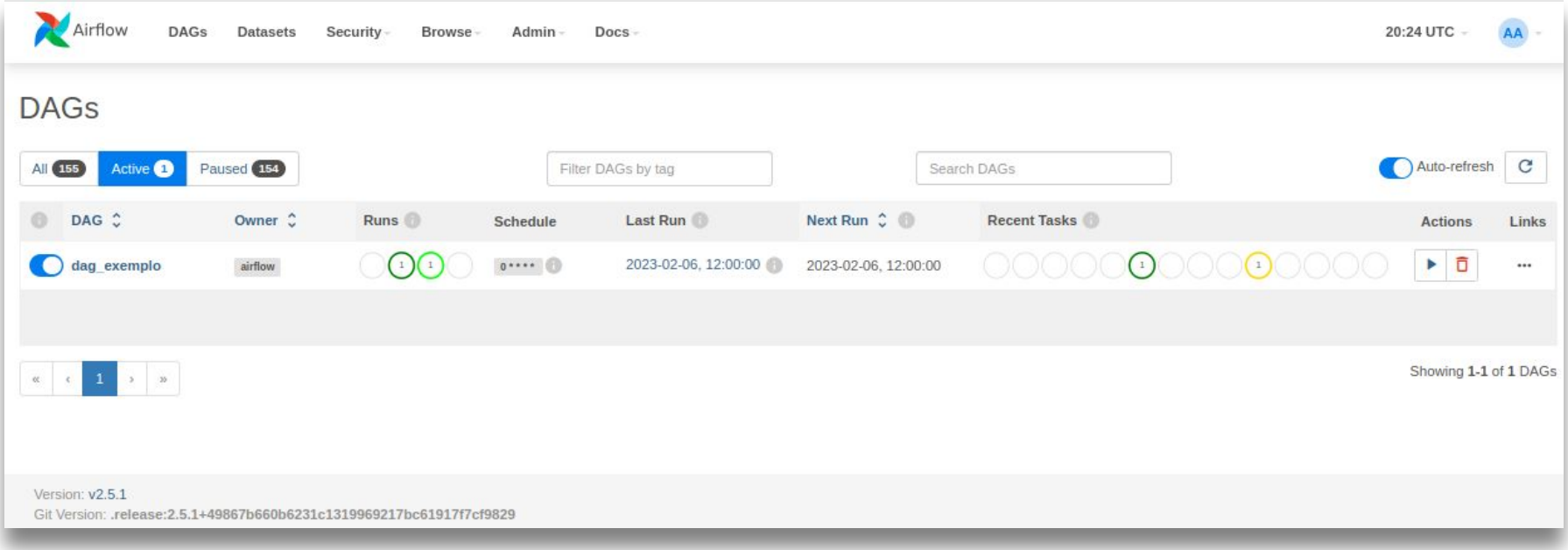
- Definir intervalos de execuções;
- Número de tentativas em casos de falhas;
- *Timeout* para finalizar execuções;
- Notificações em casos de falhas;
- Inclusão de sensores para checagens.

```
(...)

with DAG(
dag_id="dag_exemplo",
    start_date=datetime(2023, 2, 1),
    schedule_interval = "0 * * * *",  #@daily
    default_args = {'retries': 2,'retry_delay': timedelta(minutes=1)},
    max_active_runs = 1
) as dag:

    (...)

    task2 = DbtRunOperator(
        task_id='dbt_run_task',
        select='my_model_name'
        execution_timeout=timedelta(minutes=15),
    )
```

Apache Airflow

UI do airflow

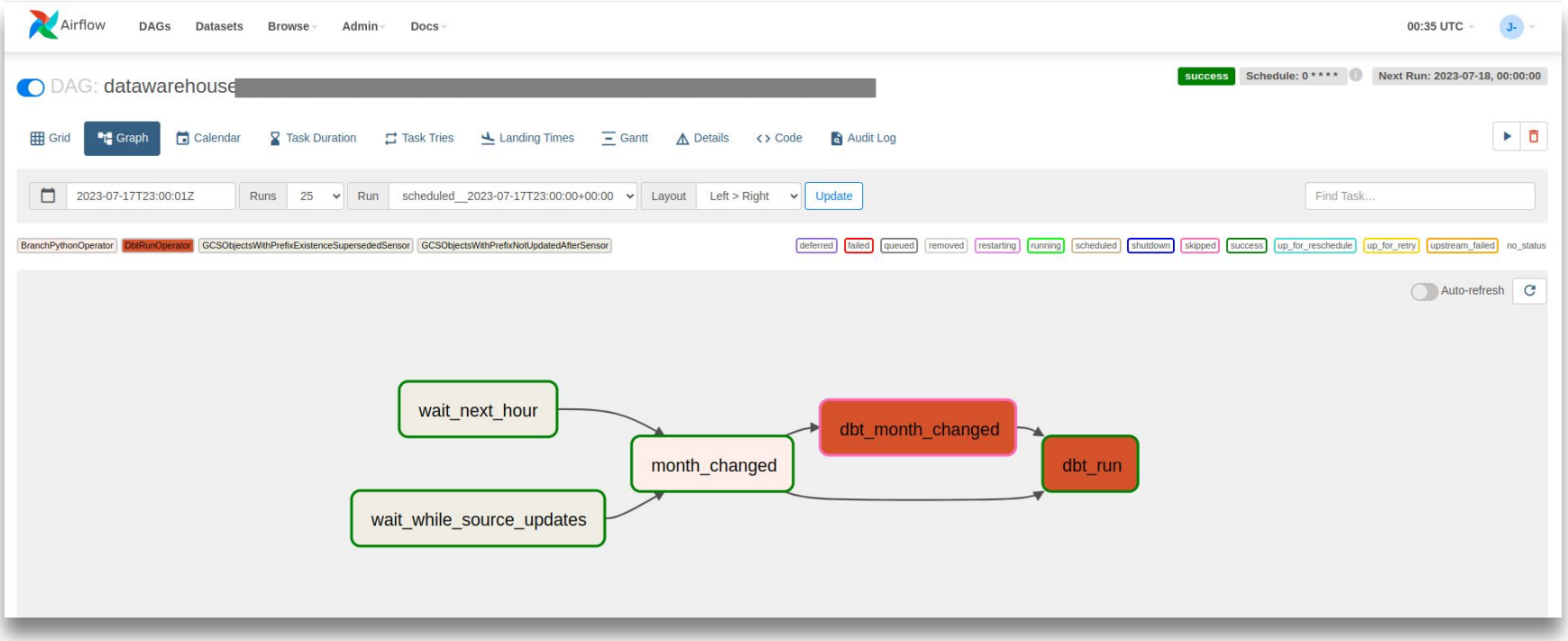
Tempest ACADEMY Conference



Apache Airflow

UI do airflow

Tempest ACADEMY Conference

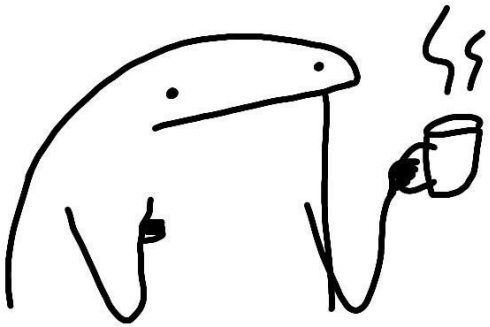


Considerações Finais

Tempest
[ACADEMY]
Conference

- **BigQuery + Dbt + airflow** mostrou-se uma combinação perfeita para **Pipelines de dados**.
    - **Boas práticas de engenharia de software** aplicadas no mundo de Analytics.
    - Dados limpos e normalizados **entregues de forma gerenciada** com a ajuda do **Airflow**.
    - **Terabytes de dados processados** nessa infraestrutura.
        - Algumas fontes chegando a processar **~ 1.3 milhões de registros** por hora.
- Grande variedade de **produtos e serviços** consumindo esses dados.
- Diversas soluções de **Machine learning**.
- Diversos **Dashboards**.

Tempest
[ACADEMY]
Conference



É isso, gente!

Espero que tenham curtido!

jose.davino@tempest.com.br
/in/jose-henrique-davino

Tempest

:ACADEMY:

Conference

2023

Classificação da Informação: PÚBLICA Autor da apresentação: José Henrique Davino