



Coordenação de Inovação e Ciência de Dados da
Coordenação-Geral de Gestão da Informação da
Diretoria de Informações, Serviços e Sistemas de Gestão
Secretaria de Gestão e Inovação



História do Apache Airflow?

O Airflow começou na Airbnb em 2014, inicialmente como uma ferramenta interna para orquestrar workflows. Em 2015, foi liberado como open-source e rapidamente ganhou popularidade. Em 2016, foi doado à Apache Software Foundation, tornando-se um projeto Apache. O Airflow 2.0, lançado em 2020, trouxe grandes melhorias, como escalabilidade, integração com Kubernetes e melhorias na UI. Hoje, o Apache Airflow é uma das ferramentas mais populares para orquestração de workflows e automação de pipelines de dados.

O que é Apache Airflow?



O Apache Airflow é uma plataforma ***open source*** de orquestração de ***workflows*** (fluxos de trabalho) usada para criar, agendar e monitorar pipelines de dados e processos automatizados com:

- Flexibilidade
- Reutilização
- Gerenciamento facilitado
- Múltiplas tarefas em sequências complexas

Características do Apache Airflow?

- 1) Desenvolvimento completamente em Python
- 2) Interface gráfica e linha de comando
- 3) API
- 4) Escalável e distribuível
- 5) Suporte para vários SGBDs, APIs e serviços de cloud

Princípios do Apache Airflow?



1. Escalável (arquitetura modular): Ao infinito
2. Dinâmico (permite geração dinâmica de pipelines)
3. Extensível (é possível criar operadores próprios)
4. Elegante (pipelines enxutos e implícitos)
5. Idempotência (uma tarefa pode ser executada várias vezes sem efeitos colaterais)



Pra que serve o Apache Airflow?

1. Automação em geral
2. Automação de pipelines de dados
3. ETL/ELT
4. Orquestração de processos
5. Machine Learning
6. Integração entre sistemas
7. Monitoramento e Alertas



Exemplos Práticos

Exemplos coletados pelo ChatGPT:

- Caso: Coletar dados de uma API externa, transformar os valores e armazená-los no PostgreSQL diariamente às 3h da manhã
- Caso: Treinar um modelo de previsão de vendas diariamente, validar métricas e salvar o modelo no S3.
- Caso: Verificar periodicamente o status de uma API, registrar logs e enviar alerta no Slack em caso de falha.
- Caso: Rodar testes automatizados de uma aplicação web toda madrugada e gerar relatórios.
- Caso: Baixar tabelas do IBGE, converter para formato parquet e armazenar no Data Lake.
- Caso: Executar um web scraper que coleta preços de concorrentes diariamente.

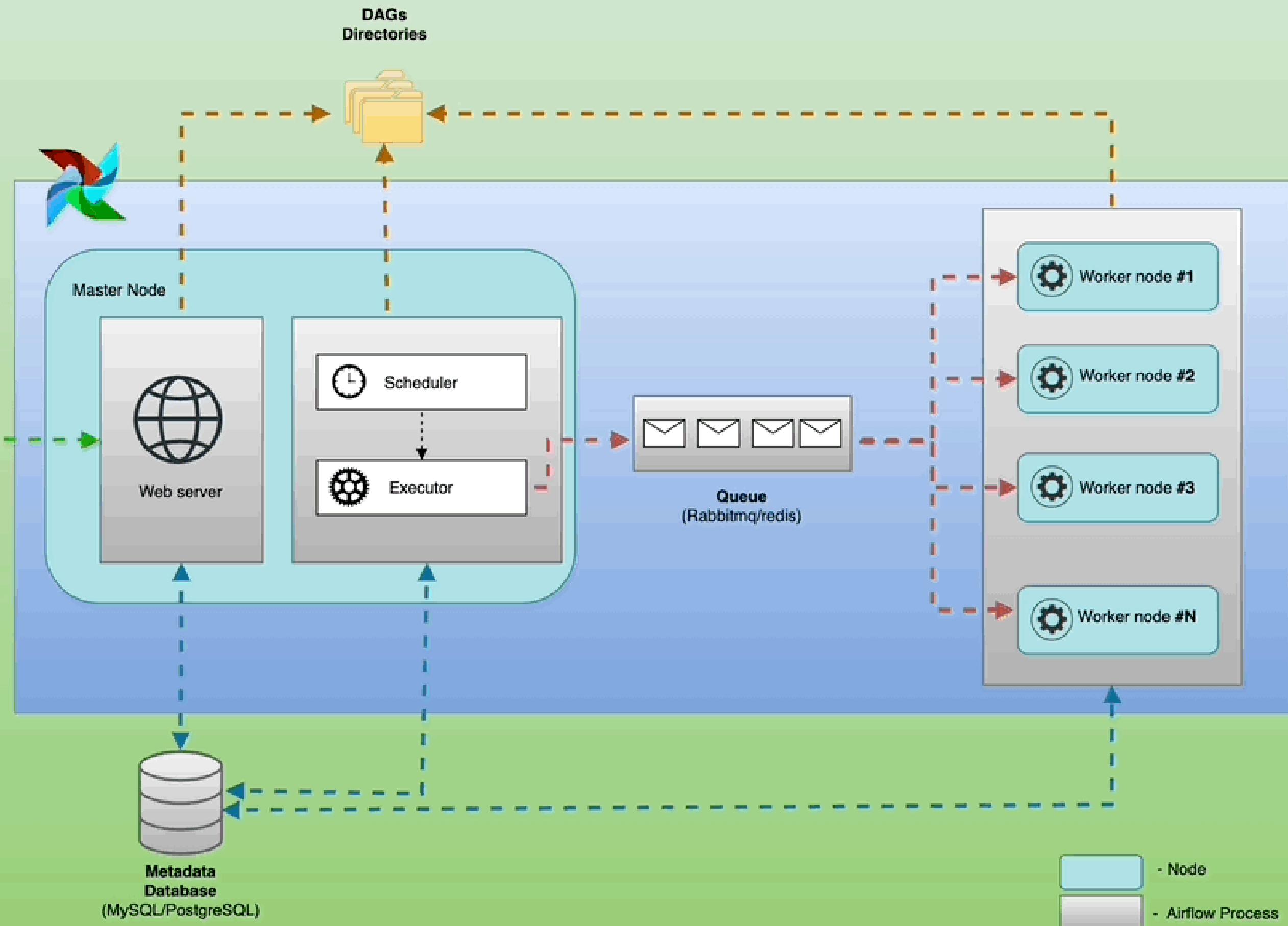


Como instalar

- Instalação via pip install
- Instalação via Docker
 - Imagem da CGINF
- Instalação via Kubernetes (Helm)
- Serviço em cloud
 - AWS Managed Workflows for Apache Airflow (MWAA)
 - Astronomer Cloud
 - Azure Data Factory
 - Google Cloud Composer



Architecture



Integrações

- AWS DynamoDB,
- Apache Hive,
- AWS S3,
- Apache Sqoop,
- Machine Learning Engine,
- Hadoop HDFS,
- Cassandra,
- AutoML,
- Redshift,
- Azure,
- Apache Spark,
- Apache Pig,
- Amazon EC2,
- Google Spreadsheet,
- MongoDB,
- MySQL,
- Docker,
- Microsoft SQL Server,
- HTTP,
- Databricks,
- PostgreSQL,
- Google Drive,
- JDBC,
- Oracle,
- SQLite,
- Kubernetes,
- SMTP,
- etc

Em quais casos o Airflow não é uma boa opção:

- Processamento de Dados em Tempo Real:
 - O Airflow não foi projetado para processar eventos em tempo real ou fluxos contínuos de dados.
 - Se você precisa de respostas em milissegundos, soluções como Apache Kafka, Apache Flink ou Spark Streaming são mais adequadas.
- Gerenciamento de dados em larga escala:
 - O Airflow pode orquestrar pipelines de dados, mas não é uma ferramenta ETL por si só.
 - Para transformações pesadas de dados, o ideal é usar dbt, Apache Spark, Airbyte ou ferramentas especializadas em ETL.

Em quais casos o Airflow não é uma boa opção:

- Workflows Simples e Únicos:
 - Se você tem um workflow muito simples ou que será executado apenas uma vez, configurar o Airflow pode ser excessivo. Scripts simples ou ferramentas de automação como cron podem ser mais apropriados.
- Workflows com Alta Frequência de Execução:
 - O Airflow pode não ser a melhor escolha para workflows que precisam ser executados em intervalos muito curtos (segundos ou minutos), devido à sobrecarga de agendamento e coordenação.

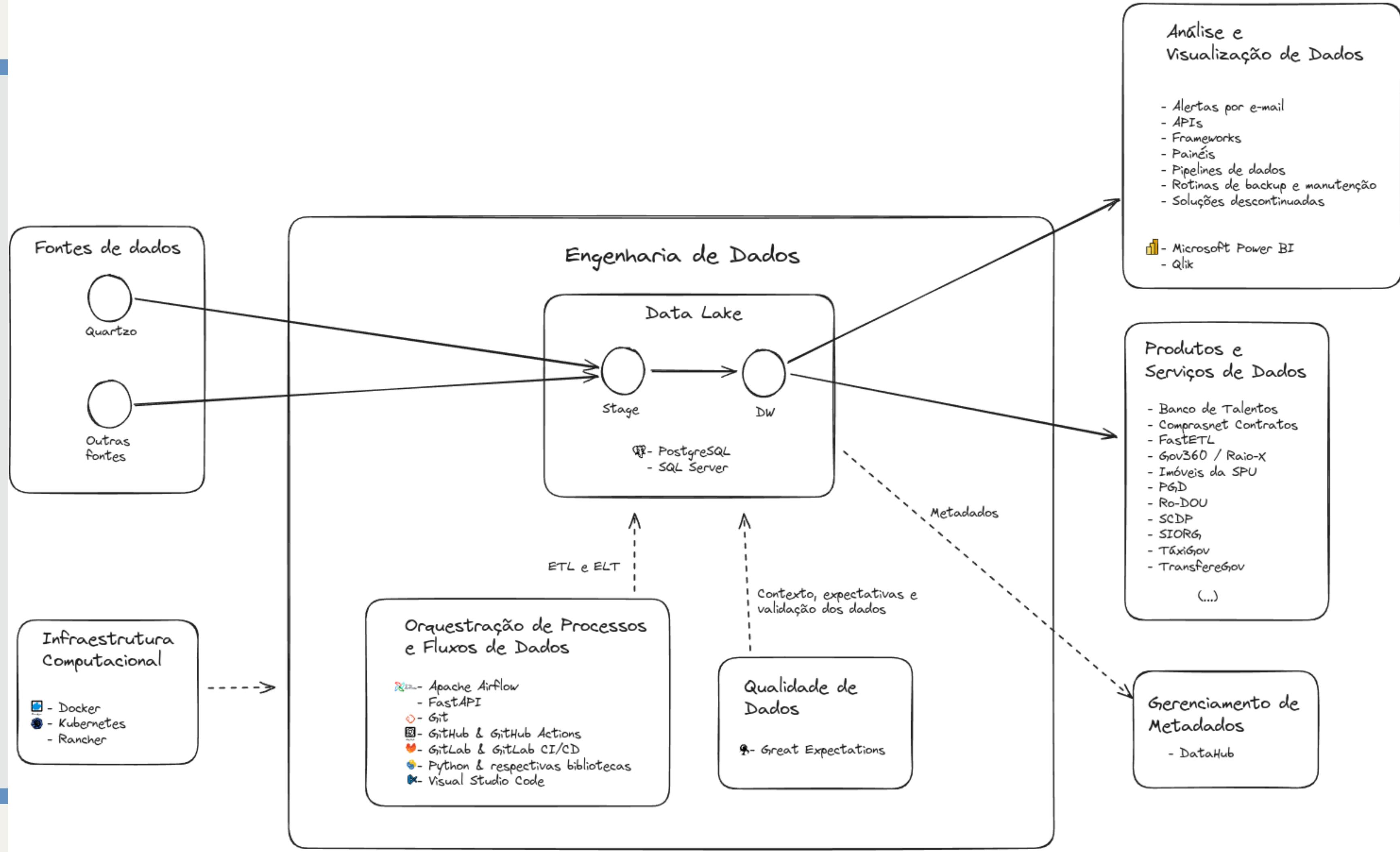
Para que utilizamos o Apache Airflow na CDATA/CGINF?

- 1) Engenharia de dados (dados brutos, data lake e data warehouse, painéis)
- 2) Publicação de dados abertos no dadosgov.br
- 3) Envio de clippings do diário oficial
- 4) Coleta de dados via sharepoint, google e APIs
- 5) Notificações e extrações de dados via e-mail, slack e discord
- 6) Automação dos relatórios de sprint da equipe
- 7) Monitoramento de infra

Contexto

- Cerca de 124 DAGs ativas





O que são DAGs?

- Uma DAG (Directed Acyclic Graph) é a estrutura fundamental do Airflow para **representar um fluxo de trabalho**. Ela define um conjunto de tarefas e suas dependências, organizadas de forma:

Direcionada: As tarefas têm uma ordem específica de execução.

Acíclica: **Não há loops ou ciclos**, ou seja, uma tarefa não pode depender de si mesma. Assim, o Airflow garante que as dependências sejam respeitadas.

O que são tasks?

- Uma task (tarefa) é a unidade básica de execução dentro de uma DAG. Juntas, elas formam a lógica do fluxo de uma DAG.

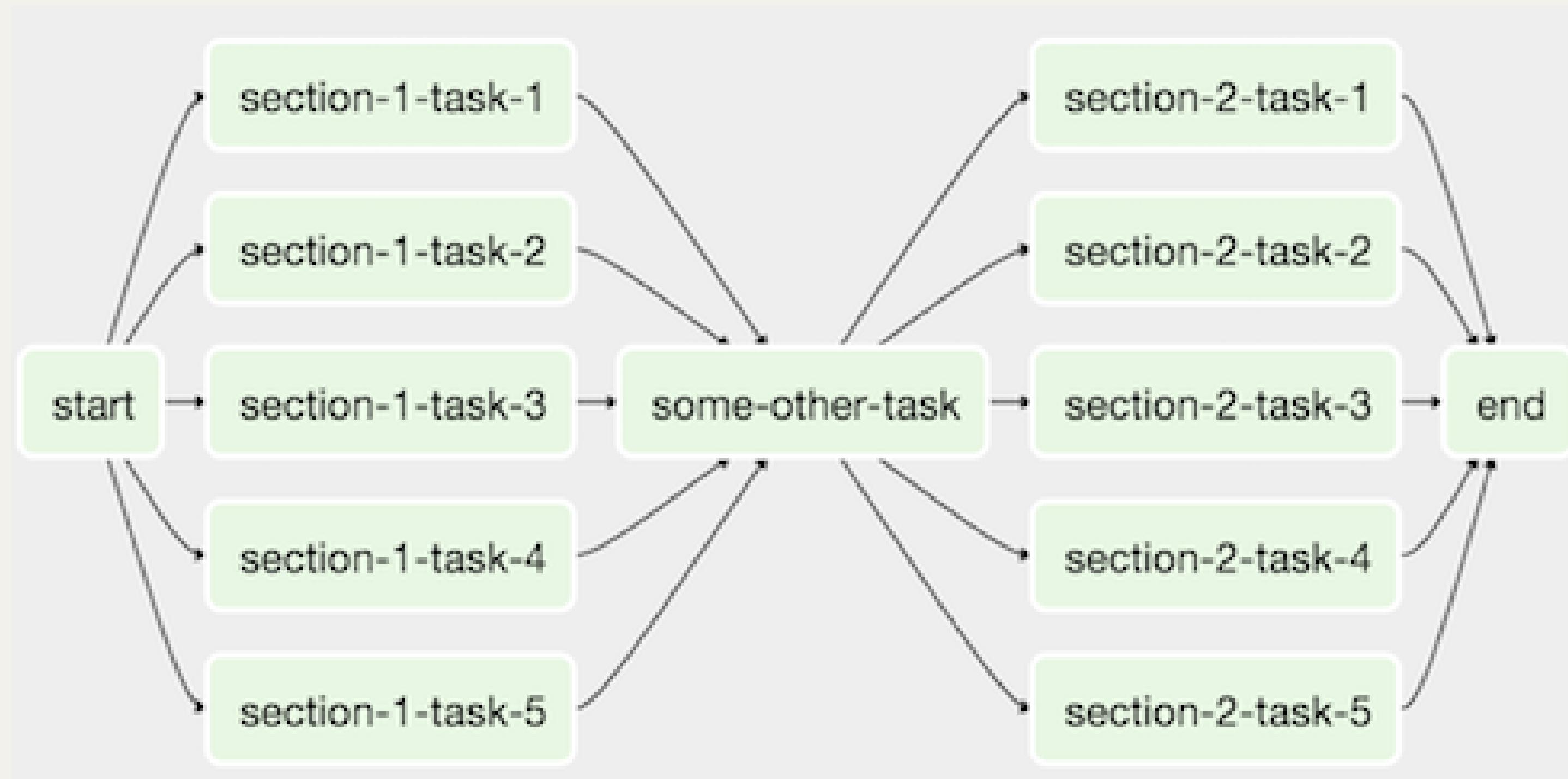
Autonomia

Tolerância a falhas

Dependência

**Execução síncrona ou
assíncrona**

Orquestração



O que são operators?

- Um operator é um template reutilizável que define o que uma task deve executar de forma encapsulada.

Exemplos:

Python Operator

Bash Operator

MSSQL Operator

Postgres operator

Branch Operator

etc

O que são hooks?

- Um hook é um componente usado para conectar a serviços externos, como banco de dados, API, entre outros.
- Muitos operators utilizam hooks para realizar operações em serviços conectados.

O que é FastETL?

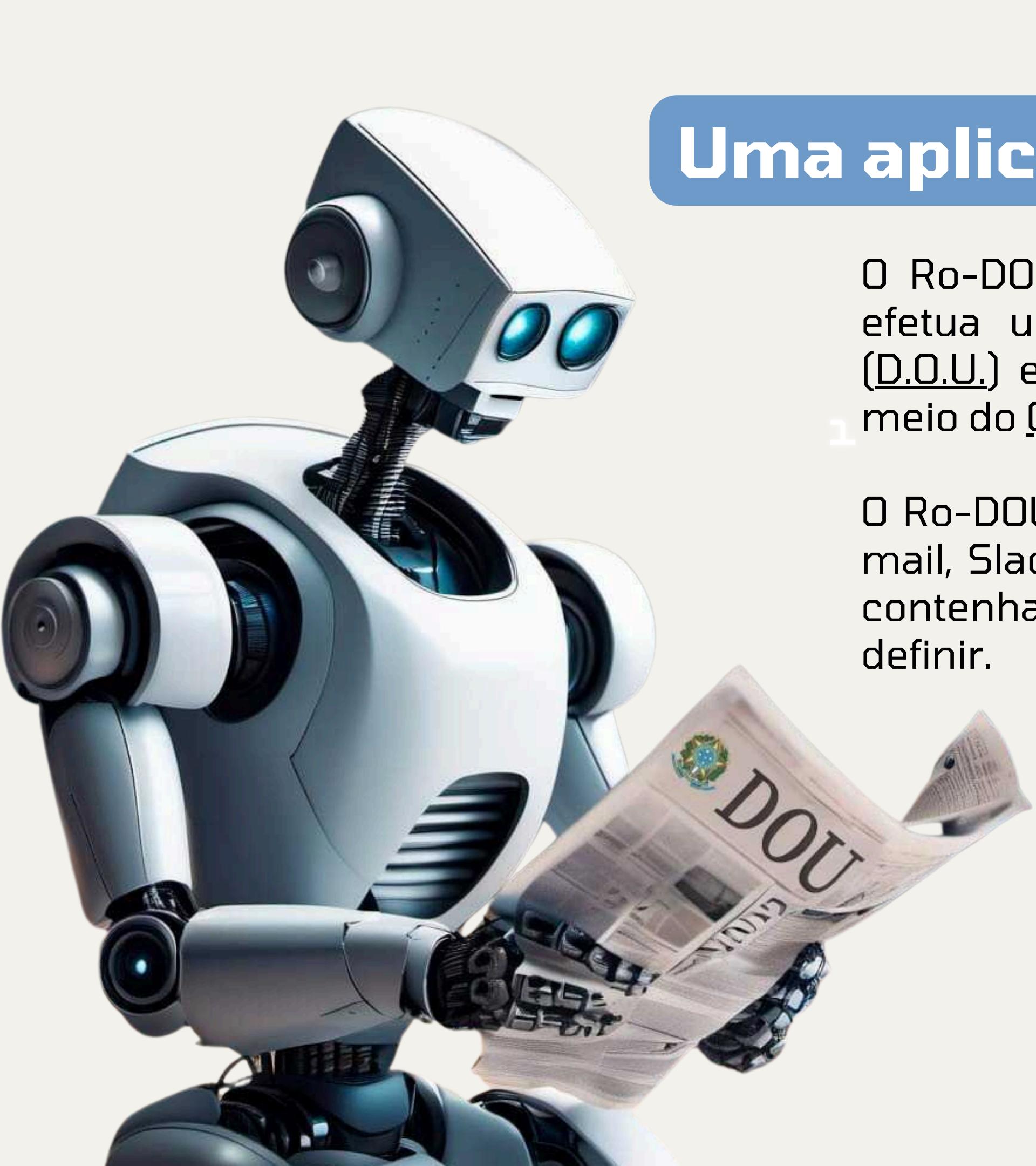


- Um framework escrito em python contendo pacotes de plugins do airflow para construção de pipelines para diversos cenários.
- Livre e open source
- Histórico: A partir de 2019 por desenvolvedores do ME durante a criação do datalake e adoção do Airflow em substituição ao SQL Server Integration Services (SSIS).

Principais funcionalidades do FastELT



- Replicação de tabelas entre SGBDs diferentes (Postgres/MSSQL/Mysql)
- Carga de dados a partir do GSheets e rede Samba Extração de CSV a partir de SQLs
- Requisições em APIs
- OSRM (Open Street Routing Machine) para calcular rotas e distâncias
- API de dados abertos ([link dadosgovbr](#))



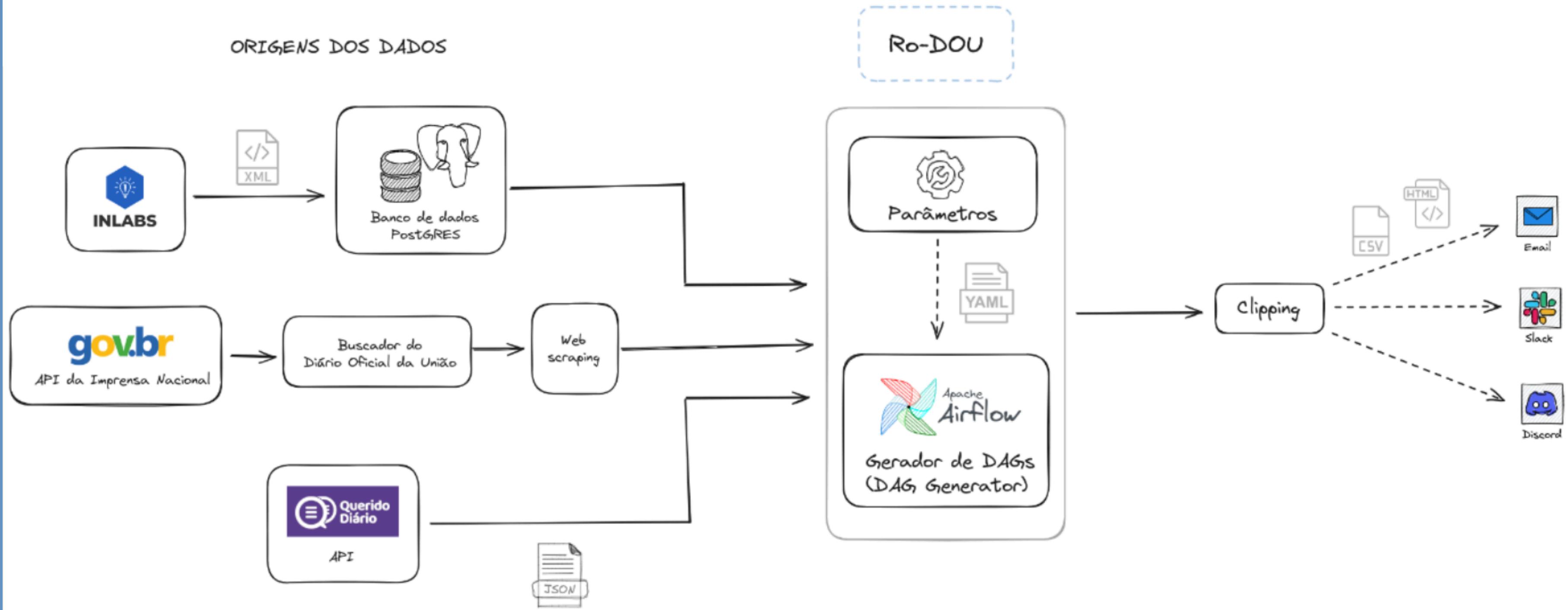
Uma aplicação do Airflow: Ro-dou

O Ro-DOU é uma **ferramenta automática** que efetua um clipping do Diário Oficial da União (D.O.U.) e dos Diários Oficiais de municípios, por meio do Querido Diário.

O Ro-DOU permite o envio de notificações (via e-mail, Slack, Discord) de todas as publicações que contenham as palavras-chave que o usuário definir.

O Ro-DOU opera por meio da ferramenta de orquestração Apache Airflow.

FLUXO DO RO-DOU



Onde aprender:

- 1) airflow.apache.org (documentação oficial)
- 2) academy.astronomer.io (Astronomer Academy)
- 3) Udemy Certificação

LINKS IMPORTANTES



gov.br/rodou

Apache Airflow -
CDATA



[https://github.com/gestaogovbr/airflow
2-docker/](https://github.com/gestaogovbr/airflow-2-docker/)

FastETL



<https://github.com/gestaogovbr/FastETL>

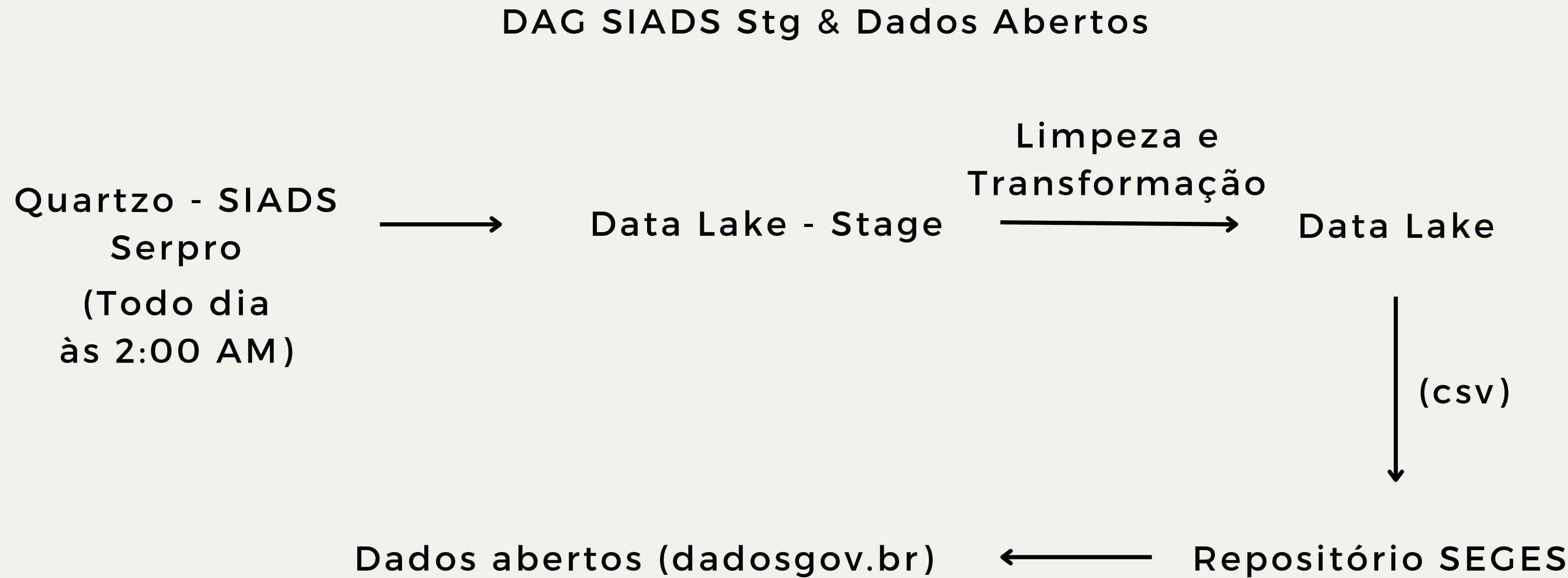
Ro-dou



<https://github.com/gestaogovbr/Ro-dou>

VISITE A PÁGINA DO RO-DOU

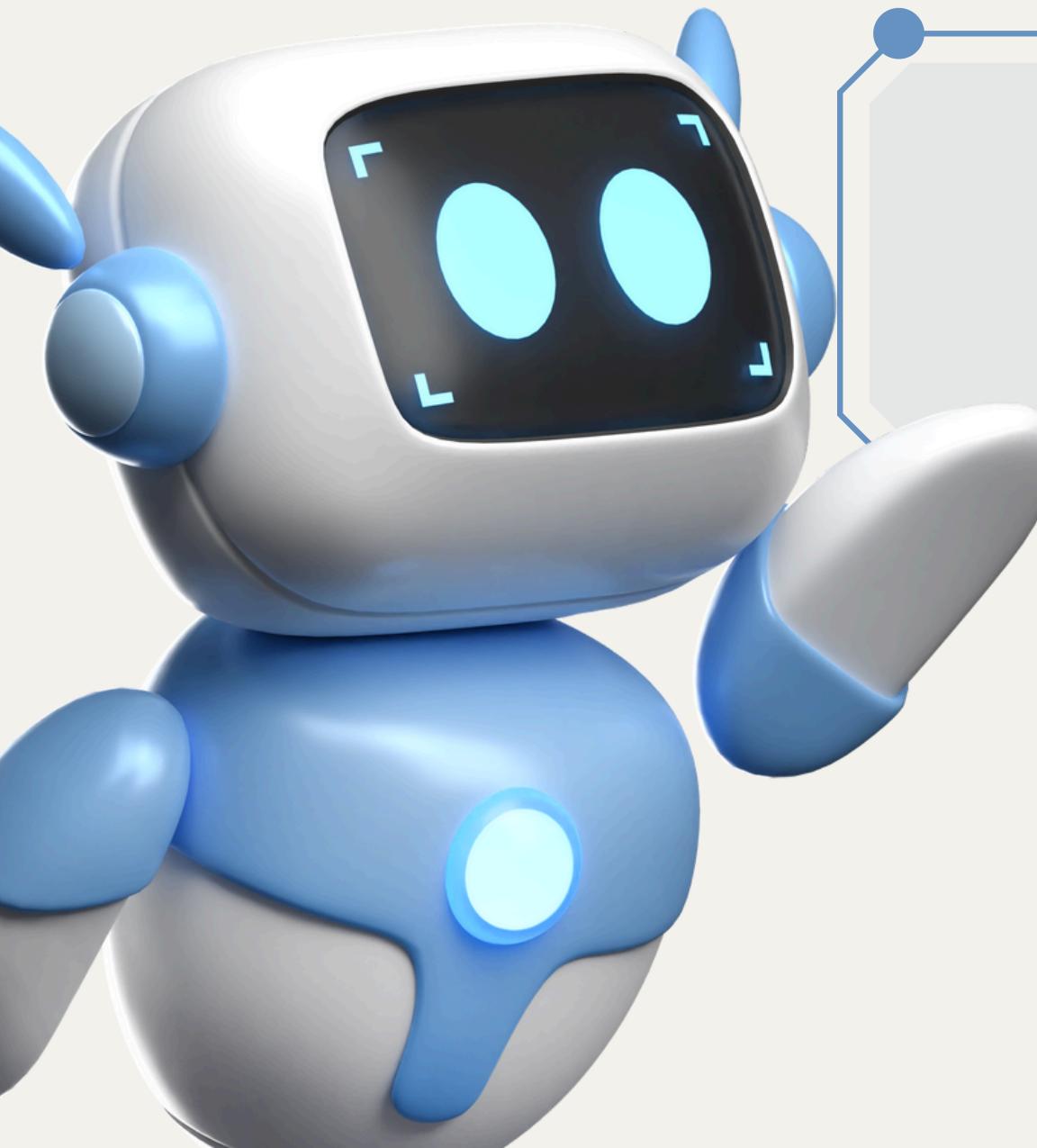






PERGUNTAS

OBRIGADO



E-MAIL: **SEGES.CGINF@GESTAO.GOV.BR**

DTGES