SPTrans

Solução “Near Real Time” para Monitoramento de Dados de Transporte Público em São Paulo - SP

Grupo 7 (DataBus)
Daniel Müeller
Lucas Fusco
Rodrigro Azevedo

Business Case

Contexto:

- O sistema de **transporte público** de São Paulo é utilizado por **milhões de passageiros** diariamente.
- A eficiência no gerenciamento da frota de ônibus impacta diretamente a **qualidade e a pontualidade** do serviço prestado.

Problema:

- **Falta de monitoramento** dos ônibus em circulação.
- Dificuldade em gerar **métricas e KPIs essenciais** para a tomada de decisão e melhoria do serviço.

Objetivo:

- Desenvolver uma **solução** que permita **monitorar e acompanhar os ônibus** em "***near real-time***".
- Fornecer métricas e KPIs para aprimorar o **gerenciamento do transporte** público.

Road Map

Início

Entendimento do problema
e definição da arquitetura

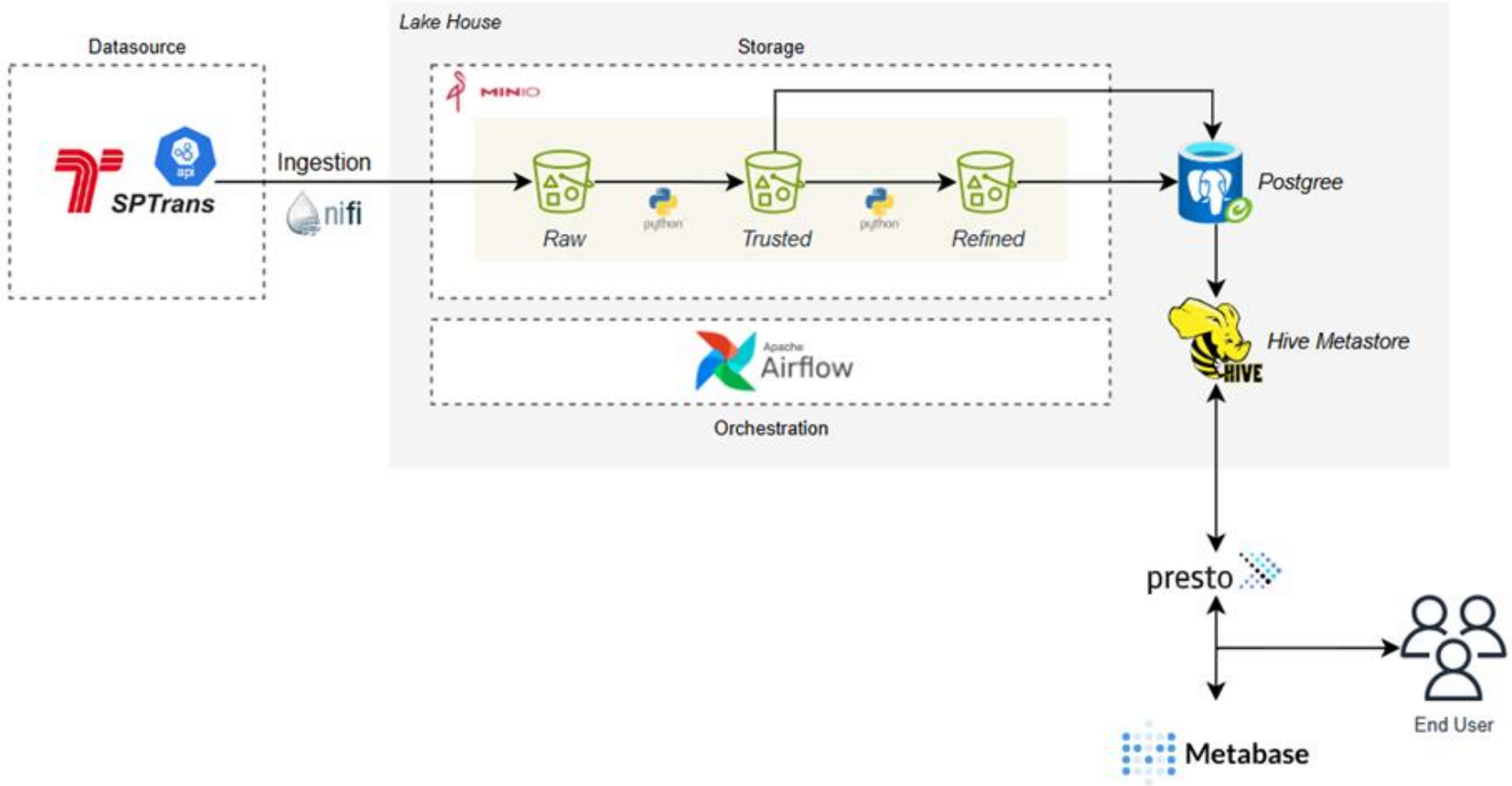
Desenvolvimento da arquitetura
proposta e validação.

Construção de métricas/KPIs
e entrega na ferramenta de BI.

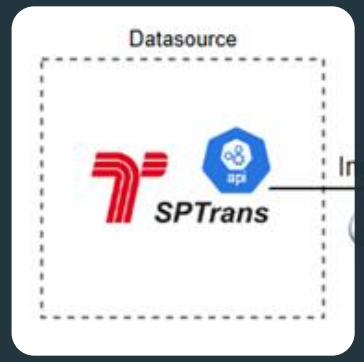
Mapeamento e definição dos
próximos passos (melhorias)

Fim

Arquitetura de Dados



Arquitetura de Dados



API Olho Vivo – SPTrans

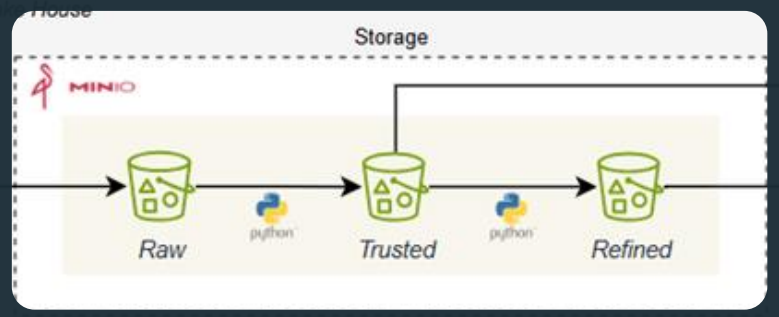
- A API SPTrans Olho Vivo fornece dados em tempo real sobre o transporte público de São Paulo, incluindo localização de ônibus, previsão de chegada e informações de linhas e paradas.

Arquitetura de Dados

Ingestão de Dados - NiFi

- **Automação e Flexibilidade:** Configuração visual de fluxos adaptáveis a diferentes fontes.
- **Processamento em Tempo Real:** Ideal para dados de streaming e grandes volumes.
- **Confiabilidade e Monitoramento:** Controle e recuperação de fluxos com interface intuitiva.

Arquitetura de Dados



Armazenamento de Dados - MinIO

- **Organização e Governança:** Estrutura clara (raw, trusted, refined) facilita gestão e compliance de dados.
- **Qualidade de Dados:** Permite melhorias graduais nos dados à medida que passam por cada camada.
- **Eficiência e Escalabilidade:** MinIO suporta grandes volumes e garante acesso rápido e seguro em cada estágio do ciclo de vida dos dados.

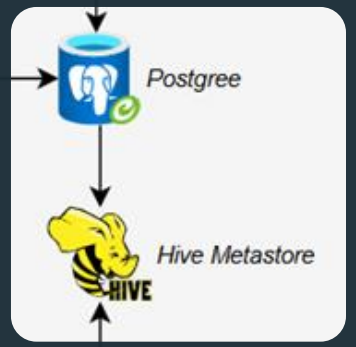
Orquestração dos Dados - AirFlow

- **Agendamento e Monitoramento:** Facilita o agendamento de tarefas complexas, permitindo a automação e monitoramento de fluxos de trabalho de dados.
- **Flexibilidade e Escalabilidade:** Suporta diferentes tipos de tarefas (ETL, chamadas de API, etc.) e pode escalar conforme a demanda.
- **Interface Visual:** Oferece uma interface gráfica para visualizar e gerenciar DAGs (Directed Acyclic Graphs), tornando mais fácil o acompanhamento do estado das tarefas e o gerenciamento de dependências.

Arquitetura de Dados

Disponibilidade de Dados – Postgree e Hive

- **PostgreSQL (Camada de Processamento e Consumo):** Suporta transações e consultas complexas, ideal para análises rápidas e integração com BI.
- **Hive (Camada de Dados Brutos e Processamento em Lote):** Processa grandes volumes com SQL em Hadoop, permitindo análises em dados históricos e dados não estruturados.
- **Complementaridade:** PostgreSQL oferece baixa latência para dados refinados, enquanto Hive gerencia dados massivos e brutos com escalabilidade.

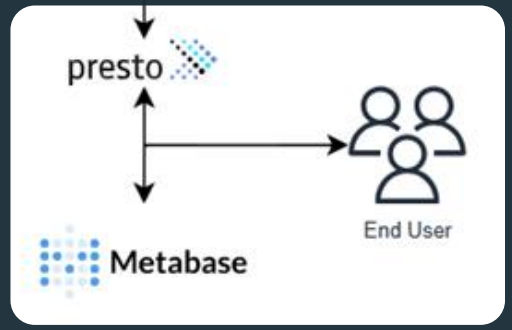


Arquitetura de Dados

Acesso e Consulta aos Dados – Presto e Metabase

- **Presto:**
  - **Motor de Consulta Distribuído:** Permite executar análises SQL em tempo real em diversas fontes de dados, facilitando a extração de insights de grandes volumes.
  - **Integração de Dados:** Suporta diferentes formatos e armazenamento, como Hadoop e PostgreSQL, oferecendo flexibilidade para análises abrangentes.

- **Metabase:**
  - **Ferramenta de Visualização e BI:** Conecta-se a bancos de dados e permite criar dashboards interativos e relatórios de forma intuitiva.
  - **Acessibilidade para Usuários Não Técnicos:** Facilita a análise de dados sem necessidade de conhecimentos avançados em SQL, promovendo a democratização da informação.



Nossa Solução (Entregáveis)

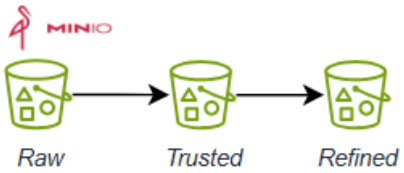
(Ingestão e Armazenamento)

Ingestão de Dados:

- **Posição** a cada 2 minutos (Part. em ano – mês – dia – hora).
- **Previsão de chegada** a cada 2 minutos (Part. em código linha – ano – mês – dia – hora).
- Cadastro de **linhas** 1x por semana (Part. em código linha – ano – mês – dia – hora).
- Cadastro de **paradas** 1x por semana (Part. em código linha – ano – mês – dia – hora).
- Cadastro de **empresas** 1x por mês (Part. em ano – mês).
- Veículos na **garagem** a cada 1 hora (Part. em código empresa – ano – mês – dia).

Armazenamento de Dados:

- Armazenamento dos dados em **3 zonas diferentes** de gerenciamento e acesso a partir de buckets do minIO.



Nossa Solução (Entregáveis)

(Orquestração, Transformação e Disponibilização)

Transformação dos dados:

- Movimentação entre as zonas de armazenamento através do python.

Orquestração dos dados:

- Agendamento e monitoramento das DAGs de transformação e movimentação entre as zonas de armazento (*raw – trusted – refined*).

Disponibilização dos dados:

- Armazenamento das zonas *trusted e refined* em um banco PostgreSQL para fácil consulta e integração com ferramentas de BI.
- Utilização do Hive para processamento dos dados não estruturados em grandes volumes, possibilitando análises históricas.

Nossa Solução (Entregáveis)

(Consulta e Visualização)

Consulta dos dados:

- Para usuários mais técnicos, os dados ficarão disponíveis para consulta e execução de queries SQL através do Presto, permitindo análises mais extensas e aprofundadas nos diferentes níveis de agregação das informações.

Visualização:

- Para visualização dos dados de maneira já estruturada e com seus KPIs definidos, estará à disposição dos usuários não técnicos, o Metabase, que servirá como uma ferramenta de BI para auxiliá-los em sua tomada de decisão.

Próximos Passos

1

Disponibilização dos dados via API

Facilitar o acesso e a integração de dados por meio da disponibilização de APIs, permitindo que usuários e sistemas consumam informações de forma rápida e eficaz.

2

Criação de novas métricas e KPIs (Enriquecimento da zona “refined”)

Enriquecer a zona "refined" com a criação de novas métricas e KPIs que ofereçam insights mais profundos e acionáveis, ajudando a impulsionar a tomada de decisões.

3

Implementação de técnica para CI/CD

Adotar práticas de Integração Contínua e Entrega Contínua para acelerar o ciclo de desenvolvimento e garantir que as alterações no código sejam testadas e implementadas de forma automatizada.

4

Set up automático de ambiente via terraform

Estabelecer um processo de configuração automática da infraestrutura, garantindo que os ambientes sejam escaláveis, consistentes e fáceis de replicar.

Obrigado

“Data Science are able to find ways to use Data to solve problems that otherwise would have been unsolved or solved using only intuition.”

Skmoroch, Peter.

Projeto Disponível em:

https://github.com/danimullller/projeto-final-bigdata

FIA BUSINESS SCHOOL