

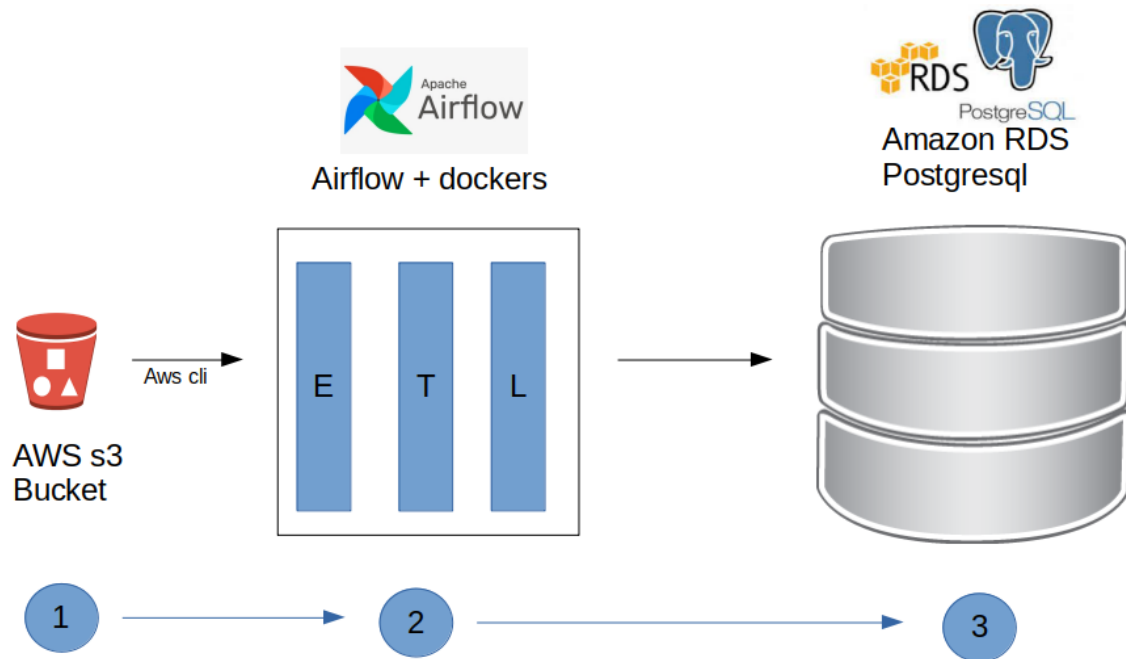
DOCUMENTAÇÃO TÉCNICA - DELIVERY CENTER



deliverycenter

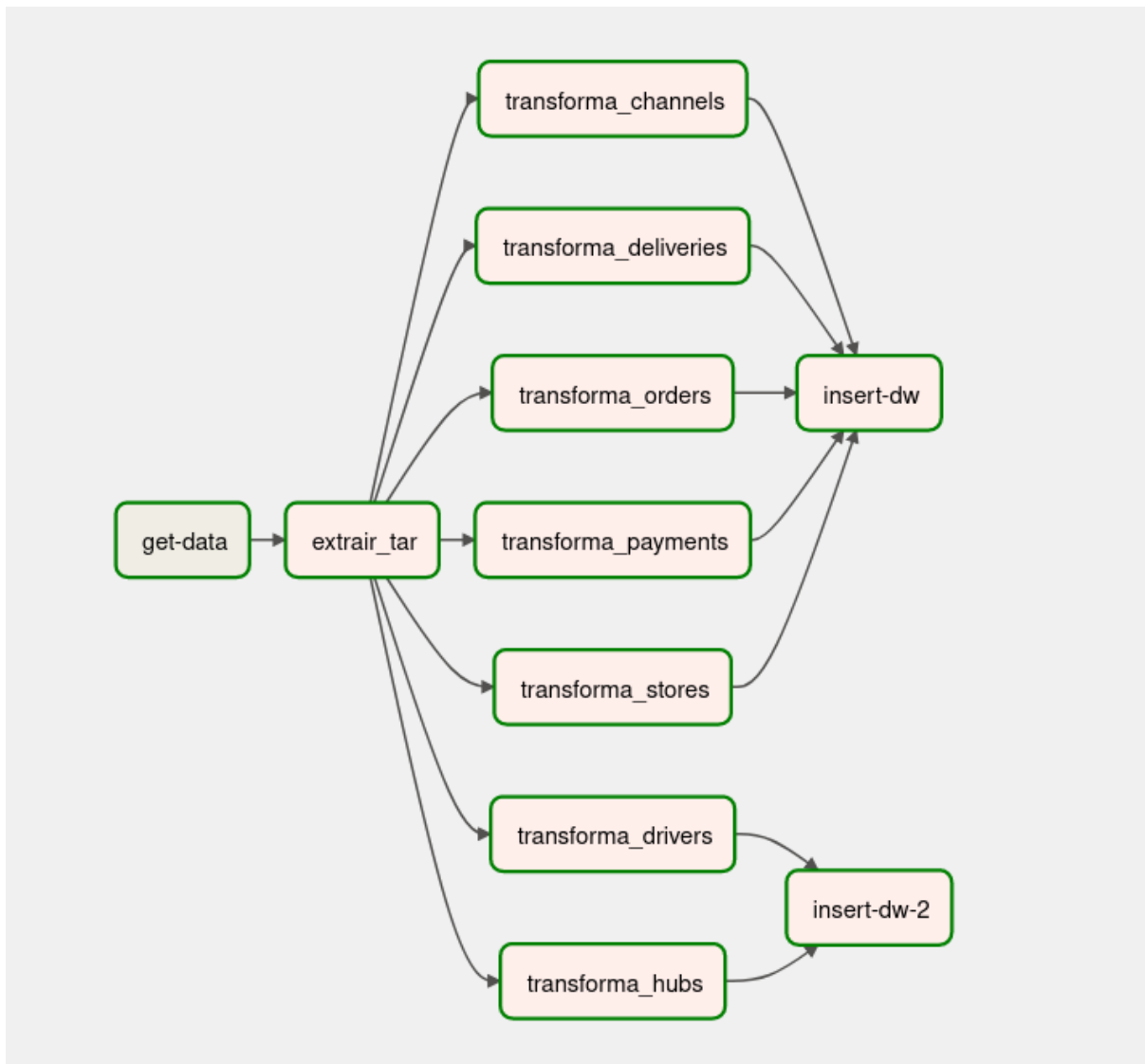
SUMÁRIO.....	2
ARQUITETURA GERAL DOS DADOS.....	3
ARQUITETURA DO PROCESSO DE ETL.....	4
Estrutura do banco de dados.....	5
ERD - Diagrama de entidade relacionamento da base de dados.....	5
Dicionário de dados.....	6
Tecnologias utilizadas.....	10
Cenários.....	10
Considerações finais.....	11

1. ARQUITETURA GERAL DOS DADOS



1. Os dados disponibilizados foram armazenados em um aws bucket s3 em formato .tar.
2. Em seguida é feita a extração dos dados através de um conexão aws cli. Os dados são limpos, e é gerado um novo arquivo csv na pasta data. E por fim eles são preparados e carregados no banco de dados.
3. Os dados são persistidos na base de dados criada utilizando AWS RDS PostgreSQL.

2. ARQUITETURA DO PROCESSO DE ETL



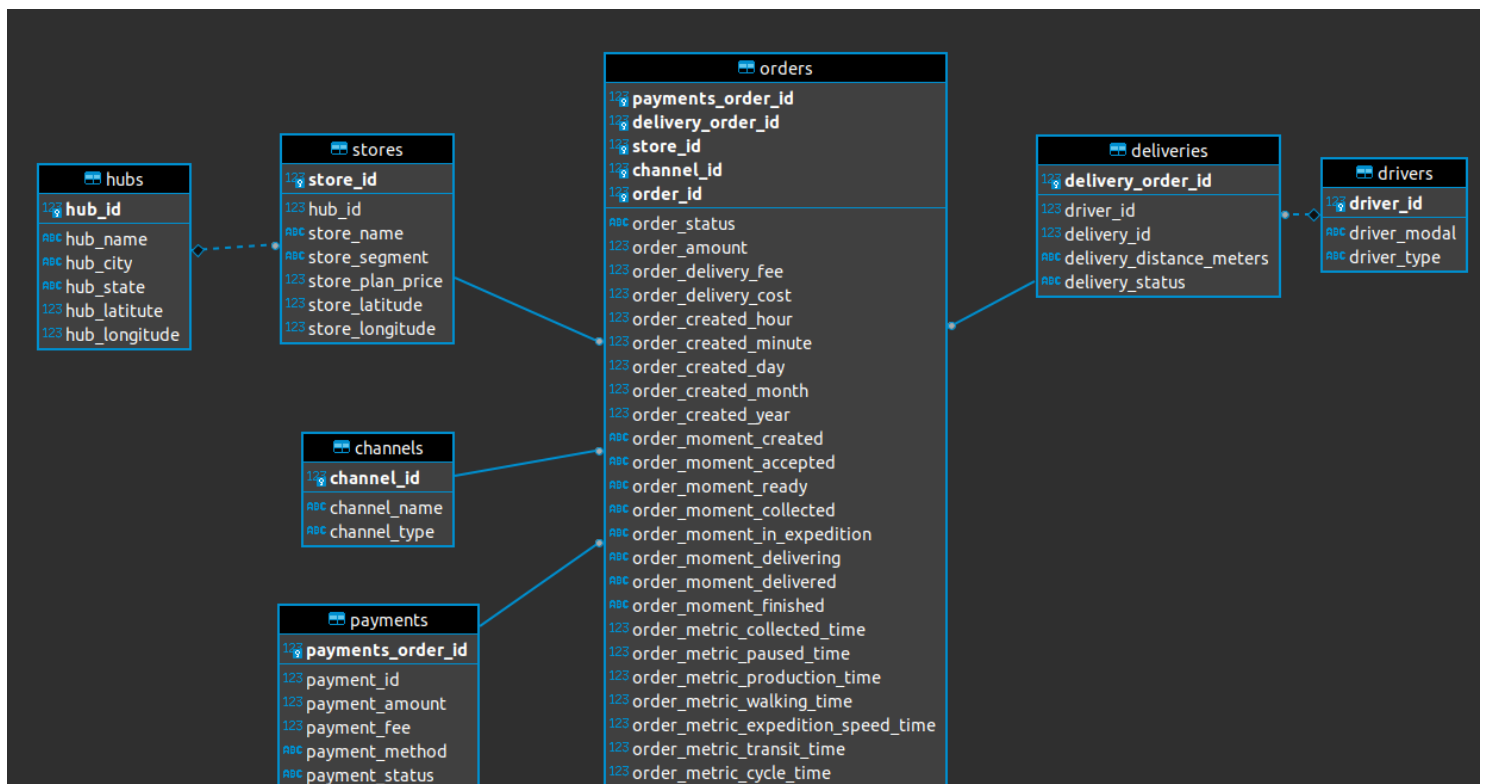
Onde,

- **get-data:** task responsável pela extração de dados do bucket s3
- **extrair-tar:** task responsável pela extração do arquivo em formato .tar
- **transforma_channels:** task responsável pela limpeza do arquivo channels.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **transforma_deliveries:** task responsável pela limpeza do arquivo deliveries.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **transforma_orders:** task responsável pela limpeza do arquivo orders.csv e inserir os dados em um novo arquivo csv dentro da pasta data.

- **transforma_payments:** task responsável pela limpeza do arquivo payments.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **transforma_stores:** task responsável pela limpeza do arquivo channels.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **transforma_drivers:** task responsável pela limpeza do arquivo drivers.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **transforma_hubs:** task responsável pela limpeza do arquivo hubs.csv e inserir os dados em um novo arquivo csv dentro da pasta data.
- **insert-dw, insert-dw-2:** tasks responsáveis pela inserção dos dados na base de dados

3. Estrutura do banco de dados

3.1 ERD - Diagrama de entidade relacionamento da base de dados



Modelo de acordo com disponibilizado (snow flake)

- **TABELA FATO:** orders
- **TABELAS DIMENSÕES:** hubs, stores, channels, payments, deliveries, drivers

3.2 Dicionário de dados

Hubs

nome do campo	chave	tipo de dado	descrição	tamanho
hub_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
hub_name		varchar	nome do hub	40
hub_city		varchar	nome da cidade onde se localiza o hub	40
hub_state		varchar	nome da estado onde se localiza o hub	2
hub latitude		float	dado de latitude do hub	4 bytes
hub_longitude		float	dado de longitude	4 bytes

Payments

nome do campo	chave	tipo de dado	descrição	tamanho
payments_order_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
payment_id		integer	identificador do pagament	4 bytes
payment_amount		numeric	valor total de pagamento	sem limite
payment_fee		numeric	taxa do pagamento	sem limite
payment_method		varchar	forma de pagamento	40
payment_status		varchar	status do pagamento	40

Stores

nome do campo	chave	tipo de dado	descrição	tamanho
stores_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
hub_id	FK	integer	identificador estrangeiro da tabela hubs	4 bytes
store_name		varchar	nome da loja	40
store_segment		varchar	segmento em que a loja atua	20
store_plan_price		float	plano de preços da loja	4 bytes
store_latitude		float	dado de latitude da loja	4 bytes
store_longitude		float	dado de longitude da loja	4 bytes

Drivers

nome do campo	chave	tipo de dado	descrição	tamanho
driver_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
driver_modal		varchar	modalidade do entregador	20
driver_type		varchar	tipo de contrato do entregador	40

Deliveries

nome do campo	chave	tipo de dado	descrição	tamanho
delivery_order_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
driver_id	FK	integer	identificador estrangeiro da tabela drivers	4 bytes

delivery_id		float	identificador da entrega	4 bytes
delivery_distance_meters		float	distância em metros da entrega	4 bytes
delivery_status		float	status da entrega	4 bytes

Channels

nome do campo	chave	tipo de dado	descrição	tamanho
channels_id	PK	serial	um número sequencial atribuído sempre que um novo registro é adicionado	4 bytes
channels_name		varchar	nome do canal de venda	40
channels_type		varchar	tipo de canal de venda	40

Orders

nome do campo	chave	tipo de dado	descrição	tamanho
payments_order_id	PK/FK	integer	identificador estrangeiro da tabela payments	4 bytes
delivery_order_id	PK/FK	integer	identificador estrangeiro da tabela delivery	4 bytes
store_id	PK/FK	integer	identificador estrangeiro da tabela store	4 bytes
channel_id	PK/FK	integer	identificador estrangeiro da tabela channels	4 bytes
order_id	PK	integer	identificador	4 bytes
order_status		varchar	status do pedido	20

order_amount		numeric	quantidade do pedido	sem limite
order_delivery_fee		float	taxa do pedido	4 bytes
order_delivery_cost		numeric	custo do pedido	sem limite

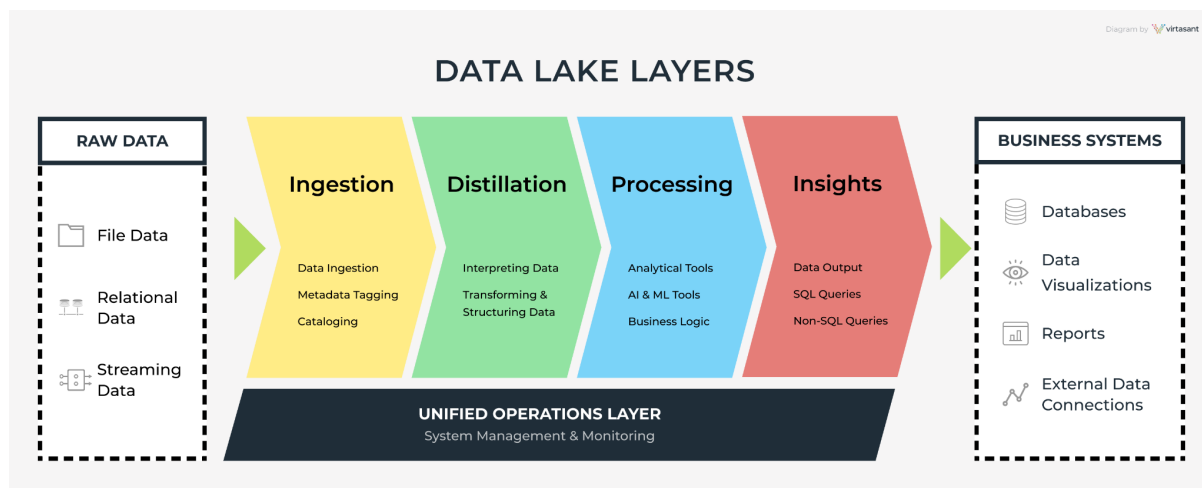
order_created_hour		integer	hora do pedido	4 bytes
order_created_minute		integer	minuto do pedido	4 bytes
order_created_day		integer	dia do pedido	4 bytes
order_created_month		integer	mês do pedido	4 bytes
order_created_year		integer	ano do pedido	4 bytes
order_moment_created		timestamp	data em que pedido é gerado	8 bytes
order_moment_accepted		timestamp	data em que o pedido é aceito	8 bytes
order_moment_ready		timestamp	data em que o pedido está pronto	8 bytes
order_moment_collected		timestamp	data em que o pedido é coletado	8 bytes
order_moment_in_expedition		timestamp	data em que pedido esta expedido	8 bytes
order_moment_delivering		timestamp	data em que pedido está em transporte	8 bytes
order_moment_delivered		timestamp	data em que pedido foi enviado	8 bytes
order_moment_finished		timestamp	data em que o pedido foi finalizado	8 bytes
order_metric_collected_time		float	métrica de tempo em que o pedido é coletado	4 bytes
order_metric_paused_time		float	métrica de tempo em que o pedido esta parado	4 bytes
order_metric_production_time		float	métrica de tempo de produção do pedido	4 bytes
order_metric_walking_time		float	métrica de tempo estimado de entrega caminhando	4 bytes
order_metric_expedition_speed_time		float	métrica de tempo estimado de velocidade de entrega	4 bytes
order_metric_transit_time		float	métrica de tempo estimado do transito do pedido	4 bytes

order_metric_cycl e_time		float	métrica de tempo estimado de entrega em bicicletas	4 bytes
-----------------------------	--	-------	---	---------

4. Tecnologias utilizadas

- AWS s3 bucket
- Apache Airflow 2.1.3
- Docker-compose
- Aws CLI 1.0
- Python 3.8
- Ubuntu 20.4 LTS
- AWS RDS Postgresql 13.3
- Pgadmin
- Dbeaver
- Libreoffice calc
- VS Code

5. Cenários



Data lake com camadas de operações. Unifica os dados e pode disponibilizar diretamente os dados transformados para análise. Dispensando o uso de DWs. Porém, existe um single-point-of-failure, onde toda estrutura dependeria apenas do data lake

6. Considerações finais

As tecnologias e ferramentas foram escolhidas de acordo com a disponibilidade de uso.

Algumas tecnologias foram inviabilizadas por custo, como o uso do AWS Redshift, devido a limitações da conta free tier.