



Cotia, 17 de dezembro de 2020

| Controle de Versão |        |            |                      |
|--------------------|--------|------------|----------------------|
| Autor              | Versão | Data       | Descrição            |
| Diego              | 1.0    | 17/12/2020 | Criação do Documento |

## **1.Introdução**

Este documento visa detalhar as necessidades do projeto de migração de dados utilizando o Azure Data Factory do ponto de vista técnico, bem como listar as possíveis soluções, suas premissas e atividades a serem executadas durante o projeto.

## **2.Modelo de extração**

O projeto será feito da seguinte forma: Os dados serão extraídos de uma API do Banco Central, será feita uma conexão via Data Factory para iniciar a migração de dados até ao banco de dados Azure SQL Database aonde será feita a ETL dos dados para a criação de um Data warehouse. Com o processo concluído o arquivo contendo os dados será disponibilizado no Azure Blob Storage.

### 3. Premissas da solução

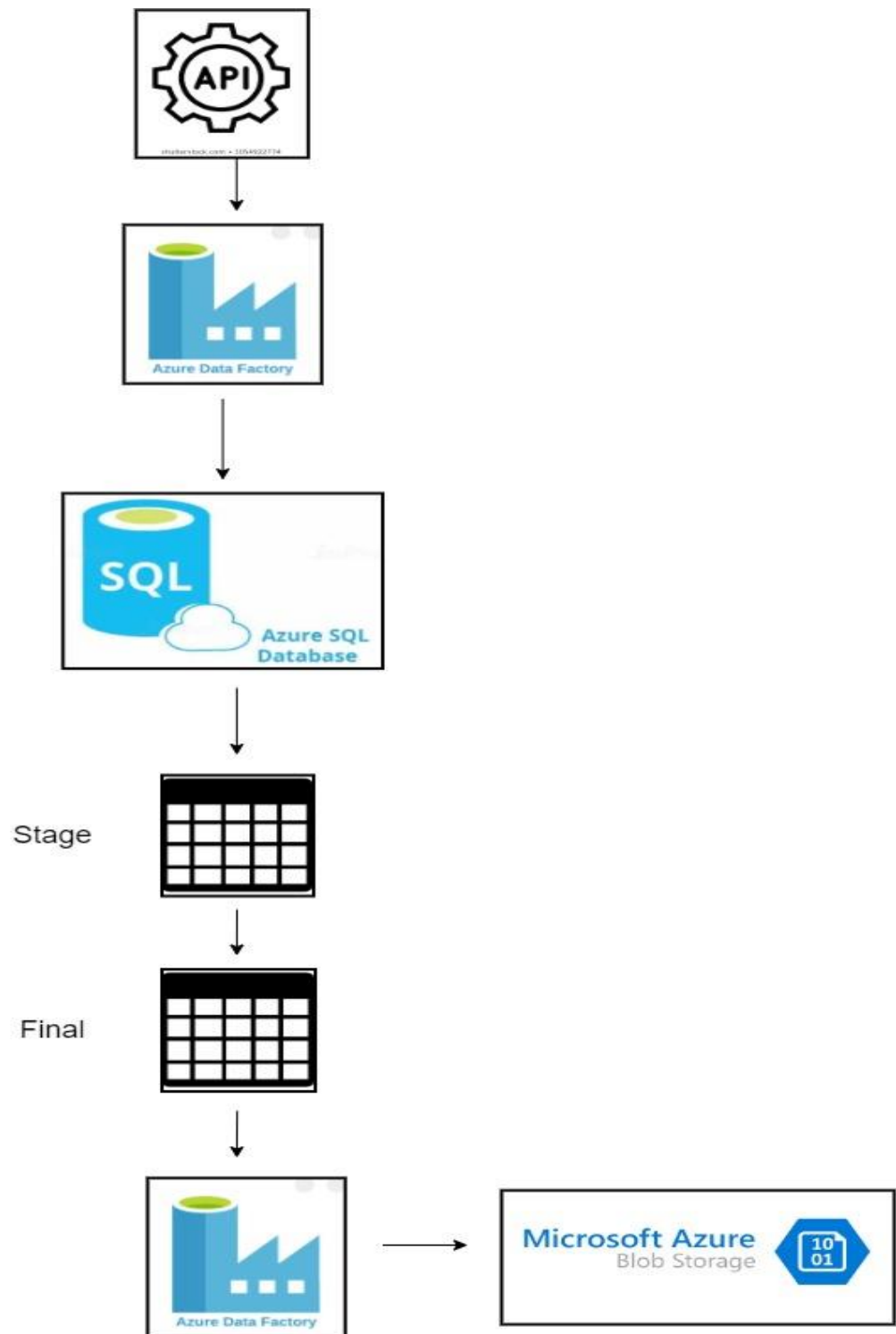
Os itens listados abaixo apresentam as premissas da solução, origem e especificação dos dados.

- Os dados serão consumido através de uma API do Banco Central
- Os dados deverão seguir o layout pré-definido e especificado neste documento;

.

## 4. Modelo da arquitetura sugerida

A figura abaixo apresenta a arquitetura da solução proposta com base no levantamento de requisitos e entendimento do negócio.



## 5. Disponibilização dos dados através da API

Os dados serão disponibilizados através da API abaixo:

[https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/odata/CotacaoDolarPeriodo\(dataInicial=@dataInicial,dataFinalCotacao=@dataFinalCotacao\)?@dataInicial='01-01-2019'&@dataFinalCotacao='12-31-2025'&\\$top=9000&\\$format=text/csv&\\$select=cotacaoCompra,cotacaoVenda,dataHoraCotacao](https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/odata/CotacaoDolarPeriodo(dataInicial=@dataInicial,dataFinalCotacao=@dataFinalCotacao)?@dataInicial='01-01-2019'&@dataFinalCotacao='12-31-2025'&$top=9000&$format=text/csv&$select=cotacaoCompra,cotacaoVenda,dataHoraCotacao)

## 6. Criação do processo de extração – Data Factory

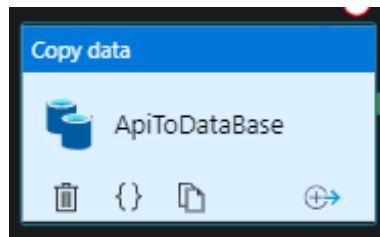
Ao todo serão usados 2 componentes:

- Copy Data
- Stored Procedure

**Copy Data:** Componente que cria uma pipeline indicando o SOURCE que é de onde os dados irão ser extraídos e o SINK que é para aonde os dados serão destinados.

**Stored Procedure:** Executa uma procedure fazendo o tratamento dos dados. a serem inseridos no Data Warehouse (Esse componente só será utilizado no momento de extração do Data Warehouse para o Azure Blob Stage).

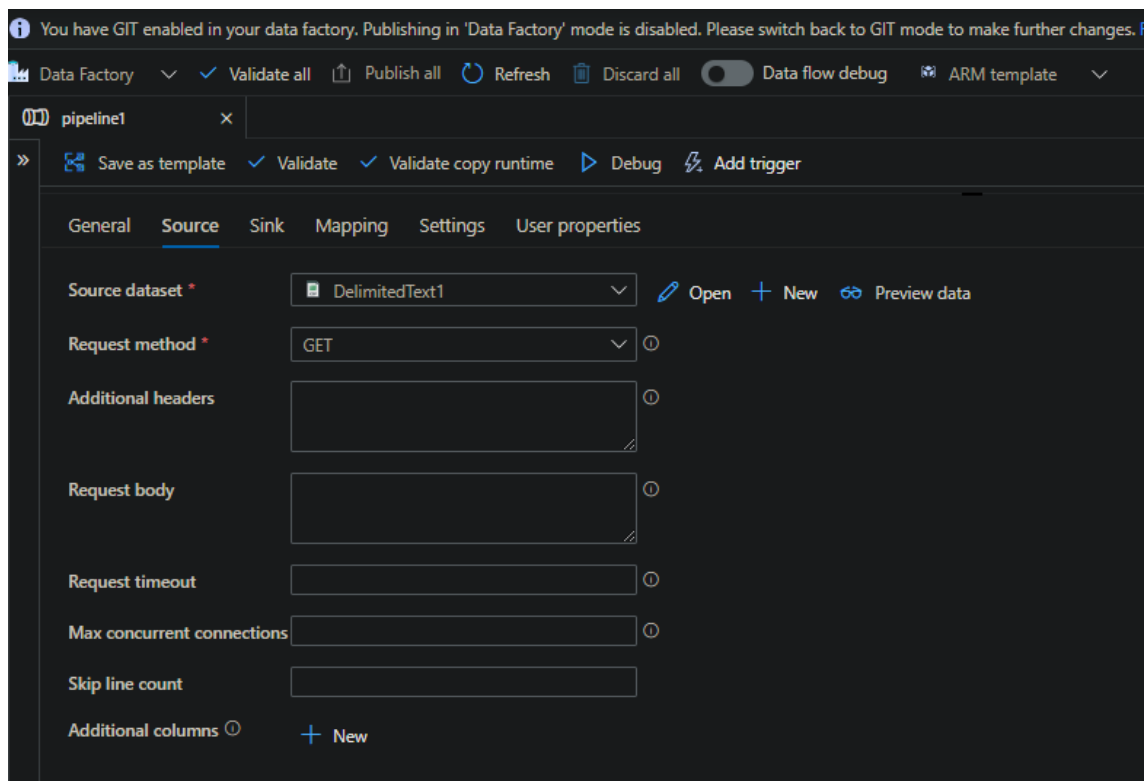
## Componente Copy Data – Api para Database



### Configurações importantes:

**Source:** Nessa aba será inserido o tipo de dataset a ser utilizado na extração dos dados. No caso da extração será utilizado o HTTP.

### Imagem ilustrativa:



**Dataset HTTP:** Deverá ser selecionado o formato do arquivo e no caso será utilizado o DelimitedText lembre-se de selecionar first line is a header para que a primeira linha seja considerada o cabeçalho.

**LinkService:** selecione a opção SELECT > NEW para fazer uma nova configuração, vá até o campo BASE URL e cole o link da API disponibilizada. No campo Authentication type selecione ANONYMOUS e finalize com create.

### Imagem ilustrativa

**New linked service (HTTP)**

Description

Connect via integration runtime \* ⓘ  
AutoResolveIntegrationRuntime

Base URL \*  
https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/odata/CotacaoDolarPeriodo(dataIn)

Add dynamic content [Alt+P]

Server Certificate Validation ⓘ  
☒ Enable ☐ Disable

Authentication type \*  
Anonymous

Annotations  
+ New

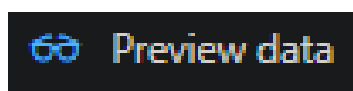
Name

Parameters

Advanced ⓘ

Create Test connection Cancel

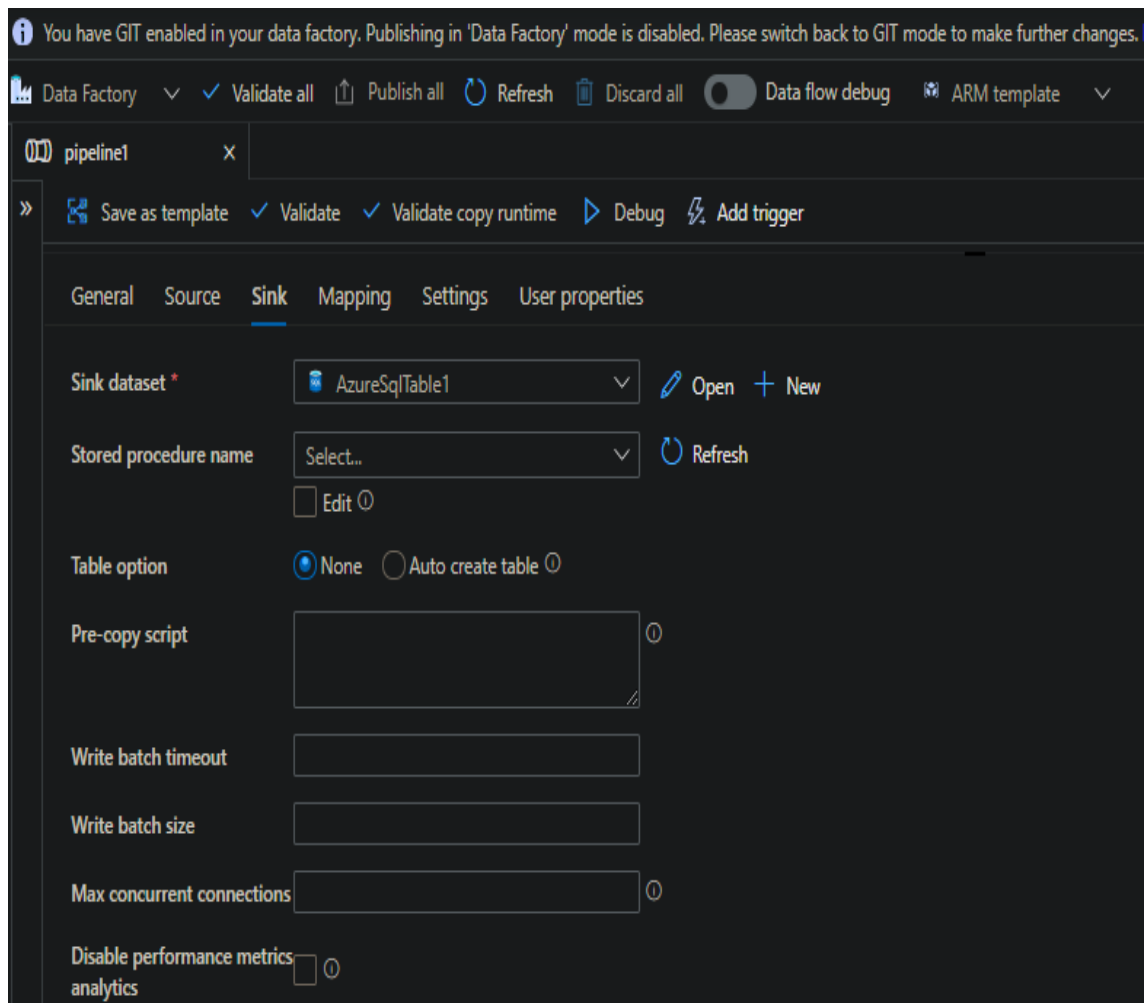
Aperte em preveiw data para visualizar os dados da API





**Sink**: Nessa aba será inserido o tipo de dataset a ser utilizado como destino dos dados migrados, no caso será utilizado o Azure SQL Database.

## Imagem ilustrativa



**Dataset Azure SQL Database:** Dataset responsável pela conexão com o Azure SQL Database

**LinkService:** Selecione a opção SELECT > NEW para fazer a configuração das credenciais que acessam seu banco de dados.

**Server name:** Nome do servidor.

**Database name:** Nome da database.

**User name:** Credencial de acesso ao database.

**Password:** Key de acesso ao banco.

**Table name:** nome da tabela de stage a ser utilizada.

**Imagem ilustrativa:**

**Edit linked service (Azure SQL Database)**

**Account selection method** ⓘ

☒ From Azure subscription ☐ Enter manually

**Azure subscription**

BLUESHIFT (8813add0-971e-44dc-817e-0a512257a763) ▼

**Server name \***

diegoserver1 ▼

**Database name \***

dolar\_diego ▼ ↻

**Authentication type \***

SQL authentication ▼

**User name \***

diego

**Password \***

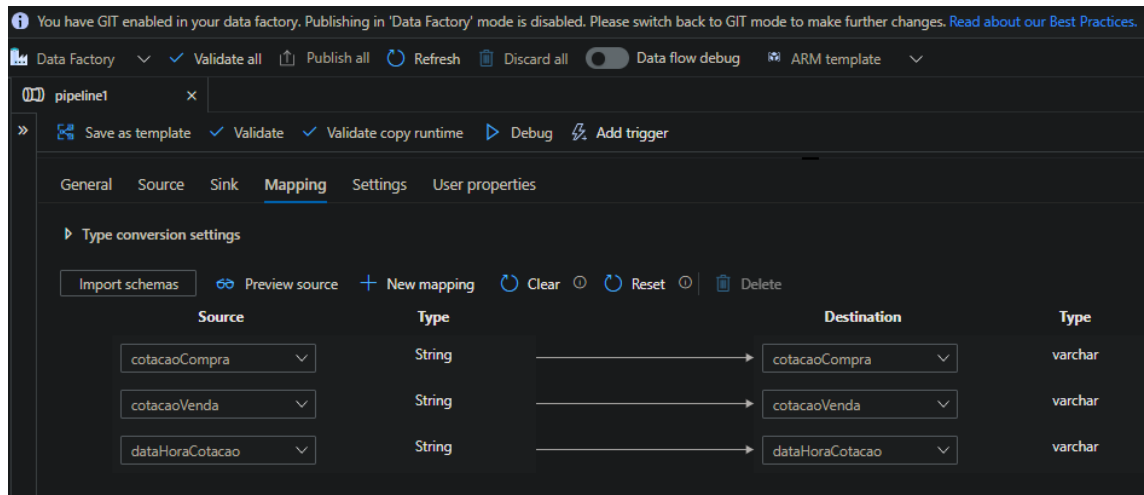
**Additional connection properties**

+ New

Apply Test connection Cancel

**Mapping:** Faça importe do schema em IMPORT SCHEMAS, caso seja necessária alguma alteração nos tipos primitivos pode ser alterada nessa sessão.

### Imagem ilustrativa



Com as configurações feitas vá até PUBLISH ALL para salvar as alterações feitas e em seguida em DEBUG para executar o fluxo.

## 7. Criação do Data Warehouse – Azure SQL Database

### Criação da tabela de Stage

Após o processo de migração da API para o Azure SQL Database feito pelo Data Factory, deverá ser criada uma tabela de stage para armazenar os dados.

Colunas:

- cotacaoCompra varchar
- cotacaoVenda varchar
- dataHoraCotacao varchar

## **Criação da tabela de Final**

A tabela deverá ser criada com as seguintes colunas e tipos primitivos:

Colunas:

- cotacaoCompra float
- cotacaoVenda float
- dataHoraCotacao date

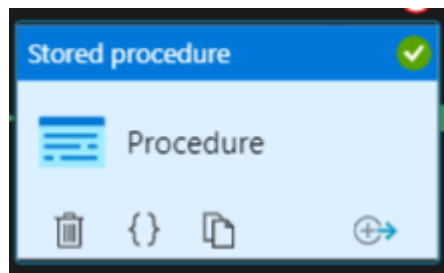
## **Tratamento dos dados**

Com os dados inseridos na tabela stage, deverá ser feita as seguintes transformações nos dados:

- Criar uma procedure que faça o cast dos dados varchar para Date e Float.
- Criar uma procedure que faça o insert dos dados convertidos na tabela Final.

## 8. Migração para o Azure SQL Database para Azure Blob Storage

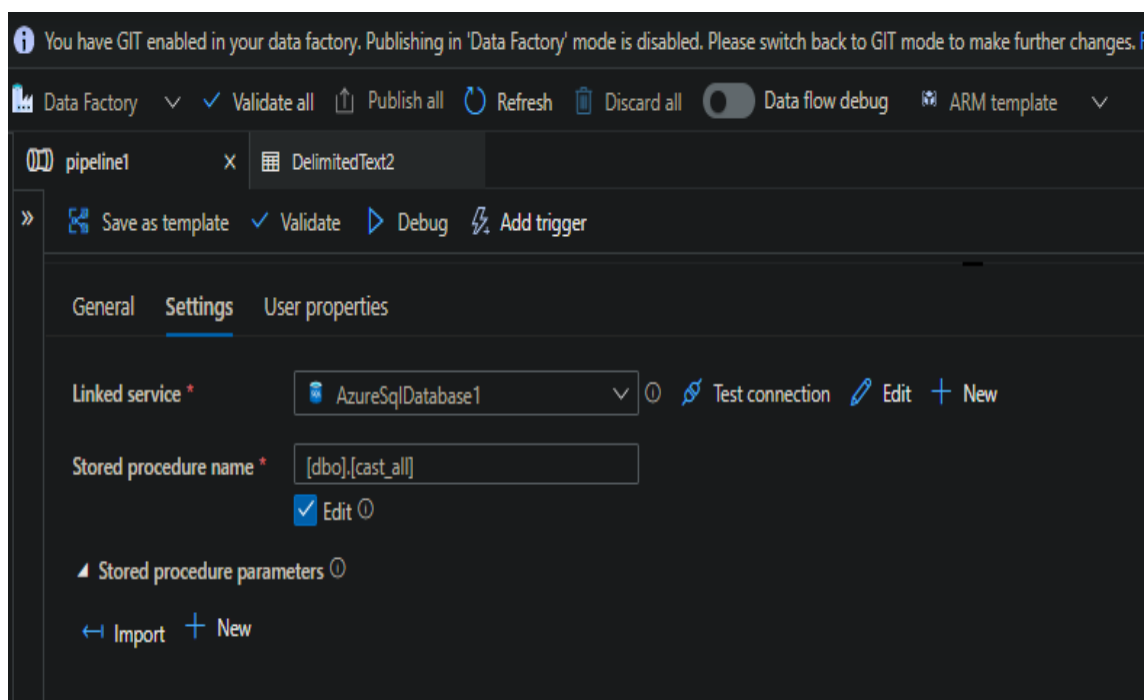
### Componente Stored Procedure



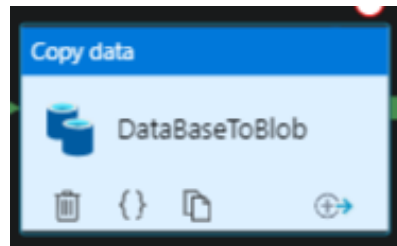
**LinkService:** Selecione o dataset Azure SQL Database já criado.

**Stored Procedure name:** Selecione a procedure criada no banco de dados

**Imagens ilustrativas:**



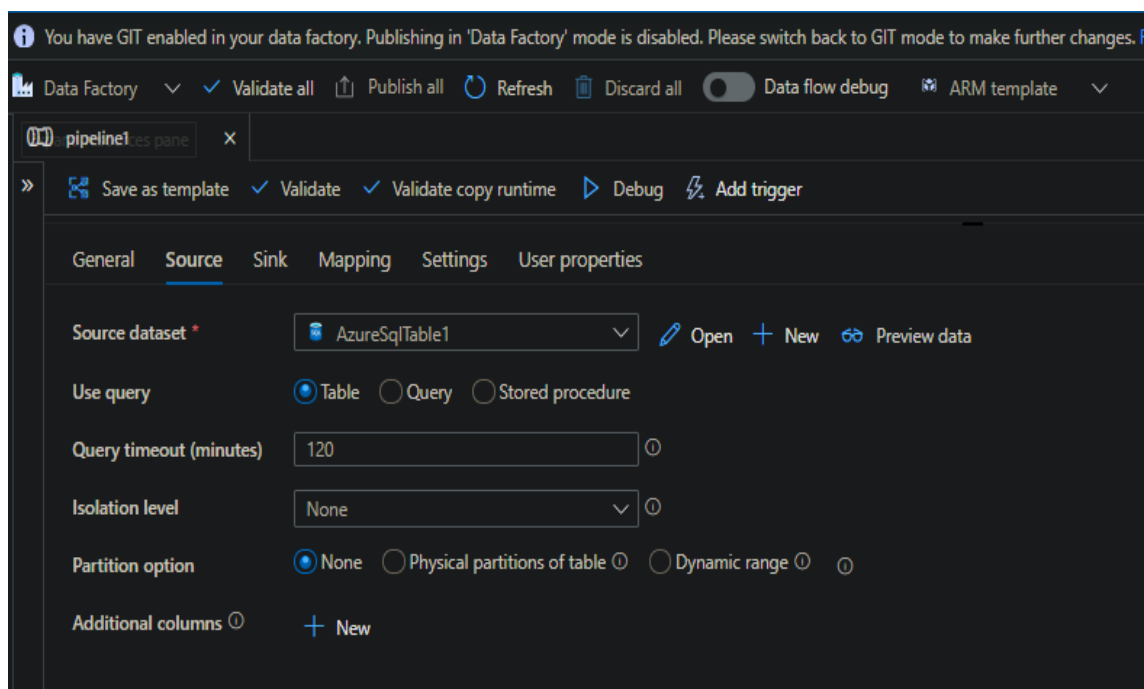
## Componente Copy Data – Database para o Blob



### Configurações importantes:

**Source:** Selecione o dataset referente ao Azure SQL database configurado anteriormente.

**Imagem ilustrativa:**



### Sink:

**Dataset Azure Blob Storage:** Deverá ser selecionado o formato do arquivo e no caso será utilizado o DelimitedText lembre-se de selecionar first line is a header para que a primeira linha seja considerada o cabeçalho e altere a extensão do arquivo de .txt para .csv.

**LinkService:** selecione a opção SELECT > NEW para fazer uma nova configuração, vá até o campo STORAGE ACCOUNT NAME e selecione a storage account que servirá de destino pra os dados, após isso selecione o container de destino

**Imagens ilustrativas:**

**Passo 1:**

**Edit linked service (Azure Blob Storage)**

**Name \***  
AzureBlobStorage1

**Description**

**Connect via integration runtime \*** ⓘ  
AutoResolveIntegrationRuntime

**Authentication method**  
Account key

**Connection string** **Azure Key Vault**

**Account selection method** ⓘ  
☐ From Azure subscription ☒ Enter manually

**Storage account name \***  
diegoblue1hdstorage

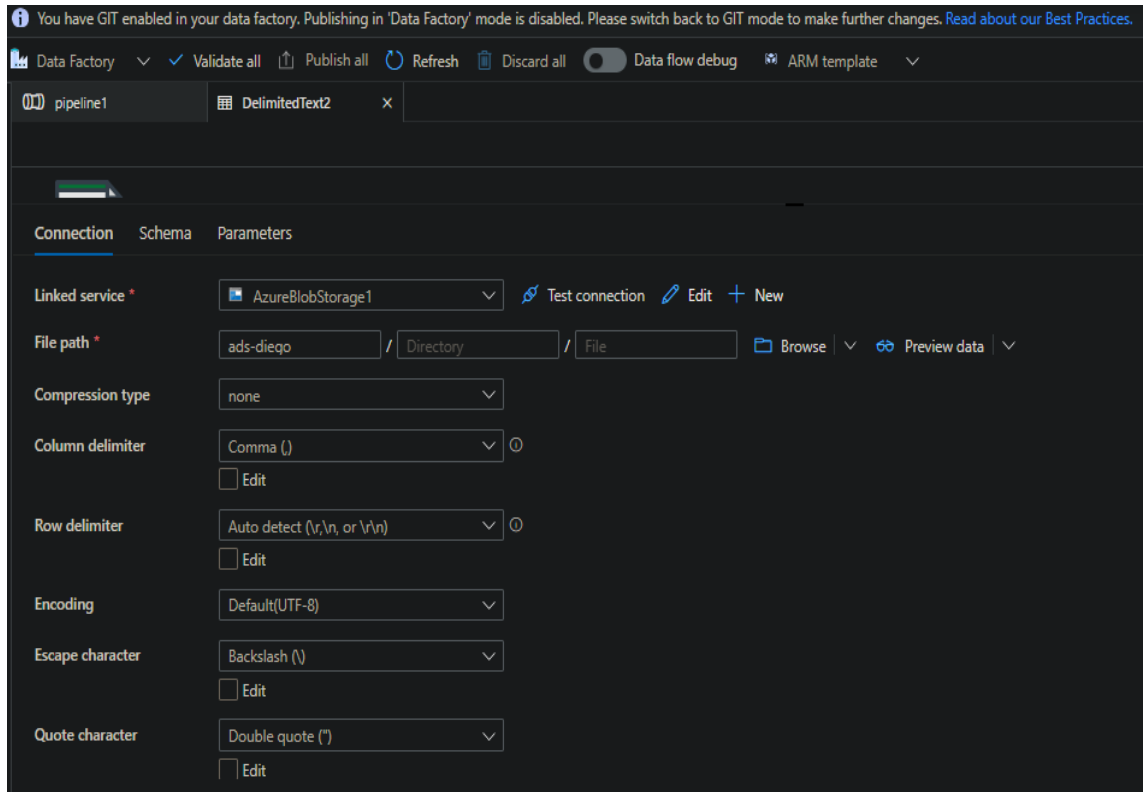
**Storage account key** **Azure Key Vault**

**Storage account key \***  
.....

**Endpoint suffix**

**Apply** **Test connection** **Cancel**

## Passo 2:

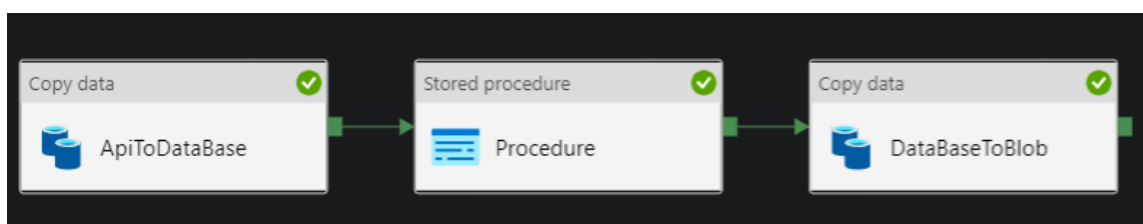


**Mapping:** Faça importe do schema em IMPORT SCHEMAS, caso seja necessária alguma alteração nos tipos primitivos pode ser alterada nessa sessão.

Com as configurações feitas vá até PUBLISH ALL para salvar as alterações feitas e em seguida em DEBUG para executar o fluxo.

Acesse a storage account para validar a disponibilização do arquivo csv.

## Visão do Fluxo Final:





## Imagem ilustrativa arquivo armazenado no Azure Blob Stage:

