

# MVP de Engenharia de Dados

Roberto Harkovsky da Cunha

## 1 Definição do Problema

O IMDb (um acrônimo para Internet Movie Database) é um banco de dados on-line de informações relacionadas a filmes, séries de televisão, podcasts, vídeos caseiros, videogames e streaming de conteúdo on-line - incluindo elenco, equipe de produção e biografias pessoais, resumos de enredos, curiosidades, classificações e análises críticas e de fãs. Como complemento aos dados, o IMDb oferece uma escala de classificação que permite aos usuários votar e avaliar os filmes em uma escala de um a dez.

Neste escopo, objetivo deste projeto é o de realizar uma análise dos títulos publicados (filmes, seriados de TV) e responder as seguintes questões:

- Qual a popularidade dos filmes de James Bond
- Qual a pontuação dos filmes de James Bond
- Quais são os gêneros de filmes mais populares
- Quais os gêneros com as melhores pontuações
- Quais os 10 filmes com maior popularidade de Steven Spielberg
- Quais os 10 filmes com maior pontuação de Steven Spielberg
- Quais os 10 diretores de filmes com maiores médias de pontuação com mais de 5 filmes realizados?
- Quais os são 10 diretores de filmes mais populares?
- Qual é o tempo de execução típico para filmes de cada gênero?
- Quantos filmes foram feitos de cada gênero por ano entre 2020/22?
- Quem são os atores que interpretaram 'James Bond' em um filme?
- Quantas vezes eles fizeram o papel de 'James Bond'?
- Quantos filmes existem em cada gênero?

## 2 Visão Geral do Projeto

Utilizaremos neste projeto a nuvem Azure da Microsoft, e seus serviços. Forma utilizados serviços de repositórios de dados para armazenar os dados

originais. Foram criados base de dados em servidores SQL serverless como local para carga dos dados e para o processo de ETL foi utilizado o Azure Data Factory (ADF) que é o serviço ETL na nuvem do Azure para integração e transformação de dados sem servidor.

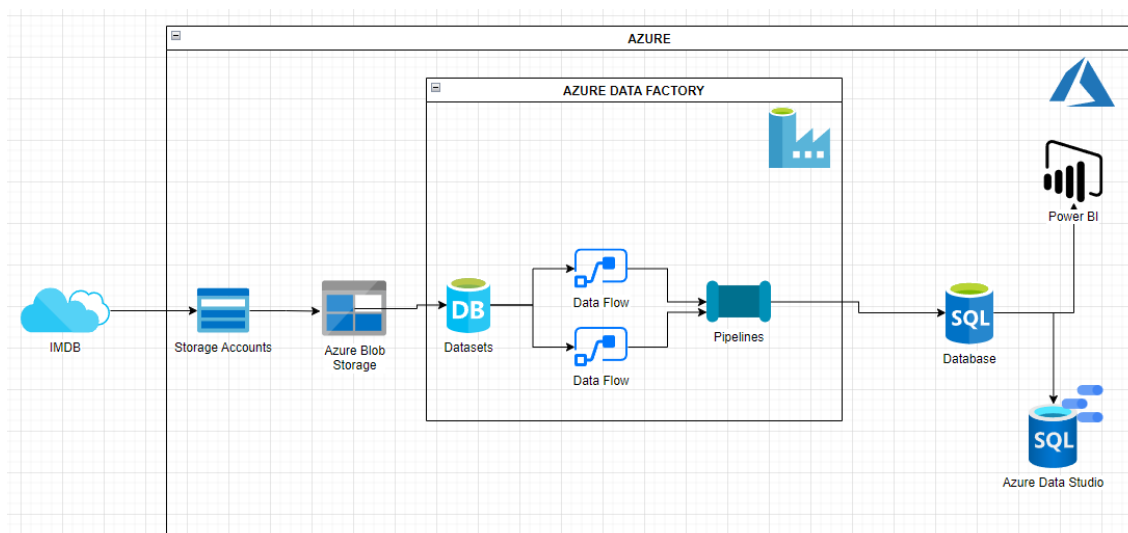
O desenvolvimento do projeto foi composto pelas seguintes etapas:

- Fonte de dados (Data Sourcing)
- Modelagem de dados
- Ingestão de dados
  - Extração
  - Transformação de dados
  - Carga de dados
- Resultados e Visualização de dados

Na etapa de fonte de dados foram utilizados arquivos públicos do portal do IMDB.

As etapas de ETL (extração, transformação e carga dos dados) foram realizadas na plataforma Azure e no Azure Data Factory. A figura abaixo apresenta uma visão geral destas etapas.

*Figura 1 - Visão geral do Processo de Ingestão dos dados numa base SQL*



A etapa de resultados e Visualização foi feita utilizando o Azure Data Studio para geração das consultas e o PowerBI da Microsoft para geração de gráficos.

Todas as etapas do projeto serão detalhadas nos itens a seguir.

### 3 Fonte de dados (Data Sourcing)

Para este projeto foi utilizado o Dataset IMDb, que consiste em 7 arquivos compactados, com valores separados por tabulação (\*.tsv), que estão disponíveis para download em <https://datasets.imdbws.com/>. Os arquivos selecionados são os seguintes:

- name.basics.tsv.gz
- title.akas.tsv.gz
- title.basics.tsv.gz
- title.crew.tsv.gz
- title.episode.tsv.gz
- title.principals.tsv.gz
- title.ratings.tsv.gz

Algumas informações adicionais sobre os dados deste dataset IMDB:

- Os dados são atualizados diariamente, embora os dados utilizados neste projeto tenham sido obtidos em 16/09/2023.
- Cada um desses arquivos compactados com valores separados por tabulação (TSV) formatados no conjunto de caracteres UTF-8.
- A primeira linha de cada arquivo contém cabeçalhos que descrevem o que há em cada coluna. Um “\N” é usado para indicar que um campo específico está faltando ou tem um valor NULL para esse título ou nome.

#### 3.1 Detalhes dos Dados IMDB

O detalhamento do conteúdo dos arquivos está a seguir:

##### 3.1.1 name.basics.tsv.gz

Contém as seguintes informações para nomes da equipe/atores:

Coluna	Descrição
<b>nconst (string)</b>	identificador alfanumérico exclusivo do nome/pessoa.
<b>PrimaryName (string)</b>	nome pelo qual a pessoa é creditada com mais frequência.
<b>birthYear</b>	no formato AAAA.
<b>deathYear</b>	no formato AAAA, se aplicável, caso contrário, “\N”.
<b>primaryProfession (matriz de strings)</b>	as 3 principais profissões da pessoa.

<b>knownForTitles (matriz de tconsts)</b>	títulos pelos quais a pessoa é conhecida.
---	---

### 3.1.2 title.basics.tsv.gz

Contém as seguintes informações para filmes:

Coluna	Descrição
<b>tconst (string)</b>	identificador alfanumérico exclusivo do título.
<b>titleType (string)</b>	o tipo/formato do título (por exemplo, filme, curta, série de TV, episódio de TV, vídeo, etc).
<b>primaryTitle (string)</b>	o título mais popular/o título usado pelos cineastas em materiais promocionais no momento do lançamento.
<b>originalTitle (string)</b>	título original, no idioma original.
<b>isAdult (booleano)</b>	0: título não adulto; 1: título adulto.
<b>startYear (YYYY)</b>	representa o ano de lançamento de um título. No caso de séries de TV, é o ano de início da série.
<b>endYear (YYYY)</b>	Ano final da série de TV. “\N” para todos os outros tipos de títulos.
<b>runtimeMinutes</b>	tempo de execução principal do título, em minutos.
<b>genres (array de strings)</b>	inclui até três gêneros associados ao título.

### 3.1.3 title.akas.tsv.gz

Contém as seguintes informações extras para filmes:

Coluna	Descrição
<b>titleId (string)</b>	um tconst que é um identificador alfanumérico exclusivo do título.
<b>ordenação (inteiro)</b>	um número para identificar exclusivamente as linhas para um determinado titleId.
<b>title (string)</b>	o título localizado.
<b>region (string)</b>	a região para esta versão do título.
<b>language (string)</b>	o idioma do título.
<b>types (array)</b>	Conjunto enumerado de atributos para este título alternativo. Um ou mais dos seguintes: “alternativo”, “dvd”, “festival”, “tv”, “vídeo”, “trabalho”, “original”, “imdbDisplay”. Novos valores poderão ser adicionados no futuro sem aviso prévio.
<b>attributes (array)</b>	Termos adicionais para descrever este título alternativo, não enumerados.
<b>isOriginalTitle (booleano)</b>	0: título não original; 1: título original.

### 3.1.4 Title.crew.tsv.gz

Contém informações do diretor e escritor de todos os títulos da IMDb. Os campos incluem:

Coluna	Descrição
<b>tconst (string)</b>	identificador alfanumérico exclusivo do título.
<b>directors (array de nconsts)</b>	diretor(es) do título determinado.
<b>writers (array de nconsts)</b>	escritor(es) do(s) título(s) fornecido(s).

### 3.1.5 title.episode.tsv.gz

Contém as informações do episódio de TV. Os campos incluem:

Coluna	Descrição
<b>tconst (string)</b>	identificador alfanumérico do episódio.
<b>parentTconst (string)</b>	identificador alfanumérico da série de TV pai.
<b>seasonNumber (inteiro)</b>	número da temporada à qual o episódio pertence.
<b>EpisodeNumber (inteiro)</b>	número do episódio do tconst da série de TV.

### 3.1.6 title.principais.tsv.gz

Contém o elenco/equipe principal dos títulos:

Coluna	Descrição
<b>tconst (string)</b>	identificador alfanumérico exclusivo do título.
<b>ordering (inteiro)</b>	um número para identificar exclusivamente as linhas para um determinado titleId.
<b>nconst (string)</b>	identificador alfanumérico exclusivo do nome/pessoa.
<b>category (string)</b>	a categoria do trabalho em que a pessoa estava.
<b>job (string)</b>	o cargo específico, se aplicável, caso contrário, “\N”.
<b>characters (string)</b>	o nome do personagem interpretado, se aplicável, caso contrário “\N” (é realmente “[role1,role2,...]” ou “\N”).

### 3.1.7 title.ratings.tsv.gz

Contém a classificação da IMDb e informações de votos para títulos:

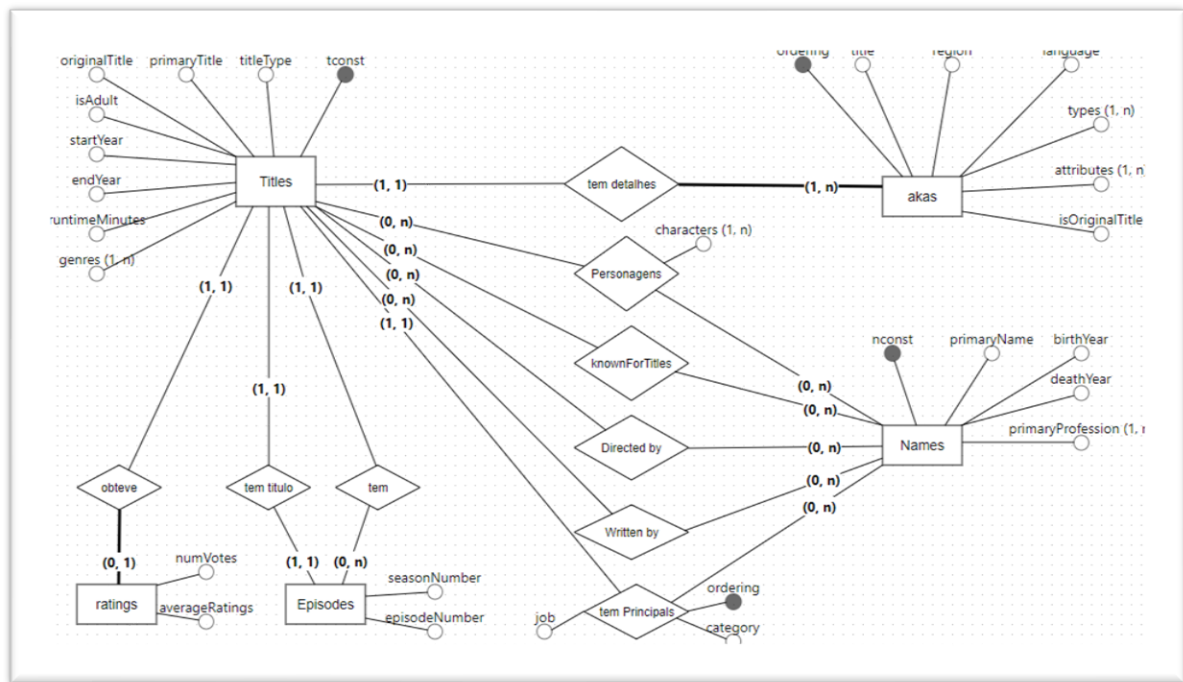
Coluna	Descrição
<b>tconst (string)</b>	identificador alfanumérico exclusivo do título.
<b>AverageRating</b>	média ponderada de todas as avaliações individuais dos usuários.
<b>numVotes</b>	número de votos que o título recebeu.

## 4 Modelagem de dados

O objetivo principal do projeto é responder perguntas ligados aos fatos “IMDBRating” e “NumVotes”, segundo as dimensões tempo (ano), diretor, escritor, gênero do filme, linguagem, personagens e episódios, bem como outras questões relacionadas.

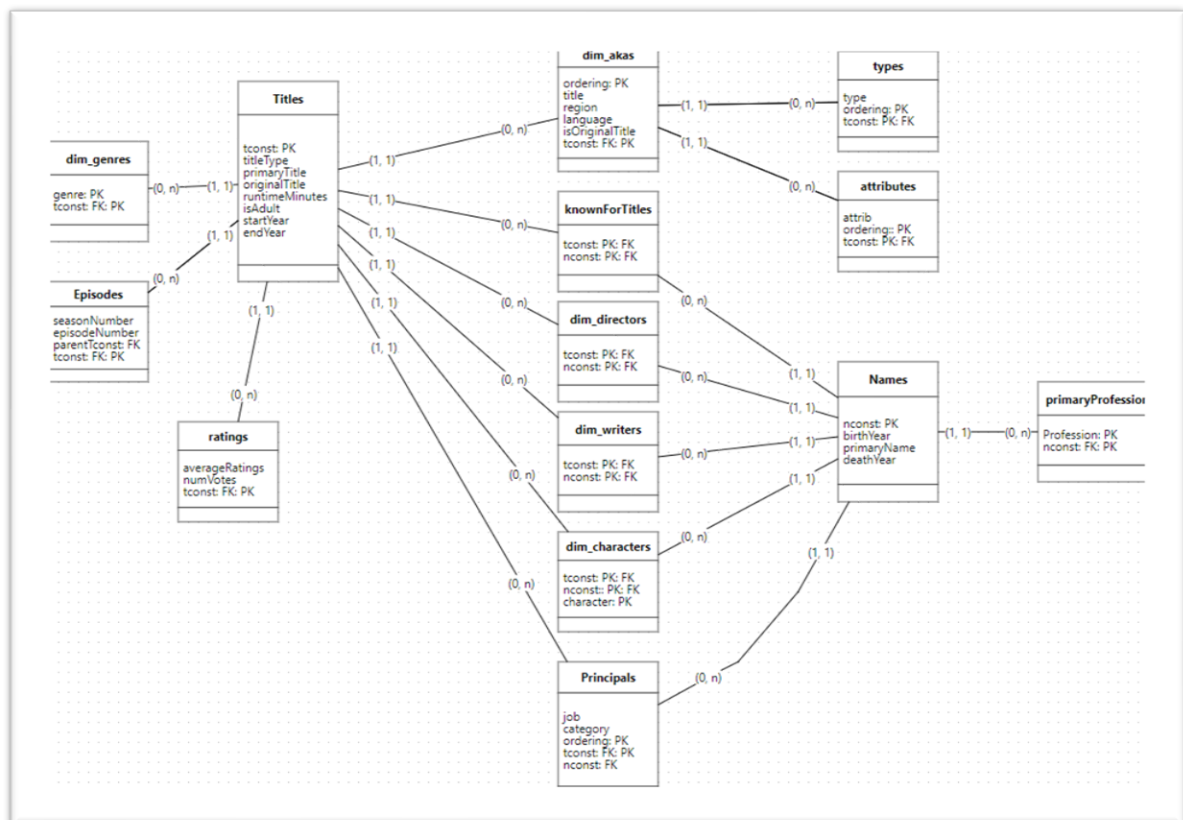
O modelo conceitual para alcançar estes objetivos é o seguinte:

*Figura 2 - modelo conceitual do projeto*



Já o modelo lógico derivado é o seguinte:

Figura 3 - modelo lógico do projeto



Este modelo lógico será utilizado como esquema de saída da transformação dos dados.

## 5 ETL

Como comentado no tópico “Visão geral”, para o processo de ETL foi utilizado o ADF. Para tal fim foi criada uma instância do ADF chamada “ccoemvpdfactory”, na qual será realizada a orquestração do ETL deste projeto.

### 5.1 Extração de dados

A primeira etapa foi a criação de um repositório no Azure de onde os arquivos serão ingeridos originalmente. O repositório foi criado por meio do serviço StorageAccount, nomeado “ccoemvpstorage” conforme a figura:

Figura 4 - Etapa de extração de Dados

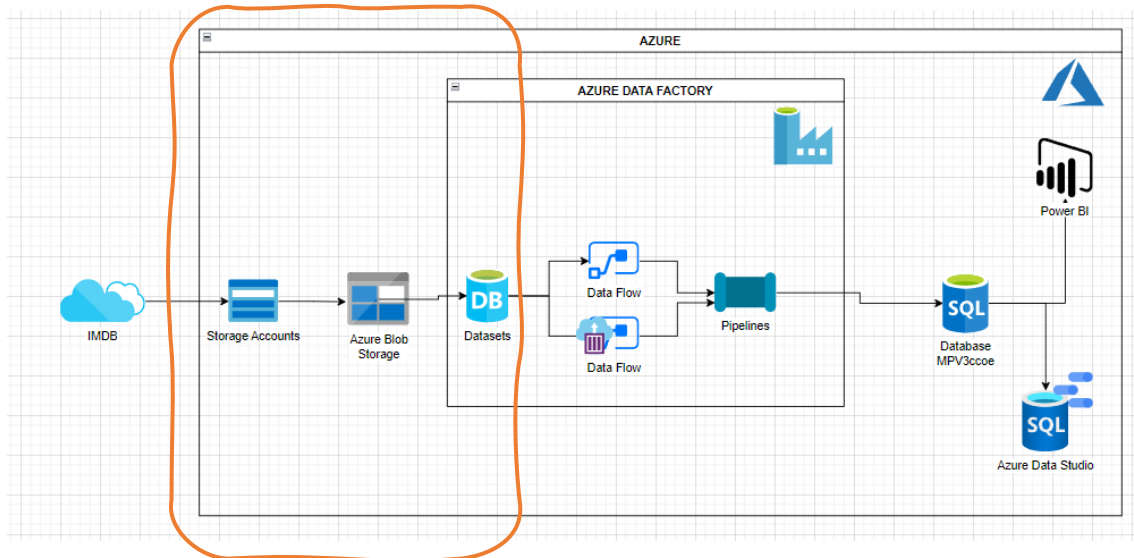
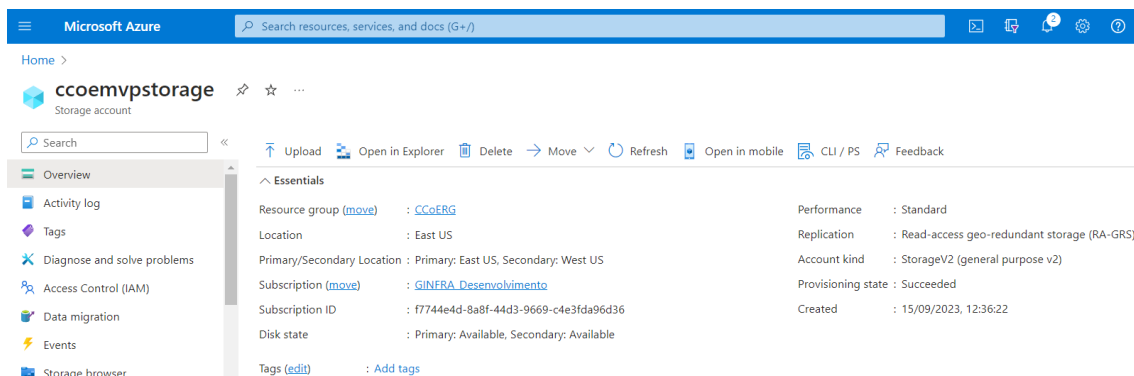


Figura 5 - storage account para repositório dos arquivos fontes

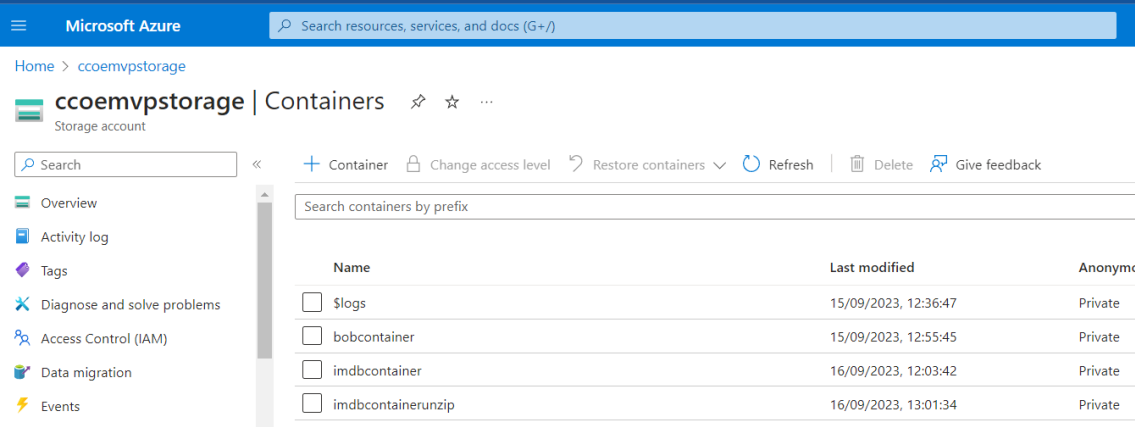


A seguir foram criados 2 containers: “imdbcontainer”, que contém os arquivos IMDB originais compactados, e o container “imdbcontainerunzip” que contém a versão descompactada dos arquivos.

A seguir foram criados 2 containers: “imdbcontainer” e “imdbcontainerunzip”.

Os arquivos originais compactados (extensão .gz) foram descompactados e carregados no storage Account “ccoempstorage” da seguinte forma: a versão compactada foi carregada no container “imdbcontainer”, e a versão descompactada foi carregada no container “imdbcontainerunzip”.

Figura 6 - Container com os arquivos fonte



Microsoft Azure		
Search resources, services, and docs (G+)		
Home > ccoempstorage		
ccoempstorage   Containers		
Storage account		
Search		
+ Container		
Change access level		
Restore containers		
Refresh		
Delete		
Give feedback		
Search containers by prefix		
Name	Last modified	Anonymc
<input type="checkbox"/> \$logs	15/09/2023, 12:36:47	Private
<input type="checkbox"/> bobcontainer	15/09/2023, 12:55:45	Private
<input type="checkbox"/> imdbcontainer	16/09/2023, 12:03:42	Private
<input type="checkbox"/> imdbcontainerunzip	16/09/2023, 13:01:34	Private

A figura a seguir evidencia a criação e o conteúdo do container “imdbcontainerunzip”.

Figura 7 - Container com arquivos IMDB descompactados



Home > ccoempvstorage | Containers >

**imdbcontainerunzip** ...

Container

Search

Upload Change access level Refresh Delete Change tier Acquire lease Break lease View snapshots

Overview

Diagnose and solve problems

Access Control (IAM)

Settings

Shared access tokens

Access policy

Properties

Metadata

Authentication method: Access key (Switch to Azure AD User Account)

Location: imdbcontainerunzip

Search blobs by prefix (case-sensitive)

Add filter

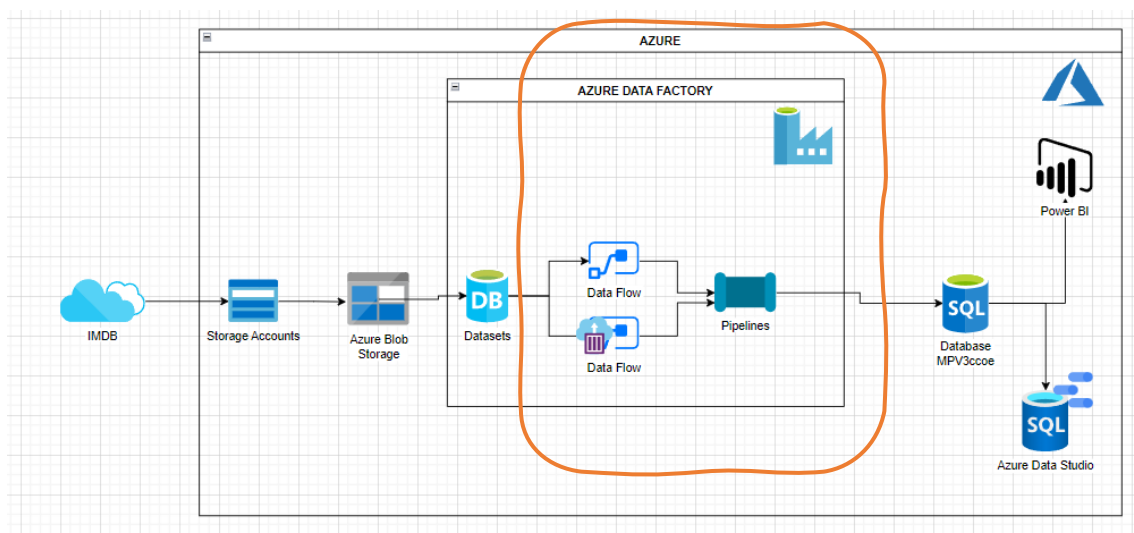
Name	Modified	Access tier	Archive status	Blob type
<input type="checkbox"/> name.basics.tsv	16/09/2023, 13:08:40	Hot (Inferred)		Block blob
<input type="checkbox"/> title.akas.tsv	16/09/2023, 13:10:47	Hot (Inferred)		Block blob
<input type="checkbox"/> title.basics.tsv	16/09/2023, 13:07:48	Hot (Inferred)		Block blob
<input type="checkbox"/> title.crew.tsv	16/09/2023, 13:04:26	Hot (Inferred)		Block blob
<input type="checkbox"/> title.episode.tsv	16/09/2023, 13:03:29	Hot (Inferred)		Block blob
<input type="checkbox"/> title.principals.tsv	16/09/2023, 13:11:53	Hot (Inferred)		Block blob
<input type="checkbox"/> title.ratings.tsv	16/09/2023, 13:03:56	Hot (Inferred)		Block blob

Para este projeto utilizaremos apenas os arquivos descompactados oriundos deste container.

## 5.2 Transformação de dados

A etapa de transformação dos dados está representadas na figura. Ele envolve basicamente a criação de fluxos de transformação de dados (Dataflow) que são agrupados e executados em uma estrutura chamada pipeline.

Figura 8 - Etapa de ETL



Para as transformações necessárias, foram criados 2 dataflows no processo de transformação dos dados: um para as tabelas de dimensões

chamado de “datalow\_dim” e um segundo chamado de “dataflow\_fact” para a tabela de fatos.

Para seleção dos atributos para as novas tabelas foi utilizado a técnica de projeção das colunas por meio do componente "Select".

Já o tratamento dados para estes campos, como mostrado na modelagem, foi de criar tabelas específicas par cada um deles. Para os campos multivalorados “genre” e “profession”, foram derivadas novas tabelas, por meio da utilização dos componentes “derived column” e "flatten". E para garantir que não ocorrência de campos nulos, foi utilizado o componente de filtragem de conteúdo "Filter".

Pela modelagem, houve a necessidade de derivar algumas tabelas de campos com valores específicos de ocorrência, como o cargo de diretor e escritor. Assim para criação das tabelas "dim\_diretor" e "dim\_escrito" foi utilizado o componente filtro de conteúdo no campo job, procurando especificamente pelos valores "Director" e "writer" respectivamente.

O tratamento dos dados para cada fluxo está detalhado a seguir.

#### 5.2.1 “dataflow\_fact”

O Dataflow “dataflow\_fact” para geração da tabela de fatos está apresentado abaixo:

Figura 9 - Fluxo Dataflow Facts



Para geração do dataflow de fatos, foram utilizadas como fonte as tabelas titleBasics e titleRatings (1)

Em cada uma delas foram projetadas as seguintes colunas (2)

- titleRatings (tconst, averageRating, numVotes)
- titleBasics (tconst, primarytitle, originaltitle, isAdult, startYear, endYear, runtimeMinutes)

Figura 10 - Datasources dataflow facts - Fatos



### 5.2.2 “dataflow\_dim”

O Dataflow “dataflow\_dim” para geração das tabelas de dimensão do projeto está apresentado na figura a seguir.

Figura 11 - Dataflow de dimensões



Para geração do dataflow de dimensões, foram utilizadas como fonte as tabelas titleBasics , titleEpisodes, names, titleprincipals (1)

Em algumas das tabelas foram projetadas as seguintes colunas (2)

- Tabela titleBasics: projetado o campo multivalorado “genre” para criação de uma tabela específica de gêneros de filmes;
- Tabela Names: projetadas as colunas “nconst”, “primaryName”, “birthDate”, “deathDate” para criação de uma tabela de apoio de nomes de pessoal;
- Tabela Names foi ainda projetado o campo multivalorado “profession” para criação de uma tabela específica de profissões na produção dos filmes.
- Da Tabela titlePrincipals foi projetado e transformado o campo multivalorado “characters” para criação de uma tabela específica de personagens de filmes;
- Da Tabela titlePrincipals foi projetado e transformado o campo “Director”, filtrando as linhas com a categoria “Director” para criação de uma tabela específica de diretores de filmes;
- Da Tabela titlePrincipals foram projetados os campos “tconst”, “ordering”, “nconst”, “category” e “job” para uso nas consultas;
- A tabela Episodes não sofreu o processo de projeção.

Para os campos multivalorados “genre” e “profession”, foram feitas diversas transformações para derivar novas tabelas dos campos multivalorados. (3)

Já para a tabela de personagens (character) foram feitas diversas transformações e limpas as possíveis ocorrências de nulos (null). (4)

Figura 12 - Dataflow\_dim - dimensões

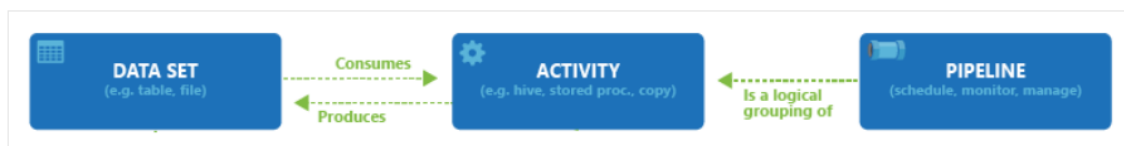


### 5.2.3 Pipelines

Um pipeline é um agrupamento lógico de atividades que juntas executam uma tarefa, que ingerem dados de um dataset e produzem novos dados. As atividades disponíveis são atividade de cópia, atividade de fluxo de dados

Desta forma as atividades de um pipeline definem as ações a serem executadas nos seus dados. No caso deste projeto, foram usadas atividades de fluxo de dados.

Figura 13 – visão geral do funcionamento de Pipelines (fonte:Microsoft)



Neste projeto foram criados 2 pipelines contendo cada um um dos fluxos principais dataflow\_facts ou dataflow\_dim. Um exemplo de execução de um deles está apresentado abaixo:

Figura 14 - exemplo de execução - Pipeline facts

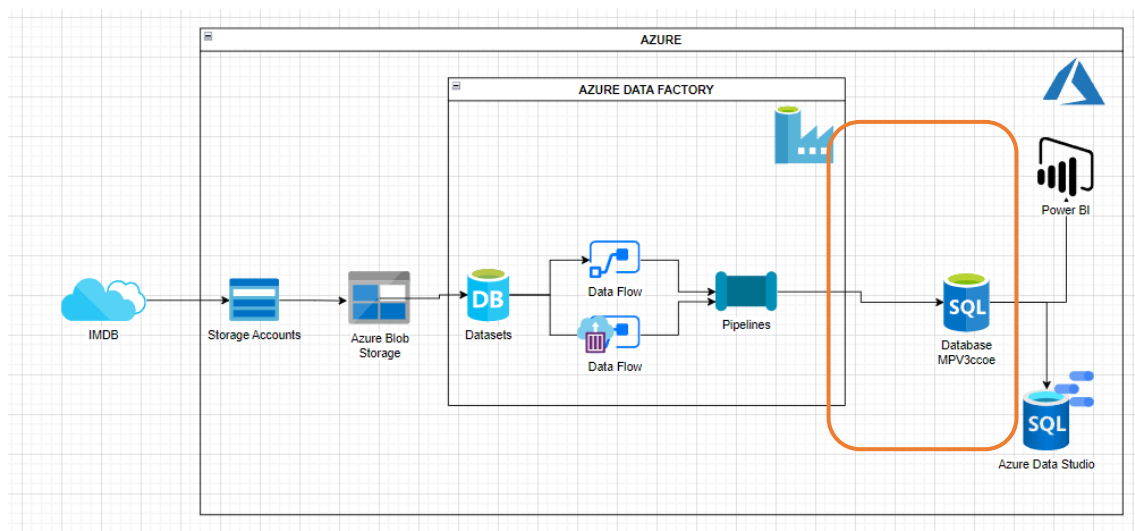
The screenshot displays the Azure Data Factory console. At the top, several tabs are visible: 'dataflow\_facts', 'pipeline\_dim', 'dataflow\_dim', 'pipeline\_facts', 'facts\_tabledb', 'dataflow\_principals', and 'Characters\_table'. The 'pipeline\_facts' tab is active, showing a 'Data flow' activity named 'dataflow\_facts' with a green checkmark indicating success. Below the activity, the 'Output' tab is selected, showing the 'Pipeline status' as 'Succeeded'. A message states: 'Data flow activity for this debug run will start as soon as the data flow debug session is ready.' Below this, a table lists the activity details:

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	User properties
dataflow_facts	Succeeded	Data flow	9/17/2023, 9:48:29 PM	4m 22s	AutoResolveIntegration	

### 5.3 Carga de dados

A etapa final do processo de ETL é a carga dos dados gerados em um repositório. Neste projeto o repositório é uma base SQL no AZURE.

Figura 15 - Etapa de carga de dados



Contudo o processo de carga exige a existência de um servidor e um banco de dados. Assim, o processo de carga de dados envolveu 3 fases distintas:

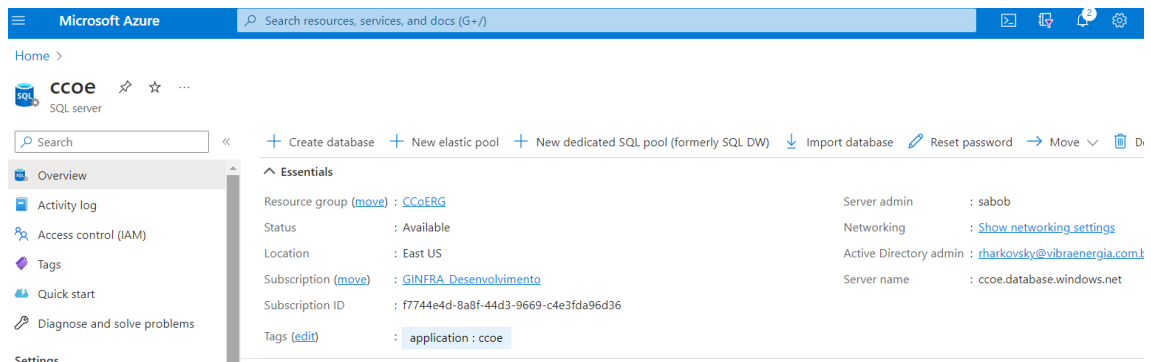
- Criação do database e esquemas no Azure
- Carga das tabelas (saída do ADF)
- Inclusão das restrições de chaves nas tabelas

Estas etapas estão descritas nos itens a seguir.

### 5.3.1 Criação do Database e Esquemas no Azures

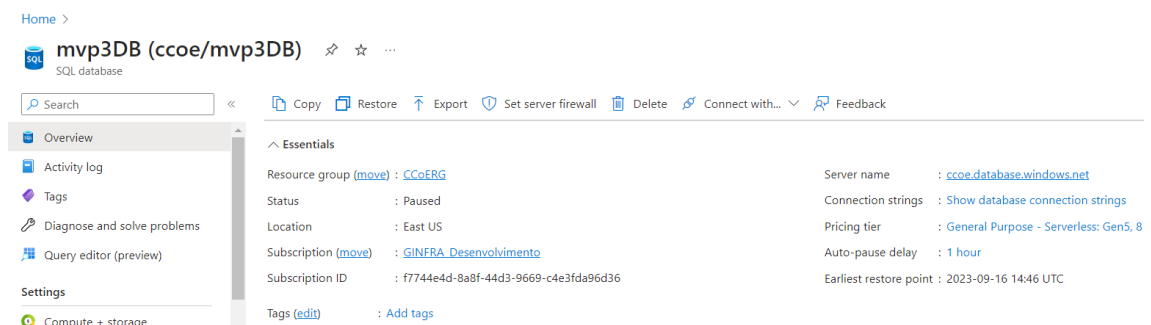
Para receber os dados oriundos do ETL foi criado no Azure um servidor de banco de dados SQL *serverless* chamado de “ccoe”, como evidenciado na figura a seguir.

Figura 16 - servidor de banco de dados “ccoe” no Azure



Em seguida, uma base de dados chamada de “mvp3DB” foi criada neste servidor (vide figura), para onde foram direcionados os dados de saída do modelo.

Figura 17 - Database no servidor "ccoe"



A próxima etapa consistiu em, a partir da modelagem realizada no item 4, proceder a criação propriamente dita das tabelas com as devidas restrições de chave primária e estrangeira. O script da figura foi elaborado e aplicado ao database, resultando na criação das tabelas.

*Figura 18 - Script de criação das tabelas*

```
CREATE TABLE title_facts
(
    tconst varchar(10) PRIMARY KEY,
    titleType varchar(30),
    primaryTitle varchar(max),
    originalTitle varchar(max),
    isAdult INT,
    startYear INT,
    endYear INT,
    runtimeMinutes INT,
);
CREATE TABLE title_names
(
    nconst varchar(10),
    primaryName varchar(150),
    birthYear INT,
    deathYear INT,
    CONSTRAINT pk_names PRIMARY KEY (nconst)
);
CREATE TABLE dim_Profession
(
    nconst varchar(10),
    profession varchar(30) ,
    CONSTRAINT pk_profession PRIMARY KEY (nconst, profession)
);
CREATE TABLE dim_principals
(
    tconst varchar(10),
    ordering INT,
    nconst varchar(10),
    job varchar(max),
    category varchar(60),
    CONSTRAINT pk_principals PRIMARY KEY (tconst,ordering)
)
CREATE TABLE dim_episodes
(
    tconst varchar(10) PRIMARY KEY,
    parentTconst varchar(10),
    seasonNumber INT,
    episodeNumber INT,
);
CREATE TABLE [dbo].[dim_directors]
(
    tconst varchar(10),
    nconst varchar(10),
    CONSTRAINT pk_Director PRIMARY KEY (tconst, nconst)
);
CREATE TABLE dim_Characters
(
```



```

tconst varchar(10),
nconst varchar(10),
characters varchar(30),
CONSTRAINT pk_Character PRIMARY KEY (tconst, nconst, characters)
);

CREATE TABLE dim_genres
(
tconst varchar(10),
genre varchar(30),
CONSTRAINT pk_genres PRIMARY KEY (tconst, genre)
);

CREATE TABLE title_Ratings
(
tconst varchar(10) PRIMARY KEY,
averageRating DECIMAL (5,1),
numVotes INT
);

CREATE TABLE dim_akas
(
tconst varchar(10) ,
ordering INT,
title varchar(max),
region varchar(10),
language varchar(5),
isOriginalTitle INT,
CONSTRAINT pk_akas PRIMARY KEY (tconst, ordering)
);

ALTER TABLE dim_episodes ADD FOREIGN KEY(parentTconst) REFERENCES title_facts (parentTconst)
ALTER TABLE dim_episodes ADD FOREIGN KEY(tconst) REFERENCES title_facts (tconst)

ALTER TABLE dim_principals ADD FOREIGN KEY(tconst) REFERENCES title_facts (tconst)
ALTER TABLE dim_principals ADD FOREIGN KEY(nconst) REFERENCES title_names (nconst)

ALTER TABLE dim_directors FOREIGN KEY(tconst) REFERENCES title_facts (tconst)
ALTER TABLE dim_directors FOREIGN KEY(nconst) REFERENCES title_names (nconst)

ALTER TABLE dim_Profession ADD FOREIGN KEY(nconst) REFERENCES title_facts (nconst)
ALTER TABLE dim_genres ADD FOREIGN KEY(genre) REFERENCES title_facts (genre)
ALTER TABLE dim_genres ADD FOREIGN KEY(tconst) REFERENCES title_facts (tconst)

ALTER TABLE dim_Characters ADD FOREIGN KEY(tconst) REFERENCES title_facts (tconst)
ALTER TABLE dim_Characters ADD FOREIGN KEY(nconst:) REFERENCES title_names (nconst)
ALTER TABLE title_Ratings ADD FOREIGN KEY(tconst) REFERENCES title_facts (tconst)

```



### 5.3.2 Carga das tabelas (saída ADF)

A saída do processo de extração e transformação dos dados foram tabelas SQL sem restrições, que foram armazenados num banco de dados SQL no Azure. Para tal, o componente “sink” do fluxo de transformação é responsável por apontar para o SQL server/database e carregar os dados na respectiva tabela. Nas figuras do item de “transformação” anteriormente apresentados, ele é o último componente como nome dim\*.

Antes de criar um dataset, é preciso criar um serviço para vincular o repositório de armazenamento de dados ao ADF. Assim, o componente SINK implementa este serviço de conexão, ou *linked service*, com a base de dados do servidor ccoe. A configuração deste *linked service* segue abaixo:

Figura 19 - linked server coma base de dados no servidor ccoe

### Edit linked service

 Azure SQL Database [Learn more](#) 

**Name \***

AzureSqlIDbMVP3

**Description**

Database MVP3

**Connect via integration runtime \*** ⓘ

AutoResolveIntegrationRuntime

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

**Account selection method** ⓘ

☐ From Azure subscription ☒ Enter manually

**Fully qualified domain name \***

ccoe.database.windows.net

**Database name \***

mvp3DB

**Authentication type \***


SQL authentication

**User name \***

sabob

**Password** **Azure Key Vault**

**Password \***

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

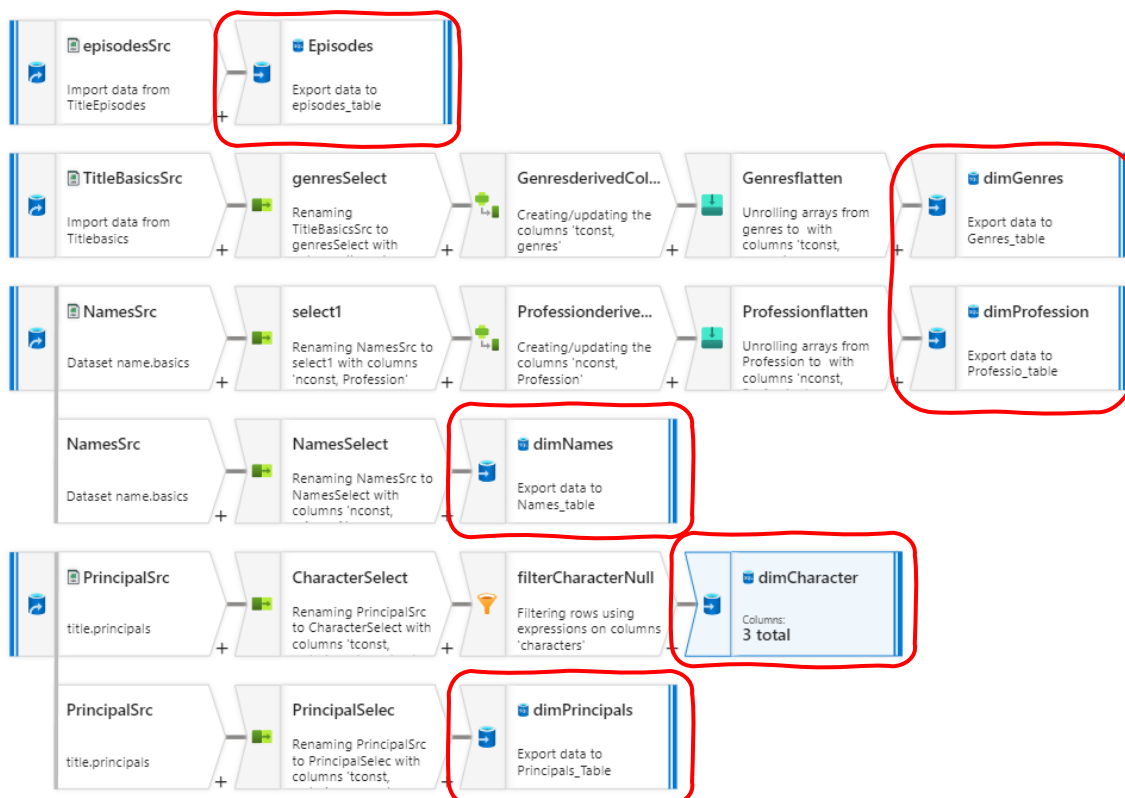
Para o fluxo de criação da tabela de fatos com o componente “Sink” está apresentado na figura, com nome “facts”.

Figura 20 - componente de saída da transformação SINK – tabela de fatos



Já para o fluxo de criação das tabelas de dimensão, temos vários sinks (um para cada tabela gerada) conforme mostrado na figura:

Figura 21 - componente de saída da transformação SINK – tabela de dimensões



### 5.3.3 Inclusão das restrições

A terceira etapa consistiu em carregar os dados das tabelas geradas pelo ADF (sem restrições) nas tabelas SQL criadas com as devidas restrições de chave Primária e estrangeira. Para isto foi utilizado o AZURE DATA STUDIO e comandos “INSERT INTO” tendo como origem as tabelas oriundas do ADF, renomeadas para “\*\_old”. O script utilizado está mostrado na figura a seguir.

Figura 22 - Script INSERT INTO de carga final das tabelas

```
--INSERT INTO
INSERT INTO [dbo].[dim_title_facts]
SELECT * FROM [dbo].[dim_title_facts_old];

INSERT INTO [dbo].[dim_title_names]
SELECT * FROM [dbo].[dim_title_names_old];

INSERT INTO [dbo].[dim_Profession]
SELECT * FROM [dbo].[dim_Profession_old];

INSERT INTO [dbo].[dim_principals]
SELECT * FROM [dbo].[dim_principals_old];

INSERT INTO [dbo].[dim_episodes]
SELECT * FROM [dbo].[dim_episodes_old];

INSERT INTO [dbo].[dim_directors]
SELECT * FROM [dbo].[dim_directors_old];

INSERT INTO [dbo].[dim_episodes]
SELECT * FROM [dbo].[dim_episodes_old];

INSERT INTO [dbo].[dim_Characters]
SELECT * FROM [dbo].[dim_Characters_old];

INSERT INTO [dbo].[dim_genres]
SELECT * FROM [dbo].[dim_dim_genres_old];

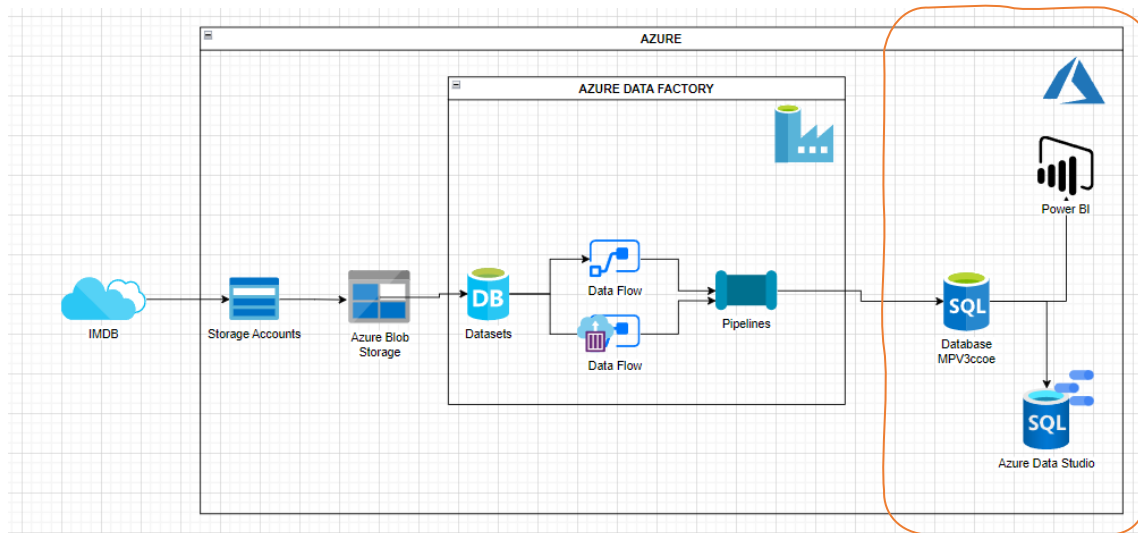
INSERT INTO [dbo].[title_Ratings]
SELECT * FROM [dbo].[title_Ratings_old];
```

## 6 Resultados e Visualização de dados

Para obtenção das respostas as perguntas propostas, foram utilizadas 2 ferramentas: Azure Data Studio, para gerar as tabelas de resposta e Power BI, para geração dos gráficos. Ambos os serviços forma conectados diretamente no Azure SQL Database.

Nas respostas abaixo consideramos “popularidade” como número de votos do filme (quanto maior o número de votos, mais popular é o filme) e “pontuação” como o valor do IMDB obtido (0 a 10).

Figura 23 - Azure data Studio e Power BI



Consideramos ainda que para termos uma pontuação razoável um mínimo de 10.000 votos precisa ser obtido.

## 6.1 Qual a popularidade dos filmes de James Bond

```

SELECT f.primaryTitle, SUM(r.numVotes) as Popularidade
FROM title_facts f INNER JOIN dim_Characters c on f.tconst=c.tconst
      INNER JOIN title_ratings r ON r.tconst= c.tconst
WHERE c.characters like 'James Bond' and f.titleType='movie' and r.numVotes>10000
GROUP by f.primaryTitle
ORDER BY Popularidade DESC
  
```

primaryTitle	Popularidade
Skyfall	716762
Casino Royale	678667
Quantum of Solace	462211
Spectre	456015
No Time to Die	428417
GoldenEye	265240
Die Another Day	225289
The World Is Not Enough	206468

Tomorrow Never Dies	201028
Goldfinger	197809
Dr. No	174824
From Russia with Love	141276
Thunderball	123931
You Only Live Twice	114603
The Spy Who Loved Me	113449
Live and Let Die	112628
Diamonds Are Forever	111339
Octopussy	110418
The Man with the Golden Gun	110394
Licence to Kill	109378
Moonraker	105995
For Your Eyes Only	105698
The Living Daylights	103325
A View to a Kill	102306
On Her Majesty's Secret Service	96554
Never Say Never Again	71231

## 6.2 Qual a pontuação dos filmes de James Bond

```
SELECT f.primaryTitle, r.averageRating as Pontuacao
```

```
FROM title_facts f INNER JOIN dim_Characters c ON f.tconst=c.tconst
```

```
INNER JOIN title_ratings r ON r.tconst= c.tconst
```

```
WHERE c.characters LIKE 'James Bond' and f.titleType='movie' and r.numVotes>10000
```

```
ORDER BY Pontuacao Desc
```

primaryTitle	Pontuacao
Casino Royale	8.0
Skyfall	7.8
Goldfinger	7.7
From Russia with Love	7.3
No Time to Die	7.3
GoldenEye	7.2
Dr. No	7.2
The Spy Who Loved Me	7.0
Thunderball	6.9
You Only Live Twice	6.8
Spectre	6.8
The Living Daylights	6.7
For Your Eyes Only	6.7
On Her Majesty's Secret Service	6.7
Live and Let Die	6.7
The Man with the Golden Gun	6.7
Quantum of Solace	6.6

Licence to Kill	6.6
Diamonds Are Forever	6.5
Octopussy	6.5
Tomorrow Never Dies	6.5
The World Is Not Enough	6.4
A View to a Kill	6.3
Moonraker	6.2
Die Another Day	6.1
Never Say Never Again	6.1

### 6.3 Quais são os gêneros de filmes mais populares

SELECT g.genre, SUM(r.numVotes) As Popularidade

FROM dim\_genres g INNER JOIN title\_Ratings r on g.tconst=r.tconst

GROUP BY g.genre

ORDER BY NumFilmes DESC

genre	Popularidade
Drama	727666892
Action	452560351
Comedy	425886436
Adventure	361059088
Crime	281332739
Thriller	204725179
Romance	158374971
Sci-Fi	155602172
Mystery	154278735
Horror	126988249
Fantasy	126255289
Animation	115494690
Biography	77677171
Family	63451833
History	39278113
Music	27457243
Documentary	26997430
War	26625783
Sport	21396309
Western	12207588
Musical	11015312
Short	9511853
Reality-TV	4474116
Film-Noir	3934127
Talk-Show	2241707
Game-Show	1863926
News	1299681

<b>Adult</b>	849689
--------------	--------

6.4 Quais os gêneros com as melhores pontuações (acima de 6.0)

```
SELECT g.genre, CAST(AVG(r.averageRating) AS DECIMAL(5,2)) As MediaPontuação
FROM dim_genres g INNER JOIN title_Ratings r on g.tconst=r.tconst
WHERE r.numVotes>10000
GROUP BY g.genre
HAVING (AVG(r.averageRating)>6)
ORDER BY MediaPontuação DESC
```

genre	MediaPontuação
News	8.06
Talk-Show	7.67
Film-Noir	7.67
Game-Show	7.59
Short	7.55
Documentary	7.54
Animation	7.52
History	7.31
War	7.29
Biography	7.22
Western	7.17
Drama	7.15
Crime	7.08
Adventure	7.01
Musical	6.93
Sport	6.88
Music	6.86
Action	6.86
Mystery	6.82
Romance	6.71
Comedy	6.70
Fantasy	6.69
Thriller	6.69
Reality-TV	6.65
Sci-Fi	6.58
Family	6.54
Horror	6.29

6.5 Quais os 10 filmes com maior popularidade de Steven Spielberg

```
SELECT top (10) f.primaryTitle, n.primaryName, r.numvotes as Popularidade
```



```

FROM title_ratings r INNER JOIN title_facts f on f.tconst=r.tconst
      INNER JOIN dim_director d on d.tconst=f.tconst
      INNER JOIN title_names n ON n.nconst=d.nconst
WHERE n.primaryName='Steven Spielberg'
ORDER BY r.numVotes desc

```

primaryTitle	primaryName	Popularidade
Saving Private Ryan	Steven Spielberg	1448909
Schindler's List	Steven Spielberg	1406454
Catch Me If You Can	Steven Spielberg	1047055
Jurassic Park	Steven Spielberg	1033739
Indiana Jones and the Raiders of the Lost Ark	Steven Spielberg	1012521
Indiana Jones and the Last Crusade	Steven Spielberg	791216
Jaws	Steven Spielberg	639216
Minority Report	Steven Spielberg	571395
Indiana Jones and the Temple of Doom	Steven Spielberg	522033
Indiana Jones and the Kingdom of the Crystal Skull	Steven Spielberg	480879

6.6 Quais os 10 filmes com maior pontuação de Steven Spielberg

```

SELECT TOP (10) f.primaryTitle, n.primaryName, r.averagerating as Pontuacao
FROM title_ratings r INNER JOIN title_facts f ON f.tconst=r.tconst
      INNER JOIN dim_director d ON d.tconst=f.tconst
      INNER JOIN title_names n ON n.nconst=d.nconst
WHERE n.primaryName='Steven Spielberg'
ORDER BY r.averagerating DESC

```

primaryTitle	primaryName	Pontuacao
Schindler's List	Steven Spielberg	9.0
Saving Private Ryan	Steven Spielberg	8.6
Indiana Jones and the Raiders of the Lost Ark	Steven Spielberg	8.4
Indiana Jones and the Last Crusade	Steven Spielberg	8.2
Jurassic Park	Steven Spielberg	8.2
Catch Me If You Can	Steven Spielberg	8.1
Jaws	Steven Spielberg	8.1
E.T. the Extra-Terrestrial	Steven Spielberg	7.9
Murder by the Book	Steven Spielberg	7.7
The Color Purple	Steven Spielberg	7.7

6.7 Quais os 10 diretores de filmes com maiores médias de pontuação com mais de 5 filmes realizados?

```
SELECT top (10) n.primaryName, CAST(AVG(r.averagerating) AS DECIMAL(10,2)) as
MediaPontuaçãoDiretor, count(*) as NumFilmes
FROM title_ratings r INNER JOIN title_facts f ON f.tconst=r.tconst
      INNER JOIN dim_director d ON d.tconst=f.tconst
      INNER JOIN title_names n ON n.nconst=d.nconst
WHERE NumVotes >10000 and f.titleType='movie'
GROUP BY n.primaryName
HAVING (COUNT(*) > 5)
ORDER BY MediaPontuaçãoDiretor DESC
```

primaryName	MediaPontuaçãoDiretor	NumFilmes
Ertem Egilmez	8.74	7
Sergio Leone	8.22	6
Christopher Nolan	8.20	12
Akira Kurosawa	8.06	17
Andrei Tarkovsky	8.00	7
Hayao Miyazaki	7.96	11
Quentin Tarantino	7.94	14
Frank Capra	7.94	8
Fritz Lang	7.93	7
Mani Ratnam	7.93	8

6.8 Quais os são 10 diretores de filmes mais populares?

```
SELECT TOP (10) n.primaryName, SUM(r.Numvotes) as PopularidadeDiretor, count(*)
as NumFilmes
FROM title_ratings r INNER JOIN title_facts f ON f.tconst=r.tconst
      INNER JOIN dim_director d ON d.tconst=f.tconst
      INNER JOIN title_names n ON n.nconst=d.nconst
WHERE NumVotes >10000 and f.titleType='movie'
GROUP BY n.primaryName
HAVING (COUNT(*) > 5)
ORDER BY PopularidadeDiretor DESC
```

primaryName	PopularidadeDiretor	NumFilmes
Christopher Nolan	15330770	12

Steven Spielberg	14115069	34
Quentin Tarantino	11505168	14
Martin Scorsese	9940049	28
Peter Jackson	8663631	14
David Fincher	8556649	11
Ridley Scott	8445499	27
Robert Zemeckis	7252067	20
James Cameron	6306824	8
Tim Burton	5422941	19

6.9 Qual é o tempo de execução típico para filmes de cada gênero?

```
SELECT g.genre, AVG(f.runtimeMinutes) AS MediaMinutos
FROM dim_genres g INNER JOIN title_facts f ON g.tconst=f.tconst
WHERE f.titleType='movie'
GROUP BY g.genre
ORDER BY g.genre
```

genre	MediaMinutos
Action	100
Adult	80
Adventure	92
Animation	81
Biography	88
Comedy	92
Crime	94
Documentary	78
Drama	96
Family	91
Fantasy	93
Film-Noir	82
Game-Show	45
History	93
Horror	88
Music	87
Musical	98
Mystery	93
News	75
Reality-TV	84
Romance	98
Sci-Fi	90
Sport	88
Talk-Show	80
Thriller	95
War	95

Western	76
---------	----

6.10 Quantos filmes foram feitos de cada gênero por ano entre 2020/22?

```
SELECT f.startYear, g.genre, COUNT(*) as FilmesporGenero
from dim_genres g inner join title_facts f on g.tconst=f.tconst
Where f.titleType='movie' and f.startYear is not null
GROUP BY f.startYear, g.genre
Order by f.startYear DESC
```

startYear	genre	FilmesporGenero
2020	Biography	427
2020	Thriller	1204
2020	Game-Show	3
2020	Western	53
2020	Action	1013
2020	Documentary	5167
2020	Fantasy	398
2020	Animation	353
2020	Romance	801
2020	Music	441
2020	Sport	313
2020	Drama	4972
2020	News	14
2020	Musical	150
2020	War	150
2020	Adventure	555
2020	Mystery	481
2020	Adult	203
2020	Talk-Show	24
2020	Reality-TV	74
2020	Horror	1245
2020	Family	428
2020	Sci-Fi	384
2020	History	322
2020	Crime	650
2020	Comedy	2260
2021	News	10
2021	Action	1295
2021	Drama	5955
2021	Musical	187
2021	Mystery	542
2021	Game-Show	2
2021	Western	61

2021	Horror	1401
2021	Sci-Fi	441
2021	Comedy	2461
2021	Romance	935
2021	Fantasy	438
2021	Sport	336
2021	Music	434
2021	Crime	796
2021	Talk-Show	21
2021	Animation	431
2021	Family	428
2021	Reality-TV	35
2021	History	325
2021	War	109
2021	Thriller	1461
2021	Biography	504
2021	Adult	111
2021	Adventure	598
2021	Documentary	5448
2022	Talk-Show	10
2022	Mystery	650
2022	Musical	177
2022	Biography	502
2022	Fantasy	486
2022	Family	513
2022	Thriller	1859
2022	Action	1396
2022	Adventure	660
2022	Western	70
2022	Animation	391
2022	Sci-Fi	456
2022	News	11
2022	Romance	1109
2022	Horror	1640
2022	Documentary	5270
2022	Music	383
2022	War	117
2022	History	357
2022	Reality-TV	30
2022	Crime	958
2022	Sport	283
2022	Adult	86
2022	Game-Show	3
2022	Drama	6608
2022	Comedy	3114

6.11 Quem são os atores que interpretaram ‘James Bond’ em um filme?

```
SELECT f.primaryTitle, f.originalTitle, n.primaryName, characters, f.startYear
FROM title_names n inner join dim_Characters c on c.nconst=n.nconst
      join title_facts f on f.tconst=c.tconst
      join title_ratings r on r.tconst=f.tconst
WHERE c.characters like 'James Bond' and f.titleType='movie' and r.numVotes>10000
ORDER BY f.startYear
```

originalTitle	primaryName	characters	startYear
<b>Dr. No</b>	Sean Connery	James Bond	1962
<b>From Russia with Love</b>	Sean Connery	James Bond	1963
<b>Goldfinger</b>	Sean Connery	James Bond	1964
<b>Thunderball</b>	Sean Connery	James Bond	1965
<b>You Only Live Twice</b>	Sean Connery	James Bond	1967
<b>On Her Majesty's Secret Service</b>	George Lazenby	James Bond	1969
<b>Diamonds Are Forever</b>	Sean Connery	James Bond	1971
<b>Live and Let Die</b>	Roger Moore	James Bond	1973
<b>The Man with the Golden Gun</b>	Roger Moore	James Bond	1974
<b>The Spy Who Loved Me</b>	Roger Moore	James Bond	1977
<b>Moonraker</b>	Roger Moore	James Bond	1979
<b>For Your Eyes Only</b>	Roger Moore	James Bond	1981
<b>Never Say Never Again</b>	Sean Connery	James Bond	1983
<b>Octopussy</b>	Roger Moore	James Bond	1983
<b>A View to a Kill</b>	Roger Moore	James Bond	1985
<b>The Living Daylights</b>	Timothy Dalton	James Bond	1987
<b>Licence to Kill</b>	Timothy Dalton	James Bond	1989
<b>GoldenEye</b>	Pierce Brosnan	James Bond	1995
<b>Tomorrow Never Dies</b>	Pierce Brosnan	James Bond	1997
<b>The World Is Not Enough</b>	Pierce Brosnan	James Bond	1999
<b>Die Another Day</b>	Pierce Brosnan	James Bond	2002
<b>Casino Royale</b>	Daniel Craig	James Bond	2006
<b>Quantum of Solace</b>	Daniel Craig	James Bond	2008
<b>Skyfall</b>	Daniel Craig	James Bond	2012
<b>Spectre</b>	Daniel Craig	James Bond	2015
<b>No Time to Die</b>	Daniel Craig	James Bond	2021

6.12 Quantas vezes eles fizeram o papel de ‘James Bond’?

```
SELECT n.primaryName, count(*) as filmesRealizados
FROM title_names n inner join dim_Characters c on c.nconst=n.nconst
```

```

join title_facts f on f.tconst=c.tconst
      join title_ratings r on r.tconst=f.tconst
WHERE c.characters like 'James Bond' and f.titleType='movie' and r.numVotes>10000
group by n.primaryName

```

primaryName	filmesRealizados
Daniel Craig	5
George Lazenby	1
Pierce Brosnan	4
Roger Moore	7
Sean Connery	7
Timothy Dalton	2

6.13 Quantos filmes existem em cada gênero?

```

SELECT genre, COUNT(*) As NumFilmes
FROM dim_genres
GROUP BY genre

```

genre	NumFilmes
Animation	513029
Sci-Fi	109979
Western	30148
Family	750829
Thriller	170300
Musical	88704
Film-Noir	886
Talk-Show	1238672
History	148923
Documentary	956911
Horror	181758
Game-Show	356825
Adult	315756
Fantasy	206108
Biography	108840
Comedy	2033882
Crime	417089
Action	419221
Mystery	203965
Reality-TV	564381
Drama	2888489
Romance	969600
News	892443
Sport	240835
Short	1113099

<b>Adventure</b>	398238
<b>War</b>	34360
<b>Music</b>	390803

## 7 Conclusão

O objetivo deste projeto foi o de realizar uma análise dos títulos (filmes, seriados de TV) publicados e responder a diversas perguntas mais comuns sobre o mercado de mídia, utilizando as informações disponibilizadas pela plataforma IMDB.

A execução deste projeto envolveu a realização das seguintes atividades:

- Entender os dados no database disponibilizado pelo IMDB;
- Modelar o banco de dados usando a técnica o modelo Entidade-Relacionamento (ER) e diagramas de esquema lógico relacional;
- Projetar um banco de dados relacional para ingestão dos dados;
- Criar um servidor relacional AZURE SQL Server, e um banco de dados relacional com as devidas restrições;
- Criar um repositório de dados no Azure e armazenar os dados fonte oriundos do IMDB;
- Realizar um ETL, utilizando o Azure Data Factory, onde extraímos os dados dos arquivos tsv (separados por tabulações), e transformando-os, e carregando-os em tabelas normalizadas e reestruturadas, segundo a modelagem prevista;
- Carregar os dados gerados nas tabelas com restrições de chave primária e estrangeira, e;
- Responder as questões colocadas utilizando o AzureData Studio e Visualizar as respostas utilizando POWER BI.

Este projeto mostrou apenas algumas possibilidades do que pode ser feito com esses dados do IMDB. Com estes dados poderíamos realizar diversas outras análises. Uma adição interessante a estes dados seria a inclusão dos dados das bilheterias obtidas pelos títulos.



Por fim, uma possível extensão do uso destes dados seria investigar mais detalhadamente as tendências, realizando análises estatísticas e possivelmente até usando alguns algoritmos de aprendizado de máquina.