**SPOTIFY DATAWAREHOUSE & DASHBOARD DOCUMENTATION**

Power BI Report:

<https://app.powerbi.com/view?r=eyJrIjoiYTkzMWViZmItZDM4MS00MjkxLWFjNDEtNzU3M2IxYTRkZmRlIiwidCI6IjE3ZjFhODdlLTJhMjUtNGVhYS1iOWRmLTlkNDM5MDM0YjA4MCIsImMiOjF9>

Business Requirements (Defining the Grain):

For each song in the playlists "Top 50 Songs Weekly US & Global," "Top 50 Daily US & Global," and "Top 50 Viral US & Global," we aim to track changes in their popularity scores and explore the corresponding audio features that influence these scores.

Spotify Popularity is a crucial metric, rated on a scale of 0 to 100, used internally by Spotify to rank artists and tracks. It primarily depends on factors like save rates, shares, and the number of plays. However, our dashboard focuses on analyzing the trends in audio feature changes concerning the average values of these features across songs within these playlists. We also aim to comprehend how these changes relate to shifts in the popularity index for each song.

This analysis helps us gain insights into how audio features contribute to the rise or fall of a song's popularity score across these influential playlists.

Data Sources and Dimensional Modeling & Views

The data is first ingested into a staging table under the STAGING schema in SNOWFLAKE it consists of the following fields for each song: artist\_id, artist\_name, song\_id, song\_name, album\_id, album\_name, release\_date, total\_tracks, url, song\_id, url, duration\_ms, popularity, song\_added, artist\_id, uri, artist\_name, external\_url, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, type, track\_href, analysis\_url, time\_signature, source, new\_release

Data is entirely extracted via the permitted data fields from the Spotify API using the Spotipy python package. Below is the list of tables and the data dictionary:

**DIM\_ARTISTS**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SURR\_KEY | BIGINT | Auto incremented surrogate key |
| ARTIST\_ID | VARCHAR | Unique Spotify ID for each Artist |
| ARTIST\_NAME | VARCHAR | Name of the Artist |
| ARTIST\_IMAGE\_URL | VARCHAR | URL of the artist from spotify api |
| EFFECTIVE\_DATE | DATE | ETL creation date |
| EXPIRY\_DATE | DATE | Date the record loses validity |
| IS\_CURRENT | VARCHAR | SCD 2 INDICATOR |

**DIM\_ALBUMS**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SURR\_KEY | BIGINT | Auto incremented surrogate key |
| ALBUM\_ID | VARCHAR | Unique Spotify ID of Album |
| ALBUM\_NAME | VARCHAR | Name of the Album |
| ALBUM\_URL | DATE | Album url from spotify API |
| RELEASE\_DATE | DATE | Album release date |
| TOTAL\_TRACK | INTEGER | Total number of tracks in the album |
| EFFECTIVE\_DATE | VARCHAR | ETL creation date |

**DIM\_SONGS**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SURR\_KEY | BIGINT | Auto incremented surrogate key |
| ARTIST\_ID | VARCHAR | Unique Spotify ID of the artist |
| SONG\_ID | VARCHAR | Unique Spotify ID of the track |
| SONG\_NAME | VARCHAR | Name of the Song |
| EFFECTIVE\_DATE | DATE | ETL creation date |

**DIM\_AUDIOFEATURES**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SURR\_KEY | BIGINT | Auto incremented surrogate key |
| SONG\_NAME | VARCHAR | Name of the song |
| DURATION\_MS | INT | The duration of the track in milliseconds. |
| URI | VARCHAR | The Spotify URI for the track. |
| DANCEABILITY | FLOAT | Danceability describes how suitable a track  is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable. |
| ENERGY | FLOAT | Energy is a measure from 0.0 to 1.0 and  represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. |
| KEY | FLOAT | The key the track is in |
| LOUDNESS | FLOAT | The overall loudness of a track in decibels (dB).  Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks |
| MODE | FLOAT | Mode indicates the modality (major or minor) of a track,  the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0. |
| SPEECHINESS | FLOAT | Speechiness detects the presence of spoken words in a track.  The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. |
| ACOUSTICNESS | FLOAT | A confidence measure from 0.0 to 1.0 of whether the track is acoustic.  1.0 represents high confidence the track is acoustic. |
| INSTRUMENTALNESS | FLOAT | Predicts whether a track contains no vocals.  "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". |
| LIVENESS | FLOAT | Detects the presence of an audience in the recording.  Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live. |
| VALENCE | FLOAT | The overall loudness of a track in decibels (dB).  Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. |
| TEMPO | FLOAT | The overall estimated tempo of a track in beats per minute (BPM).  In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration. |
| TIME\_SIGNATURE | FLOAT | The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure). The time signature ranges from 3 to 7 indicating time signatures of "3/4", to "7/4". |
| EFFECTIVE\_DATE | FLOAT | ETL create date |
| EXPIRY\_DATE | FLOAT | ETL updated date based on slowly changing dimension |
| IS\_CURRENT | FLOAT | SCD 2 indicator |

**TRANSACTION\_FACT** (Periodically updated snapshots)

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Surr\_key | BIGINT | Auto incremented surrogate key |
| ARTIST\_ID | STRING | Unique Spotify ID of the Artist |
| ALBUM\_ID | VARCHAR | Unique Album ID of the Artist |
| SONG\_ID | VARCHAR | Unique Song ID of the Artist |
| PLAYLIST | VARCHAR | The features playlist the song belongs to |
| POPULARITY | INTEGER | Popularity Score of the song |
| EFFECTIVE\_DATE | DATE | ETL creation date |

**Views:**

**CURRENT\_POPULARITY (view consists of songs and their respective popularity scores)**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SONG\_ID | BIGINT | Spotify Song ID. |
| POPULARITY | VARCHAR | Popularity score of the song. |
| PLAYLIST | VARCHAR | The playlist the song belongs to. |

**DROPPED\_FROM\_PLAYLIST** (view provides a list of songs that get dropped from the playlists, updated daily)

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SONG\_ID | BIGINT | Spotify Song ID |
| SONG\_NAME | VARCHAR | Name of the Song |
| PLAYLIST | VARCHAR | The playlist the song belongs to |
| EXPIRY\_DATE | DATE | Last date the song was in the playlist |

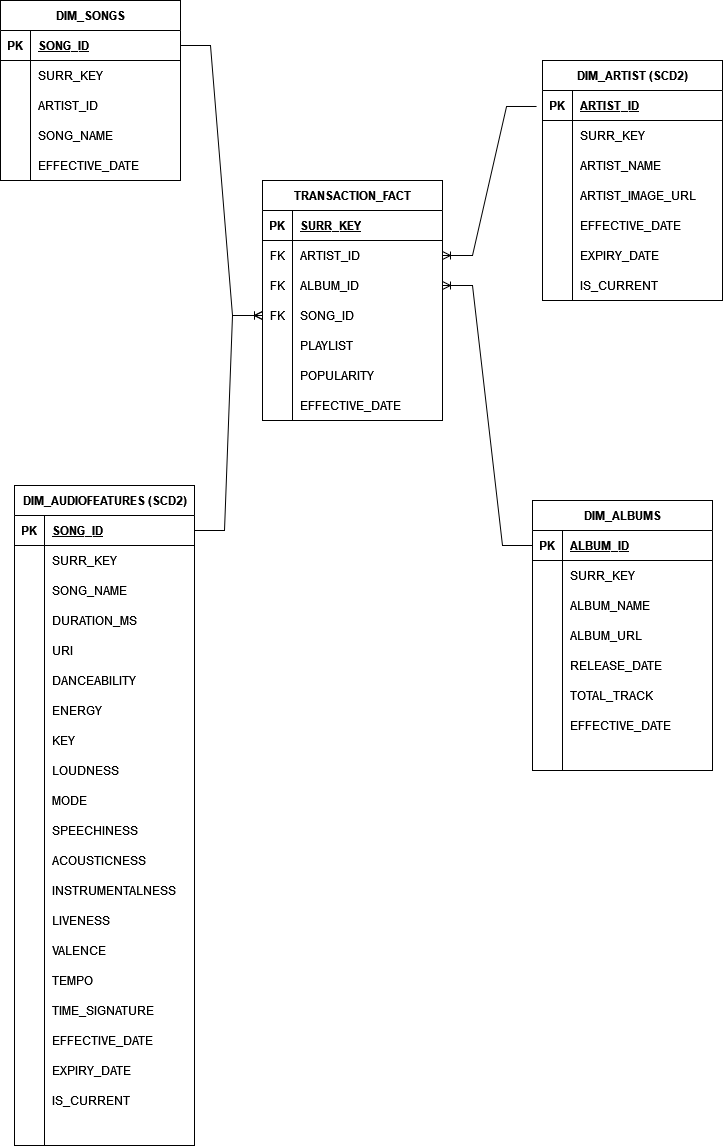
**NEW\_RELEASE\_INPLAYLIST** (New songs/albums that are included in the playlist)

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| ALBUM\_NAME | VARCHAR | Name of the album |
| SONG\_NAME | VARCHAR | Name of the song |
| PLAYLIST | VARCHAR | Name of the playlist |

**TRANSFACT\_LASTTENDAYS** (view consisting of last 10 days from the TRANSACTION\_FACT table)

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Surr\_key | BIGINT | Auto incremented surrogate key |
| ARTIST\_ID | STRING | Unique Spotify ID of the Artist |
| ALBUM\_ID | VARCHAR | Unique Album ID of the Artist |
| SONG\_ID | VARCHAR | Unique Song ID of the Artist |
| PLAYLIST | VARCHAR | The features playlist the song belongs to |
| POPULARITY | INTEGER | Popularity Score of the song |
| EFFECTIVE\_DATE | DATE | ETL creation date |

**E-R Diagram:**



**SnowPipe Configuration with Blob Storage: (AZURE\_PIPE)**

Please refer to the following video for steps on how to setup the storage que and integrate it with snopipe:

https://www.youtube.com/watch?v=\_xKfJzL\_Bz0

**ETL Processes and Procedures:**

The Tasks are scheduled such that the Dimensional Tables, specifically the tasks consisting of the procedure to update the DIM\_ARTIST and DIM\_ALBUMS tables first (Task A) with the new data from the staging table followed by updating the DIM\_SONGS table (Task B), update/upsert DIM\_AUDIOFEATURES tables (Task C) and lastly insert new periodic data into the transactional snapshot fact table (Task D) that has the updated popularity scores for all the songs.

We finally truncate the data in the staging table (Task E).

A diagram of a network

Description automatically generated

A – Update/Upsert records to DIM\_ARTIST and DIM\_ALBUMS tables

B – Update DIM\_SONGS table

C – Update/Upsert DIM\_AUDIOFEATURES table

D – Load data into transactions snapshot fact table

C – Truncate Stage table

**Application Workflow**

New data is first pulled into Azure via the Spotify API key by scheduling a periodic job on Azure Databricks:

* The new data is first gathered and stored in an azure blob storage.
* Next, we make you of Azure Event grid to gather Azure storage blob queues when a new file is uploaded.
* Snowpipe is triggered as this queue is consumed in Snowflake and via the AZURE\_STAGE procedure the new data is copied into a stagging table.
* Once in the staging table various tasks are triggered based on the Task tree highlighted in the previous steps.
* Upon completion of the necessary ETL processes new data UPSERTED/INSERTED into the required dim and fact tables and views are updated. The Power BI report has a real-time connection to the warehouse via Direct Query.

A diagram of a diagram

Description automatically generated

**Data Warehouse Architecture:**

Warehouse consist of Dimension and one transactional fact table as shown below. Along with stored procedures to update each of these table with fresh data that is retrieved in the staging table.

We also have two views in the public schema, the view dropped from playlist provides a list of songs that get dropped out from any of the 6 playlists that we are tracking and new releases in playlist tracks those songs that are new releases which get added to any of these 6 playlists.

A screenshot of a computer

Description automatically generated

**Power BI :**

Relationship View:

All tables are loaded via the direct query framework as this dashboard is updated with new data daily. Using power query only required columns are loaded into the dashboard and the relationship view in model view is as below:

A screenshot of a computer

Description automatically generated

Power BI Report: A screenshot of a computer screen

Description automatically generated