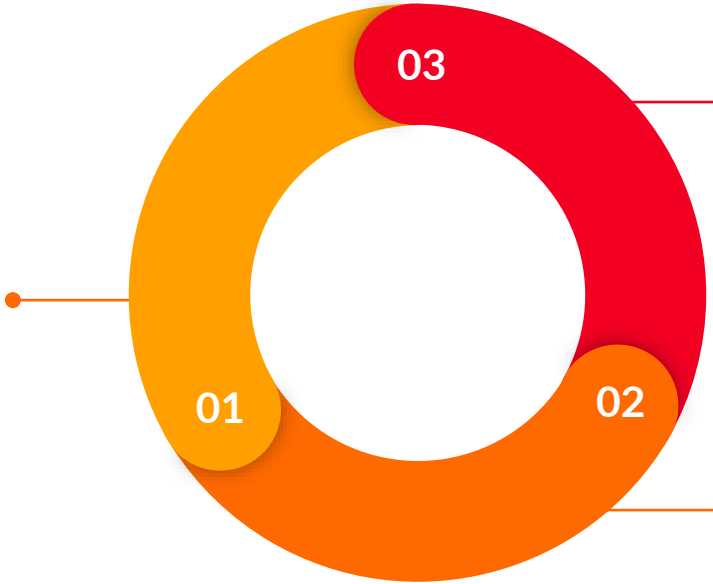
Information Integration

Project Presentation

Valentina Sisti
1952657

Main Phases



Implementation Phase

Modeling Phase

Data Gathering Phase

Phase 1.
Data Gathering

Domain of Interest

♫ Music & Spotify

- Songs
- Artists
- Albums

- Genres
- Lyrics
- Charts

Data Collection Phase

Spotify®

8+M. Spotify Tracks Dataset

- *tracks.csv*
- *artists.csv*
- *albums.csv*
- *audio_features.csv*
- *r_albums_artists.csv*
- *r_albums_tracks.csv*
- *r_artist_genre.csv*
- *r_tracks_artist.csv*

Data Collection Phase

billboard-hot-100-tracks.csv

**Billboard Hot 100 -
Full History Dataset**

billboard HOT 100

Data Collection Phase

5 Millions Genius Lyrics Dataset

genius_lyrics.csv

GENIUS

Data Collection Phase

The Influence of Music, Spotify and AllMusic Data

influence_data.csv

Spotify®

ALLMUSIC

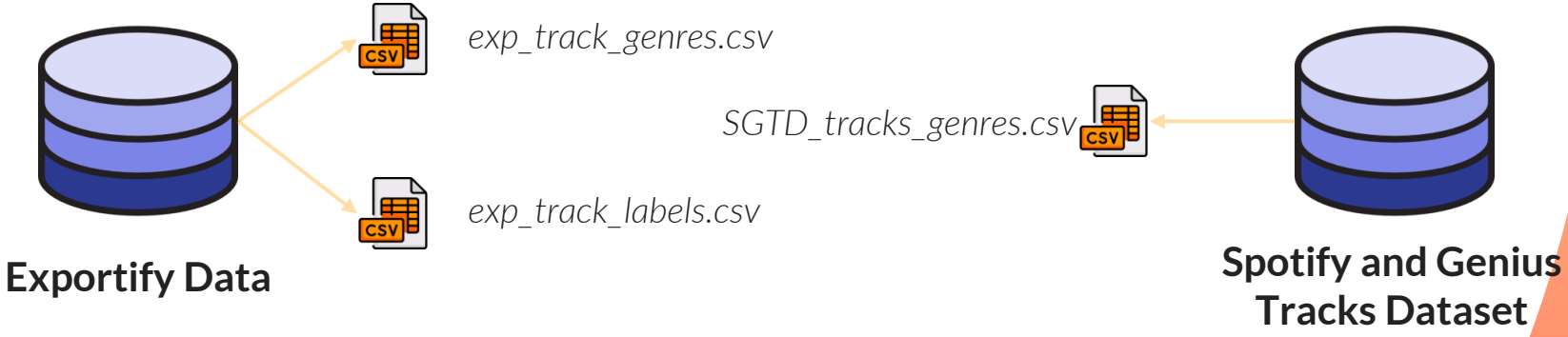
Data Collection Phase

record_labels.csv

Record Labels - All Universal Music Group Artists

UNIVERSAL MUSIC GROUP

Data Collection Phase



Exportify GENIUS Spotify®

7 Datasets

15 CSV Files

Tasks definition

- Find out which are the most popular ***artists***, ***songs*** or ***albums***
- Find out songs of which ***genre*** are most likely to reach the Billboard Hot 100, based on the history
- Find out if there are some ***labels*** that have very popular artists and in turn very popular songs
- Find out which is the ***influence*** that very important artists have had on other artists

Problems of the Data

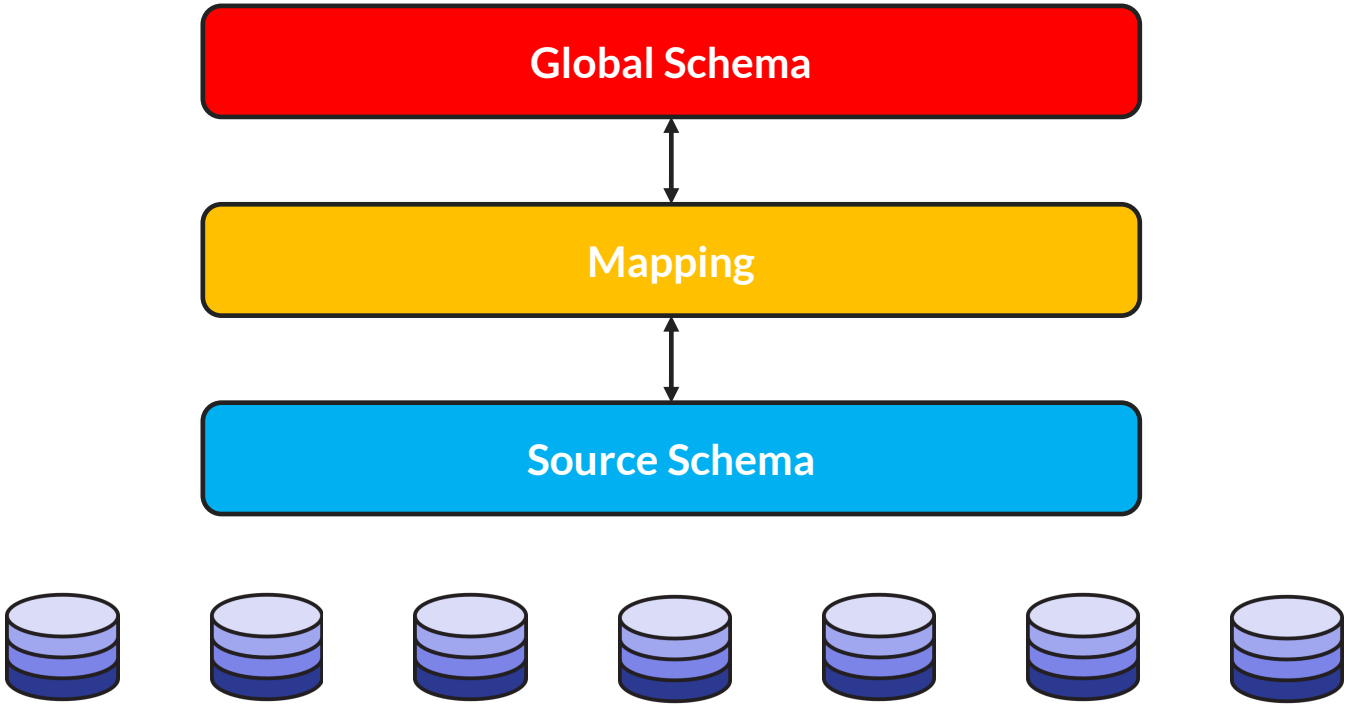
- **Conceptual heterogenity**
  - Concept of «id» may refer to Spotify ID, Billboard ID or custom ID internally defined
- ***track_id*** and ***audio_feature_id*** are the same in *tracks.csv*
- Songs' ***duration*** stored in milliseconds
- Albums' ***release_date*** stored in epoch format

Problems of the Data (2)

- Some data sources contain a ***huge volume*** of data
- A great part of it refers to quite ***unknown artists*** and songs
- In order to gain interesting insights, we want to restrict our discussion to just ***most relevant*** artists and songs

Phase 2.
Modeling

Information Integration System



Source Schema

$Artists8M(name, id, popularity, followers)$

$Songs8M(id, disc\_number, duration, explicit, audio\_feature\_id, name, preview\_url, track\_number, popularity, is\_playable)$

$Albums8M(id, name, album\_group, album\_type, release\_date, popularity)$

$TracksArtists8M(track\_id, artist\_id)$

$AlbumsTracks8M(album\_id, track\_id)$

$AlbumsArtists8M(album\_id, artist\_id)$

$ArtistsGenres8M(artist\_id, genre)$

Source Schema (2)

$GeniusLyrics(title, tag, artist, year, views, features, lyrics, id)$

$BillboardFullHistory(billboard\_id, rank, date, artist, title, last\_week, weeks\_on\_chart, peak\_rank)$

$Influence(influencer\_name, influencer\_main\_genre, influencer\_active\_start, follower\_name, follower\_main\_genre, follower\_active\_start)$

$AudioFeatures8M(id, acousticness, danceability, duration, energy, instrumentalness, key, liveness, loudness, mode, speechiness, tempo, time\_signature)$

Source Schema (3)

$SGTDTracksGenres(artist\_popularity, followers, genre, id, name, track\_id, track\_name\_prev, type)$

$ExportifyTracksGenres(track\_URI, track\_name, artist\_id, artist\_name, album\_URI, album\_name, track\_id, genre)$

$ExportifyTracksLabels(track\_URI, track\_name, artist\_id, artist\_name, track\_id, genre)$

$RecordLabels(id, group, label, artist\_name)$

Global Schema

$Song_{/7}(id, name, popularity, track\_number, duration, explicit, preview\_url)$

$Artist_{/4}(id, name, popularity, followers)$

$Album_{/6}(id, name, album\_type, popularity, release\_date)$

$BillboardHot100Song_{/5}(song\_id, rank, date, weeks\_on\_chart, peak\_rank)$

$AudioFeature_{/12}(song\_id, acousticness, danceability, energy,$
$instrumentalness, key, liveness, loudness, mode, speechiness,$
$tempo, time\_signature)$

Global Schema (2)

$isSongOf_{/2}(song\_id, artist\_id)$

Song with «*song_id*» is performed by Artist with «*artist_id*»

$isSongInAlbum_{/2}(song\_id, album\_id)$

Song with «*song_id*» is in Album with «*album_id*»

$isAlbumOf_{/2}(album\_id, artist\_id)$

Album with «*album_id*» is of Artist with «*artist_id*»

Global Schema (3)

$Genre_{/1}(genre\_name)$

$doesGenre_{/2}(artist\_id, genre\_name)$

Artist with «artist_*id*» does the music Genre «genre_name»

$hasInfluenced_{/2}(artist\_id1, artist\_id2)$

Artist with «*artist_id1*» has influenced Artist with «*artist_id2*»

$hasLyrics_{/2}(song\_id, lyrics)$

Song with «*song_id*» has Lyrics «lyrics»

Global Schema (4)

$Label_{/1}(label\_name)$

$LabelGroup_{/1}(group\_name)$

$hasLabel_{/2}(artist\_id, label)$

Artist «*artist_id*» has published a Song under the Label «*label*»

$isLabelInGroup_{/2}(label, group)$

Label «label» belongs to the Label Group «*group*»

Conjunctive GAV Mapping

Artist:

$$\forall\, id, n, p, f.\, (Artist8M(p, f, id, n) \;\wedge\; (p \geq 50)) \rightarrow Artist(id, n, p, f)$$

Song:

$$\forall\, id, n, d, e, pu, p, tn.\, (\exists\, dn, afi, ap, af, aid, an.\, Song8M(\boldsymbol{id}, dn, d, e, afi, n, pu, p, tn) \wedge TrackArtist8M(\boldsymbol{id}, \boldsymbol{aid}) \;\wedge\; Artist8M(ap, af, aid, an) \;\wedge\; (ap \geq 50)) \;\rightarrow\; Song(id, n, p, tn, d, e, pu)$$

Album:

$$\forall\, id, n, at, p, rd.\, (\exists\, ap, af, aid, an.\, Album8M(\boldsymbol{id}, n, ag, at, rd, p) \wedge AlbumArtist8M(\boldsymbol{id}, \boldsymbol{aid}) \;\wedge\; Artist8M(ap, af, aid, an) \;\wedge\; (ap \geq 50)) \;\rightarrow\; Album(id, n, at, p, rd)$$

Conjunctive GAV Mapping (2)

isSongOf:

$$\forall\, sid, aid.\, (\exists\, p, f, n.\, TrackArtist8M(sid, \boldsymbol{aid}) \;\wedge\; Artist8M(p, f, \boldsymbol{aid}, n) \;\wedge\; (p \geq 50)) \rightarrow isSongOf(sid, aid)$$

isSongInAlbum:

$$\forall\, sid, albid.\, (\exists\, p, f, aid, n.\, AlbumTrack8M(\boldsymbol{sid}, albid) \;\wedge\; TrackArtist8M(\boldsymbol{sid}, \boldsymbol{aid}) \;\wedge\; Artist8M(p, f, \boldsymbol{aid}, n) \;\wedge\; (p \geq 50)) \rightarrow isSongInAlbum(sid, albid)$$

isAlbumOf:

$$\forall\, albid, aid.\, (\exists\, p, f, n.\, AlbumArtist8M(albid, \boldsymbol{aid}) \;\wedge\; Artist8M(p, f, \boldsymbol{aid}, n) \;\wedge\; (p \geq 50)) \rightarrow isAlbumOf(albid, aid)$$

Conjunctive GAV Mapping (3)

AudioFeature:

$$\forall\, sid, a, d, e, i, k, li, lo, m, s, t, ts.\ (\exists\, ap, af, aid, an, du.\ AudioFeature8M(\boldsymbol{sid}, a, d, du, e, i, k, li, lo, m, s, t, ts) \wedge TrackArtist8M(\boldsymbol{sid}, \boldsymbol{aid}) \wedge Artist8M(ap, af, aid, an) \wedge (ap \geq 50) \rightarrow AudioFeature(sid, a, d, e, i, k, li, lo, m, s, t, ts)$$

Genre:

$$\forall\, genre.\ (\exists tu, tn, aid, an, alu, aln, tnb.\ ExportifyTracksGenres(tu, tn, aid, an, alu, aln, tnb, genre) \rightarrow Genre(genre)$$

$$\forall\, genre.\ (\exists\, p, f, id, n, tid, tnp, t.\ SGTDTracksGenres(p, f, genre, id, n, tid, tnp, t)) \rightarrow Genre(genre)$$

$$\forall\, genre.\ (\exists artist\_id.\ ArtistGenre8M(artist\_id, genre)) \rightarrow Genre(genre)$$

Conjunctive GAV Mapping (4)

doesGenre:

$$\forall\, artist\_id, genre.\ (\exists\, ap, af, artist\_name, tu, tn, alu, aln, tnb.\ ExportifyTracksGenres(tu, tn, \boldsymbol{artist\_id}, \boldsymbol{artist\_name}, alu, aln, tnb, genre)\ \wedge\ Artist8M(ap, af, \boldsymbol{artist\_id}, \boldsymbol{artist\_name})\ \wedge\ (ap \geq 50)) \rightarrow doesGenre(artist\_id, genre)$$

$$\forall\, artist\_id, genre.\ (\exists\, p, f, id, ap, af, track_{id}, dn, d, e, afi, n, pu, tp, tn.\ SGTDTracksGenres(p, f, genre, id, n, \boldsymbol{track\_id}, tnp, t)\ \wedge Song8M(\boldsymbol{track\_id}, dn, d, e, afi, n, pu, tp, tn)\ \wedge\ TrackArtist8M(\boldsymbol{sid}, \boldsymbol{aid}) \wedge Artist8M(ap, af, \boldsymbol{artist\_id}, \boldsymbol{artist\_name}) \wedge (ap \geq 50)) \rightarrow doesGenre(artist\_id, genre)$$

$$\forall\, artist\_id, genre.\ (\exists ap, af, artist\_name.\ ArtistGenre8M(\boldsymbol{artist\_id}, genre)\ \wedge\ Artist8M(ap, af, \boldsymbol{artist\_id}, \boldsymbol{artist\_name})\ \wedge\ (ap \geq 50)) \rightarrow doesGenre(artist\_id, genre)$$

Conjunctive GAV Mapping (5)

hasLyrics:

$$\forall\, sid, lyrics.\,(\exists\, ap, af, aid, an, title, t, y, v, ft, i, dn, d, e, afi, pu, sp, tn.$$
$$Lyrics5M(\boldsymbol{title}, t, \boldsymbol{an}, y, v, ft, \boldsymbol{lyrics}, i) \wedge Song8M(\boldsymbol{sid}, dn, d, e, af, \boldsymbol{title}, pu, sp, tn) \wedge$$
$$TrackArtist8M(\boldsymbol{sid}, \boldsymbol{aid}) \wedge Artist8M(ap, af, aid, \boldsymbol{an}) \wedge (ap \geq 50)) \rightarrow hasLyrics(sid, lyrics)$$

BillboardHot100Song:

$$\forall\, sid, r, d, woc, pr.\,(\exists\, bid, a\_name, title, lw, ap, af, aid, dn, du, e, afi, pu, sp, tn.$$
$$BillboardFullHistory(bid, r, d, a\_name, \boldsymbol{title}, lw, woc, pr) \wedge Song8M(\boldsymbol{sid}, dn, du, e, afi, \boldsymbol{title}, pu, sp, tn) \wedge$$
$$TrackArtist8M(\boldsymbol{sid}, \boldsymbol{aid}) \wedge Artist8M(ap, af, \boldsymbol{aid}, an) \wedge (p \geq 50))$$
$$\rightarrow BillboardHot100Song(sid, r, d, woc, pr)$$

Conjunctive GAV Mapping (6)

hasInfluenced:

$$\forall\, aid1, aid2.\ (\exists\, infl\_name, img, ias, foll\_name, fmg, fas, ap1, ap2, af1, af2.\ Influence(\boldsymbol{infl\_name}, img, ias, \boldsymbol{foll\_name}, fmg, fas) \wedge Artist8M(ap1, af1, aid1, \boldsymbol{infl\_name}) \wedge (ap1 \geq 50) \wedge Artist8M(ap2, af2, aid2, \boldsymbol{foll\_name}) \wedge (ap2 \geq 50)) \rightarrow hasInfluenced(aid1, aid2)$$

Label:

$$\forall\, label.\ (\exists\, tu, tn, aid, an, tnb.\ ExportifyTracksLabels(tu, tn, aid, an, tnb, label)) \rightarrow Label(label)$$

$$\forall\, label.\ (\exists\, id, group, artist\_name.\ RecordLabels(id, group, label, artist\_name)) \rightarrow Label(label)$$

Group:

$$\forall\, group.\ (\exists id, label, artistname.\ RecordLabels(id, group, label, artist\_name)) \rightarrow Group(group)$$

Conjunctive GAV Mapping (7)

hasLabel:

$$\forall\, label, artist\_id.\ (\exists\ ap, af, aid, an, tu, tn, tnb.\ ExportifyTracksLabels(tu, tn, \boldsymbol{artist\_id}, an, tnb, label)\ \wedge\ Artist8M(ap, af, \boldsymbol{artist\_id}, an) \wedge (ap \geq 50)) \rightarrow hasLabel(artist\_id, label)$$

$$\forall\, label, artist\_id.\ (\exists\ ap, af, aid, artist\_name, id, group.\ RecordLabels(id, group, label, \boldsymbol{artist\_name})\ \wedge\ Artist8M(ap, af, artist\_id, \boldsymbol{artist\_name}) \wedge (ap \geq 50)) \rightarrow hasLabel(artist\_id, label)$$

isLabelInGroup:

$$\forall\, label, group.\ (\exists\ ap, af, aid, artist\_name, id, group.\ RecordLabels(id, group, label, \boldsymbol{artist\_name})\ \wedge\ Artist8M(ap, af, aid, \boldsymbol{artist\_name})\ \wedge (ap \geq 50)) \rightarrow isLabelInGroup(label, group)$$

The Query Answering Problem

- It can be proved that, given the **Retrieved Global Database**, in the case of Conjunctive GAV Mapping, we can compute *certain answers* for CQ and UCQ.

- Now, in order to compute and obtain the RGDB, we can exploit the **Materialization-based approach** in the GAV context, simply by materializing the global schema views.

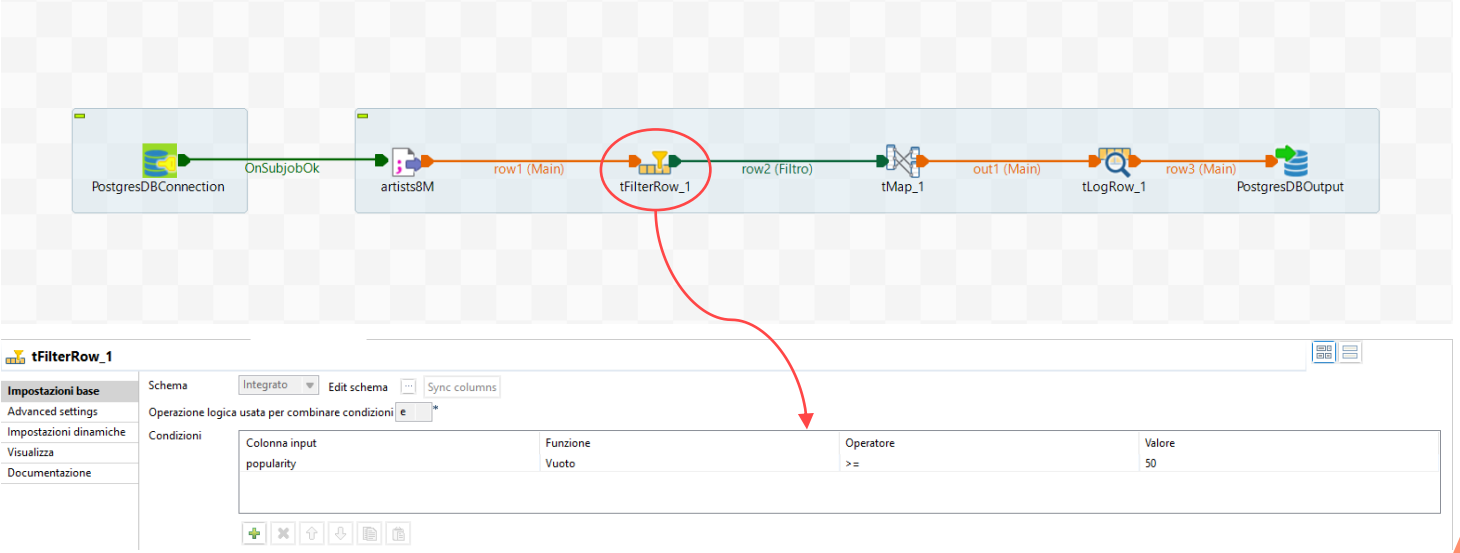
- In order to do this in practice, we can exploit a **data integration tool** and this leads us to the final phase of the project: the Implementation Phase.

Phase 3.
Implementation

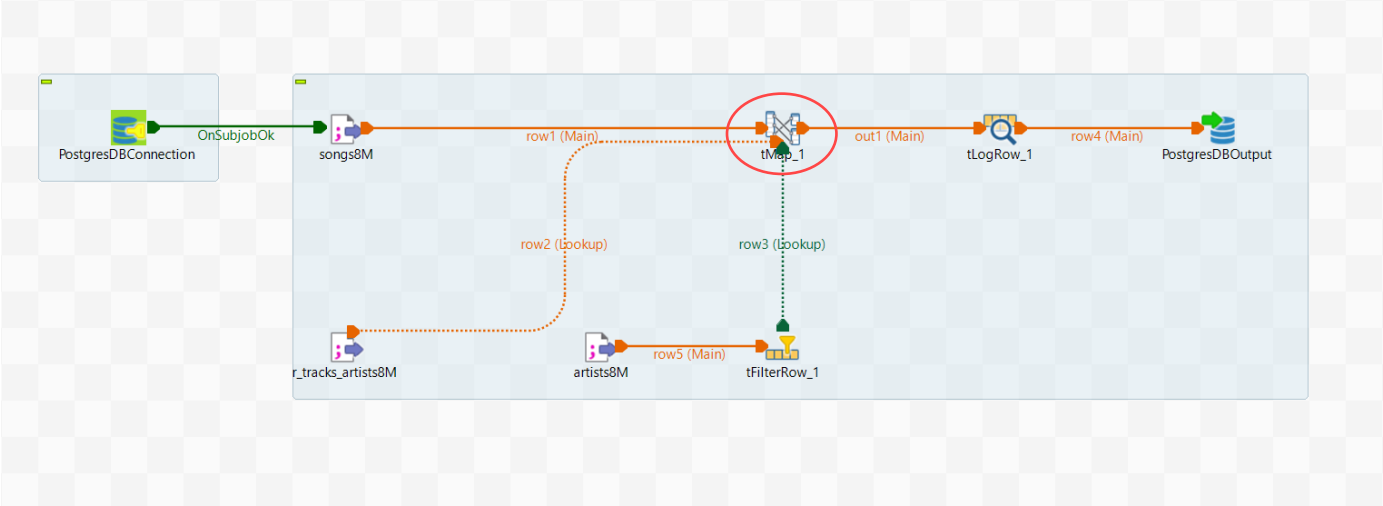
Data Integration Tool: Talend

- Talend is the *leader tool* in cloud and big data integration
- Fully *open source*
- Embeds *existing* Java code libraries and allows to create *custom* components
- Allows to *easily* integrate data from different sources thanks to a GUI

Artists

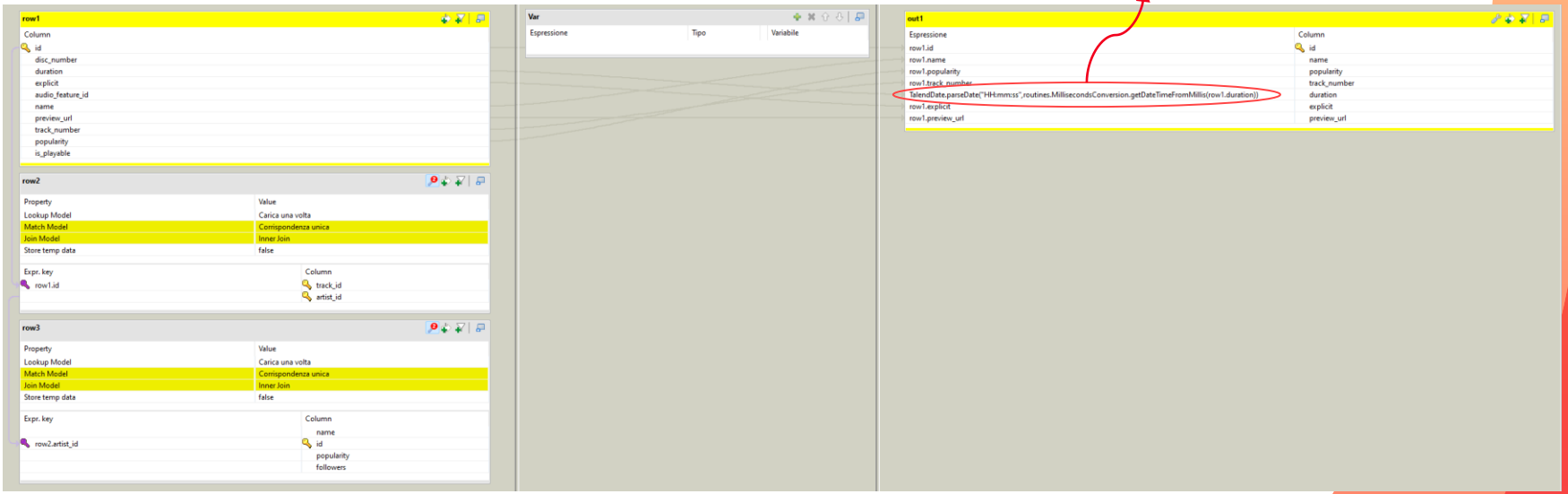

PostgresDBConnection
OnSubjobOk
artists8M
row1 (Main)
tFilterRow_1
row2 (Filtro)
tMap_1
out1 (Main)
tLogRow_1
row3 (Main)
PostgresDBOutput
tFilterRow_1
Impostazioni base
Advanced settings
Impostazioni dinamiche
Visualizza
Documentazione
Schema
Integrato
Edit schema
Sync columns
Operazione logica usata per combinare condizioni
e
Condizioni
Colonna input
Funzione
Operatore
Valore
popularity
Vuoto
>=
50

Songs

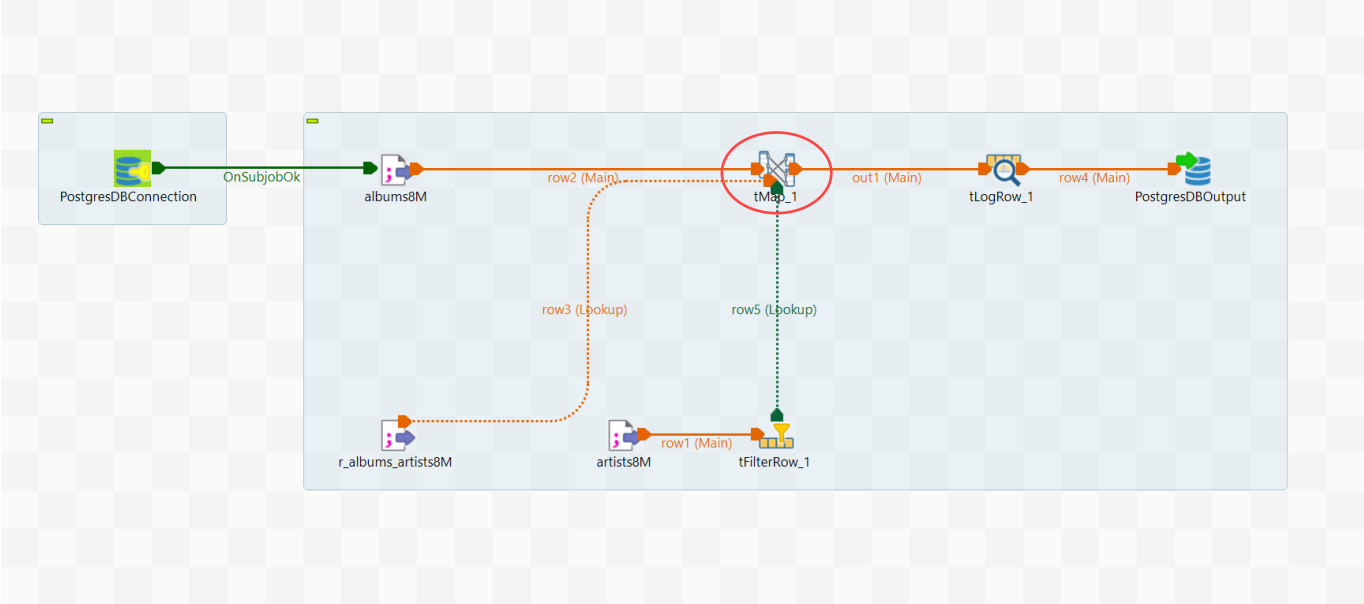

PostgresDBConnection
OnSubjobOk
songs8M
row1 (Main)
tMap_1
out1 (Main)
tLogRow_1
row4 (Main)
PostgresDBOutput
row2 (Lookup)
row3 (Lookup)
r_tracks_artists8M
artists8M
row5 (Main)
tFilterRow_1

Songs - tMap

```java
public class MillisecondsConversion {

    public static String getDateTimeFromMillis(long millis) throws ParseException{

        String timeInHHMMSS = "";

        if(millis != 0) {
            Duration duration = Duration.ofMillis(millis);
            long HH = duration.toHours();
            long MM = duration.toMinutesPart();
            long SS = duration.toSecondsPart();
            timeInHHMMSS = String.format("%02d:%02d:%02d", HH, MM, SS);
        }
        return timeInHHMMSS;
    }
}
```

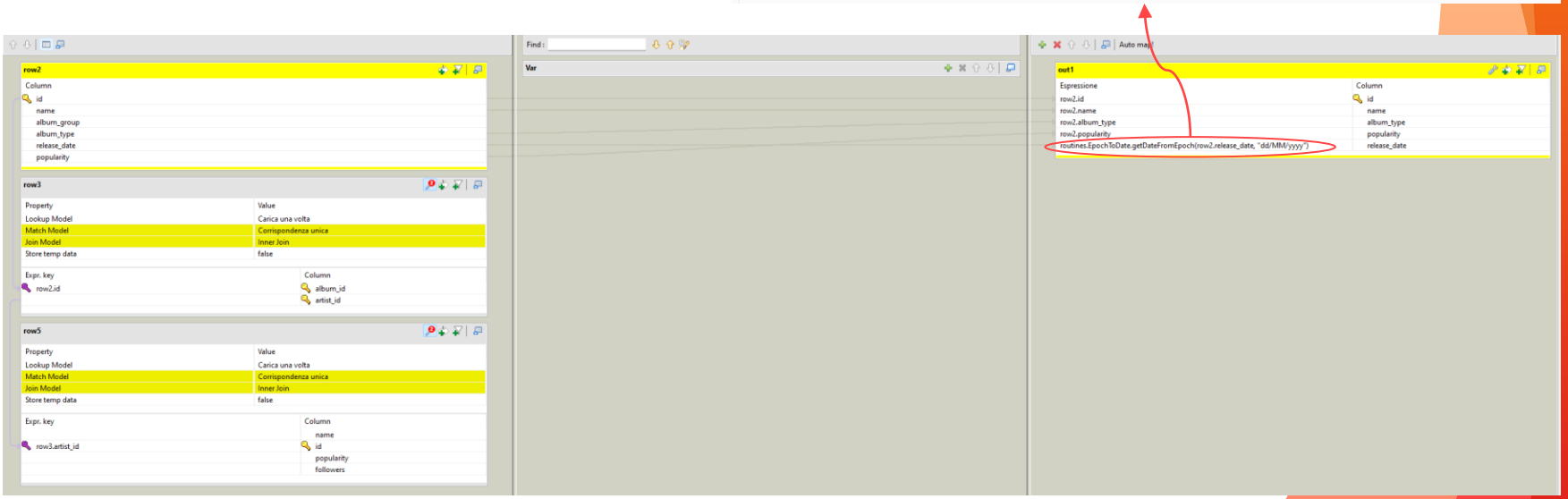


Albums

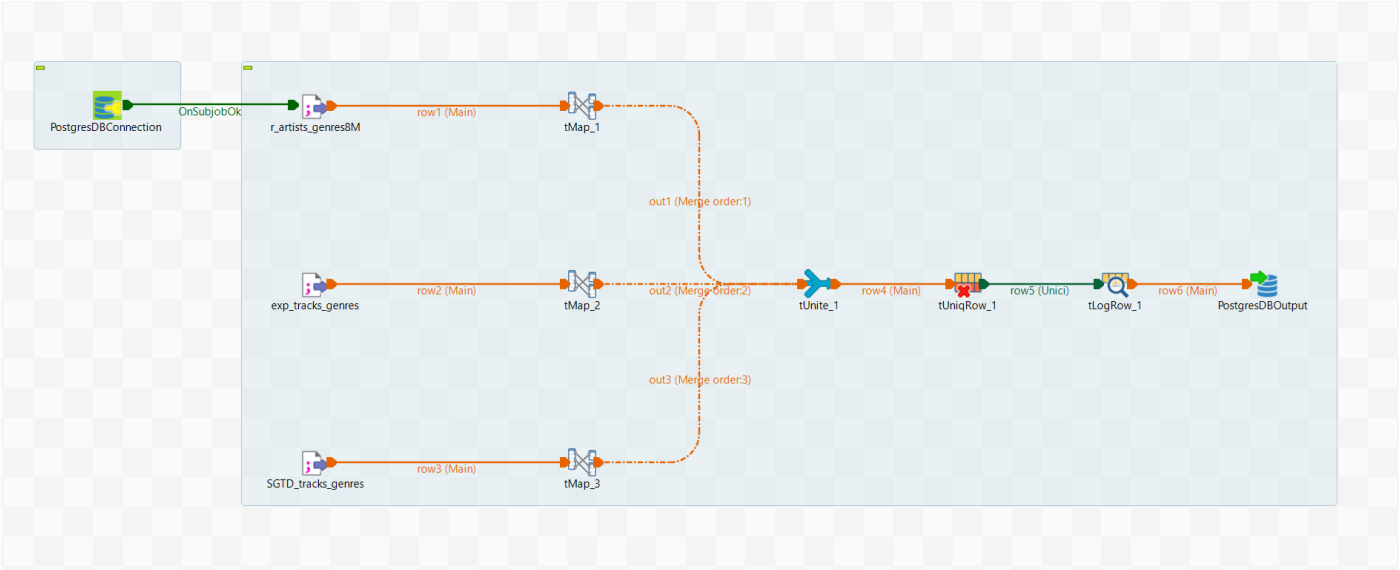

PostgresDBConnection
OnSubjobOk
albums8M
row2 (Main)
tMap_1
out1 (Main)
tLogRow_1
row4 (Main)
PostgresDBOutput
row3 (Lookup)
row5 (Lookup)
r_albums_artists8M
artists8M
row1 (Main)
tFilterRow_1

Albums - tMap

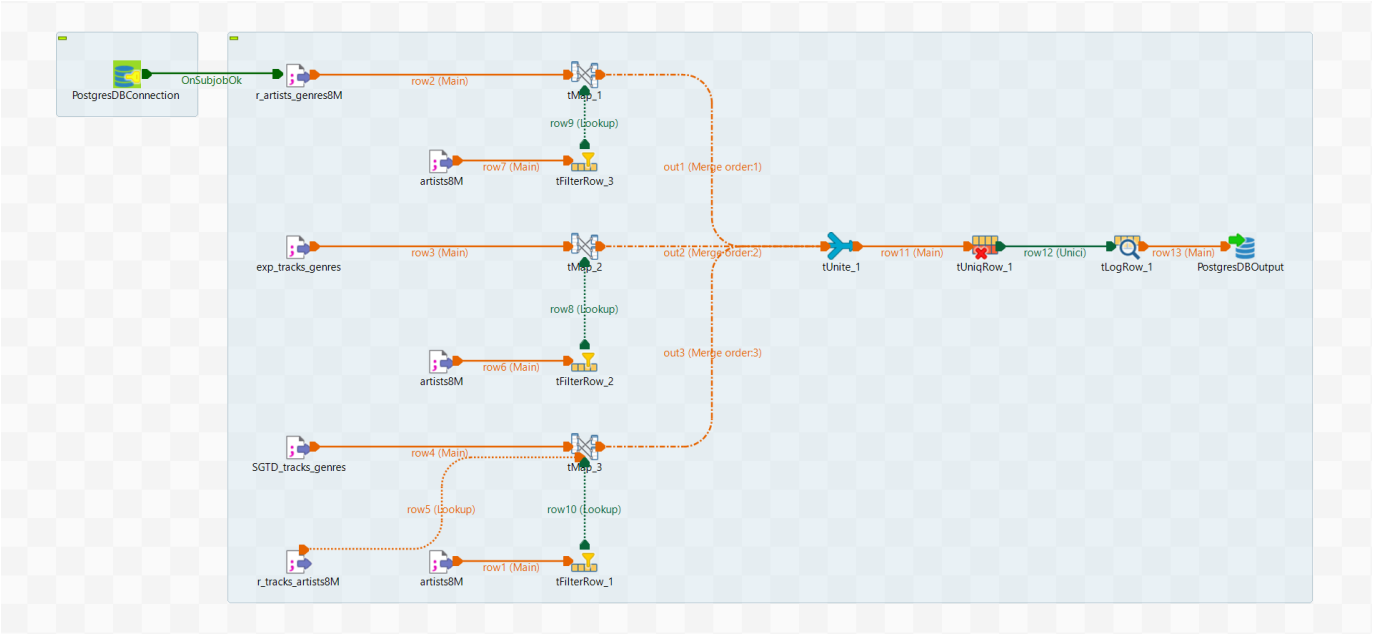
```java
public class EpochToDate {

    public static Date getDateFromEpoch(Long timeAsLong, String javaDatePattern) throws ParseException{

        String returnVal = "";
        Date returnDateVal = null;

        if(timeAsLong!=null){
            SimpleDateFormat sdf = new SimpleDateFormat(javaDatePattern);
            returnVal = sdf.format(new Date(timeAsLong));
            returnDateVal = sdf.parse(returnVal);

        }
        return returnDateVal;
    }
}
```



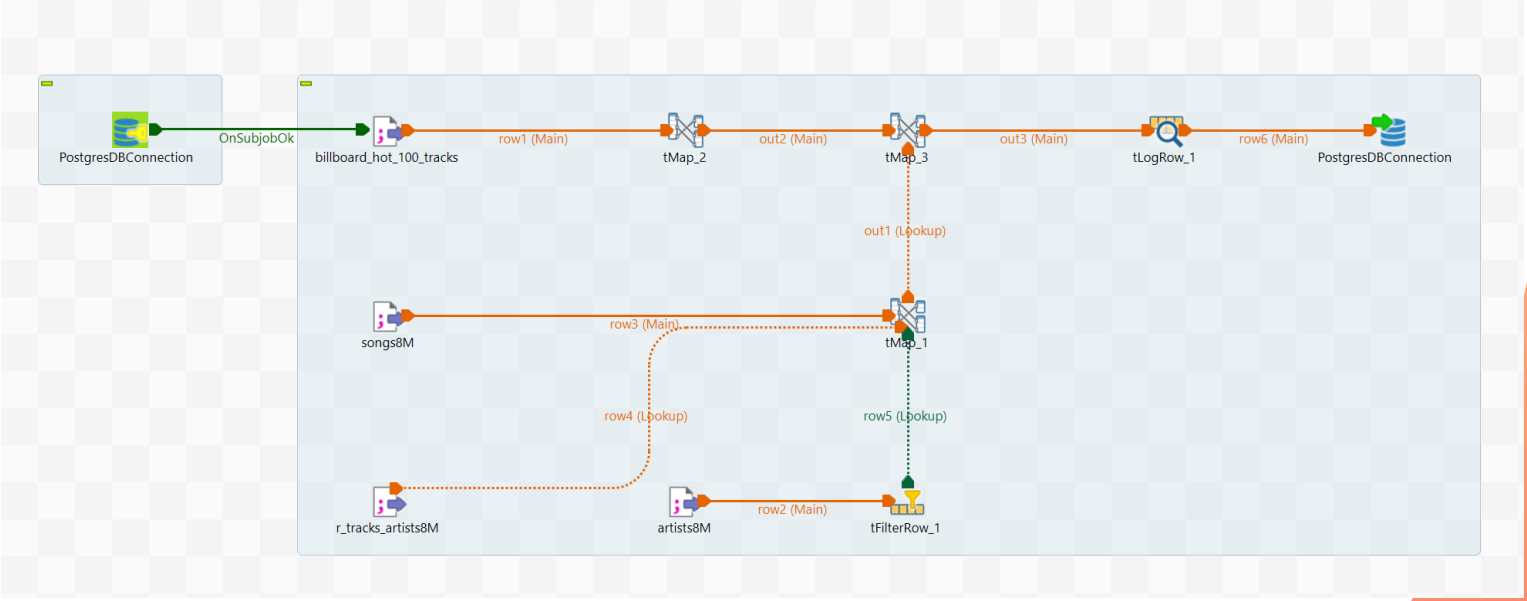
Genres


PostgresDBConnection
OnSubjobOk
r_artists_genres8M
row1 (Main)
tMap_1
out1 (Merge order:1)
exp_tracks_genres
row2 (Main)
tMap_2
out2 (Merge order:2)
tUnite_1
row4 (Main)
tUniqRow_1
row5 (Unici)
tLogRow_1
row6 (Main)
PostgresDBOutput
out3 (Merge order:3)
SGTD_tracks_genres
row3 (Main)
tMap_3

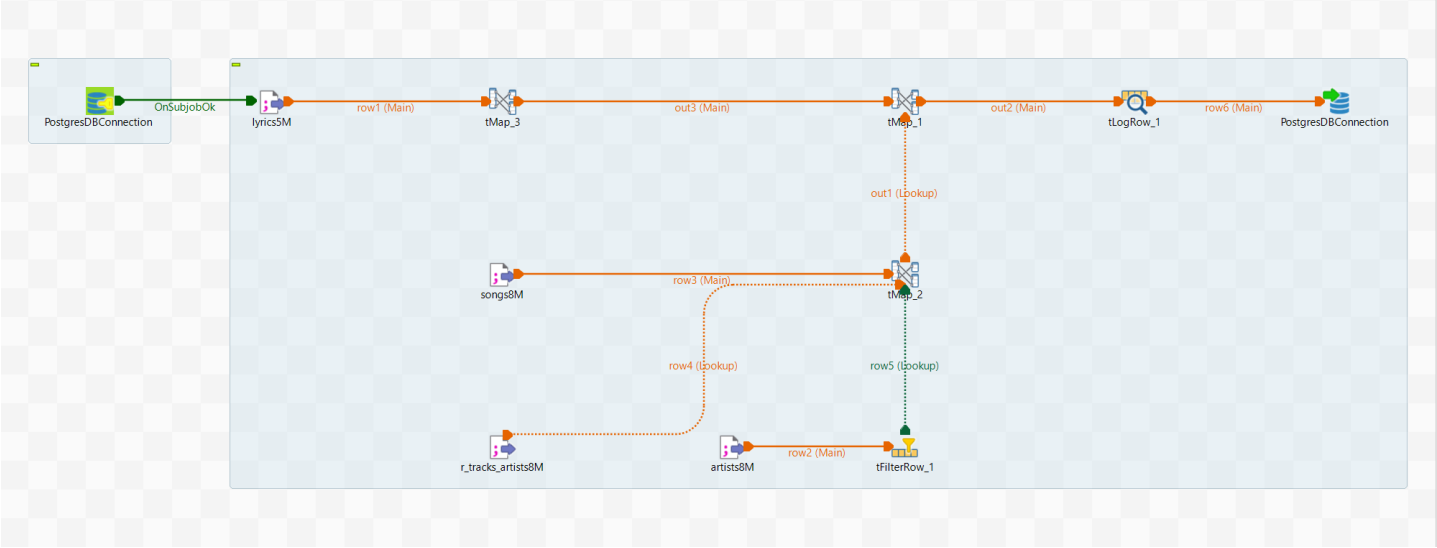
doesGenre


PostgresDBConnection
OnSubjobOk
r_artists_genres8M
row2 (Main)
tMap_1
row9 (Lookup)
artists8M
row7 (Main)
tFilterRow_3
out1 (Merge order:1)
exp_tracks_genres
row3 (Main)
tMap_2
out2 (Merge order:2)
tUnite_1
row11 (Main)
tUniqRow_1
row12 (Unici)
tLogRow_1
row13 (Main)
PostgresDBOutput
row8 (Lookup)
artists8M
row6 (Main)
tFilterRow_2
out3 (Merge order:3)
SGTD_tracks_genres
row4 (Main)
tMap_3
row5 (Lookup)
row10 (Lookup)
r_tracks_artists8M
artists8M
row1 (Main)
tFilterRow_1

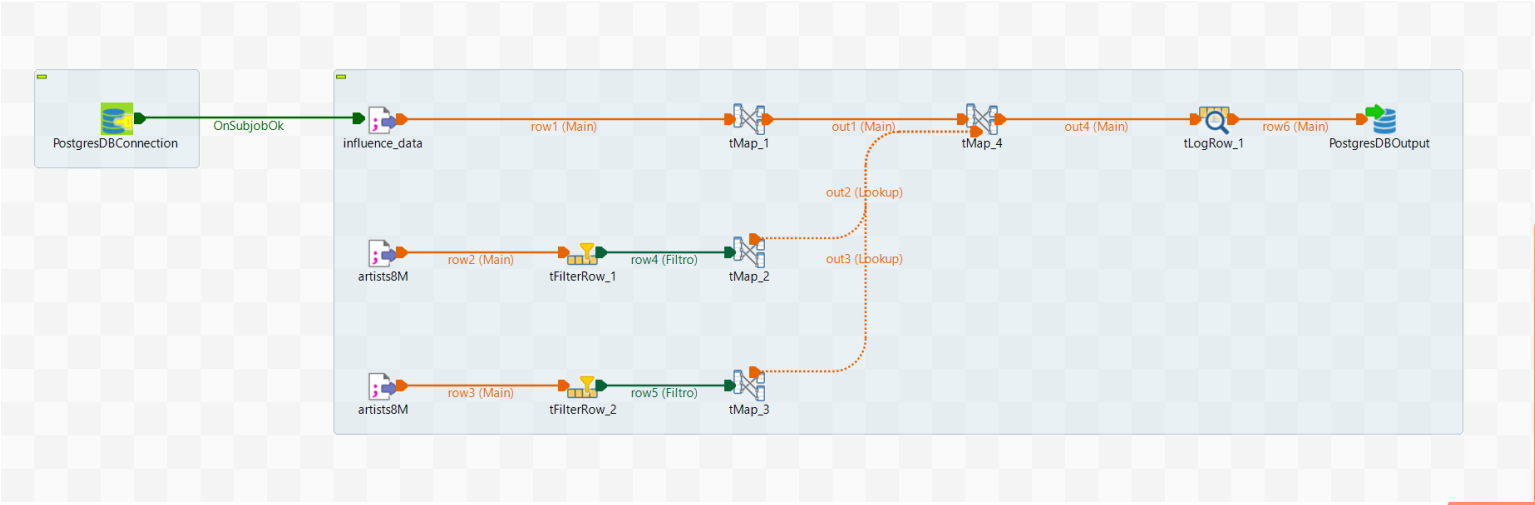
BillboardHot100Song



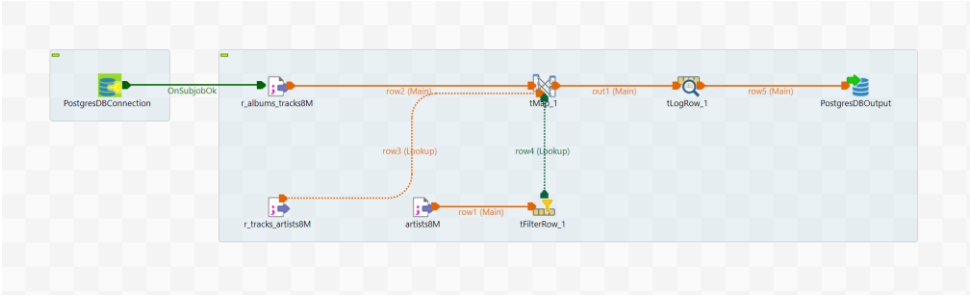
hasLyrics

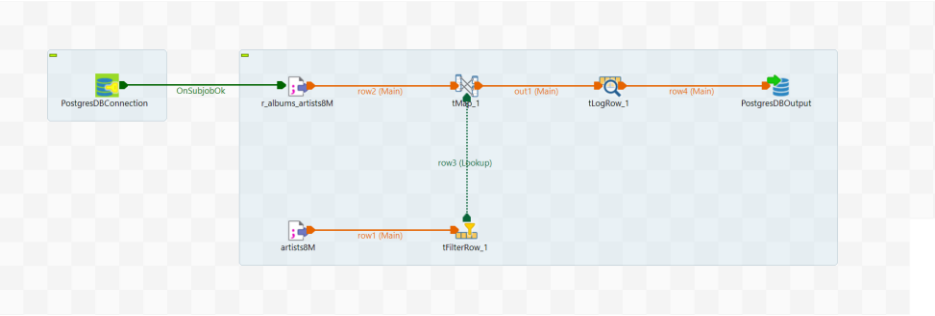

PostgresDBConnection
OnSubjobOk
lyrics5M
row1 (Main)
tMap_3
out3 (Main)
tMap_1
out2 (Main)
tLogRow_1
row6 (Main)
PostgresDBConnection
out1 (Lookup)
songs8M
row3 (Main)
tMap_2
row4 (Lookup)
row5 (Lookup)
r_tracks_artists8M
artists8M
row2 (Main)
tFilterRow_1

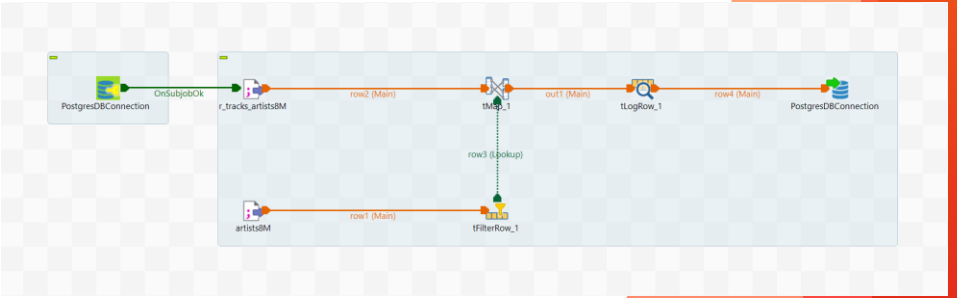
hasInfluenced



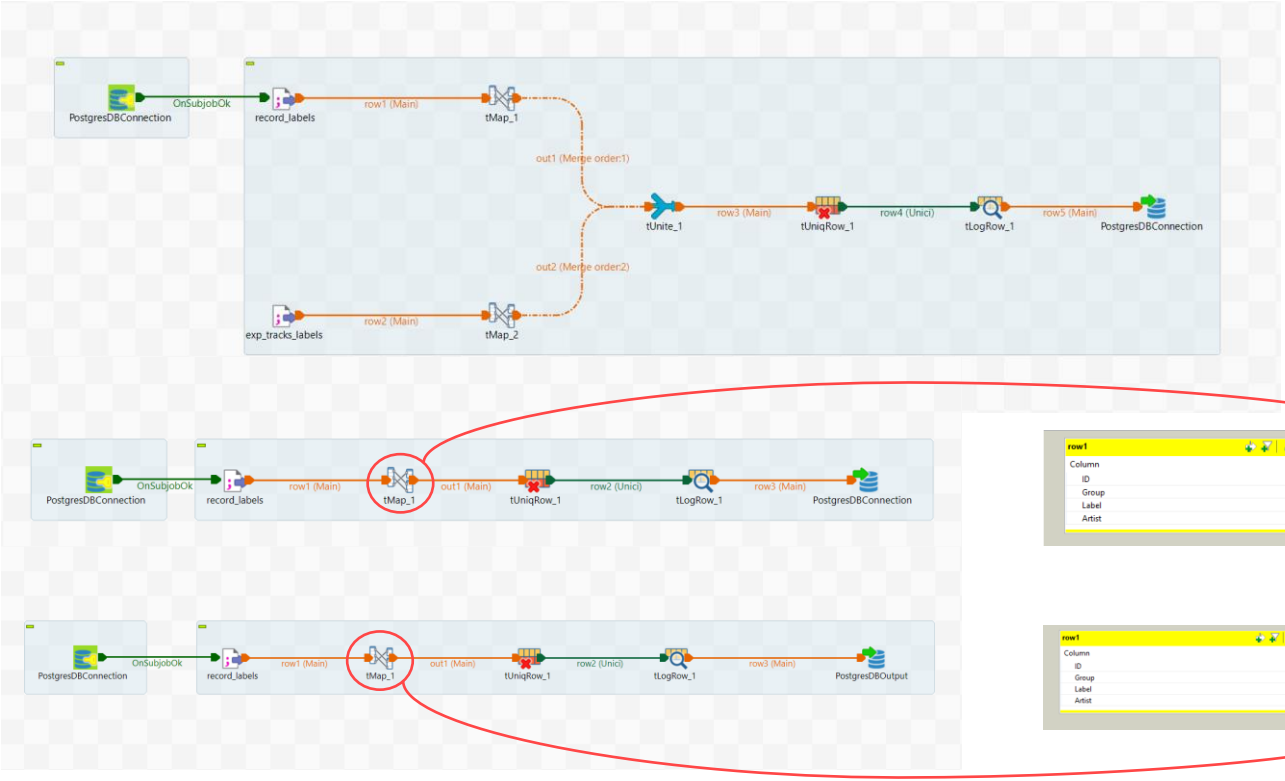
isSongOf – isAlbumOf – isSongInAlbum







Label – Group - isLabelInGroup



Query 1 - FOL

Find Albums of Artists with more than 500k followers that contain Songs that are present in the Billboard Hot 100 Chart at least 10 years after their release

$$
\begin{aligned}
q = \{(&\boldsymbol{album\_name}, \boldsymbol{artist\_name}, \boldsymbol{song\_name}, \boldsymbol{billboard\_date}, \boldsymbol{release\_date}, \boldsymbol{rank}, \boldsymbol{peak\_rank}) \quad | \\
&\exists\, al\_id, al\_type, al\_pop, ar\_id, ar\_pop, foll, s\_id, s\_pop, t\_num, dur, exp, prev\_url, woc\ . \\
&Album(al\_id, \boldsymbol{album\_name}, 'album', al\_pop, \boldsymbol{release\_date}) \quad \wedge \\
&AlbumOf(al\_id, ar\_id) \quad \wedge \\
&Artist(ar\_id, \boldsymbol{artist\_name}, ar\_pop, foll) \quad \wedge \\
&SongInAlbum(s\_id, al\_id) \quad \wedge \\
&Song(s\_id, \boldsymbol{song\_name}, s\_pop, t\_num, dur, exp, prev\_url) \quad \wedge \\
&BillboardSong(s\_id, \boldsymbol{rank}, \boldsymbol{billboard\_date}, woc, \boldsymbol{peak\_rank}) \quad \wedge \\
&(foll > 500000) \quad \wedge \quad ((billboard\_date \,/\, release\_date) \geq 10)\}
\end{aligned}
$$

Query 1 - SQL

Find Albums of Artists with more than 500k followers that contain Songs that are present in the Billboard Hot 100 Chart at least 10 years after their release

```
SELECT DISTINCT al.name as album_name, ar.name as artist_name,
                s.name as song_name, al.release_date,
                bhs.date as billboard_date, bhs.peak_rank, bhs.rank
FROM albums as al, artists as ar, is_album_of as iao,
     is_song_in_album as isia,
   songs as s, billboard_hot100_songs as bhs
WHERE al.id = iao.album_id AND
      ar.id = iao.artist_id AND
      al.id = isia.album_id AND
      s.id = isia.track_id AND
      s.id = bhs.track_id AND
      (DATE_PART('year', bhs.date) -
      DATE_PART('year', al.release_date)) ⩾ 10 AND
      ar.followers > 500000 AND
      al.album_type = 'album'
ORDER BY al.release_date DESC
```

	album_name character varying (500)	artist_name character varying (500)	song_name character varying (1000)	release_date date	billboard_date date	peak_rank integer	rank integer
1	Cole World: The Sideline Story	J. Cole	Interlude	2011-09-27	2021-05-22	8	8
2	Cole World: The Sideline Story	J. Cole	Interlude	2011-09-27	2021-05-29	8	17
3	Cole World: The Sideline Story	J. Cole	Interlude	2011-09-27	2021-06-05	8	71
4	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-01-04	1	1
5	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-11-28	1	29
6	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-12-05	1	14
7	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-12-12	1	2
8	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-12-19	1	1
9	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2020-12-26	1	2
10	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2021-01-02	1	1
11	Merry Christmas II You	Mariah Carey	All I Want For Christmas Is You	2010-01-01	2021-01-09	1	9
12	Merry Christmas II You	Mariah Carey	Oh Santa!	2010-01-01	2020-12-19	76	76
13	Relapse: Refill	Eminem	Forever	2009-05-15	2020-03-14	64	64
14	The Definition of X: Pick Of The Litter	DMX	Ruff Ryders' Anthem	2007-06-12	2021-04-24	16	16
15	The Definition of X: Pick Of The Litter	DMX	X Gon' Give It To Ya	2007-06-12	2021-04-24	46	46
16	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2019-01-05	45	45
17	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2020-01-04	36	36
18	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2020-12-19	36	40
19	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2020-12-26	29	29
20	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2021-01-02	10	10
21	Johnny B. Goode/His Complete '50s Chess Recordings	Chuck Berry	Run Rudolph Run	2007-01-01	2021-01-09	10	29
22	Songs for Christmas	Nat King Cole	Deck the Halls	2004-01-01	2020-12-26	47	47
23	Songs for Christmas	Nat King Cole	Deck the Halls	2004-01-01	2021-01-02	43	43
24	Meteora	Linkin Park	Numb	2003-03-24	2017-08-12	11	34
25	Meteora (Bonus Edition)	Linkin Park	Numb	2003-03-24	2017-08-12	11	34
26	A Day Without Rain	Enya	Only Time	2000-11-11	2013-12-07	10	43
27	Hybrid Theory	Linkin Park	In the End	2000-10-24	2017-08-12	2	37
28	Hybrid Theory (Bonus Edition)	Linkin Park	In the End	2000-10-24	2017-08-12	2	37
29	It's Dark And Hell Is Hot	DMX	Ruff Ryders' Anthem	1998-05-01	2021-04-24	16	16
30	The Deram Anthology 1966 - 1968	David Bowie	Space Oddity	1997-06-09	2016-01-30	15	42
31	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Billie Jean	1995-06-16	2014-06-07	1	14
32	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2013-11-16	4	42
33	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2014-11-15	4	35
34	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2015-11-21	4	45
35	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2018-11-10	4	31
36	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2019-11-09	4	44
37	HIStory - PAST, PRESENT AND FUTURE - BOOK I	Michael Jackson	Thriller	1995-06-16	2020-11-14	4	48
38	Merry Christmas	Mariah Carey	All I Want for Christmas Is You	1994-11-01	2012-12-22	29	29
39	Merry Christmas	Mariah Carey	All I Want for Christmas Is You	1994-11-01	2012-12-29	25	25

Query 2 - FOL

Find acousticness, danceability, energy and loudness of Songs of Artists that do the ‘rock’ Genre and that reached the #1 of the Billboard Hot 100 Chart

$$
\begin{aligned}
q = \{(&\boldsymbol{song\_name}, \boldsymbol{artist\_name}, \boldsymbol{acousticness}, \boldsymbol{danceability}, \boldsymbol{energy}, \boldsymbol{loudness}, \boldsymbol{billboard\_date}) \quad | \\
&\exists\, ar\_id, ar\_pop, foll, genre, s\_id, s\_pop, t\_num, dur, exp, prev\_url, woc, peak\_rank, i, k, l, m, s, t, ts. \\
&Artist(ar\_id, \boldsymbol{artist\_name}, ar\_pop, foll) \quad \wedge \\
&doesGenre(ar\_id, 'rock') \quad \wedge \\
&isSongOf(s\_id, ar\_id) \quad \wedge \\
&Song(s\_id, \boldsymbol{song\_name}, s\_pop, t\_num, dur, exp, prev\_url) \quad \wedge \\
&BillboardSong(s\_id, 1, \boldsymbol{billboard\_date}, woc, peak\_rank) \quad \wedge \\
&AudioFeature(s\_id, \boldsymbol{acousticness}, \boldsymbol{danceability}, \boldsymbol{energy}, i, k, l, \boldsymbol{loudness}, m, s, t, ts)\}
\end{aligned}
$$

Query 2 - SQL

Find acousticness, danceability, energy and loudness of Songs of Artists that do the ‘rock’ Genre and that reached the #1 of the Billboard Hot 100 Chart

```
SELECT DISTINCT s.name as song_name, ar.name as artist_name,
                af.acousticness, af.danceability, af.energy,
                af.loudness, bhs.rank, bhs.date as week_of_date
FROM artists as ar, songs as s, is_song_of as iso,
     billboard_hot100_songs as bhs, does_genre as dg,
     audio_features as af
WHERE ar.id = iso.artist_id AND
      s.id = iso.track_id AND
      s.id = bhs.track_id AND
      dg.artist_id = ar.id AND
      af.track_id = s.id AND
      dg.genre = 'rock' AND
      bhs.rank = 1
ORDER BY week_of_date DESC
```

	song_name character varying (1000)	artist_name character varying (500)	acousticness real	danceability real	energy real	loudness double precision	rank integer	date date
1	Havana	Young Thug	0.186	0.768	0.517	-4.32299995422363	1	2018-01-27
2	With Arms Wide Open	Creed	0.0037	0.414	0.533	-8.44499969482422	1	2000-11-11
3	With Arms Wide Open	Creed	0.00425	0.41	0.539	-8.41199970245361	1	2000-11-11
4	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	2000-01-08
5	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	2000-01-01
6	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-12-25
7	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-12-18
8	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-12-11
9	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-12-04
10	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-11-27
11	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-11-20
12	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-11-13
13	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-11-06
14	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-10-30
15	Smooth	Santana	0.189	0.521	0.815	-15.0869998931885	1	1999-10-23
16	Have You Ever Really Loved A Woman?	Bryan Adams	0.535	0.566	0.477	-10.1829996109009	1	1995-07-01
17	Have You Ever Really Loved A Woman?	Bryan Adams	0.536	0.57	0.478	-10.1850004196167	1	1995-07-01
18	Have You Ever Really Loved A Woman?	Bryan Adams	0.55	0.563	0.545	-7.76200008392334	1	1995-07-01
19	Have You Ever Really Loved A Woman?	Bryan Adams	0.535	0.566	0.477	-10.1829996109009	1	1995-06-24
20	Have You Ever Really Loved A Woman?	Bryan Adams	0.536	0.57	0.478	-10.1850004196167	1	1995-06-24
21	Have You Ever Really Loved A Woman?	Bryan Adams	0.55	0.563	0.545	-7.76200008392334	1	1995-06-24
22	Have You Ever Really Loved A Woman?	Bryan Adams	0.535	0.566	0.477	-10.1829996109009	1	1995-06-17
23	Have You Ever Really Loved A Woman?	Bryan Adams	0.536	0.57	0.478	-10.1850004196167	1	1995-06-17
24	Have You Ever Really Loved A Woman?	Bryan Adams	0.55	0.563	0.545	-7.76200008392334	1	1995-06-17
25	Have You Ever Really Loved A Woman?	Bryan Adams	0.535	0.566	0.477	-10.1829996109009	1	1995-06-10
26	Have You Ever Really Loved A Woman?	Bryan Adams	0.536	0.57	0.478	-10.1850004196167	1	1995-06-10
27	Have You Ever Really Loved A Woman?	Bryan Adams	0.55	0.563	0.545	-7.76200008392334	1	1995-06-10
28	Have You Ever Really Loved A Woman?	Bryan Adams	0.535	0.566	0.477	-10.1829996109009	1	1995-06-03

Query 3 - FOL

Top 2 songs on Billboard in the same week of two different artists that belong to two different labels but of the same Group of labels.

$$
\begin{aligned}
q = \{(&\boldsymbol{an1}, \boldsymbol{sn1}, \boldsymbol{r1}, \boldsymbol{an2}, \boldsymbol{sn2}, \boldsymbol{r2}, \boldsymbol{group}, \boldsymbol{week\_of\_date}) \quad | \\
&\exists\ sid1, woc1, pr1, sid2, woc2, pr2, arid1, arid2, sp1, tn1, d1, e1, pu1, sp2, tn2, d2, e2, pu2, ap1, f1, ap2, f2, label1, label2\ . \\
&BillboardSong(s\_id1, \boldsymbol{r1}, \boldsymbol{week\_of\_date}, woc1, pr1) \ \wedge \ BillboardHot100Song(s\_id2, \boldsymbol{r2}, \boldsymbol{week\_of\_date}, woc2, pr2) \\
&isSongOf(s\_id1, ar\_id1) \ \wedge \ isSongOf(s\_id2, ar\_id2) \ \wedge \\
&Song(s\_id1, \boldsymbol{sn1}, sp1, tn1, d1, e1, pu1) \ \wedge \ Song(s\_id2, \boldsymbol{sn2}, sp2, tn2, d2, e2, pu2) \ \wedge \\
&Artist(ar\_id1, \boldsymbol{an1}, ap1, f1) \ \wedge \ Artist(ar\_id2, \boldsymbol{an2}, ap2, f2) \\
&hasLabel(a\_id1, label1) \ \wedge \ hasLabel(a\_id1, label2) \ \wedge \\
&isLabelInGroup(label1, \boldsymbol{group}) \ \wedge \ isLabelInGroup(label2, \boldsymbol{group}) \\
&(r1 = 1) \ \wedge \ (r2 = 2) \ \wedge \ \neg(ar\_id1 = ar\_id2) \ \wedge \ \neg(label1 = label2)\}
\end{aligned}
$$

Query 3 - SQL

Top 2 songs on Billboard in the same week of two different artists that belong to two different labels but of the same Group of labels.

```
SELECT DISTINCT ar1.name as artist1_name, s1.name as song_name, bhs1.rank as s1_rank, ar2.name as artist2_name,
                s2.name as song2_name, bhs2.rank as s2_rank2, ilig1.group_name, bhs1.date as week_of_date
FROM billboard_hot100_songs as bhs1, billboard_hot100_songs as bhs2, is_song_of as iso1, is_song_of as iso2,
     songs as s1, songs as s2, artists as ar1, artists as ar2, has_label as hl1, has_label as hl2,
     is_label_in_group as ilig1, is_label_in_group as ilig2
WHERE bhs1.rank = 1 AND
      bhs2.rank = 2 AND
      bhs1.track_id = s1.id AND
      bhs2.track_id = s2.id AND
      iso1.track_id = s1.id AND
      iso2.track_id = s2.id AND
      iso1.artist_id = ar1.id AND
      iso2.artist_id = ar2.id AND
      hl1.artist_id = iso1.artist_id AND
      hl2.artist_id = iso1.artist_id AND
      ilig1.label_name = hl1.label_name AND
      ilig2.label_name = hl2.label_name AND
      ilig1.group_name = ilig2.group_name AND
      ar1.id <> ar2.id AND
      hl1.label_name <> hl2.label_name AND
      bhs1.date = bhs2.date
ORDER BY week_of_date DESC
```

	artist1_name character varying (500)	song_name character varying (1000)	s1_rank integer	artist2_name character varying (500)	song2_name character varying (1000)	s2_rank2 integer	group_name character varying (50)	date date
1	Eminem	Lose Yourself	1	Kelly Rowland	Dilemma	2	Interscope Geffen A&M	2002-11-09
2	Eminem	Lose Yourself	1	Nelly	Dilemma	2	Interscope Geffen A&M	2002-11-09
3	50 Cent	In Da Club	1	Jennifer Lopez	All I Have	2	Interscope Geffen A&M	2003-03-08
4	50 Cent	In Da Club	1	LL Cool J	All I Have	2	Interscope Geffen A&M	2003-03-08
5	50 Cent	In Da Club	1	Jennifer Lopez	All I Have	2	Interscope Geffen A&M	2003-03-15
6	50 Cent	In Da Club	1	LL Cool J	All I Have	2	Interscope Geffen A&M	2003-03-15
7	50 Cent	In Da Club	1	Jennifer Lopez	All I Have	2	Interscope Geffen A&M	2003-03-22
8	50 Cent	In Da Club	1	LL Cool J	All I Have	2	Interscope Geffen A&M	2003-03-22
9	50 Cent	In Da Club	1	R. Kelly	Ignition	2	Interscope Geffen A&M	2003-03-29
10	50 Cent	In Da Club	1	R. Kelly	Ignition	2	Interscope Geffen A&M	2003-04-05
11	50 Cent	In Da Club	1	R. Kelly	Ignition	2	Interscope Geffen A&M	2003-04-12
12	50 Cent	In Da Club	1	R. Kelly	Ignition	2	Interscope Geffen A&M	2003-04-19
13	50 Cent	In Da Club	1	R. Kelly	Ignition	2	Interscope Geffen A&M	2003-04-26
14	50 Cent	In Da Club	1	Sean Paul	Get Busy	2	Interscope Geffen A&M	2003-05-03
15	Sean Paul	Get Busy	1	50 Cent	In Da Club	2	Island Records	2003-05-10
16	Sean Paul	Get Busy	1	50 Cent	21 Questions	2	Island Records	2003-05-17
17	Sean Paul	Get Busy	1	Nate Dogg	21 Questions	2	Island Records	2003-05-17
18	Sean Paul	Get Busy	1	50 Cent	21 Questions	2	Island Records	2003-05-24
19	Sean Paul	Get Busy	1	Nate Dogg	21 Questions	2	Island Records	2003-05-24
20	50 Cent	21 Questions	1	Sean Paul	Get Busy	2	Interscope Geffen A&M	2003-05-31
21	50 Cent	21 Questions	1	Sean Paul	Get Busy	2	Interscope Geffen A&M	2003-06-07
22	50 Cent	21 Questions	1	Sean Paul	Get Busy	2	Interscope Geffen A&M	2003-06-14
23	50 Cent	21 Questions	1	Sean Paul	Get Busy	2	Interscope Geffen A&M	2003-06-21

Query 4 - FOL

Find which albums contain a song that is a featuring between «Machine Gun Kelly» and another artist who belongs to his same label

$$
\begin{aligned}
q = \{(&\boldsymbol{aname2}, \boldsymbol{sname}, \boldsymbol{albname}, \boldsymbol{aname3}, \boldsymbol{releasedate}) \quad | \\
&\exists aid1, p, f, sid1, sid2, aid2, p2, f2, label1, label2, sp, tn, d, e, pu, albid, at, abp, aid3, p3, f3. \\
&Artist(aid1, 'Machine\ Gun\ Kelly', p, f) \quad \wedge \quad \text{is}SongOf\ (sid1, aid1) \quad \wedge \quad \text{is}SongOf\ (sid2, aid2) \quad \wedge \\
&(sid1 = sid2) \quad \wedge \quad \neg(aid1 = aid2) \quad \wedge \quad Artist(aid2, \boldsymbol{aname2}, p2, f2) \quad \wedge \\
&hasLabel(aid1, label1) \quad \wedge \quad hasLabel(aid2, label2) \quad \wedge \quad (label1 = label2) \quad \wedge \\
&Song(sid1, \boldsymbol{sname}, sp, tn, d, e, pu) \quad \wedge \quad isSongInAlbum(sid1, albid) \quad \wedge \\
&Album(albid, \boldsymbol{albname}, at, abp, \boldsymbol{releasedate}) \wedge \\
&\text{is}AlbumOf(albid, aid3) \quad \wedge \quad Artist(aid3, \boldsymbol{aname3}, p3, f3)\}
\end{aligned}
$$

Query 4 - SQL

Find which albums contain a song that is a featuring between «Machine Gun Kelly» and another artist who belongs to his same label

```
SELECT DISTINCT ar2.name as artist_name, s.name as song_name, al.name as album_name,
                ar3.name as album_of, al.release_date
FROM is_song_of AS iso1, is_song_of AS iso2, artists AS ar1, artists AS ar2,has_label AS hl1
     has_label AS hl2, songs as s, albums as al, is_song_in_album as isia, is_album_of as iao, artists as ar3
WHERE ar1.name = 'Machine Gun Kelly' AND
      ar1.id = iso1.artist_id  AND
      iso1.artist_id <> iso2.artist_id AND
      iso1.track_id = iso2.track_id AND
      iso1.track_id = s.id AND
      iso1.track_id = isia.track_id AND
      isia.album_id = al.id AND
      isia.album_id = iao.album_id AND
      iao.artist_id = ar3.id AND
      ar2.id = iso2.artist_id AND
      hl1.artist_id = iso1.artist_id AND
      hl2.artist_id = iso2.artist_id AND
      hl1.label_name = hl2.label_name
```

	artist_name character varying (500)	song_name character varying (1000)	album_name character varying (500)	album_of character varying (500)	release_date date
1	blackbear	e.z. (feat. Machine Gun Kelly)	cybersex	blackbear	2017-11-27
2	blackbear	End Of The Road	Lace Up (Deluxe)	Machine Gun Kelly	2012-01-01
3	blackbear	my ex's best friend (with blackbear)	Tickets To My Downfall	Machine Gun Kelly	2020-09-25
4	blackbear	my ex's best friend (with blackbear)	Tickets To My Downfall (SOLD OUT Deluxe)	Machine Gun Kelly	2020-09-29
5	Papa Roach	Sunrise Trailer Park (feat. MGK)	Crooked Teeth (Deluxe Version)	Papa Roach	2017-05-19
6	X Ambassadors	Home (with Machine Gun Kelly, X Ambassadors & Bebe Rexh...	Home (with Machine Gun Kelly, X Ambassad...	Bebe Rexha	2017-11-18
7	X Ambassadors	Home (with Machine Gun Kelly, X Ambassadors & Bebe Rexh...	Home (with Machine Gun Kelly, X Ambassad...	Machine Gun Kelly	2017-11-18
8	X Ambassadors	Home (with Machine Gun Kelly, X Ambassadors & Bebe Rexh...	Home (with Machine Gun Kelly, X Ambassad...	X Ambassadors	2017-11-18
9	YUNGBLUD	acting like that (feat. Machine Gun Kelly)	acting like that (feat. Machine Gun Kelly)	Machine Gun Kelly	2020-12-02
10	YUNGBLUD	acting like that (feat. Machine Gun Kelly)	acting like that (feat. Machine Gun Kelly)	YUNGBLUD	2020-12-02
11	YUNGBLUD	acting like that (feat. Machine Gun Kelly)	weird!	YUNGBLUD	2020-12-04
12	YUNGBLUD	body bag (feat. YUNGBLUD & Bert McCracken of The Used)	Tickets To My Downfall (SOLD OUT Deluxe)	Machine Gun Kelly	2020-09-29
13	YUNGBLUD	I Think I'm OKAY (with YUNGBLUD & Travis Barker)	Hotel Diablo	Machine Gun Kelly	2019-07-05

Query 5 - FOL

Artists that have been influenced by «Nirvana» and that have done a featuring between them that reached the Billboard Hot 100

$$
\begin{aligned}
q = \{(&\boldsymbol{song\_name}, \boldsymbol{foll1\_name}, \boldsymbol{foll2\_name}, \boldsymbol{peak\_rank}) \quad | \\
&\exists\, infl\_aid, ip, if, foll1\_aid, foll2\_aid, sid1, sid2, sp, tn, d, e, pu, fp1, fp2, ff1, ff2, r, wod, woc. \\
&Artist(infl\_aid, 'Nirvana', ip, if) \quad \wedge \quad hasInfluenced(infl\_aid, foll1\_aid) \quad \wedge \\
&hasInfluenced(infl\_aid, foll2\_aid) \quad \wedge \quad \neg(foll1\_aid = foll2\_aid) \quad \wedge \\
&isSongOf(sid1, foll1\_aid) \quad \wedge \quad isSongOf(sid2, foll2\_aid) \quad \wedge \quad (sid1 = sid2) \quad \wedge \\
&Song(sid1, \boldsymbol{song\_name}, sp, tn, d, e, pu) \quad \wedge \\
&Artist(foll1\_aid, \boldsymbol{foll1\_name}, fp1, ff1) \quad \wedge \\
&Artist(foll2\_aid, \boldsymbol{foll2\_name}, fp2, ff2) \quad \wedge \\
&BillboardHot100Song(sid1, r, wod, woc, \boldsymbol{peak\_rank})\}
\end{aligned}
$$

Query 5 - SQL

Artists that have been influenced by «Nirvana» and that have done a featuring between them that reached the Billboard Hot 100

```
SELECT DISTINCT s.name as song_name, ar1.name as ar1_name, ar2.name as ar2_name, bhs.peak_rank
FROM has_influenced as hi1, has_influenced as hi2, is_song_of as iso1, is_song_of as iso2, artists as ar1,
     artists as ar2, songs as s, artists as influencer_artist, billboard_hot100_songs as bhs
WHERE influencer_artist.name = 'Nirvana' AND
      hi1.influencer_id = influencer_artist.id AND
      hi2.influencer_id = influencer_artist.id AND
      hi1.follower_id <> hi2.follower_id AND
      iso1.artist_id = hi1.follower_id AND
      iso2.artist_id = hi2.follower_id AND
      iso1.track_id = iso2.track_id AND
      iso1.track_id = s.id AND
      hi1.follower_id = ar1.id AND
      hi2.follower_id = ar2.id AND
      iso1.track_id = bhs.track_id
```

	song_name character varying (1000)	ar1_name character varying (500)	ar2_name character varying (500)	peak_rank integer
1	Under Pressure	My Chemical Romance	The Used	41
2	Under Pressure	My Chemical Romance	The Used	54
3	Under Pressure	The Used	My Chemical Romance	41
4	Under Pressure	The Used	My Chemical Romance	54

Query 6 - FOL

Lyrics of songs with popularity ≥ 80, of artists with popularity ≥ 80, that do either *«alternative rock»* or *«alternative metal»* or *«punk»* music genre

$$
\begin{aligned}
q = \{(&\boldsymbol{aname}, \boldsymbol{apopularity}, \boldsymbol{sname}, \boldsymbol{spopularity}, \boldsymbol{lyrics}) \quad | \\
&(\exists\, aid, af, sid, tn, d, e, pu. \\
&Artist(aid, \boldsymbol{aname}, apopularity, af) \ \wedge \ (apopularity \geq 80) \ \wedge \ isSongOf(sid, aid) \ \wedge \\
&Song(sid, \boldsymbol{sname}, \boldsymbol{spopularity}, tn, d, e, pu) \wedge (spopularity \geq 80) \ \wedge \\
&doesGenre(aid, \text{"}alternative\ rock\text{"}) \ \wedge \ hasLyrics(sid, \boldsymbol{lyrics})) \\
&\vee \\
&(\exists\, aid, af, sid, tn, d, e, pu. \\
&Artist(aid, \boldsymbol{aname}, apopularity, af) \ \wedge \ (apopularity \geq 80) \ \wedge \ isSongOf(sid, aid) \ \wedge \\
&Song(sid, \boldsymbol{sname}, \boldsymbol{spopularity}, tn, d, e, pu) \wedge (spopularity \geq 80) \ \wedge \\
&doesGenre(aid, \text{"}alternative\ metal\text{"}) \ \wedge \ hasLyrics(sid, \boldsymbol{lyrics})) \\
&\vee \\
&(\exists\, aid, af, sid, tn, d, e, pu. \\
&Artist(aid, \boldsymbol{aname}, apopularity, af) \ \wedge \ (apopularity \geq 80) \ \wedge \ isSongOf(sid, aid) \ \wedge \\
&Song(sid, \boldsymbol{sname}, \boldsymbol{spopularity}, tn, d, e, pu) \wedge (spopularity \geq 80) \ \wedge \\
&doesGenre(aid, \text{"}punk\text{"}) \ \wedge \ hasLyrics(sid, \boldsymbol{lyrics}))\}
\end{aligned}
$$

Query 6 - SQL

```
(SELECT DISTINCT ar.name as artist_name, ar.popularity as artist_popularity,
                 s.name as song_name, s.popularity as song_popularity, hl.lyrics
FROM songs as s, artists as ar, is_song_of as iso, has_lyrics as hl, does_genre as dg
WHERE ar.popularity ⩾ 80 AND
       iso.artist_id = ar.id AND
       iso.track_id = s.id AND
       s.popularity ⩾ 80 AND
       iso.artist_id = dg.artist_id AND
       dg.genre = 'alternative rock' AND
       hl.track_id = s.id
)
UNION
(SELECT DISTINCT ar.name as artist_name, ar.popularity as artist_popularity,
                 s.name as song_name, s.popularity as song_popularity, hl.lyrics
FROM songs as s, artists as ar, is_song_of as iso, has_lyrics as hl, does_genre as dg
WHERE ar.popularity ⩾ 80 AND
       iso.artist_id = ar.id AND
       iso.track_id = s.id AND
       s.popularity ⩾ 80 AND
       iso.artist_id = dg.artist_id AND
       dg.genre = 'alternative metal' AND
       hl.track_id = s.id
)
UNION
(SELECT DISTINCT ar.name as artist_name, ar.popularity as artist_popularity,
                 s.name as song_name, s.popularity as song_popularity, hl.lyrics
FROM songs as s, artists as ar, is_song_of as iso, has_lyrics as hl, does_genre as dg
WHERE ar.popularity ⩾ 80 AND
       iso.artist_id = ar.id AND
       iso.track_id = s.id AND
       s.popularity ⩾ 80 AND
       iso.artist_id = dg.artist_id AND
       dg.genre = 'punk' AND
       hl.track_id = s.id
) ORDER BY artist_name
```

Lyrics of songs with popularity ≥ 80, of artists with popularity ≥ 80, that do either «*alternative rock*» or «*alternative metal*» or «*punk*» music genres

	artist_name character varying (500)	artist_popularity integer	song_name character varying (1000)	song_popularity integer	lyrics character varying (10000000)
1	blink-182	81	All The Small Things	82	[Verse 1: Tom DeLonge] All the small things True care, truth brings I'll
2	Foo Fighters	81	Everlong	80	[Verse 1] Hello, I've waited here for you Everlong Tonight, I throw myse
3	Green Day	83	American Idiot	81	[Verse 1] Don't wanna be an American idiot Don't want a nation under
4	Green Day	83	Basket Case	81	[Verse 1] Do you have the time To listen to me whine About nothing ar
5	Linkin Park	87	Numb	83	[Verse 1: Chester Bennington] I'm tired of being what you want me to l
6	Linkin Park	87	In the End	85	[Verse 1: Mike Shinoda & Chester Bennington] (It starts with one) One
7	Nirvana	84	Smells Like Teen Spirit	83	[Verse 1] Load up on guns, bring your friends It's fun to lose and to pre
8	Radiohead	81	Creep	83	[Verse 1] When you were here before Couldn't look you in the eye You'
9	Red Hot Chili Peppers	86	Otherside	80	[Chorus] How long, how long will I slide? Well, separate my side I don't
10	Red Hot Chili Peppers	86	Under the Bridge	82	[Instrumental Intro] [Verse 1] Sometimes I feel like I don't have a partn
11	Red Hot Chili Peppers	86	Californication	83	[Verse 1] Psychic spies from China try to steal your mind's elation And
12	The Killers	81	Mr. Brightside	83	[Verse] Coming out of my cage and I've been doing just fine Gotta, got
13	The Strokes	80	The Adults Are Talking	80	[Verse 1] They've been sayin' you're sophisticated They're complainin',

Thanks!