Project of Decision Support Systems

Module II

Anna Monreale, Cristiano Landi

Introduction

This project consists of several assignments that must be completed in sequence. First, you have to analyze the data, design the database, and populate it. Next, you have to solve problems on the database you created using SQL Server Integration Services (SSIS), performing computations on the client side. Minimize the use of SQL commands within nodes and rely on native SSIS components whenever possible[1]. Then, you have to create a datacube based on your database and use it to answer business questions. Document the datacube creation process in your report and write MultiDimensional eXpressions (MDX) queries in SQL Server Management Studio (SSMS). Finally, you have to design interactive dashboards. We recommend using the tool demonstrated in class, but you may use alternatives such as MicroStrategy, provided the dashboard is interactive and can connect to the database. In the report, include a screenshot and a short explanation of why each dashboard is relevant to the business.

In the following, you can find a set of incremental assignments, each with a brief description of what you are required to produce and which tools you can use for the task. The project aims to simulate a decision support system for a music streaming company. Attached to this document, you can find 2 distinct files: **tracks.json** and **artists.xml**.

- **tracks.json**: contains the main dataset with song details, lyrics information, and the number of streams one month after release.
- **artists.xml**: contains information about the artists, including gender, age, birthplace, and a short description.

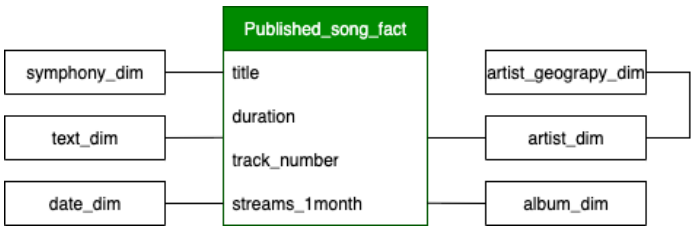

Figure 1: Data warehouse schema of reference. Fact table in blue.

[1]SQL commands run only on the server, while SSIS nodes distribute operations between server and client

Using Python

All tasks must be completed without using any `pandas`-like package, unless explicitly specified otherwise.

***Assignment 1**: data understanding*

Understand the data you are working with: Are there missing values? Can they be recovered or filled *easily*? Can you integrate external data (e.g., hierarchical GeoHash, Uber H3, or Google S2 encodings for spatial data) with reasonable effort? Can you derive additional information from the provided data (for example, from textual fields)?

For this assignment, you can use any software/package you want!

***Assignment 2**: data cleaning*

Given the information collected in the previous assignment, address the problem related to the missing data (if any) and integrate the additional data (if any).

***Assignment 3**: song profiling*

Given the information collected in the previous assignment, build some song categories based on the lyrics or/and the melodic information available in the data.

For this assignment, you can use any software/package you want!

***Assignment 4**: DW Schema*

It's time to switch from operational databases to analysis-oriented data warehouses. To design the DW schema, the fact table should capture details about the number of streams one month after release. You can use the data warehouse schema in Figure 1 as a reference. Next, create the data warehouse tables on the database server. You must use the database named *Group_ID_DB* (example: Group_01_DB) as specified in the credentials' email.
Please note that the DW schema in Figure 1 is just a suggestion; you can modify it as you prefer. You can use both Python and SQL Server Management Studio to create the DW.

***Assignment 5**: Data preparation*

Write a Python program that splits the data into different files, one for each table in the data warehouse proposed in the previous step.

***Assignment 6**: Data uploading with python*

Write a Python program that populates the database *Group_ID_DB* according to the schema relations with all the data you prepared in Assignment 4.
Please note that this operation could take a while, so design the code accordingly!

***Assignment 7**: Data uploading*

Duplicate each table without the records, renaming them as TABLENAME_SSIS. Then, create a SSIS project that populates the new set of tables TABLENAME_SSIS with 30% of the data you prepared in Assignment 4.

At this point, you should have two fact tables along with their corresponding (duplicated) dimension tables. From now on, **perform all assignments using the fact table that contains all the data**[2].

[2]We suggest using the smaller fact table to verify the correctness of your results manually

Using Microsoft SQL Server Integration Services

Assignment 8

For each year, list all artists ordered by the total number of songs published.

Assignment 9

For each region, compute the *summer–winter score*, defined as the ratio between the number of songs released in summer and those released in winter. Also, compute the same score using the number of streams after one month.

Assignment 10

For each song category (see Assignment 3) and for each region corresponding to the birthplace of the main artist, compute the ratio between the cumulative number of streams for that category and region, and the total number of streams for the same category in all other regions.

Assignment 11

For each artist, compute the following trending statistics:

1. *Trending percentage*: the percentage of trending songs, i.e., songs where the number of streams exceeds the artist's average by at least one standard deviation.
2. *Trending factor*: the difference between the number of trending songs and flopping songs, i.e., songs where the number of streams falls below the artist's average by at least one standard deviation. This value must be normalized such that -1 indicates only flopping songs, while 1 indicates only trending songs.

Edge cases must be considered. For example, for an artist with only one song, that song may be considered trending only if it exceeds the average streams of the first published songs of all other artists. All edge cases and the strategy used to address them must be documented in the report.
Finally, compute the same statistics for songs in which the artist is featured. In this case, a song is considered trending only if it is trending for the main artist.

Assignment 12

Based on the analytical results of the previous queries and the insights from the data understanding phase, define an **interesting** query and answer it using SSIS. In your report, explain why this query is important from a business perspective.

Assignment 13

Now imagine that the streaming company you are working with is providing consultancy to a record label. Based on the analytical results of the previous queries and the insights from the data understanding phase, define an **interesting** query and answer it using SSIS. In your report, explain why this query is important from the record label's business perspective.

Using MDX, SQL Management Studio, and PowerBI

Assignment 14

Build a datacube from the tables in your database, defining the appropriate hierarchies. Create the necessary measures based on the queries you need to answer.

Assignment 15

For each month, show the total streams for each region and the overall total for the country.

Assignment 16

Compute the average yearly artist streams: for each artist, calculate the total streams of all their songs. If the artist is the main artist, weight the streams by 0.8; otherwise by 0.2. Then, compute the sum of these values and divide it by the grand total of streams in the same year.

Assignment 17

For each song category, show the percentage increase or decrease in streams compared to the previous year.

Assignment 18

For each season, list the categories where the number of streams exceeds the average of the same season in the previous year. Also, report the increase as a percentage.

Assignment 19

Propose and solve a query that reveals interesting and **non-trivial** insights from the previous analyses. In your report, explain why this query is important from a business perspective.

Assignment 20

Create a dashboard showing the total streams for each artist's birthplace by song category.

Assignment 21

Create a plot or dashboard of your choice that provides meaningful insights about the lyrics data available in your cube.

Assignment 22

Create a plot or dashboard of your choice that provides meaningful insights about trending song categories over time.

Delivery Instructions

When your project is ready, compress all files, including the PDF report, into a .zip file named `LDS_project_GroupID.zip`. Upload this file using the correct Google form with your *studenti.unipi.it* email. If you encounter issues with the form, email all teachers with the subject: **[LDS] Project Delivery Group_ID**. Do not include the original dataset in the zip file.

The project **can** be delivered in two parts:

1. Code from the first assignment through assignment 14 (inclusive) **must** be delivered by December 19th. Google form: forms.gle/TFSKgfZAFPbiDjjU8

2. Remaining assignments (MDX queries and dashboards) and the report **must** be delivered by December 27th. **Warning**: starting from December 20th, you will no longer be able to modify your group database. Google form: forms.gle/MNgi8j6GZxK5iFih9