



ARCHITECTING AND DESIGNING A DATA WAREHOUSE FOR REPORTING AND OLAP

AN ETL PROJECT USING YELP AND CLIMATE DATA

NAME: **PIERRE MORA ERAZO**

DATE: **06 / 15 / 2024**

PROJECT OVERVIEW



Objective: To architect and design a Data Warehouse (DWH) for the purpose of reporting and Online Analytical Processing (OLAP) using skills acquired in the Designing Data Systems course.

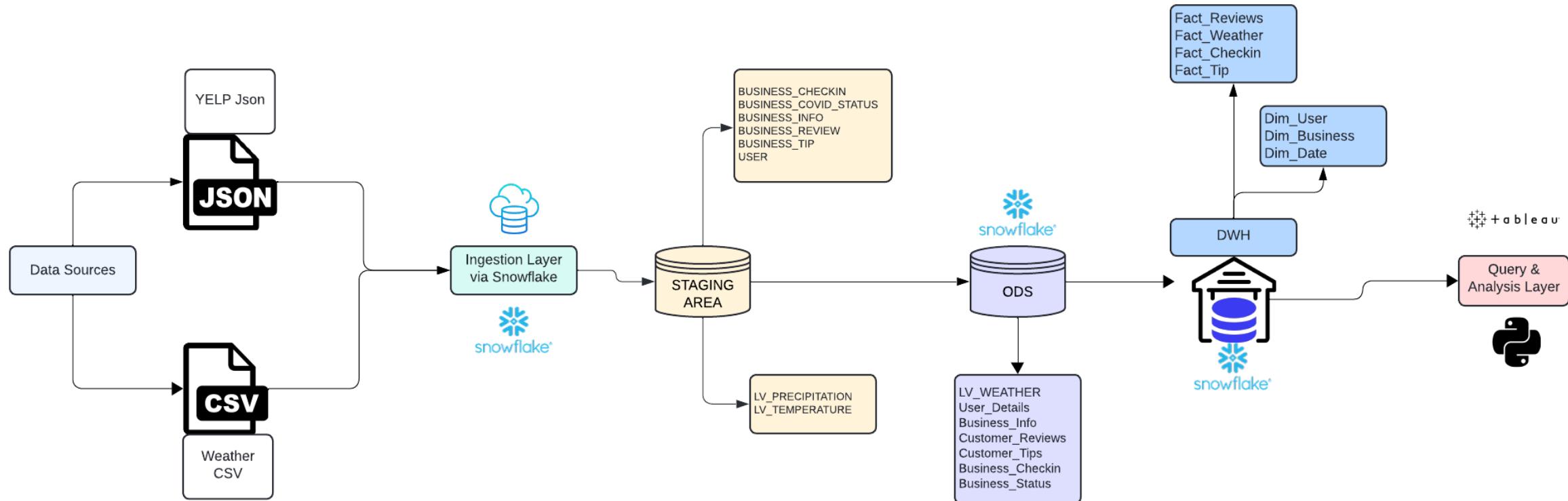
Data Sources:

- **Yelp Data:** Business information, reviews, tips, and check-ins.
- **Climate Data:** Temperature and precipitation observations.

Tools Used: Snowflake for data storage and transformation.

DATA ARCHITECTURE FOR ETL PROCESS

- Ingesting and Migrating Data from Sources to Data Warehouse



CREATING THE STAGING SCHEMA

- Creation of the staging environment in Snowflake and the upload process for Yelp and climate data:

```
TYPE SQL STATEMENTS OR MCTP
PMORAER#COMPUTE_WH@(no database).(no schema)>USE DATABASE YELP_COVID;
+-----+
| status
+-----+
| Statement executed successfully.
+-----+
1 Row(s) produced. Time Elapsed: 0.114s
PMORAER#COMPUTE_WH@YELP_COVID.PUBLIC>CREATE SCHEMA STAGING;
+-----+
| status
+-----+
| Schema STAGING successfully created.
+-----+
```

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE OR REPLACE FILE FORMAT mycsvformat
  TYPE = 'CSV'
  COMPRESSION = 'AUTO'
  FIELD_DELIMITER = ','
  RECORD_DELIMITER = '\n'
  SKIP_HEADER = 1
  ERROR_ON_COLUMN_COUNT_MISMATCH = TRUE
  NULL_IF = ('NULL', 'null')
  EMPTY_FIELD_AS_NULL = TRUE;
+-----+
| status
+-----+
| File format MYCSVFORMAT successfully created.
+-----+
1 Row(s) produced. Time Elapsed: 0.243s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE OR REPLACE STAGE MY_CSV_STAGE
  FILE_FORMAT = mycsvformat;
+-----+
| status
+-----+
| Stage area MY_CSV_STAGE successfully created.
+-----+
```

```
1 Row(s) produced. Time Elapsed: 0.340s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE OR REPLACE FILE FORMAT myjsonformat
  TYPE = 'JSON'
  STRIP_OUTER_ARRAY = TRUE;
001003 (42000): SQL compilation error:
  syntax error line 2 at position 48 unexpected 'outer_array'.
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE OR REPLACE FILE FORMAT myjsonformat
  TYPE = 'JSON'
  STRIP_OUTER_ARRAY = TRUE;
+-----+
| status
+-----+
| File format MYJSONFORMAT successfully created.
+-----+
1 Row(s) produced. Time Elapsed: 0.170s
```

UPLOADING CSV FILES TO STAGING SCHEMA

- Creating LV_TEMPERATURE table and uploading the CSV. File to the staging schema:

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE TABLE "LV_TEMPERATURE"(
    "DATE" VARCHAR(8),
    "MIN_TEMP" VARCHAR(10),
    "MAX_TEMP" VARCHAR(10),
    "NORMAL_MIN_TEMP" FLOAT,
    "NORMAL_MAX_TEMP" FLOAT);
+-----+
| status |
+-----+
| Table LV_TEMPERATURE successfully created. |
+-----+
1 Row(s) produced. Time Elapsed: 0.700s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/usw0023169-temperature-degreef.csv @MY_CSV_STAGE AUTO_COMPRESS=TRUE;
+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+
| usw0023169-temperature-degreef.csv | usw0023169-temperature-degreef.csv.gz | 816174 | 196592 | NONE | GZIP | UPLOADED | |
+-----+
1 Row(s) produced. Time Elapsed: 1.159s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "LV_TEMPERATURE" FROM @MY_CSV_STAGE/usw0023169-temperature-degreef.csv.gz file_format=mvcsvformat ON_ERROR = 'CONTINUE' PURGE = TRUE;
+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+
| mv_csv_stage/usw0023169-temperature-degreef.csv.gz | LOADED | 28241 | 28241 | 28241 | 0 | NULL | NULL | NULL | NULL |
+-----+
1 Row(s) produced. Time Elapsed: 0.265s
```

- Creating LV_PRECIPITATION table and uploading the CSV. File to the staging schema:

```
+-----+
| Row(s) produced. Time Elapsed: 0.725s |
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE TABLE "LV_PRECIPITATION"(
    "DATE" VARCHAR(8),
    "PRECIPITATION" VARCHAR(5),
    "PRECIPITATION_NORMAL" FLOAT
);
+-----+
| status |
+-----+
| Table LV_PRECIPITATION successfully created. |
+-----+
```

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/usw0023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv @MY_CSV_STAGE AUTO_COMPRESS=TRUE;
+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+
| usw0023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv | usw0023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv.gz | 528165 | 118992 | NONE | GZIP | UPLOADED | |
+-----+
1 Row(s) produced. Time Elapsed: 1.147s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "LV_PRECIPITATION" FROM @MY_CSV_STAGE/usw0023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv.gz file_format=mvcsvformat ON_ERROR = 'CONTINUE' PURGE = TRUE;
+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+
| mv_csv_stage/usw0023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv.gz | LOADED | 28241 | 28241 | 28241 | 0 | NULL | NULL | NULL | NULL |
+-----+
```

UPLOADING JSON FILES TO STAGING SCHEMA

- Creating BUSINESS_COVID_STATUS and BUSINESS_INFO Tables and uploading their respective JSON files to the staging schema:

```
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>CREATE TABLE "BUSINESS_COVID_STATUS"("BUSINESS_JSON" VARIANT);
+-----+
| status
+-----+
Table BUSINESS_COVID_STATUS successfully created.
1 Row(s) produced. Time Elapsed: 0.466s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>CREATE OR REPLACE STAGE my_json_stage
FILE_FORMAT = myjsonformat;
+-----+
| status
+-----+
Stage area MY_JSON_STAGE successfully created.
1 Row(s) produced. Time Elapsed: 0.434s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_academic_dataset_covid_features.json @my_json_stage AUTO_COMPRESS=TRUE;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
yelp_academic_dataset_covid_features.json | yelp_academic_dataset_covid_features.json.gz | 64835031 | 6954672 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 4.947s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>copy into BUSINESS_COVID_STATUS from @my_json_stage/yelp_academic_dataset_covid_features.json.gz file_format=myjsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+
my_json_stage/yelp_academic_dataset_covid_features.json.gz | LOADED | 209795 | 209795 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 5.499s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>CREATE TABLE "BUSINESS_INFO"("BUSINESS_INFO_JSON" VARIANT);
+-----+
| status
+-----+
Table BUSINESS_INFO successfully created.
1 Row(s) produced. Time Elapsed: 0.661s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_academic_dataset_business.json @my_json_stage AUTO_COMPRESS= TRUE parallel=4;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
yelp_academic_dataset_business.json | yelp_academic_dataset_business.json.gz | 118863795 | 20340592 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 13.454s
PMORAER#COMPUTE_WH@YELP_COVID-STAGING>COPY INTO "BUSINESS_INFO" FROM @my_json_stage/yelp_academic_dataset_business.json.gz file_format=myjsonformat on_error= 'skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+
my_json_stage/yelp_academic_dataset_business.json.gz | LOADED | 150346 | 150346 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

UPLOADING JSON FILES TO STAGING SCHEMA

- Creating BUSINESS_CHECKIN Table and uploading the respective JSON files to the staging schema:

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE TABLE "BUSINESS_CHECK_IN"("BUSINESS_CHECK_IN_JSON" VARIANT);
+-----+
| status
+-----+
| Table BUSINESS_CHECK_IN successfully created.
+-----+
1 Row(s) produced. Time Elapsed: 0.526s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_checkin.json @my_json_stage AUTO_COMPRESS= TRUE parallel=6;
+-----+
| source          | target          | source_size | target_size | source_compression | target_compression | status      | message
+-----+
| yelp_academic_dataset_checkin.json | yelp_academic_dataset_checkin.json.gz | 286958945 | 80199328 | NONE           | GZIP           | UPLOADED   |
+-----+
1 Row(s) produced. Time Elapsed: 87.528s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS_CHECK_IN" FROM my_json_stage/yelp_academic_dataset_checkin.json.gz file_format=myjsonformat on_error='skip_file';
001003 (42000): SQL compilation error:
syntax error line 1 at position 35 unexpected 'my_json_stage'.
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS_CHECK_IN" FROM @my_json_stage/yelp_academic_dataset_checkin.json.gz file_format=myjsonformat on_error='skip_file';
+-----+
| file           | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name
+-----+
| my_json_stage/yelp_academic_dataset_checkin.json.gz | LOADED | 131930      | 131930     | 1           | 0           | NULL        | NULL        | NULL        | NULL
+-----+
1 Row(s) produced. Time Elapsed: 12.194s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE TABLE "BUSINESS_REVIEW"("BUSINESS REVIEW_JSON" VARIANT);
```

UPLOADING JSON FILES TO STAGING SCHEMA

- Creating BUSINESS REVIEW Table and uploading the respective JSON file to the staging schema:

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_review_4.json @my_json_stage AUTO_COMPRESS= TRUE parallel=6;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
| yelp_academic_dataset_review_4.json | yelp_academic_dataset_review_4.json.gz | 534186718 | 218442272 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 104.867s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS REVIEW" FROM @my_json_stage/yelp_academic_dataset_review_4.json.gz file_format=jsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_review_4.json.gz | LOADED | 694506 | 694506 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 25.239s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_review_5.json @my_json_stage AUTO_COMPRESS= TRUE parallel=6;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
| yelp_academic_dataset_review_5.json | yelp_academic_dataset_review_5.json.gz | 534187004 | 218262784 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 104.768s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS REVIEW" FROM @my_json_stage/yelp_academic_dataset_review_5.json.gz file_format=jsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_review_5.json.gz | LOADED | 693888 | 693888 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 24.990s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_review_6.json @my_json_stage AUTO_COMPRESS= TRUE parallel=6;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
| yelp_academic_dataset_review_6.json | yelp_academic_dataset_review_6.json.gz | 534187484 | 218223184 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 105.146s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS REVIEW" FROM @my_json_stage/yelp_academic_dataset_review_6.json.gz file_format=jsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_review_6.json.gz | LOADED | 700857 | 700857 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 26.124s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_review_7.json @my_json_stage AUTO_COMPRESS= TRUE parallel=6;
+-----+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
| yelp_academic_dataset_review_7.json | yelp_academic_dataset_review_7.json.gz | 534186339 | 218089664 | NONE | GZIP | UPLOADED |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 104.603s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS REVIEW" FROM @my_json_stage/yelp_academic_dataset_review_7.json.gz file_format=jsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_review_7.json.gz | LOADED | 695442 | 695442 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 25.107s
```

UPLOADING JSON FILES TO STAGING SCHEMA

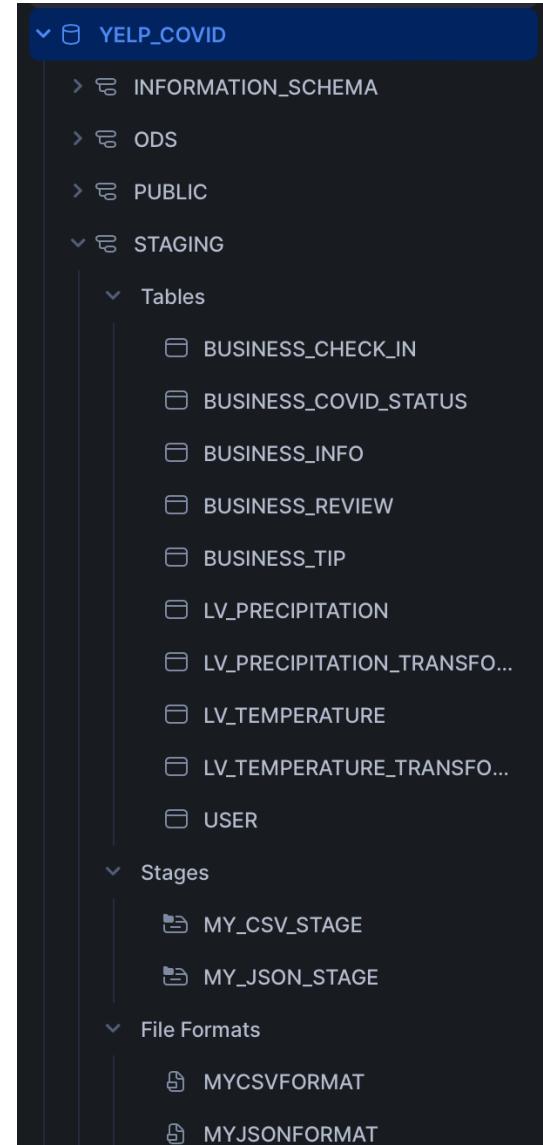
- Creating BUSINESS_TIP AND USER Tables and uploading their respective JSON files to the staging schema:

```
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "BUSINESS_TIP" FROM @my_json_stage/yelp_academic_dataset_tip.json.gz file_format=myjsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_tip.json.gz | LOADED | 908915 | 908915 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 12.979s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>CREATE TABLE "USER" ("USER_JSON" VARIANT);
+-----+
| status |
+-----+
| Table USER successfully created. |
+-----+
1 Row(s) produced. Time Elapsed: 1.212s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>PUT file:///Users/pierremora/Desktop/Data_engineering/yelp_dataset/yelp_academic_dataset_user_01.json @my_json_stage AUTO_COMPRESS= TRUE parallel=9;
+-----+-----+-----+-----+-----+-----+-----+
| source | target | source_size | target_size | source_compression | target_compression | status | message |
+-----+-----+-----+-----+-----+-----+-----+-----+
| yelp_academic_dataset_user_01.json | yelp_academic_dataset_user_01.json.gz | 840838797 | 517441376 | NONE | GZIP | UPLOADED | |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 230.400s
PMORAER#COMPUTE_WH@YELP_COVID.STAGING>COPY INTO "USER" FROM @my_json_stage/yelp_academic_dataset_user_01.json.gz file_format=myjsonformat on_error='skip_file';
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| file | status | rows_parsed | rows_loaded | error_limit | errors_seen | first_error | first_error_line | first_error_character | first_error_column_name |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| my_json_stage/yelp_academic_dataset_user_01.json.gz | LOADED | 401784 | 401784 | 1 | 0 | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 Row(s) produced. Time Elapsed: 33.144s
```

- This slide demonstrates the creation of the staging environment in Snowflake and the upload process for Yelp and climate data, including handling large JSON files by splitting them into smaller chunks.

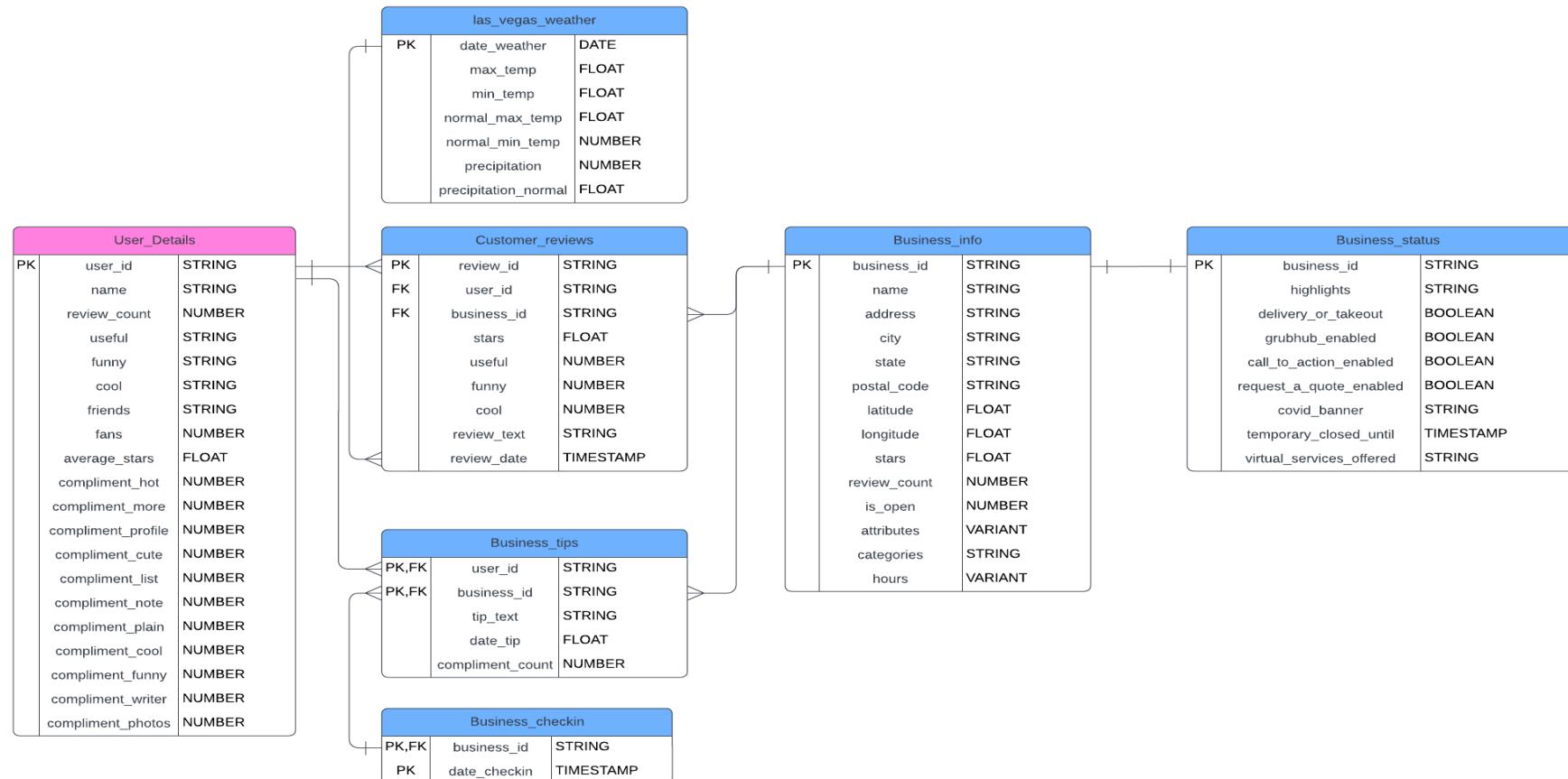
STAGING SCHEMA IN SNOWFLAKE

- Overview of the Created Staging Schema and tables
- This screenshot displays the Snowflake UI interface with the created staging schema and its respective tables, showcasing the initial setup before data transformation.



ENTITY-RELATIONSHIP (ER) DIAGRAM

- This ER diagram visualizes the relationships between different entities in the data structure, providing a clear overview of how Yelp and climate data are interconnected.



TRANSFORMING DATA FROM STAGING TO ODS

- This slide demonstrates the SQL queries used to transform data from staging to ODS, including the use of JSON functions to extract and transform data. It also provides a comparison of data sizes in raw files, staging tables, and ODS tables

```
YELP_COVID.ODS ✓ Settings ▾
Code Versions Q
1 CREATE TABLE User_Details(
2   user_id STRING PRIMARY KEY,
3   name STRING,
4   review_count NUMBER,
5   yelping_since TIMESTAMP,
6   useful STRING,
7   funny STRING,
8   cool STRING,
9   friends STRING,
10  fans NUMBER,
11  average_stars FLOAT,
12  compliment_hot NUMBER,
13  compliment_more NUMBER,
14  compliment_profile NUMBER,
15  compliment_cute NUMBER,
16  compliment_list NUMBER,
17  compliment_note NUMBER,
18  compliment_plain NUMBER,
19  compliment_cool NUMBER,
20  compliment_funny NUMBER,
21  compliment_writer NUMBER,
22  compliment_photos NUMBER );
23
24
25
26
27 INSERT INTO ODS.User_Details (user_id, name, review_count, yelping_since, useful, funny, cool, friends, fans, average_stars, compliment_hot, compliment_more, compliment_profile, compliment_cute, compliment_list, compliment_note, compliment_plain, compliment_cool, compliment_funny, compliment_writer, compliment_photos)
28 SELECT
29   "USER_JSON":user_id::STRING,
30   "USER_JSON":name::STRING,
31   "USER_JSON":review_count::NUMBER,
32   "USER_JSON":yelping_since::TIMESTAMP,
33   "USER_JSON":useful::NUMBER,
34   "USER_JSON":funny::NUMBER,
35   "USER_JSON":cool::NUMBER,
36   "USER_JSON":friends::STRING,
37   "USER_JSON":fans::NUMBER,
38   "USER_JSON":average_stars::FLOAT,
39   "USER_JSON":compliment_hot::NUMBER,
40   "USER_JSON":compliment_more::NUMBER,
41   "USER_JSON":compliment_profile::NUMBER,
42   "USER_JSON":compliment_cute::NUMBER,
43   "USER_JSON":compliment_list::NUMBER,
44   "USER_JSON":compliment_note::NUMBER,
45   "USER_JSON":compliment_plain::NUMBER,
46   "USER_JSON":compliment_cool::NUMBER,
47   "USER_JSON":compliment_funny::NUMBER,
48   "USER_JSON":compliment_writer::NUMBER,
49   "USER_JSON":compliment_photos::NUMBER
50 FROM STAGING.User;
51
```

```
YELP_COVID.ODS ✓ Settings ▾
Code Versions Q
...
54 CREATE TABLE Business_info(
55   business_id STRING PRIMARY KEY,
56   name STRING,
57   address STRING,
58   city STRING,
59   state STRING,
60   postal_code STRING,
61   latitude FLOAT,
62   longitude FLOAT,
63   stars FLOAT,
64   review_count NUMBER,
65   is_open NUMBER,
66   attributes VARIANT,
67   categories STRING,
68   hours VARIANT
69 );
70
71
72 INSERT INTO ODS.Business_Info (
73   business_id,
74   name,
75   address,
76   city,
77   state,
78   postal_code,
79   latitude,
80   longitude,
81   stars,
82   review_count,
83   is_open,
84   attributes,
85   categories,
86   hours
87 )
88 SELECT
89   "BUSINESS_INFO_JSON":business_id::STRING,
90   "BUSINESS_INFO_JSON":name::STRING,
91   "BUSINESS_INFO_JSON":address::STRING,
92   "BUSINESS_INFO_JSON":city::STRING,
93   "BUSINESS_INFO_JSON":state::STRING,
94   "BUSINESS_INFO_JSON":postal_code::STRING,
95   "BUSINESS_INFO_JSON":latitude::FLOAT,
96   "BUSINESS_INFO_JSON":longitude::FLOAT,
97   "BUSINESS_INFO_JSON":stars::FLOAT,
98   "BUSINESS_INFO_JSON":review_count::NUMBER,
99   "BUSINESS_INFO_JSON":is_open::NUMBER,
100  "BUSINESS_INFO_JSON":attributes::VARIANT,
101  "BUSINESS_INFO_JSON":categories::STRING,
102  "BUSINESS_INFO_JSON":hours::VARIANT
103 FROM STAGING.BUSINESS_INFO;
104
105
```

TRANSFORMING DATA FROM STAGING TO ODS

```
YELP_COVID.ODS ✓ Settings ▾
Code Versions Q
106 CREATE TABLE customer_reviews(
107     review_id STRING PRIMARY KEY,
108     user_id STRING,
109     business_id STRING,
110     stars FLOAT,
111     useful NUMBER,
112     funny NUMBER,
113     cool NUMBER,
114     review_text STRING,
115     review_date TIMESTAMP,
116     FOREIGN KEY (user_id) REFERENCES ODS.User_Details(user_id),
117     FOREIGN KEY (business_id) REFERENCES ODS.Business_info(business_id)
118 );
119
120 INSERT INTO ODS.customer_reviews (
121     review_id,
122     user_id,
123     business_id,
124     stars,
125     useful,
126     funny,
127     cool,
128     review_text,
129     review_date
130 )
131 SELECT
132     "BUSINESS REVIEW_JSON".review_id:STRING,
133     "BUSINESS REVIEW_JSON".user_id:STRING,
134     "BUSINESS REVIEW_JSON".business_id:STRING,
135     "BUSINESS REVIEW_JSON".stars:FLOAT,
136     "BUSINESS REVIEW_JSON".useful:NUMBER,
137     "BUSINESS REVIEW_JSON".funny:NUMBER,
138     "BUSINESS REVIEW_JSON".cool:NUMBER,
139     "BUSINESS REVIEW_JSON".text:STRING AS review_text,
140     "BUSINESS REVIEW_JSON".date:TIMESTAMP AS review_date
141 FROM STAGING.BUSINESS_REVIEW;
142
143
144 CREATE TABLE Business_tips(
145     user_id STRING,
146     business_id STRING,
147     tip_text STRING,
148     date_tip TIMESTAMP,
149     compliment_count NUMBER,
150     PRIMARY KEY (user_id,business_id),
151     FOREIGN KEY (user_id) REFERENCES ODS.User_Details(user_id),
152     FOREIGN KEY (business_id) REFERENCES ODS.Business_info(business_id)
153 );
154
```

```
YELP_COVID.ODS ✓ Settings ▾
Code Versions Q
154
155     INSERT INTO Business_tips(
156         user_id,
157         business_id,
158         tip_text,
159         date_tip,
160         compliment_count)
161     SELECT "BUSINESS_TIP_JSON".user_id:STRING,
162         "BUSINESS_TIP_JSON".business_id:STRING,
163         "BUSINESS_TIP_JSON".text:STRING AS tip_text,
164         "BUSINESS_TIP_JSON".date:TIMESTAMP AS date_tip,
165         "BUSINESS_TIP_JSON".compliment_count:NUMBER
166     FROM STAGING.BUSINESS_TIP;
167
168
169     CREATE TABLE Business_checkin(
170         business_id STRING,
171         date_checkin TIMESTAMP,
172         PRIMARY KEY (business_id,date_checkin),
173         FOREIGN KEY (business_id) REFERENCES ODS.Business_info(business_id)
174     );
175
176     INSERT INTO ODS.Business_checkin (
177         business_id,
178         date_checkin
179     )
180     SELECT
181         BUSINESS_CHECK_IN_JSON.business_id:STRING AS business_id,
182         TRY_TO_TIMESTAMP(trim(value)) AS date_checkin
183     FROM STAGING.BUSINESS_CHECK_IN,
184     LATERAL FLATTEN(INPUT => SPLIT(BUSINESS_CHECK_IN_JSON.date, ',')) AS date_checkin_data;
185
186
187
188     CREATE TABLE business_status (
189         business_id STRING PRIMARY KEY,
190         highlights STRING,
191         delivery_or_takeout BOOLEAN,
192         grubhub_enabled BOOLEAN,
193         call_to_action_enabled BOOLEAN,
194         request_a_quote_enabled BOOLEAN,
195         covid_banner STRING,
196         temporary_closed_until TIMESTAMP,
197         virtual_services_offered STRING,
198         FOREIGN KEY (business_id) REFERENCES ODS.Business_info(business_id)
199     );
200
```

TRANSFORMING DATA FROM STAGING TO ODS

```
YELP_COVID.ODS ▾ Settings ▾ Code Versions ▾

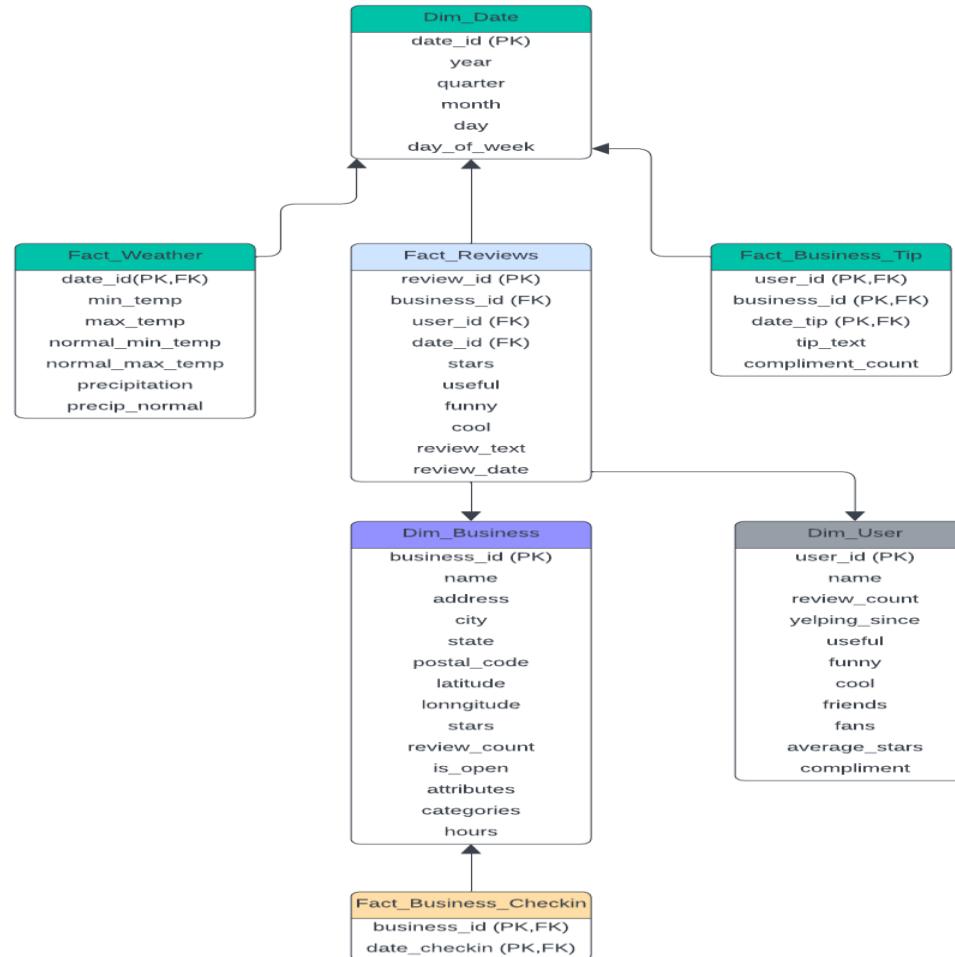
200
201
202 INSERT INTO business_status (
203     business_id,
204     highlights,
205     delivery_or_takeout,
206     grubhub_enabled,
207     call_to_action_enabled,
208     request_a_quote_enabled,
209     covid_banner,
210     temporary_closed_until,
211     virtual_services_offered
212 )
213 SELECT
214     BUSINESS_JSON:business_id::STRING,
215     IFF(BUSINESS_JSON:highlights::STRING = 'FALSE', NULL, BUSINESS_JSON:highlights::STRING) AS highlights,
216     BUSINESS_JSON:delivery_or_takeout::BOOLEAN AS delivery_or_takeout,
217     BUSINESS_JSON:grubhub_enabled::BOOLEAN AS grubhub_enabled,
218     BUSINESS_JSON:Call To Action Enabled::BOOLEAN AS call_to_action_enabled,
219     BUSINESS_JSON:Request a Quote Enabled::BOOLEAN AS request_a_quote_enabled,
220     IFF(BUSINESS_JSON:"Covid Banner":::STRING = 'FALSE', NULL, BUSINESS_JSON:"Covid Banner":::STRING) AS covid_banner,
221     IFF(BUSINESS_JSON:"Temporary Closed Until":::STRING = 'FALSE', NULL, TRY_TO_TIMESTAMP(BUSINESS_JSON:"Temporary Closed Until":::STRING)) AS temporary_closed_until,
222     BUSINESS_JSON:"Virtual Services Offered":::STRING AS virtual_services_offered
223 FROM STAGING.business_covid_status;
224
225
226 CREATE TABLE lv_weather(
227     date_weather DATE,
228     min_temp FLOAT,
229     max_temp FLOAT,
230     normal_min_temp FLOAT,
231     normal_max_temp FLOAT,
232     precipitation NUMBER(38,2),
233     precipitation_normal FLOAT
234 )
235
236
237 INSERT INTO ODS.lv_weather(
238     date_weather,
239     min_temp,
240     max_temp,
241     normal_min_temp,
242     normal_max_temp,
243     precipitation,
244     precipitation_normal)
245     SELECT t.DATE,
246             t.MIN_TEMP,
247             t.MAX_TEMP,
248             t.NORMAL_MIN_TEMP,
249             t.NORMAL_MAX_TEMP,
250             p.PRECIPITATION,
251             p.PRECIPITATION_NORMAL
252     FROM STAGING.LV_TEMPERATURE_TRANSFORMED AS t
253     INNER JOIN STAGING.LV_PRECIPITATION_TRANSFORMED AS p
254     ON t.DATE = p.DATE;
255
256
257 ALTER TABLE lv_weather
258 ADD CONSTRAINT pk_lv_weather PRIMARY KEY (date_weather);
259
```

COMPARISON OF DATA SIZES IN RAW FILES, STAGING TABLES, AND ODS TABLES

SCHEMA_NAME	TABLE_NAME	ROW_COUNT	SIZE_MB	Query Details
1 ODS	BUSINESS_CHECKIN	26713750	192.7	...
2 ODS	BUSINESS_INFO	150346	11.5	Query duration 1.7s
3 ODS	BUSINESS_STATUS	209795	7.0	Rows 28
4 ODS	BUSINESS_TIPS	908915	45.7	Query ID 01b5c5f9-0305-f106-0...
5 ODS	CUSTOMER_REVIEWS	7691076	2153.3	Show more ▾
6 ODS	LV_WEATHER	28241	0.2	SCHEMA_NAME ▾
7 ODS	ODS_TABLE_SIZES	null	null	ODS 10
8 ODS	RAW_FILE_SIZES	8	0.0	Staging 10
9 ODS	STAGING_TABLE_SIZES	null	null	Raw 8
10 ODS	USER_DETAILS	1987897	1891.6	TABLE_NAME ▾
11 Raw	usw00023169-las-vegas-mccarran-intl-ap-precipitation-inch.csv	null	1.0	100% filled
12 Raw	usw00023169-temperature-degreef.csv	null	1.0	ROW_COUNT #
13 Raw	yelp_academic_dataset_business.json	null	118.0	26713750
14 Raw	yelp_academic_dataset_checkin.json	null	287.0	SIZE_MB #
15 Raw	yelp_academic_dataset_covid_status.json	null	65.0	5000.0
16 Raw	yelp_academic_dataset_review.json	null	5000.0	0
17 Raw	yelp_academic_dataset_tip.json	null	180.0	5000
18 Raw	yelp_academic_dataset_user.json	null	3400.0	
19 Staging	BUSINESS_CHECK_IN	131930	80.5	
20 Staging	BUSINESS_COVID_STATUS	209795	5.0	
21 Staging	BUSINESS_INFO	150346	11.0	
22 Staging	BUSINESS_REVIEW	7691076	2156.2	
23 Staging	BUSINESS_TIP	908915	46.1	
24 Staging	LV_PRECIPITATION	28241	0.1	
25 Staging	LV_PRECIPITATION_TRANSFORMED	28241	0.1	
26 Staging	LV_TEMPERATURE	28241	0.3	
27 Staging	LV_TEMPERATURE_TRANSFORMED	28241	0.2	
28 Staging	USER	1987897	1892.8	

DATA WAREHOUSE STAR SCHEMA

- This slide illustrates the star schema of the data warehouse, showcasing the relationships between dimension and fact tables for efficient querying and analysis.



DATA WAREHOUSE STAR SCHEMA

Explanation of Why We Have Multiple Fact Tables:

Having multiple fact tables in a star schema design is crucial for several reasons:

1. Different Types of Analysis:

- Fact_Reviews: Focuses on the review data, capturing details about user reviews, ratings, and associated business and user information
- Fact_Weather: Captures weather data, which is essential for analyzing the impact of weather conditions on business reviews.
- Fact_Business_Tip: Stores business tip data, which provides insights into additional user feedback separate from formal reviews.
- Fact_Business_Checkin: Tracks business check-ins, offering another dimension of user interaction with businesses.

2. Separation of Concerns:

- Each fact table represents a distinct type of business process or event. By separating these into different fact tables, we maintain clarity and focus for each type of analysis.

3. Performance Optimization:

- Smaller, specialized fact tables can improve query performance because each table is optimized for specific types of queries and analysis.

4. Easier Maintenance and Scalability:

- Managing and scaling the data warehouse is easier when fact tables are organized by distinct business processes, allowing for more straightforward updates and maintenance.

SQL QUERIES FOR DATA MOVEMENT (ODS TO DWH)

```
YELP_COVID.WAREHOUSE ▾ Settings ▾ Code Versions Q
1 CREATE TABLE Dim_User(
2     user_id STRING PRIMARY KEY,
3     name STRING,
4     review_count NUMBER,
5     yelping_since DATE,
6     useful STRING,
7     funny STRING,
8     cool STRING,
9     friends STRING,
10    fans NUMBER,
11    average_stars FLOAT,
12    compliment_hot NUMBER,
13    compliment_more NUMBER,
14    compliment_profile NUMBER,
15    compliment_cute NUMBER,
16    compliment_list NUMBER,
17    compliment_note NUMBER,
18    compliment_plain NUMBER,
19    compliment_cool NUMBER,
20    compliment_funny NUMBER,
21    compliment_writer NUMBER,
22    compliment_photos NUMBER );
23
24 INSERT INTO Dim_User (
25     user_id,
26     name,
27     review_count,
28     yelping_since,
29     useful,
30     funny,
31     cool,
32     friends,
33     fans,
34     average_stars,
35     compliment_hot,
36     compliment_more,
37     compliment_profile,
38     compliment_cute,
39     compliment_list,
40     compliment_note,
41     compliment_plain,
42     compliment_cool,
43     compliment_funny,
44     compliment_writer,
```

```
YELP_COVID.WAREHOUSE ▾ Settings ▾ Code Versions Q
45     compliment_photos
46 )
47 SELECT
48     user_id,
49     name,
50     review_count,
51     CAST(yelping_since AS DATE) AS yelping_since,
52     useful,
53     funny,
54     cool,
55     friends,
56     fans,
57     average_stars,
58     compliment_hot,
59     compliment_more,
60     compliment_profile,
61     compliment_cute,
62     compliment_list,
63     compliment_note,
64     compliment_plain,
65     compliment_cool,
66     compliment_funny,
67     compliment_writer,
68     compliment_photos
69 FROM
70     ODS.User_Details;
71
72
73
74 CREATE TABLE Dim_Business (
75     business_id STRING PRIMARY KEY,
76     name STRING,
77     address STRING,
78     city STRING,
79     state STRING,
80     postal_code STRING,
81     latitude FLOAT,
82     longitude FLOAT,
83     stars FLOAT,
84     review_count NUMBER,
85     is_open NUMBER,
86     attributes VARIANT,
87     categories STRING,
88     hours VARIANT);
```

SQL QUERIES FOR DATA MOVEMENT (ODS TO DWH)

YELP_COVID.WAREHOUSE ▾ Settings ▾

Code Versions Q

```
89
90     INSERT INTO Dim_Business(
91         business_id,
92         name,
93         address,
94         city,
95         state,
96         postal_code,
97         latitude,
98         longitude,
99         stars,
100        review_count,
101        is_open,
102        attributes,
103        categories,
104        hours)
105    SELECT business_id,
106        name,
107        address,
108        city,
109        state,
110        postal_code,
111        latitude,
112        longitude,
113        stars,
114        review_count,
115        is_open,
116        attributes,
117        categories,
118        hours
119    FROM ODS.BUSINESS_INFO
120    WHERE categories ILIKE '%restaurant%';
121
122
123
```

YELP_COVID.WAREHOUSE ▾ Settings ▾

Code Versions Q

```
124     CREATE TABLE Dim_Date (
125         date_id DATE PRIMARY KEY,
126         year NUMBER,
127         month NUMBER,
128         day NUMBER,
129         quarter NUMBER,
130         week NUMBER,
131         weekday NUMBER,
132         day_name STRING,
133         month_name STRING,
134         is_weekend BOOLEAN
135     );
136
137     -- Generate and insert date range into Dim_Date
138     INSERT INTO Dim_Date (date_id, year, month, day, quarter, week, weekday, day_name, month_name, is_weekend)
139     WITH RECURSIVE date_sequence AS (
140         SELECT TO_DATE('1970-01-01') AS date_id
141         UNION ALL
142         SELECT DATEADD(DAY, 1, date_id)
143         FROM date_sequence
144         WHERE date_id < TO_DATE('2024-12-31')
145     )
146     SELECT
147         date_id,
148         EXTRACT(YEAR FROM date_id) AS year,
149         EXTRACT(MONTH FROM date_id) AS month,
150         EXTRACT(DAY FROM date_id) AS day,
151         EXTRACT(QUARTER FROM date_id) AS quarter,
152         EXTRACT(WEEK FROM date_id) AS week,
153         EXTRACT(DAYOFWEEK FROM date_id) AS weekday,
154         TO_CHAR(date_id, 'Day') AS day_name,
155         TO_CHAR(date_id, 'Month') AS month_name,
156         CASE WHEN EXTRACT(DAYOFWEEK FROM date_id) IN (6, 7) THEN TRUE ELSE FALSE END AS is_weekend
157     FROM
158     date_sequence;
159
```

SQL QUERIES FOR DATA MOVEMENT (ODS TO DWH)

```
YELP_COVID.WAREHOUSE ▾ Settings ▾ Code Versions ▾ Q

160 CREATE TABLE Fact_Reviews (
161   review_id STRING PRIMARY KEY,
162   user_id STRING,
163   business_id STRING,
164   date_id DATE,
165   stars FLOAT,
166   useful NUMBER,
167   funny NUMBER,
168   cool NUMBER,
169   review_text STRING,
170   FOREIGN KEY (user_id) REFERENCES Dim_User(user_id),
171   FOREIGN KEY (business_id) REFERENCES Dim_Business(business_id),
172   FOREIGN KEY (date_id) REFERENCES Dim_Date(date_id),
173   );
174
175 INSERT INTO Fact_Reviews (
176   review_id,
177   user_id,
178   business_id,
179   date_id,
180   stars,
181   useful,
182   funny,
183   cool,
184   review_text
185   )
186   SELECT
187     r.review_id,
188     u.user_id,
189     b.business_id,
190     d.date_id,
191     r.stars,
192     r.useful,
193     r.funny,
194     r.cool,
195     r.review_text
196   FROM
197     ODS.Customer_reviews r
198   JOIN
199     Dim_User u ON r.user_id = u.user_id
200   JOIN
201     Dim_Business b ON r.business_id = b.business_id
202   JOIN
203     Dim_Date d ON r.review_date::DATE = d.date_id;
204
205
206 CREATE TABLE Fact_Weather (
207   date_id DATE PRIMARY KEY,
208   min_temp FLOAT,
209   max_temp FLOAT,
210   normal_min_temp FLOAT,
211   normal_max_temp FLOAT,
212   precipitation FLOAT,
213   precipitation_normal FLOAT,
214   );
215   FOREIGN KEY (date_id) REFERENCES Dim_Date(date_id);
216   );
```

```
YELP_COVID.WAREHOUSE ▾ Settings ▾ Code Versions ▾ Q

217
218   INSERT INTO Fact_Weather (
219     date_id,
220     min_temp,
221     max_temp,
222     normal_min_temp,
223     normal_max_temp,
224     precipitation,
225     precipitation_normal
226     )
227   SELECT
228     date_weather,
229     min_temp,
230     max_temp,
231     normal_min_temp,
232     normal_max_temp,
233     precipitation,
234     precipitation_normal
235   FROM
236     YELP_COVID.ODS.LV.WEATHER;
237
238
239 CREATE TABLE Fact_Tip (
240   tip_id STRING PRIMARY KEY,
241   user_id STRING,
242   business_id STRING,
243   date_id DATE,
244   compliment_count NUMBER,
245   tip_text STRING,
246   FOREIGN KEY (user_id) REFERENCES Dim_User(user_id),
247   FOREIGN KEY (business_id) REFERENCES Dim_Business(business_id),
248   FOREIGN KEY (date_id) REFERENCES Dim_Date(date_id),
249   );
250
251
252 INSERT INTO Fact_Tip (
253   tip_id,
254   user_id,
255   business_id,
256   date_id,
257   compliment_count,
258   tip_text
259   );
260
261   SELECT
262     CONCAT("tip_", CAST(row_number() over (order by user_id) as varchar)) AS tip_id, -- Generate unique ID for tips
263     user_id,
264     business_id,
265     date_tip::DATE AS date_id,
266     compliment_count,
267     tip_text
268   FROM
269     YELP_COVID.ODS.BUSINESS_TIPS;
270
271
272
273 CREATE TABLE Fact_Checkin (
274   checkin_id STRING PRIMARY KEY,
275   business_id STRING,
276   date_id DATE,
277   FOREIGN KEY (business_id) REFERENCES Dim_Business(business_id),
278   FOREIGN KEY (date_id) REFERENCES Dim_Date(date_id)
279   );
280
281
282
283 INSERT INTO Fact_Checkin (
284   checkin_id,
285   business_id,
286   date_id
287   )
288   SELECT
289     CONCAT(business_id, '_', CAST(row_number() over (order by business_id) as varchar)) AS checkin_id, -- Generate a unique ID
290     business_id,
291     date_checkin::DATE AS date_id
292   FROM
293     YELP_COVID.ODS.BUSINESS_CHECKIN;
294
```

```
295
296
297
298
299
```

QUERYING THE DATA WAREHOUSE FOR INSIGHTS

```
297 WITH TemperatureRanges AS (
298     SELECT
299         r.date_id,
300         r.business_id,
301         r.stars,
302         CASE
303             WHEN w.min_temp <= 32 THEN 'Very Cold (<= 32°F)'
304             WHEN w.min_temp > 32 AND w.min_temp <= 50 THEN 'Cold (33-50°F)'
305             WHEN w.min_temp > 50 AND w.min_temp <= 70 THEN 'Moderate (51-70°F)'
306             WHEN w.min_temp > 70 AND w.min_temp <= 85 THEN 'Warm (71-85°F)'
307             ELSE 'Hot (> 85°F)'
308         END AS temp_range
309     FROM
310         Fact_Reviews r
311     JOIN
312         Fact_Weather w ON r.date_id = w.date_id
313 )
314     SELECT
315         b.name AS business_name,
316         tr.temp_range,
317         COUNT(*) AS number_of_reviews,
318         ROUND(AVG(tr.stars), 2) AS average_rating
319     FROM
320         TemperatureRanges tr
321     JOIN
322         Dim_Business b ON tr.business_id = b.business_id
323     GROUP BY
324         b.name,
325         tr.temp_range
326     ORDER BY
327         b.name,
328         tr.temp_range;
```

Results **Chart**

BUSINESS_NAME	TEMP_RANGE	NUMBER_OF_REVIEWS	AVERAGE_RATING
"Genuino" Italian Cafe'	Cold (33-50°F)	3	4.33
"Genuino" Italian Cafe'	Hot (> 85°F)	22	4.32
"Genuino" Italian Cafe'	Moderate (51-70°F)	21	4.48
"Genuino" Italian Cafe'	Warm (71-85°F)	23	4.35
#1 Mongolian BBQ - Best Stir Fried Noodles In Boise	Cold (33-50°F)	2	5
#1 Mongolian BBQ - Best Stir Fried Noodles In Boise	Hot (> 85°F)	57	3.81
#1 Mongolian BBQ - Best Stir Fried Noodles In Boise	Moderate (51-70°F)	30	3.87
#1 Mongolian BBQ - Best Stir Fried Noodles In Boise	Warm (71-85°F)	24	2.83
\$5 Fresh Burger Stop	Hot (> 85°F)	1	4
\$5 Fresh Burger Stop	Moderate (51-70°F)	3	5
\$5 Fresh Burger Stop	Warm (71-85°F)	1	5
&pizza - UPenn	Cold (33-50°F)	2	3
&pizza - UPenn	Hot (> 85°F)	27	4.04

Query Details

- Query duration 3.2s
- Rows 128.1K
- Query ID 01b5c649-0305-f101-0...

BUSINESS_NAME **A**
100% filled

TEMP_RANGE **A**
Hot (> 85°F) 36,478
Moderate (51-70°F) 36,478
Warm (71-85°F) 36,478

Ask Copilot

CONCLUSIONS AND ACKNOWLEDGEMENTS

1. Conclusions:

- **Summary of Data Architecture:**
- Successfully designed a robust data architecture for integrating Yelp and climate datasets using Snowflake.
- Created a well-structured data pipeline from raw data ingestion to a fully-functional data warehouse.
- Implemented staging, ODS, and DWH schemas to facilitate efficient data processing and analysis.
- Ensured data integrity and consistency through proper transformation and normalization techniques.

2. Project Achievements:

- Developed a comprehensive data architecture diagram to visualize the data flow and relationships between different layers.
- Utilized Snowflake's capabilities to manage large datasets and perform complex transformations.
- Designed and implemented a star schema for the data warehouse, optimizing for OLAP queries.
- Achieved a scalable and maintainable data warehouse structure that supports future data integration and analysis needs.

3. Next Steps:

- Further enhance the data warehouse by integrating additional data sources and refining existing transformations.
- Explore advanced data modeling techniques to improve query performance and data insights.
- Implement data governance and security measures to ensure the protection and privacy of the data.