FINAL PROJECT

Designing Advanced Data Architectures for Business Intelligence

Team 17

Akhil Rao
Divyesh Rajput
Nitant Jatale
Rutuja More

3 source tables
Motor Vehicle Collisions - Vehicles
Motor Vehicle Collisions - Person
Motor Vehicle Collisions - Crashes (using Google BigQuery)

Tools: Talend Studio, Alteryx, Microsoft SQL Server Management Studio

Reason for our approach

Pivot tool can be avoided and after having n number of brainstorming sessions, we decided to do that. As per our understanding, we think that the relationship between the bigquery table Crashes and the Vehicle.tsv file is very much similar to an order header and order line item relationship.
For eg: If a collision occurs between 3-4 vehicles, then in the crashes table we will have 3-4 columns associated with each of those vehicles. And in the vehicles.tsv file, we will have 3-4 rows for each of the vehicles in the collision instance.

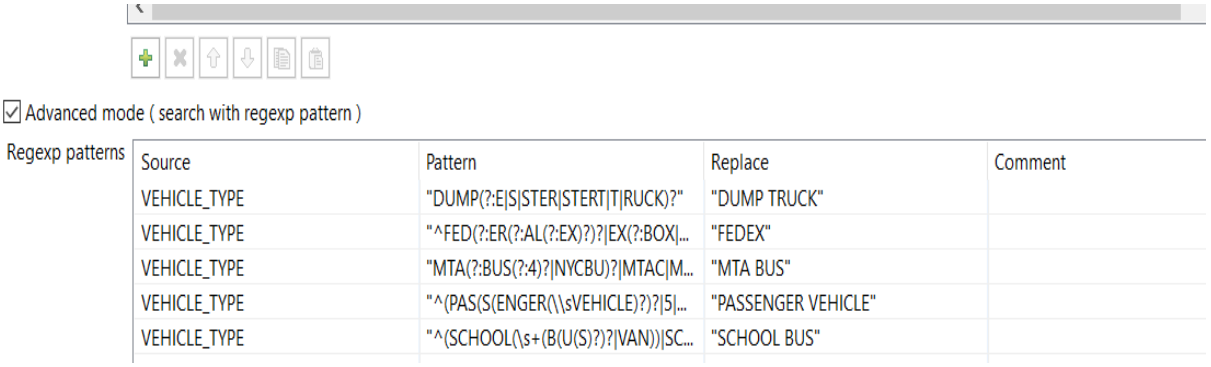
Person:

1. Crash_date - Date format (10/26/2019 and 10-6-2019)
2. Age - (Null, Negative values, values > 100)
3. Safety_Equipment - (7185 records with Safety_Equipment as -): Not Available
4. All other columns have Blank rows which we are replacing with ‘NO VALUE PROVIDED’
5. All columns uppercase
6. Empty Person_sex row replaced with ‘N’ as the size is 1byte

Crashes:
1. Date Parse - We have date column with data like 'mm/dd/yyyy', 'dd/mm/yyyy', 'mm-dd-yyyy', 'yyyy-mm-dd' Cleaned it and stored as yyyy-mm-dd
2. ZipCode columns contains null values replace nulls with -99999
3. Contributing factor columns has numeric value like 1 & 80 which we have replaced with 'Unspecified' string
4. String columns with nulls we replaced it with 'No Value Provided'

Vehicle:

Applied Regex and tReplace to remove a number of data discrepancies so that we get the right facts for each of the vehicle types.



☑ Advanced mode (search with regexp pattern)

Regexp patterns

Source	Pattern	Replace	Comment
VEHICLE_TYPE	"DUMP(?:E|S|STER|STERT|T|RUCK)?"	"DUMP TRUCK"	
VEHICLE_TYPE	"^FED(?:ER(?:AL(?:EX)?)?|EX(?:BOX|...	"FEDEX"	
VEHICLE_TYPE	"MTA(?:BUS(?:4)?|NYCBU)?|MTAC|M...	"MTA BUS"	
VEHICLE_TYPE	"^(PAS(S(ENGER(\\sVEHICLE)?)?|5|...	"PASSENGER VEHICLE"	
VEHICLE_TYPE	"^(SCHOOL(\s+(B(U(S)?)?|VAN))|SC...	"SCHOOL BUS"	

Omitted columns:

Based on our learning and expertise, we have mutually decided to omit certain columns and keep them as it is in the staging table. For the visualizations and the storytelling purposes, we have taken all the columns necessary columns to do that.

Crashes:

1. Zip
2. Off_street_name
3. Cross_street_name
4. Lat, long
5. Location
6. Contributing_factor_1
7. Contributing_factor_2
8. Contributing_factor_3
9. Contributing_factor_4
10. Contributing_factor_5

Vehicle

1. Vehicle damage_1
2. Vehicle damage_2
3. Vehicle damage_3
4. Vehicle make
5. Vehicle model
6. Travel direction
7. Pre_crash
8. Point_of_impact
9. public_property_damage_type

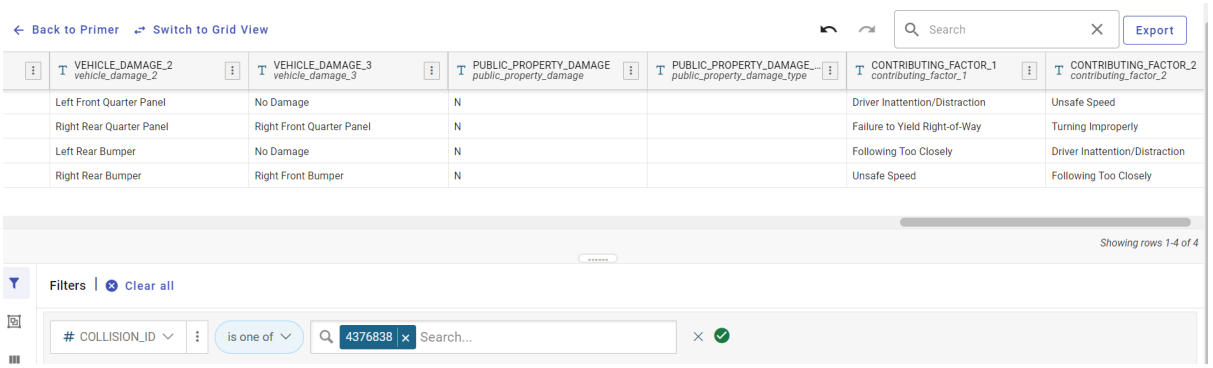
Person

1. Position_in_vehicle
2. Complaint
3. Safety equipments
4. Ped_location
5. Bodily_injury
6. Ejection
7. ped_action

Person's emotional status cannot be directly included in the fact table because emotional status is what the person is feeling at that moment. It is bound to change. It is not an inherited property by the person. And for this reason, we have made a different dimension table to store the emotional status of a person.

In crashes big query source tables, contributing factor columns are storing contributing factors of each vehicle involved in the accident. So instead of using pivot, we are fetching the rows directly from vehicles.tsv source table as it contains the values in row format.
For eg: search collision_id = 4376838 (vehicles, crashes)

So, we have decided to omit contributing factors 2,3,4, and 5 because either ways, we are storing unique values in the Dim_Contributing_Factor..



We are associating driver's license status with Fact_Vehicle and not with Fact_Person, because if we store driver_license_status with a person, we will need to create a new column. The Person table has different Person_type such as pedestrians, bicyclists etc. If we store driver_license_status with a person, we will have to add unwanted values for pedestrians, bicyclists. Instead we decided to associate it with the Fact_vehicle where every vehicle has a driver who has to have a drivers_license.

For every collision_ID from Fact_vehicle and Fact_Person, there is a record present in the Fact_Crashes. In Fact_vehicle, a vehicle can be involved in more than one collision over time, and in a collision there can be more than one person involved. That's why we are joining the fact tables directly with Collision_SK as a one to many relationship.

Contributing factors is a dimension which has distinct values from both Vehicles and Person.tsv files. So, it will have surrogate keys pointing to both the fact tables.