**Solution Design Document**

**Solution**

Following is the solution used to design the application,

Step 1: upload data in s3

Step 2: Create a connection with databricks.

Step3: Read the data from s3 to pyspark dataframes.

Step 4: Clean the data like remove the null, delete the duplicates.

Step 5: transform the dataframe to get the required output tables.

Step 6: Create the connection in redshift

Step 7: Write the output from df to redshift

Use Cases - List down all the use cases on which this solution will be applicable.

1. **Which disease has a maximum number of claims.**

For this use case, I can use the claims data. Hence, I read the data from claims to a data frame first. Now I grouped the data with disease and got the count of each claim id. So, this gave me the number for claims made of each disease. Now by performing a sort on count in descending order I can get a disease with highest on the top and I can write that to the redshift as the result.

1. **Find those Subscribers having age less than 30 and they subscribe any subgroup**

For this use case I will use the subscriber table. Once I get the data frame from S3 I will filter the data frame based on two conditions. The first one is we have a date of birth column so I can use that to find the current is of this subscriber and filter the ones that have the is less than 30. Along with that it should also satisfy another condition that is the subscriber ID column should not be null or empty. If both of these conditions are satisfied then only the data will be selected. Finally the resulting data frame will be returned to the redshift.

1. **Find out which group has maximum subgroups.** For this use case I will need two data frames from 2 tables from S3. And I will need to join these tables based on the group ID. Then I can group the data frame using group ID and then count the number of subgroup ID. Now if I sort it in descending order I will get the group ID with maximum number of subgroup on top which can be the result.

**Tables Metadata Info with Pk/FK relationship**

A screenshot of a computer

Description automatically generated

Technologies and Platforms to be used in this solution

1. Pyspark
2. AWS S3
3. Databricks
4. Redshift
5. Jira
6. GitHub