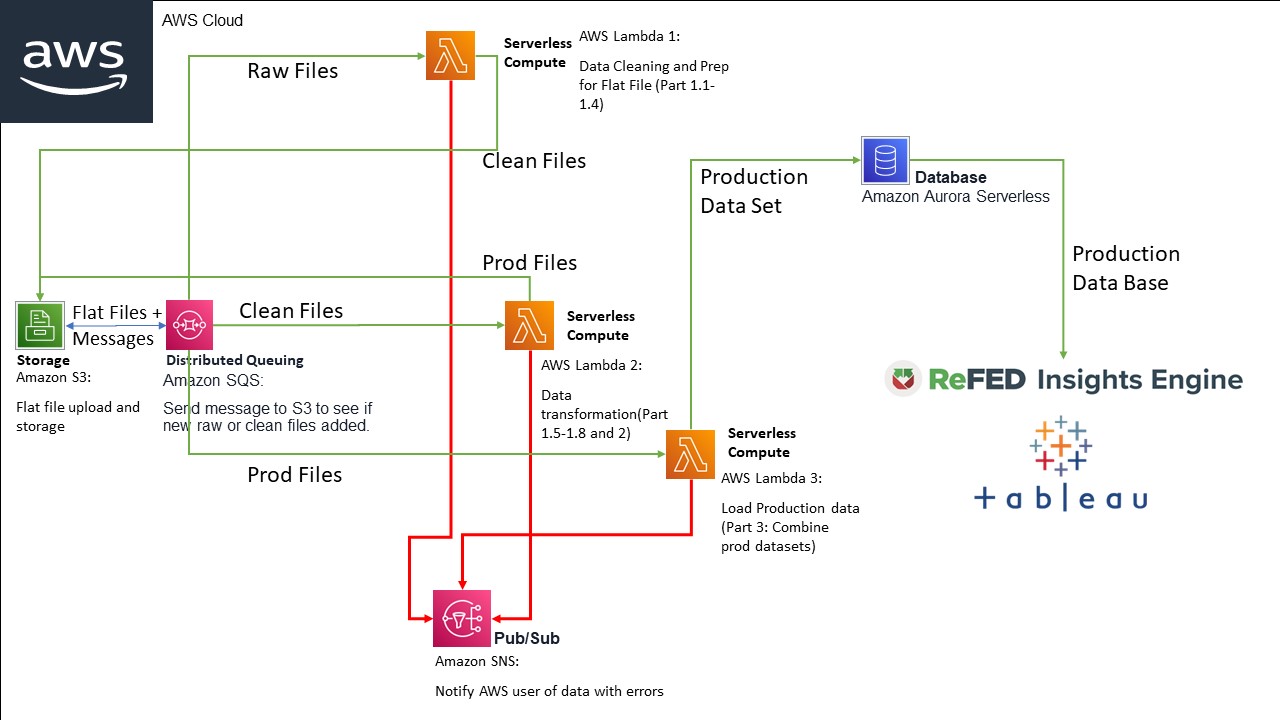
When developing an analytics pipeline for flat files to be cleaned, formatted, and transformed into a production database, the ETL stages must be developed and implemented with availability and scalability in mind. The below description is based on an AWS serverless architecture that allows for no provisioning and termination of resources by the user but by the system itself. This type of system will help greatly improve scalability without additional development/engineering, but would come with some potential price increases depending on usage.

Please refer to the figure below for an overview of how this serverless architecture could work.



Based on the prompt of datasets being provided as flat files, the Extraction stage for these files will not be part of this consideration. Instead the starting point for this analytics pipeline will be with many flat files saved in an AWS S3 bucket.

The Transform stage would be handled by various Lambda functions each based on the transformations from the challenge. The communication between S3 and Lambda will be an SQS distributed queuing service that can send triggers to activate the Lambda’s based on certain rules. The SQS as opposed to Pub/Sub or SNS as the delivery is guaranteed and allows for decoupling of storage S3 and Lambda whereas the SNS service is used for error message handling from the various Lambda functions.

The Lambda’s output the next stage of the transformation to S3 bucket to be routed again to the next Lambda as well as serve as a flat file redundant backup for the production data set.

Once all the Lambda’s complete their transformations, the production data set is loaded into Aurora Serverless, a mySQL database that has serverless capacities to be able to scale as data requires. This would be the last stage of the ETL for the Load function and where the ReFED Insights Engine would read the data from. Another internal alternative for data exploration could be a native integration from AWS Aurora to Tableau where the data can be sliced and diced for custom reports that might be too advanced for the everyday user of the Insights Engine.

Please also see the GitHub repo for the other part of the Code Challenge at:

<https://github.com/nimasajedi1/ReFED-Code-Challenge>