Batch Processing in Telecom Data Engineering

This document outlines the step-by-step approach for batch processing of telecom data in a data engineering project. The goal is to perform ETL processes on call records, customer usage data, billing, and other data to generate reports for business use cases.

# Tech Stack

1. Databricks: Used for data extraction, transformation, and storage in Delta Lake.  
2. Delta Lake: Storage layer for batch data with ACID properties.  
3. Power BI: Visualize and report batch data.  
4. Python: Used for batch ETL scripts.

# Step-by-Step Solution

## 1. Data Extraction

Use Databricks to extract telecom call records, billing data, and customer usage data. Data sources include network switches and billing systems. Extract data in batches and load it into Delta Lake.

## 2. Data Transformation

Perform ETL processes to clean, transform, and aggregate data. Use Python with Databricks to apply transformations such as filtering null values, joining tables, and calculating metrics like total call duration, data usage, and outstanding balances.

## 3. Storage in Delta Lake

Store the processed data in Delta Lake, ensuring that data is structured in a curated format. Use Delta Lake’s ACID properties to ensure data integrity. Organize data in a star schema with fact tables for usage, billing, and network metrics.

## 4. Integration with Power BI

Connect Power BI to the Delta Lake tables and create reports on key metrics. Use Power BI's capabilities to generate monthly billing reports and customer usage trends. Implement data refresh schedules to update the reports with the latest batch data.

## Use Case: Batch Ingestion of Telecom Data for Monthly Reports

This use case involves extracting call records, billing, and customer usage data in batch mode. The data is transformed and loaded into Delta Lake, and Power BI reports are generated to provide insights into monthly customer billing and usage trends.

# Batch Processing Architecture

The batch processing architecture for telecom data involves several key components that work together to process large volumes of data efficiently. Below is a breakdown of each component in the architecture:

## 1. Data Sources

Telecom call records, billing data, customer usage data, and network performance metrics are collected from various sources such as network switches, billing systems, and customer devices. These data sources feed into the batch processing pipeline on a scheduled basis.

## 2. Databricks (ETL Processing)

Databricks acts as the primary ETL platform for performing batch processing on the collected data. Data is extracted, cleaned, transformed, and aggregated using Databricks. Python or Scala scripts can be used to implement custom ETL processes. The batch jobs are scheduled to run periodically (e.g., daily or monthly).

## 3. Delta Lake (Storage)

The transformed data is stored in Delta Lake, which acts as the central storage layer. Delta Lake provides ACID transaction support, ensuring data consistency and reliability. It also enables efficient querying and versioning of batch data. Data is organized in a curated star schema format for easy analysis.

## 4. Power BI (Reporting)

Power BI connects to Delta Lake to generate reports and dashboards. Batch data is visualized in Power BI to provide business insights such as customer billing, usage trends, and network performance. Power BI can also be configured for scheduled refresh to ensure up-to-date reports.

## Batch Processing Workflow

1. Data is extracted from multiple sources (call records, billing data, usage data).  
2. The extracted data is transformed using Databricks ETL jobs (scheduled daily/monthly).  
3. Transformed data is loaded into Delta Lake (batch load).  
4. Power BI queries the batch data and generates scheduled reports.