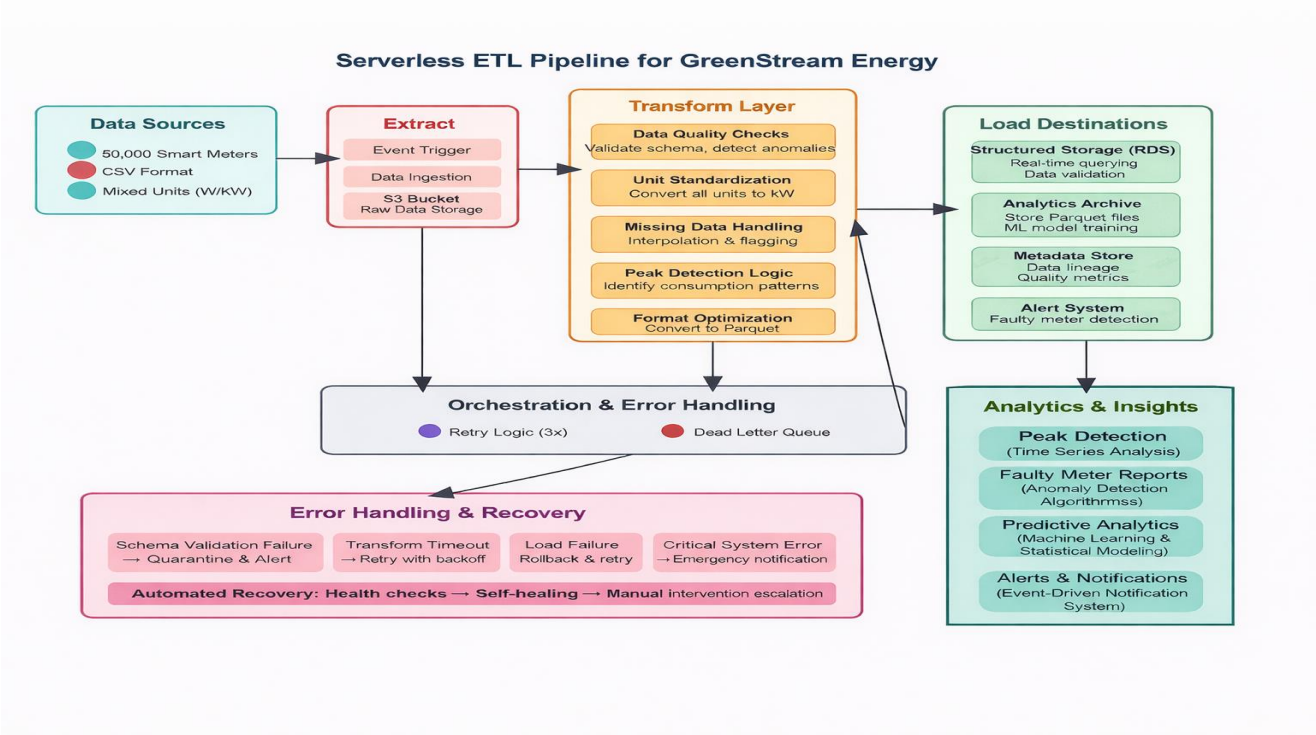
Name : Botayla Amin
ID : 412200017

Task A:



The serverless ETL pipeline starts with ingesting raw smart-meter data in CSV format into a raw storage layer to preserve the original records for traceability. The arrival of new data automatically triggers the orchestration layer, which controls the execution flow, manages dependencies, and handles retries in case of failures.

The transformation layer applies all data science logic, including unit standardization, missing value handling, data validation, and basic faulty meter detection. Invalid or suspicious records are routed to failure paths, while clean records continue through the pipeline.

Validated data is stored in a structured storage layer to support fast querying and data validation. In parallel, the same data is converted into analytics-optimized Parquet format and archived for long-term analysis. This analytics layer enables peak consumption detection, consumption pattern analysis, and prepares the data for future predictive analytics and forecasting.

Across all stages, automated error handling ensures reliability through retries, logging, and isolation of persistent failures.

Task B:

1- Unit Standardization Rules

- the energy unit is "W", divide the value by 1000 and convert the unit to "kW".
- If the energy unit is "kW", keep the value unchanged.
- If the unit is neither "W" nor "kW", flag the record as invalid and exclude it from analytical processing while logging it in the metadata store.

2- Missing Data Handling Rules

- If an energy reading is NULL, flag the record as missing.
- If the missing interval is short (e.g., a single timestamp), apply interpolation using neighboring values.
- If the missing interval is long, exclude the record from peak-usage calculations and predictive analytics.
- All interpolated values must be labeled as estimated in the metadata.

3. Data Validation Rules

- If the energy value is negative, mark the record as invalid and quarantine it.
- If the energy value exceeds a realistic household consumption threshold, flag it as an outlier.
- If timestamps are duplicated, missing, or outside the expected time range, reject the record.
- All validation failures are routed through the error-handling workflow and logged for auditability.

4. Faulty Smart Meter Detection Rules

- If a meter reports zero consumption for an unusually long continuous period, flag it as potentially faulty.
- If a meter reports the same constant value over extended intervals, mark it as suspicious.
- If sudden, extreme deviations occur compared to historical patterns, classify the reading as anomalous.
- Flagged meters are excluded from peak-usage calculations and forwarded to the alert system.

5. Data Format Optimization Rules

- Convert all validated and standardized data into columnar Parquet format.
- Partition data by time (e.g., date) and meter identifier to optimize querying.
- If conversion fails, retry automatically; unresolved failures are sent to a dead-letter queue.

Task C:

Step 1: Upload to Raw Storage

A smart meter uploads a single energy consumption record in CSV format to the raw data storage layer.At this stage, the data is stored exactly as received, without modification.

Step 2: Triggering the Transformation Process

The arrival of a new file in raw storage automatically triggers the transformation workflow. The orchestration layer monitors the event and initiates the data processing pipeline.

Step 3: Data Cleaning and Validation

During the transformation phase:

- The energy unit is checked and converted to kilowatts (kW) if necessary.
- Missing values are detected and either interpolated or flagged.
- The record is evaluated for abnormal patterns indicating a faulty meter.

Only validated records proceed further

Step 4: Storage in Structured Format (RDS)

Validated and cleaned records are stored in structured relational storage to support:

- Fast querying
- Data validation
- Operational reporting

This layer represents the “single source of truth” for clean operational data

Step 5: Conversion and Archival in Parquet Format

The same validated record is then converted into Parquet format and archived in the analytics storage layer.
This optimized format supports long-term historical analysis, machine learning, and forecasting

Step 6 :Analytics & Insights Consumption

The archived Parquet data is consumed by the **Analytics & Insights layer**, where it is used to identify peak energy consumption periods, analyze usage patterns, detect anomalies, and support future predictive analytics and forecasting

Step 7: Success and Failure Handling

- If all steps succeed, the record is marked as successfully processed.
- If a failure occurs:
    - Automatic retries are attempted (3 times).
    - Persistent failures are sent to a dead-letter queue(DLQ).
    - Critical issues trigger alerts for manual investigation.

This ensures pipeline reliability and fault tolerance