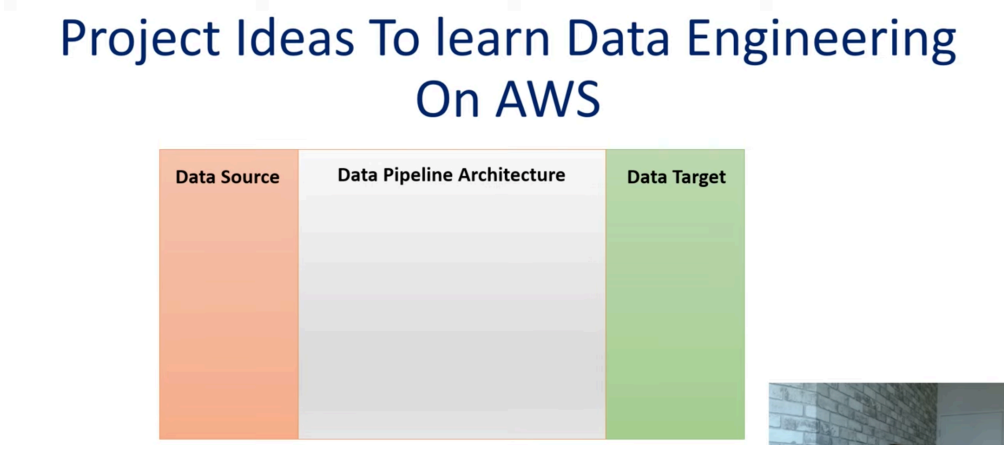
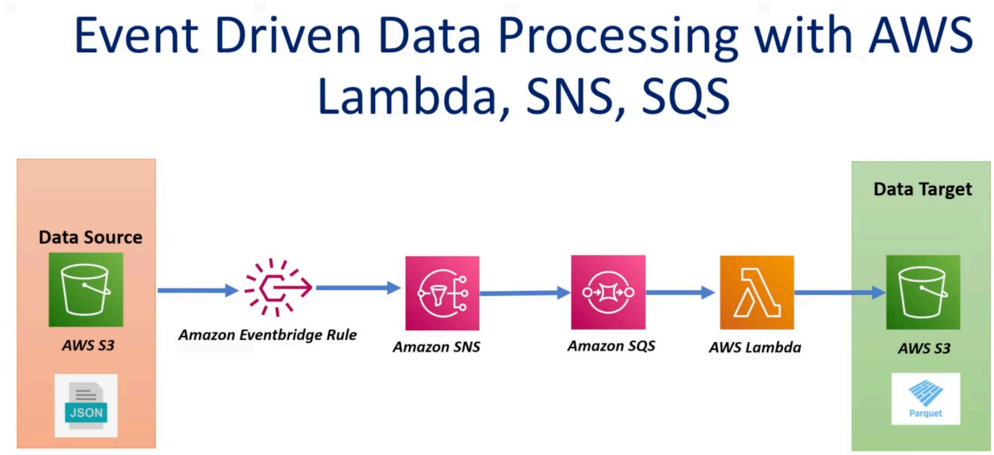
Practical Project Ideas for Data Engineering on AWS



- Learn data engineering on AWS to handle a variety of real-world challenges
- Emphasis on building skills beyond just batch data processing pipelines
- Focus on architecture patterns rather than specific datasets

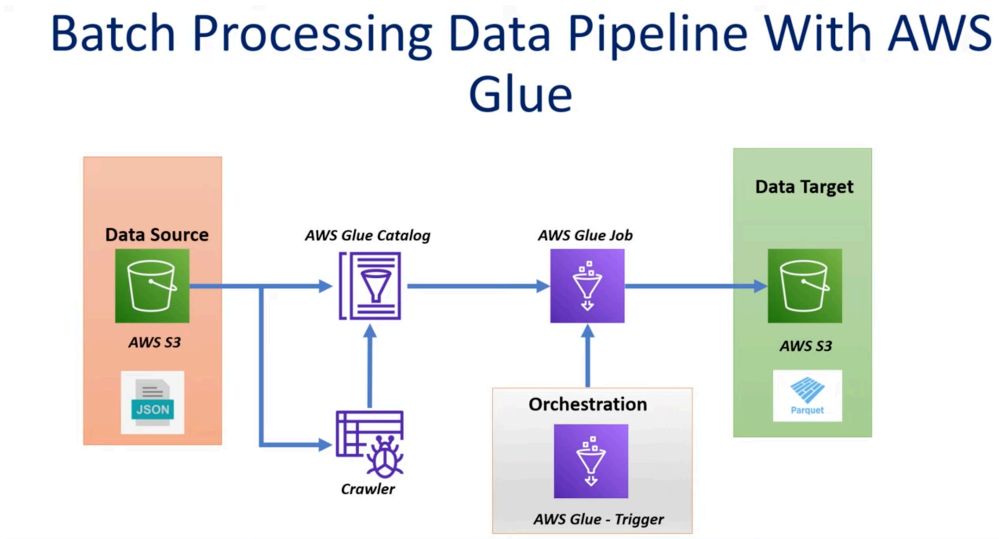
Event-Driven Data Pipeline



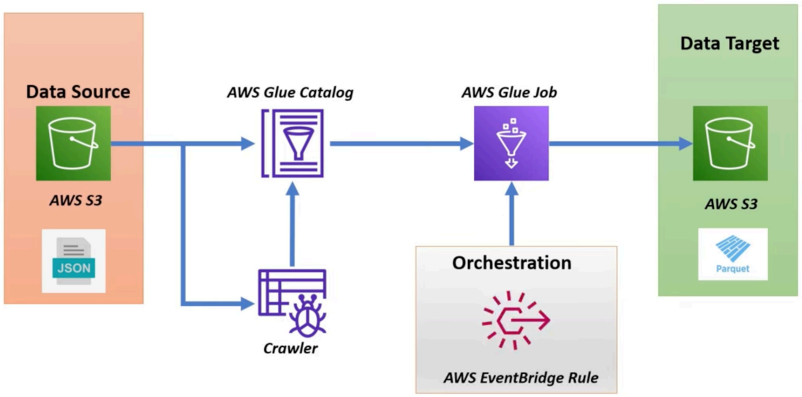
- **Objective**
  - Process RAW files uploaded into an AWS S3 bucket
  - Perform transformations before loading into a data target
- **Steps and Components**
  - **Event Trigger Creation**
    - Use Amazon EventBridge rule
  - **Message Queuing**
    - EventBridge rule connected to Amazon SNS topic
    - Amazon SNS topic connected to Amazon SQS queue
  - **Processing Logic**
    - SQS queue feeds AWS Lambda function

        - Lambda function performs data transformation
- **Benefits and Considerations**
    - **Benefits**
        - Completely serverless
        - No need to manage infrastructure
        - Scalable by design
    - **Considerations**
        - Lambda's concurrent execution limits
        - SQS queue helps manage file processing backlog
        - SNS topic allows for future subscriber additions
        - Dead letter queue for resilience
- **Example Implementation**
    - Transformed data written to another S3 bucket in Parquet format
    - Python used for data transformation
    - AWS Data Wrangler library for reading and writing data in Lambda functions

Batch Processing Data Pipeline with AWS Glue



Batch Processing Data Pipeline With AWS Glue



- **Objective**
  - Handle processing of big data files, potentially gigabytes in size
- **Steps and Components**
  - **Data Source Definition**
    - Data resides in AWS S3 bucket
    - Defined in AWS Glue catalog using a Glue Crawler
  - **AWS Glue Job Development**
    - Jobs written in Python or Scala
    - Perform necessary data transformations
  - **Data Loading**
    - Transformed data written to a target (e.g., RDS, Redshift, S3)
- **Orchestration**
  - Schedule or trigger AWS Glue jobs
  - Use AWS Glue's built-in orchestration or AWS EventBridge rule

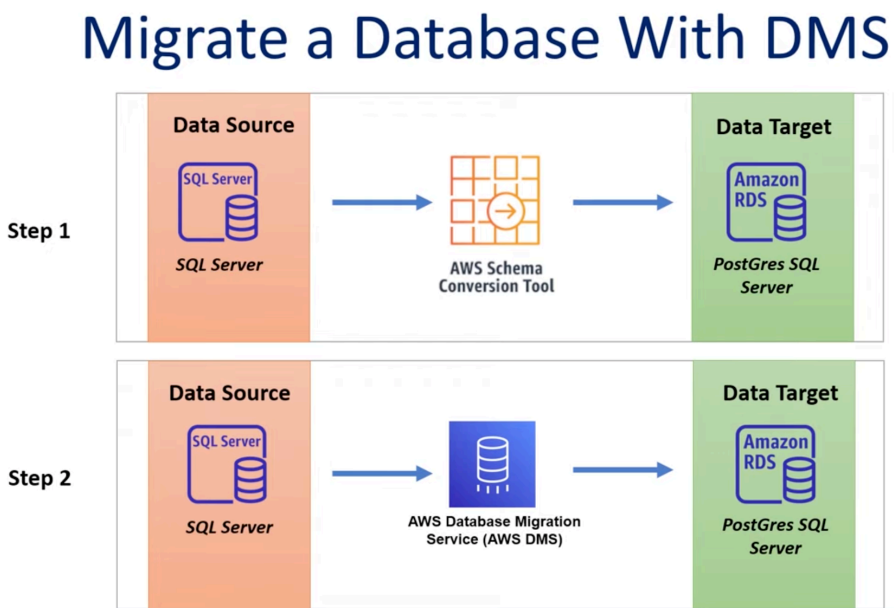
Building a Low-Cost Serverless Dashboard on AWS

Serverless Dashboard Powered By AWS Athena

- **Objective**
  - Provide data analytics to business users using transformed data

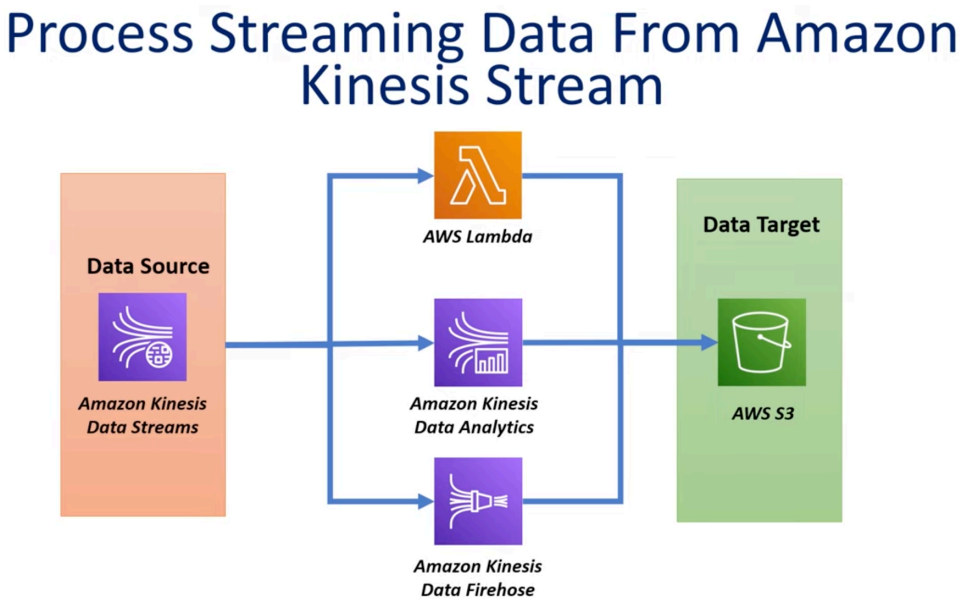
- **Steps and Components**
  - **Data Source**
    - Parquet files stored in AWS S3
    - Data already transformed and cleaned
  - **Data Definition**
    - Defined in AWS Glue catalog
  - **Query Engine**
    - Use Amazon Athena for serverless queries
  - **Visualization**
    - Build dashboards with Amazon QuickSight
    - Dashboards powered by Amazon Athena queries

Database Migration to AWS



- **Objective**
  - Migrate databases from on-premises or other cloud databases to AWS
- **Steps and Components**
  - **Source and Target Databases**
    - Example: Migrating from Microsoft SQL Server to PostgreSQL on AWS
  - **Schema Conversion**
    - Use AWS Schema Conversion Tool
    - Automatically convert schema and database objects
  - **Data Migration**
    - Use AWS Database Migration Service (DMS)
    - Perform one-time migration or ongoing replication

Real-Time Data Processing on AWS



- **Objective**
  - Perform analytics on real-time streaming data
- **Common Use Cases**
  - Website clickstreams, database event streams, financial transactions, social media feeds, location tracking events
- **Methods and Tools**
  - **AWS Lambda**:
    - Custom logic for stream processing
  - **Amazon Kinesis Data Analytics**:
    - Analyze streaming data using Apache Flink
  - **Amazon Kinesis Firehose**:
    - Transform raw streaming data into formats like Apache Parquet
    - Dynamically partition streaming data