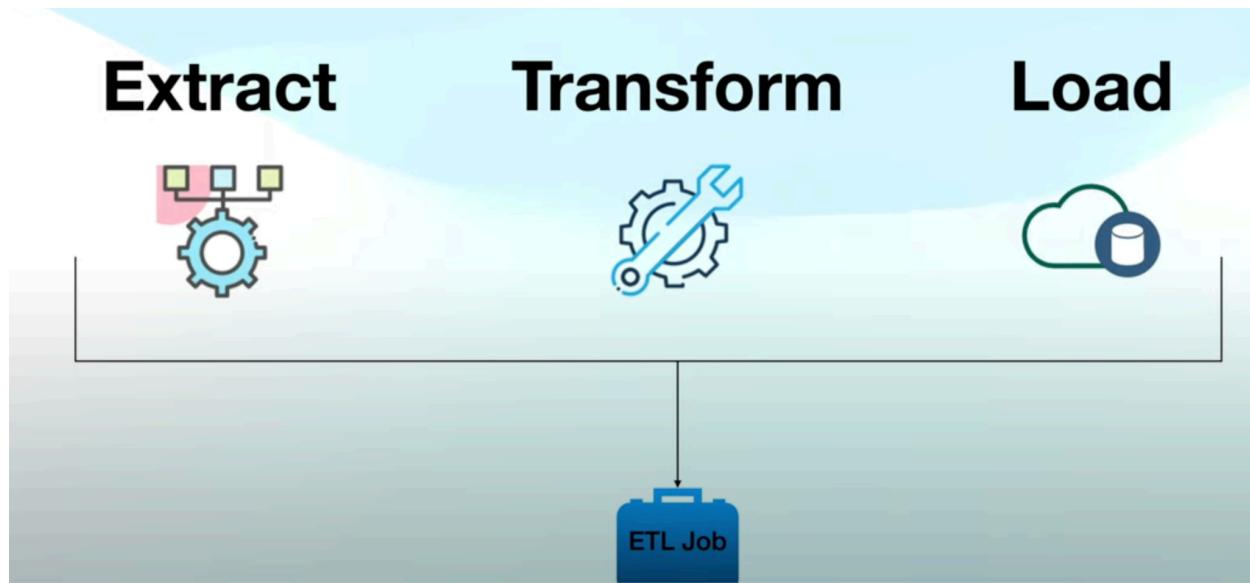


ETL JOB



Running ETL options

On-Prem ETL	Serverless ETL
Team has to take care of the infrastructure like servers, its maintenance, software updates, apply patches, etc	The infrastructure part is maintained by the service provider
Over allocation of resources for an ETL task results in idle resources + Before hand investment for servers/infra setup	Only pay for what you use. No need to own a setup/server
Aware of what is provisioned	Little or no knowledge of what is provisioned
Focus on infra, choosing the ETL tools, software for ETL job + building core data pipelines	Allows developers more time to focus on data preparation, building data pipelines
Code for ETL jobs is written from scratch	Generate the serverless code for the ETL job

Introduction to aws GLUE

What is AWS Glue?

- AWS Glue is the fully managed, serverless ETL tool
- Used to discover the data, transform it and prepare it for analytics
- AWS Glue provides all the capabilities for data integration via visual & code-based interfaces
- It consists of Central Metadata Repository known as AWS Glue Data Catalog
- Apache Spark ETL engine
- Flexible scheduler

Components of aws GLUE

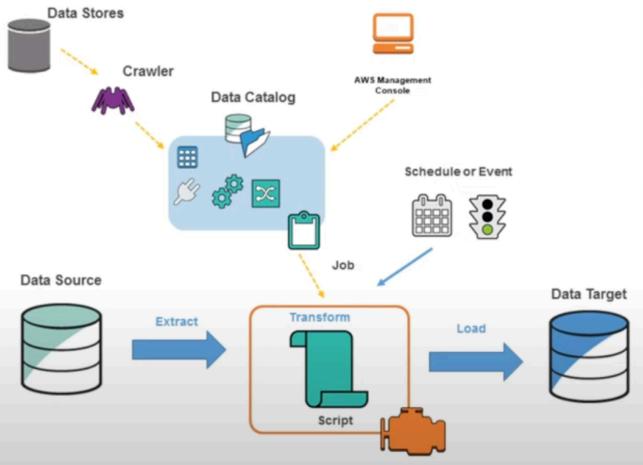
Components of AWS Glue

- AWS Glue Console
 - Orchestrate ETL workflows
- AWS Glue Data Catalog
 - Persistent metadata store
- AWS Glue crawlers and classifiers
 - Detect & infer schemas to store it in Data catalog
- AWS Glue ETL operations
 - Automatic ETL code generation in Python or Scala
- Streaming ETL in AWS Glue
 - ETL operations on streaming data
- AWS Glue Jobs System
 - Enable to orchestrate the ETL workflow which can be scheduled or triggered based on events

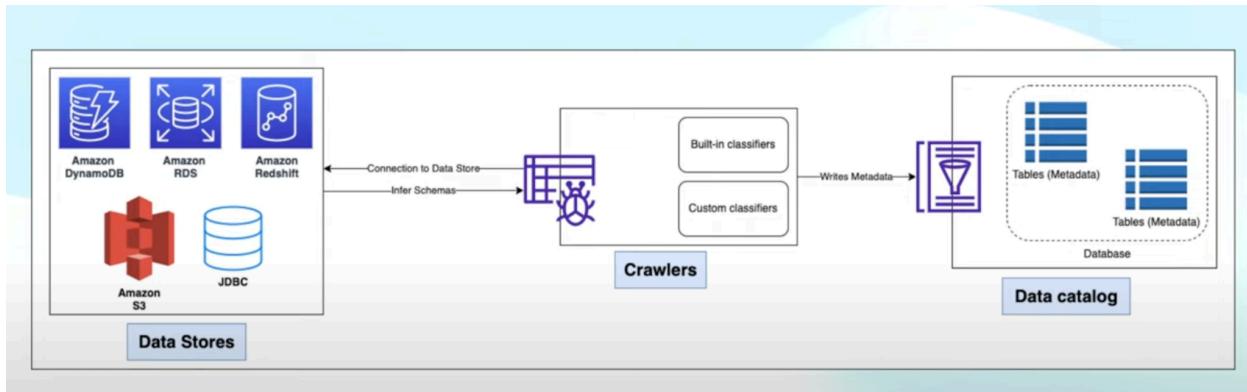
AWS GLUE WORK flow

WorkFlow | AWS Glue

- Define crawler to populate the Data catalog
- Create the ETL job
- AWS Glue generate script for the ETL job or you can also provide/write one
- Run the job on-demand or define the scheduler or Trigger
- Extract data from DS, transforms it & load it into the Data Target



More info

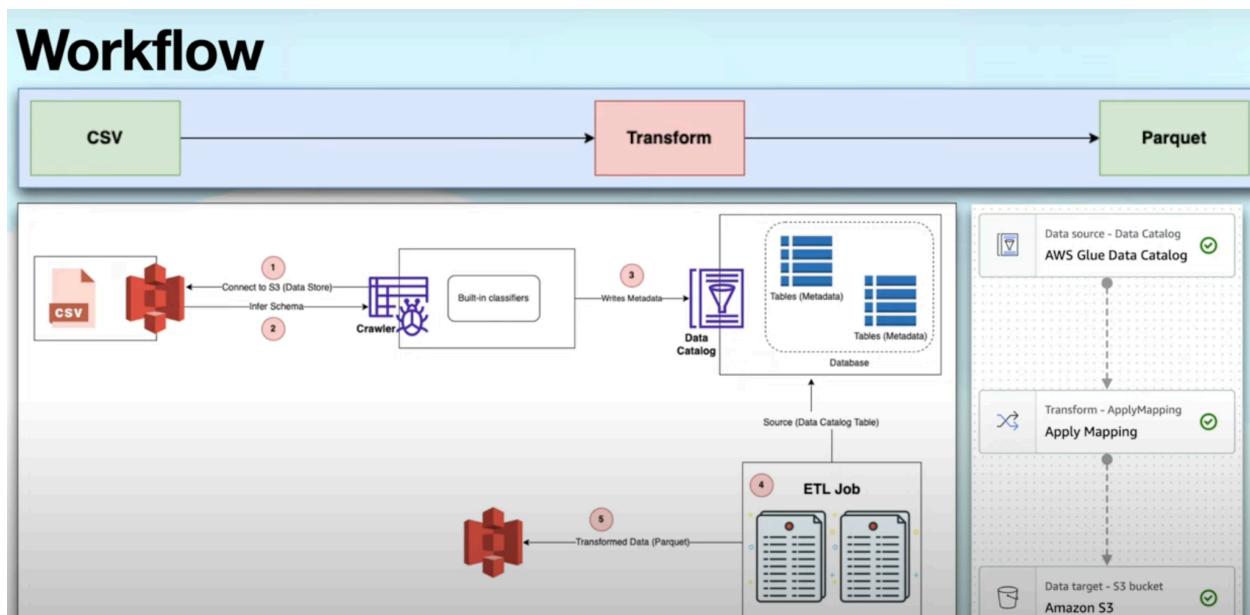


Limitations

Limitations

- Limited to Python & Scala programming language
- Tightly coupled with AWS eco-system
- Dependency on Spark
- Little or no information about what's going on under the hood

CSV to Parquet :



Steps :

- Create S3 Bucket
- Create IAM role with appropriate permissions for AWS Glue
- Create database in AWS Glue
- Create & run crawler
- Create ETL Job to transform CSV to Parquet