**APPLICATION KT SESSION DOCUMENT:**

**5th Dec 2022**

**ARCHITECTURE:**

**The above diagram shows the Architecture of the Project:**

* Our Project has Source Database i.e., SQL Server and Oracle from where we are getting the data.
* There are two types of data i.e., Structured and Unstructured data.
* We are working on structured data only.
* There are three ways to pull the data.
  + Streaming
  + CDC – Change Data Capture
  + Batch
* We are working on CDC and Batch data only.
* For CDC, we are using QDI (Qlik Data Integration) Tool.
* For Batch, we are using SQOOP Tool.
* QDI is not directly connected to database it’s connected to transactional logs & used to extract the data from transactional logs
* SQOOP is connected to database & it used to extract the data directly from the database.
* CDC will capture the data from the source at the time of change happen in source side.
* Batch will capture the group of data in a particular interval of time.
* From source data, the data is coming into Mirror Layer.
* Mirror Layer stores the Replica of the Source data.
* We use the Mirror Layer instead of directly using Source Databases to protect the Source Data from any data loses while operating on the Source Data.

In Mirror Layer, we perform our required operation by using our technology like PySpark, R, Python and ELK (Elastic Seach-used to handle big data) and we store the final output data in Consumption Layer.

* From the Mirror Layer, the data will be transferred to the Consumption Layer.
* From Mirror Layer, it goes to the Consumption Layer.
* From Mirror Layer, it also goes to the Curated Layer, In Curated layer, we have some few tables that will be used by multiple applications.
* In Curated Layer, we store those data which is using multiple time.
* From Consumption Layer, the data will be transferred to Dashboard Layer.
* In Dashboard Layer, we will review and check the data whether the data is updated or not.
* QlikView application is used for visualization.
* We have oracle and SQL server source databases from which we are extracting data to our environment (Data Lake – central repository of structured and unstructured data in an organization).
* There are 3 ways to injecting the data from the source database
  + 1. Streaming
    2. CDC – Change Data Capture
    3. Batch – Group of Data
* But we are using only CDC and Batch Process to inject the data
* CDC will capture the live changes data at Source Side. (if 2cr data available in db – Updated 1cr then it will bring remaining 1cr only)
* Batch/ Full Load will bring complete data at a time. (Total 2cr)
* Incremental Load is there – From last run time to till now.
* Compute Engine Layer part is handled by Platform Team only.
* The data captured by QDI and SQOOP will not directly send to the target instead of it will send to the Staging file (it is the intermediate location of data transfer) to reduce the cost.
* The data will be transfer to the target by using Data Sync (it is a script which will transfer the data from Staging to the Target).
* Target is a S3 Bucket but main target is Consumption layer .
* We are using S3 Bucket for both Staging and Target.   
  In Consumption Layer has the final data inside it and it will be replicated in Dashboard.
* We are using QlikView as a Visualization Tool where user can see the final data.
* We have some Angular Application where we are using RDS as a Database because Hive is not compatible for Angular Applications.

There are 5 databases – RMD, ODW, E-Service, Proficy, Proficygrr – CDC Process

We are extracting from 2 different types of databases which is Orcale(RMD, ODW, E-Service) and SQL Server (Proficy, Proficygrr)

Compute Engine Layer – Platform Team is handling this layer.

EMR – Acts like Temporary memory. It will provide memory while jobs are running like RAM.

Managed Scaling – It is a scaling similar to autoscaling, but we can also set maximum limit for nodes.

**6th Dec 2022 Appworx**

* It is used to Schedule a Job.
* We have Flows and Jobs.
* Flows contains dependent jobs while Jobs contains script which is used to export the data.
* In Appworx, we are checking the status of the job.
* We have different types of status
  + Initiated – A Flow has triggered
  + Running – A Job is running
  + Aborted – Failed
  + Pred-wait – Waiting state for the previous job to be complete
  + PW-Skip – Automatically deleted
  + Self-wait – Same job gets triggers
* We have three Agents
  + WABDALAKEP01 – used to load the data for mirror jobs
  + WABDALAKEP02 – used to load the data for consumption jobs
  + QlikView – trigger the job in QlikView side
* All Mirror and Consumption jobs are scheduling here.
* If any job is running more than its average time, we need to check the progress in backend (if it is not progressing or failed, will kill the job and rerun) and raise the incident ticket to the Platform team.
* Mirror Jobs are Single Jobs, so they don’t have Parent name while Consumption Jobs are Flows which contains Parent name.
* If some job gets aborted then we need to download the logs file and check the error.
* To check the logs file –> click on filter – history – paste the job name – uncheck current date – click on okay