IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ETL FOR UPDATING DASHBOARDS

[1]Nutan Sonale TV, [2]Dr B Renuka Prasad,

Abstract: Business intelligence is a key factor for any manufacturer to distribute the resources strategically. processing of data to get incites is a tedious task and will take up a large resource. The dashboards that consume this processed information require the data to be in a certain form. Some dashboards require the data to be uploaded periodically. Processing data as it is produced from sources is a waste of resources and time. ETL(Extract Transform Load) provides a way of aggregating data, transforming it, and loading it to the destination. It is a process that is custom-built for every job but, the architectural structure is the same. Even if the structure is the same there are different strategies for carrying out the ETL task. The automated ETL tasks will save time and will be reliable since they are purpose-built. The standard ETL task only transforms the data and the analytics and other processes are carried out. The analytics regarding the characteristics of data will help maintain the data and access the quality of the data.

***Index Terms* - Component, formatting, style, styling, insert.**

I. INTRODUCTION

There are certain ways a dashboard can be updated. the dashboards used for IoT can be updated depending on the architecture. one can update directly from the sensor or store it on the cloud and update the same data to the dashboard. the need of processing that data gives us different challenges. We can consider different levels of IoT architecture. There are six levels of IoT deployment templates. In those templates, we see services that provide processed data for publishing on the dashboard or use it for other purposes. This process includes the transformation stage of data. The processed data can be stored on a database or can be directly used for publishing on a dashboard but, the data should be formatted so that the consumer of that data finds no difficulty in processing or using the data. This is the basic task in an ETL process. But the need for processed data may vary depending on data. Generally, ETL is used in data warehousing and migrating data from one source to another. But one can use the same approach to get the data from the source on demand.

II. PROBLEM STATEMENT

In Big data analytics, we normally use the ETL to extract data from the sources, transform them and store the data in an appropriate environment. This approach is effective when the data need high resources for transforming it to the required format. the stored data is then used for analysis. This puts all the intensive work on the ETL pipeline before storing. This work can be shared between that stages when the data is fetched for analytics.

There are a variety of dashboards that uses a source to publish some processed information charts and many more. The dashboards use a source for fetching the data. The source may be a database or a feed from a device. When we are processing these data the dashboard has to compute the analytics or just have to publish the incoming data. Instead of this one can process all the data in a data pipeline and the dashboard can just display the already processed results. This will reduce resources used by the dashboard and the pipeline behind the dashboard can be automated so that the data will be updated automatically.

III. ETL (EXTRACT TRANSFORM LOAD)

ETL is a process that is used for data integration. The ETL stands for Extract Transform and Load process in the integration procedure it can also be modeled as the directed acyclic graph where every node is a process that is related either to extract, transform or load process. These are the key important steps where one extracts data from one or multiple sources transform the data to the desired schema and load the data to the destination. ETL is processed in batch or can be performed in real-time. The batch processing extracts the data periodically and carries out the further process. The real-time ETL will be a streaming service or an event-driven process. The batch ETL tasks apart from transforming data to desired form one can perform analytics while the data is in the pipeline the batch size and the purpose of analytics should be appropriate for performing the analytics in a pipeline. The obtained results can be stored or can be consumed by the applications. This gives us the insights required and the data in the desired form.
comparison with data pipeline and ELT

IV. Comparison with Data Pipeline and ELT

4.1ETL Vs Data Pipeline

ETL is a general name given to many tasks that are intended to fetch data from different sources and putting it to the desired form and loading it to a destination source. The ETL is used in the data warehouse and data lakes to migrate the data from legacy systems or can be used for merging the data from different sources to a single destination. Data pipeline on the other hand is built to transport the data without necessarily transforming it.

4.2ETL Vs ELT

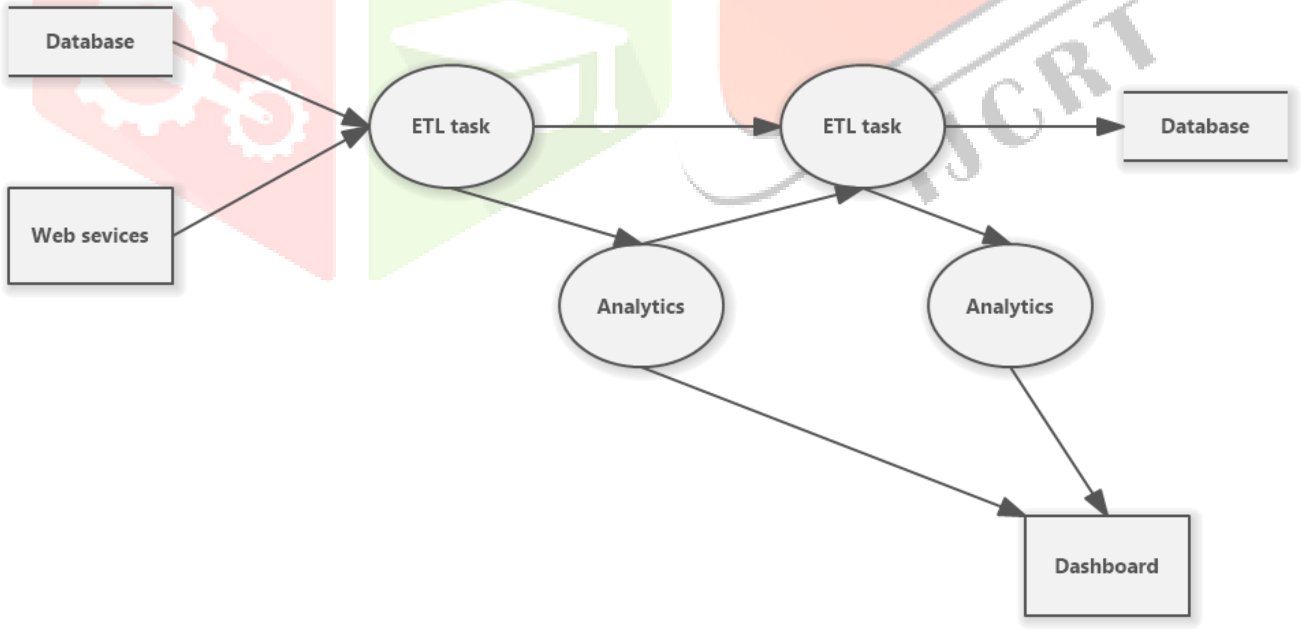
ELT is said to be the modern way for ETL where data is extracted and loaded before transforming it. This depends on the requirement and the resources. If we consider the batch extracting approach there is a restricted size for the data collected and one can estimate the resources required for processing every batch. In a real-time data flow, one can store the data before transforming it if one can't process the data in real-time. The need for certain architecture comes from the requirement and the resources available.

4.3ETL Vs Data Analytics Pipeline

The data analytics pipeline is a group of processes where one gets the data and performs series of operations to get insights. The information gathered can be in the form of charts or plots. The data extracted will be in a well-defined format and the output generated can be used in dashboards are for producing the reports. ETL does not directly provide the analytical results but the data in the pipeline can be used for producing the analytical results.

V. Proposed System

ETL processes after extracting the data from sources the process of transforming the data takes over, this stage can be advantageous. While transforming the data with multiple sources there will be an intermediate phase for the data. The data might not be in the final form but if we include the analytics on these intermediate data we can have insights into the data required. These insights can be further included to access the data processed and filter or categorize the data. This gives us the combination of ETL and data analytics pipeline where data is being transformed depending on the analytics. For example, consider we are reading logs for the sessions of a webs service on different customer sites. The logs can be categorized based on different pages and errors on different pages. One can get the information for how many errors occurred and how many requests came for a particular resource while the data is fetched from each resource then it can be combined with different customer sites and can be further analyzed or loaded into a database. This is a small example that analytics may be part of ETL or may be included to support the ETL process. This approach can be used for accessing the quality of the data and faults found in the data. This will get the information regarding the situations where the data is being faulty. The information gained can be stored and also can be used for a dashboard regarding the data or generating reports.



Block Diagram of Proposed system

5.1Software and Tools

For this particular approach we can use the Knime data integration tool with python scripts for analytics this can be automated by the automation tools such as Jenkins. Knime is an automation tool but specifically built for handling data aggregation and processing to some extent. knime handles the data aggregation part of the pipeline and it puts the data into a format that can be consumed by the python scripts. Jenkins is a free and open-source automation tool that is used for continuous integration and continuous delivery. Its conventional use is to automate the operations depending on the structure of the project and deploy stable builds. But we are using the Jenkins tool for the automation of data integration. Python scripts play the role of computing the analytical steps. Python has a variety of analytical libraries such as NumPy, pandas. This gives us the ability to perform an analytical operation and also plot the charts required using the matplotlib library of python and exporting the results to the desired destination or use the result again by the knime workflow for further transformation.

```
pipeline {
   agent {label 'ops_ubuntu' }
   stages {
      stage('SSH_tunnel') {
         steps{
         retry(count:3){
            sh '''#!/bin/bash
             ssh -tt -N -o StrictHostKeyChecking=no -F /home/user/.ssh/config -i /home/user/.ssh/id_rsa -L 5433:hostname:hostport
hostname &
            '''
        }}}
      stage('knime_workflow_1'){
         steps{
         retry(count: 3){
             sh '''#!/bin/bash
             /home/user/knime/knime_4.3.2/knime -nosave -nosplash -reset -application
org.knime.product.KNIME_BATCH_APPLICATION -preferences=/home/user/Knimeworkflow /linux_knime_pref1.epf -
workflowDir=/home/user/Knimeworkflow/knime-workspace1/workflow1
            '''
         }} }
      stage('process and publish'){
         steps{
         retry(count: 3){
            sh '''#!/bin/bash
            python3 /home/user/pthonscripts/program.py
            '''
       }}} }
      stage('knime_workflow_2'){
         steps{
         retry(count: 3){
             sh '''#!/bin/bash
             /home/user/knime/knime_4.3.2/knime -nosave -nosplash -reset -application
org.knime.product.KNIME_BATCH_APPLICATION -preferences=/home/user/Knimeworkflow /linux_knime_pref1.epf -
workflowDir=/home/user/Knimeworkflow/knime-workspace1/workflow2
            ''' }}}

}
```

The above code is the declarative pipeline used in the Jenkins tool. Here the python script does the analytics for the files or data that are the output of the first workflow the other workflows can take the files from the results from the python scripts and continue the process. The knime has the nodes that can write to csv files and can read from csv files or one can store the data on a database as well.

VI. Constraints

here we are only considering an ETL task that is triggered periodically hence this approach is suitable for batch processing. the analytics added to the ETL can increase the throughput time of every batch of data.

VII. Conclusion

Here we are using an approach that includes analytics into the ETL process this gives us insight into the data collected and this can be published on the dashboard. This data can be used for generating reports about the quality of data. The approach of performing analytics after loading all the data is a traditional way but one perform the same analytics at different stages of ETL if needed. The ELT approach is a suitable approach for real-time data and large batches of data. ETL can be used for moderate and small batches of data where one can include a quality check and analytics with the ETL pipeline.

Acknowledgment

I thank Dr. B. renuka prasad for providing guidance during the time of the project. My institute RV college of engineering for providing me opportunity to take up internship. my colleague chandan kumar for providing guidance while the study of Knime tool.

REFERENCES

[1]P. O’Donovan, K. Leahy, K. Bruton, and D. O’Sullivan, "An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities", Journal of Big Data, vol. 2, no. 1, 2015.

[2]"Pipeline Syntax", Pipeline Syntax, 2021. [Online]. Available: https://www.jenkins.io/doc/book/pipeline/syntax/. [Accessed: 25- May- 2021]

[3]KNIME Flow Control Guide, 4th ed. Zurich, Switzerland: KNIME AG, 2021.

[4]KNIME File Handling Guide, 4th ed. Zurich, Switzerland: KNIME AG, 2021.

[5]KNIME Components Guide, 4th ed. Zurich, Switzerland: KNIME AG, 2021.

[6]D. M. Tank, A. Ganatra, Y. P. Kosta and C. K. Bhensdadia, "Speeding ETL Processing in Data Warehouses Using High-Performance Joins for Changed Data Capture (CDC)," 2010 International Conference on Advances in Recent Technologies in Communication and Computing, 2010, pp. 365-368, doi: 10.1109/ARTCom.2010.63