International Journal of Management, IT & Engineering
Vol. 8 Issue 9, September 2018,
ISSN: 2249-0558 Impact Factor: 7.119
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

AMAZON WEB SERVICES (AWS) GLUE

Kalyan Sudhakar[*]

Abstract

In the era of technological advancement, companies have resorted to using ETL tools in an effort to integrate their data. This research paper seeks to investigate the features of the AWS Glue which involves the management of data extraction, loading, and transformation. The paper is informed by an in-depth literature review of websites and scholarly materials that contain relevant information about the use of AWS Glue. The findings of the literature review are then analyzed based on the theory of constraint. The paper also compares the similarities and differences between AWS Glue and other ETL tools that are commonly used. The study reveals that AWS Glue has various limitations that need to be addressed by the developers in order to enhance its efficiency.

Keywords: Data, management AWS Glue, ETL, Pipelines

1 Introduction

AWS glue is a service that entails the complete management of data extraction, loading, and transformation. The technology operates on the basis of the interaction of many elements to establish, manage and operate an individual's data warehouse [1]. The service aims at categorizing and preparing data for efficient movement from one store to another. The process involves the use of a catalog, which is the central metadata repository, a scheduler that ensures dependency of resolutions, consistency, and monitors entries, as well as an engine which generate either Scala or python code automatically. The use of AWS glue is essential for any business or organizational context given that the service enables individuals to prepare data for analysis. This research paper aims at exploring the key features of Amazon AWS glue, its

[*]

benefits, and the data pipelines that characterize the glue. Moreover, it seeks to make a comparison between Amazon AWS glue and other Extraction, Loading, and Transformation (ELT) tools.

Given that the success and profitability of most companies in the contemporary world rely on efficient data flow; information technology experts have developed several tools to enhance the process of maintaining the data. ELT tools such as the AWS Glue enable individuals to spend less time finding and accessing data for use. In order to achieve its objective, the paper contains a review of relevant scholarly literature, which discusses the service and provides insights from various websites, which contain information that pertains to the service. Similar to other ETL tools, the AWS Glue follows these stages to accomplish the planned outcome: the initiation stage, creation of reference data, extraction of data from the source, validation, transformation of data, loading the data into tables, writing audit reports, publishing the reports, achieving the job, and cleaning up the system.

The major function of an AWS glue is to build a data warehouse on the basis of the ETL principles. To accomplish its role, the AWS glue relies on other AWS services. It transforms data, stores the logic of a job, establishes notifications for monitoring purposes and creates runtime logs by initiating API operations [2]. The AWS Glue console is used to connect the related services into an application that can be managed conveniently, thus, enabling an individual to concentrate on the creation and monitoring of data extraction, transformation, and loading. The console serves as an administrative assistant as it establishes operations in place of a human being. The latter only requires supplying the necessary credentials and other properties to the former [2]. Users of the console can manipulate a data catalog to create a job. Some of the major functions of the AWS Glue console include the following; scheduling the initiation of a crawler second, the definition of the objects such as connections and tables, searching and evaluating the objects, definition of schedules and events for initiating jobs, and editing scripts.

In order to access resources and features of the AWS Glue, a user is required to establish necessary policies and roles by using an AWS Identity and Access Management. Setting up the environment involves five major steps as listed below; first, the user should establish the policy required to gain access to the features of the AWS Glue. Secondly, an individual should set up an

Identity and Access Management role for the service. The third step involves paring policies to the users of the established IAM in order to grant them access whenever they sign in to the console. The fourth step entails creating a notebook server policy to enable the creation of notebook servers. The last step involves the creation of a role for the created notebooks. Below is a flow chart that displays workflow in an AWS Glue system.

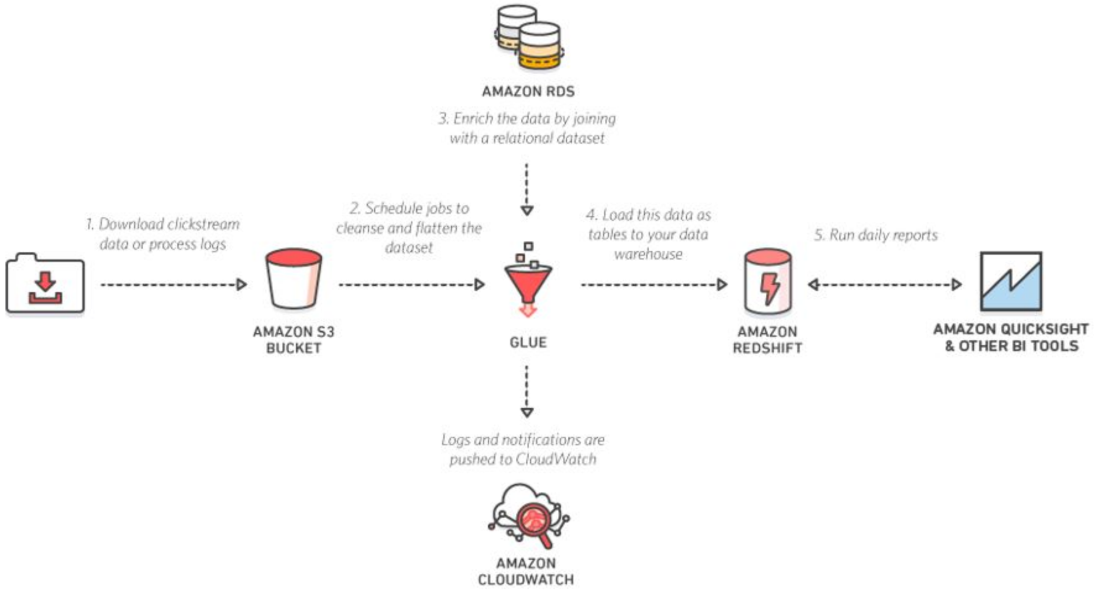
AMAZON RDS
3. Enrich the data by joining with a relational dataset
1. Download clickstream data or process logs
2. Schedule jobs to cleanse and flatten the dataset
4. Load this data as tables to your data warehouse
5. Run daily reports
AMAZON S3 BUCKET
GLUE
AMAZON REDSHIFT
AMAZON QUICKSIGHT & OTHER BI TOOLS
Logs and notifications are pushed to CloudWatch
AMAZON CLOUDWATCH

Figure 1. A flow chart of workflow using AWS Glue [1]

AWS Glue enables the processing of ETL jobs without relying on a server. The design of the console is meant to segregate data, protect the data during transits and allow for the access to the data as requested by a customer. Users are required to input data sources and targets in the virtual private cloud when provisioning an ELT job. Other than the VPC, users are required to provide certain essentials that are necessary for the access of data. Such credentials include VPC ID, Subnet ID, security group, and IAM role.

The AWS Glue uses private ID addresses to create elastic network interfaces in a user’s subnet. The spark job makes use of the created interfaces within the subnet to enable an individual gain access to the data sources and targets. The traffic within and without the spark environment is controlled by a user’s VPC and his or her networking policies [2]. Nonetheless, an individual can make calls to the AWS glue libraries in order to replicate the traffic, which heads towards the AWS Glue API via the VPC. In order to accomplish the work that is necessary to manipulate data, the user has to define the jobs in the system. This process entails the following actions; first, description of the crawler to ensure that the catalog contains the necessary metadata table definition contents. Secondly, generation of a script in order to facilitate the transformation of the data- this process is carried out by the AWS Glue. Lastly, running the job on demand or alternatively set it to start when a specific command is issued by the user.

Key Features

The following are the key elements that make up an AWS Glue. First is the data catalog, which is the metadata storage system. Every AWS account has a catalog, which entails job and table definitions among other credentials which are used to control the environment of the AWS Glue. Another key feature is the table, which is the definition that represents the users' data. The table also determines the schematic arrangement of data in the account. It contains the names of columns, definitions of data type and other metadata components. The AWS Glue creates ETL jobs by using the tables as targets and sources.

Another important element is the crawler. The crawler is a program, which connects the system to either a source or a target data store. It follows a set of protocols in order to determine the schema for a specific data, thus, establishing the metadata in the catalog. The other key feature is the classifier which establishes the schema for a user's data. There are various types of classifiers that are required to accomplish the ETL work. As such, the AWS Glue produces several classifiers in order to accommodate various types of files including JSON, and AVRO among others [1]. Moreover, the system avails classifiers that can be used for managing common relational databases. An individual can use a grok pattern to create his own classifier. Alternatively, someone can simply write a specific row tag in an XML document.

The connection is also a key feature of the system. It entails the elements that are necessary to create a link between the data, the system, and the data store. The database is a component of the AWS Glue system that is made up of a set of table definitions, which are arranged in an organized and logical manner. The element of job in the context of the AWS Glue system refers to the logic, which the system uses to carry out an ETL work. It is made up of scripts, data targets, and sources. The job is the central feature that makes up the AWS Glue job system, which provides a platform for the orchestration of the ETL workflow. The workflow involves supplying details of a job to the system and defining the source as well as the target. The system then generates a script which is then stored in a data catalog. The logic can either be initiated using triggers or they can be set up to start at a specified time or command.

The script is the code that facilitates the migration of data by obtaining it from the sources, transforms it, and transfers it to the target [1]. There are two types of scripts that are associated with the AWS Glue. These are Scala scripts and PySpark script. Another feature that characterizes the system is called transform which refers to the logic code that the system employs while changing data into a different format. The process of initiating an ETL job is referred to as Trigger. As mentioned before, the process can be set up on the basis of a time or action. Development endpoint is another key feature, which refers to the environment that can be used to create and test scripts. Finally yet importantly, a notebook server is an element of the AWS Glue, which is used to initiate PySpark statements.

The service can be used to migrate data from the Amazon Web services cloud data to an individuals' preferred data store. The process of storing information in a data warehouse- which is made possible by the AWS glue- is essential for all business because it gives room for the integration of information and creation of a common source of data which can be referred to when making decisions. The use of AWS glue while building a data warehouse is also important as it enables the simplification of various tasks which would otherwise require more resources to set up and maintain.

How the AWS Glue Works

The first step involves using the AWS management console to input the necessary resources. The AWS Glue then crawls the registered data in order to establish a catalog. This process involves using the use of pre-built classifiers such as CSV and parquet among others. The next step involves selecting a data target and source. Afterward, the Glue extracts data from the source; transforms it to replicate the target schema, and then loads it to the target. At this stage, users can use the console to edit, test, and debug the code. The last stage entails scheduling the ETL jobs for retrieval by the user. The schedule can either be based on an even or a specified time.

The process of extracting and transforming data from the source to the target using an AWS Glue involves five major phases as outlined and presented in the flowchart below. The job is initiated by a trigger. Afterward, the data is extracted from the source and transformed based on the scripts that have been created. Notably, the transformation of data is done by a code (either PySpark Python or Scala code, which is contained in the scripts). The next phase entails loading the transformed data into the selected target. The last stage involves collecting the statistics of the job and transferring it to the catalog.

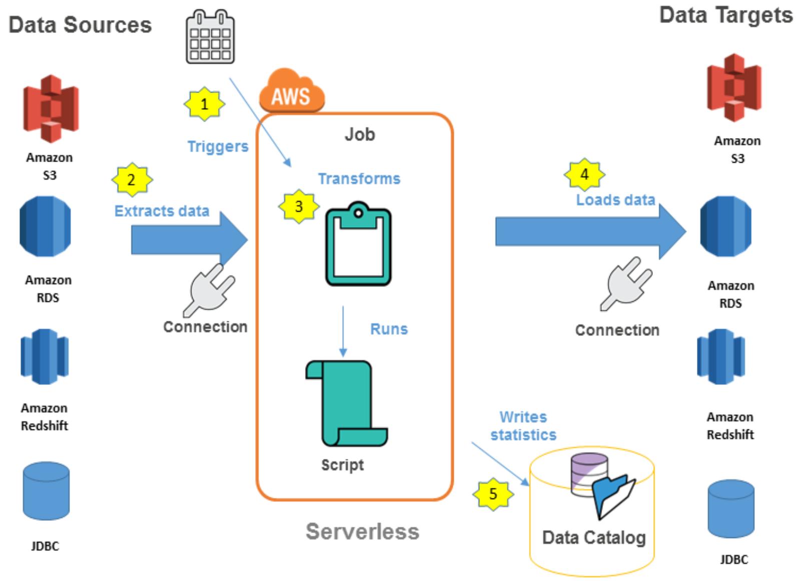

Figure 2. A graphical presentation of the phases of extracting and transforming data [2]

Benefits of Using an Amazon AWS Glue

Some of the benefits derived from using an AWS Glue include the following; first, it is easy to use, thus, making it convenient to analyze data. Given that it is an automatic process, users find it quite convenient to manipulate data as they desire. Users also experience convenience when using the service because the system allows them to change the nested JSON to key-value pairs, which is made possible by a concept called Relationalize. The process simplifies the ETL process. Secondly, AWS Glue is part of an integrated system that includes other AWS services. As such, a user can store and retrieve his data in a variety of Amazon services such as Amazon RDS. This benefit is attributed to the fact that the AWS Glue can support the processing of data retrieved from numerous services. In addition, the service enables an individual to use various applications such as Amazon EMR simultaneously [2]. Notably, access to such application increases the value for users' money in the sense that they acquire the services offered by the application besides those that they receive from the AWS Glue.

Thirdly, the maintenance of servers do not require special set up infrastructure as mentioned before. Given that it does not rely on any servers to operate, it does not require any maintenance cost. Therefore, it can be argued that it is one of the cheapest ways of processing an ELT job [1]. The customers are only required to pay for resources that are the system uses while processing their jobs. Fourthly, the service is developer friendly in the sense that an individual can retrieve a customized ETL code. Moreover, the codes that are generated from the system are both portable and reusable. An individual can also use customized resources from other services and use them alongside the ETL code generated from the Glue. The codes can be used anywhere given that they are generated using open frameworks, thus, they lack lock-in. Another benefit of using the service is because it is set up to scale the underlying resources automatically. Additionally, the system ensures that all jobs are processed accordingly by retrying automatically if they fail.

Finally, the system is set up to ensure that data is not lost in the process of extraction and transformation. Users can monitor and report any errors using three major tools: the Amazon CloudWatch Events, which enables a user to compute events automatically, Amazon CloudWatch Logs – which allows the users to store and monitor log files, and AWS CloudTrail, which identifies the API calls that the customer has made and transfers them to a preferred Amazon S3 bucket.

Data Pipelines with Glue

A data pipeline is a web-based service that facilitates the automatic movement and transformation of data. AWS Glue data pipelines enable users to establish workflows that can only be initiated upon the successful completion of the defined tasks. The user is required setting up the parameters within which he needs his data to be transformed and the rest of the functions are performed by the data pipeline- including the enforcement of the logic as set up by the user. Some of the features that characterize using AWS Glue data pipeline include the following; Definition of the pipeline which entail the process of setting up the specific logic for managing data. A pipeline is a component that initiates the defined tasks. This component fulfills its function by establishing Amazon EC2 instances. Notably, users can halt the data management process while the pipeline is processing the data, redefine or edit the initial command and reactivate the process again in order for the change to take effect [3].

In addition, users have the option of keeping the pipeline for reuse or alternatively deleting it if it will not be required again. The last component is the Task Runner, which evaluates the defined tasks and perform them thereafter. Users can install a Task Runner in their computers and use it to compute the resources automatically based on the pipelines that they have defined. In addition, an individual can either write his own Task runner application or use the one that is preinstalled in the AWS data pipeline. Organizations can use the AWS Data pipeline to generate traffic report by storing their web server's logs to Amazon S3 on a daily basis then run a weekly cluster of the logs [3]. The process of copying data daily and launching the Amazon EMR cluster on a weekly basis is carried out automatically by the AWS Data Pipeline. Further, the Amazon EMR is set up to wait until the data from the last day according to the schedule is uploaded to the Amazon S3, after which it begins to analyze the data. The process stalls even if the logs are not uploaded in time. This mechanism is convenient for the user as it ensures that he or she obtains a full analysis of all the relevant data.

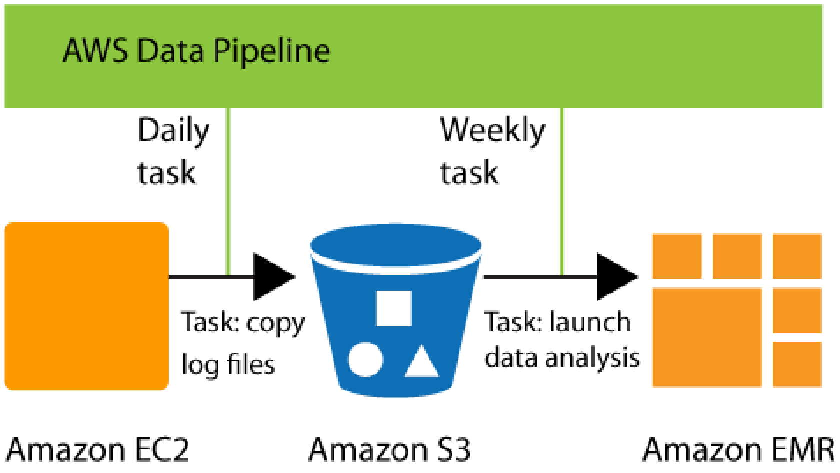

Figure 3. A flow chart of the AWS Data Pipeline [3]

Companies can use various interfaces to manage their AWS data pipelines including the following; AWS management console, AWS SDKs, Query API and AWS Command Line Interface. Users of the pipeline are charged only the cost of using the service. The cost of the service depends on how often an individual uses to process their data and the location where the activities are run.

2. Research Method

As mentioned in the introduction section, the paper seeks to understand the components of Amazon AWS Glue and its benefits by reviewing the available literature which contains information that pertains to the service. The major sources that inform the paper include the Amazon website and scholarly articles which entail analysis of the service and other tools that are used to perform ETL jobs. The information obtained from the reviewed literature will be analyzed in light of the theory of constraint. Golmohammadi highlights that the theory of constraints can be applied to identify the weakness of a system and come up with solutions to the limitations accordingly [4]. Arguably, most systems or web-based services are subject to constraints that may affect the workflow or interfere with the expected results. As such, it is vital to identify the limitations that characterize the systems involved.

Unveiling the weakness is undoubtedly important as it enables the developers to make appropriate changes in order to enhance its efficiency and improve users' experience. The properties observed from the literature review of the AWS Glue will be compared with other information that pertains to other tools used in carrying out ELT jobs. This paper hypothesizes that the Amazon AWS Glue has limited weaknesses that limit the users form completing their extraction, loading, and transformation jobs efficiently. It also hypothesizes that the services offered by the Amazon AWS Glue are better compared to other tools in as far as ELT jobs are concerned.

3. Results and Analysis

In view of the literature review of the properties of the AWS Glue, someone can observe that the AWS Glue enables users to manage data extraction, loading, and transformation jobs. The well-organized components of the service enable individuals to move their data efficiently from one data warehouse to another at an affordable price. Given the importance of data analysis in the contemporary business environment, the use of AWS Glue is critical to the success of any company or organization. For instance, users take minimal time to carry out ELT jobs, thus, saving more time to carry out other necessary activities. In addition, the service does not rely on any servers, thus, companies do not require to set up any infrastructure in order to their jobs. Notably, setting up infrastructure may be expensive, thus, the use of Amazon AWS Glue is beneficial with respect to the cost.

The Amazon Service Website also reveals that there are several components, which make the system upon which the AWS Glue operates [1]. For instance, the system entails the Data Catalog, which contains metadata. It provides the platform where users can manage their databases, creates tables and defines their schemas. Another component identified from the literature is the ETL Engine, which enables the creation of ETL jobs. Analysts rely on this component to generate the necessary PySpark code which in turn facilitates the loading and transformation of data from the source to the targeted store. The Scheduler is also identified as a component of the system, which starts the process of extracting, loading, and transforming data. The scheduler can either be set the activities to begin based on an event or time.

The literature review reveals that the creation and management of an ETL job is a simple process that involves using the above-mentioned components among others to process one's data appropriately. Additionally, the review indicates that individuals can use the AWS Glue in different ways by simply decoupling the components of the system. For instance, the service can come in handy while carrying out various activities including data exploration, export, cataloging and aggregation of logs.

Similarity with other ETL tools

The need for data-dependency has forced several organizations and businesses to invest in different data warehouse systems. All ETL tools were developed for the purpose of enhancing the data management system while reducing the time consumed and cost of operation simultaneously. Some of the ETL tools which have the same characteristics and are commonly used include the following; first, Adeptia Integration Suite has the same properties as AWS Glue in the sense that both tools provide users with the platform for creating data integration connections [5]. Just like the AWS Glue, Adeptia Integration Suite transforms data and sends it to a targeted store as defined by the user. In addition, the systematic design of both tools allows for data to be moved from several sources and formats and moved to the same schema that the user has defined.

Adeptia Integration Suite can be used to manage data obtained from different sources in a central repository and publish the rules used in a pdf document. Similarly, AWS Glue allows companies to extract data from various sources, transform them, and send them to a selected target from a central point. Sybase is also an ETL tool that is used by companies to load data from different sources simultaneously, categorize them appropriately, and transform them for ease of analysis. Similarly to AWS Glue, Sybase enables users to integrate data automatically and it facilitates the process of designing, planning, and monitoring of data as defined by the users [6]. In addition, both tools have the ability to transform data into any format. In addition, both tools have real-time schedulers, which trigger a workflow process.

Finally yet importantly, the Jasper ETL tool also contains the same tools as the AWS Glue. Both tools can be set up by the users to initiate the data management process automatically. In addition, users are able to generate a statistical report of a job while using JASPERETL just like the AWS Glue generates a statistical outcome of the data management process [7]. The similarity between the two tools can further be identified in the generation of a code that can be used to transform and transfer data in any machine. Users can also monitor and adjust their workflow when using both tools to integrate, load, and transform data for purposes of analysis. Overall, all the tools mentioned above are easily managed using a single console, which enables users to monitor the entire workflow process and track the execution activities as scheduled. Lastly, the AWS Glue enables the storage of logs and relevant data for ease of retrieval when they are needed.

Differences between AWS Glue and Other ETL Tools

Notably, the AWS Glue has various unique features that are different from other commonly used ETL tools. Some of the differences that can be notable between the AWS Glue and the other

ETL tools include the following; Adeptia Integration Suite has a feature that enables human intervention whenever it is needed. For instance, the tool sends an email to the users as well as a notification on the task manager in order to alert them of the impending action [5]. The users can then proceed to make the necessary inputs in the system through a web-interface in order to facilitate the completion of the tasks. On the other hand, the AWS Glue only allows users to edit and debug the code generated by the system using the console.

Unlike the Adeptia Integration Suite, the AWS Glue does not rely on human intervention to complete the tasks as scheduled. Instead, the entire system is fully automated. In addition, the Adeptia Integration Suite provides users with the platform to merge the processes of data integration with business management [5]. On the other hand, AWS Glue only focusses on the integration of data and relevant analysis which the users retrieve for use in making a decision in the business context. Furthermore, the Adeptia Integration Suite supports not only human-to-system data flows but also system-to-system data flows [5]. This is, however, not the case for the AWS Glue which only supports system-to-system data flow.

Unlike the AWS Glue whose benefits are limited to the creation of data warehouses by orchestrating users' ETL jobs, the Jasper ETL can be used as a stand-alone tool for enhancing the functionality of other systems and applications; besides its primary function of integrating data from different sources into a data warehouse for retrieval. In addition, JasperETL allows users to create customized components externally then implement them in the tool [7]. On the other hand, AWS Glue does not have this feature, thus, the users of the latter manage their data using only the preset components. A notable difference between AWS Glue and Sybase is that the latter can only operate on windows system while the former can be used on a different operating system other than windows.

Overall, all the ETL tools identified above rely on servers to extract and load information, unlike the AWS Glue which does not require users to set up the necessary infrastructure to run a job. As such, the use of AWS Glue is more beneficial in the sense that it saves both the time and the resources of the company. For instance, users of Sybase have to ensure that their source databases are connected with the connectivity libraries and the targeted libraries where the data is being loaded.

Analysis

It is evident from the results that the AWS Glue has outstanding features that enable the efficient integration of data and facilitation of workflow. In view of the theory of constraints, some of the weaknesses that characterize the service include the inability to enhance the performance of other application or system, unlike other tools. Notably, the inclusion or modification of this feature would most certainly increase the usability and relevance of the AWS Glue services. Also, the results unveil that the AWS Glue does not allow users to create their own components and use them to advance the data integration process and improve the output significantly. There is no doubt that the lack of this feature may be an inconvenience for the users who would prefer customizing their workflow.

As highlighted in the previous section, the theory of constraints stipulates that information technology systems have weaknesses which can be identified and rectified in order to improve their efficiency. As such, it is vital to point out that the developers of AWS Glue require modifying the system in a manner that accommodates the users' ability to create components externally and include them in the system in order to acquire customized results. Moreover, it would be commendable for the developer of the AWS Glue system to create features that enable companies to use it for other related functions besides integrating data or rather as an ETL tool. There is no doubt that more users would subscribe to the service if it addresses some-if not all- of the limitations that have been highlighted in this paper.

Conclusion

The Amazon AWS Glue is a web-based service that enables users to integrate data into a defined warehouse based on the principles of extracting, transforming and loading data. Similarly to other ETL tools, AWS Glue was developed to enhance users' management and analysis of data. There is no need for setting up or managing any infrastructure in order to operationalize the AWS glue because it does not require a server. The system's console is designed in a manner that allows for the segregate of data, and to protect the data while it is being transformed and transferred to the warehouse. Some of the key components of the AWS Glue include the console, catalog, crawlers, classifiers, and the job system which are designed to produce the users' desired results. Users of the system derive several benefits including the ease of analyzing data, the interconnection with other AWS services, thus, providing users with several platforms for storing and retrieving their integrated data. In addition, the system is user-friendly and automated, thus, making it easy for information to operate. It is cheap to operate as it does not rely on a server, therefore, companies do not need to set up any infrastructure. AWS Glue data pipelines are used to establish workflows that can only be initiated upon the successful completion of the defined tasks. The users are required to set up the parameters within which the data is supposed to be transformed. The subsequent functions are performed by the system automatically.

The Amazon AWS Glue is similar to other ETL tools with respect to the structural and functional properties. Some of the common similarities include the provision of a platform for integration of data by connecting the source and the target. Another common similarity is the users' ability to use the tools to publish a statistical report of the integrated data, and the workflow among other similarities. On the other hand, the AWS Glue differs from other commonly used ETL tools. Such difference includes the inability to use the system to for other purposes other than the integration of data. An analysis of the AWS Glue in comparison with other ETL tools reveals that the former has significant limitations that need to be addressed despite its efficiency. Therefore, it is vital for the developers of the Amazon AWS Glue to modify the system in order to enable companies to use it for other purposes and also modify the components to suit their needs.

References

[1] Amazon. (n.d.) *AWS Glue – Fully Managed ETL Service*. Retrieved from https://aws.amazon.com/glue/

[2] Kalyani, D., and Sayyed, E. (2017). *Applying Amazon Glue for ETL in data processing*. Retrieved from: https://www.accenture.com/us-en/blogs/blogs-kalyani-sayyed-amazon-glue-etl

[3] Amazon. (n.d.) *What is AWS Data Pipeline? - AWS Data Pipeline.* Retrieved from: https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/what-is-datapipeline.html

[4] Golmohammadi, D. (2015). A study of scheduling under the theory of constraints. *International Journal of Production Economics, 165,* 38-50.

[5] *Top notch ETL data integration software for connecting business data*. Retrieved from: https://adeptia.com/products/etl-data-integration

[6] *Commercial ETL tools.* (n.d.). Available from: https://www.etltools.net/sybase-etl.html

[7] Vidhya, S. Sarumathi, S., and Shanthi, N. (2014). Comparative analysis of diverse collection of big data analytics tools. *International Journal of Computer, Electrical, Automation, Control and Information Engineering, 9*(7).