International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Design and Development of Data Pipelines

Karthik Cottur[1], Veena Gadad[2]

[1]*B.E Department of Computer Science Engineering, RV College of Engineering, Bengaluru*
[2]*Assistant Professor, B.E Department of Computer Science Engineering, RV College of Engineering, Bengaluru*

Abstract - *An In recent days, the prime commodity across the world is the ever-growing data. Huge corporates keep generating data in real time with respect to its clients, customers and employees. This data in its raw form, cannot be interpreted easily but once it is processed and transformed, it can be extensively used for analytics. This enhances the various aspects of the existential characteristics of the aforementioned corporate, such as market competencies, customer feedback, organizational administration amongst others. Considering the scale of data generated by the corporate, it becomes inevitable to dedicate enormous resources, several skilled personnel, time and effort to conquer the objective of data processing, calibrating and storage in-house. The objective is to overcome the challenges corporates bring in data-pipelining technologies and directly receive processed data at the end of the data sync cycle. One sync cycle represents the continuous fetching of data generated or data changed during a specific period such as a fortnight or one month.*

Key Words: Data Pipelines, Cloud, Data Warehouse, Data Analytics, Sync Cycle

1. INTRODUCTION

The rule behind the proposed framework is an information pipeline that associates and concentrates information from source, forms it and pushes it into a cloud-based warehouse for the comfort of analytics and Business Intelligence (BI). Conceptually, a data pipeline is a pathway from source to destination via some transformations for various analytical applications. The first step is to connect to the source of the raw data and fetch the data load using REST API services. The system processes the load, to handle data integrity issues, such as redundant data, skipped data, updated and deleted data, and change of data types for a particular field based on the schema available at the source. Final step is to load the clean data into the cloud-based warehouse.

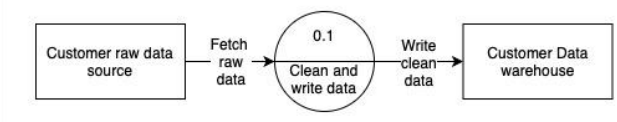

Fig-1: Basic ELT Process

Before you Before, the data pipeline services depended on Extract-Transform-Load (ETL), where information was separated from the source, changed by the explanatory question prerequisites and stacked into the warehouse. The system mentioned was incorporated because of lack of cost-efficient remote data storage facilities at the time; thus, data was transformed so as to store data that were supposed to be queried for analytics. The transformation included consolidation of data using certain algorithms to generate smaller amount of data that would suffice the query requirements. This method had several limitations such as, raw data cannot be directly queried since it is not available at the warehouse end, subject matter experts helped design algorithms and queries that can combine data and extract information enough for analytics from smaller extent of data. With onset of cost-effective cloud-based storage services, a new method came to existence Extract Load Transform (ELT) where the data is extracted from source, loaded on to an intermediate cloud facility known as data lake, where the data is cleaned and then loaded into the warehouses for analytical queries.

1.1 Objectives of Paper

Objectives of the system are set with the requirements of each module.

1. The first objective is to model the endpoints available at the source, to follow the schema and structure of the data fetched

2. The second objective is to actually fetch the data by making API calls and consume them according to the structure of the data

3. The third objective is to clean the data using third party infrastructure. The last objective is to load the clean data onto a remote warehouse, which can be further used for various transformations and analytical procedures

1.2 Organization of Paper

Objectives of the system are set with the requirements of each module.

1. The first objective is to model the endpoints available at the source, to follow the schema and structure of the data fetched

2. The second objective is to actually fetch the data by making API calls and consume them according to the structure of the data

3. The third objective is to clean the data using third party infrastructure. The last objective is to load the clean data onto a remote warehouse, which can be further used for various transformations and analytical procedures

2. Literature Review

The concept is data pipelines is fairly recent and the advancements in this particular domain have been enhanced with recent developments in cloud infrastructure and cloud storage. These are the few innovations in the corresponding field of data pipelining.

In 2009, a research was conducted on ETL Technology [1] and it was based on the following principle. The initial software programs that encourage the first stacking and the occasional refreshment of the warehouse are usually known as Extraction-Transformation-Loading (ETL) forms. There were sure restrictions to this, the extraction of information despite everything stays a difficult issue, for the most part because of the shut nature of the sources, streamlining and resumption issues and nonattendance of a benchmark is preventing future research.

Then in 2012, real-time ETL Data Warehousing was studied [2]. The aim was to achieve Real-Time Data Warehousing which is highly dependent on the choice of a process in data warehousing technology known as Extract, Transform, and Load (ETL).

In 2013, synchronous investigation [3] was progressing in the field of ELT, utilizing an information distribution center's capacity to straightforwardly import crude, natural records, concede the change and cleaning of information until required by pending reports.

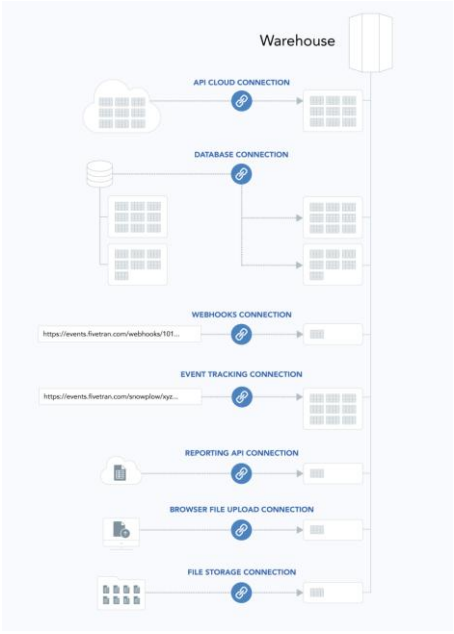

Fig-2: A connector of each of the supported types

Later in 2016, ETL was being embraced for a few applications, for example, in clinical area [4]. This information must be appropriately removed, changed, and stacked into the warehouse while keeping up the integrity of these information. It approved the accuracy of the extract, transform, and load (ETL) process, in this manner populating Clinical research database.

At the same time Amazon's S3 [5] can came in feature, since it gave bulk storage that need not be cleaned or packed to be accommodated. Amazon.com had presented the Simple Storage Service (S3), a low-price capacity utility. S3 plans to give storage as a minimal effort, profoundly accessible help, with a straightforward 'pay-more only as costs arise' charging model.

After one year in 2017, with the onset of Data lakes, the incorporation of Extract - Load - Transform grew faster [6]. The most straightforward expectation of information lake is to munge each datum delivered by an association to give progressively significant knowledge in better granularity.

Another year later 2018, a standardized approach was incorporated to develop a R based platform using SQL. To develop a predictable and pipe-able framework for R that influences SQL to make reproducible research on medium information an effortless reality. It hence carried the challenges of scalability based on amount of data, and also algorithms were not instantaneous in case of medium data which increased latency time. Later that year, another implementation was done to aggregate scientific data for research purposes. A horizontally scalable distributed extract-transform-load system to tackle scientific data aggregation, transformation and enhancement for scientific data discovery and retrieval [8].

In 2019, research was being done to enhance privacy for ETL processes, in particular with biomedical data [9]. To make the necessary huge datasets, data from unique sources can be coordinated into clinical and translational distribution centers. This was acknowledged on the grounds that current ETL instruments didn't bolster anonymization. In addition, basic anonymization instruments cannot be consolidated in ETL work processes at that point.

In the same year, another work process was being examined concerning circulated On-Demand ETL system. DOD-ETL [10], an instrument that addresses, in an imaginative way, the fundamental bottleneck in BI arrangements, the Extract Transform Load process (ETL), giving it in close to real-time. The essential challenge was to deal with different information sources yet additionally give insignificant latency to respond in real-time.

Later that year, usage in financial sector was also being explored. Extract-Transform-Load (ETL) concepts, big data processing methods and oriented containers clustering architecture, so as to supplant the exemplary information combination and investigation process by our new idea (RDD4OLAP) cubes consumed by Spark SQL or Spark Core basics [11]. But additionally, give negligible latency to respond in real-time.

3. Proposed Approach

This project is currently done keeping in mind the relative requirement of a standard ELT pipeline observed for a small-scale source. The system can be used in real world and be of great help to corporates generating huge amount of data on daily basis by increasing the compute and memory used in the cloud.

Conceptually, a data pipeline is a pathway from source to destination via some transformations for various analytical applications. The first step is to connect to the source of the raw data and fetch the data load using REST API services. The system processes the load, to handle data integrity issues, such as redundant data, skipped data, updated and deleted data, and change of data types for a particular field based on the schema available at the source. Last advance is to stack the perfect information into the cloud-based warehouse.

Before, the information pipeline administrations depended on Extract-Transform-Load (ETL), where information was extricated from the source, changed by the analytical query requirements and stacked into the warehouse. This system was incorporated because of lack of cost-efficient remote data storage facilities at the time; thus, data was transformed so as to store data that were supposed to be queried for analytics. The transformation included consolidation of data using certain algorithms to generate smaller amount of data that would suffice the query requirements. This method had several limitations such as, raw data cannot be directly queried since it is not available at the warehouse end, subject matter experts helped design algorithms and queries that can combine data and extract information enough for analytics from smaller extent of data. With onset of cost-effective cloud-based storage services, a new method came to Fig. 2. difference between ETL and ELT.

Existence Extract Load Transform (ELT) where the data is extracted from source, and then loaded into the warehouses for analytical queries.

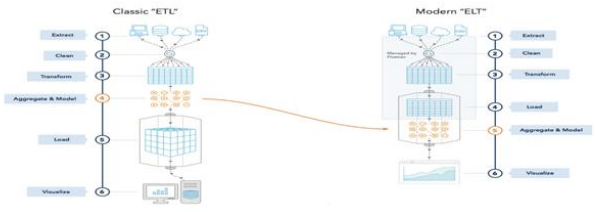

Fig-3: Difference between ETL and ELT

4. Results and Discussion

In an ideal world, data analysts have access to all their required data without concern for where it's stored or how it's processed—analytics just work.

Until recently, the reality of analytics has been much more complicated. Expensive data storage and underpowered data warehouses meant that accessing data involved building and maintaining fragile ETL (Extract, Transform, Load) pipelines that pre-aggregated and filtered data down to a consumable size. ETL software vendors competed on how customizable, and therefore specialized, their data pipelines were.

Technological advances now bring us closer to the analysts' ideal. Practically free cloud data storage and dramatically more powerful modern columnar cloud data warehouses make fragile ETL pipelines a relic of the past. Modern data architecture is ELT—extract and load the raw data into the destination, then transform it post-load. This difference has many benefits, including increased versatility and usability. Read our blog post, The Modern Data Pipeline, to learn more about the difference between ETL and ELT.

Limitations of ETL Overall, the traditional ETL process has three serious and related downsides:

1. Complexity. Data pipelines run on custom code dictated by the specific needs of specific transformations. This means the data engineering team develops highly specialized, sometimes non-transferrable skills for managing its code base.

2. Brittleness. For the aforementioned reasons, a combination of brittleness and complexity makes quick adjustments costly or impossible. Parts of the code base can become nonfunctional with little warning, and new business requirements and use cases require extensive revisions of the code.

3. Inaccessibility. More importantly, ETL is all but inaccessible to smaller organizations without dedicated data engineers. On-premise ETL imposes further infrastructure costs. Smaller organizations may be forced to sample data or conduct manual, ad hoc reporting.

5. CONCLUSIONS

Technological trends know that computation, storage and bandwidth have become cheap and ubiquitous. With advances in computing, the cost of computation has plummeted over time. Likewise, in a span of about 35 years, the cost of a gigabyte has plummeted from nearly $1 million to a matter of cents. One effect of these radical cost reductions is that data warehouses can accommodate much larger volumes of data. Organizations no longer need to pre aggregate and, in the process, discard a great deal of source data. This enables analysts to perform deeper and more comprehensive analysis than ever before. Although the World Wide Web did not exist until 1991, the cost of internet transit has also decreased radically. In less than twenty years, it dropped from about $1,200/ Mpbs to a matter of cents. The convergence of these three cost-reduction trends led to the cloud — namely, the use of remote, decentralized, web-enabled computational resources. Cloud technology, in turn, has given rise to a huge range of cloud-native applications and services.

Many organizations rely on a manual, ad hoc approach to data integration — in fact, 62% use spreadsheets like Excel and Google Sheets to stitch together elements from data files and visualize data. 2 This involves downloading files, manually altering or cleaning values, producing intermediate files, and similar actions.

Ad hoc data integration has a host of drawbacks; specifically, it is:

- Suitable only for very small volumes of data
- Slow
- Prone to human error
- Insufficiently secure for sensitive information
- Often unreproducible

A more sustainable approach is to maintain the silos between separate data sources while bridging the gaps between them with "federated" queries, which directly query multiple source systems and merge data on the fly. Organizations may do this with SQL query engines like Presto. The disadvantage of this federated approach is that it involves many moving parts, and its performance degrades at large scales of data. The reality is that a scalable, sustainable approach to analytics requires a systematic, replicable approach to data integration — a data stack.

REFERENCES

[1] Panos Vassiliadis, 'A Survey of Extract-Transform-Load Technology.,' July 2009 International Journal of Data Warehousing and Mining 5:1-27

[2] Kamal Kakish, Theresa A Kraft, 'ETL Evolution for Real-Time Data Warehousing', presented at Conference: 2012 Proceedings of the Conference on Information Systems Applied Research, At New Orleans Louisiana, USA

[3] Florian Waa, Tobias Freudenreich, Robert Wrembel, Maik Thiele, Christian Koncilia, Pedro Furtado, 'On-Demand ELT Architecture for Right-Time BI: Extending the Vision', International Journal of Data Warehousing and Mining 9(2):21-38 · April 2013

[4] Michael J. Denney, MA,1 Dustin M. Long, PhD,2 Matthew G. Armistead, BS,1 Jamie L. Anderson, RHIT, CHTS-IM,3 and Baqiyyah N. Conway, PhD4, 'Validating the Extract, Transform, Load Process Used to Populate a Large Clinical Research Database, 'Int. J. Med. Inform., 94 (2016), pp. 271-274

[5] Valerio Persico, Antonio Montieri, Antonio Pescapè, 'On the Network Performance of Amazon S3 Cloud-Storage Service', 2016 5th IEEE International Conference on Cloud Networking (Cloudnet)

[6] Pwint Phyu Khine, Zhao Shun Wang, 'Data Lake: A New Ideology in Big Data Era', 2017 4th International Conference on Wireless Communication and Sensor Network [WCSN2017], At Wuhan, China

[7] Benjamin S. Baumer, 'A Grammar for Reproducible and Painless Extract-Transform-Load Operations on Medium Data', arXiv:1708.07073v3 [stat.CO] 23 May 2018

[8] Ibrahim Burak Ozyurt and Jeffrey S Grethe, 'Foundry: a message-oriented, horizontally scalable ETL system for scientific data integration and enhancement', Database (Oxford). 2018; 2018: bay130.

[9] FabianPrasser, HelmutSpengler, RaffaelBild, JohannaEicher, Klaus A.Kuhn, 'Privacy-enhancing ETL-processes for biomedical data', International Journal of Medical Informatics, Volume 126, June 2019, Pages 72-81

[10] Gustavo V. Machado, Ítalo Cunha, Adriano C. M. Pereira, Leonardo B. Oliveira , 'DOD-ETL: distributed on-demand ETL for near real-time business intelligence ', Journal of Internet Services and Applications volume 10, Article number: 21 (2019)

[11] Noussair Fikri, Mohamed Rida, Noureddine Abghour, Khalid Moussaid & Amina El Omri, 'An adaptive and real-time based architecture for financial data integration', Journal of Big Data volume 6, Article number: 97 (2019)