ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2158

The Data Lakes: A Leap Forward Future of Data Warehousing

Bhushan Fadnis
Independent Researcher, USA

Abstract:- With the rise of data and technological advancements, organizations are more interested than ever in exploring infinite data. As data grows, there are no limits to what we can analyze and derive from it. An organization needs a central data repository that should be one trustworthy source. A data lake will benefit any company by helping it make data-driven decisions and identify the right business strategy. Unlike data warehouses built for specific use cases, a data lake can be built for broader use cases addressing current or future business rising needs. Data Lakes are a steppingstone in the data exploration journey, and they have come a long way from traditional databases and data warehouses. This research paper will describe the data lake architecture, functionality, and ways to build it. To build a lake, this paper will examine Amazon Web Services (AWS) and the various tools it provides for this case. Every organization today should consider data lakes strongly and consider their advantages.

Keywords:- *Data Lakes, Data Warehouse, Database, Analytics.*

I. INTRODUCTION

The data lake is a centralized storage of raw data in structured, semi-structured, and unstructured formats [1]. Structured data consists of a relational database, semi-structured data contains CSV and JSON files, and unstructured data includes images, video, and PDF. An organization can bring such data from various systems into a single place to perform rigorous analytics and derive valuable insights for data reporting, visualization, and machine learning. The data lakes have been proven beneficial to organizations as they gain a competitive advantage by learning from data insights and acting on growth opportunities. The data lakes can be on-premised based on the Company's infrastructure or hosted in the Cloud by AWS, Google, or Microsoft. These cloud systems will store Terabytes or Petabytes of data in object storage, which are cheap, effective, uniquely identified, and accessible across multiple regions.

II. RELATED RESEARCH

The data lake has been an intriguing topic for data practitioners as the use cases and how we understand data have evolved. The research below has been conducted on the data lake.

- *Data Lake: A New Ideology in the Big Data Era*

This research [2] is focused on the overall concept of a data lake and architecture approach. The concept refers to including all the source data from various source systems in different formats. The architecture elaborates on new technology such as Apache Hadoop (Highly Available Object-Oriented Data Platform), which is divided into two main components- HDFS (Hadoop Distributed File System) and Map Reduce. HDFS takes care of a single point of contact and scalability, and Map Reduce stores data in data block format with key-value pairs.

- *Data lakes in business intelligence: reporting from the trenches*

The above research [3] does an exploratory study on understanding the data lake based on 12 interviews with the data practitioners, and it concluded that the data lake is not a replacement for the data warehouse and should be considered an extension. It further adds that the data lake could be used as a staging area for the data warehouse. The inherent business uses could differ for data warehouses and data lakes, where a data warehouse is more specific to the business needs, and a data lake could be open-ended with evolving requirements.

- *An Overview of Data Warehouse and Data Lake in Modern Enterprise Data Management*

This study [4] discusses various architectural factors such as governance, metadata, stewardship, orchestration, and ETL layer. The design aspects also consider data modeling, on-premises vs. cloud models, and ETL vs. ELT design. It provided several other tools that can be used to build the data lake, such as AWS, Google, and Azure.

III. DATA WAREHOUSE VS DATA LAKE

As we can see, data warehouses (DWH) have been running in industries for decades and have satisfied all different business cases in the past. So, we need to understand what makes the data lake different and how it can be used as described by AWS [5].

- Data—A Data warehouse consists of a relational database that stores all structured data and a data lake is built for structured, semi-structured, and unstructured data.
- Schema—The schema is pre-defined in the data warehouse due to the table and database structure; in the data lake, the schema is flexible and dynamic, so it is a schema on-write.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2158

- Performance—Query response is faster due to table and data lake storage, and compute is separated, so there is more query response time.
- Business Case—DWH is built for a pre-identified use case that supports business intelligence and visualization; however, data lakes are used for analytics, machine learning, data patterns and trends identification, and historical reporting.
- Data Quality – High-quality, clean data in DWH and raw, crude data in data lakes.

IV. DATALAKE ON AWS

There are various ways to build a data lake, such as On-Premises vs. Cloud. For the on-premises service, all the technical development falls under the Company's internal software development team, which requires vast hours of planning, development, testing, and execution and can increase the speed to market. On the other hand, cloud services offer many fully automated services that users can use from day 1, resulting in quick delivery with expert support from the Cloud. Amazon Web Service (AWS) is one of the cloud providers that provides a variety of services and tools for building data lakes [6], such as follows:

➢ *AWS Lake Formation*

Lake Formation is a fully managed AWS service that ingests data from multiple sources, catalogs it, manages permissions, and makes it available for future processing in EMR, Athena, or Redshift. It also secures and governs data for access control and permission management. Lake formation provides detailed security, allowing column, row, and cell-level granularity for access control. The architecture is shown below.

Fig 1 AWS Lake Formation Architecture

- Data Ingestion—The Lake formation can ingest data from various sources. As shown, it is getting data from S3, an object-level structure with various file types and formats, relational databases, or NoSQL data sets. It will first fetch the metadata to understand the schema and then bring the data, supporting bulk and incremental loading. For all external databases such as PostgreSQL, MySQL, or MS SQL Server, a Java Database Connectivity (JDBC) will be used.
- Data Catalog—AWS Glue pulls the table's metadata. The catalog folder will have all the databases listed and tables underneath, with all columns and data types identified. We can label, share, and mainly use this catalog to control column—or row-level data access.
- Security Management—It allows permission to access row, column, and cell-level data and hides sensitive data from broad access. These permissions and policies are part of IAM (Identity and Access Management) systems and flow through all the tools that use Lake Formation data. For instance, data accessed in Redshift or Athena will only allow users to see data assigned in Lake Formation security management. Audit trails are also present, which log the access, changes to access, and history in the cloud trail service. In addition, a tagging mechanism is present to tag specific changes or policies for text-based search and recording of the event.
- Data Sharing—This feature does not require data transfer. Instead, it sets up permissions for other data storage, such as S3, Redshift, or AWS Data Exchange, allowing it to manage and share resources from other organizations.
- Permission Management—Whenever users need to query S3 files, they can use the Athena tool, which follows the steps below.
- Get Metadata – When a user queries, the analytical engine identifies the requested table and sends the metadata request to the Data Catalog.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2158

- Check Permissions – The Data Catalog will check the user's permission and return the metadata if the permissions are granted.
- Get Credentials – Temporary access is granted if the requested table is registered in Lake Formation.
- Get Data – The analytical engine fetches the data from S3 with a filtered view per permissions and presents it to the user.
- If the table is not part of the Lake Formation catalog, user data is retrieved based on the S3 IAM permissions setup.

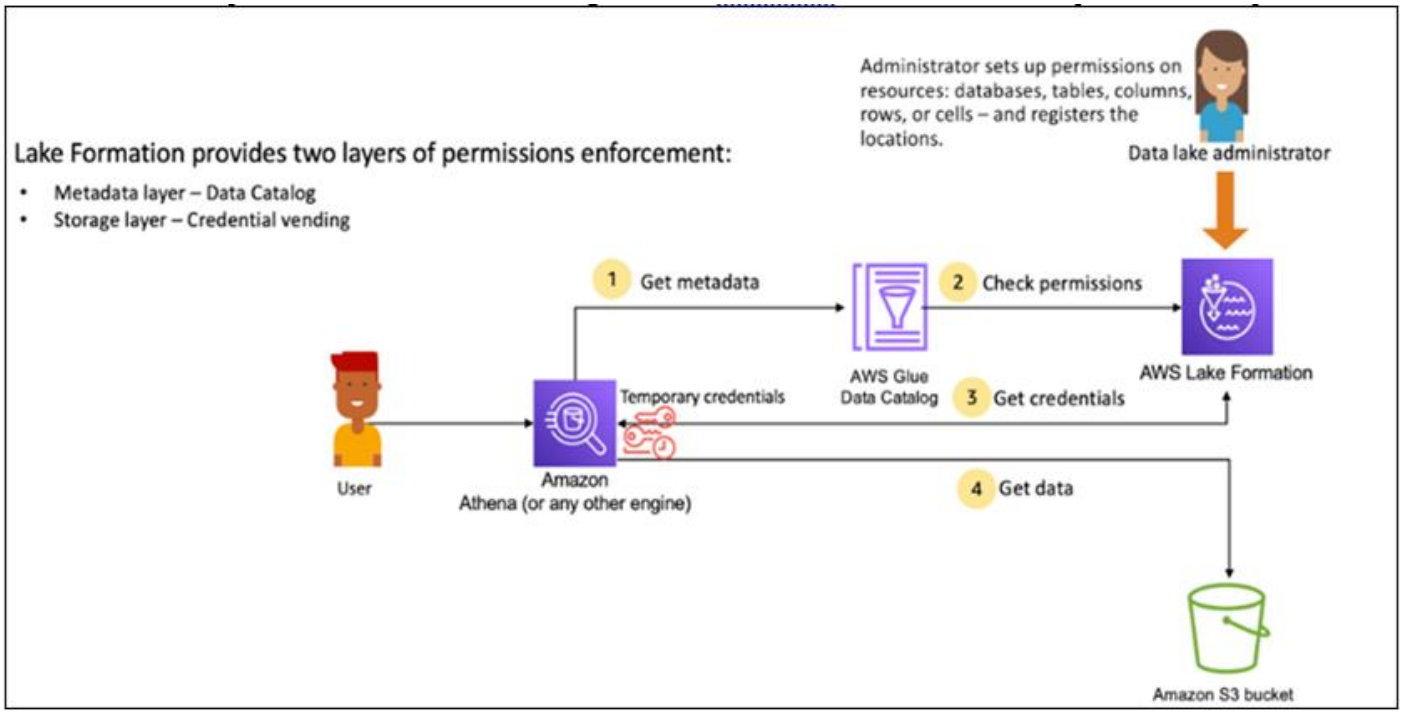

Fig 2 Permission Flow AWS Lake Formation

➢ *AWS S3 (Simple Storage Service)*

S3 is highly scalable object-level storage that offers data availability, security, and performance. It's low-cost and technically unlimited in data storage capacity. S3 is the main storage for AWS data lakes. It provides storage monitoring using CloudWatch and CloudTrail, and its storage lens will monitor the S3 files for usage and access and recommend cost-effective and optimization policies. S3 comes in different types based on the requirements such as S3 Intelligent-Tiering, S3 Standard, S3 Express One Zone, S3 Standard-Infrequent Access (S3 Standard-IA), S3 One Zone-Infrequent Access (S3 One Zone-IA), S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval, S3 Glacier Deep Archive, and S3 Outposts. To perform high performance, S3 supposed 3500 requests per second to add data and 5500 requests per second to retrieve data. For consistency, S3 provided read as soon as after writing by default on all objects. Even though its file storage, the S3 Select feature allows queries in place on S3 files instead of loading those into other AWS tools or databases. S3 supports data transfer using a storage gateway and Data sync and Data exchange using AWS Data Exchange service for S3.

➢ *AWS Redshift*

Redshift is a data warehouse tool that can input data from S3 using ETL jobs. It is a serverless, fully managed, highly available, and scalable service that uses SQL to analyze structured and semi-structured data. It is based on massively parallel processing to handle large data sets, and it separates storage and compute, so they are de-coupled. Each can achieve scalability on demand based on the data and request load. To achieve high availability, it is deployed in multiple regions, and data gets replicated on a time basis and serves backup and disaster poverty purposes. Redshift supports end-to-end data encryption using AES-256 encryption for the data at rest. Advanced level data making is supported to protect sensitive data from wider exposure. To achieve quicker query retrieval, Redshift uses a caching mechanism that stores the result of commonly run frequent queries. The data is stored in columnar storage format, which is effective in performing aggregated queries for analytics instead of row-level storage.

➢ *AWS Glue*

- Glue is a fully managed serverless Extract Transform and Load (ETL) tool that quickly discovers, prepares, moves, and integrates data from multiple sources and loads into S3 or Redshift for future analytical processing. AWS Glue has a crawler that will connect with any Database using a JDBC connection to fetch the schema and prepare a data catalog. Lake Formation will use this catalog to grant users permission and access.
- Glue is a no-code ETL that connects the source data and loads the data into S3 files based on the preferences set. However, it also allows customizing the code in Python or Spark for additional transformations to add the business logic.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2158

- Glue jobs can be scheduled or event-driven based on the S3 new file landing. The glue will dump all job logs to the cloud watch, and its workflow can be set up with AWS workflows. Based on preference, they can be set up to load the data into S3 and Redshift by using AWS BOTO3 libraries to connect with all resources. Glue supports Python, Spark, and Scala languages and all external libraries.

➢ *AWS Athena*

Athena is another serverless, highly scalable, interactive analytics platform that supports querying multiple file formats, S3 files, or open tables. It is built on Trino and Preto open-source technologies and doesn't need to be set up, as no provisioning and configuration are required. Athena will allow users to query the S3 files and analyze the data; it will also be used internally when S3 data needs to be fetched into another tool, such as visualizations. The customer doesn't need to manage any servers or infrastructure for Athena. Cluster tuning or hardware optimization is needed as it already runs parallel to achieve quick results. It just needs to be activated at the account level, and the costing is done based on the amount of data scanned and the number of seconds queries run internally in AWS. Athena supports a federated query approach, which connects more than 30+ AWS data sources to join tables and generate output.

➢ *File Formats*

Choosing the correct S3 file format is important, depending on the data use case. Below are a few popular file formats that can be used.

- JSON is a text-based, easy-to-read format with key-value data formats.
- Avro is suitable for real-time streaming with data and schema stored together and for serialization.
- Parquet is a columnar format that is highly effective in aggregating and querying vast data.

➢ *Data Compression*

In addition to file format, compression also plays a key role in file processing; it reduces the size of bigger files, saves space, and increases data retrieval time. The most common types of compression are as follows:

- Bzip2- a Burrows-Wheeler Transform and Huffman coding algorithm, achieves a good compression ratio but requires more time and resources.
- The Gzip—DEFLATE algorithm compresses files with a well-balanced speed and compression ratio and is compatible with multiple systems.
- Xz - LZMA2 algorithm achieves the highest compression but takes the most time.

➢ *File Partition*

The file partition is used for quicker data retrieval as it acts similarly to a table partition from the RDBMS. The correct file partition should be the field we use regularly to query this data in SQL queries. For instance, if the user queries based on the account number and the account number is a limited set of numbers, then it's a good idea to partition by account number. If the user wants to perform daily analysis and see day-over-day patterns, partitioning by day makes the most sense. Once the file partition is finalized, the Glue script must be updated so that every file-writing operation follows the partitioning pattern.

V. DATA SECURITY, GOVERNANCE, PII

The data in the lake should be secured, governed, and avoided having PII (Personally Identifiable Information). Securing data should be the highest priority in today's world to stop data breaches. Lake formation policy controls will hide the PII-related columns from the lake, so any user querying the data will not find any PII data. Additionally, all the data in S3 is encrypted with AWS or customer-provided keys, so any unauthorized access cannot read data without proper keys. All global-level permissions in AWS are achieved through the Identity and Access Management (IAM) service, which works on the least access approach. IAM allows users, groups, and service users to be created and tied to policies for reading, writing, updating, etc. The policies are very granular and should vary from use case to use case. Apart from that, access controls at the Lake Formation level will further control data based on different criteria for security, PII, and masking purposes. If lake formation policies are not set up, then by default, users get access to data based on their original IAM policy setup.

➢ *Challenges*

Though data lakes are effective, they also have several challenges, some noted below.

- Expensive to build for the organization due to the plethora of services required, both on-premises and cloud.
- Long development process to follow as this will be a big initiative for any company and requires significant development and testing time from multiple teams.
- Due to multiple new technical tools and deployments, it could be overwhelming for software development teams.

VI. CONCLUSION

The data lakes have many business and technical advantages if deployed correctly within strict timelines and budgets. Every data-driven organization should consider building lakes and running their analytics, data reporting, and machine learning algorithms. Unlike data warehouses, while building data lakes, the business doesn't need a use case ready; they can build the lake and always use the data for analytical use cases coming up in the future.

➢ *Author*

Bhushan Fadnis received an MS in Information Science from San Diego State University, USA, in 2017. He has more than 12+ years of technology experience working in various MNCs and is now a Business Intelligence Engineer in a leading software company in the USA.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2158

REFERENCES

[1]. What is a data lake? - introduction to Data Lakes and analytics - AWS. (n.d.). https://aws.amazon.com/what-is/data-lake/

[2]. Khine, P. P., & Wang, Z. S. (2018, February 2). *Data lake: A new ideology in Big Data Era*. ITM Web of Conferences. https://doi.org/10.1051/itmconf/20181703025

[3]. Llave, M. R. (2018). Data lakes in business intelligence: Reporting from the trenches. *Procedia Computer Science*, *138*, 516-524. https://doi.org/10.1016/j.procs.2018.10.071

[4]. Nambiar A, Mundra D. An Overview of Data Warehouse and Data Lake in Modern Enterprise Data Management. *Big Data and Cognitive Computing*. 2022; 6(4):132. https://doi.org/10.3390/bdcc6040132

[5]. Data Lake vs Data Warehouse vs Data Mart - difference between Cloud Storage Solutions - AWS. (n.d.-a). https://aws.amazon.com/compare/the-difference-between-a-data-warehouse-data-lake-and-data-mart/

[6]. Secure data lake - aws lake formation - AWS. (n.d.-c). https://aws.amazon.com/lake-formation/