**CONTENTS**

**S**L.NO Particulars

1. Introduction
2. ETL
3. Data Warehouse
4. AWS
5. ML PaaS
6. Python
7. Code
8. Snapshots
9. Conclusion
10. **INTRODUCTION**

Dynamic ETL on ML PaaS

1. **ETL**

Extract, Transform and Load (ETL) is a process in database usage and especially in data warehousing that involves:

* Extracting data from outside sources.
* Transforming it to fit operational needs (which can include quality levels).
* Loading it into the end target (database or data warehouse).
  1. **ETL Tools**

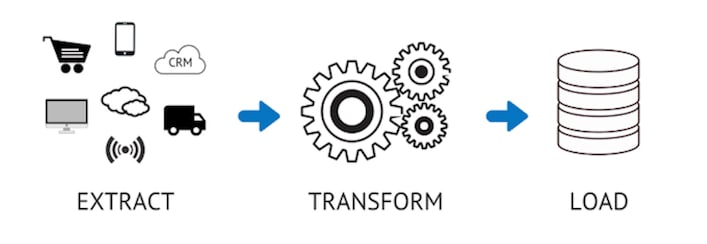
ETL tools enable data integration strategies by allowing companies to gather data from multiple data sources and consolidate it into a single, centralized location. ETL tools also make it possible for different types of data to work together.

ETL tools also makes it possible to migrate data between a variety of sources, destinations, and analysis tools. As a result, the ETL process plays a critical role in producing business intelligence and executing broader data management strategies. We are also seeing the process of  [Reverse ETL](https://www.talend.com/resources/reverse-etl/) become more common, where cleaned and transformed data is sent from the data warehouse back into the business application.

**2.2 How ETL Works**

The ETL process is comprised of 3 steps that enable data integration from source to destination: [data extraction](https://www.talend.com/resources/data-extraction-defined/), data transformation, and data loading.

ETL tools automate the extraction process and create a more efficient and reliable workflow.



**Step 1: Extraction**

Most businesses manage data from a variety of data sources and use a number of data analysis tools to produce business intelligence. To execute such a complex data strategy, the data must be able to travel freely between systems and apps.

Before data can be moved to a new destination, it must first be extracted from its source — such as a data warehouse or data lake. In this [first step of the ETL process](https://www.talend.com/resources/data-extraction-defined/), structured and unstructured data is imported and consolidated into a single repository.

Although it can be done manually with a team of data engineers, hand-coded data extraction can be time-intensive and prone to errors.

**Step 2: Transformation**

During this phase of the ETL process, rules and regulations can be applied that ensure data quality and accessibility. You can also apply rules to help your company meet reporting requirements.

Transformation is generally considered to be the most important part of the ETL process. Data transformation improves data integrity — removing duplicates and ensuring that raw data arrives at its new destination fully compatible and ready to use.

**Step 3: Loading**

The final step in the ETL process is to load the newly transformed data into a new destination (data lake or data warehouse.) Data can be loaded all at once (full load) or at scheduled intervals (incremental load).

* 1. **ETL BENEFITS**

1. Visual flow

The single greatest advantage of an ETL tool is that it provides a visual flow of the system’s logic (if the tool is flow based). Each ETL tool presents these flows differently, but even the least-appealing of these ETL tools compare favorably to custom systems consisting of plain SQL, stored procedures, system scripts, and perhaps a handful of other technologies.

1. Structured system design

ETL tools are designed for the specific problem of data integration: [populating a data warehouse](https://www.passionned.com/data-integration/data-warehouse/) or integrating data from multiple sources, or even just moving the data. With maintainability and extensibility in mind, they provide, in many cases, a metadata-driven structure to the developers. This is a particularly big advantage for teams building their first data warehouse.

1. Operational resilience

Many of the home-grown data warehouses we evaluated are rather fragile: they have many emergent operational problems. ETL tools provide functionality and standards for operating and monitoring the system in production. It’s certainly possible to design and build a well-instrumented, hand-coded ETL application. Nonetheless, it’s easier for a data warehouse / business intelligence team to build on the features of an ETL tool to build a resilient ETL system.

1. Data-lineage and impact analysis

We would like to be able to right-click on a number in a report and see exactly how it was calculated, where the data was stored in the data warehouse, how it was transformed, when the data was most recently refreshed, and from what source system(s) the numbers were extracted. Impact analysis is the flip side of lineage: we’d like to look at a table or column in the source system and know which ETL procedures, tables, cubes, and user reports might be affected if a structural change is needed.

1. Advanced data profiling and cleansing

Most data warehouses are structurally complex, with many data sources and targets. At the same time, requirements for transformation are often fairly simple, consisting primarily of lookups and substitutions. If you have a complex transformation requirement, for example if you need to de-duplicate your customer list, you should buy on additional module on top of the ETL solution (data profiling / data cleansing). At the very least, ETL tools provide a richer set of cleansing functions than those available in SQL

1. Performance

You might be surprised that performance is listed as one of the last under the advantages of the ETL tools. It’s possible to build a high-performance data warehouse whether you use an ETL tool or not. It’s also possible to build an absolute dog of a data warehouse whether you use an ETL tool or not. But the structure imposed by an ETL platform makes it easier for a (novice) ETL developer to build a high-quality system

1. Big Data

A lot of ETL tools are now capable of combining structured data with unstructured data in one mapping. In addition, they can handle very large amounts of data that don’t necessarily have to be stored in data warehouses. Nowadays, Hadoop-connectors, or similar interfaces to big data sources, are provided by almost 40% of the ETL tools. And the support for Big Data is growing continually.

* 1. **Uses of ETL**

ETL is commonly used to do the following:

* Data warehousing
* Machine learning and artificial intelligence
* Marketing data integration
* IoT data integration
* Database replication
* Cloud migration
  1. **Advantages and Disadvantages**

Advantages of ETL:

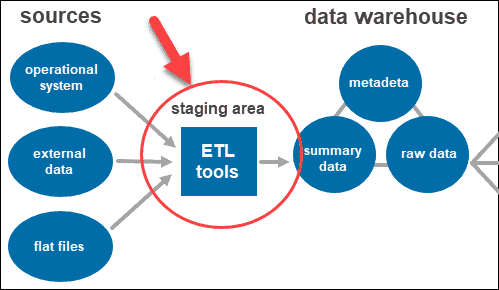
* Good for bulk data movements with complex rules and transformations.
* Makes maintenance and traceability much easier than hand-coding.
* Good for data warehouse environment.

Disadvantages of ETL:

* You must be data-oriented developer or database analyst to use
* Not ideal for near real-time or on-demand data access, where fast response is required
* Difficult to keep up with changing requirements.

1. **Data Warehouse**

A data warehouse is a type of data management system that is designed to enable and support business intelligence (BI) activities, especially analytics. Data warehouse are solely intended to perform queries and analysis and often contain large amounts of historical data.



1. **AWS**

AWS stands for Amazon Web Services.

The AWS service is provided by the Amazon that uses distributed IT infrastructure to provide different IT resources available on demand. It provides different services such as infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS).

Amazon launched AWS, a cloud computing platform to allow the different organizations to take advantage of reliable IT infrastructure.

* 1. **AWS Services**

There are three main cloud service models. Each model represents a different part of the cloud computing stack and gives you a different level of control over your IT resources:

• Infrastructure as a service (IaaS): Services in this category are the basic building blocks for cloud IT and typically provide you with access to networking features, computers (virtual or on dedicated hardware), and data storage space. IaaS provides you with the highest level of flexibility and management control over your IT resources. It is the most similar to existing IT resources that many IT departments and developers are familiar with today

.

• Platform as a service (PaaS): Services in this category reduce the need for you to manage the underlying infrastructure (usually hardware and operating systems) and enable you to focus on the deployment and management of your applications.

• Software as a service (SaaS): Services in this category provide you with a completed product that the service provider runs and manages. In most cases, software as a service refers to end-user applications. With a SaaS offering, you do not have to think about how the service is maintained or how the underlying infrastructure is managed. You need to think only about how you plan to use that particular piece of software. A common

example of a SaaS application is web-based email, where you can send and receive email without managing feature additions to the email product or maintaining the servers and operating systems that the email program runs on.



* 1. **Uses of AWS**
* A small manufacturing organization uses their expertise to expand their business by leaving their IT management to the AWS.
* A large enterprise spread across the globe can utilize the AWS to deliver the training to the distributed workforce.
* An architecture consulting company can use AWS to get the high-compute rendering of construction prototype.
* A media company can use the AWS to provide different types of content such as ebox or audio files to the worldwide files.

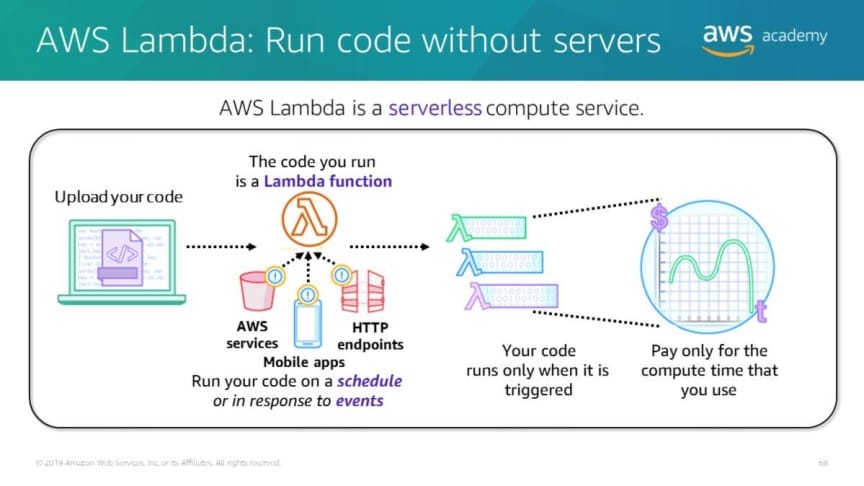
## **4.3 Pay-As-You-Go**

AWS provides services to customers when required without any prior commitment or upfront investment. Pay-As-You-Go enables the customers to procure services from AWS.

* Computing
* Programming models
* Database storage
* Networking

**4.4 AWS Lambda**

AWS Lambda is a serverless computing service that runs us code in response to events and automatically manages the underlying computing resources for us. We can use AWS Lambda to extend other AWS services with custom logic or create our back-end services. AWS Lambda can automatically run code in response to multiple events, such as HTTP requests via Amazon API Gateway, modifications to objects in Amazon S3 buckets, table updates in Amazon DynamoDB. In our case, it will run the code according to schedule.



**4.5 AWS S3 Bucket**

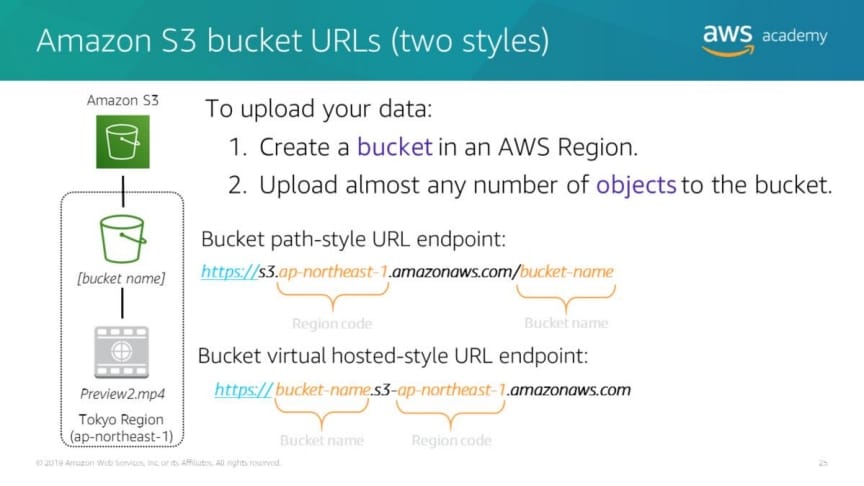
Amazon S3 is object-level storage, which means that if you want to change a part of a file, you must make the change and then re-upload the entire modified file. Amazon S3 stores data as objects within resources that are called buckets.



Amazon S3 includes event notifications that enable you to set up automatic notifications when certain events occur, such as when an object is uploaded to a bucket or deleted from a specific bucket. Those notifications can be sent to you, or they can be used to trigger other processes, such as AWS Lambda functions.

With storage class analysis, you can analyze storage access patterns and transition the right data to the right storage class. The Amazon S3 Analytics feature automatically identifies the optimal lifecycle policy to transition less frequently accessed storage to Amazon S3 Standard – Infrequent Access (Amazon S3 Standard-IA). You can configure a storage class analysis policy to monitor an entire bucket, a prefix, or an object tag.

When an infrequent access pattern is observed, you can easily create a new lifecycle age policy that is based on the results. Storage class analysis also provides daily visualizations of your storage usage in the AWS Management Console. You can export them to an Amazon S3 bucket to analyze by using the business intelligence (BI) tools of your choice, such as Amazon Quick Sight.



To use Amazon S3 effectively, you must understand a few simple concepts. First, Amazon S3 stores data inside buckets. Buckets are essentially the prefix for a set of files, and must be uniquely named across all of Amazon S3 globally. Buckets are logical containers for objects.

You can have one or more buckets in your account. You can control access for each bucket—who can create, delete, and list objects in the bucket. You can also view access logs for the bucket and its objects, and choose the geographical region where Amazon S3 stores the bucket and its contents.

To upload your data (such as photos, videos, or documents), create a bucket in an AWS Region, and then upload almost any number of objects to the bucket.

In the example, Amazon S3 was used to create a bucket in the Tokyo Region, which is identified within AWS formally by its Region code: ap-northeast-1

The URL for a bucket is structured like the examples. You can use two different URL styles to refer to buckets.

Amazon S3 refers to files as objects. As soon as you have a bucket, you can store almost any number of objects inside it. An object is composed of data and any metadata that describes that file, including a URL. To store an object in Amazon S3, you upload the file that you want to store to a bucket.