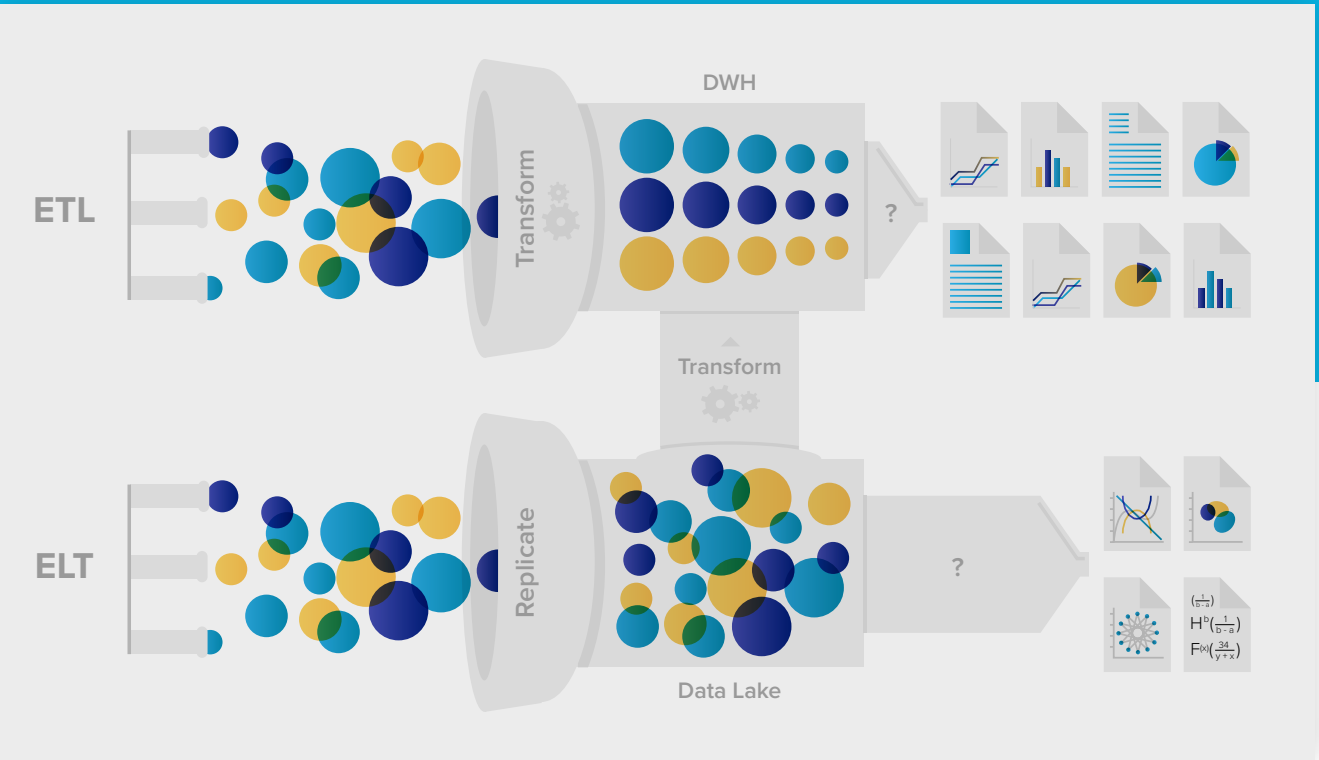
Xplenty

ELT Vs. ETL



INTRODUCTION

For over a decade, the data world has been flooded with new technologies, methodologies and buzzwords to handle the growing amount of data, and leverage it to increase competitive advantage and ROI based on it.

One of the ongoing debates in the field is centered around the following question: “Which is better: ETL or ELT?”

In this paper, we’ll break down the concepts of ETL and ELT, explore whether data transformation should occur inside or outside the target database, and learn how all this relates to Data Warehouses and Data Lakes. We’ll also compare different scenarios and provide examples in which each technology is the most suitable to apply.

01 THE DATA CHALLENGE

Data comes in many shapes, forms, timings and sizes.

STRUCTURED AND UNSTRUCTURED

- Structured - Columns are clearly separated (for example by a comma) and the type of data in each column is defined
- Semi-structured - JSON and XML
- Unstructured - Video, audio or simply meshed-up long strings

REAL TIME, BATCH, AND EVERYTHING IN-BETWEEN

- Real Time or Streaming – In real time, once an event happens, the data flow runs immediately on that event and/or on all events as they happen. This is usually required for products that compete against other automated systems - for example price adjustments in online retail or online bidding used to purchase online advertising real estate.
- Batch - Batch loading is based on accumulated data being processed as a single unit at pre-determined intervals.
- Near Real Time - This term is more vague and generally describes a batc data flow that runs with low latency to meet a specific business case or organizational demand.
- Hourly and Daily - The traditional method of loading data (and still quite adequate for most data needs), this method involves all data events in a given day or hour being updated throughout the entire data architecture.

DATA SIZE

While there are many definitions and interpretations of the term Big data, for the purpose of clarity, we will define Big Data vs Data in this whitepaper as follows:

- Data – The amount of data loaded to the database is smaller than the disk capacity dedicated to the database.
- Big Data - The amount of data collected is larger than the disk capacity dedicated to the database, and cannot store the necessary amount of data required for analytics.

02 DATA WAREHOUSE AND DATA LAKE DEFINED

DATA WAREHOUSE

A Data Warehouse (DWH) resides in a database. There are many types of databases and brands - from Redshift and BigQuery to MySQL and postgres.

A data warehouse is a central repository of information gathered from one or more data sources. Using data warehouses, it is possible to govern data and run fast analytics on large volumes of data – uncovering hidden patterns in data leveraging BI tools.

DWHs store current and historical data and are used for creating analytical reports for data consumers throughout the organization. Examples of reports could range from annual finance reports to hourly trends of sales analysis.

DATA LAKE

“Data Lake” is a relatively new term that is credited to James Dixon, Pentaho’s CTO. Essentially, a data lake is a large storage repository of raw or slightly-prepared data in its native format.

Data lakes store data in a flat structure, usually as files. Data “in the lake” is associated with a unique ID and tagged with metadata. When a business question arises, the data lake can be queried for relevant data, and that smaller set of data can then be analyzed to help answer the question. Hadoop, Google cloud storage, Azure Storage and the Amazon S3 platform can be used to build data lake repositories.

Data lakes do not require much planning— typically, there is no schema or ETL process in place. Thanks to declining cost of data storage both on-premise and in the cloud, and the abundance of virtual instances, a data lake can be set up quickly. Even before anyone knows what questions they want to ask, data can be poured into the lake from different sources and in several formats right away.

However, since data lakes contain a wide variety of data formats and huge volumes of data, querying them is much more difficult. Traditional BI tools do not yet support data lakes, often requiring coding to generate insights

from the data. This makes the organization's data lake a playground for people with advanced data skills - data scientists and experienced developers – but less accessible to business users.

Luckily, a data lake does not need to stand on its own in the wild. Its data can be organized by running an ETL process, storing the processed data back in the lake or in a data warehouse—a win–win for all data consumers.

03 ETL AND ELT DEFINED

ETL

(EXTRACT, TRANSFORM, LOAD)

- Normally a continuous, ongoing process with a well-defined workflow.
- During this process, data is initially extracted from one or more sources. Then, the data is cleansed, enriched, transformed and ultimately stored into a data warehouse.

ELT

(EXTRACT, LOAD, TRANSFORM)

- A variant of ETL wherein the extracted data is first loaded into the target system.
- Transformations are performed after the data is loaded into the data warehouse.
- ELT typically works well when the target system is powerful enough to handle transformations. Analytical databases like Amazon Redshift and Google BigQuery are often used in ELT pipelines because they are highly efficient in performing transformations.

04 WHEN SHOULD WHICH TECHNOLOGY BE USED?

WHEN SHOULD I USE A DATA LAKE?

When data needs to be collected immediately, and there is no time for planning.

- When data sources and formats are highly dynamic.
- When data is too large to store on a database due to budget considerations.
- When analytical queries are not known in advance, change frequently, or need to be asked ad hoc.
- When data experts need a playground to find and develop new insights.
- When more people in the organization require access to the data.

WHICH PLATFORMS CAN I USE TO IMPLEMENT A DATA LAKE?

- Hadoop Distributed File System (HDFS)
- Amazon Simple Storage Service (S3)Google cloud storage, Azure data lake store
- Can also be combined with a data warehouse, HBase or a NoSQL database like MongoDB

WHEN SHOULD I USE A DWH?

- When data sources are relatively constant
- When data queries are mostly known in advance
- When data can be modelled into a schema structure
- When high level of data accuracy is required, e.g. for accounting
- When tight access control and high level of security is required

WHICH PLATFORMS CAN I USE TO IMPLEMENT A DWH?

- RDBMS: MySQL, Oracle, SQL Server, postgres, etc.Amazon Simple Storage Service (S3)
- Columnar databases: Vertica, ParAccel, Amazon Redshift, Google BigQuery

05

KEY QUESTIONS

REAL TIME OR BATCH?

Data needs to leave and arrive in the data lake in real time. In this scenario there is a need to move away from a batch-oriented approach to real-time or streaming. This can be achieved by utilising technologies like Amazon Kinesis Firehose, that allows processing of real time data straight into the DWH or data lake storage.

STRUCTURED OR UNSTRUCTURED?

Data needs to be structured for several reasons:

- Reliability - Data needs to be trusted. For this to happen, we need to be able to understand it. Structuring the data is the process of defining how the data looks and making it reliably understandable.
- Only after this step can machine learning be trusted to run on unstructured realtime data.
- Costs – Structuring data means that only the required parts are saved. By way of example, an entire original URL is unnecessary: better to dissolve it into domain, page and parameters used. Moreover, these parameters could be numbers - and the cost of storing numbers is significantly lower than storing strings.
- Modeling - Structuring data is critical to bridge the gap between business questions and how things actually work under the hood. Without structuring, blind spots in the product or its reporting capabilities will either remain hidden or be discovered too late.

The process of structuring data is related mainly to the ‘T’ in ETL and ELT. In each approach, whether using a DWH or data lake, transformations are a pivotal ingredient to achieving meaningful insights and simply cannot be skipped

.**WHAT ABOUT DATA VOLATILITY?**

In some cases the data should not be structured due to its high volatility. Ever-changing data is a counter-case to structuring data. This is relevant when:

- Data sources change frequently - If data is collected from the web or online repositories or different partners like advertisement and marketing companies, there will be a need to change the data sources and their content frequently - even on a daily basis.
- Business questions cannot be defined in advance - For example, if there is a need to evaluate user experience in a game or on a multi-page web site, the questions asked will evolve based on previous questions, with a very high level of complexity in binding the data together.

In such cases, a data lake with transformations on-the-fly is the recommended solution. However, in most cases, including the ones described above, there are still core business questions that can be defined and modeled into structured data. These questions are crucial and can be used as pillars on which free data exploration can be based.

SUMMARY

Data lakes and DWHs, ETL and ELT are concepts of data design architecture. Whichever data strategy is chosen, the work of transforming the data is not done automatically. The transformations can be done inside the DWH as classic SQL queries, or using Hadoop to parse and structure data. In all these cases, transformation takes place at a different stage.

Based on an open architecture, the data lake has the potential to analyze anything, whereas DWH is governed and modeled – meaning that any new type of data or business logic has to go through modeling and development.

- Data lake has nearly unlimited potential but requires transformations before achieving insights.
- DWH requires significant investment in advance, but in return delivers the ability to easily analyze everything, delivering valuable insights quickly and reliably, without the need for additional transformations.

SOME FINAL DATA LAKE VS. DWH THOUGHTS:

- A data lake is open to translate the data in different ways
- DWH provides a single version of truth

- A data lake can handle unstructured data; however extracting value from this unfiltered pool of data can be difficult
- DWH stores structured and modeled data that is organized and ready for analysis

- DWH holds only data that is needed for analytics
- A data lake stores ALL data, relevant as well as irrelevant for analytics. Useful data cannot be separated from redundant until it is analyzed.

- A data lake can quickly become a giant data graveyard
- DWH can sometimes become too rigid to handle timely changes

To learn how to do data transformations across solutions with Xplenty click here.