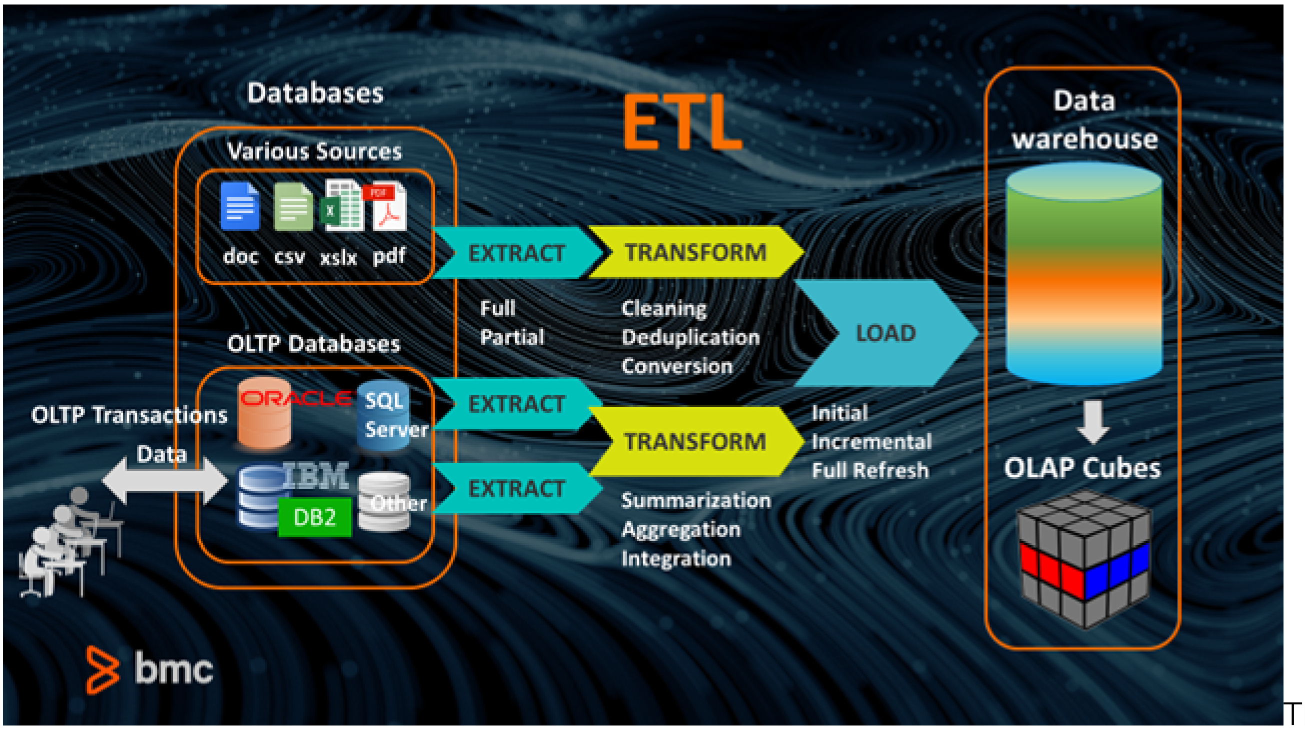
IS ETL (EXTRACT, TRANSFORM, LOAD) STILL RELEVANT?

To explain what ETL (Extract, Transform, Load) is, I'll take you back to the 1970s. For a second, forget about Systems of Engagement, where humans are engaged through social media, online services and cloud, and generate human data in the system, such as Facebook likes, tweets, images, and YouTube statistics. Forget about Internet of Things (IoT), where sensors, cameras, connected cars and other smart devices are linked together in the internet network and generate machine data.

In the 1970s, most data was generated as a result of transactions. Examples of transactional data include payments, sales orders, invoices, booking documents, and insurance claims. If we try to imagine that data, it's about records of people (e.g., customers, employees, vendors, suppliers), places (e.g., locations, sales territories, offices), and things (e.g., accounts, products, assets).

Is there any information content in that data? Even if data and information are often used interchangeably, the reality is that data in a raw format does not bring any information or value as it is.

ETL is the process retrieving information and value out of data. Each of the three phases – Extract, Transform and Load – contributes to that purpose.



The ETL process starts with data **extraction** from various source files (doc, csv, xlsx, pdf) or OLTP (Online Transactional Processing) databases. To understand what an OLTP database is, imagine a simple sales transaction, where the business sells a product or service as soon as it collects cash from the customer. An OLTP application gathers input data (product or service, price, payment, customer), manages the transaction to completion, and stores the data through real-time insertions or updates into an OLTP database (Oracle, SQL Server, IBM DB2 or other).

At that point, is data ready to provide value? No. Extracted data is stored in separate databases and complex tables and may include duplicates. It's difficult to query and analyze.

The next ETL phase is **transformation**. Data is transformed for quick and easy analysis. Transformation may happen in a variety of ways. Data may be cleaned, deduplicated, converted into another format, processed through business rules, filtered, joined, split, validated, enriched, but – most importantly – its summarized, aggregated and integrated.

The last ETL phase is **loading**. Data is loaded into shared storage, typically warehouses, that may contain large data volumes. Finally, ETL processes give last command and move data from data warehouses to OLAP (Online Analytical Processing) cubes. OLAP cubes contain the same data as warehouses, but aggregate and structure it in a special multidimension format that allows users to rapidly submit queries and obtain business reports.

Is ETL still relevant today?

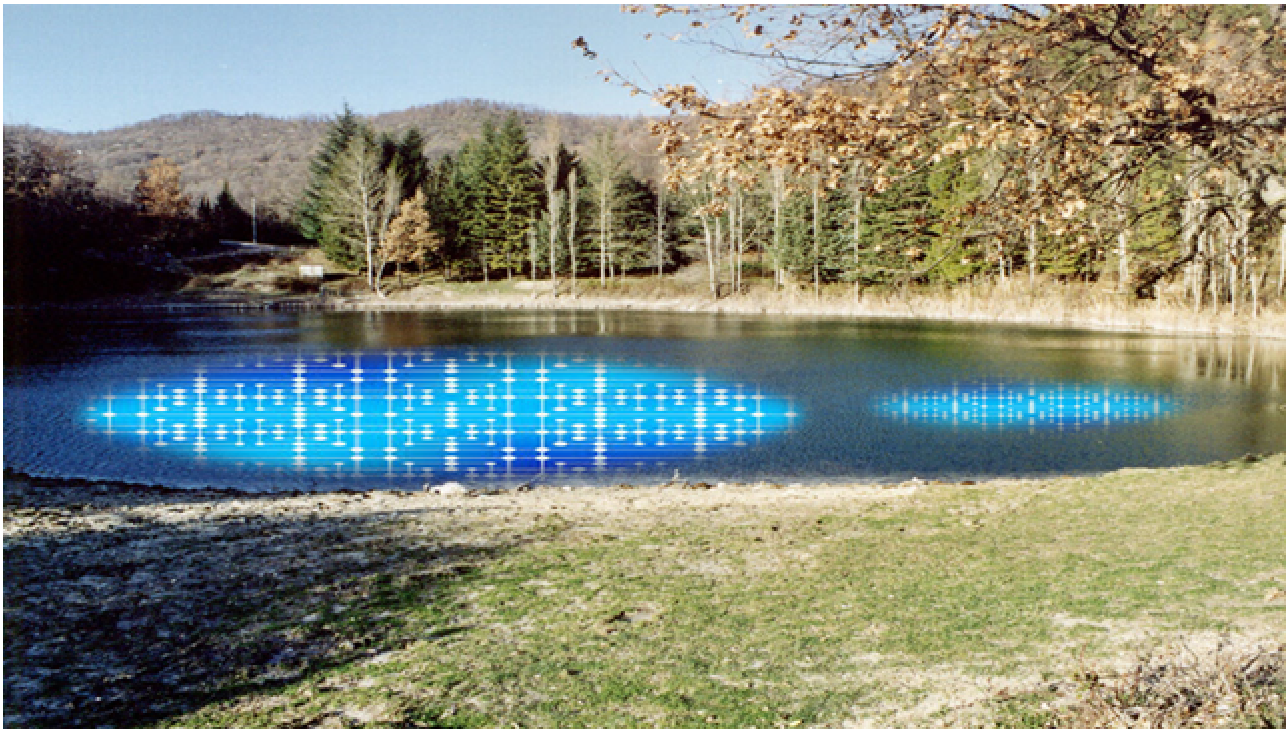
Now, let's get back to 2018. Instead of the structured data residing in OLTP databases or other sources, let's think about the unstructured and volatile big data produced by Systems of Engagement and IoT.

Is ETL still relevant in this context? There's a lot of debate about this.

One key debate point is whether ETL must give way to ELT (Extract, Load, Transform). A simple shift in pipeline phases implies a radical change in the infrastructures, tools, setup, times and resources involved.

While ETL focuses on retaining important data, through business logic, elaborations, decisions, filters and aggregation, and produces a data warehouse ready for easy consumption and business reports, ELT retains all data in its natural state, as it flows from the sources, including both the data that is important today, and the data that might be used someday.

In ELT processes, transformations are performed after the data is loaded into the data warehouse. It works well when the target system is a data lake, rather than a data warehouse. What is a data lake?



Unlike the data warehouse, which is a highly structured data model designed to answer specific questions through reporting, a data lake retains all types of structured, semi-structured, and unstructured data (for example web server logs, sensor data, social network activity, text and images). The data lake structure is better suited for data scientists and analytics developers, capable of defining their own schemas after the data has been loaded ("schema on read") and not before ("schema on write").

Choosing between ETL and ELT depends on multiple considerations. For example:

- What is the nature of my data?
- What's the business value case I want to accomplish?
- Who are the people who need to query my data store?
- What are their skills? What types of queries will they need to perform?
- Which technologies do I have in place or do I plan to deploy?

Control-M in ETL/ELT

Whether you are going with an ETL or an ELT approach, Control-M can help you automate and orchestrate all phases in ETL/ELT, through predictable and ordered workflows, simplifying complex environments. From raw data, to business reports, or data lakes, Control-M drives these processes by automating data transfers and workload execution – applying predictive analytics to prevent job failures; automatically retrying jobs that were interrupted; and presenting user dashboards and self-service capabilities. Business users get the insights they need, in a reliable, consistent and repeatable way. Data scientists can focus on delivering new analysis instead of debugging and fixing upstream problems.

What’s your take? Are you leveraging ETL or ELT processes to manage your data projects? I’d love to hear about your experience.

Want to learn more about Control-M for Big Data?