- Article Traffic: 331/mo
- Traffic Value/mo: $578
- Ranks for: 157 keywords
- Content: 2,500 words

This A+ content piece ranks as one of the most successful pages on our client's website. It's a pure money page that consistently drives leads and revenue.

What Is An ETL Pipeline? Examples & Tools (Guide 2023)

The phrase Extract Transform Load explains most of the philosophy behind it. It picks up data from one system, transforms it, and then loads it to another system. To explain it more deeply, ETL pipelines are a series of processes that help a business collect data from different sources and move it to destination systems — usually a data storage system such as a data warehouse — while transforming it along the way.

For professionals who count on business intelligence and data-driven insights, these pipelines are the make-or-break element, not to mention essential for data scientists and analysts.

If ETL pipelines are a new ground for you to break, you have landed on the right spot. As this article is value-packed with insights regarding ETL pipelines, the processes involved, their advantages, and some real-life use cases.

By the time you finish reading, you will have a full grasp of what ETL pipelines do. Not only that, you'll have gained insights into the difference ETL can make to your business and tools that will help you stay ahead of the competition.

What Are ETL Pipelines?

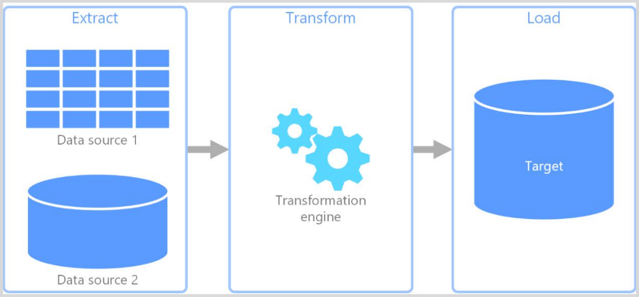


Image Source

ETL stands for **extract**, **transform**, and **load**. Together, this set of processes moves data from a source to a destination system.

Moving data around a company is extremely important. A modern enterprise typically has **many disparate sources of data**, as well as various goals that require data to be consumed in different ways and by different systems.

For example, you might have security measures, marketing and sales goals, financial optimization needs, and organization-level decisions to make. All of these things require data. In many places and many forms.

How do you make sure data is in the right place in the right format?

That's where data pipelines, including ETL pipelines, come in.

They allow businesses to **move data between systems in an automated fashion**, with data transformations included along the way.

Now, let's take a look at the three steps of ETL in depth.

I. Extract

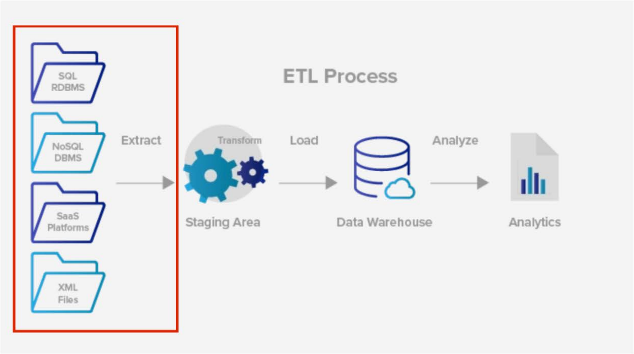


Image Source

The pipeline must first extract, or ingest, information and data from a source or multiple sources.

Data source systems often include:

- APIs
- Websites
- Data lakes
- SaaS applications
- Relational databases, or transactional databases

How does data extraction actually work?

The technical details **depend on the exact pipeline architecture**. Because the pipeline is a separate type of data system from the source, it requires some sort of interface to communicate: an **integration.**

A company might purchase an ETL tool from a vendor. In this case, integrations with different source systems are usually included with the tool. In other cases, data engineers employed at the company might **build the ETL pipeline in-house**, including the necessary integrations with sources.

Pipelines may extract source data in **real-time,** meaning as soon as it appears in the source. They can also use a **batch processing** workflow, meaning that new changes are picked up at a set time interval.

II. Transform

Transformation is the second step of the ETL process. In this step, raw data extracted from the source is transformed into data that is compatible with the specific destination and use case.

For example, your business intelligence tool might require JSON data that conforms to a certain schema, but your source data is in CSV format. The transformation step of your ETL pipeline could change the data from one format to another as it passes through.

That's just one example, and use cases vary widely. **But one thing is almost always true: you must transform data at some point during its lifecycle.** In an ETL pipeline, that transformation happens almost immediately after the data is extracted.

What types of data processing happen during the transformation step?

Most often, they include simple processes like:

- Filtering
- Aggregation
- Data cleaning
- Feature extraction
- Re-shaping data to conform to a schema

These relatively simple transformations have a critical purpose. They ensure that when the dataset arrives in storage or another application, it's **clean, consistent, and won't cause errors.** It's ready to be used to meet business goals.

The fact that they are simple also makes them easy to automate. This saves data team members time from a task that would otherwise be quite tedious.

By contrast, more complex, exploratory data transformations and deep analysis can occur later in the data life cycle. This work is **more complex and usually requires human attention**. But automating the easy stuff up-front allows you to arrive here faster.

III. Load

The last step of the ETL process is loading the data into the target system, or destination. It's the inverse of extraction.

In a typical, popular use case, the destination is a data warehouse or some other type of data store designed to power data analysis. Once in the warehouse, all sorts of analytic workflows can take place: business intelligence, machine learning, exploratory analytics, and more.

But use cases vary widely, and there are many other possible destinations. Generally speaking, destination systems can include:

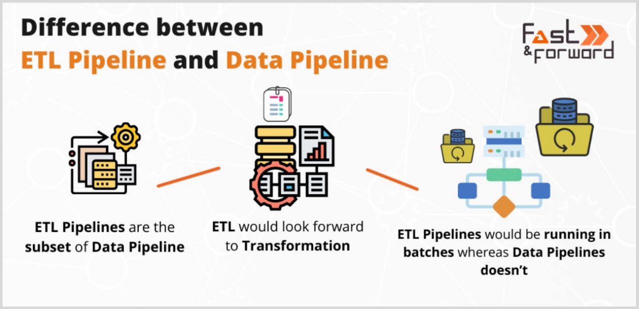
- Data warehouses
- SaaS applications
- Operational business systems
- Visualizations and dashboards

Just as with the source, the pipeline needs to integrate with the destination system to load data. The technical details of the loading process depend on whether you use an ETL tool, what tool that is, or whether your ETL pipeline was built in-house.

Once data is loaded, you can start to notice the **latency** of the data flow**;** that is, the amount of time it takes the ETL data to move from source to destination.

Depending on the pipeline, latency can be in the hours, minutes, or, if real-time **data streaming** is used, down to the millisecond. We'll discuss whether real-time ETL is possible toward the end of this article.

ETL Pipeline Vs Data Pipeline



Difference between ETL Pipeline and Data Pipeline

ETL Pipelines are the **subset** of **Data Pipeline**

ETL would look forward to **Transformation**

ETL Pipelines would be **running in batches** whereas **Data Pipelines doesn't**

Image Source

ETL pipelines fall under the category of **data integration:** they are data infrastructure components that *integrate* disparate data systems.

More specifically, ETL pipelines are a subset of data pipelines. Simply put, a **data pipeline** is any system that takes data from its various sources and funnels it to its destination.

But it’s not really that simple.

Every company’s data infrastructure and strategy are complex. Companies have different sources, destinations, and goals for their data. As a result, when you look closer, you’ll find that every data pipeline is unique.

ETL pipelines are a sub-category of data pipelines designed to serve a subset of the tasks performed by data pipelines, in general.

If you’re familiar with the world of data, you may have heard “data pipeline” and “ETL pipeline” used interchangeably, although they are different. This usage pattern has to do with the history of data pipelines.

ETL was among the first data pipeline architectures to gain popularity in the enterprise. This is because it was fairly simple for a data engineer to build a basic on-premises ETL pipeline specifically for every one of their employer’s use cases.

In those days, you couldn’t purchase an ETL tool; you *had* to build your own.

Over the years, data infrastructure has moved into the cloud, compute has grown more powerful, and service providers started to sell different types of tools. As a result, the possibilities for data pipelines have grown complex.

Despite all this, it’s very possible to hone in on a few things that set ETL pipelines apart from other types of data pipelines.