eBook

From ETL to ELT

The Next Generation of Data Integration Success in 2024

matillion

This eBook will explain the evolution of data warehousing and in turn ETL to ELT, as the two are intrinsically linked.

Contents

01

Introduction

The data warehouse has evolved over the last 30 years with changes to supporting processes, applications and technologies.

In 2024, ELT and pushdown data processing are pivotal for agile and scalable analytics. ELT streamlines data ingestion by deferring transformations, while pushdown architecture minimizes data movement, optimizing performance. These approaches align with the demand for real-time insights from diverse and voluminous data sources. By leveraging modern data warehouses and cloud platforms, organizations achieve cost-effective and rapid analytics, facilitating swift decision-making and maintaining competitiveness in the dynamic tech landscape.

The origin, growth and decline of ETL can be mapped directly to databases, data warehousing and cloud innovations. In this eBook we review pivotal moments in data warehousing history to understand the changes in ETL, ultimately resulting in the shift from E-T-L to E-L-T for modern cloud data platforms.

The idea of ETL can be traced back to the 1970s and the rise of centralized data repositories. However, it truly entered the technology landscape in the late 1980s and early 1990s when data warehouses took center stage. It was at this time we saw the introduction of primitive tools built to help load data into these new data stores.

As early adopters looked to set up their new data warehouses, they needed a way of bringing together many siloed data sources into a single repository. They needed to 'extract' their data, 'transform' it into the destination format and then 'load' it. These are the fundamentals of E-T-L. With large, data- centric businesses adopting data warehouses, the number of ETL offerings started to increase. However, this is the story of primitive ETL tools made specifically for on-premises databases and data warehouses. Technology has come a long way since then.

Modern cloud data platforms of the 21st century are just as, if not more, disruptive than the on-premises counterparts of the 20th century.

As early adopters looked to set up their new data warehouses, they needed a way of bringing together many siloed data sources into a single repository.

02

The Rise and Fall of ETL

The first ETL solutions were built hand-in-hand with their data repositories.

The first ETL solutions were built hand-in-hand with their data repositories. Relational databases were the new hot technology in the 1970's then came data warehouses in the late 1980's and early 1990's. Large companies, mostly in finance and retail, began acquiring on-premises databases, building out data center estates managed by IT teams with specialist knowledge. While these systems were far from perfect, they were revolutionary and fundamentally changed the way we would come to understand data.

How did companies get data into their new systems? By extracting it from the source system, transforming it into the destination format and then loading it into a target data store like a data warehouse.

This process grew in importance with the number of source systems a company used. For example, data could originate in a payments system, Excel document, CRM and ERP systems. Thus, ETL was the traditional method for 'extracting' data from numerous source platforms, 'transforming' the data on an ETL server and then 'loading' all the transformed data into a data warehouse ready for analytics and reporting.

As companies realized the potential data warehouses offered in terms of reporting and analytics, clean and structured data became fundamental to Business Intelligence(BI).

The ETL engine was therefore a compute resource, and needed to be powerful enough to handle growing amounts of data to be transformed, used, and reused.

To do ETL well you needed fast, expensive disk space to store large data sets, sometimes temporarily; fast, expensive processors to perform calculations on the data; and lots of fast, expensive memory to perform data operations, such as aggregates and joins efficiently.

This made databases and ETL capabilities of the time extremely costly, inflexible, and thus only afforded by large enterprises. The need to onboard highly specialized IT staff that could use and develop on premises databases and data warehouses came at a high price. Once embedded, a few people held all the knowledge, and even small team changes impacted smooth day to day operations.

As for the technology being developed, high levels of customization and bespoke solutions were fragile. Any changes in the technology stack incurred further bespoke configurations and highly specialized staff to implement. This model was not ideal for inevitable business change.

There were consequently a few disadvantages to ETL, even at its heyday, which have persisted in modern times. The target schema needs to be known at the time of loading since the data will be transformed to match the schema when in transit.

Furthermore, the granularity of raw data is often sacrificed in order to gain performance in the ETL process. Regardless of opting for performance or granularity, ETL was, and still is expensive and therefore primarily adopted by large companies. With only big enterprise needs

To perform ETL well, you need two important things: fast, expensive disk space for storing large data sets (sometimes just temporarily), and fast, costly processors to handle the calculations.

The velocity and variety of data is expanding to seemingly infinite scales with the proliferation of structured and unstructured data sources.

in mind, ETL can be considered inflexible for data-driven, agile SMEs. These limitations and challenges are further exacerbated by technological advancements in the nature of data (volume, velocity, variety, and veracity) today.

Thanks to technological advancements in cloud computing and data management, there is a steady migration from on-premises data warehouses to cloud data platforms. This cloud migration wave is challenging the relevance of ETL. Legacy environments running ETL software weren't built to scale in the same way as cloud data platforms. Therefore, as data volumes increase and workloads become more complex, these environments consume more IT resources, creating bottlenecks in the data chain and negatively impacting reports and analytics. The worst outcome – bad business decisions, being made slowly, resulting in missed opportunities and ultimately losses.

Today when companies talk about data they are referring to 'big data' - and that is not just big businesses. Smaller and medium sized companies are recognizing the value of data, pushing for more investment in capturing, storing, and analyzing larger amounts of data for advanced analytics and projects like customer 360.

Furthermore, the velocity and variety of data is expanding to seemingly infinite scales with the proliferation of structured and unstructured data sources. These paradigmatic changes in data technology are finding traditional data warehouses to be inflexible, too costly and painfully slow for the modern tech savvy, data-driven company. All of these are playing a critical role in challenging the relevancy of ETL in the cloud.

Historical ETL

Data is extracted from the source(s) and transformed by an ETL engine en route to its permanent home in the data store, which is usually a relational database/data warehouse.

However, it is not necessarily all doom and gloom. If you're still using an on-premises infrastructure and your data is predictable, coming from only a small number of sources requiring only minimal transformations – ETL could still be a legitimate cost-effective strategy. We suspect, however, that is not the case for most modern companies.

E

Extracting data from the source(s), and usually also implying that:

- There are multiple sources
- Data is staged into files or another relational database

T

Transforming (i.e. converting) the raw data into a format that's suitable for reporting and analytics. This typically includes:

- Enforcing consistency (currencies, time zones, units of measurement)
- Enriching and validating (dealing with missing values, duplicates)
- Denormalizing to a simpler data model (usually a star schema)
- Joining disparate data sources
- Applying business rules

L

Loading data into a target platform (e.g. a relational database, data mart, or data warehouse).

03

The Modern Cloud Data Platform

To break the 'Goldilocks Dilemma', most cloud data platforms have instant elasticity to scale up and down to meet demand.

The migration from on-premises data warehouses to cloud data platforms is fundamentally changing the way we view and understand data.

The migration from on-premises data warehouses to cloud data platforms is fundamentally changing the way we view and understand data. Cloud data platforms, including cloud data warehouses, data lakes, and data lakehouses, are not only data repositories but potential data gold mines. They need to be secure but accessible, cost efficient but scalable, and capable of importing and exporting data from seemingly endless sources. The underlying architectures of modern cloud data platforms means ETL and legacy solutions are no longer fit for purpose in light of modern data challenges and needs.

Differences between On-Premises and the Cloud

The most obvious difference between cloud data platforms and traditional on-premises data stores is their "serverless" design. Of course, there are plenty of virtual machines, networks and disks behind the scenes making it work, but this is all orchestrated and managed by the cloud vendor on your behalf. The fully managed nature of the service also offers security, caching, backups, encryption, availability, disaster recovery and other safeguards to ensure your data is as safe as technologically possible. These activities are carried out by the cloud vendor experts, meaning you don't have to train or acquire these resources, often resulting in overhead savings.

Instead, you are left with the business focused task of gaining value and insights from your data without the headaches of hardware and software provisioning. Historically, with on-premises systems you would be burdened with laboriously extrapolating your data needs 1, 3, or 5 years down the line. It would be impossible for any company to exactly forecast and predict data needs and scale accordingly. This would put companies in a 'Goldilocks Dilemma'. You may overestimate, paying up-front for dedicated hardware and software that might not be fully utilized for years; or underestimate and over utilize within the first 6 months sending you back to your CTO to go through the painful procurement process all over again.

To break the 'Goldilocks Dilemma', most cloud data platforms have instant elasticity to scale up and down to meet demand, and are pay-as-you-go for compute resources. This means you pay compute usage by the second, hour, month, or query, with no upfront costs. This model can also reduce your risk of vendor lock-in. In the event of a growth spurt, you can take advantage of the Massively Parallel Processing (MPP) architecture and just pay for that time period of use. Today's modern cloud data platforms support both scaling up and scaling out for improved performance and increased concurrency, respectively.

What does this mean for ETL? There are some key fundamental differences between on-premises data warehouses and cloud data platforms. These changes have impacted the supporting processes, applications and technologies. ETL is no exception.

04

ELT for Data Integration Success

The obstacle posed by data integration, or more accurately the lack thereof, is only increasing.

Articles and blogs on the death of ETL are being published with increased frequency.[1] But there are some use cases for which ETL is still a legitimate and necessary tool. Furthermore, if ETL is dead, that doesn't mean there isn't a need for data integration tools.

The obstacle posed by data integration, or more accurately the lack thereof, is only increasing. Bespoke technology platforms, CRM, ERP, finance, marketing, email, and hundreds of other SaaS systems need to speak to your data platform to push data in and pull information and insights out.

So what options do you have if you have a lot of data from numerous sources and you want to shape, clean, filter, join and transform that data? ELT is the next generation of data integration success to overcome the siloed data epidemic.

What is ELT?

'ELT' means you extract data from the source, load it unchanged into a target platform (which is often a cloud data platform), and then transform it afterwards, to make it ready for use. There's also an important implied assertion that:

ETL is highly targeted (always driven by known requirements), and so only extracts certain very specific data from sources, transforms it in a separate staging area, then loads it into the data warehouse.

ELT in contrast, can be less selective, where you quickly extract and load all the source data into the cloud data platform, and then allow the user(s) to decide later what's needed for analytics or their use case.

E-L-T, as opposed to E-T-L, 'Extracts' data from source systems, 'Loads' it in its raw form into a target platform, and then allows you to 'Transform' it in the data platform. It immediately becomes convenient to access, won't disappear, and is easy to audit. ELT then re-uses the power of the MPP cloud data platform to do the transformations, which gets the data ready for presentation, reporting, analysis and modelling. With ELT tools such as those offered by Matillion, this involves running push-down SQL, and means that you only need one powerful piece of infrastructure — the cloud data platform.

ELT leverages the power of the data platform itself to perform transformations and get the data into a business-ready format. Cloud data platforms, such as Snowflake, Delta Lake on Databricks, Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery make transforming your data easier, faster and more cost efficient. This design results in savings on infrastructure, better performing workloads, and shorter development cycles. Your data is quickly migrated and immediately available for transformations and analysis based on current business questions and needs.

Also it means you don't need to know how you are going to use that data from the start. You have the freedom to apply transformations at a later stage once its use case

ELT leverages the power of the data platform itself to perform transformations and get the data into a business-ready format.

[1] For example; Mintz, Daniel. "ETL Is Dead." InfoWorld, 13 Oct. 2017, www.infoworld.com/article/3231652/analytics/etl-is-dead.html

E-L-T, as opposed to E-T-L, ‘Extracts’ data from source systems, ‘Loads’ it in its raw form into a target platform, and then allows you to ‘Transform’ it in the data platform.

becomes clearer. This ability is increasingly appealing given the changing nature of development with the rise of iterative agile methodologies. Plus, different groups have different needs. For example the data science team often wants raw data, and the analytics group wants processed, structured data — so ELT fits both needs.

That being said, ELT has some obstacles that you shouldn't ignore. Since your transformations are being done in the data platform, you will need available space and compute power. Without this performance, queries will suffer. The aforementioned cloud data platforms however, facilitate scalability in a cost efficient manner that helps address this challenge!

Another problem is the timely and labour-intensive process of scripting. Seeming like a "quick win" early on in a project, scripting becomes more difficult to maintain as the project grows in scope. This is not just for loading data, but transforming it too. It applies especially when dealing with unusual or unstructured data types, or where access is not simply file-based. Dealing with an increasing amount of data, data sources, and required outputs can make these jobs increasingly complex, subjecting them to human error. Where mistakes are made with hand coding, it could take days or weeks to script, run, rollback and start again. Luckily, there are a number of products on the market that make this process quicker and easier, such as Matillion Data Productivity Cloud.

ELT Applied

While ELT is less selective, you will need to relationalize the data if; you are not in control of the source data formats, the source data changes frequently or without notice, or your company has informal or unstructured procedures for storing data.

05

Exactly why is ELT better?

ELT is better than ETL because it is faster and more cost efficient.

With the growing demand of data across the business, from insights and operational analytics, to data science, machine learning, and AI, companies now need to provide data in different stages: raw, cleansed and aggregated.

Why?

To answer this question we need to break down and compare the component parts of 'E', 'T' and 'L'.

ELT is Faster

Let's start with Extract. In both the E-T-L and E-L-T scenario the extract performance is very often outside of the control of the overall process. Data extraction speed is instead a factor of source system performance, and the network speed between either the source and the ETL infrastructure or the target data platform in the case of ELT. Implementation choices such as incremental, or even real-time, data extraction can help with the overall performance here, but come at the cost of complexity in either case. In summary, the extract part of ETL probably does have a massive bearing on overall performance, but fortunately if this part of the process is a bottleneck there are many ways to help (more network bandwidth, use of database read replicas, change-data-capture and so on). As the need for real-time data to power operational analytics and data science has increased, companies are moving to use change-data-capture, or CDC, to quickly overcome batch processing limitations and replicate each data change as it occurs to the cloud data platform. CDC fits very well in the E-L-T paradigm.

Next, there is Loading data. In E-T-L this is done last, post transformation. In E-L-T it's done earlier. The loading of data is also a function of the performance of the target system, however, modern cloud data platforms have excellent data loading capabilities when compared to generalist databases. So, while loading could be a factor if the target is a generalist database, it doesn't need to be.

Transformation, which happens last in E-L-T, is where most of the performance comes from. This improves speed because transformations can be broken down and executed in parallel across multiple hardware nodes.

Transformations are heavily optimized by the cloud data platform where storage and compute have been designed and optimized for each other. In addition, the data platform has highly detailed knowledge of how the data is stored and distributed, the data types, lengths and ranges, the compression used, the context of the query, and much more. Other data platforms such as Hadoop/Spark operate without this deep knowledge of the source data and cannot optimize queries and transformations so heavily.

ELT is More Cost-Efficient

With the growing demand of data across the business, from insights and operational analytics, to data science, machine learning, and AI, companies now need to provide data in different stages: raw, cleansed and aggregated. For example, data

ELT is using the benefits of the cloud data platform, and you do not need to stand up and pay for additional cloud servers and networking costs.

science teams often want the raw data (unstructured, semi-structured, and structured), machine learning and AI models may want cleansed data (structured), and the analytics team wants the cleansed or aggregated data (structured).

In the past, on-premise data warehouses often just retained the cleansed or aggregated data, since it was too costly to retain the raw data as well. Modern cloud data platforms were designed with infinite scale to meet today's data demands and cost-effectively support all three storage stages, providing the benefit that you can always go back and query the raw data as business needs and strategies change.

The cost issue occurs when using ETL as you need to load data out of the cloud data platform into a separate cloud service staging area for transformation then load it back into the cloud data platform. So this would happen twice, once going from raw to cleansed, then once going from cleansed to aggregated. ELT, on the other hand, is more performant and cost efficient as the data stays within the cloud data platform. Data is not being transmitted between other heterogeneous systems across a network. The internal networks joining the nodes of the cloud data platform are highly optimized.

In summary, ELT is using the benefits of the cloud data platform, and you do not need to stand up, and pay for, additional cloud servers and networking costs.

06

ETL vs. ELT. Practical Differences

How ETL vs. ELT tackles analytic functions, unstructured data, calculations, lookups (joins) and aggregations.

We have talked about how modern cloud data platforms are challenging ETL's performance and capability.

ELT tools built for the cloud offer a superior alternative to cloud-based ETL tools and traditional ETL tools that have been 'ported' to the cloud. The table below explains the practical differences in how ETL vs. ELT tackle analytic functions, unstructured data, calculations, lookups (joins) and aggregations.

Analytic Functions

ETL

Analytic functions return an aggregate value that is somehow related to the current record. This implies that all such records relating to the current record are available for processing. However, **an ETL process only sees a single batch of data, which means the analytic function needs to be done by the data warehouse** (which has all records - apart from the ones in the current batch which haven't been loaded yet!).

For example, a year-to-date value, or Lifetime value metric may require a summary over thousands of records not in the current batch window. This can be incredibly expensive and time-consuming to process.

We have seen ETL users tie themselves in knots trying to engineer solutions to analytic function problems in ETL without running out of memory, CPU or some other resource. The ETL paradigm just does not scale like the cloud platforms.

ELT

Analytic functions are a crucial feature of all good cloud data platforms, **excelling with the parallel database engine and columnar data storage. This combination allows for the rapid and efficient extraction of analytic insights from raw datasets**, providing immediate query-ability without constraints on the size of the data context. All the data is immediately queryable, meaning there are no limitations on the size of the context you can see the data in.

For example, when ranking a datapoint within a category, the database engine must first sort the entire category and apply the rank, as the size of the category is unknown. It's a huge advantage to be able to rapidly sort all of the dataset.

Modern ETL tools are fairly quick with smaller datasets but performance may vary when dealing with large volumes.

Unstructured Data

ETL

Since ETL relies on transforming the data to a structured form before loading, this makes **dealing with proprietary formats or unstructured data more difficult.** Also, transforming a large dataset can become a bottleneck in the ETL process.

ELT

ELT can easily handle both structured and unstructured data of any size. You do not need to map and transform it before loading, so it is available to users quickly. Then the data analyst, data engineer or data scientist, can decide which data to use and when to transform for analytics or other projects.

Calculations

ETL

Calculations can be expressions or applied functions within the ETL server itself. Modern ETL tools are fairly quick with smaller datasets but **performance may vary when dealing with large volumes.** Calculations may result in overwriting existing columns or a new derived column appended to the dataset and pushed to the target platform.

ELT

Bring raw data into the target data platform and then easily add a calculated/derived column to existing data. Some ELT tools, such as Matillion ETL have **pre-built components to make calculations quick and easy** to set up and allow you to specify SQL expressions compatible with your target platform to drive your calculations.

ELT differs from ETL in that cloud data platforms are designed and optimized for handling large quantities of data by crunching individual transactions in parallel.

Lookups (Joins)

ETL

To perform a lookup, ETL would go row by row to map a fact value to a dimension key from another table or source. An API lookup or execute function would bring back a key which is then appended to the data and pushed to target. Usually this works okay, but there is a constraint of needing both facts and dimensions or whatever sources are helping you with this data to be available at that point in time. Another challenge is the amount of data you're working with when performing lookups. **If the dimension table is really big you might have to partially or fully cache the data set.** Performance is dependent on the capability of your ETL server and the options provided by your ETL tools.

ELT

In an incremental scenario where a fact table is being added to new data on a regular basis, it's often necessary to join back to the complete fact to perform valuable analytic business calculations. In ETL this involves large and complex lookups that need to be done in memory and over the network. By contrast, in **ELT flows this computationally heavy task can be delegated to the cloud data platform.**

With ELT you can marry fact table records with appropriate dimension keys. Again this is implemented using SQL and typically a Left Join in order to find matches. All the data you need for your join is already present since it was extracted and loaded previously. **ELT differs from ETL in that cloud data platforms are designed and optimized for handling large quantities of data by crunching individual transactions in parallel.** This inevitably leads to faster processing times to populate your data platform.

Performing aggregations on the ETL server can be very expensive and you may need sufficiently powerful ETL servers with huge amounts of memory to handle large datasets.

Aggregations

ETL	ELT
These are very tricky in the ETL world, especially if you want to keep granular and aggregated data. You will end up with multiple stores of the same dataset with different granularity levels. Aggregations are further complicated by very large datasets. **Performing aggregations on the ETL server can be very expensive** and you may need sufficiently powerful ETL servers with huge amounts of memory to handle large datasets. Some ETL tools allow you to perform pushdowns where possible, but require a lot of hand holding and manual coding and is a departure from how ETL usually works.	**By loading the data first you can then use the capability and power of the cloud data platform to apply transformations.** You can easily multiplex to use the same input with different transformation flows or use tables with transformed data from previous jobs to build complex workflows on large datasets. You can write the table(s) that result from the aggregation to storage platforms like S3 or Google Cloud Storage. The table(s) can then be imported into another database/data lake.

History of ETL

Data Warehousing and Matillion

1970s

On-premises ETL products emerge

1980s

On-premises data warehousing and ETL products go mainstream

1990s

Rise of business intelligence and need to consolidate data

2000s

Market moving to cloud computing for increased agility

ETL preferred way to integrate SaaS to traditional data warehouses

2010s

Increasing volume, velocity, variety of data to analyze

Rise of cloud-based data warehouses and data lakes with cost efficient scaling, instant elasticity and PAYG

Cloud-native ELT products emerge to leverage the power and economics of cloud data platforms

ETL vendors introduce ELT solutions

2017 Matillion for Google BigQuery

2017 Matillion for Snowflake

2015 Matillion for Amazon Redshift

2020s

Increasing demand (AI/ML, data science) to analyze a large volumes of raw and processed data

2022 Matillion Data Productivity Cloud

2021 Matillion for Delta Lake on Databricks

2020 Matillion for Azure Synapse Analytics

Conclusion

The migration from on-premises to the cloud should also trigger a switch from ETL to ELT.

At Matillion, we believe that ELT is the best architecture for the majority of modern cloud data platforms.

On-premises data warehouses are quickly being eclipsed by cloud data platforms catching the attention of data-driven companies. The migration from on-premises to the cloud should also trigger a switch from ETL to ELT which is specifically designed for these new, advanced cloud technologies.

At Matillion, we believe that ELT is the best architecture for the majority of modern cloud data platforms. Cloud data platforms and ELT, offer simplicity, superior scalability, improved performance, and lower costs when compared to on-premises data warehouses and ETL or even ETL in the Cloud. When used in combination with data interpretation and relationalization services, this end-to-end approach enables you to make the most out of your data.

About Matillion

Matillion helps teams get data business-ready, faster — accelerating time-to-value and increasing the impact data can have.

Matillion makes it easy to extract data from virtually any data source and load it to your cloud platform of choice. We then leverage the compute and storage of your cloud platform to transform the data with unprecedented scalability and performance. And the whole process is made as easy as possible with Matillion's low-code/no-code functionality and integration of the native functionality of destination cloud data platforms.

Thousands of enterprises including Cisco, DocuSign, Pacific Life, Slack, and TUI trust Matillion to load, transform, sync, and orchestrate their data for a wide range of use cases from insights and operational analytics, to data science, machine learning, and AI.

Experience the Matillion Data Productivity Cloud

Understand the value that Matillion can bring to your organization, all at your own pace. Get hands-on with interactive product demonstrations, watch specially prepared videos, get a demo, or start a free trial with Matillion ETL for Snowflake, Delta Lake on Databricks, Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery.

Explore Matillion

Get a demo

Try now

Everyone Ready

Onboard any data, user, or application easily, with high and low code tools that create business-ready data in minutes.

Stack Ready

Significantly enhance the value of your existing cloud data platforms, by extending capabilities without coding.

Future Ready

Provide infinite speed, scale and integration, with enterprise assurance as you experiment, evolve and grow.