
MATILLION
BUSINESS
search
Accelerate
Time to Insight
Speed up and simplify data transformation
by automating processes

Contents

02 Insight is the new currency

03 Human intelligence + strategic data automation

04 Why automate the data transformation process?

05 What should you automate?

06 Automation methods

07 Automate the data ingestion process

09 Don't forget

10 What questions should I ask about ETL automation?

11 ETL: The five automation functions you must have

12 Automation and the future

Insight is the new currency

Have you ever thought, I wish this process was faster? You're not the only one – your entire business is thinking the same thing. When it comes to gaining insight on how to get ahead and stay ahead in the market, the speed of data and the pressure on organizations to operate in near-real time require that everything go faster. Rapid time to insight is one of the biggest competitive edges an enterprise can have today. Without it, we can't know what our customers are thinking and doing and how best to serve them. We can't improve processes to operate more efficiently. And we can't know how to innovate and come up with products that change people's lives and increase the bottom line.

As data professionals, what can we do? We can speed up and simplify the data transformation process with automation. One of the biggest pain points we've experienced is the need for automation. We need to make Extract, Transform, Load (ETL) processes more efficient, so data is there when we need it – no matter how much data there is. Running jobs or processes manually and hand-coding will not help on either front. Scaling without automation means adding people. Soon we won't be able to hire employee resources fast enough to handle the amount of data we collect. Nor does it make financial sense to hire loads of people or to pay a premium for employees who possess the highly technical skills needed to do complex and rapid hand coding.

It's clear: doing everything manually is not feasible. And it only increases the bottleneck that keeps organizations from getting the valuable data and information that they need. Today, there's no other way but to automate and streamline the data transformation process.

By automating parts or all of the data transformation journey, we can transform more data and get it analytics-ready in a fraction of the time. This enables all parts of the business to accelerate insights, make better decisions based on accurate information, accomplish business goals, and innovate. Automation also gives the data team time back on their calendars to work on other valuable projects and develop new processes that can be automated.

Human intelligence + strategic data automation

Automation has a lot of connotations today, not all of them positive. When some people hear the A-word, they think of robots, the Singularity, and all humans getting a permanent vacation. That's not what we're talking about at all. In fact, at this time (and probably for a while), there's absolutely no substitute for data expertise and human intelligence in complex data analytics.

We're not even talking about automating the entire data analytics process. While much of the process can be automated, business logic and rules have to be applied within the code. That needs to happen either in the form of traditional hand-coding or by using an ETL product that generates the code for you. The same applies for the business intelligence and reporting process. Our expertise is better used for mining data and looking for new insights, not running the same job over and over. Interpreting and drawing out decisions and actions for businesses require human intelligence, and always will. Automating more tedious processes like data cleaning and preparation frees up time for more valuable data science activities.

We still need manual intervention for development and maintenance, but using an ETL product that helps write the code that transforms the data can save costs and time. No matter how good a data team is at hand-coding, we're rapidly approaching a point where it will be impossible to keep up with the deluge of data, which only increases the time to insight. And hand-coding on top of other hand-coding can create a brittle bird's nest of code that is susceptible to breaking with any change.

When thinking about automation, it is helpful to have an open mind and explore new technologies or processes that can help achieve faster time to insight.

Why automate the data transformation process?

Besides making your life easier and giving your organization the kind of insight it needs to compete in a data-driven world, automation has several additional benefits.

Maintain up-to-date records
Automating data transformation methods and processes can help companies regularly ingest new data so that up-to-date records are available when needed.

Ensure teams focus on top priorities
Your BI team can not only deliver timely, relevant, insightful reports to the management team, but automation can also free them to work on critical, innovative initiatives that are strategically vital to the business.

Enable better decision-making with more accurate information
Faster access to more complete and accurate reporting in turn enables the management team, to make faster and smarter business decisions and focus on more strategic endeavors.

Be BI- and AI-ready
Though we are just on the cusp of integrating machine learning and AI into analytics, automated ETL processes will only be more important in preparing data for machine learning and AI models. Especially as the data we collect continues to multiply exponentially and business moves at a faster pace.

Run a more cost-effective business
And last but definitely not least, time is money. Automating the ETL process and other aspects of the data journey helps reduce costs and use resources more wisely.



What should you automate?

If you're reading this, you are most likely analyzing data for business intelligence in a cloud data warehouse. If you want to accelerate time to insight, you need to start with the automation of your Extract, Transform, Load (ETL) or Extract, Load, Transform (ELT) workflow. In the ETL process, you extract data from various disparate data sources and load it into your database – in this case a cloud data warehouse. With ETL, your data team can also automate the transformation of that data, getting it ready for your BI analysts to create charts, dashboards, and other reporting assets.

Most enterprises are likely to be getting that data from hundreds of data sources, including social networks, marketing applications, and Internet of Things (IoT) devices. The time saved when the data team is able to automate the transformation of all of that data is invaluable. Every use case is different, and there will always be some degree of human intervention. But overall, there are several ways that strategic automation can lead to exponential improvements in the speed and quality of your data analytics and insight. Within the ETL process, there are several ways to make that happen.

Automation methods

There are several ways to automate your workflows, rather than run them manually, to create data that is ready for analysis.

Built-in scheduler

One benefit of using an ETL product that provides a built-in scheduler is that you do not need to rely on a third-party source or other mechanism to launch your jobs. This allows you to centrally manage your ETL job schedules in a product dedicated to such tasks, making your ETL processes easier to maintain, debug, and monitor.

Another benefit of using a built-in job scheduler is dependency management. Parent jobs can be scheduled to trigger child jobs. This practice allows you to componentize your jobs and reuse them, saving development time and making job management simpler.

Modern ETL platforms should also allow for error-handling, so notifications can be sent or corrective actions taken should errors occur while running a schedule. Not all ETL products are the same, and each one's approach to this technique may vary.

Third-party scheduler

Various third-party products can also execute ETL jobs. The benefit of using one of these is that they provide other functionality, such as the ability to integrate with legacy systems within your development landscape. For example, you can set up an ETL process to run after another system process completes. However, you will likely pay an additional cost to use a third-party scheduling product, not to mention expend time and resources to implement the product.

Execute jobs via API

Some platforms also allow you to execute jobs via an API call. Such API calls can be scheduled in a number of ways, including through an operating system's built-in scheduler. Here are a couple of examples:

- Bash script executing a cURL command, and the script can be scheduled to run via cron on UNIX and macOS operating systems.
- PowerShell script executing an Invoke-RestMethod command, where the script can be scheduled to run via Task Scheduler on Windows operating system.

Cloud service provider (CSP) services

As companies move to the cloud, there are additional services that CSPs provide that can assist in automation, including messaging services that can trigger a job to run. Processes can be created that monitor these job queues, though some products have this capability built-in. Whether it's a custom process or a production that supports such messaging, either can listen for an incoming message in a queue and trigger a job based on the contents of that message. Each product has different capabilities and options, but they generally work the same. Examples of messaging services include:

AWS SQS
Google Cloud Pub/Sub
Microsoft Azure Queue Storage

In addition to these messaging services, CSPs provide serverless functions that can be used to help with automation. For example, the function/code can be set up in such a way that other CSP services will automatically trigger it. The function can also work with the messaging service to help execute an ETL job. These functions can be written in a few different languages, including Node.js and Python. What's great about these services is that you pay only when the code is running and you don't need to manage real or virtual servers. Examples of function services include:

AWS Lambda
Google Cloud Functions
Microsoft Azure Functions

Automate the data ingestion process

In addition to the automation methods and functionality discussed, there are different ingestion processes that can help you fully automate your data workflow. You can set these up based on company requirements, for either near-real-time or batch intervals.

Change Data Capture (CDC)

CDC is the process of capturing changes from a data source and keeping a target in sync with those changes. For example, the target data source can be your cloud data warehouse tables where the source might be your on-premises Oracle database. Once the data is in sync, you can apply any additional transformations, and any tables referenced in a BI product will be up-to-date. An Operational Data Store (ODS) layer gets synced from the source. At that point, downstream reporting layers can be updated to reflect the newly synced data in the ODS. CDC is especially helpful to keep your data in sync, and better yet, this process can be automated. Here are some examples of methods you can use to achieve CDC:

- **High-water mark**
  Some challenges when using this method are that it will not capture deletes, last modified column can be updated by a user and has to be added on all tables (this is not always the case), and additional overhead on the source servers due to querying for new data since the last modified date. You also need to transform the data to capture new, updated, and deleted records, which can add some time to this process.

- **Transaction logs**
  Most databases have a transaction log or similar record. This log houses all changes made to the source database; scanning and processing those logs allows the same changes to be applied to the target system. What's great about log-based changes is that they cause minimal overhead on source servers. This method is especially helpful when working with transactional databases that are being used by a software application in production.

- **History triggers**
  This is an older method that records time-stamped database changes along with the type of database operation (insert, update, or deletion) that precipitated the change. This technique has some challenges, namely that it causes additional overhead on the source along with other inefficiencies, and it requires you to create and monitor additional resources.

- **Third-party vendors or built-in functionality of an ETL product**
  There are third-party vendors that perform CDC and save you the hassle of setting it up. Some ETL products also provide CDC functionality that may work hand in hand with other services on CSPs.

Batch processing

This method is used to process a large volume of data in batches rather than real or near-real time. Batch processing is best when the target data does not have to be up-to-date with the source, or if you don't update source data as frequently. You can automate this process by running it on a schedule, but it might run infrequently, for example, once a day.

Micro batch processing

This type of batch processing can be set to run on a schedule to obtain smaller volumes of data at a time and/or can be scheduled to run more frequently. If you need to frequently update data, it's a good approach.



continued...

With Matillion, one way to accomplish micro batch processing is by using a high-water mark and SQS to call the job (itself) to run. In addition, your workflow can allow for conditional statements based on maintenance windows and other parameters. In this case, the job will check for those conditions, and based on the return values, it will either start the job or wait.

One design pattern for micro batch processing is a high-frequency incremental load process. This pattern requires that the source data has a high-water mark column, such as a time stamp when data was last modified. When you load data into the cloud data warehouse, the job should have logic to obtain the latest record that was modified. For example, you can query your cloud data warehouse table or a view for the max last modified date and use that value such that all the data after that high-water mark are selected from the source. In this way, your micro batches will capture and apply only new records to the target.

Stream processing

Stream processing is useful when trying to keep your data up-to-date in near-real time. There are a few different streaming services, such as Amazon Kinesis and Apache Kafka. You can use a streaming service to load data to a cloud storage location and a cloud function to trigger a job to process that data. You can use this method in conjunction with other CSP services, such as messaging queues and function services. Before going this route, consider whether analytics will be made more meaningful or useful by having near-real-time data. For example, if you need to compare month-over-month statistics, near-real-time data is moot, and gathering it may be a waste of money and resources.

Job templates

Job templates or shared jobs can also assist in automation. Rather than developing multiple jobs that complete the same process, you can create a job template. You can set up these jobs so they can be referenced repeatedly. Note that you must use variables to parameterize the job template, and doing so is especially useful when working in different environments.

Don't forget

When you automate any process, it's not necessarily set it and forget it. To help keep things running smoothly and reduce pain points, remember:

No need to write documentation when you have a tool that will create it for you. Matillion offers auto-generated documentation with our Enterprise subscription.

Error handling for failures
Some things in life are certain. When your ETL and data analytics process is automated and running, it will inevitably fail at some point. There are a few reasons why. The culprits could be source and target data server downtimes, network disruption, corrupt or bad data, or old-fashioned human error. In these situations, you want to have error handling in place within your workflow. For example, if there is network disruption, you can implement a retry pattern to re-execute a job. If the job fails again after a retry or for another reason, you can implement alerting and notification. Alerting can happen via email or via a message sent to a messaging queue, where the appropriate team can review it.

Manual intervention may be needed
There will be instances where parts of the ETL process cannot be completely automated, depending on the company and use case. For example, source data formats can change. You may need to intervene manually to update the target or workflow to handle these changes. Or an employee may have made changes in a process or data source/target that impacts the workflow. That may require investigation and code updates by an actual person.

Document, document, document
After you automate the ETL process, you must document those processes for current and future employees. That way, when a project is handed off, it will have everything the next person needs for reference. This documentation can also be instrumental in disaster recovery.

Continuous Integration/Continuous Deployment (CI/CD)
If you are thinking about automating the ETL process, you need to consider how you will keep your code and software up-to-date. You may have a need or business requirement for CI/CD. With CI/CD, you are continuously checking in code to a source control repository and continuously deploying that code to different environments. These terms are often referenced with software applications, but are also important when working with the data development process. Matillion supports Git integration through our GUI - and supports CI/CD through third party tools such as Jenkins via API calls into the Matillion server.

Testing and QA
Once you automate the ETL process, you must create testing and quality assurance (QA) checks to verify that everything is working correctly. There are various testing methods, including regression, data quality, and metadata testing. ETL testing tools or code can be written for this. The testing and QA method can also be automated depending on the tool or process used.



What questions should I ask about ETL automation?

When evaluating ETL products for your cloud data architecture, there are several questions you should ask around automation and other steps to speed up your time to insight:

- Is there a built-in job scheduler?
- Can dependencies be managed effectively within the tool?
- Can the tool handle errors?
- Is there an API available to execute jobs?
- Are there any CDC features or functionality?
- Is there integration with other CSP services?
- Are there alerting capabilities when a job fails or succeeds, or achieves a result based on other parameters?

ETL: The five automation functions you must have

When looking to automate your ETL data process, there are five product functionalities that you can't skip. All of these functions are present in Matillion ETL software. They can enable you to more quickly provide valuable insights to your organization. Those functionalities and the options provided within Matillion are highlighted here:

Job triggers

Matillion has a concept of Orchestration (think of the **E** and **L** in the ELT process). Jobs that can trigger other Orchestration or Transformation (think of the **T** in the ELT process). Jobs, with conditional statements in place to deal with error handling and notifications.

CSP services integration

Matillion provides components and features that work with different cloud service provider products such as AWS SQS, Google Cloud Pub/Sub, and Azure Queue Storage. These can be leveraged with other technologies to listen for messages to trigger a job to run.

CDC

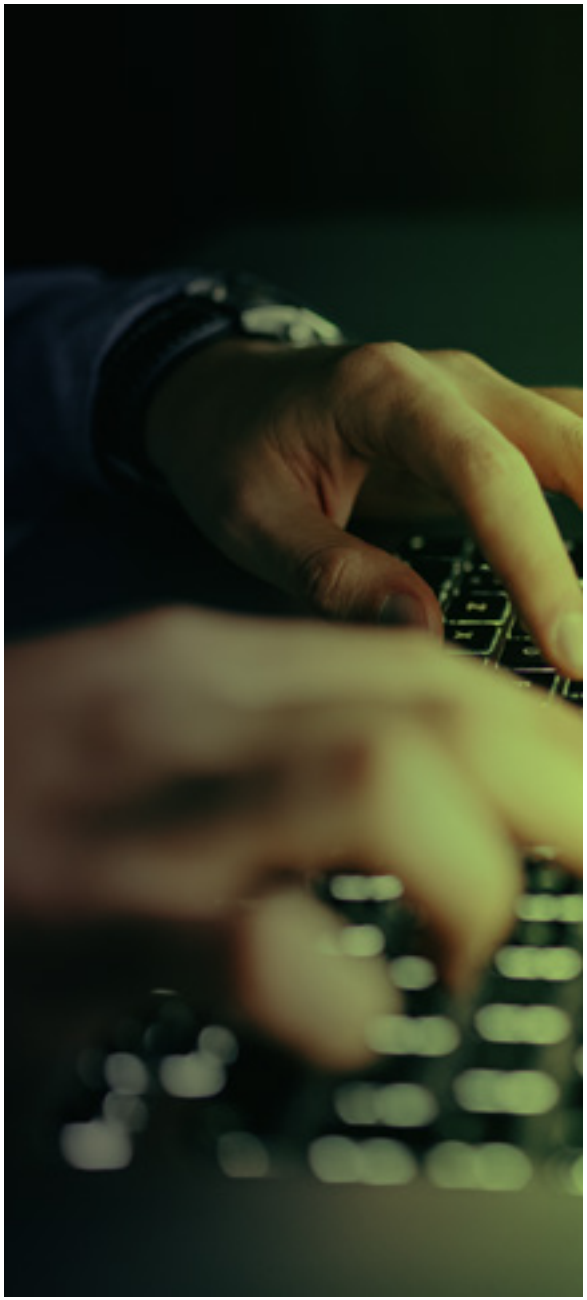
CDC functionality is built into the Matillion product running in AWS (Redshift and Snowflake) and can be used with several databases including MySQL, Microsoft SQL Server, PostgreSQL, and Oracle. Matillion works hand in hand with AWS services (DMS, SQS, S3, and Lambda) to accomplish this.

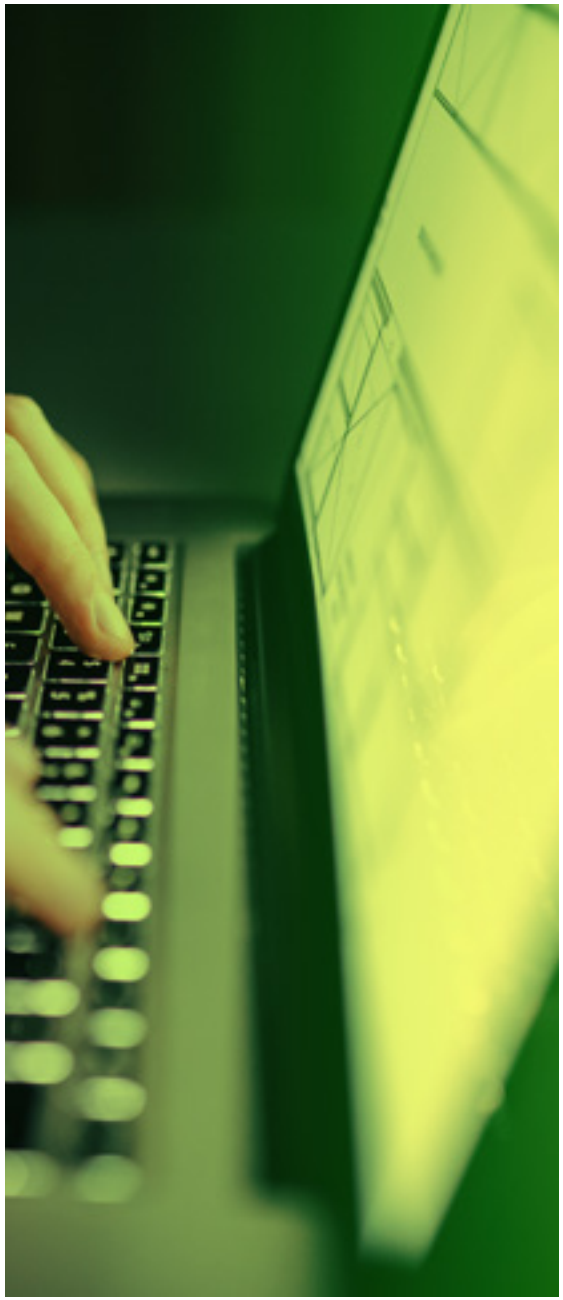
Built-in job scheduler

The Matillion UI provides a cron-based job scheduler that is easy to set up and maintain. With this scheduler, you can set up both time- and event-based scheduling.

API access

Matillion provides an API where a call can be made to execute jobs. This can be leveraged within a script to execute a job via the Matillion API connection. In addition to running jobs, the Matillion API provides several endpoints to access other useful items.





Automation and the future

We've talked about how automation and streamlining the ETL process can revolutionize the way we work now and lead to faster time to insight. But people always ask, what about automation and the future?

AI and machine learning are here, and automation will help us prepare data to get the most out of these technologies as they evolve. And as these and other technologies do evolve, it's in an organization's best interest to keep up with those changes and adapt. In that spirit, we need to embrace strategic automation of our work processes in the present so we are ready for the future. As for what that future looks like, here's to having the time and the insight to figure it out, adapt to changes, and continue to innovate.

© 2019 Matillion. All rights reserved

MATILLION

Simplicity.
Speed.
Scalability.
Savings.

Get a demo to see first-hand the power of Matillion data transformation

GET A DEMO

Purpose-built data transformation for cloud data warehouses

- *Intuitive UI and approach to data transformation.*
- *Achieve the fastest time to value, from launch to development to production.*
- *Take advantage of the power and features of Amazon Redshift, Snowflake, and Google BigQuery with a cloud-native tool.*
- *Pay-as-you-go with no long term commitments.*
- *Automate ELT processes to speed time to insight.*



matillion.com/get-a-demo/

© 2019 Matillion. All rights reserved