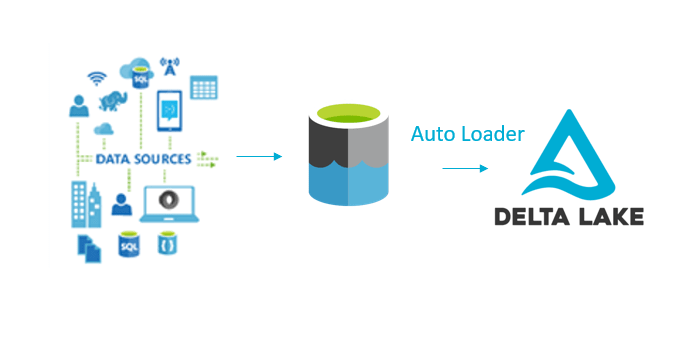
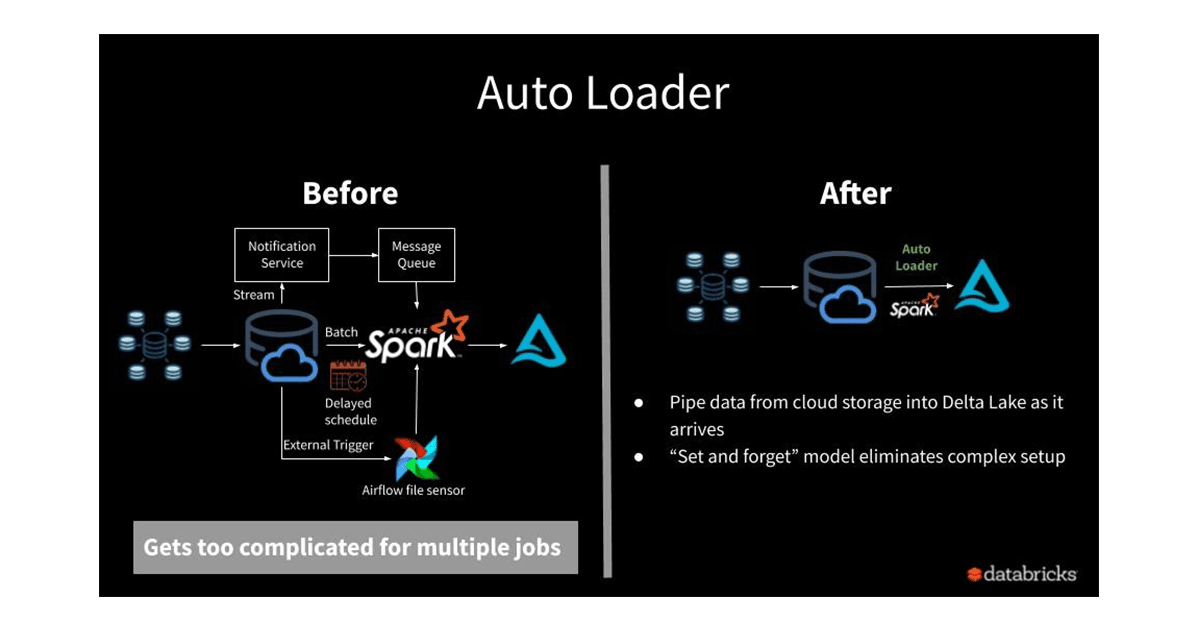
**Reference:** <https://hevodata.com/learn/databricks-autoloader/#:~:text=Databricks%20Autoloader%20is%20an%20Optimized%20File%20Source%20that,presents%20a%20new%20Structured%20Streaming%20Source%20called%20cloudFiles>.

<https://www.databricks.com/blog/2020/02/24/introducing-databricks-ingest-easy-data-ingestion-into-delta-lake.html>

Databricks recommends Auto Loader in Delta Live Tables for incremental data ingestion. Delta Live Tables extends functionality in Apache Spark Structured Streaming and allows you to write just a few lines of declarative Python or SQL to deploy a production-quality data pipeline.



Databricks Autoloader is an Optimized File Source that can automatically perform incremental data loads from your Cloud storage as it arrives into the Delta Lake Tables. Databricks Autoloader presents a new Structured Streaming Source called cloudFiles. With the Databricks File System(DBFS) paths or direct paths to the data source as the input, it automatically sets up file notifications that subscribe to file events to handle new files on arrival with an option to process the existing ones in the directory.

**Databricks Autoloader supports two methods to detect new files in your Cloud storage namely:**

Directory Listing: This approach is useful for cases when only a few files are required to be streamed regularly. Here, the new files are recognised from listing the input directory. With just access to your Cloud Storage data, you can swiftly enable your Databricks Autoloader Streams. From the beginning, Databricks Autoloader automatically detects if the input directory is good for Incremental Listing. Though, you have the option to explicitly choose between the Incremental Listing or Full Directory Listing by setting cloudFiles.useIncrementalListing as true or false.

File Notification: As your directory size increases, you may want to switch over to the file notification mode for better scalability and faster performance. Using the Cloud services like Azure Event Grid and Queue Storage services, AWS SNS and SQS or GCS Notifications, and Google Cloud Pub/Sub services, it subscribes to file events in the input directory.

**Key Features of Databricks Autoloader:**

**Using Autoloader can simplify your Data Ingestion process providing the following benefits to you:**

**Scalability:** Databricks Autoloader can track billions of files by leveraging the Cloud Services and RockDB without the need to list all the files in your directory.

**Cost-Effective**: The notification mode for file detection eliminates the need for a directory list, thereby reducing costs. The cost of file discovery also directly depends on the number of files that are ingested rather than the number of directories in which the files arrive.

**Ease of Use:** The Databricks Autoloader sets up the notification mode and message queue services to perform Incremental Data Load. You also don’t require to track files or manage any state information on what files have arrived.

**Schema Inference and Evolution**: For cases when there are schema drifts such as new columns, Databricks Autoloader will manage it and notify you whenever schema changes. Using the semi-structured data access APIs, you can also rescue data (unexpected data in a column such as different data types) that otherwise may be lost or ignored.