BEST PRACTICES FOR DATA ENGINEERING

How to scale and process your data faster

CHAMPION GUIDES

TABLE OF CONTENTS

INTRODUCTION

There's never been a better time to be a data engineer.

But the parameters of the job are changing quickly. Databases and data warehouses are moving to the cloud and new tools and data pipelines are taking over traditional data engineering tasks such as manually writing ETL code and cleaning data. As a result, companies are asking engineers to provide guidance on data strategy and pipeline optimization. In addition, as information grows exponentially and as the sources and types of data become more complicated, engineers must know the latest strategies and tools to help the business leverage that data for increased profitability and growth.

"Data engineers have become valuable resources that can harness the value of data for business objectives, which ultimately plays a strategic role in a complex landscape that is essential to the entire organization," says big-data news portal Datanami. "Understanding and navigating data needs has the ability to empower data engineers to propel an organization into a thriving data-first company."[1]

If you're a data engineer looking to make the right decisions about data strategies and tools for your organization, here are 11 best practices for data engineering that can mean the difference between profitability and loss.

BEST PRACTICES FOR DATA ENGINEERING

1. ENABLE YOUR PIPELINE TO HANDLE CONCURRENT WORKLOADS

To be profitable, businesses need to run many data analysis processes simultaneously, and they need systems that can keep up with the demand. Data comes into the enterprise 24 hours a day, seven days a week, from the web, mobile devices, and Internet of Things (IoT) devices. Your data pipeline has to load and process that data while scientists are analyzing the data and downstream applications are processing it for further use. A modern data pipeline that lives in the cloud features an elastic multi-cluster, shared data architecture that enables the handling of concurrent workloads. It can allocate multiple independent, isolated clusters for processing, data loading, transformation, and analytics while sharing the same data concurrently without resource contention.

2. TAP INTO EXISTING SKILLS TO GET THE JOB DONE

Many pipelines use complex algorithms that seemingly require data engineers to use Apache Spark, Apache Kafka, or Python. But you don't have to learn new platforms to solve problems. Instead, find a way to use your current skills. For example, modern ETL enables you to accomplish your stream processing task using direct SQL statements rather than using Kafka. Maximize your current skills before you invest resources learning something new.

3. USE DATA STREAMING INSTEAD OF BATCH INGESTION

Data comes into your business 24 hours a day, so a periodic batch ingestion can miss recent events. This can have catastrophic consequences, such as failure to detect fraud or a data breach. Stale data can affect profitability, as well. For example, a company running an online shopping event wants immediate insights into which products are most viewed, most purchased, and least popular as soon as possible, so they can quickly take actions such as changing the website's layout to drive more sales. Set up continuous streaming ingestion to decrease pipeline latency and enable the business to use data from a few minutes ago, instead of a day ago. Understand the available streaming capabilities and how they work with different architectures, and implement pipelines that can handle both batch and streaming data.

4. STREAMLINE PIPELINE DEVELOPMENT PROCESSES

To ensure the validity of production data, build pipelines in a test environment, where you can test code and algorithms iteratively until they are ready for a production environment. By using a cloud data platform as the foundation for running data pipelines, creating test environments can be as simple as creating a clone of an existing environment without the rigor of managing new databases and infrastructure. This will greatly accelerate the time to go from development to test to production far faster than building these same pipelines on premises.

5. OPERATIONALIZE PIPELINE DEVELOPMENT

After creating a pipeline, you may have to modify it or scale it to accommodate more data sources. Design your pipelines so they can be easily modified or scaled. The concept is known as "DataOps," or DevOps for data, and it consists of building continuous integration, delivery, and deployment into the pipeline using automation and, in some cases, artificial intelligence (AI). Incorporating DataOps in your pipeline will make your data more reliable and more available.

6. INVEST IN TOOLS WITH BUILT-IN CONNECTIVITY

A modern, cloud-based data pipeline accommodates many tools and platforms that need to communicate with each other. Building connections between source systems, data warehouses, data lakes, and analytics applications takes time, labor, and money. Instead, invest in tools that have built-in connections to each other. If your tools don't have connectivity, do the extra step of storing data in a generic form such as the format used by Amazon Simple Storage Service (S3) so other tools can pick it up.

7. INCORPORATE EXTENSIBILITY

Organizations use many disparate tools to derive meaning from their data. For example, organizations may write custom APIs to scan images and extract text from the images. Another example of a custom algorithm is doing sentiment analysis of customer service chats. Make sure you build modern pipelines that can leverage this code. By using APIs and pipelining tools, you can create a data flow that uses outside code seamlessly.

8. ENABLE DATA SHARING IN YOUR PIPELINES

Often, multiple groups inside and outside of your organization need the same core data to perform their analyses. For example, a retailer may need to share sales data with three different suppliers. Building separate pipelines with the same data would take time and cost money. As an alternative, modern tools in the cloud enable you to create a shared pipeline that enables you to govern who can access the data. Shared pipelines get the right information to the right people quickly.

9. CHOOSE THE RIGHT TOOL FOR DATA WRANGLING

A data wrangling tool can fix inconsistencies in data, transforming distinct entities such as fields, rows, or data values within a data set so they're easier to leverage. For example, the store name "Giantmart" might arrive in your pipeline from different sources in different ways, such as "Giant-Mart," "Giantmart Megacenter," and "Giant-mart Inc." This can cause problems as the data is loaded and analyzed. Cleaner data equals better, more accurate insights for business decision-making.

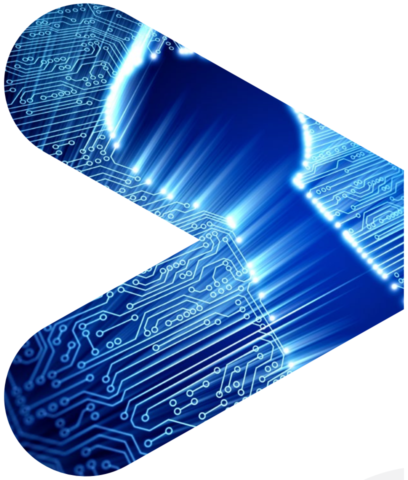
10. BUILD DATA CATALOGING INTO YOUR ENGINEERING STRATEGY

Analysts may have questions about the data in your pipeline such as where it came from, who has accessed it, or which business process owns it. A data scientist may need to view the data in its raw form in order to ensure its veracity. End users may also want to know which data sets can be trusted and which data sets are a work in progress. Build a data catalog that keeps track of the data lineage so you can trace the data if needed. This will increase the end users' trust in the data and will also improve the data's accuracy.

11. RELY ON DATA OWNERS TO SET SECURITY POLICY

Data engineers may not understand how to set the security policy—who can see it and what kind of access they have to it. For example, they might not realize that certain data fields need to be obfuscated before the data is sent to a particular user, potentially causing a security or regulatory issue. To prevent this scenario, the owner or producer of the data should set the security policy. Others can provide recommendations, but ultimately, the owner is most aware of how data needs to be secured before it is distributed.

The world of data engineering is changing quickly. Technologies such as IoT, AI, and the cloud are transforming data pipelines and upending traditional methods of data management. The decisions that you make about your data pipeline, whether large or small, can have a significant impact on the business. The wrong choices mean increased costs and time spent on unnecessary tasks. The right decisions enable the business to harness the power of data to achieve profitability and growth for years to come.



ABOUT SNOWFLAKE

Snowflake delivers the Data Cloud—a global network where thousands of organizations mobilize data with near-unlimited scale, concurrency, and performance. Inside the Data Cloud, organizations unite their siloed data, easily discover and securely share governed data, and execute diverse analytic workloads. Wherever data or users live, Snowflake delivers a single and seamless experience across multiple public clouds. Snowflake's platform is the engine that powers and provides access to the Data Cloud, creating a solution for data warehousing, data lakes, data engineering, data science, data application development, and data sharing. Join Snowflake customers, partners, and data providers already taking their businesses to new frontiers in the Data Cloud.
snowflake.com

CITATIONS

[1] datanami.com/2019/07/18/data-engineers-the-c-suites-savior