Brought to you by:

databricks

The Data Lakehouse Platform

for dummies®
A Wiley Brand

Manage all your organization's data

Combine the best of data lakes and warehouses

Build one open, unified system for all data

Databricks Special Edition

Ulrika Jägare

About Databricks

Databricks is the data and AI company. More than 5,000 organizations worldwide — including Comcast, Condé Nast, H&M, and over 40% of the Fortune 500 — rely on the Databricks Lakehouse Platform to unify their data, analytics and AI. Databricks is headquartered in San Francisco, with offices around the globe. Founded by the original creators of Apache Spark™, Delta Lake and MLflow, Databricks is on a mission to help data teams solve the world's toughest problems.

Twitter: www.twitter.com/databricks
LinkedIn: www.linkedin.com/company/databricks
Facebook: www.facebook.com/databricksinc

The Data Lakehouse Platform

Databricks Special Edition

by Ulrika Jägare

for dummies®
A Wiley Brand

The Data Lakehouse Platform For Dummies®, Databricks Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN: 978-1-119-85633-7 (pbk); ISBN: 978-1-119-85634-4 (ebk)

Publisher's Acknowledgments

Some of the people who helped bring this book to market include the following:

Project Editor: Carrie Burchfield-Leighton

Sr. Managing Editor: Rev Mengle

Acquisitions Editor: Steve Hayes

Production Editor: Tamilmani Varadharaj

Business Development Representative: Karen Hattan

Table of Contents

Introduction

As companies started to collect large amounts of data from many different sources, data architects began envisioning a single system to store data for many different types of usage scenarios, including analytical products and machine learning (ML) workloads.

Historically, many different solutions have been created and used to address this need; a database, a data warehouse, and, during the last decade, the data lake concept. These solutions have all had their obvious benefits at the time but also different types of limitations that grew apparent as data management (DM) needs have changed over the years. The emergence of the cloud is creating an opportunity for data teams to rethink their approaches. Modern cloud data platforms are following a new DM architecture — lakehouse.

The lakehouse radically simplifies the enterprise data infrastructure and accelerates innovation in an age when ML and artificial intelligence (AI) are used to disrupt every industry. This new architecture merges the best parts from data lakes with the best parts from data warehouses. Therefore, more traditional data warehouse use cases are also supported with the lakehouse approach.

In the past, most of the data that went into a company's products or decision making was structured data from operational systems, whereas today, many products incorporate AI in the form of computer vision and speech models, text mining, and others. That puts completely new demands on the DM system, and it's not just about the capabilities; it's about the architectural approach.

About This Book

The Data Lakehouse Platform For Dummies, Databricks Special Edition, is about using the principles of a well-designed platform that leverages the scalable resources of the cloud to manage all of an organization's data. This book introduces the lakehouse in DM, and you not only discover the evolution of DM solutions, but also you find out how the limitations of current solutions impact the efficiency of DM. This book explains why a lakehouse is more capable of solving the challenges of today, as well as how you can

design and build a cloud data platform and what it actually means for your company or organization.

Icons Used in This Book

I occasionally use special icons to focus attention on important items. Here's what you find:

This icon reminds you of information that's worth recalling.

Expect to find something useful or helpful by way of suggestions, advice, or observations here, leveraging experiences from other implementations.

Warning icons are meant to get your attention to steer you clear of potholes, money pits, and other hazards. Paying extra attention to these parts in the book helps you avoid unnecessary roadblocks.

This icon may be taken in one of two ways: Techies zero in on the juicy and significant details that follow; others will happily skip ahead to the next paragraph.

Beyond the Book

This book helps you understand more about how the lakehouse makes your DM efforts more effective and efficient in your company. However, because this is a relatively short, introductory book to lakehouses and cloud data platforms, I also recommend checking out the following:

- databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html
- databricks.com/discover/data-lakes/history
- databricks.com/product/unified-data-analytics-platform

IN THIS CHAPTER

- Explaining the role of relational databases
- Positioning data warehouses
- Describing the concept of data lakes
- Knowing why you need more than the data lake

Chapter 1
Describing Current Data Management Limitations

Data management (DM) consists of methods, architectural techniques, and tools for gaining access to and managing delivery of data in a consistent way across different data types in a company. The purpose of DM on an enterprise-wide scale is to fulfill all data requirements for use cases, applications, and business processes in a company.

This chapter describes how the approach to managing data has changed over time due to various factors and how these changes have pushed for new DM approaches to evolve. Some of these aspects include easier access to data, increased volume of data, the emergence of unstructured data, the need for speed in data preparation, and the necessity of reliable data pipelines that can constantly feed new types of use cases with data. The need for performing analytics on all your data across multiple data sources, as well as running end-to-end ML, puts high demands on data management support. There is usually a varied set of use cases in both analytics and ML that need to be taken into account.

Exploring Relational Databases

In the early days of DM, the relational database was the primary method that companies used to collect, store, and analyze data. Relational databases offered a way for companies to store and analyze highly structured data about their customers using Structured Query Language (SQL). For many years, relational databases were sufficient for companies' needs mainly due to the fact that the amount of data that needed to be stored was relatively small, and relational databases were simple and reliable.

However, with the rise of the Internet, companies found themselves drowning in data. To store all this new data, a single database was no longer sufficient. Companies, therefore, often built multiple databases organized by lines of business to hold the data. But as the volume of data just continued to grow, companies often ended up with dozens of disconnected databases with different users and purposes, and many companies failed to turn their data into actionable insights.

Sorting out Data Warehouses

Without a way to centralize and efficiently use their data, companies ended up with decentralized, fragmented stores of data, called *data silos,* across the organization. With so much data stored in different source systems, companies needed a way to integrate them. *Data warehouses* were born to meet this need and to unite disparate databases across the organization.

TECHNICAL STUFF

The concept of data warehousing dates back to the late 1980s, and in essence, the data warehousing concept was intended to provide an architectural model for the flow of data from operational systems to decision support environments.

WARNING

As data volumes grew even larger (*big data*), and as the need to manage unstructured and more complex data became more important, data warehouses had limitations:

- Data warehouses for a huge IT project can involve high maintenance costs.
- Data warehouses only support business intelligence (BI) and reporting use cases.

- There's no capability for supporting ML use cases.
- Data warehouses lack scalability and flexibility when handling various sorts of data in a data warehouse.

  This started the push for yet another DM solution: *data lakes* that could offer repositories for raw data in a variety of formats.

Diving into Data Lakes

To make big data analytics on various formats possible, and to address concerns about the cost and vendor lock-in of data warehouses, Apache Hadoop emerged as an open-source distributed data processing technology. Apache Hadoop is a collection of open-source software for big data analytics that allowed large data sets to be processed with clusters of computers working in parallel.

The introduction of Hadoop was a watershed moment for big data analytics for two main reasons:

- It meant that some companies could shift from expensive, proprietary data warehouse software to in-house computing clusters running on free and open-source Hadoop.
- It allowed companies to analyze massive amounts of unstructured data (big data) in a way that wasn't possible before.

REMEMBER

Early data lakes built on Hadoop MapReduce and HDFS enjoyed varying degrees of success. Some early data lakes succeeded, while others failed due to Hadoop's complexity and other factors. Today, many modern data lake architectures have shifted from on-premises Hadoop to running Apache Spark in the cloud. Still, these initial attempts were important as these Hadoop data lakes were the precursors of the modern data lake.

Shortly after the introduction of Hadoop, Spark was introduced. Spark was the first unified analytics engine that facilitated large-scale data processing, SQL analytics, and ML. Spark was also 100 times faster than Hadoop. When Spark was introduced, it took the idea of MapReduce a step further, providing a powerful, generalized framework for distributed computations on big data. Over

time, Spark has become increasingly popular among data practitioners, largely because it's easy to use, performs well on benchmark tests, and provides additional functionality that increases its utility and broadens its appeal.

Today, many modern data architectures use Spark as the processing engine that enables data engineers and data scientists to perform ETL, refine their data, and train ML models. Cheap blob storage (AWS S3 and Microsoft Azure Data Lake Storage) is how the data is stored in the cloud, and Spark has become the processing engine for transforming data and making it ready for BI and ML.

Why a Traditional Data Lake Isn't Enough

While suitable for storing data, data lakes lack some critical features:

- They don't support transactions.
- They don't enforce data quality.
- Their lack of consistency and isolation makes it almost impossible to mix appends and reads, and batch and streaming jobs.

For these reasons, many of the promises of the data lakes haven't materialized, and in many cases, it has led to a loss of many of the previous benefits of data warehouses.

However, the need for a flexible, high-performance DM system hasn't decreased. More than ever, companies require systems for diverse data applications, including SQL analytics, real-time monitoring, and ML. Most of the recent advances in AI have been in better models to process unstructured data (text, images, video, audio). Still, these are precisely the types of data for which a data warehouse isn't optimized.

WARNING

A common approach is to use multiple systems — a data lake, several data warehouses, and other specialized systems such as streaming, time-series, graph, and image databases to address the increasing needs. However, having a multitude of systems introduces additional complexity and, more importantly, introduces delays as data professionals consistently need to move or copy data between different systems.

IN THIS CHAPTER

- Describing the lakehouse architecture
- Comparing the lakehouse approach to data warehouses and data lakes
- Tackling challenges with the lakehouse approach

Chapter 2
Explaining the Concept of a Lakehouse

New systems are beginning to emerge in the industry that address the limitations with and complexity of the two different stacks for business intelligence (BI) (data warehouses) and machine learning (ML) (data lakes). A lakehouse is a new architecture that combines the best elements of data lakes and data warehouses.

REMEMBER

Lakehouses are enabled by a new system design using similar data structures and data management (DM) features to those in a data warehouse, directly on the kind of low-cost object storage used for data lakes. They're what you would get if you had to redesign data warehouses in the modern world, now that cheap and highly reliable storage (in the form of object stores) are available.

In this chapter, you discover all you need to know about lakehouses, including what types of problems this approach helps to overcome and why this is significantly different from other DM solutions.

Sorting Out the Concept of a Lakehouse

A lakehouse is a new DM architecture that enables users to do everything from BI, SQL analytics, data science, and ML on a single platform. To understand the concept of a lakehouse, you should first dive deeper into the challenges of data lakes:

- **Appending data is hard:** Users want their changes to appear all at once. However, appending new data into the data lake while also trying to read it causes data consistency issues.
- **Modification of existing data is difficult:** You need to be able to modify and delete specific records, especially with GDPR and CCPA. Unfortunately, it takes a rewrite of petabytes on the data lake to make specific changes.
- **Jobs failing mid-way:** Job failures usually go undetected for weeks or months and aren't discovered until later when you're trying to access the data and find that some of it's missing.
- **Real-time operations are hard:** Combining real-time operations and batch leads to inconsistencies because data lakes don't support transactions.
- **It's costly to keep historical data versions:** Regulated organizations need to keep many versions of their data for auditing and governance reasons. They manually make a lot of copies of the data, which is time intensive and costly.
- **Data lakes make it difficult to handle large metadata:** If you have petabytes of data in the data lake, then the metadata itself becomes gigabytes and most data catalogs can't support those sizes.
- **You have "too many files" problems:** Because data lakes are file-based, you can end up with millions of tiny files or a few gigantic files. In either case, this impacts performance negatively.
- **Data lakes perform poorly:** It's hard to get great performance with big data. You have to use a number of manual techniques like partitioning that are error-prone.
- **You may have data quality issues:** All the challenges eventually lead to data quality issues. It becomes harder to ensure that your data is correct and clean.

The lakehouse takes an opinionated approach to building data lakes by adding data warehousing attributes — reliability, performance, and quality, while retaining the openness and scale of data lakes. It supports

- **ACID transactions:** Every operation is transactional. This means that every operation either fully succeeds or aborts. When aborted, it's logged, and any residue is cleaned so you can retry later. Modification of existing data is possible because transactions allow you to do fine-grained updates. Real-time operations are consistent, and the historical data versions are automatically stored. The lakehouse also provides snapshots of data to allow developers to easily access and revert to earlier versions for audits, rollbacks, or experiment reproductions.
- **Handling large metadata:** Lakehouse architecture treats metadata just like data, leveraging Apache Spark's distributed processing power to handle all its metadata. As a result, it can handle petabyte-scale tables with billions of partitions and files with ease.
- **Indexing:** Along with data partitioning, lakehouse architecture includes various statistical techniques like bloom filters and data skipping to avoid reading big portions of the data altogether, and therefore deliver massive speed ups.
- **Schema validation:** All your data that goes into a table must adhere strictly to a defined schema. If data doesn't satisfy the schema, it's moved into a quarantine where you can examine it later and resolve the issues.

Comparing a Lakehouse to Other DM Solutions

In the past, decision making was based mainly on structured data from operational systems. It's essential for a DM system of today to be much more flexible and also support unstructured data in basically any format, enabling advanced ML techniques.

REMEMBER

With the lakehouse approach, this flexibility is achieved through deeply simplifying the data infrastructure to enable accelerating innovation. This is especially important at a time when ML is revolutionizing all industries and demands an elastic infrastructure supporting speed and operational efficiency.

Figure 2-1 gives a high-level overview of three different approaches to DM. The first two — data warehousing and data lakes — have been leading the industry during different time periods. The lakehouse approach is a brand-new architecture for 2020 and comes with some obvious architectural differences. This section sorts out the differences in more detail.

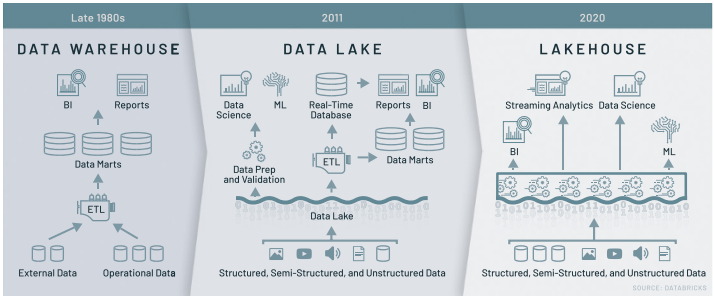


FIGURE 2-1: The differences between a data warehouse, a data lake, and a lakehouse.

Types of data that can be used

One significant difference between these approaches regards what data it's intended for. Although a data warehouse only handles structured data, both a data lake and a lakehouse handle structured data, semi-structured data, and unstructured (raw) data. In the current data landscape in most companies, that's not only a good capability to have, but also it's an essential one.

Cost of DM operations and vendor lock-in

Vendor lock-in refers to how applications or system solutions, which store data in proprietary formats, can make it hard for other systems to use the data. It can cause a customer to become dependent on a particular vendor for products and services and make the customer unable to use another vendor solution without substantial costs for switching solutions. This problem can lead

to companies being forced to create multiple data copies to make data accessible to other third-party systems. This approach isn't good for making your data architecture future-proof.

WARNING

In the DM space, the data warehouse approach comes with significant operational cost and vendor lock-in, which makes the data warehouse solution inflexible and less cost efficient than both the data lake and the lakehouse approach because both approaches come with low operational cost and no vendor lock-in. In a lakehouse, data is stored in open data formats, which is a good foundation for making the data architecture future-proof.

Ability to scale

Scalability in DM solutions refers to the capability of the system to handle a growing amount of data as well as increased workload. Scalability is essential in that it contributes to competitiveness, efficiency, reputation, and quality. These factors are especially important to consider for small businesses because they have the biggest growth potential.

When analyzing the data warehouse approach, you soon realize that it comes with clustered or coupled storage, and the compute resources don't scale. Clustered or coupled storage refers to the use of two or more storage servers working together to increase performance, capacity, or reliability.

TECHNICAL STUFF

Clustering distributes workloads to each server, manages the transfer of workloads between servers and provides access to all files from any server regardless of the physical location of the file. This should be compared to the data lake and the lakehouse, which are both highly scalable and use low-cost scalable storage and on-demand elastic compute.

Support for BI and ML

While the main capabilities of a data warehouse may once have addressed the data related challenges at hand by offering support for SQL queries, BI reporting, and dashboarding, the DM challenges of today are quite different. The data warehouse approach isn't future-proof because it's missing support for predictions, real-time (streaming) data, flexible scalability, and managing raw data in any format.

The data lake approach, on the other hand, may, at a first glance, look almost the same as the lakehouse approach with its support for low operational cost, flexibility, scalability, and allowing storage of the raw data in any format needed for ML. But it has several drawbacks (I cover these in Chapter 1).

REMEMBER

But the key benefit with the lakehouse is that it allows you to unify all your data and run all of your analytics and ML in a single place.

Solving Problems with a Lakehouse

A lakehouse enables business analytics and ML at a massive scale. The challenges that can be overcome with a lakehouse approach are several:

- **Unifying data teams:** One of the biggest benefits of a lakehouse is that it unifies all your data teams — data engineers, data scientists, and analysts — on one architecture.
- **Breaking data silos:** A lakehouse approach facilitates breaking data silos by providing a complete and firm copy of all your data in a centralized location. This enables everyone in your organization to access and manage both structured and unstructured data.
- **Preventing data from becoming stale:** In a continuous manner, the lakehouse approach can process batch and streaming data, updating tables and dashboards in near real time so your data is always generating value, staying updated, and never becoming stale.
- **Reducing the risk of vendor lock-in:** The lakehouse approach uses open formats and open standards that allow your data to be stored independent of the tools you currently use to process it, making it easy at any time to move your data to a different vendor or technology.

IN THIS CHAPTER

- Describing benefits with the lakehouse approach
- Listing benefits for BI activities
- Sorting out the value for exploratory data science needs
- Supporting your ML efforts

Chapter 3
Capturing the Value of the Lakehouse Approach

The main benefit with a lakehouse is that all data is kept within its open format, which acts as a common storage medium across the whole architecture. The tools used to process and query that data are flexible enough to use either approach — the adaptable, schema-on-read querying that comes with engines like Apache Spark, or a more structured, governed approach like that of a SQL-based data system. This chapter describes these benefits in more detail.

Defining Values for Building Reliable Data Lakes

With the lakehouse approach, you can build reliable data lakes by unifying data pipelines across both batch and streaming data. Lakehouses enable your efforts with regards to efficiently ingesting data, building scalable data pipelines, running them in production, and automating these cost-effective processes for simplicity, reliability, and scale.

Common data reliability problems

WARNING

Data reliability is a big hindrance for extracting value from data across the enterprise. Failed jobs can corrupt and duplicate data with partial writes. Multiple data pipelines reading and writing concurrently to your data lake can compromise data integrity. The situation that many companies end up in with their data pipeline efforts are complex, redundant systems with significant operational challenges to process both batch and streaming data jobs, where requirements on streaming data is especially challenging. This often results in unreliable data processing jobs that require manual cleanup and reprocessing after failed jobs, which in turn causes a lot of lead time delay.

Figure 3-1 shows an example of a complex and inefficient data pipeline setup with the purpose of serving both batch and streaming data jobs.

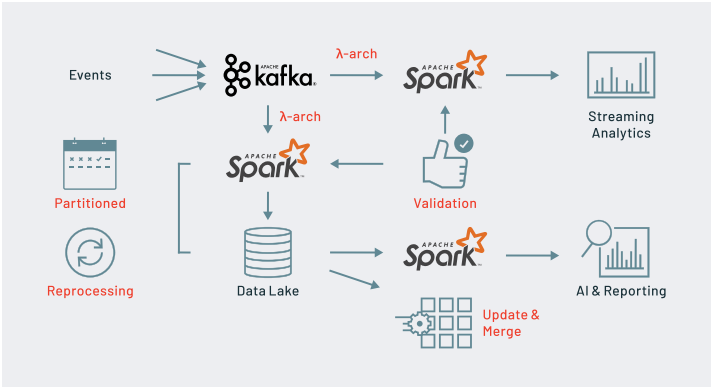


FIGURE 3-1: An example of an inefficient data pipeline setup.

Many companies experience long data processing times and increased infrastructure costs from inefficient data pipelines, many times due to static infrastructure resources incurring expensive overhead costs and limited workload scalability. This, in turn, results in nonscalable processes with tight dependencies, complex workflows, and system downtime. All in all, this outcome isn't desirable for any company.

Data reliability benefits with the lakehouse approach

REMEMBER

With the lakehouse approach, you benefit from a unified and simplified architecture that brings reliability across all your batch and streaming data. You experience robust data pipelines that ensure data reliability with atomicity, consistency, isolation, durability (ACID) transaction, and data quality guarantees throughout.

Figure 3-2 shows a more streamlined and efficient data pipeline setup as part of a lakehouse approach. In this setup, you experience reduced compute times and costs with a scalable cloud runtime. This process is powered by highly optimized Spark clusters and elastic cloud resources that can intelligently auto-scale up with increased workloads and auto-scale down for cost savings.

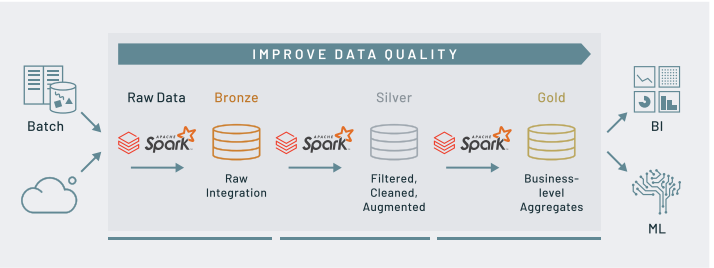


FIGURE 3-2: An example of an efficient data pipeline setup as part of a lakehouse approach.

The lakehouse approach to data pipelines offers modern data engineering best practices for improved productivity, system stability, and data reliability, including streaming data to enable reliable real-time analytics.

Delta Lake is an open-source storage layer that brings data reliability to your existing data lake by providing.

TECHNICAL STUFF

- Atomicity, consistency, isolation, durability (ACID) transactions

  ACID transactions ensure that multiple data pipelines can simultaneously read and write data reliably on the same table.
- Scalable metadata handling
- Unified streaming and batch data processing

The data reliability guarantees provided by Delta Lake across batch and streaming enable new data architecture patterns. Streaming data pipelines can automatically read and write the data through the different tables, with data reliability ensured by Delta Lake. This results in data continuously flowing through your data lake and providing end-users with the most complete, reliable, and up-to-date data available.

Specifying Benefits for Business Intelligence

The ability to create a central and single source of truth for your business intelligence (BI) application to execute on is usually an important goal for a company. To get this right, how you collect and ingest data into your data lake to serve different use cases is important.

Problems with a traditional BI approach

Some of the most common challenges that companies face today in BI is that the data in a DW is incomplete and stale. Neither can you put streaming data into a DW.

WARNING

Built-in complexities and costs are associated with transferring data from data lakes to data warehouses for ETL workloads, and proprietary data formats prevent direct data access with other tools and increase the risk of vendor lock-in. There is also increased cost and governance challenges associated with managing multiple copies of data and security models a cross your infrastructure.

Enabling BI on all your data

REMEMBER

With the lakehouse approach comes modern data lake handling where all structured and unstructured data are available through a central access point. This means immediate access to complete and most recent data. This approach enables cost-effective pipelines to progressively refine data through data lake tables with open data formats to ensure that data is accessible across all tools and teams, reducing the risk of vendor lock-in. The lakehouse software architecture enables any tool to access data with its open

APIs. Its SQL interface is fundamental for analytics, reporting, and dashboards.

Lakehouses facilitate curated and shared data across data science, ML, and BI. Industry-standard open-source tooling is used for sharing data and resources between data scientists and data analysts. The build once, access many times across use cases are used for a consolidated administration and high-level of self-service approach throughout the data architecture.

Describing the Payoff for Exploratory Data Science and ML

Data scientists face numerous challenges throughout the data science workflow, which is hindering productivity. As organizations continue to become more data-driven, a collaborative environment for easier access and visibility into the data, models trained against the data, reproducibility of results, and insights uncovered within the data are critical; however, this collaborative environment isn't that easy to achieve.

Barriers toward data science productivity

Data science collaboration and data exploration at scale are generally both difficult and costly, and companies tend to spend too much time managing the infrastructure and DevOps aspects.

WARNING

There is often a need to stitch together various open-source libraries and tools for further analytics, and the multiple handoffs between data engineering and data science teams are error-prone and increase risks. Many companies also find it hard to transition from local to cloud-based development environments due to the complex ML environments and dependencies built into the data science workflow.

Everything you can do to simplify the ML life cycle is something to strive for. However, the reality is that most companies are stuck in organizational and technological silos that are difficult to break out of. The sheer amount and diversity of ML frameworks needed also makes it hard to manage ML environments. The disparate tools and process steps, from data preparation to

experimentation and production, make handoffs difficult to manage efficiently between teams. Due to the data dependency and sometimes lack of model transparency, there is also a built-in risk from a security and compliance perspective. In ML, it's hard to track experiments, models, dependencies, and artifacts, which makes it hard to reproduce results.

Gains for data science collaboration


TIP

With the lakehouse approach, the key benefit is that you gain quick access to clean and reliable data for downstream analytics and get one-click access to pre-configured clusters from the data science workspace. Lakehouses also

- Facilitate tasks of preparing data sets, training models with large datasets, and tracking data versions used to build models
- Allow you to bring your own environment and multi-language support for maximum flexibility
- Migrate or execute your code remotely on pre-configured and customizable ML clusters
- Enable a unified approach to streamline the end-to-end data science workflow from data preparation to modeling and insights sharing
- Give you one-click access to ready-to-use, optimized, and scalable ML environments across the life cycle
- Simplify handoffs between teams and different steps in the ML life cycle as it uses one platform for data ingest, feature development, model building, tuning, and deploying models in production, as well as monitoring models in production that requires streaming analytics
- Track experiments, code, results, and artifacts and manage models in one central hub
- Meet compliance needs with fine-grained access control, data lineage, and versioning of data and models

IN THIS CHAPTER

- Launching you lakehouse with Databricks
- Introducing Delta Lake and Delta Engine
- Leveraging Databricks Unified Analytics Platform
- Learning from a customer case study

Chapter 4
Building a Modern Cloud Data Platform with Databricks

This chapter guides you through how Databricks applies the lakehouse architecture to deliver a modern cloud data platform

Getting Started with Your Lakehouse by Using Databricks

Because today's analytics use cases range from building simple SQL reports to more advanced machine learning (ML) predictions, you need to build a central data lake in an open format with data from all your data sources and make it accessible for various use cases. Achieving that at scale is beyond the reach of most organizations today because of the complexity and cost involved in meeting these requirements. But with Databricks' open, unified data platform that simplifies data and artificial intelligence (AI)

for massive-scale data engineering, collaborative data science, full life cycle ML, and business analytics, this becomes achievable.


TIP

A lakehouse utilizes inexpensive cloud object storage as the data storage layer, which is capable of scaling to virtually any size at low cost. For example, you can easily set this up by creating an AWS S3 Bucket or a Microsoft Azure Data Lake Storage Gen2 repository. To move over your data from current applications, databases, data warehouses, and other data stores, you can use Databricks Ingest, a service that quickly and easily loads data into your lakehouse.

Utilizing Delta Lake to Add Reliability to Your Lakehouse

Delta Lake addresses the data reliability problems that have plagued data lakes, making them data swamps. The open-source storage layer that Delta Lake provides brings improved reliability to data lakes. Delta Lake on Databricks allows you to configure data lakes based on your workload patterns and provides optimized layouts and indexes for fast, interactive queries and sits on top of object storage. The format and the compute layer help simplify building big data pipelines and increase the overall efficiency of your pipelines.

Figure 4-1 runs Delta Lake on top of your existing data lake and is fully compatible with Apache Spark APIs.

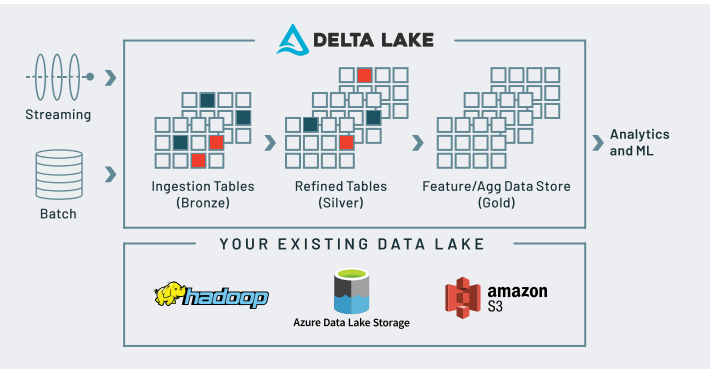


FIGURE 4-1: Delta Lake is an open-source storage layer that brings improved reliability to the lakehouse.

Ever since Databricks open sourced Delta Lake in 2019, thousands of organizations have lakehouses in an open format much more reliably and efficiently than before. Delta Lake's atomicity, consistency, isolation, durability (ACID) transactions, and efficient indexing are critical for exposing the data for various access patterns, ranging from ad-hoc SQL queries in business intelligence (BI) tools, to ML models with Python and R.

REMEMBER

This pattern of building a central, reliable, and efficient single source of truth for data in an open format for use cases ranging from BI to ML with decoupled storage and compute is the foundation of the lakehouse approach.

However, it's important to build data reliability into a lakehouse from the get-go to prevent downstream data corruption issues. In general, you need to manage two data ingestion scenarios:

- **Data ingestion from third-party sources:** You typically have valuable user data in various internal data sources. Databricks Data Ingestion Network enables an automated way to populate your lakehouse from hundreds of data sources into Delta Lake.
- **Data ingestion from cloud storage:** You already have a mechanism to pull data from your source into cloud storage. As new data arrives in cloud storage, you can load this new data by using the Delta Lake Auto-Loader capability in Databricks.

Adding Delta Engine to Bring Performance to Your Lakehouse

Delta Engine brings high performance to all workloads on Delta Lake, including ETL data pipelines, SQL analytics, real-time analytics, data science, and ML. Delta Engine is fully compatible with Spark APIs. Delta Engine comprises three critical components:

- **Vectorized query engine:** This includes a new massively parallel processing (MPP) engine built from scratch in C++ and a fully vectorized engine for modern Single Instruction, Multiple Data (SIMD) hardware. It's optimized for modern workloads, eliding null checks and faster string processing.

- **Improved query optimizer:** An improved cost-based optimizer for better physical plans is included as well as an adaptive query execution that can dynamically re-plan while executing. This also includes dynamic partition pruning and runtime filters to skip irrelevant data.
- **Intelligent caching:** Delta Engine automatically caches input data and load balances across a cluster. It also leverages the advances in non-volatile memory express (NVMe) solid-state drive (SSD) hardware with state-of-the-art columnar compression techniques and can improve interactive and reporting workloads performance by up to ten times.

Leveraging Databricks Unified Data Analytics Platform as Your Lakehouse

The Databricks Unified Analytics Platform has the architectural features of a lakehouse. Companies who want to build and implement their own systems have access to open-source file formats like Delta Lake that are suitable for building a lakehouse. Figure 4-2 shows you the simplicity of the lakehouse approach in the Databricks Unified Analytics Platform.

Users of a lakehouse enabled by Databricks Unified Data Analytics Platform also have access to a variety of standard tools (Spark, Python, R, ML libraries) for non-BI workloads like data science and ML. Data exploration and refinement are standard for many analytics and data science applications. Delta Lake is designed to let users incrementally improve the quality of data in their lakehouse until it's ready for consumption.


TIP

To make ML management more efficient and get it under business control, your company needs a solution to orchestrate and manage its models in a way that may speed up model deployment without losing model governance. Databricks offers this support, which provides a business process management solution with support for operationalizing ML models. This includes support for model build, register, test, compare, approve, publish, monitor, and, if needed retraining those models in an automated and controlled

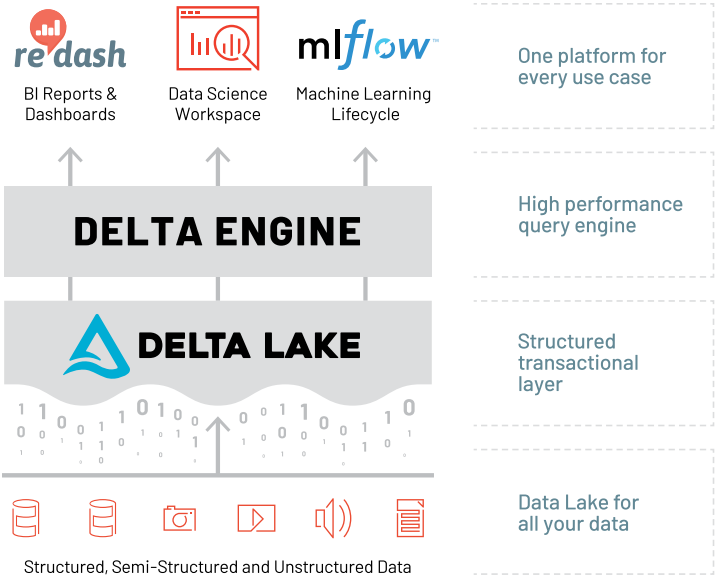


FIGURE 4-2: The Databricks Unified Data Analytics Platform.

manner at the same time. This automated architecture is a build-one, use-many solution that reduces manual human intervention and accelerates customer capabilities of operationalizing ML models. This is achieved through MLflow, an open-source platform developed by Databricks to help manage the complete ML life cycle with enterprise reliability, security, and scale.

Results from various Databricks implementations show that data team productivity improves by 20 percent with the lakehouse solution.

Sharing a Customer Case Study

A global leader in technology and entertainment needed a cloud data platform. This company had millions of residential broadband customers using video, high-speed Internet, and phone services. It also had massive volumes of data based on billions of events from people's entertainment systems and voice remotes,

which were generating petabytes of data that needed to be prepared for analysis. The company also had a setup with fragile, complicated data pipelines that frequently stopped working and were hard to recover.

Poor data science collaboration also existed in the organization with globally dispersed data scientists working in different scripting languages, and the company struggled to share and reuse code. A friction existed between development and deployment responsibilities, where the development teams wanted to use the latest tools and models, and the operations guys only wanted to deploy on proven infrastructure.

The client needed the lakehouse approach, and Databricks helped design the company's new cloud data platform.

The solution

By using the Databricks Unified Analytics Platform, the technology and entertainment company built rich data sets at massive scale. The solution also allowed the organization to optimize ML at scale, streamline workflows across teams, foster collaboration, reduce infrastructure complexity, and deliver superior customer experiences.

A simplified infrastructure management also reduced operational costs through automated cluster management and cost-efficiency features such as autoscaling and spot instances. By introducing collaborative workspaces throughout the company, interactive notebooks helped improve cross-team collaboration and data science creativity, allowing it to greatly accelerate model prototyping for faster iteration.

Reliable ETL at scale through using Delta Lake provided efficient analytics pipelines at scale that could reliably join historic and streaming data for richer insights.

The results

Databricks helped the company to realize many positive results:

- Created a highly innovative and award-winning viewer experience with intelligent voice commands that boosted engagement
- Reduced compute costs by ten times by replacing 640 machines with 64, while improving query performance
- Higher data team productivity through improved collaboration across data teams globally, made possible by a single interactive workspace
- Faster model deployment via reduced deployment times from weeks to minutes

IN THIS CHAPTER

- Getting a single cloud data platform
- Making data-driven decisions across the organization
- Enabling BI and ML on all your data
- Reducing costs by consolidating systems

Chapter 5
Ten Reasons Why You Need a Lakehouse Approach

The concept of lakehouses is in the early stages, but you can realize many benefits identified with this approach:

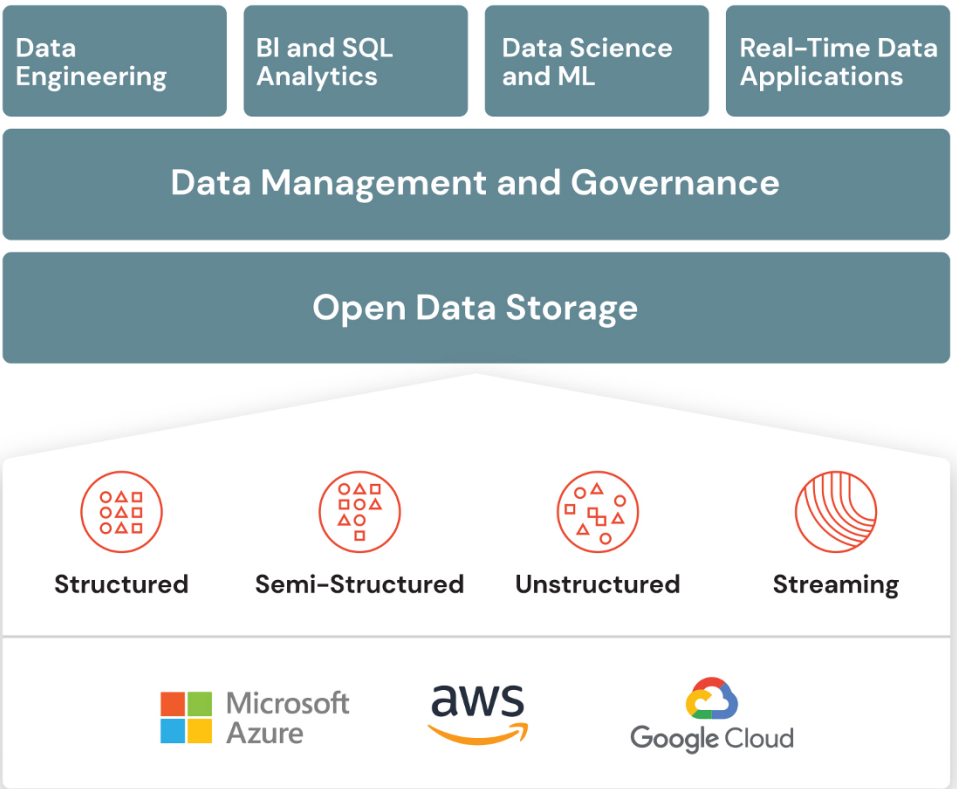
- **Enables a single combined cloud data platform:** With a lakehouse, all data is kept within its lake format; it's a common storage medium across the whole architecture.
- **Unifies data warehousing and machine learning (ML):** One platform for data warehousing and ML supports all types and frequency of data.
- **Increases data team efficiency:** Lakehouses are enabled by a new system design that implements similar data structures and data management features to those in a data warehouse, directly on the kind of low-cost storage used for data lakes. Merging them into a single system means that ML teams can move faster because they're able to use data without needing to access multiple systems.

- **Reduces cost:** With the lakehouse approach, you have one system for data warehousing and ML. Multiple systems for different analytics use cases are eliminated. You can store data in cheap object storage such as Amazon S3, Azure Blob Storage, and so on.
- **Simplifies data governance:** A lakehouse can eliminate the operational overhead of managing data governance on multiple tools.
- **Supports data versioning:** Uniqueness can be used by data consumers to determine whether data has changed (and how) over time and specifically which version of a data set they're working with.
- **Simplifies ETL jobs:** With the data warehousing technique, the data has to be loaded into the data warehouse to query or to perform analysis. But by using the lakehouse approach, the ETL process is eliminated by connecting the query engine directly to your data lake.
- **Removes data redundancy:** The lakehouse approach removes data redundancy by using a single tool to process your raw data. Data redundancy happens when you have data on multiple tools and platforms such as cleaned data on data warehouse for processing, some meta-data on business intelligence (BI) tools, and temporary data on ETL tools.
- **Enables direct data access:** Data teams can use a query engine to query the data directly from raw data, giving them the power to build their transformation logics and cleaning techniques after understanding basic statistical insights and quality of the raw data.
- **Connects directly to BI tools:** Lakehouses enables tools, such as Apache Drill and supports the direct connection to popular BI tools like Tableau, PowerBI, and so on.
- **Handles security:** Data related security challenges are easier to handle with a simplified data flow and single source of truth approach.

databricks

Databricks Lakehouse Platform

One simple platform to unify all your data, analytics and machine learning workloads



Sign up for a free trial: databricks.com/try

Build a modern cloud data platform for all data

Today, data is easier to access than ever, but the volume, variety, and ways it's used has increased greatly. In particular, the rise in analytics and machine learning (ML) requires using massive amounts of data across multiple data sources, and this surge puts high demands on your data management support team. Enter the lakehouse that combines the best of data warehouses and data lakes to simplify enterprise data in the cloud, accelerates innovation, and increases data team productivity up to 20 percent.

Inside...

- Understand the value of a lakehouse
- Apply an open source data platform
- Move faster with lower costs
- Collaborate better with fewer data silos

databricks

Ulrika Jägare is Head of AI at Ericsson North America. She has a decade of experience in data, analytics, and AI/ML, as well as 20 years in telecommunications. She's the author of *Data Science Strategy For Dummies*, as well as several other *For Dummies* custom titles.

ISBN: 978-1-119-85633-7
Not For Resale

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.