The Guide to

DATABRICKS OPTIMIZATION

intel Granulate

Table of Contents

Executive Summary

In today's world, data is being generated at an exponential rate. To process this data, companies need robust platforms that can scale to meet their requirements. Databricks is one such platform that provides a managed Spark service, allowing organizations to scale their big data processing capabilities without having to worry about the underlying infrastructure.

In this guide, we will explore some of the challenges, best practices and background knowledge necessary for optimizing Databricks and potentially reducing your cloud costs.

Databricks is a unified data analytics platform that combines big data processing, machine learning, and collaborative analytics tools in a cloud-based environment. It is designed to simplify and accelerate data-driven workflows, enabling organizations to gain insights and make data-driven decisions more efficiently. The platform offers a collaborative workspace, supports multiple programming languages, and integrates with popular data storage and processing systems.

Databricks is different from other platforms like Amazon EMR and GCP Dataproc. It is a SaaS platform but you can run the actual engine ("compute") on your CSP in your own VPC. Databricks manages clusters for you instead of using dynamic allocation. It also has its own scheduler, notebook solution, and data viewer, making it a one-stop solution for big data processing needs.

Databricks is an excellent platform for big data processing needs. However, optimizing Databricks usage is essential to ensure that you are not overspending on infrastructure costs. By following the best practices outlined in this guide and using **Intel Granulate's Big Data solutions**, you can optimize your Databricks usage and save costs.

Databricks Use Cases and Components

Databricks Features

Databricks is used for various data-related tasks and offers a wide range of functionalities, including:

Data engineering

Databricks allows users to ingest, process, clean, and transform large volumes of structured and unstructured data using Apache Spark. It supports ETL (extract, transform, load) processes and optimizes data pipelines for performance and scalability.

Data analytics

With Databricks, users can perform advanced analytics, including real-time stream processing, SQL queries, and interactive data exploration. It offers visualization tools and integration with popular BI tools like Tableau and Power BI for creating dashboards and reports.

Machine learning

Databricks provides an environment for developing, training, and deploying machine learning models. It offers built-in algorithms, support for popular ML libraries like TensorFlow and PyTorch, and tools for hyperparameter tuning, feature engineering, and model evaluation.

Collaboration

The platform features a collaborative workspace where data engineers, data scientists, and business analysts can work together on shared notebooks using different programming languages like Python, Scala, R, and SQL. Version control, commenting, and access controls facilitate teamwork and knowledge sharing.

Key Components of Databricks

Databricks works by integrating various open-source technologies and proprietary components to provide a unified data analytics platform. Some key components include, but are not limited to:

Apache Spark

At the core of Databricks is Apache Spark, a powerful open-source distributed computing framework for processing large-scale data. Spark uses SQL queries or dataframe APIs and supports multiple programming languages, including Python, Scala, and R.
Databricks builds on Spark by offering a managed, optimized, and cloud-based version of the framework, simplifying deployment, scaling, and resource management.

Delta Lake

Delta Lake is an open-source storage layer that brings ACID transactions (atomicity, consistency, isolation, durability) and other data reliability features to big data workloads. It sits on top of existing data lake storage systems (e.g., Amazon S3, Azure Data Lake Storage) and enables versioning, schema enforcement, and data indexing. This makes it easier to manage and maintain data consistency, quality, and performance in data pipelines.

Databricks on AWS, Azure and Google Cloud

Databricks can be deployed on various cloud computing platforms. The platform's compatibility with multiple cloud providers allows organizations to leverage Databricks' capabilities within their preferred cloud infrastructure.

Here is a brief review explaining how Databricks works across different cloud platforms:

Databricks on AWS

Databricks offers a fully managed service on the Amazon Web Services (AWS) platform. It integrates with various AWS services such as Amazon S3 for data storage, AWS Glue for data cataloging, and Amazon Redshift for data warehousing.

It enables users to benefit from the scalability, performance, and reliability of AWS while leveraging Databricks' features for data engineering, analytics, and machine learning. This deployment supports single sign-on (SSO) and RBAC, as well as integration with AWS PrivateLink for secure, private connectivity between Databricks and other AWS services.

Databricks on Azure

Databricks is also available as a first-party service on Microsoft Azure, known as Azure Databricks. It provides seamless integration with Azure services such as Azure Data Lake Storage, Azure Blob Storage, Azure Synapse Analytics, and Azure Machine Learning.

Azure Databricks features a native integration with Azure Active Directory (AD), enabling SSO and centralized access management across Azure services. As a Microsoft-branded service, Azure Databricks offers a consistent experience for organizations that rely heavily on the Azure ecosystem.

Databricks on Google Cloud

In 2021, Databricks announced a partnership with Google Cloud to offer a fully managed Databricks service on the Google Cloud Platform (GCP). This partnership enables organizations to utilize Databricks alongside Google Cloud services such as Google Cloud Storage, BigQuery, and Google Cloud AI Platform.

Databricks on Google Cloud supports integration with Google Cloud's data and AI services, Anthos for hybrid and multi-cloud deployments, and Google's global network for enhanced performance and security.

Databricks Performance Issues and How to Solve Them

Databricks is designed for high-performance processing, analytics, and machine learning tasks. However, like any platform, performance issues can arise due to various factors. Identifying and resolving these issues is essential to maintain optimal performance. Here are some common performance issues in Databricks and suggestions for optimization:

Data Skew

Data skew occurs when the data is unevenly distributed across partitions, causing some tasks to take longer than others. This can lead to performance bottlenecks and slow down job execution.

How to optimize

Identify and address the root cause of data skew, which may involve repartitioning the data, using salting techniques, or using bucketing to distribute the data more evenly across partitions.

Inefficient Data Formats

Using inefficient or uncompressed file formats can increase I/O overhead and slow down query performance.

How to optimize

Convert data to more efficient formats like Parquet, which offer built-in compression, columnar storage, and predicate pushdown to improve performance.

Inadequate Caching

Not caching frequently accessed data in memory can cause repeated disk I/O operations, leading to slow query execution.

How to optimize

Use Spark's caching features to persist frequently accessed data in memory, which will speed up iterative algorithms and queries on large datasets.

Large Shuffles

Shuffling large amounts of data between Spark stages can cause network congestion and slow down job execution.

How to optimize

Optimize your queries and transformations to minimize shuffling. This can include using techniques like broadcast joins, filtering data early in the processing pipeline, and using partition-aware operations.

Inefficient Queries

Writing inefficient or suboptimal queries can result in slow query execution and poor performance.

How to optimize

Analyze and optimize your queries using techniques like predicate pushdown, partition pruning, and query rewrites. Use the Databricks Query UI or Spark UI to identify performance bottlenecks and analyze query plans.

Suboptimal Resource Allocation

Improper allocation of resources like CPU, memory, and storage can lead to performance issues.

How to optimize

Ensure your Spark clusters have adequate resources to handle the workload. Monitor resource usage and adjust the cluster size, executor configurations, and memory settings accordingly.

Garbage Collection (GC) Settings

Inappropriate GC settings can cause frequent full GC events, leading to performance issues and even job failures.

How to optimize

Monitor GC activity using Spark UI and adjust the JVM settings, such as the GC algorithm and heap size, to minimize the impact of garbage collection on performance.

Outdated Software Versions

Using outdated versions of Databricks, Apache Spark, or other libraries can lead to suboptimal performance due to missing optimizations and bug fixes.

How to optimize

Regularly update Databricks and related software to take advantage of the latest performance improvements and bug fixes.

7 Best Practices for Databricks Optimization

Follow these guidelines to prevent waste and unnecessary expenses in your Databricks platform:

1 Turn off clusters that are not in use and enable auto-termination

Databricks allows you to turn off clusters that are not in use. This can save you a lot of money in terms of infrastructure costs. You can also enable auto-termination to automatically terminate clusters after a specified period of inactivity.

2 Share clusters between different groups

Databricks allows you to share clusters between different groups. This means that you can allocate resources to different groups as required. This can be useful if you have teams that have different data processing requirements.

3 Track costs

It is essential to keep track of your costs when using Databricks. This will help you ensure that you are not overspending on infrastructure costs. Databricks provides auditing features that allow you to monitor your usage and spend.

4 Consistently audit

Regular auditing of your Databricks usage is essential. This will help you identify which teams or users are spending the most and take corrective measures as required. You can also track the usage of active DBUs (Databricks Units), which is a measure of computational resources used by Databricks.

5 Enable spot instances

Databricks supports spot instances, which are unused EC2 instances that are available at a discounted price. This can help you save money on infrastructure costs.

6 Use photon acceleration

Databricks supports photon acceleration, which is a feature that speeds up SQL queries using vectorized execution. This feature is only effective if you are using the Spark SQL API.

7 Use Intel Granulate

Intel Granulate optimizes Apache Spark clusters and Java. This can help you optimize your cluster's performance and reduce infrastructure costs.

Optimizing Databricks Workloads With Intel Granulate

For the next level of optimizing Databricks workloads, there are autonomous, continuous solutions that can improve speed and reduce costs. Intel Granulate continuously and autonomously optimizes large-scale Databricks workloads for improved data processing performance.

With Intel Granulate's optimization solution, companies can minimize processing costs across Spark workloads in Databricks environments and allow data engineering teams to improve performance and reduce processing time.

By continuously adapting resources and runtime environments to application workload patterns, teams can avoid constant monitoring and benchmarking, tuning workload resource capacity specifically for Databricks workloads.

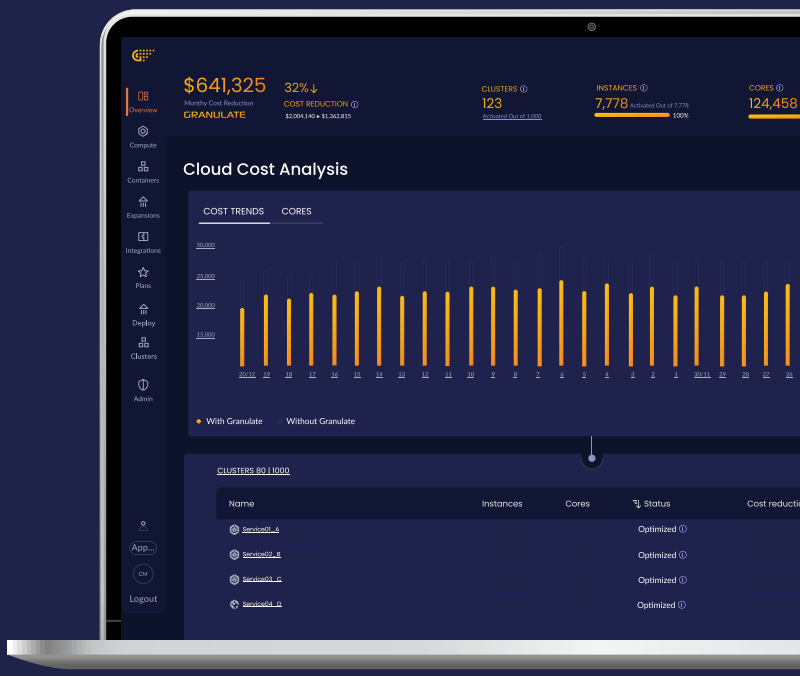
About Intel Granulate

Intel Granulate, an Intel company, empowers enterprises and digital native businesses with real-time, continuous application performance optimization and capacity management, on any type of workload, resulting in cloud and on-prem compute cost reduction.

Available in the AWS, GCP, Microsoft Azure and Red Hat marketplaces, the AI-driven technology operates on the runtime level to optimize workloads and capacity management automatically and continuously without the need for code alterations.

Intel Granulate offers a suite of cloud and on-prem optimization solutions, supporting containerized architecture, big data infrastructures, such as Spark, MapReduce, and Kafka, as well as resource management tools like Kubernetes and YARN. Intel Granulate provides DevOps teams with optimization solutions for all major runtimes, such as Python, Java, Scala, and Go. Customers are seeing improvements in their job completion time, throughput, response time, and carbon footprint while realizing up to 45% cost savings.

Book a Demo



intel Granulate

Visit us at Intel Granulate.io to learn more