Seekho Bigdata
Data is the New Oil

Catalyst Optimizer & Tungsten Execution

Seekho Bigdata Institute

https://www.seekhobigdata.com/

1. Catalyst Optimizer

- The Catalyst Optimizer is the query optimization framework in Spark.
- It works on Spark SQL and DataFrame API queries.
- Its primary goal is to improve the performance of queries by applying a series of transformations and optimizations.

How it works ?

Logical Plan:

- When you write a query, Catalyst first creates a logical plan, which is essentially a tree structure that represents the query.

Optimizations:

- Catalyst then applies various optimization techniques, like constant folding, predicate pushdown, filter pushdown, projection pruning, and join reordering to optimize the plan.

Physical Plan:

- Finally, the logical plan is transformed into a physical plan, which defines the actual steps needed to execute the query.
- The physical plan is then further optimized based on the cluster environment (e.g., partitioning).

Key Features:

- **Rule-based Optimization:** Catalyst applies predefined rules to improve the query.
- **Cost-based Optimization:** It estimates the cost of different execution plans and picks the one with the lowest cost.
- **Extensibility:** Developers can add custom optimization rules if needed.

2. Tungsten Execution

- The Tungsten Execution project focuses on low-level physical execution, aiming to optimize Spark's performance by reducing memory overhead and improving CPU efficiency.

- It's focused on improving the execution of Spark queries by leveraging off-heap memory management and binary processing.

How it works ?

Memory Management:

- Tungsten manages memory outside of Java's garbage collection (known as "off-heap memory"), which reduces overhead and improves performance for large datasets.

Code Generation:

- Tungsten uses whole-stage code generation to produce Java bytecode dynamically for each query.
- This helps remove the overhead of interpreting and analyzing queries repeatedly.

Cache and Shuffle Improvements:

- It improves data caching strategies, reduces the time spent on data shuffling (moving data between workers), and optimizes CPU usage.

Key Features:

Off-Heap Memory Management:

- Data is stored in memory that Spark controls directly, outside of Java's heap, reducing garbage collection overhead.

Efficient Code Generation:

- Tungsten generates optimized bytecode at runtime for data processing, reducing the need for repetitive interpretation and improving CPU usage.

Data Serialization:

- It uses efficient, custom serialization formats, which is much faster than the default Java serialization.

Why They Matter for Data Engineers:

- **Catalyst Optimizer** improves query execution by optimizing the logical and physical plans of SQL queries and DataFrame operations.
- It helps reduce time spent on data shuffling, improves execution efficiency, and reduces resource consumption.
- **Tungsten Execution** focuses on physical execution, providing better memory management and efficient code execution, making Spark run faster and more efficiently on large datasets.

Thank you

Seekho Bigdata Institute

https://www.seekhobigdata.com/
+91 99894 54737