Capstone Abstract - Workload-Aware Partitioning Advisor for Spark–Hive

Choosing between Hive partitioning and Spark repartitioning—and the partition count—for mixed workloads is often done by manual trial-and-error, which is time-consuming and inconsistent across teams. This project addresses that by benchmarking partitioning strategies in Apache Spark with Hive—comparing native Hive partitioning to explicit Spark repartitioning with 4, 16, and 32 partitions across multiple data sizes (5MB to 2GB) and three query types (aggregation, join, and window functions)—and implements a lightweight, workload-aware **Partitioning Advisor** that, given a data size and query type, automatically recommends whether to use Hive partitioning or Spark repartitioning and, in the latter case, the number of partitions (4, 16, or 32), with an optional optimization objective (runtime, CPU, or memory). The Advisor uses a rule-based lookup over an experiment summary table built from the project's runtime and resource-usage measurements; on all 12 (data size, query type) combinations in that summary, its recommendations match the empirically best configuration with 100% agreement. The results show that at larger scales (e.g., 2GB-equivalent) Hive partitioning consistently outperforms repartitioning in runtime, that join workloads are most sensitive to strategy choice, and that a moderate partition count (e.g., 4) often outperforms higher counts. The project is fully reproducible and provides a Dockerized Spark–Hive cluster, experiment and data-collection scripts, the summary-generation pipeline, the Advisor CLI, and an evaluation script, together with documentation of data sources and design choices for replication and extension. The Advisor's value is to give practitioners and pipelines a consistent, data-driven recommendation instead of ad hoc trial-and-error when choosing partition strategy and count.