IE494 BIG DATA PROCESSING

Project Proposal

Problem Area:

The advent of big data has created significant challenges for traditional relational database management systems (RDBMS), particularly in managing petabyte-scale datasets. Existing systems, like DBMS-X, struggle with scalability, performance, and cost efficiency, prompting organizations to explore alternative solutions.

Google's transition to the MapReduce framework offered a way to harness distributed computing for processing large datasets. However, the lack of SQL compatibility in MapReduce systems posed a barrier to adoption, as SQL remains the standard for data querying and manipulation.

Tenzing addresses this gap by integrating SQL capabilities within the MapReduce execution model. It aims to resolve key issues such as:

- **Scalability**: Efficiently processing large datasets across distributed systems.
- **Performance**: Achieving low-latency query execution.
- **SQL Compatibility**: Supporting SQL standards and extending functionality for advanced use cases.
- **Data Diversity**: Handling various data formats and sources.

By tackling these challenges, Tenzing seeks to enhance the MapReduce framework for large-scale data analytics.

Group Members:

- Jayswal Krishil - 202203040

Expected Outcomes:

The exploration of Tenzing as presented in the paper is anticipated to yield several key outcomes that contribute to the understanding of its impact on SQL querying in distributed systems:

1. **Comprehensive Understanding of Tenzing's Architecture**:
   - A detailed analysis of the high-level architecture of Tenzing, including how it integrates SQL capabilities with the MapReduce framework. This understanding will clarify how SQL queries are executed in a distributed environment and the overall workflow of query processing.

2. **Insights into Performance Optimizations**:
    - An in-depth examination of Tenzing's performance optimizations, including query execution plans and indexing strategies. The expected outcome is to identify the specific techniques that enable low-latency query performance and efficient resource utilization within a MapReduce context.
3. **Evaluation of Scalability Features**:
    - An evaluation of Tenzing's ability to handle massive datasets while maintaining system reliability. This includes understanding the mechanisms it employs for fault tolerance and how it achieves scalability across thousands of cores and petabyte-scale data.
4. **Analysis of SQL Extensions and Advanced Features**:
    - A comprehensive overview of the advanced SQL features supported by Tenzing, such as complex user-defined functions, nested queries, and support for multiple storage formats. The outcome will highlight how Tenzing enhances SQL for large-scale distributed environments without sacrificing performance.
5. **Comparison with Other SQL-on-MapReduce Systems**:
    - A comparative analysis of Tenzing's performance against other SQL-on-MapReduce systems, such as Hive and Pig. This comparison will illuminate the advantages Tenzing offers in terms of scalability, efficiency, and compatibility with SQL standards.
6. **Implications for Future Big Data Systems**:
    - The research is expected to provide insights into how Tenzing's design and functionality can influence the development of future big data systems, shaping the integration of SQL querying capabilities in distributed computing environments.

By achieving these outcomes, the study will not only enhance understanding of Tenzing's role in the evolution of SQL on MapReduce but also contribute valuable knowledge to the fields of data processing and distributed systems.

Selected Readings:

- Chattopadhyay, Biswapesh, et al. "Tenzing a sql implementation on the MapReduce framework." Proceedings of the VLDB Endowment 4.12 (20011): 1318-1327