Optimization

Database Administration
Lab Guide 2

2020/2021

Consider now the optimization of the toy benchmark application by introducing redundancy and adjusting server configuration. For each change, evaluate its the impact by (i) observing the query plan and (ii) running the benchmark for 1, 2, 4, 8, ... threads and plotting the corresponding scalability curve.

Steps

1. Use indexes to improve the performance for the current account operation.
2. Use a materialized view to improve the performance for the top 10 operation.
3. Determine what resource is limiting the performance for each configuration. Consider changes to memory configuration and CPU vs. I/O weights in the server configuration file.
4. Consider using prepared statements for each operation.

Questions

1. Are the indexes useful? Is clustering the index useful?
2. Is the materialized view useful? Which update strategy makes it useful?
3. Is changing the parameters useful?
4. How does each of these optimizations impact each operation alone?
5. Considering all the optimizations that you have performed, to what extent have you improved the maximum throughput of your application? What was the improvement in response time?
6. What is the relative impact of redundancy/algorithmic changes vs. configuration changes?

Learning Outcomes Recognize and explain the role of redundancy in relational databases, in particular, by contrasting it with implementation decisions in object-oriented programming. Discuss trade-offs between different optimization decisions. Plan, conduct, and justify the steps to optimize the performance of a relational application.

Useful Tools

- https://explain.depesz.com/s/dosQ
- https://explain.dalibo.com
- http://pgbadger.darold.net/