Compaction management in distributed key-value datastores

Muhammad Yousuf Ahmad
muhammad.ahmad2@mail.mcgill.ca

Bettina Kemme
kemme@cs.mcgill.ca

REFERENCES


RESULTS

– Compaction server assumes execution and overheads.
– Compactions are shorter; read latency improves.
– Cache misses less costly; reads faster over network vs. disk.
– Incremental warmup eliminates cache misses altogether.
– Multiple compaction servers allow for load balancing.

EXPERIMENTS

– YCSB update workload triggers compactions.
– YCSB read workload measures get/scan latency.
– 1x region server (RS), 1x compaction server (CS).

SOLUTION

≡ Compaction Offloading
– Offload compactions to specialized compaction servers.
– Dedicate region server resources to workload execution.

≡ Remote Caching
– Compaction server caches compaction results locally.
– Region server reads back results over network.

≡ Incremental Warmup
– Do not evict invalidated file blocks en masse.
– Gradually phase old data out, block by block.
– Replace with new data from remote cache.
– Sequential transfer (files are already sorted).

GOALS
1. Reduce compaction overheads on region server.
2. Prevent large spikes in read latency.

SCALABILITY

Fig. 1 – Standard Compaction

Fig. 2 – Remote Caching

Fig. 3 – Incremental Warmup

Fig. 4 – Standard
5x RS - No CS

Fig. 5 – Offloaded
5x RS - 1x CS

Fig. 6 – Under-Provisioned
10x RS - 1x CS

Fig. 7 – Balanced
10x RS - 2x CS

Compactions are distributed across two compaction servers.

ACKNOWLEDGEMENTS

This work is partially funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Ministère de l’Enseignement supérieur, Recherche, Science et Technologie, Québec, Canada.