

Architecture/Method/Idea/Concept/Algorithm Benchmarked:

Tema mog eksperimentalnog rada bila je 'Materialized view: OLAP query throughput from materialized view (data cube) vs facts table'. Materialized view je objekt baze podataka koji pohranjuje rezultate prethodno izračunatih upita, pružajući poboljšane odzive upita smanjenjem potrebe za kompleksnim i resursno intenzivnim upitima. Pohrana prethodno izračunatih rezultata omogućuje brže izvođenje upita, posebice u scenarijima koji uključuju kompleksne i resursno zahtjevne upite. Facts tablica je temeljna komponenta u skladištima podataka ili dimenzionalnim bazama podataka. To je vrsta tablice koja obično sadržava kvantitativne podatke, poznate kao činjenice, povezane s određenim poslovnim procesom ili aktivnošću.

U ovom benchmarku usporedili smo performanse upita OLAP kocke podataka iz Materialized viewa naspram direktnog upita Facts table. Koncept Materialized viewa poboljšava performanse sustava prethodnim izračunom i pohranom agregiranih rezultata, smanjujući potrebu za složenim upitima. Iako može značajno poboljšati performanse čitanja, nedostaci uključuju povećane zahtjeve za pohranom i potencijalnu latenciju pri održavanju materijaliziranog prikaza s čestim ažuriranjima podataka. Ovaj pristup je koristan u scenarijima gdje su operacije čitanja značajno češće od operacija pisanja, poput analitičke obrade u skladištu podataka.

System Architecture Image:

Arhitektura uključuje Flask web poslužitelj s SQLite in-memory bazom podataka. Dva komponenta, "facts_table" i "materialized_view", predstavljaju direktnu tablicu i prethodno izračunanu kocku podataka, redom. Aplikacija također uključuje Docker koji povezuje cijeli skup podataka što ga čini prijenosnim i jednostavnim za implementiranje u različitim okolinama.

Simulation of Load and Data:

Kako bi simulirali realne uvjete, generirali smo sintetske podatke koristeći Faker biblioteku i istaknuli prednosti Materialized viewa izvođenjem OLAP upita na Facts table i Materialized view. Sintetsko opterećenje dizajnirano je kako bi odražavalo scenarije u kojima se određeni ključevi češće upituju, prikazujući prednosti prethodno izračunatih agregata. Ovaj namjerni pomak u distribuciji podataka omogućio nam je isticanje dobiti u učinkovitosti u scenarijima gdje se određeni podskupovi podataka upituju češće od drugih.

Measurement of Response Times or Throughput:

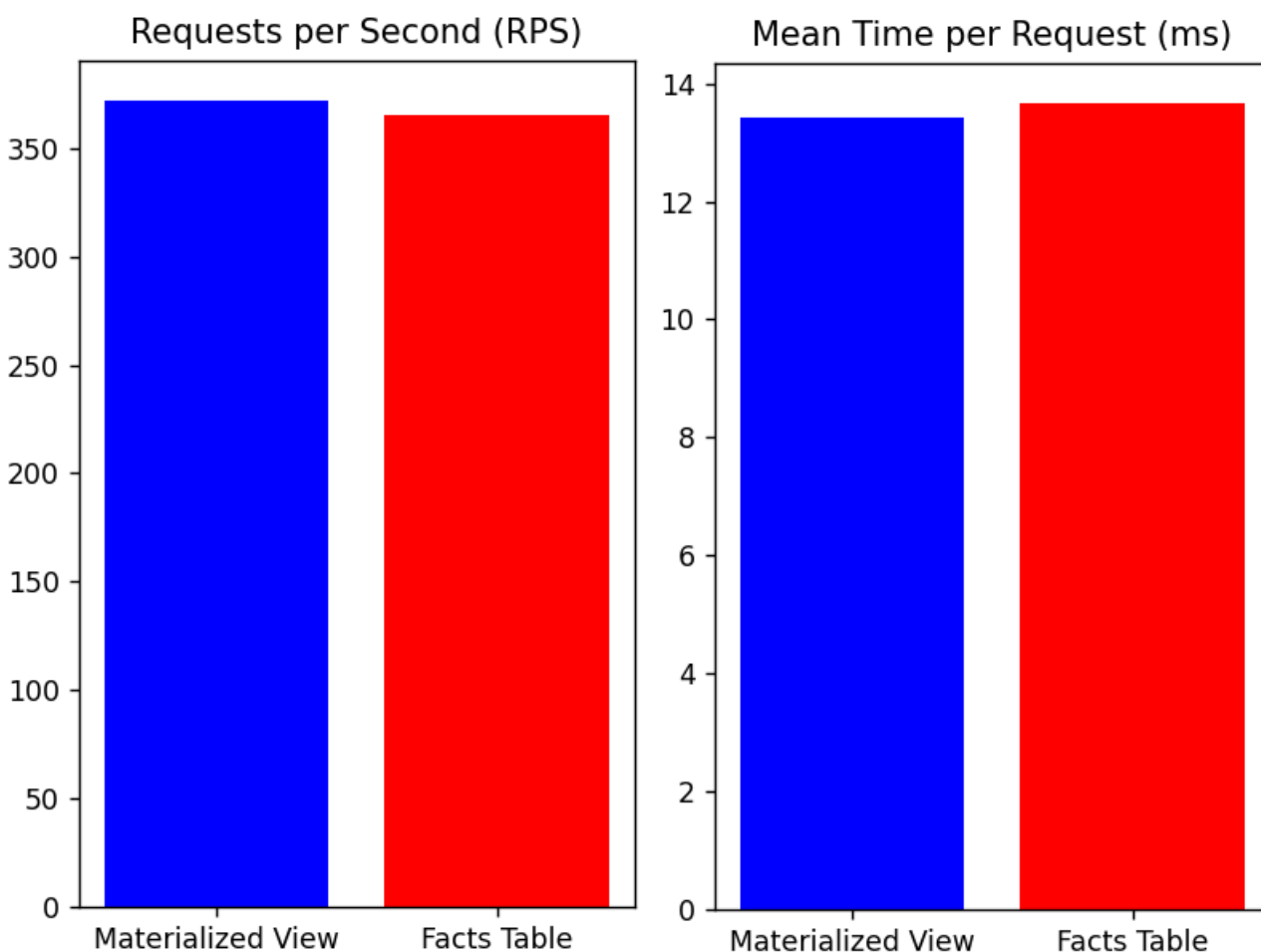
Mjerenje vremena odaziva napravili smo koristeći se Apache HTTP Server benchmarking alata. Apache HTTP Server Benchmark je alat za ispitivanje i procjenu performansi web poslužitelja. Jedna od prednosti korištenja Apache Benchmarka je što omogućuje slanje velikog broja HTTP zahtjeva prema određenom URL-u, što omogućuje mjerenje performansi web poslužitelja pod određenim opterećenjem te nakon izvršavanja testa pruža detaljan izlaz koji uključuje ključne metrike poput broja zahtjeva po sekundi (Requests per Second - RPS), vremena po zahtjevu, statistike uspješnosti i slično.

Kako bismo procijenili performanse Materialized viewa i Facts tablea, koristili smo alat Apache Benchmark (ab) konfiguriran za izvršavanje 10.000 zahtjeva s konkretnim postavkama konkurentnosti na 5. Ovaj sklop simulira umjerenu razinu istovremenih korisničkih zahtjeva prema Flask poslužitelju.

Za Materialized view mjerili smo vrijeme odgovora za OLAP upite, bilježeći vrijeme potrebno za dohvat i agregaciju prethodno izračunatih rezultata. Upiti su izvršeni pomoću nasumično generiranog ključa. Te za Facts table na sličan način mjerili smo vrijeme odgovora za OLAP upite.

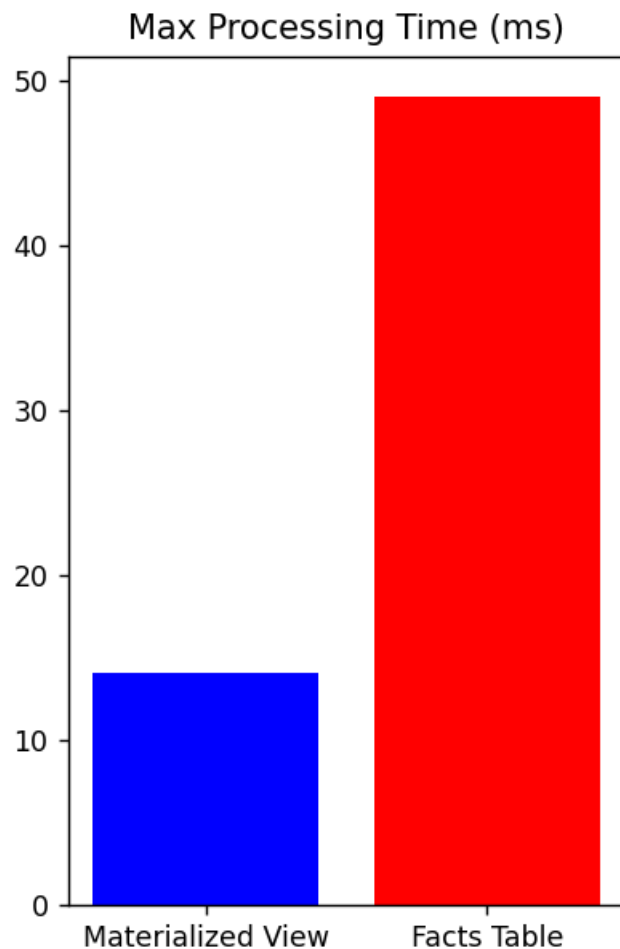
Za svako izvođenje benchmarka, izračunali smo prosječna vremena odgovora za Materialized view i Facts table. Prohodnost, izražena u zahtjevima po sekundi (RPS), izračunata je kao ukupan broj zahtjeva podijeljen s ukupnim vremenom odgovora za Materializirani Prikaz i Tablicu Činjenica.

e. Performance Comparison Plots and Results:



Na prvoj tablici možemo primijetiti kako Materialized view prikazuje malo višu propusnost od Facts tablea.

Na drugoj tablici možemo primijetiti kako Materialized view i facts table imaju slična srednja vremena odgovora.



Na trećoj tablici primijećujemo kako Materialized view ima nešto niže vrijeme procesiranja što upućuje na konzistentnije performance.

Na temelju rezultata Apache Benchmarka, Materialized view pokazuje nešto veću propusnost i konzistentnije performanse naspram Facts tablea, posebno u pogledu maksimalnog vremena procesiranja. To sugerira da, pod određenim uvjetima, pristup korištenja Materialized viewa može pružiti bolje ukupne performanse upita.