

Faculdade de Tecnologia – Unicamp

Curso: Engenharia e administração de banco de dados

Aluno: Gustavo Ferreira Lima

Matéria: Programação e Otimização em SQL

Prof. Manoel A. da Silva Jr.

Resumo e comentários do trabalho de graduação “*TUNING - Técnicas de Otimização de Banco de Dados Um Estudo Comparativo: Mysql e Postgresql*”

Extra: Estudo do banco Oracle: Performance e boas práticas!

Resumo

Conforme foi analisado o trabalho e discutido em aula, o trabalho dos alunos da Universidade Federal do Rio Grande (FURG), nos apresenta o conceito de TUNING, técnicas de otimização de banco de dados.

Pontos como a **reescrita de consulta**, **sintonia de parâmetros** no PostgreSQL, **criação de índices** são boas práticas que devem ser seguidas.

Além disso, a realização de testes com Benchmarks como TPC-C, TPC-H e TPCW é sugerida para avaliar e comparar o desempenho dos SGBDs. O processo de otimização contínuo, baseado em medições e ajustes, pode resultar em melhorias significativas no desempenho do sistema, garantindo uma operação mais eficiente e estável.

A reescrita de consultas é uma prática fundamental para evitar a criação de planos de execução ineficientes em sistemas de gerenciamento de banco de dados (SGBDs). Ao reescrever consultas, os administradores podem otimizar a forma como as consultas são processadas, garantindo que o plano de execução escolhido seja o mais eficiente possível. Isso é importante porque planos de execução inadequados podem resultar em sobrecarga no sistema, prejudicando o desempenho e a eficiência das consultas. Portanto, a reescrita de consultas é uma estratégia proativa para melhorar o desempenho e a resposta do sistema, evitando problemas de lentidão e uso excessivo de recursos.

A sintonia de parâmetros no PostgreSQL é um processo crucial que tem um impacto direto no desempenho do sistema de gerenciamento de banco de dados (SGBD). Através da configuração adequada de parâmetros como `shared_buffers`, `work_mem` e `effective_cache_size`, os administradores podem otimizar o PostgreSQL para atender às necessidades específicas de uma aplicação. Esses ajustes permitem melhorar a eficiência do sistema, garantindo um desempenho otimizado e uma resposta mais rápida às consultas. Portanto, a sintonia de parâmetros no PostgreSQL é

essencial para maximizar o desempenho e a eficiência do SGBD, adaptando-o às demandas específicas de cada ambiente de aplicação.

O **processo de otimização contínuo** é uma prática que pode resultar em melhorias significativas no desempenho de um SGBD. Ao realizar ajustes e otimizações de forma regular, os administradores podem identificar e corrigir possíveis gargalos de desempenho, garantindo uma operação mais eficiente e estável. Essa abordagem proativa permite que o sistema se adapte às mudanças nas demandas da aplicação, resultando em melhorias contínuas ao longo do tempo. Portanto, o processo de otimização contínuo é essencial para manter o SGBD funcionando de forma eficaz e atendendo às necessidades em constante evolução.

Também teve o caso comentado no trabalho, sobre aumento da memória de 64mb para 128mb. foram ajustados os parâmetros para esta otimização: max_connection e shared_buffers.

- Max_connection: este parâmetro foi alterado de 100 para 120, com o objetivo de elevar a capacidade de conexões concorrentes no SGBD PostgreSQL

Shared_buffers: este parâmetro foi aumentado de 64MB para 128MB. Aumentaram a memória compartilhada porque foi visto que as conexões são permitidas pelo SGBD PostgreSQL.

Aumentar permite ao SGBD armazenar mais dados em memória, acelerando o acesso e processamento de consultas. Por outro lado, aumentar permite mais conexões simultâneas, melhorando a capacidade de resposta do sistema em ambientes com alta demanda. Portanto, esses ajustes são essenciais para garantir um desempenho otimizado e uma operação eficiente do SGBD.

Já a análise de consultas e a criação de índices são práticas recomendadas para otimizar o desempenho de consultas em um sistema de banco de dados. A análise de consultas permite identificar consultas com desempenho inferior e otimizá-las, enquanto a criação de índices melhora a eficiência das consultas ao acelerar o acesso aos dados.

A criação de índices pode precisar de revisão devido a consultas que podem demorar devido à índices não utilizados ou índices que causam sobrecarga devido a alterações frequentes nos atributos. A análise de sintonia pode diagnosticar problemas, resultando na exclusão e criação de novos índices com base nessa análise.

Essas práticas são essenciais para garantir um desempenho eficiente e uma resposta rápida às consultas no ambiente do banco de dados.

Por fim, realizando ajustes e otimizações de forma regular, é possível identificar e corrigir possíveis gargalos de desempenho, adaptando o sistema às mudanças nas demandas da aplicação. Dessa forma, o processo contínuo de otimização pode resultar em melhorias significativas na eficiência e estabilidade do SGBD.

Comentários

Todo administrador, analista de sistemas e DBAs, tem que ter em mente as boas práticas conforme mencionadas. Seguindo desta forma, nos profissionais da área temos muito trabalho a ser feito como criação de novas consultas, índices e manutenção preventivas para otimização, para cada vez que atualizamos o sistema, ou implantamos algo novo.

Esses conceitos de Tuning, são importantes para toda organização ter como rotina, pois é essencial para manter o SGBD funcionando de forma eficaz para os analistas, que muitas vezes, precisam de dados para apresentar para Gerencia e gestão da empresa.

Ter conhecimento técnico avançado sobre o SGBD, também é importante para saber configurar os melhores parâmetros e evitar eventuais gargalos nas consultas.

No meu trabalho já tive a experiência de encontrar várias consultas com inúmeros Joins, causando lentidão. Muitos dos Joins, estavam conectando com tabelas que não estavam sendo usadas. Após reescrever as consultas mantendo apenas as tabelas necessárias, foi avaliado uma performance significativa.

Exemplo de consulta com filtros:

```
SELECT DISTINCT A.SEQPRODUTO,
                B.DESCCOMPLETA,
                A.CODDEPOSITO DEPOSITO,
                A.CODRUA RUA,
                A.NROPREDIO PREDIO,
                A.NROAPARTAMENTO APARTAMENTO,
                A.NROSALA SALA,
                A.QTDATUAL ESTOQUE_ATUAL,
                C.EMBALAGEM || ' ' || C.QTDEMBALAGEM AS EMBALAGEM
FROM CONSINCO.MLO_ENDereco A
LEFT JOIN CONSINCO.MLO_PRODUTO B
  ON A.SEQPRODUTO = B.SEQPRODUTO
LEFT JOIN CONSINCO.MAP_FAMEMBALAGEM C
  ON B.SEQFAMILIA = C.SEQFAMILIA
AND A.QTDEMBALAGEM = C.QTDEMBALAGEM
WHERE A.SEQPRODUTO IN ('829', '408') -- Produto
      AND A.NROEMPRESA = 97 -- Empresa CD Sumare
      AND A.CODDEPOSITO = 03 -- Deposito Avarias
ORDER BY A.SEQPRODUTO ASC;
```

	SEQPRODUTO	DESCCOMPLETA	DEPOSITO	RUA	PREDIO	APARTAMENTO	SALA	ESTOQUE_ATUAL	EMBALAGEM
1	408	AGUA MIN MINALBA 1.5L S GAS-	03	011	8	0	1	6.000	FD 6
2	829	REFRI COCA COLA 350ML-Ref55404	03	012	12	6	1	2376.000	FD 12
3	829	REFRI COCA COLA 350ML-Ref55404	03	012	12	4	1	2364.000	FD 12
4	829	REFRI COCA COLA 350ML-Ref55404	03	012	13	4	1	2376.000	FD 12
5	829	REFRI COCA COLA 350ML-Ref55404	03	012	11	0	1	24.000	FD 12

19:28 5 rows selected in 0.046 seconds

Consulta extraída do banco Oracle com o SGBD PL Developer v8.0, apresentando dados dos produtos com endereço no CD e saldo disponível

Por fim, um ponto importante é mencionar os **filtros**. Esse exemplo acima, mostra como é importante a utilização de filtros no WHERE. Se eu puxar todo o estoque do CD, irá apresentar uma grande quantidade de informação, demorando um bom tempo para gerar. Utilizando esses filtros como por exemplo filtro de produto, empresa e deposito, consigo filtrar a informação que preciso em segundos.

Extra

Estudo do banco Oracle: Performance e boas práticas!

No meu atual trabalho, tive uma experiência bem impactante no ano passado de 2023. Após implantamos um novo processo na logística no sistema WMS, para ao faturar as notas, ser impresso uma nota por palete, para que cada caminhão ter todos os paletes com notas. Desta forma conseguimos manter um melhor controle ao enviar 100% das cargas com notas, evitando problemas legais. E melhorando o processo de separação das mercadorias.

Na semana seguinte após implantamos este novo processo, a logística me reportou que várias queries view (Consultas), pararam de funcionar. Outro ponto, foi vários dashboards do Power BI não estarem mais funcionando. Em análise, descobri que **todas as tabelas** que este novo processo utiliza para faturar as notas, eram novas. Até as tabelas de registrar os produtos separados, eram novas.

Fiquei a semana recriando as consultas que já estavam prontas, utilizando as novas tabelas. **Lição aprendida:** todo projeto e demanda, é necessário avaliar o impacto que teremos no banco, e utilizar da **melhoria contínua**, para manter os processos em funcionamento para operação Logística e para empresa.

O banco de dados que trabalho diariamente no trabalho é Oracle, nele já utilizávamos algumas técnicas como a reescrita de consultas, para melhorar a performance.

Desde ano passado, venho utilizando esta técnica de criação de índices, e a cada dia temos um ambiente onde temos as informações da forma mais rápida possível.

Tenho contato direto com as áreas de negócio, e utilizo deste benefício da influência com as pessoas, para entender as dificuldades do dia a dia. Muitos casos, após várias conversas e análises, localizo meu objetivo: a tabela com lentidão extrema. Neste exemplo, após localizar a tabela/view com esta lentidão, realizo as boas práticas para melhorar a performance, causando um grande impacto positivo nas vidas das pessoas, e também impacto financeiro, pois quando há uma

análise rápida das informações extraídas do banco, é possível localizar e enviar os produtos para as lojas, evitando perdas e gerando lucro.