

TÉCNICAS SIMPLES DE OTIMIZAÇÃO EM UM BANCO DE DADOS RELACIONAL

Introdução: O presente estudo analisa algumas técnicas simples mas essenciais para otimização de bancos de dados relacionais, sempre respeitando a regra de negócio envolvida em sua implementação e tendo como referencial a extensa documentação do SQL Server (linguagem T-SQL) e MySQL (Microsoft, 2019). Então neste trabalho, é apresentado alguns conceitos chaves para uma otimização concisa, de forma a melhorar e agilizar o processamento das consultas. **Objetivos:** O objetivo é analisar alguns conceitos basilares de bancos de dados relacionais, como 1) A correta utilização dos operadores lógicos mais utilizados (AND e OR), 2) A implementação ou conversão de argumentos de pesquisa (*Search Arguments* – SARG), 3) Procedimentos armazenados (Stored Procedures) devidamente nomeados e 4) O uso eficiente de índices clusterizados e não clusterizados. Todos estes conceitos são utilizados nos vários distribuidores que existem atualmente, como SQL Server, MySQL, Oracle e etc. **Métodos:** A presente pesquisa seguiu o método descritivo, utilizando ferramentas presentes no ambiente de desenvolvimento integrado (*Integrated Development Environment* – IDE) da Microsoft, o SQL Server Management Studio (SSMS), como as estatísticas de acesso e tempo (Set Statistics IO) e estatísticas do cliente (*Client Statistics*). Todavia, qualquer sistema gerenciador de banco de dados (SGBD) pode lhe mostrar de forma gráfica os detalhes dos conceitos aqui apresentados. **Resultados:** Com a devida implementação dos conceitos de otimização (Bertolini, Cunha, Fortes, 2017), na conjunção os dois operandos precisam ser verdadeiros para o resultado ser verdadeiro, enquanto na disjunção basta que uma afirmação seja verdadeira para que a afirmação seja verdadeira. Logo, as análises mostram que o banco de dados é mais eficiente utilizando o elemento de maior quantidade como primeira condição no operador lógico OR – respeitando as características lógicas do operador de disjunção – e o elemento de menor quantidade como primeira condição no operador lógico AND – com base nas características lógicas do operador de conjunção. O otimizador de consultas valida uma expressão SARG restringindo uma busca a uma correspondência exata em um intervalo de valores ou um conjunto de duas ou mais expressões unidas pelo operador “AND”, interpretando seu conteúdo com base nessas

informações e escolhe o melhor índice para fazer essa busca. Como descrito na documentação do SQL Server na seção de criação de procedimentos (Microsoft, 2017), nomear os procedimentos armazenados diferentemente dos procedimentos armazenados do sistema – com prefixo SP_ - evita do SGBD pesquisar o procedimento armazenado no banco de dados do sistema criado pelo próprio SGBD para guardar os metadados, evitando assim uma busca desnecessária ocasionando em ganho de performance - no caso do SQL Server utilizando o *SQL Server Management Studio* – SSMS (Microsoft, 2019). Utilizar índices clusterizados para chaves primárias, com numeração sequencial e crescente e índices não clusterizados para pesquisas frequentes nas tabelas dos bancos de dados aumentam drasticamente o desempenho das consultas pelo otimizador de consultas – como mostra na documentação da Microsoft sobre índices clusterizados e não clusterizados descritos - lendo poucas páginas de dados para achar o resultado das condições propostas. **Conclusão:** É importante salientar que apesar de se tratar de conceitos básicos, são de extrema importância em bancos de dados grandes, pois todas essas técnicas otimizam bancos de dados relacionais de forma a melhorar os processos como um todo. De toda forma, muitas dessas técnicas podem inclusive ser utilizadas em outros distribuidores – sendo necessário consultar a documentação dos mesmos. Outro ponto positivo é que aprender e implementar tais conceitos lhe tornam um profissional diferenciado, mais completo e requisitado pelo nosso mercado tão competitivo.

- Referências:**
1. Bertolini, Cristiano; Bernardino, Guilherme; Rodrigues, Patricia; Licenciatura em Computação – Lógica Matemática. 1ª edição UAB/NTE/UFSM. 2017 [acessado em 12 Junho 2019]. Disponível em https://nte.ufsm.br/images/identidade_visual/LogicaMatematica.pdf
 2. Microsoft - SQL Server Management Studio (SSMS), 10/2018. [acessado em 12 Junho 2019]. Disponível em <https://docs.microsoft.com/pt-br/sql/ssms/sql-server-management-studio-ssms?view=sql-server-2017>
 3. Microsoft - Documentação do SQL Server, 02/2019. [acessado em 12 Junho 2019]. Disponível em <https://docs.microsoft.com/pt-br/sql/relational-databases/indexes/clustered-and-nonclustered-indexes-described?view=sql-server-2017>
 4. Microsoft - Documentação do SQL Server – Argumentos. [acessado em 12 Junho 2019]. Disponível em <https://docs.microsoft.com/pt-br/sql/t-sql/statements/create-procedure-transact-sql?view=sql-server-2017>
 5. Microsoft – Documentação do SQL Server - Guia de arquitetura e design de índices do SQL Server. [acessado em 12 Junho 2019]. Disponível em <https://docs.microsoft.com/pt-br/sql/relational-databases/sql-server-index-design-guide?view=sql-server-2017>

6. MySQL – Documentação - *How MySQL Uses Indexes*. [citado em 12 Junho 2019]. Disponível em <https://dev.mysql.com/doc/refman/8.0/en/mysql-indexes.html>
7. MySQL – Documentação - *Logical Operators*. [acessado em 12 Junho 2019]. Disponível em <https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html>

Palavras-Chave: Otimização de performance, Técnicas performáticas em bancos de dados relacionais, Performance com SQL Server.