

Alunos:

João Vítor Morandi Lemos - 160010195

Filipe Toyoshima Silva - 160049971

Professor: Vandor Rissoli

Aula 9 Extra 1 – Índices

O que é um Índice em banco de dados relacionais

Basicamente, índice é um otimizador de banco de dados. Ele funciona, por exemplo, como um índice de livro: é colocado separado do conteúdo (indexado antes) e é utilizado para procurar algum dado sem precisar percorrer todas as páginas do documento para encontrar o conteúdo desejado. No caso de um sistema de banco de dados, não é mais necessário percorrer todas as tuplas. Alguns tipos de índices:

- Índice ordenado: os dados são colocados em ordem crescente, para assim poder ser feita a busca binária dos termos.

- Índices de hash: os identificadores da tabela de índices são ligados aos identificadores da tabela de dados.

A utilização de índices se assemelha muito a utilização da técnica de banco de dados orientado a colunas. Ambos melhoram o desempenho em análise e pioram o desempenho em updates, além de focar nos atributos. A diferença reside no fato de o banco de colunas separar o banco em diversos fragmentos, pequenos segmentos do banco, cada um com uma coluna, guardados em lugares diferentes. Já os índices consistem na criação de uma outra tabela separada contendo identificadores para cada tupla fáceis de serem pesquisados.

Para que servem índices em banco de dados relacionais

Para acelerar o processo de consulta e recuperação de dados, transformando uma busca exaustiva e demorada na tabela de dados em uma busca simples na tabela de índices. O número de comparações feitas é reduzido significativamente, já que não precisa ser feita uma a cada tupla até encontrar o que se precisa. Ao invés disso, a pesquisa é feita na tabela de índices, que é criada justamente com esse objetivo, onde pode-se usar de técnicas como a busca binária para otimização.

Mas não seria mais fácil apenas ordenar a tabela-base e fazer a busca binária pelo atributo na própria tabela? O desempenho é o mesmo e se o objetivo é fazer uma busca em apenas um atributo da tabela, é até aconselhável fazer dessa forma. Porém, se é necessário realizar uma consulta em mais de um atributo da tabela, não é possível ordenar a tabela-base de

duas formas diferentes ao mesmo tempo, sendo necessário a criação de tabelas de índices, uma para cada atributo que se deseja consultar. Além disso, a reordenação de um grande volume de dados em disco é extremamente cara em termos de processamento.

Por outro lado, o uso de índices deve ser descartado se não houver uma utilidade específica ou se o banco de dados é focado em modificações, já que em toda e qualquer atualização realizada sobre a base de dados a tabela de índices também será atualizada, o que causa uma queda no desempenho.

Quais estruturas de dados que os índices mais utilizam para tornar o banco eficiente

Hash

Um hash é uma estrutura de dados que transfere ou converte um dado de uma tabela grande e cheia de atributos para uma tabela de índices. Essa tabela é então ordenada e pode ser utilizada para realizar buscas mais rapidamente e depois encontrar a informação na tabela-base, sem a necessidade de fazer a busca nela.

No exemplo da figura, a tabela de índices faz uma ligação com o atributo indexado da tabela-base. Com essa ligação, é possível fazer uma busca pela tabela de índices, e depois de encontrado, é retornado o valor da tabela-base representado por aquele índice.

Árvore B+

São variações de árvores B, que por sua vez são generalizações de árvores binárias balanceadas. A diferença é que árvores binárias podem armazenar apenas dois filhos por nó, e contêm apenas uma chave por nó, enquanto em uma árvore B não há essas limitações, o que aumenta consideravelmente a capacidade de armazenamento.

Por sua vez, árvores B+ funcionam de maneira mais eficiente em relação ao desempenho da busca: os nós internos da árvore são na verdade índices que apontam para os nós externos, onde estão localizados os dados.

No exemplo da figura, os dados estão contidos de d1 a d7. Os números contidos em cada um dos nós são índices que vão percorrendo o caminho até o dado desejado.