

Banco de dados II

1 - Introdução a Armazenamento e Indexação

Marcos Roberto Ribeiro



Instituto Federal Minas Gerais - Campus Bambuí

2018

Introdução

- Basicamente, um banco de dados é um conjunto de registros
- O SGBD precisa armazenar estes registros em arquivos
- Precisamos entender como isto é feito para usar o SGBD de forma eficiente
- Dependendo de como os registros são organizados nos arquivos, podemos ter algumas operações eficiente e outras ineficientes

Exemplo

- Suponha que que desejamos recuperar os funcionários ordenados pela data de nascimento
- Uma alternativa é gravar os registros em arquivo ordenados pela data de nascimento:

Arquivo Ordenado

id	...	nascimento
⋮	⋮	↓

Prós

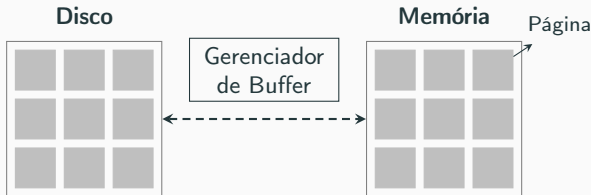
- Bom para recuperar registros pela data de nascimento

Contras

- Difícil de manter (atualizar) o arquivo
- Se houver outro tipo de consulta como: “Obter os funcionários com salários maiores do que 5.000,00”, é preciso ler todo o arquivo

- O uso da indexação auxilia a recuperar os dados de múltiplas formas

Armazenamento Externo de Dados



- Cada registro possui um **rid** para identificar sua página

Disco

- Maior custo para ler ou gravar
- Maior capacidade
- Leitura sequencial de páginas mais eficiente do que a leitura aleatória

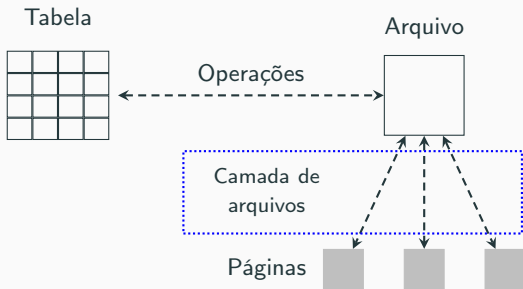
Memória

- Menor custo de acesso
- menor capacidade

Fitas

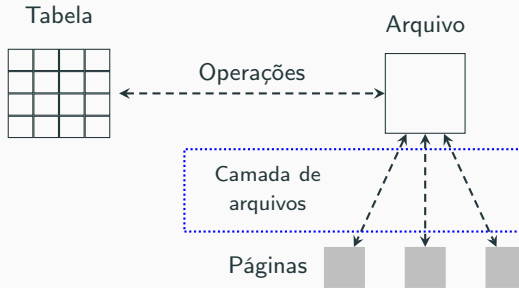
- Ideias para backup

Organizações de Arquivos e Indexações I



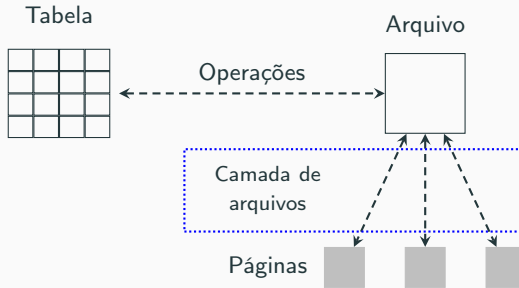
- Basicamente, uma tabela é armazenada em um arquivo pelo SGBD
- Portanto, as operações sobre a tabela precisam ser replicadas para o arquivo
- O SGBD quebra o arquivo em páginas para facilitar o gerenciamento de memória
- Por outro lado, o SGBD precisa de uma camada de arquivos que implementa as operações sobre as páginas de forma transparente

Organizações de Arquivos e Indexações II



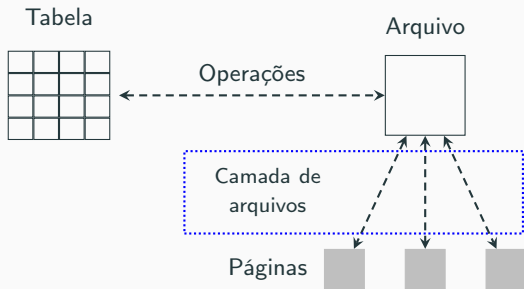
- Quais operações podem ser feitas sobre tabelas ou arquivos?
 - Inserção
 - Deleção
 - Modificação
 - Varredura

Organizações de Arquivos e Indexações II



- Quais operações podem ser feitas sobre tabelas ou arquivos?
 - Inserção
 - Deleção
 - Modificação
 - Varredura

Organizações de Arquivos e Indexações III



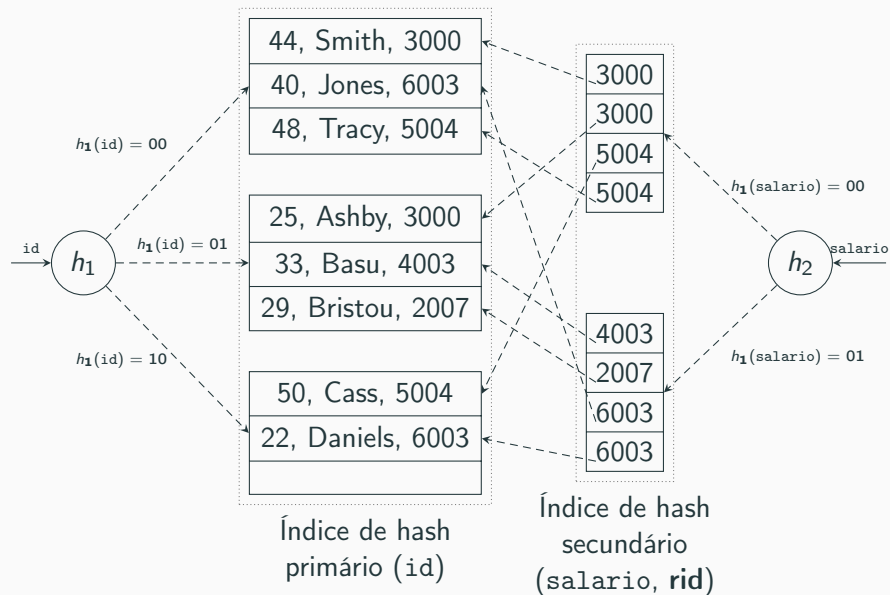
- Os índices são estruturas de dados que otimizam a fase de busca de registros de acordo com uma **chave de pesquisa**
- As chaves de pesquisa podem ser campos ou conjuntos de campos (ex.: data de nascimento, salário)
- As entradas de dados dos índices podem ser de três tipos:
 - O próprio registro da tabela: para **índices agrupados**¹
 - (chave de pesquisa k , **rid**): para índices de valores únicos
 - (chave de pesquisa k , lista de **rid**): para índices com repetição de valores

¹Índice integrado com arquivo

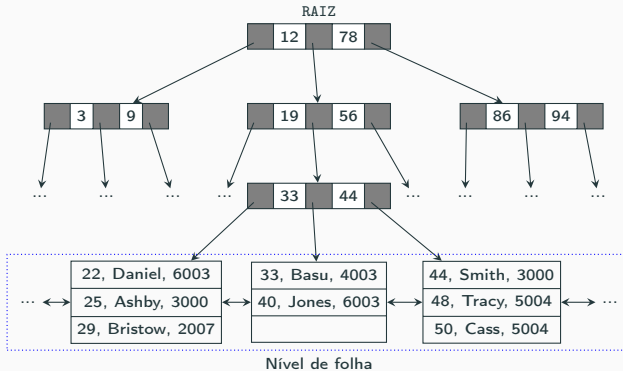
Chave primária: Índice Primário (não possui duplicatas)

Demais chaves: Índices secundário (pode conter duplicatas)

Indexação Baseada em Hash



Indexação Baseada em Árvore



Comparação das Organizações de Arquivos

- Esta comparação considera um conjunto de registros de funcionários com chave de pesquisa (nascimento, salario)
- Todas as operações são especificadas sobre estes campos (nascimento, salario)
- As organizações de arquivos consideradas são:
 - Arquivo heap (ordem aleatória)
 - Arquivo ordenado por (nascimento, salario)
 - Arquivo com índice agrupado de árvore B+ sobre (nascimento, salario)
 - Arquivo com índice não agrupado de árvore B+ sobre (nascimento, salario)
 - Arquivo com índice não agrupado de hash sobre (nascimento, salario)

Comparação das Organizações de Arquivos

- As operações consideradas serão:
 - A) Varredura
 - B) Pesquisa (igualdade)
 - C) Pesquisa (intervalo)
 - D) Inserção de um registro
 - E) Exclusão de um registro

Modelo de Custo

B: Número de páginas

R: Número de registros por página

F: *Fan-out* (número de filhos de um nó em árvores B+)

D: Tempo de E/S de uma página (operação mais cara)

C: Tempo para processar um registro

H: Tempo para computar uma função hash

Custo de Arquivo Heap

A) Varredura: $B(D + RC) = BD$

B) Pesquisa (igualdade): $B(D + RC) = BD$ (o registro pode não existir)

C) Pesquisa (intervalo): $B(D + RC) = BD$

D) Inserção: $2D + C$ (insere sempre no fim do arquivo)

E) Exclusão: $\underbrace{B(D + RC)}_{\text{pesquisa do registro}} + \underbrace{C + D}_{\text{grava na página do registro}} = BD + D$ (considerando um registro)

Custo de Arquivo Ordenado

A) Varredura: $B(D + RC) = BD$

B) Pesquisa (igualdade): $\underbrace{D \log B}_{\substack{\text{busca} \\ \text{pela} \\ \text{página}}} + \underbrace{C \log R}_{\substack{\text{busca} \\ \text{pelo} \\ \text{registro}}} = D \log B$

C) Pesquisa (intervalo): $D \log B + C \log R + \underbrace{\quad ? \quad}_{\substack{\text{páginas} \\ \text{adicionais}}} = D \log B + ?$

D) Inserção: $\underbrace{D \log B + C \log R}_{\text{pesquisa}} + \underbrace{B(D + CR)}_{\substack{\text{realocação dos} \\ \text{registros} \\ \text{seguintes}}} = BD + D$

E) Exclusão: mesmo caso da inserção

Custo de Arquivo com Índice Árvore B+ Agrupado

- Normalmente as árvores B+ possuem páginas com 67% de ocupação, com isto o número de páginas é $1.5B$

A) Varredura: $1.5B(D + RC) = 1.5BD$

B) Pesquisa (igualdade): $\underbrace{D \log_F 1.5B}_{\text{busca pela folha}} + \underbrace{C \log R}_{\text{busca binária na folha}} = D \log_F 1.5B$

C) Pesquisa (intervalo):
 $D \log_F 1.5B + C \log R + \underbrace{\quad ? \quad}_{\text{páginas adicionais}} = D \log_F 1.5B + ?$

D) Inserção: $D \log_F 1.5B + C \log R + D = D \log_F 1.5B + D$ (em geral, o registro cabe na folha)

E) Exclusão: mesmo caso da inserção

Custo de Arquivo com Índice Árvore B+ Não Agrupado I

- Consideramos páginas com 67% de ocupação e entradas de índices com 10% do tamanho dos registros, temos:
 - $0.1 \times 1.5B = 0.15B$ páginas
 - $10 \times 0.67R = 6.7R$ registros / página

A) Varredura: $\underbrace{0.15B(D + 6.7R)}_{\text{leitura de todas as entradas do índice}} + \underbrace{R(D + C)}_{\text{processamento dos registros}} = 0.15BRD^2$

- Melhor não usar o índice $B(D + RC)$

B) Pesquisa (igualdade):

$$\underbrace{D \log_F 0.15B}_{\text{busca pela folha}} + \underbrace{C \log 6.7R}_{\text{busca na folha}} + \underbrace{D}_{\text{leitura do arquivo}} = D \log_F 0.15B + D$$

Custo de Arquivo com Índice Árvore B+ Não Agrupado II

C) Pesquisa (intervalo):

$$D \log_F 0.15B + C \log 6.7R + D \times \underbrace{?}_{\substack{\text{número de} \\ \text{registros a} \\ \text{serem} \\ \text{retornados}}} = D \log_F 0.15B + D \times ?$$

D) Inserção:

$$\underbrace{2D + C}_{\substack{\text{gravação} \\ \text{no} \\ \text{arquivo}}} + \underbrace{D \log_F 0.15B + C \log 6.7R}_{\substack{\text{busca pela folha no índice}}} + \underbrace{D}_{\substack{\text{grava} \\ \text{página} \\ \text{no} \\ \text{índice}}} = 2D + D \log_F 0.15B + D$$

E) Exclusão: mesmo caso da inserção

Custo de Arquivo com Índice Hash Não Agrupado I

- Consideramos páginas com 67% de ocupação e hash sem overflow
- Consideramos também páginas com 80% de ocupação e uma página adicional / bucket
- Assim temos
 - $0.125 \times 0.1B = 0.125B$ páginas
 - $10 \times 0.8R = 8R$ registros / página

A) Varredura: $B(D + RC)$ (mesmo caso do índice não agrupado de árvore B+)

B) Pesquisa (igualdade):

$$\underbrace{H}_{\text{identificação da página}} + \underbrace{D}_{\text{leitura da página}} + \underbrace{8RC}_{\text{varredura da página}} + \underbrace{D}_{\text{leitura do arquivo}} = 2D$$

Custo de Arquivo com Índice Hash Não Agrupado II

C) Pesquisa (intervalo): $B(D + RC)$ (o índice hash não suporta pesquisa por intervalo)

D) Inserção: $\underbrace{2D + C}_{\text{gravação no arquivo}} + \underbrace{H}_{\text{identificação da página no índice}} + \underbrace{2D + C}_{\text{leitura e alteração da página}} = 4D$

E) Exclusão: $\underbrace{H + 2D + 8RC}_{\text{pesquisa pelo registro}} + \underbrace{2D}_{\text{leitura e alteração da página}} = 4D$

Comparação

Tipo de Arquivo	Varredura	Pesquisa (Igualdade)	Pesquisa (Intervalo)	Inserção	Exclusão
Heap	BD	BD	BD	$2D$	Pesquisa + D
Ordenado	BD	$D \log B$	$D \log B + \#$ páginas	Pesquisa + BD	Pesquisa + BD
Agrupado	$1.5BD$	$D \log 1.5B$	$D \log 1.5B + \#$ páginas	Pesquisa + D	Pesquisa + D
Árvore B+ não agrupado	BD	$D \log 0.15B$	$D \log 0.15B + D \times \#$ registros	Pesquisa + $3D$	Pesquisa + $2D$
Hash não agrupado	BD	$2D$	BD	$4D$	Pesquisa + $2D$

Vantagens e Desvantagens

- Arquivo heap
 - (+) Inserção rápida (sempre no final)
 - (-) Pesquisa e exclusão lentos
- Arquivo ordenado
 - (+) Pesquisa rápida
 - (-) Inserção e exclusão lentos
- Arquivo com índice agrupado de árvore B+
 - (+) Inserção e exclusão eficientes, pesquisa rápida
 - (-) Pequeno overhead
- Arquivo com índice não agrupado de árvore B+
 - (+) Pesquisa, inserção e exclusão eficientes
 - (-) Pesquisa por intervalo pode se tornar lenta
- Arquivo com índice não agrupado de hash
 - (+) Eficiente em pesquisa e atualização
 - (-) Não suporta pesquisa por intervalo

Índices e Sintonização de Desempenho

- Índices podem ser criados para tornar consultas mais eficientes
- Porém certas alterações nos dados podem causar atualizações de vários índices
- Estas atualizações podem causar problemas de desempenho no sistema
- O ideal é haver um equilíbrio, ou seja, os índices são ferramentas poderosas que devem ser usados com cautela

Chaves de Pesquisas Compostas

- As chaves de pesquisas compostas possui mais de um atributo, por exemplo, (id, salario)
- Tais chaves podem suportar faixas maiores de consultas
- Além disto, elas aumentam as chances de *avaliações somente de índice*
- Por outro lado, o índice precisa ser atualizado se ocorrer qualquer alteração em um dos campos que compõem a chave
- Outra desvantagem é o espaço mais ocupado pelo índice

Avaliações Somente de Índice

- Quando todos os campos buscados em uma consulta fazem parte da chave do índice, tal consulta pode ser avaliada usando apenas o índice
- Não é necessário buscar os dados no arquivo e isto faz com que a consulta seja avaliada mais rapidamente

Referências I



Date, C. J. (2004).

Introdução a sistemas de bancos de dados.

Elsevier, Rio de Janeiro.



Elmasri, R. and Navathe, S. B. (2011).

Sistemas de banco de dados.

Pearson Addison Wesley, São Paulo, 6 edition.



Ramakrishnan, R. and Gehrke, J. (2008).

Sistemas de gerenciamento de banco de dados.

McGrawHill, São Paulo, 3 edition.



Silberschatz, A., Korth, H. F., and Sudarshan, S. (2007).

Sistema de bancos de dados.