

PROCESSAMENTO DE CONSULTAS – Cap. 12

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de índices para o arquivo, tamanho de bloco, tamanho e organização do arquivo, número de tuplas, tipo e tamanho do registro de dados.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de **índices para o arquivo**, tamanho de bloco, tamanho e organização do arquivo, número de tuplas, tipo e tamanho do registro de dados.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de índices para o arquivo, **tamanho de bloco**, tamanho e organização do arquivo, número de tuplas, tipo e tamanho do registro de dados.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de índices para o arquivo, tamanho de bloco, **tamanho e organização do arquivo**, número de tuplas, tipo e tamanho do registro de dados.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de índices para o arquivo, tamanho de bloco, tamanho e organização do arquivo, **número de tuplas**, tipo e tamanho do registro de dados.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

O PROCESSAMENTO DE CONSULTAS

- atividades para se extrair dados de um BD da forma mais eficiente possível, considerando-se a existência de índices para o arquivo, tamanho de bloco, tamanho e organização do arquivo, número de tuplas, **tipo e tamanho do registro de dados**.
- Exemplo:
- Traduzir consultas de alto nível em expressões que podem ser implementadas no nível físico do BD. (Tradução de uma expressão SQL).
- Passos
 - Análise sintática e tradução;
 - Otimização;
 - Avaliação.

Passo 1 - Análise sintática e tradução

- validar a expressão SQL,
- conferir a sintaxe,
- verificar nome das relações,
- atributos, etc.
- e traduzir para expressões de algebra relacional fundamental.
-

Passo 1 - Análise sintática e tradução

Exemplo de situações:

```
SELECT SALDO FROM CONTA WHERE SALDO < 2500
```

Pode ser traduzido em:

- $\sigma_{\text{saldo} < 2500} (\Pi_{\text{saldo}}(\text{conta}))$:: aqui se faz uma projeção e filtra-se após o resultado da projeção;
- $\Pi_{\text{saldo}} (\sigma_{\text{saldo} < 2500}(\text{conta}))$:: aqui durante a execução da projeção se executa a operação de seleção.

Além de alternativas de algoritmos, como:

- leitura seqüencial em toda a tabela;
- Se houver índice, usar o índice para localizar tuplas.

Passo 1 - Análise sintática e tradução

$\sigma_{\text{saldo} < 2500}$ (II saldo (conta))

σ :: aqui se faz uma projeção e filtra-se após o resultado da projeção;

Saldo
2500
1000
1200
4000
5333
1100

Saldo
1000
1200
1100

Passo 1 - Análise sintática e tradução

$\Pi_{\text{saldo}} (\sigma_{\text{saldo} < 2500}(\text{conta}))$

$\Pi::$ aqui durante a execução da projeção se executa a operação de seleção.

Nome	Saldo	conta
Maria	1000	434343-33
José	1200	444224-44
Marta	1100	039540-44

Saldo
1000
1200
1100

Passo 1 - Análise sintática e tradução

OUTRO EXEMPLO:

```
SELECT nome, descCargo  
FROM cargo, empregado  
WHERE empregado.codCargo = cargo.CodCargo
```

Alternativas:

- Produto cartesiano;
- Junção natural

Alternativas de algoritmos:

- Executa leitura a partir de empregado e do empregado se relaciona com cargo;
- Executa leitura a partir de cargo e do cargo se relaciona com o empregado.

Passo 2 - Otimização

- Visa escolher o melhor caminho e de melhor desempenho na execução de uma expressão SQL.
- SQL - linguagem declarativa e o usuário não especifica como será a recuperação de dados.
- Em bancos de dados mais antigos (modelo em rede, hierárquicos e outras variações) ou em banco de dados onde não há o conceito que envolve a linguagem SQL - linguagens procedurais embutidas na linguagem de programação hospedeira.
 - Responsabilidade do usuário de banco de dados (programador, analista) escrever o algoritmo que garanta a melhor alternativa na recuperação de dados. Exemplo: Natural/Adabas.

Passo 3 - Avaliação

- Definir o algoritmo a ser usado para cada operação específica.
- Diferentes planos de avaliação, custos diferentes. BD determinar qual. Com base em estatísticas.

A otimização de consulta é o processo de selecionar o plano de avaliação de consulta mais eficiente.

12.2 Informações para estimativa de custo

BD mantém as seguintes informações estatísticas:

- Número de tuplas de uma relação (tabela);
- Número de blocos que contém tuplas de uma relação;
- Tamanho em bytes de uma tupla (registro de uma tabela);
- Fator de bloco – número de tuplas (registros) que cabem em um bloco
- Etc.
-
- Manter dados estatísticos causa overhead (atualização periódica para organização, manutenção, backup, etc., do banco)

Medidas de Custo

- Acessos a disco : Número de blocos a serem recuperados é o fator mais importante.
- Tempo de CPU para executar consulta
- Custo de comunicação para banco de dados distribuídos

12.4 Informações gerais

- Caminhos de leitura para uma tabela leva em consideração tamanho e organização do BLOCO;
- Nem sempre escolhe-se o acesso por índice.
 - Leitura sequencial do arquivo pode ser mais rápida (ex. SELECT com faixa de valores).
 - A quantidade de blocos em que o arquivo (da relação) e do índice estão ocupando.
 - Ex.: Se a tabela é pequena e ocupa um bloco, a leitura por índice implicará em acessar mais um bloco. Neste caso, mais rápido recuperar o bloco e fazer busca sequencial (o bloco estará em memória).

12.9 Cuidados (expressões SQL)

- Dicas para garantir melhor performance no comando SQL e depende da avaliação que o banco a ser usado faz em relação a expressão SQL. Exemplos dados, baseiam-se em como o otimizador de consultas do Oracle trata:

- **WHERE a AND b AND c**

- Ordem de execução das preposições é : c, b e a
- portanto as expressões mais custosas (que impliquem em menor performance) devem ser as primeiras.
- Exemplo:

```
SELECT codigo, nome FROM tabelaA WHERE  
      (tabelaA.situacao = SELECT situacao FROM tabSituacao) AND  
      codigo > 10
```

- **WHERE a OR b**

- Neste caso a ordem de execução é das preposições a , b.

12.9 Cuidados (expressões SQL)

- *Utilização de índices*
- Índices são usados se o interpretador de consultas na avaliação da expressão SQL detectar algum indício da necessidade do índice.
- Prioridades:
 - Avaliar primeiro as cláusulas WHERE;
 - Depois ORDER BY
 - E GROUP BY.
- Exemplo:
- Tabela TESTE, tem os atributos a, b, c, d. O atributo a é chave primária, b + c compõe um índice secundário, c outro índice, c + a outro índice.
- `SELECT * FROM TESTE WHERE a > 10 ORDER BY b`
 - O índice a ser usado é o de a.
 - `SELECT * FROM TESTE WHERE d = "x" ORDER BY C, A`
 - Neste caso utiliza-se o índice c+a
 - `SELECT * FROM TESTE WHERE d < "a" ORDER BY A, C`
 - Neste caso não é utilizado nenhum tipo de índice.

Determinação do melhor caminho

- `SELECT nomeA, nomeB FROM tabelaA, tabelaB
WHERE ...`
- Se o otimizador não encontrar o melhor caminho para executar o comando SQL (a partir da tabelaA ou da tabelaB) ele opta em ler a partir da última tabela especificada no comando SQL, ou seja, tabelaB. Portanto é interessante, quando se vai executar o comando SQL e se sabe qual o melhor caminho já colocar nesta ordem.
-

Repetição de comandos SQL

- Se você vai dar comandos SQL repetidos: `SELECT * FROM TABELA`, escreva sempre da mesma forma, sem diferenças, pois o banco pode comparar a última expressão executada e utilizar o “seu cache” para retornar a resposta.
- `SELECT * FROM TABELA` é diferente de `select * from tabela`.
- Nota: diferente aqui não significa no resultado do comando.

Exercícios

12.3. Considere a consulta SQL:

```
select T.nome_agencia  
from agencia T, agencia S  
where T.fundos > S.fundos and S.cidade_agencia = "Belo  
Horizonte"
```

- a) Escreva uma expressão de álgebra relacional eficiente que seja equivalente a essa consulta. Justifique sua escolha.
- b) Para o exemplo acima, faça a representação de tabelas caso as condições tivessem sua ordem trocada.

12.7. Quais são as vantagens e as desvantagens de índices hash em relação aos índices Árvore-B+? Como pode o tempo de índice disponível influenciar na escolha de uma estratégia de processamento de consulta?

- Considere o seguinte esquema relacional para o banco de dados de um grande banco com cobertura nacional.

AGENCIAS (NOME_AGENCIA, CIDADE_AGENCIA, FUNDOS);

CONTAS (NOME_AGENCIA, NUMERO_CONTA, SALDO) NOME_AGENCIA REFERENCIA AGENCIAS;

CLIENTES (NOME_CLIENTE, CIDADE_NASCIMENTO, NUMERO_CONTA) NUMERO_CONTA REFERENCIA CONTAS;

- Considere, ainda, que os atributos sublinhados correspondam às chaves primárias das respectivas relações e, após as definições das relações CONTAS e CLIENTES, sejam descritas as regras de integridade referenciais. Suponha que o banco de dados armazene
- informações de 500 agências, de 1.000.000 de contas e de 1.500.000 clientes, sendo que 200.000 contas são de agências da cidade de São Paulo e 100.000 clientes nasceram em Recife. Considere, finalmente, que esse sistema de banco de dados tenha um otimizador
- de consultas embasado em heurísticas e que se precise realizar a seguinte consulta.

```
SELECT *  
FROM AGENCIAS, CONTAS, CLIENTES  
WHERE CONTAS.NOME_AGENCIA = AGENCIAS.NOME_AGENCIA  
AND CLIENTES.NUMERO_CONTA = CONTAS.NUMERO_CONTA  
AND CIDADE_AGENCIA = 'SAO PAULO'  
AND CIDADE_NASCIMENTO = 'RECIFE'  
AND SALDO > 1000;
```

- A partir dessas informações e considerando \bowtie o operador de junção natural e σ o operador de seleção, assinale a opção que apresenta o melhor plano de avaliação de consultas para a consulta apresentada acima.

