



1. CONCEITOS E NOÇÕES SOBRE SISTEMAS DE BANCOS DE DADOS

Continuação da Apostila 1...

1.12. OTIMIZAÇÃO DA CONSULTA

Otimização da consulta é o processo de selecionar o plano de avaliação de consulta mais eficiente dentre as muitas estratégias normalmente possíveis para o processamento de determinada consulta, especialmente se esta for complexa. (SILBERSCHATZ; KORTH; SUDARSHAN, 2006, p. 381).

Geralmente a otimização da consulta entra em ação em casos em que o usuário escreveu uma consulta não tão eficiente, causando uma maior complexidade na avaliação da consulta.

Um aspecto da otimização ocorre no nível da álgebra relacional, em que o sistema tenta encontrar uma expressão que seja equivalente à expressão dada, porém, cuja execução seja mais eficiente. Outro aspecto é a seleção de uma estratégia detalhada para processamento da consulta, como a escolha do algoritmo a ser usado para a execução de uma operação, a escolha dos índices específicos a utilizar, e assim por diante.

A diferença no custo (tempo de avaliação) entre uma boa estratégia e uma estratégia ruim normalmente é substancial e pode chegar a várias ordens de grandeza. Logo, vale a pena para o sistema gastar uma quantidade de tempo substancial na seleção de uma boa estratégia para processar uma consulta, mesmo que a consulta seja executada apenas uma vez.

1.12.1. Visão Geral

Considere a expressão da álgebra relacional para a consulta “Encontrar os nomes de todos os clientes que tenham uma conta em qualquer agência localizada no Brooklyn”.

π NOMECLIENTE (σ CIDADEAGENCIA = “Brooklyn” (*agencia x (conta x depositante)*)))

Essa expressão constrói uma grande relação intermediária, *agencia x conta x depositante*. Porém, o interesse está em apenas algumas poucas tuplas dessa relação (aquelas pertencendo a agências localizadas no Brooklyn), e em apenas um dos seis atributos dessa relação. Como o interesse é em apenas as tuplas que pertencem a agências localizadas em Brooklyn, não é necessário considerar as tuplas que não têm *cidadeagencia* = “Brooklyn”. Reduzindo o número de tuplas da relação *agência*, ocorre a redução do tamanho do



resultado intermediário. A consulta agora pode ser representada pela seguinte expressão:

π NOMECLIENTE ((CIDADEAGENCIA = "Brooklyn" (*agencia*)) \times (*conta x depositante*))

1.13. FUNÇÕES RELACIONADAS À UTILIZAÇÃO DE UM SGBD

Para definir, construir e manter um pequeno BD só é necessária uma pessoa. Entretanto, um grande BD envolve vários especialistas nas tarefas de projeto, utilização e gerenciamento:

Usuários Finais: são as pessoas cujo trabalho requer o acesso ao BD para consultas, atualizações e relatórios. O BD, por princípio, existe em função das necessidades dos usuários.

Analistas de Sistemas e Programadores de Aplicação: Analista de Sistemas determinam os requisitos dos usuários finais, e desenvolvem as especificações das transações necessárias. Programadores desenvolvem essas especificações, testam e dão manutenção nos programas, que em geral, serão operados pelos próprios usuários. Analistas e programadores devem estar familiarizados com o SGBD que será utilizado a implementação da aplicação.

Administrador de Banco de Dados (DBA): em qualquer organização onde muitas pessoas utilizam os mesmos recursos é necessário um chefe que gerência esses recursos. Num banco de dados o recurso primário é o próprio BD e secundário é o SGBD. Administrar esses recursos é responsabilidade do DBA. Esse administrador é a pessoa que tem total controle sobre o BD. Ele deve interagir com cada grupo de desenvolvedores para entender seus requisitos e projetá-los **integradamente**. O projeto final do BD deve suportar os requisitos de todos os grupos de usuários. Por exemplo, o DBA deve interagir com os analistas encarregados dos sistemas de RH e Pessoal para identificar os dados necessários (individuais e comuns), para então projetar um novo BD ou ampliar o existente, definindo as estruturas adequadas. Sua função inclui a autorização de acessos ao banco de dados e a coordenação e monitoramento de seu uso. O DBA é responsável pela utilização eficiente do espaço de armazenamento, pela segurança dos dados e pelo desempenho (tempo de resposta) do sistema.

1.14. LINGUAGEM DE DEFINIÇÃO DE DADOS (DDL)

A Linguagem de Definição de Dados (**Data Definition Language - DDL**), específica, através de um conjunto de definições expressas, um esquema de banco de dados, onde o resultado da compilação de comandos de uma DDL é



um conjunto de tabelas que são armazenadas em um arquivo chamado dicionário (ou diretório) de dados.

Podemos considerar diretório de dados como um arquivo que contém metadados, ou seja, “dados sobre dados”.

A estrutura da armazenagem e os métodos de acesso usados em um sistema de bancos de dados são especificados por um conjunto de definições em um tipo especial de DDL chamada linguagem de armazenagem e definição de dados. O resultado da compilação destas definições é um conjunto de instruções para especificar a implementação de detalhes do esquema de bancos de dados que estão escondidos dos usuários.

Dentre as instruções DDL's estão: **Create Table, Alter Table, Drop Table, Truncate, Rename**, etc.

1.15. LINGUAGEM DE MANIPULAÇÃO DE DADOS (DML)

A Linguagem de Manipulação de Dados (**Data Manipulation Language - DML**) é a linguagem que permite aos usuários fazer o acesso ou a manipular dados como organizados pelo modelo de dados apropriado. Existem dois tipos:

DML procedurais: requerem do usuário a especificação de qual dado é necessário e de como obtê-lo.

DML não procedurais: requerem do usuário a especificação de qual dado é necessário, sem especificar como obtê-lo.

Pode-se dizer que a manipulação de dados significa:

- a busca da informação armazenada no banco de dados.
- a inserção de novas informações nos bancos de dados.
- a eliminação de informações do banco de dados.
- a modificação de dados armazenados no banco de dados.

As instruções DML's são: **Select, Insert, Update e Delete**.

1.16. ADMINISTRADOR DE BANCO DE DADOS

O Administrador de Banco de Dados (**Database Administrator**), ou DBA é a pessoa responsável pelo controle central dos dados e dos programas de acesso a eles. Suas funções incluem:



Decidir qual o conteúdo de informação do banco de dados: identificar as entidades que interessem à empresa e a informação a ser registrada sobre essas entidades. Uma vez isto feito, o DBA tem então que definir o conteúdo do banco de dados escrevendo o esquema conceitual (usando a linguagem de definição de dados conceitual), que por sua vez é usado pelo SGBD ao responder às solicitações de acesso.

Decidir qual a estrutura de armazenagem e a estratégia de acesso: decidir também como os dados deverão estar representados no banco de dados, e precisa especificar esta representação escrevendo a definição da estrutura de dados (usando a linguagem de definição interna de dados).

Decidir quais as modificações de esquema e de organização física: decidir quais modificações devem ser feitas no esquema e na organização física, e como realizá-las.

Ser o elo de ligação com os usuários: é função do DBA servir como elemento de ligação com os usuários, para garantir a disponibilidade dos dados de que estes necessitam, e para preparar os esquemas externos que sejam necessários.

Decidir as verificações de autorização para acesso aos dados: decidir a quais partes do banco de dados os diversos usuários deverão ter acesso autorizado.

Decidir as especificações de restrição de integridade: decide através de restrições de integridade, que são mantidas em uma estrutura especial do sistema, se as atualizações feitas no sistema atendem as especificações exigidas.

É claro que o DBA irá precisar de diversos programas utilitários para ajudá-lo em suas tarefas. Esses utilitários devem ser uma parte essencial de um sistema prático de banco de dados. Abaixo encontramos alguns exemplos dos tipos de utilitários necessários:

- **Rotinas de carga:** para criar a versão inicial do banco de dados.
- **Rotinas de reorganização:** para, por exemplo, rearrumar o banco de dados e reutilizar espaços ocupados por dados obsoletos.
- **Rotinas de controle de uso:** para anotar cada operação feita no banco de dados, juntamente com a informação sobre o usuário que a realizou e os registros dos estados anteriores e posteriores.
- **Rotinas de recuperação:** para restaurar o banco de dados a um estado anterior depois de uma falha de hardware (máquina) ou de programação.



- **Rotinas de análise estatística:** para ajudá-lo na monitoração do desempenho.

1.17. USUÁRIOS DE BANCO DE DADOS

Programadores de Aplicativos: profissionais da computação interagem com o sistema por meio de chamadas a DML, que são embutidas em um programa escrito em uma linguagem hospedeira e/ou ferramentas de desenvolvimento de aplicações, como por exemplo: Java, Delphi, Visual Basic, .Net, etc. Estes programas são freqüentemente referenciados como programas de aplicação.

Uma vez que a sintaxe da DML é normalmente bastante diferente da sintaxe das linguagens hospedeiras, chamadas na DML são usualmente precedidas por um caractere especial e então o código apropriado pode ser gerado. Um pré-processador especial, chamado pré-compilador DML, converte o comando da DML em chamada de procedimento normal na linguagem hospedeira. O programa resultante é então passado pelo compilador da linguagem hospedeira, o qual gera o código-objeto apropriado.

Existem tipos especiais de linguagens de programação que combinam estruturas de controles de linguagens tipo Pascal com estruturas de controle para a manipulação de um objeto de um banco de dados (por exemplo, relações). Essas linguagens, algumas vezes chamadas de linguagens de quarta geração, freqüentemente incluem recursos especiais para facilitar a criação de formulários e a disposição de dados na tela. A grande maioria dos sistemas de bancos de dados comerciais inclui uma linguagem de quarta geração.

Usuários de Alto Nível: esses usuários interagem com o sistema sem escrever programas. Em vez disso, eles formulam suas consultas em uma linguagem de consulta ("QUERY") a banco de dados. Cada consulta é submetida a um processador de consulta, cuja função é gerar um comando da DML e dividi-lo em instruções que o gerenciador do banco de dados compreenda.

Usuários Especializados: alguns usuários padrões escrevem aplicativos especializados que não se ajustam a padrões tradicionais de processamento de dados. Entre esses aplicativos, estão os sistemas de projeto apoiado por computador (CAP), sistemas especialistas, sistemas que armazenam dados com tipos complexos (como, por exemplo, dados gráficos, dados de áudio (som) e dados de vídeo (imagem)) e sistemas de modelagem ambiental.

Usuários Ingênuos: esses usuários interagem com o sistema invocando um dos programas aplicativos permanentes. Por exemplo, um contador do banco que precisa transferir R\$ 50,00 da conta A para a conta B invocaria um



programa chamado transferência. Este programa perguntaria ao contador que quantidade de dinheiro está sendo transferida, a conta a partir da qual a transferência será feita e a conta para qual o dinheiro deve ser transferido.

1.18. ESTRUTURA GERAL DO SISTEMA

Os Componentes funcionais de um sistema de bancos de dados incluem:

Gerenciador de Arquivos: gerencia a alocação do espaço na armazenagem do disco e as estruturas de dados usadas para representar a informação armazenada no disco.

Gerenciador do Banco de Dados: fornece interface entre os dados de baixos níveis armazenados no disco e os programas aplicativos e de consulta submetidos ao sistema.

Processador de Consultas: traduz comandos numa linguagem de consulta para instruções de baixo nível que o gerenciador do banco de dados pode interpretar.

Pré-compilador da DML: converte comandos da DML embutidos em um aplicativo para chamadas de procedimentos normais na linguagem hospedeira.

Compilador da DDL: converte comandos da DDL em um conjunto de tabelas contendo metadados ou "dados sobre dados".

Estruturas de dados requeridas na implementação do sistema físico:

Arquivos de Dados: que armazenam o banco de dados propriamente dito

Dicionário de Dados: armazena metadados sobre a estrutura do banco de dados, é usado com frequência.

Índices: fornecem acesso rápido aos itens de dados guardando determinados valores.

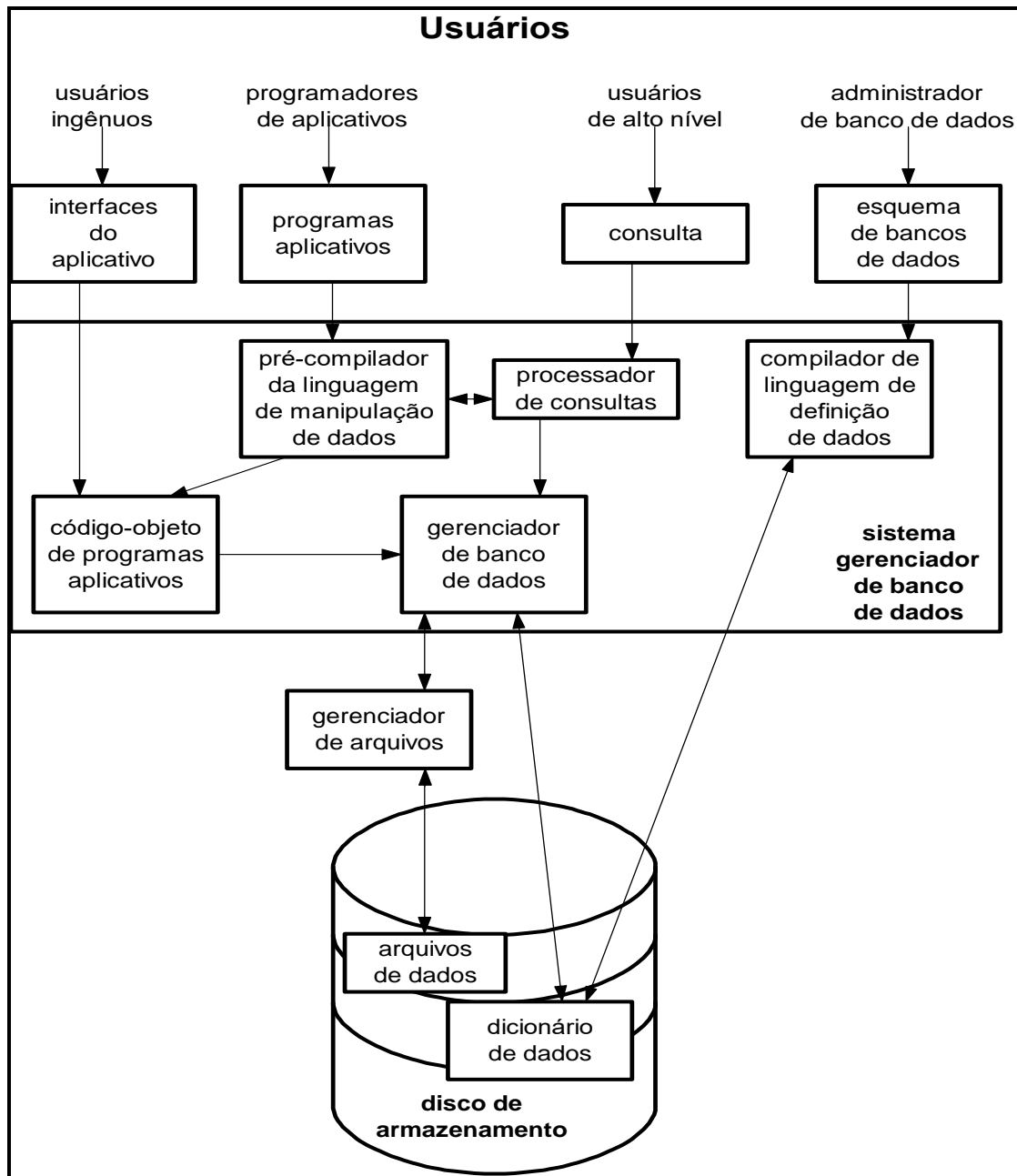


Figura 8. Ilustração Detalhada de um Sistema de Banco de Dados