

Administração de Banco de Dados

José Antônio da Cunha
CEFET-RN

Funcionamento do Otimizador de um SGBD

O Otimizador de um Sistema de Gerenciamento de Banco de Dados Relacionais (SGBDR) é responsável por analisar uma consulta SQL e escolher qual a forma mais eficiente de executá-la. A escolha leva em consideração várias informações internas no banco de dados.

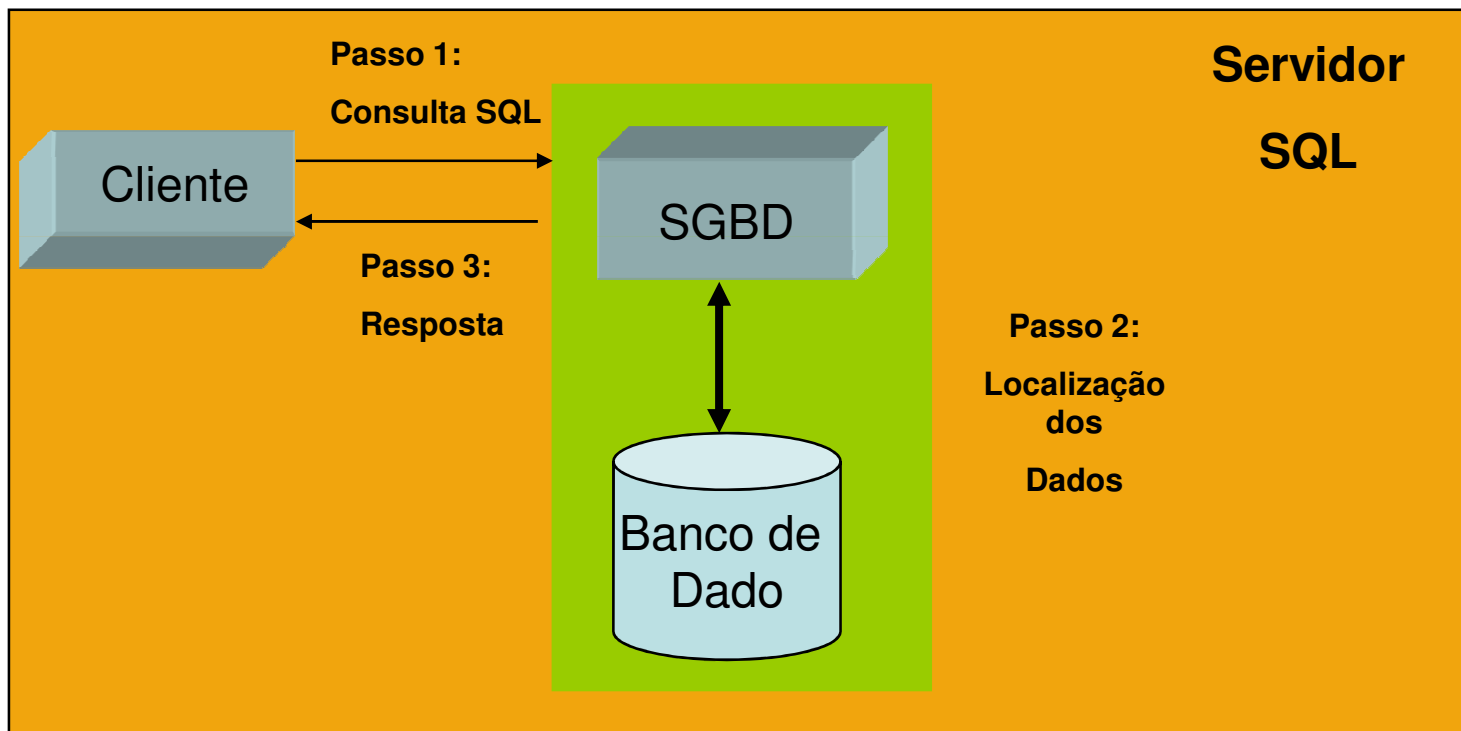


Figura 1: Representa os passos envolvidos na busca de informações.

Funcionamento do Otimizador de um SGBD

A otimização em um SGBDR pode ser feita porque em SQL não se expressa “**como**” a consulta deve ser executada, mas sim “**o que**” se pretende recuperar, permitindo que o otimizador escolha o “**como**” da maneira mais adequada em determinado momento.

Funcionamento do Otimizador de um SGBD

Seletividade de Atributos

É um conceito importante que afeta a decisão sobre usar ou não um índice. A seletividade estima, em média, a porcentagem dos registros que deverão aparecer na resposta, dado um determinado atributo. Ela pode ser calculada pela função:

$$\text{Seletividade} = 1 - (\text{registros-na-resposta-filtrada-pelo-atributo} / \text{total-de-registros})$$

Quanto mais próximo de 1, melhor é a seletividade de um atributo. Vejamos um exemplo: suponha uma tabela de cliente com 500 clientes. Vamos realizar uma consulta pelo CPF. Sabemos que o CPF não se repete e que existe um índice sobre este campo. Então teremos:

$$\text{Seletividade(CPF)} = 1 - (1/500) = 0.998$$

Isto indica que o CPF é um excelente atributo para realização de consultas através do uso de índice, pois cada valor de CPF descarta quase todos os registros da tabela.

Funcionamento do Otimizador de um SGBD

Seletividade de Atributos

Agora vamos realizar uma consulta pelo atributo cidade, sabendo que existem clientes de 110 cidades diferentes e que também existe um índice sobre esse campo.

$$\text{Selevidade(Cidade)} = 1 - (110/500) = 0.78$$

Esta pesquisa irá retornar em média mais de 20% dos registros. Isto pode indicar para o otimizador não utilizar o índice sobre o campo cidade e sim fazer uma varredura total sobre a tabela.

Vamos fazer alguns exemplos práticos:

Funcionamento do Otimizador de um SGBD

Seletividade de Atributos

Considere a tabela cliente composta pelos campos:

```
Cliente (nome_cliente, rua_cliente, cidade, obs1, ob2, ob3)
```

A tabela possui um total de 100.000 registros armazenados. Foi criado um índice de árvore B+ sobre o campo cidade, onde existem 95000 registros com valores iguais a “São Paulo” e 500 registros com valores iguais a “Fernandopolis”. Além disto, também existe um índice B+ sobre o campo chave nome_cliente.

Agora considere a seguinte consulta:

```
Select nome_cliente from cliente where cidade = “São Paulo”
```

Execute a consulta acima e analise se o otimizador usará ou não o índice sobre o campo cidade. Sugestão, analise o plano de execução.

Funcionamento do Otimizador de um SGBD

Seletividade de Atributos

Na consulta anterior, uma vez que 95% dos registros possuem o campo cidade igual a “São Paulo”, o uso do índice torna-se dispendioso, indicando para o otimizador que é mais econômico fazer uma varredura em toda a tabela cliente.

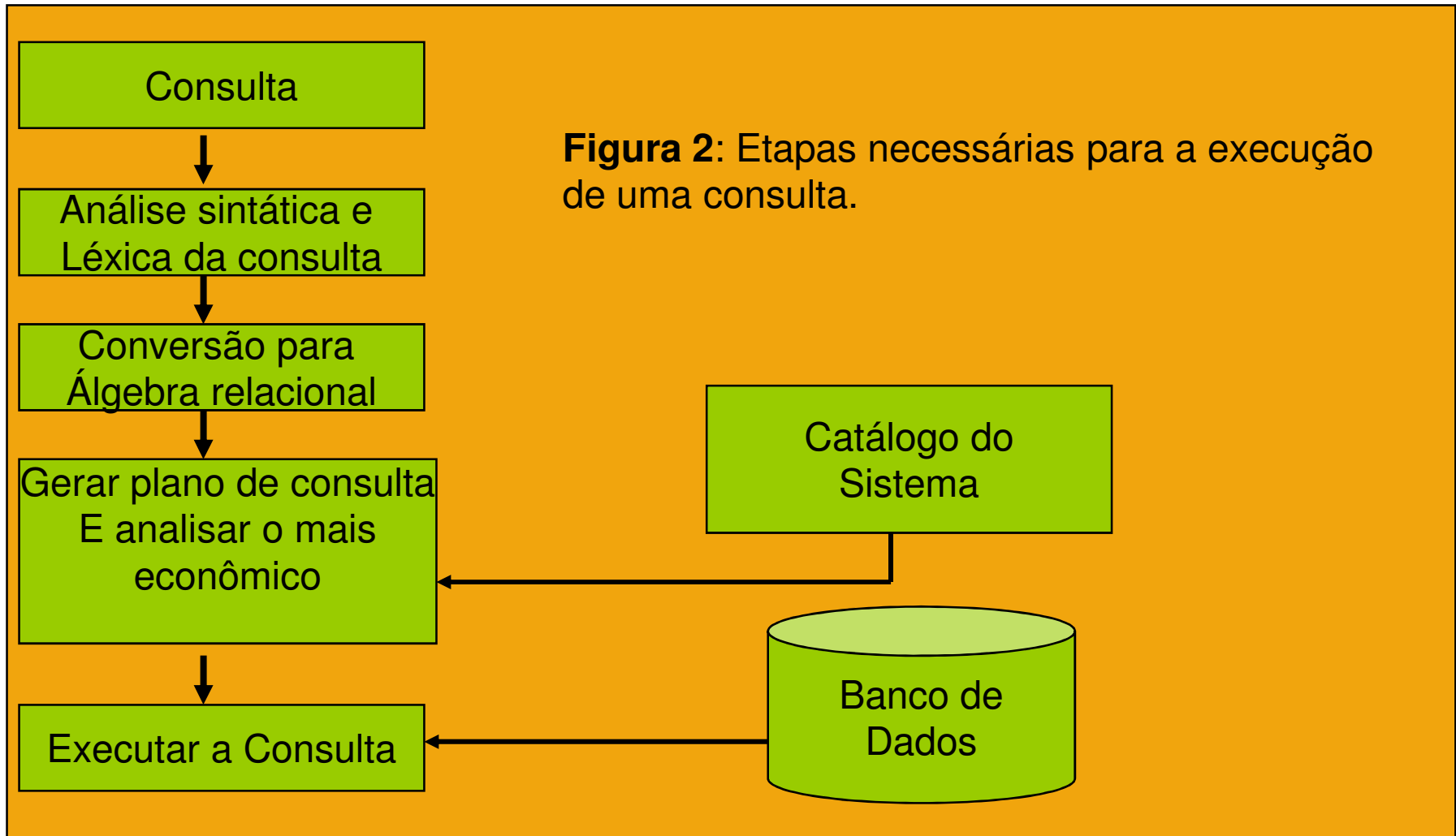
Agora vamos considerar a consulta:

```
Select nome_cliente from cliente where cidade = “Fernandopolis”
```

Uma vez que somente 5% dos registros possuem o valor “Fernandopolis” para o atributo cidade, é interessante para o otimizador fazer o uso do índice para resolver esta consulta.

Funcionamento do Otimizador de um SGBD

As fases de execução de uma consulta



Funcionamento do Otimizador de um SGBD

Fase 1 – Análise sintática e léxica da consulta

Nesta fase é feito um parsing. É aqui que o otimizador valida sintaticamente uma expressão SQL

Fase 2 – Conversão da consulta para a forma relacional

O modelo relacional permite que consultas sejam expressas através de operadores sobre as relações. As tabelas são consideradas conjuntos de registros e, portanto, podem ser operadas pelos operadores de conjunto como união (**U**), intersecção (\cap) e produto cartesiano (**X**). Além disso, podem ser filtradas por operadores de seleção (σ), ou podem ser projetadas (π) em relações com menos atributos.

Funcionamento do Otimizador de um SGBD

Fase 2 – Conversão da consulta para a forma relacional

Por tratar os algoritmos de manipulação de relações como operações bem definidas, o gerenciador pode usar as propriedades matemáticas desses operadores para escolher as melhores opções de consulta. Então, neste estágio, a consulta em SQL é convertida internamente para uma forma algébrica equivalente. Por exemplo, a consulta feita em SQL:

```
SELECT nome, telefone FROM cliente WHERE nome like 'Pedro%' AND  
cidade = 'São Carlos'
```

Pode ser expressa algebricamente, por exemplo como:

Alternativa 1: π (nome, telefone) (σ nome like 'Pedro%')
(σ cidade='São Carlos')(cliente))

Funcionamento do Otimizador de um SGBD

Fase 2 – Conversão da consulta para a forma relacional

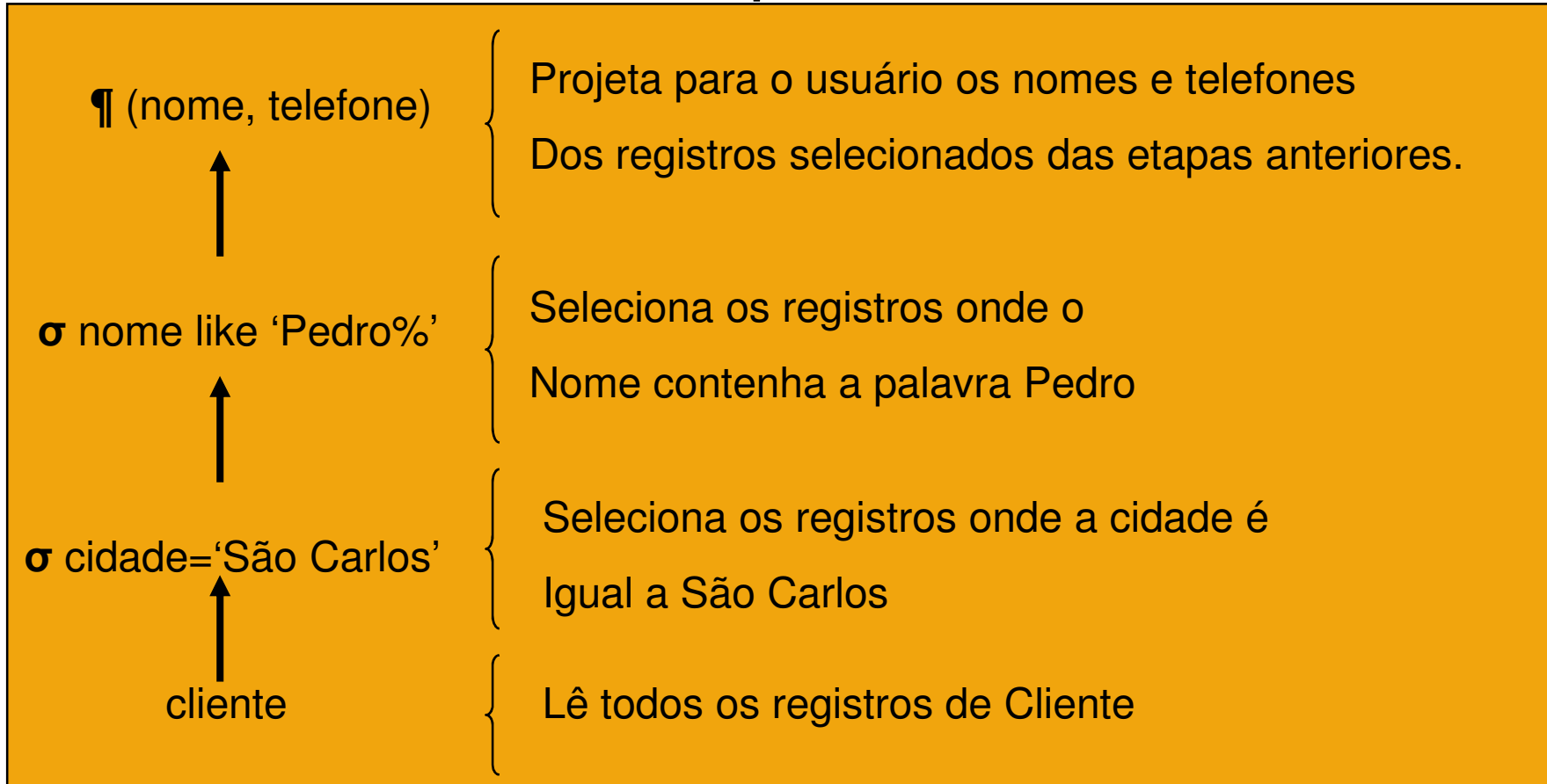


Figura 3: Árvore de Consulta referente à alternativa 1.

Funcionamento do Otimizador de um SGBD

Fase 2 – Conversão da consulta para a forma relacional

Uma alternativa é expressar a consulta original como:

Alternativa 2: π (nome, telefone) (σ (nome like 'Pedro%') \wedge σ cidade='São Carlos')(cliente))

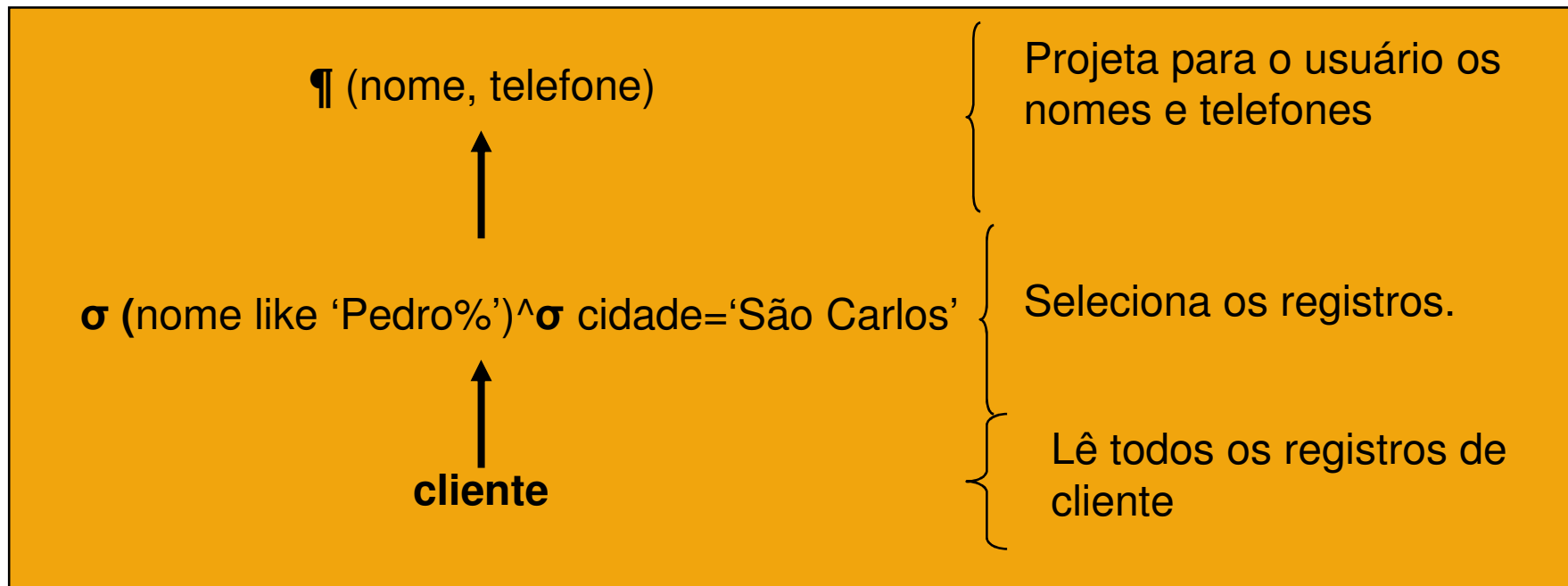


Figura 4: Árvore de Consulta referente à alternativa 2.

Funcionamento do Otimizador de um SGBD

Fase 3 – Gerar planos de consultas e analisar o mais econômico

Este passo envolve a criação de um conjunto de planos alternativos expressos em álgebra relacional, considerando aspectos como a existência de índices, distribuição de valores de dados (seletividade), agrupamento físico de dados armazenados, etc.

Funcionamento do Otimizador de um SGBD

Fase 4 – Executar a consulta

É neste estágio que a consulta escolhida será executada e o resultado, entregue para o usuário.

Exemplos práticos: Veja o script random. Analise cada um dos planos de execução.