

Resumo sobre algoritmos de operações da álgebra relacional



Dupla:

Gabriel Ferreira de Jesus

Rafael Oliveira Ledo

Resumo

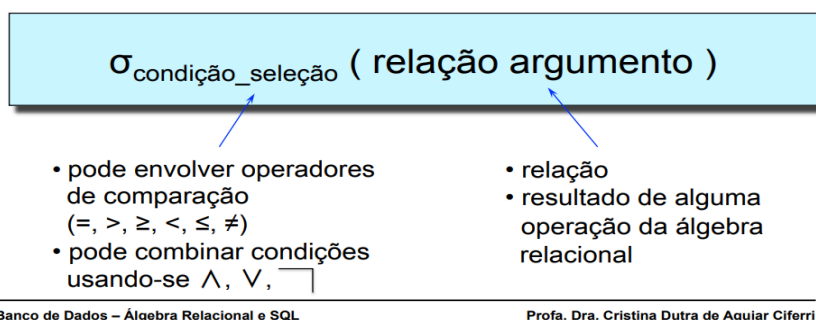
A álgebra relacional é uma ferramenta poderosa para entender o funcionamento de um banco de dados desde a base, com princípios e conhecimentos fundamentais para um analista de dados. A linguagem SQL usada na maior parte dos bancos de dados, é fundamentada em álgebra relacional que é uma derivação descendente da lógica de primeira ordem e da álgebra de conjuntos. Ela ficou conhecida em 1970 com a publicação do modelo relacional de dados de Edgar Frank Codd, que propôs a mesma como base das linguagens de consulta de banco de dados. A álgebra relacional pode ser entendida como uma coleção de operações de alto nível, que são elas: 1-restrição(seleção), 2-projeção, 3-produto, 4-união, 5-diferença, 6-interseção, 7-junção, 8-divisão, 9-agregação. Dessas 9 operações apenas as 5 primeiras respectivamente são primitivas as outras 4 são consideradas operações de conjuntos. Sua fundamentação formal baseada na lógica faz com que sua otimização seja possível e as melhorias possam ser contínuas a partir de uma análise detalhada das consultas. Abordaremos cada uma delas nesse resumo logo a seguir.

1. Seleção - σ

A operação de Seleção é uma das mais usadas sem sombra de dúvidas pelos usuários de SQL pois é a partir dela que selecionamos quais são os resultados queremos que sejam processados a partir da condição que iremos escolher.

Álgebra Relacional: Seleção

- **Seleciona tuplas** que satisfaçam à condição de seleção



Banco de Dados – Álgebra Relacional e SQL

Profa. Dra. Cristina Dutra de Aguiar Ciferri

Como podemos ver na explicação acima a seleção pode no seu contexto envolver operadores de comparação e ter uma relação com outras seleções ou operações de álgebra relacional.

Álgebra Relacional: Seleção

cliente (nro_cli, nome_cli, end_cli, saldo, cod_vend)

nro_cli	nome_cli	end_cli	saldo	cod_vend
1	Márcia	Rua X	100,00	1
2	Cristina	Avenida 1	10,00	1
3	Manoel	Avenida 3	234,00	1
4	Rodrigo	Rua X	137,00	2

Banco de Dados – Álgebra Relacional e SQL

Profa. Dra. Cristina Dutra de Aguiar Ciferri

Já aqui temos um exemplo prático de uma tabela cliente que retorna 2 resultados de 4 possíveis a partir de um commando :

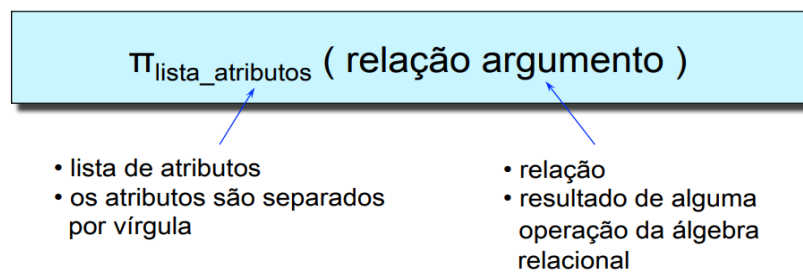
SELECT * FROM CLIENTE WHERE nro_cli = 2 or nro_cli = 3

2.Projeção- π

A operação de Projeção é recorrente nas mais diversas consultas sejam simples ou complexas, feita por profissionais ou amadores, a diferença para a seleção é que na projeção selecionamos as colunas de nosso interesse a serem consultadas.

Álgebra Relacional: Projeção

- **Projeta** as **colunas** solicitadas (i.e. produz um subconjunto vertical)



Na imagem acima temos um exemplo teórico da representação da projeção, com as colunas solicitadas representadas pela lista de atributos, isso produz justamente um subconjunto vertical.

Álgebra Relacional: Projeção

cliente (nro_cli, nome_cli, end_cli, saldo, cod_vend)

nro_cli	nome_cli	end_cli	saldo	cod_vend
1	Márcia	Rua X	100,00	1
2	Cristina	Avenida 1	10,00	1
3	Manoel	Avenida 3	234,00	1
4	Rodrigo	Rua X	137,00	2

Dentre as 5 opções de coluna da tabela cliente selecionamos 2 colunas que são elas nome_cli e end_cli representada no comando:

SELECT nome_cli, end_cli from CLIENTE

3. Produto Cartesiano - \times

O produto cartesiano também é bem utilizado em consultas, ele consiste na relação de $A \times B$ que formam um produto C .

Álgebra Relacional: Produto Cartesiano

- **Combina tuplas** de duas relações
 - relações não precisam ter atributos comum
- Tuplas da relação resultante
 - todas as combinações de tuplas possíveis entre as relações participantes

relação argumento 1 \times relação argumento 2

- relação
- resultado de alguma operação da álgebra relacional

Banco de Dados – Álgebra Relacional e SQL

Profa. Dra. Cristina Dutra de Aguiar Ciferri

Essa combinação de tuplas gera outra resultante como vemos no exemplo:

Cliente \times Vendedor

nro_cli	nome_cli	end_cli	saldo	cliente. cod_vend	vendedor. cod_vend	nome_vend
1	Márcia	Rua X	100,00	1	1	Adriana
1	Márcia	Rua X	100,00	1	2	Roberto
2	Cristina	Avenida 1	10,00	1	1	Adriana
2	Cristina	Avenida 1	10,00	1	2	Roberto
3	Manoel	Avenida 3	234,00	1	1	Adriana
3	Manoel	Avenida 3	234,00	1	2	Roberto
4	Rodrigo	Rua X	137,00	2	1	Adriana
4	Rodrigo	Rua X	137,00	2	2	Roberto

grau: número de atributos de cliente + número de atributos de vendedor

número de tuplas: número de tuplas de cliente * número de tuplas de vendedor

Banco de Dados – Álgebra Relacional e SQL

Profa. Dra. Cristina Dutra de Aguiar Ciferri

As tuplas de vendedor se propagam apesar de estarem em menor quantidade em relação as de vendedor.

4.União - U

A união de AUB, traz um resultante que é C, porém diferente do plano cartesiano não há repetição de tuplas o que melhora o desempenho em relação ao exemplo anterior.

Ex: (clientesA) \cup (clientesB)

nome	cpf
Daniel	1234
Carlos	2233
Diego	9908
Robert	3276

 \cup

nome	cpf
Vitor	4567
Henrique	8827

 =

nome	cpf
Daniel	1234
Carlos	2233
Diego	9908
Robert	3276
Vitor	4567
Henrique	8827

A vantagem de desempenho é nítida em relação ao plano cartesiano que repete tuplas sequencialmente.

5.Interseção - \cap

A interseção é a relação de $A \cap B$ com uma resultante C, porém aqui só temos as linhas em comum que existem nas duas tabelas.

Ex: (clientesA) \cap (clientesC)

nome	cpf
Daniel	1234
Carlos	2233
Diego	9908
Robert	3276

 \cap

nome	cpf
Vitor	4567
Carlos	2233
Diego	9908

 =

nome	cpf
Carlos	2233
Diego	9908

Podemos notar que apenas Carlos e Diego com seus cpfs respectivos que fazem parte de ambas as tabelas foram exibidos como resultado da consulta

6. Diferença de conjuntos - $-$

Como o nome já sugere aqui temos a diferença entre a relação de A-B com uma resultante C que são as tuplas que existem em A mas não em B.

Ex: $(\text{clientesA}) - (\text{clientesC})$

nome	cpf		nome	cpf		nome	cpf
Daniel	1234		Vitor	4567		Daniel	1234
Carlos	2233		Carlos	2233		Robert	3276
Diego	9908		Diego	9908			
Robert	3276						

Daniel e Robert são nomes que não existem na Tabela B e por isso são os resultantes da tabela C nesse exemplo acima.

7. Junção- \bowtie

A junção é um aprimoramento do plano cartesiano porque além de fazer uma combinação de tuplas ela também seleciona e por fim faz uma projeção, para remoção de duplicatas.

Álgebra Relacional: Junção

- Sintaxe

relação argumento 1 $\bowtie_{\text{condição_junção}}$ relação argumento 2

- relação
- resultado de alguma operação da álgebra relacional

É uma das operações mais usadas no SQL por ter resultados precisos e bem definidos a partir das condições selecionadas.

Ex: (cliente) ⋈ (veiculo)

nome	cpf
Daniel	1234
Carlos	2233
Diego	9908
Robert	3276

carro	cpf
Bmw	1234
Ferrari	9908

nome	cpf	carro
Daniel	1234	Bmw
Diego	9908	Ferrari

Nesse exemplo temos um relacionamento entre a tabela A ⋈ B e a condição para que seja feita é que o cpf seja igual nas duas colunas exibindo tanto informação de nome que vem da tabela A e de carro que vem da Tabela B , tendo como resultado a tabela C.

8. Divisão- ÷

Essa operação é bem simples já que a relação da divisão entre $A \div B$ gera a Tabela C que exibe os campos comuns entre A e B como vemos no exemplo a seguir:

Ex: (cliente) ÷ (cpfProcurado)

nome	cpf
Daniel	1234
Carlos	2233
Diego	9908
Robert	3276

cpf
1234
9908

nome
Daniel
Diego

9. Agregação- Γ

As funções de agregação são extremamente utilizadas no dia a dia de um analista de dados elas permitem simplificar as relações de consultas. Dentre elas temos:

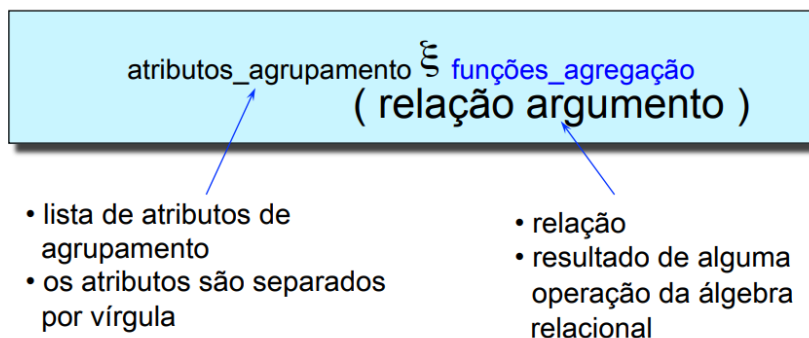
Funções de Agregação

- Funções
 - Média: **AVG**()
 - Mínimo: **MIN**()
 - Máximo: **MAX**()
 - Total: **SUM**()
 - Contagem: **COUNT**()
- Observação
 - **DISTINCT**: não considera valores duplicados
 - **ALL**: inclui valores duplicados

Dentre suas principais características temos que elas recebem uma coleção de valores e retornam um único valor como resultado, sendo numéricos ou não numéricos.

Álgebra Relacional: Agregação

- Permite a utilização de **funções de agregação**



A sua lógica é baseada em agrupamento uma lista de atributos é submetida a uma operação que gera um único resultado esperado.

REFERÊNCIAS :

https://www.ic.unicamp.br/~geovane/mo410-091/Ch04a-AlgR_pt.pdf

<http://wiki.icmc.usp.br/images/2/2c/SCC578920131-algebraSQL.pdf>

<https://spaceprogrammer.com/bd/aprendendo-as-principais-operacoes-da-algebra-relacional/>

https://pt.wikipedia.org/wiki/%C3%81lgebra_relacional

https://eadsalvador.ifba.edu.br/moodle/pluginfile.php/110876/mod_resource/content/1/AlgebraRelacional.pdf