

## 4 Álgebra Relacional

Conforme estudado anteriormente, um modelo de dados inclui um conjunto de operações para manipular o banco de dados, além dos conceitos de modelo de dados para a definição das restrições e estrutura do banco de dados. O conjunto básico de operações para o modelo relacional é a **álgebra relacional**. Essas operações permitem a um usuário especificar as solicitações básicas de recuperação. O resultado de uma recuperação será uma nova relação, que pode ter sido formada de uma ou mais relações. As operações de álgebra produzem, assim, novas relações, que podem ser manipuladas, adiante, usando-se as operações da mesma álgebra. Uma sequência de operações de álgebra relacional forma uma **expressão de álgebra relacional** cujos resultados também serão uma relação que representa o resultado de uma consulta de banco de dados (ou solicitação de recuperação).

A álgebra relacional é muito importante por diversas razões. Primeira, porque provê um fundamento formal para operações do modelo relacional. Segunda, e talvez a mais importante, porque é usada como uma base para implementar e otimizar as consultas em sistemas de gerenciadores de banco de dados relacional (SGBDRs). Terceira, alguns de seus conceitos são incorporados na linguagem de consulta-padrão SQL para os SGBDRs.

A álgebra relacional freqüentemente é considerada como uma parte do modelo relacional de dados e suas operações podem ser divididas em dois grupos. Um grupo inclui um conjunto de operações da teoria de conjunto matemática, onde essas operações são aplicadas porque cada relação é definida como um conjunto de tuplas no modelo relacional formal e é apresentada como **Grupos de Operadores Tradicionais**. O outro grupo consiste em operações desenvolvidas especificamente para os bancos de dados relacionais. Estas incluem os **Grupos de Operadores Especiais**, apresentados a seguir.

### 4.1 Grupos de Operadores:

#### a) Tradicionais:

- UNION;
- INTERSECTION;
- MINUS;
- PRODUCT.

#### b) Especiais:

- JOIN;
- DIVIDE;
- SELECT;
- PROJECT.

#### 4.1.1 JOIN

- Precisa de elementos comuns nos conjuntos envolvidos.
- *Sintaxe: **JOIN** relação a **AND** relação b **OVER** atributo comum*

#### 4.1.2 SELECT

- Seleciona linhas e só opera uma tabela;
- *Sintaxe:* **SELECT** *relação a* **WHERE** *condição*

#### 4.1.3 PROJECT

- Seleciona colunas;
- Elimina redundância;
- *Sintaxe:* **PROJECT** *atributo(s)* **OVER** *relação a*

#### 4.1.4 DIVIDE

- Apresenta os elementos de um conjunto que contém todos os elementos de um outro conjunto.
- *Sintaxe:* **DIVIDE** *relação a* **BY** *relação b*

#### 4.1.5 MINUS

- *Sintaxe:* *relação a* **MINUS** *relação b*

#### 4.1.6 UNION

- *Sintaxe:* *relação a* **UNION** *relação b*

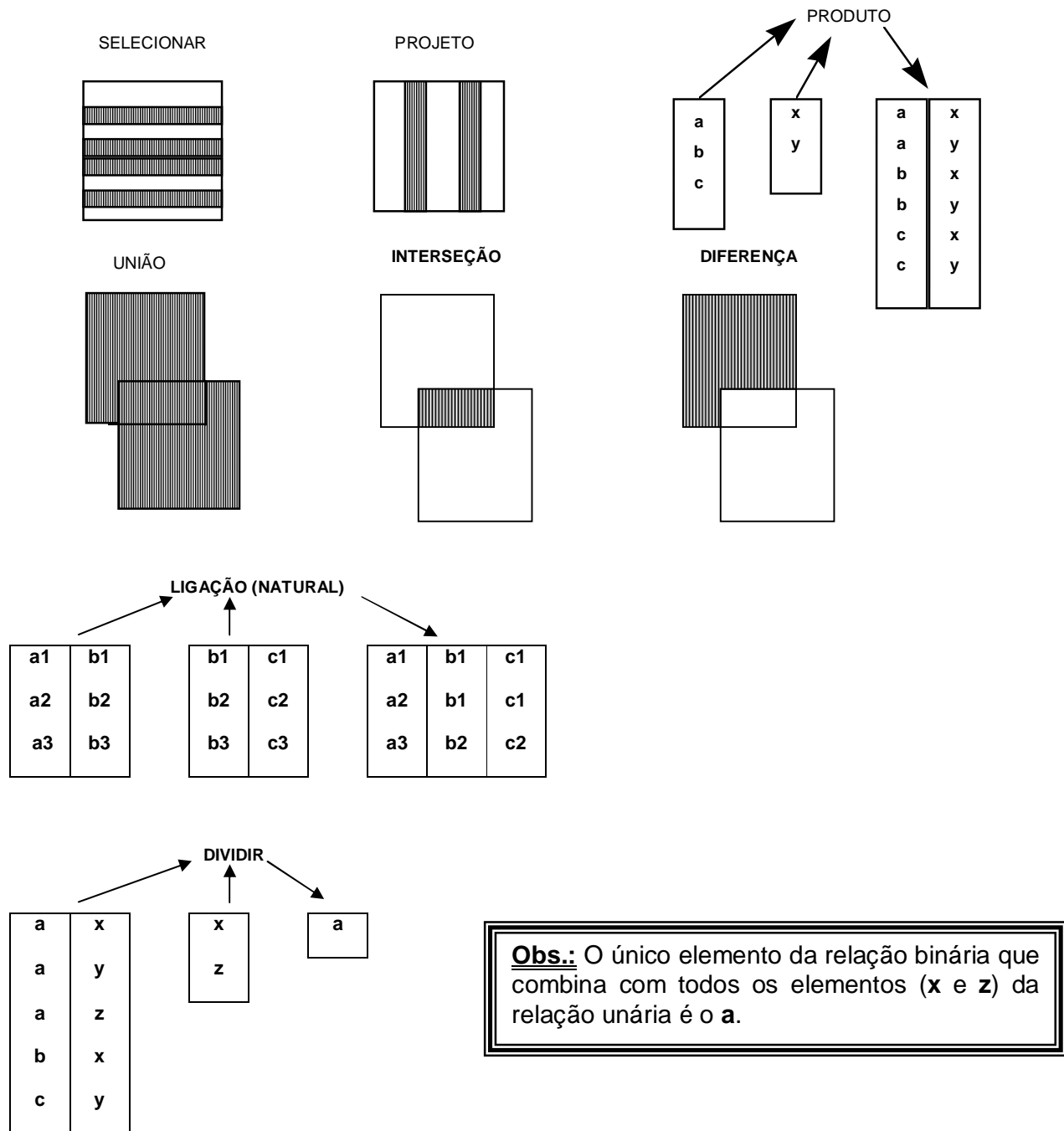
#### 4.1.7 INTERSECTION

- *Sintaxe:* *relação a* **INTERSECTION** *relação b*

#### 4.1.8 PRODUCT (TIMES)

- *Sintaxe:* *relação a* **TIMES** *relação b*

A Figura abaixo demonstra simbolicamente estas oito operações, e é explicado como segue:



**SELECT:** Extraí tuplas específicas de uma relação específica.

**PROJECT:** Extraí atributos específicos de uma relação específica.

**PRODUCT (TIMES):** Constrói uma relação a partir de duas relações específicas consistindo em todas as possibilidades de pares de tuplas concatenados, uma de cada duas relações específicas.

**UNION:** Constrói uma relação consistindo em todas as tuplas que aparecem em ambos os pares de relações específicas.

**INTERSECTION:** Constrói uma relação consistindo em todas as tuplas comuns que aparecem em ambos os pares de relações específicas.

**MINUS:** Constrói uma relação, consistindo em todas as tuplas que aparecem na primeira, mas não na segunda, do par de relações específicas.

**JOIN:** Constrói uma relação a partir de duas relações específicas, consistindo em todas as possibilidades de pares de tuplas concatenados, uma de cada uma das duas relações específicas, de forma que em cada par as duas tuplas satisfaçam uma condição específica.

**DIVIDE:** Toma duas relações, uma binária e outra unária, e constrói uma relação consistindo em todos os valores de um atributo da relação binária com equivalência (no outro atributo) a todos os valores da relação unária.

Observemos que a saída de cada uma das operações algébricas é (naturalmente) uma outra relação.

**Nota:** No conjunto tradicional de operações (união, interseção e diferença), as duas relações envolvidas devem ser compatíveis com a união, isto é, o atributo de número  $i$  de cada ( $i=1,2,\dots,n$ ), deve basear no mesmo domínio (não precisa ter o mesmo nome). A regra de compatibilidade com a união impõe-se de forma a assegurar que o resultado ainda seja uma relação. Isto porque, de outro modo, como por exemplo, ao formar a união da relação EMP (matric, nome, salário, cdepto) com PROJ (cproj, nproj), o resultado, embora ainda um conjunto, certamente não seria uma relação, seria uma mistura heterogênea de tuplas da relação EMP com as tuplas da relação PROJ).

## ÁLGEBRA RELACIONAL – Operadores Básicos (Resumo) e suas duas formas de escrever:

- seleção ou restrição:  $\sigma$  ou **WHERE**
- projeção:  $\Pi$  ou **{ }**
- união:  $\cup$  ou **UNION**
- interseção de conjuntos ou interseção :  $\cap$  ou **INTERSECT**
- diferença de conjuntos ou diferença:  $-$  ou **MINUS**
- produto cartesiano ou produto : **X** ou **TIMES**
- junção natural ou junção: **|X|** ou **JOIN**
- divisão:  $\div$  ou **DIVIDE**
- atribuição:  $\leftarrow$
- renomeação:  $\rho$

## EXEMPLOS DE ALGUNS OPERADORES

## UNIÃO

“Operação sobre duas relações que gera uma terceira relação contendo todas as tuplas das duas relações sem repetições.”

Tem duas relações como entrada e uma relação como saída

As relações de entrada precisam ser do mesmo tipo, ou seja, ter a mesma quantidade de colunas, na mesma ordem e do mesmo tipo, mas não necessariamente com o mesmo nome.

Sintaxe:

Relação1 U Relação2      ou      Relação1 UNION Relação2

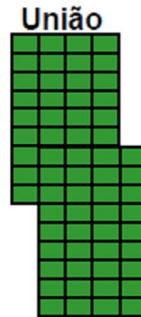
Exemplo:

A		B		A U B	
Matr	Nome	Matr	Nome	Matr	Nome
02	João	01	Gomes	02	João
01	Maria	07	José	01	Maria
07	José	04	Davi	07	José
08	Carla			08	Carla
				01	Gomes
				04	Davi

Note que “A U B” é uma relação;

A tupla (“07”, “José”) aparece apenas uma vez na relação “A U B”, pois, não podem ocorrer repetições;

Duas tuplas só são consideradas repetidas se todos os seus atributos forem iguais, ainda não existe o conceito de “chave primária”, assim, (“01”, “Maria”) e (“01”, “Gomes”) são duas tuplas diferentes.



## INTERSEÇÃO

“Operação sobre duas relações que gera uma terceira relação contendo as tuplas comuns às duas relações.”

Tem duas relações como entrada e uma relação como saída

As relações de entrada precisam ser do mesmo tipo, ou seja, ter a mesma quantidade de colunas, na mesma ordem e do mesmo tipo, mas não necessariamente com o mesmo nome.

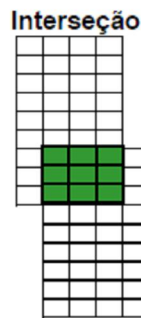
Sintaxe:

Relação1  $\cap$  Relação2      ou      Relação1 INTERSECT Relação2

Exemplo:

A		B		A $\cap$ B	
Matr	Nome	Matr	Nome	Matr	Nome
02	João	01	Gomes	07	José
01	Maria	07	José		
07	José	04	Davi		
08	Carla				

Apenas tupla (“07”, “José”) aparece nas duas relações.



## DIFERENÇA

“Operação sobre duas relações que gera uma terceira relação contendo as tuplas da primeira relação que não existem na segunda.”

Tem duas relações como entrada e uma relação como saída

As relações de entrada precisam ser do mesmo tipo, ou seja, ter a mesma quantidade de colunas, na mesma ordem e do mesmo tipo, mas não necessariamente com o mesmo nome.

Sintaxe:

Relação1 - Relação2 ou Relação1 MINUS Relação2

Exemplo:

A		B		A - B		B - A	
Matr	Nome	Matr	Nome	Matr	Nome	Matr	Nome
02	João	01	Gomes	02	João	01	Gomes
01	Maria	07	José	01	Maria	04	Davi
07	José	04	Davi	08	Carla		
08	Carla						

Note que a tupla (“07”, “José”) não aparece nas relação resultante.

É importante ressaltar que “A - B” é diferente de “B - A”.

Diferença

## PRODUTO

“Operação sobre duas relações que gera uma terceira relação contendo todas combinações entre as tuplas das duas relações.”

Tem duas relações como entrada e uma relação como saída

As relações de entrada podem ser de tipos diferentes.

A Quantidade de atributos resultantes será a quantidade de atributos da primeira relação + da segunda relação e a quantidade de tuplas será a quantidade de tuplas da primeira multiplicado da segunda

Sintaxe:

Relação1 X Relação2 ou Relação1 TIMES Relação2

Exemplo:

A relação “A X B” tem 4 atributos (4=2+2, 2 de A + 2 de B) e 12 tuplas (12=4X3, 4 de A X 3 de B).

A		B	
Matr	Nome	Matr	Nome
02	João	01	Gomes
01	Maria	07	José
07	José	04	Davi
08	Carla		

A X B			
Matr	Nome	Matr	Nome
02	João	01	Gomes
02	João	07	José
02	João	04	Davi
01	Maria	01	Gomes
01	Maria	07	José
01	Maria	04	Davi
07	José	01	Gomes
07	José	07	José
07	José	04	Davi
08	Carla	01	Gomes
08	Carla	07	José
08	Carla	04	Davi

Produto Cartesiano

a	x	a x
b	y	a y
c		b x
		b y
		c x
		c y

Entrada

Saída