



Sistemas de Informação

# **Banco de Dados 1**

Prof. Dr. Ronaldo Castro de Oliveira

[ronaldo.co@ufu.br](mailto:ronaldo.co@ufu.br)

FACOM



# Álgebra Relacional

# Introdução

- Relembrando:

- Um modelo de dados inclui um conjunto de **OPERAÇÕES** para manipular um banco de dados além dos **CONCEITOS** de modelagem necessários para a estruturação do BD.
- **OPERAÇÕES**: Linguagem de Consulta

- O que é **LINGUAGEM DE CONSULTA**?

- É uma linguagem por meio da qual os usuários obtém informações do banco de dados
- Linguagens de mais **ALTO NÍVEL** que as linguagens de programação tradicionais
  - Exemplo - SQL - Structured Query Language

- O que é **LINGUAGEM FORMAL DE CONSULTA** ?

# Álgebra Relacional

## ◆ Modelo de dados inclui

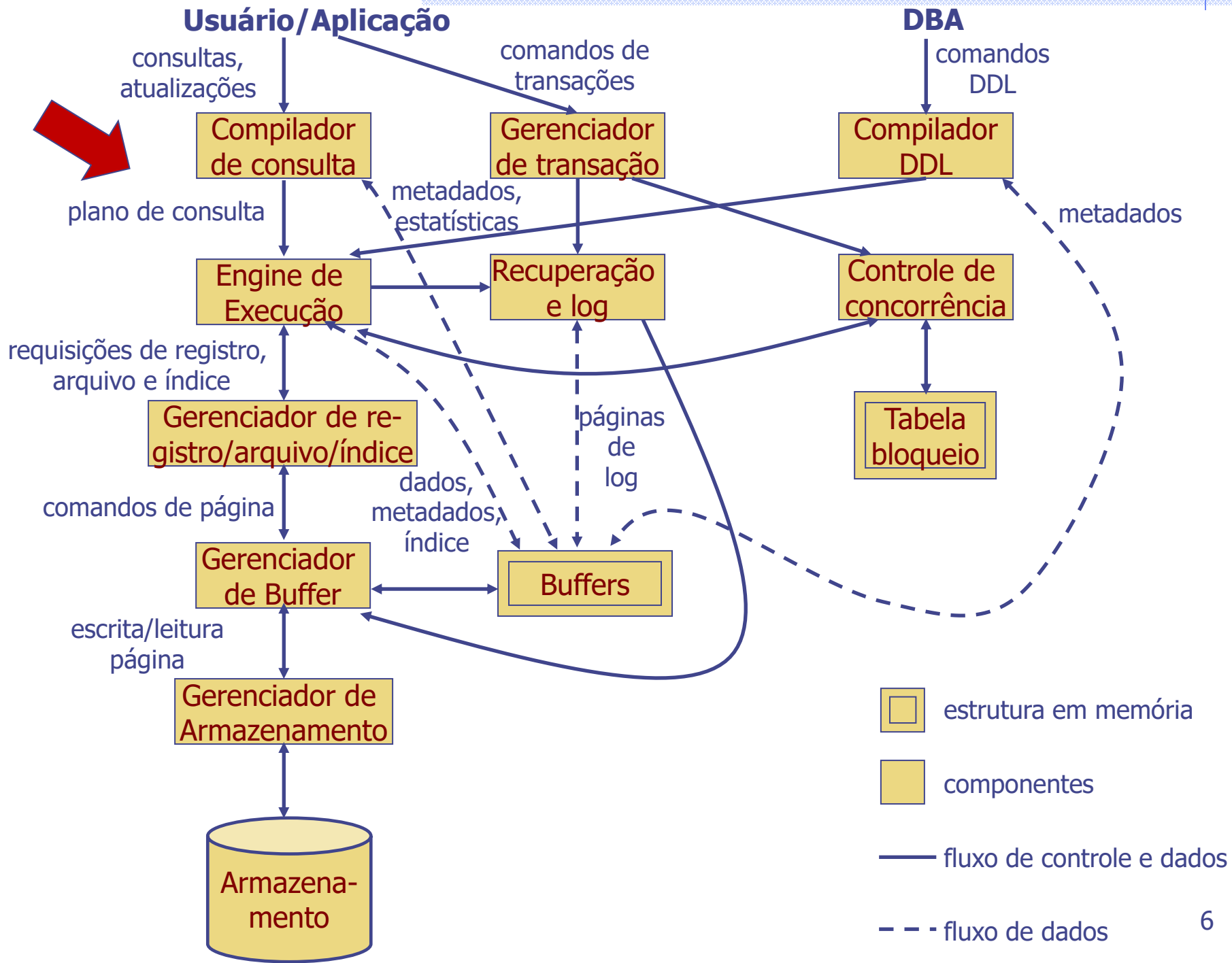
- Conceitos para a definição das restrições e estrutura do BD
- **Conjunto de operações** para manipular o BD

## ◆ Álgebra relacional

- Maneira teórica de se manipular o BD relacional

## ◆ Importância

- Fundamento formal para as operações no modelo relacional
- Base para implementar e otimizar consultas em SGBDR
- Introduz conceitos incorporados na SQL



# Álgebra Relacional

- ◆ Linguagem de consulta procedural
  - usuários especificam os dados necessários e como obtê-los
- ◆ Consiste de um conjunto de operações
  - entrada: uma ou mais relações e restrições
  - saída: uma nova relação resultado

# Classificação das Operações

## ◆ Unárias

- seleção
- projeção
- renomear

operam sobre uma  
única relação

## ◆ Binárias

- produto cartesiano
- união
- diferença de conjuntos
- intersecção de conjuntos
- junção natural
- divisão

operam sobre duas  
relações

# Esquema relacional

cliente (nro\_cli, nome\_cli, end\_cli,  
saldo, vendedor)

vendedor (cod\_vend, nome\_vend)

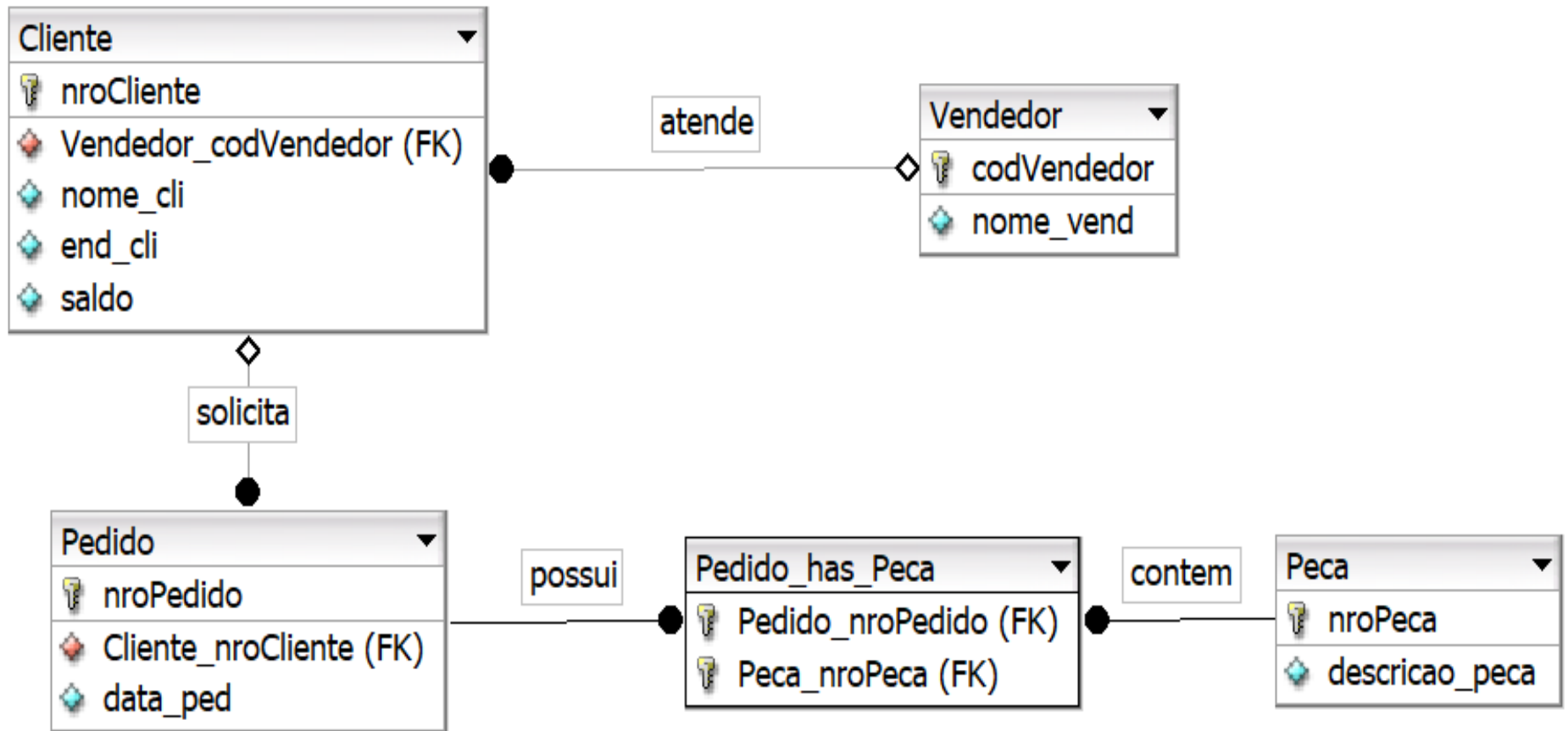
pedido (nro\_ped, data, nro\_cliente)

pedido\_peça (nro\_ped, nro\_peça)

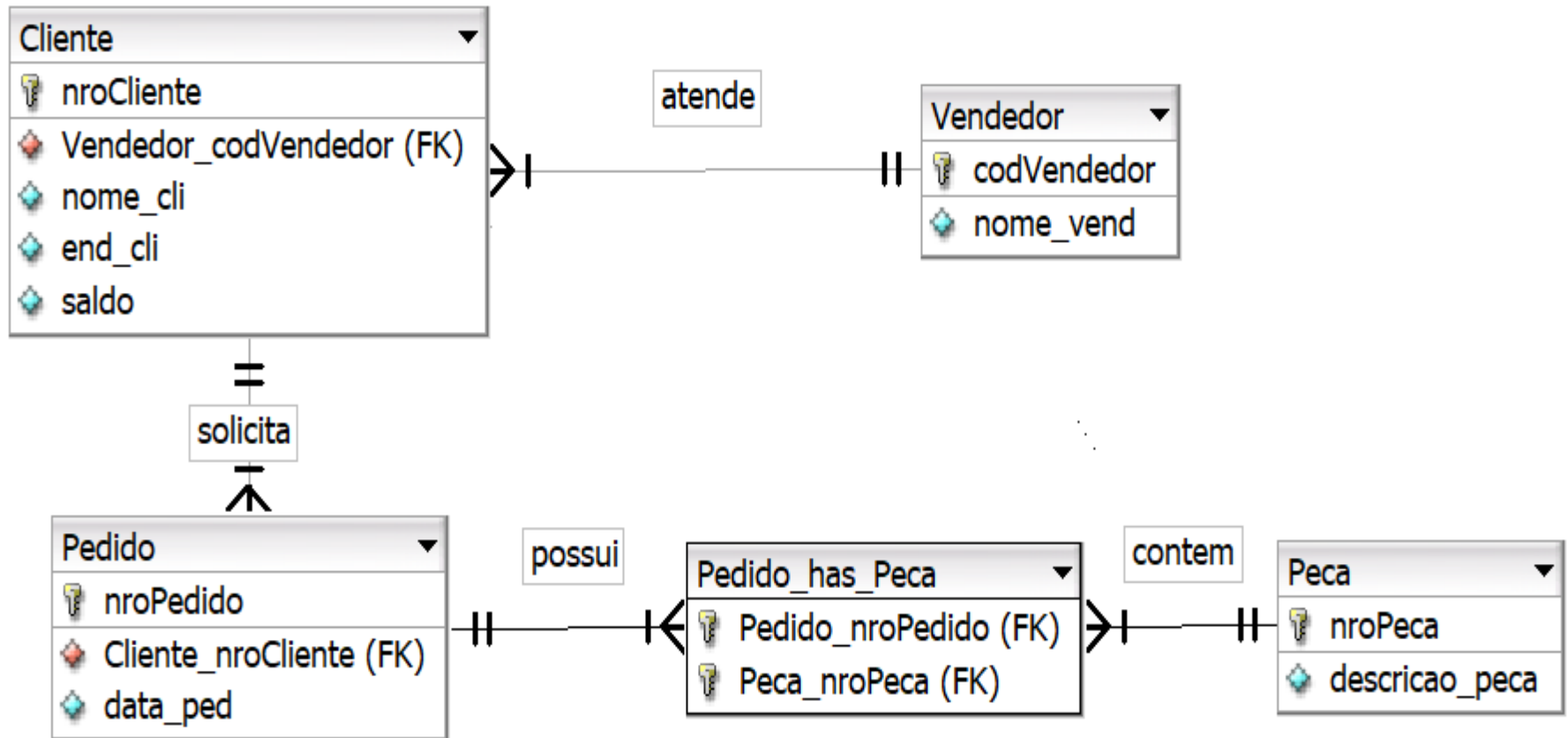
peça (nro\_peça, descrição\_peça)



# Esquema relacional – IDEF1X



# Esquema relacional – Crows Foot



# Seleção $\sigma$ sigma

- ◆ Selecciona tuplas da relação argumento que satisfaçam à condição de seleção

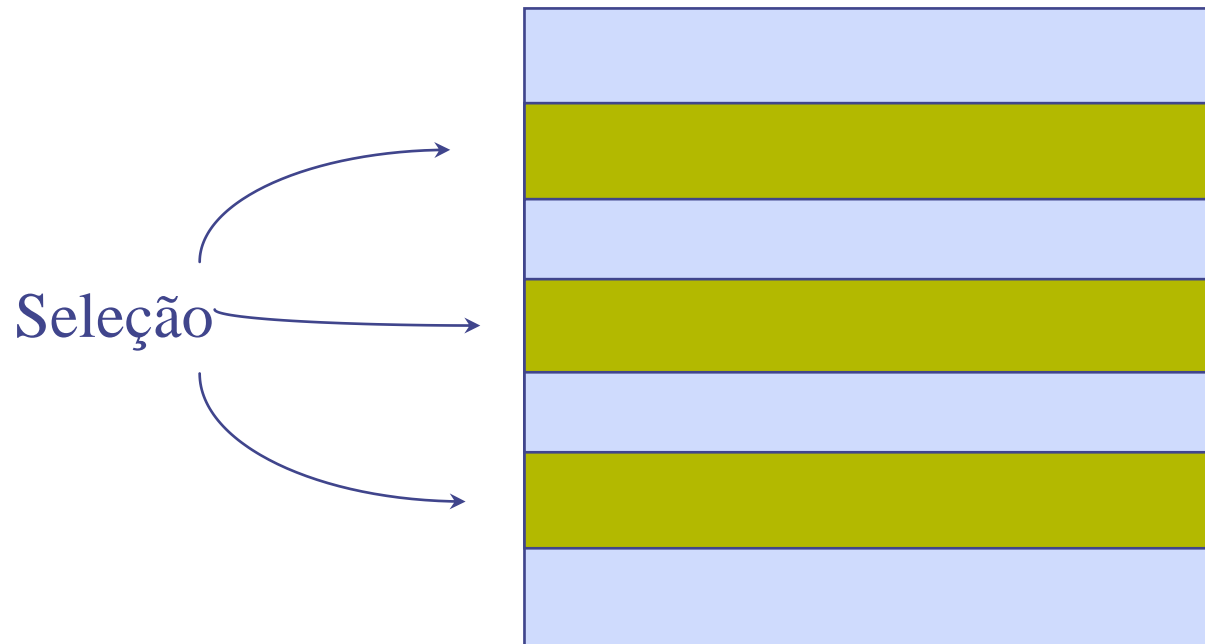
$\sigma_{\text{condição\_seleção}}$  ( relação argumento )

- pode envolver operadores de comparação  
(=, <, ≤, >, ≥, ≠)
- pode combinar condições usando-se  $\wedge$ ,  $\vee$ ,  $\neg$

- relação
- resultado de alguma operação da álgebra relacional

# Seleção $\sigma$ sigma

- Produz um subconjunto **horizontal** de uma relação



# Relação Cliente

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

# Consulta 1

- ◆ Liste toda a informação da relação cliente referente ao cliente de número 4.

$\sigma_{\text{nro\_cli} = 4} (\text{cliente})$

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

# Consulta 1

## ◆ Relação resultado



nro_cli	nome_cli	end_cli	saldo	cod_vend
4	Rodrigo	Rua X	137,00	2

grau: mesmo grau da  
relação argumento

número de tuplas:  
menor ou igual ao  
número de tuplas da  
relação argumento

# Consulta 2

- ◆ Liste toda a informação da relação cliente para clientes que possuam saldo inferior a R\$ 200,00 e que morem na Rua X.

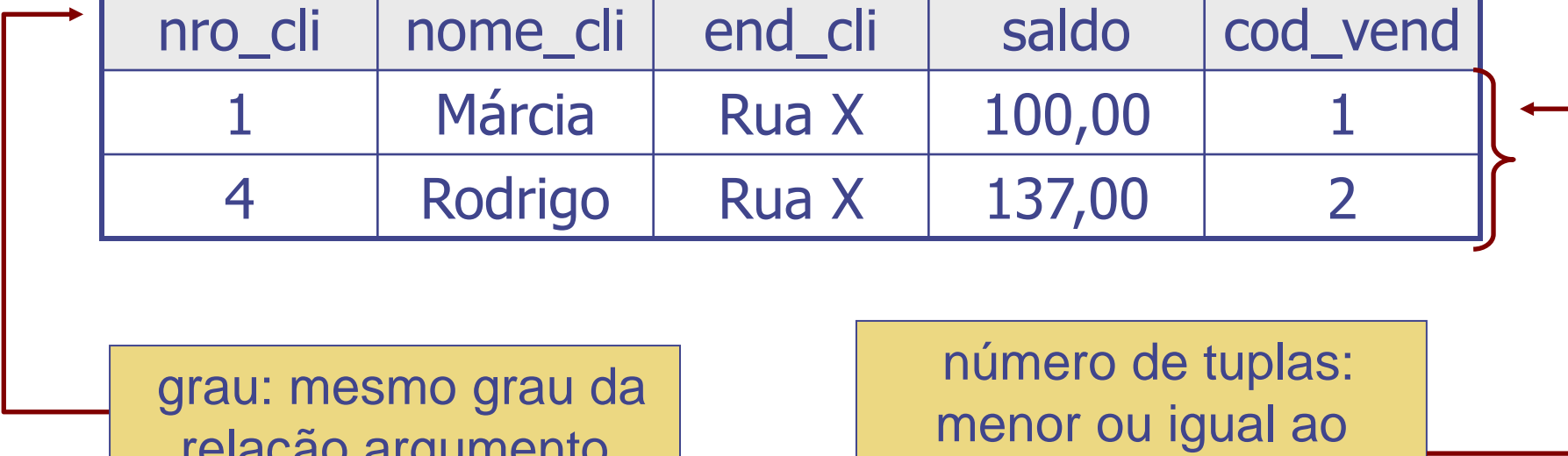
$\sigma_{\text{saldo} < 200,00 \wedge \text{end\_cli} = \text{"Rua X"}}(\text{cliente})$

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



# Consulta 2

## ◆ Relação resultado



nro_cli	nome_cli	end_cli	saldo	cod_vend
1	Márcia	Rua X	100,00	1
4	Rodrigo	Rua X	137,00	2

grau: mesmo grau da  
relação argumento

número de tuplas:  
menor ou igual ao  
número de tuplas da  
relação argumento

# Vale lembrar...

◆ As condições booleanas  $\wedge$  (and),  $\vee$  (or) e  $\neg$  (not) têm sua interpretação conforme segue:

- $(\text{cond1} \wedge \text{cond 2})$

- ◆ é verdadeira se ambas cond1 e cond2 forem verdadeiras
- ◆ caso contrário é falsa

- $(\text{cond1} \vee \text{cond 2})$

- ◆ verdadeira se cond1 ou cond2 forem verdadeiras
- ◆ caso contrário é falsa

- $(\neg \text{cond})$

- ◆ verdadeira se cond for falsa
- ◆ caso contrário é falsa

# Projeção

- ◆ Produz uma nova relação contendo um subconjunto vertical da relação argumento, sem duplicações

$\pi_{\text{lista\_atributos}}$  ( relação argumento )

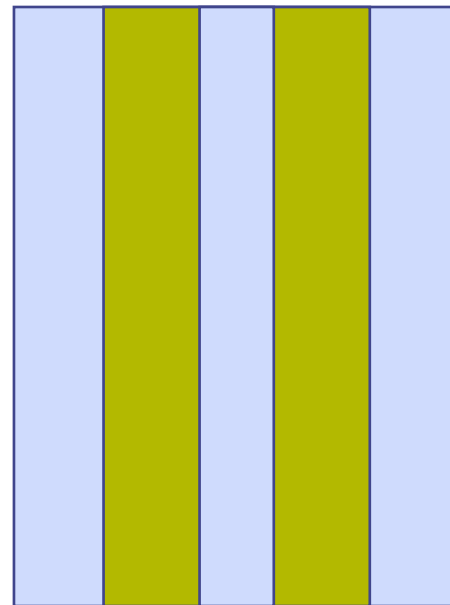
- lista de atributos
- os atributos são separados por vírgula

- relação
- resultado de alguma operação da álgebra relacional

# Projeção

- Extrai atributos (Colunas) específicos de uma relação específica
- Produz um subconjunto **vertical** de uma relação


## Projeção

 $\pi$  pi

# Consulta 3

- ◆ Liste o número e o nome de todos os clientes

$\pi_{\text{nro\_cli, nome\_cli}}(\text{cliente})$



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

# Consulta 3

## ◆ Relação resultado

grau: número de atributos listados em lista\_atributos



nro_cli	nome_cli
1	Márcia
2	Cristina
3	Manoel
4	Rodrigo

número de tuplas: menor ou igual ao número de tuplas da relação argumento

# Consulta 4


◆ Liste o endereço de todos os clientes

$\pi_{\text{end\_cli}}(\text{cliente})$



end_cli
Rua X
Avenida 1
Avenida 3

Relação Resultante  
- sem repetição



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

# Operações

## Propriedades dos operadores unários

◆ A operação de Seleção é comutativa

- $\sigma_{\langle \text{condição-A} \rangle}(\sigma_{\langle \text{condição-B} \rangle})$

◆ Uma sequência de seleções pode ser executada em qualquer ordem, ou pode ser transformada em uma única seleção com uma condição conjuntiva (termos cujo valor é VERDADEIRO ou FALSO, ligados pelo operador  $\wedge$  (AND))

- $\sigma_{\langle \text{condição-1} \rangle}(\sigma_{\langle \text{condição-2} \rangle}(\dots(\sigma_{\langle \text{condição-n} \rangle}(R))))$

- $\sigma_{\langle \text{condição-1} \rangle} \wedge \langle \text{condição-2} \rangle \wedge \dots \langle \text{condição-n} \rangle (R)$



# Operações

## Propriedades dos operadores unários

- ◆ A operação de Projeção não é comutativa
- ◆ Se  $\langle \text{lista\_atribos\_B} \rangle$  contém  $\langle \text{lista\_atribos\_A} \rangle$ , então ambas as expressões seguintes são corretas, e vale a igualdade:

$$\blacksquare \pi_{\langle \text{lista\_atribos\_A} \rangle} (\pi_{\langle \text{lista\_atribos\_B} \rangle} R) = \pi_{\langle \text{lista\_atribos\_A} \rangle} R$$

# Consulta 5

- ◆ Liste o número e o nome de todos os clientes que possuam saldo devedor inferior a 200,00 reais e morem na Rua X.

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

# Consulta 5

## ◆ Passos

- realizar uma operação de seleção para criar uma nova relação que contém somente aqueles clientes com o saldo e o endereço apropriados;
- realizar uma projeção sobre a relação resultante do passo anterior, restringindo o resultado desejado às colunas indicadas.

# Consulta 5

## ◆ Primeiro passo


$\sigma$  saldo\_dev < 200,00  $\wedge$  end\_cli = "Rua X" (cliente)

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

# Consulta 5

## ◆ Segundo passo

$\pi_{\text{nro\_cli, nome\_cli}}$  (primeiro passo)



nro_cli	nome_cli	end_cli	saldo	cod_vend
1	Márcia	Rua X	100,00	1
4	Rodrigo	Rua X	137,00	2

# Consulta 5

- ◆ Liste o número e o nome de todos os clientes que possuam saldo inferior a R\$ 200,00 e que morem na Rua X.

$\pi_{\text{nro\_cli}, \text{nome\_cli}} (\sigma_{\text{saldo\_dev} < 200,00 \wedge \text{end\_cli} = \text{"Rua X"}} (\text{cliente}))$

## ■ Relação Resultado

nro_cli	nome_cli
1	Márcia
4	Rodrigo

# Atribuição

## ◆ Funcionalidades

- associa uma relação argumento a uma relação temporária
- permite o uso da relação temporária em expressões subsequentes

relação temporária ← relação argumento

# Consulta 5

- ◆ Liste o número e o nome de todos os clientes que possuam saldo inferior a R\$ 200,00 e que morem na Rua X.

$\pi_{\text{nro\_cli, nome\_cli}} (\sigma_{\text{saldo\_dev} < 200,00 \wedge \text{end\_cli} = \text{"Rua X"}} (\text{cliente}))$

- ◆ Usando atribuição

- $\text{temp} \leftarrow \sigma_{\text{saldo\_dev} < 200,00 \wedge \text{end\_cli} = \text{"Rua X"}} (\text{cliente})$
- $\pi_{\text{nro\_cli, nome\_cli}} (\text{temp})$



# Atribuição

## ◆ Características adicionais

- permite renomear os atributos de relações intermediárias e final
- $R(\text{código}, \text{nome}) \leftarrow \pi_{\text{nro\_cli}, \text{nome\_cli}}(\text{temp})$

## ◆ Observações

- não adiciona semântica adicional à álgebra relacional
- geralmente utilizada para expressar consultas complexas

# Renomear ( $\rho = r\hat{o}$ )

## ◆ Renomeia

- nome da relação
- nomes dos atributos da relação
- nome da relação e nomes dos atributos

$\rho_{S(B_1, B_2, \dots, B_n)}$  (relação)

nome *novo* da relação

nome *antigo* da relação

nomes *novos* dos atributos

# Renomear

## ◆ Exemplos

- $\rho_{\text{comprador}}$  (cliente)
- $\rho_{(\text{código}, \text{nome}, \text{rua}, \text{saldo}, \text{vendedor})}$  (cliente)
- $\rho_{\text{comprador}(\text{código}, \text{nome}, \text{rua}, \text{saldo}, \text{vendedor})}$  (cliente)

## ◆ Observação

- indicada para ser utilizada quando uma relação é usada mais do que uma vez para responder à consulta

# Produto Cartesiano

- ◆ Combina tuplas de duas relações (quaisquer)
- ◆ Tuplas da relação resultante
  - todas as combinações de tuplas possíveis entre as relações participantes

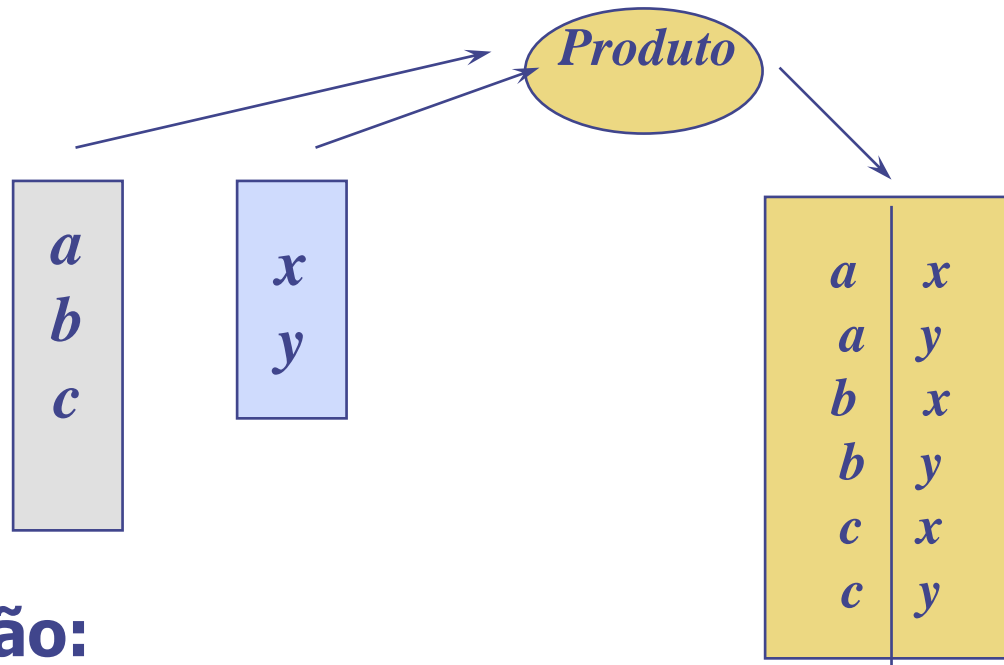
relação argumento 1 x relação argumento 2



- relação
- resultado de alguma operação da álgebra relacional

# Produto Cartesiano

- Utilizado quando se necessita obter dados presentes em duas ou mais relações



**Notação:**

**<relação 1> X <relação 2>**

# Relações Cliente e Vendedor

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

vendedor (cod\_vend, nome\_vend)

cod_vend	nome_vend
1	Adriana
2	Roberto

# Cliente x Vendedor

nro_cli	nome_cli	end_cli	saldo	cliente. cod_vend	vendedor.c od_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

# Exemplo 1

## ◆ Considere as seguintes relações

- usuário ( cliente\_nome, gerente\_nome )
- cliente ( cliente\_nome, rua, cidade )

cliente_nome	gerente_nome
Márcia	Manoel
Rodrigo	Maria
Bruno	Manoel

cliente_nome	rua	cidade
Márcia	Rua X	Itambé
Rodrigo	Rua X	Maringá
Bruno	Rua Z	Uberlândia

- ## ◆ Liste o nome de todos os usuários atendidos pelo gerente Manoel, assim como as cidades nas quais eles vivem



# Solução

## ◆ Primeiro passo

- determinar quem são os usuários atendidos pelo gerente Manoel

$\text{temp}_1 \leftarrow \pi_{\text{cliente\_nome}} (\sigma_{\text{gerente\_nome} = \text{"Manoel"}} (\text{usuário}))$

- relação resultado  $\text{temp}_1$

cliente_nome
Márcia
Bruno

# Solução

## ◆ Segundo passo

- realizar o produto cartesiano das relações  
$$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$$
- relação resultado  $\text{temp}_2$

# Solução

$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$

$\text{temp}_1$ . cliente_nome	cliente. cliente_nome	rua	cidade
Márcia	Márcia	Rua X	Itambé
Márcia	Rodrigo	Rua X	Maringá
Márcia	Bruno	Rua Z	Uberlândia
Bruno	Márcia	Rua X	Itambé
Bruno	Rodrigo	Rua X	Maringá
Bruno	Bruno	Rua Z	Uberlândia

# Solução

## ◆ Terceiro passo

- eliminar informações inconsistentes

$\text{temp}_3 \leftarrow \sigma_{\text{temp1.cliente\_nome} = \text{cliente.cliente\_nome}} (\text{temp}_2)$

- relação resultado  $\text{temp}_3$

$\text{temp}_1$ . cliente_nome	cliente. cliente_nome	rua	cidade
Márcia	Márcia	Rua X	Itambé
Bruno	Bruno	Rua Z	Uberlândia

# Solução

## ◆ Quarto passo

- exibir as informações solicitadas

$\pi_{\text{temp1.cliente\_nome, cidade}}(\text{temp3})$

- relação resultado

temp <sub>1</sub> . cliente_nome	cidade
Márcia	Itambé
Bruno	Uberlândia

# Exemplo 2

◆ Considere a seguinte relação

■ cliente ( cliente\_nome, rua, cidade )

cliente_nome	rua	cidade
Márcia	Rua X	Itambé
Rodrigo	Rua X	Maringá
Cristina	Rua XTZ	Maringá
Sofia	Rua X	Maringá
Ricardo	Rua AAA	Itambé

◆ Liste o nome dos clientes que moram na mesma rua e na mesma cidade que Rodrigo (exceto o próprio Rodrigo)

# Solução

## ◆ Primeiro passo

- determinar o nome da rua e o nome da cidade na qual Rodrigo mora

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

- relação resultado  $\text{temp}_1$

rua	cidade
Rua X	Maringá

# Solução

## ◆ Segundo passo

- realizar o produto cartesiano das relações

$$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$$

- relação resultado  $\text{temp}_2$

$\text{temp}_1.\text{rua}$	$\text{temp}_1.\text{cidade}$	cliente_nome	cliente.rua	cliente.cidade
Rua X	Maringá	Márcia	Rua X	Itambé
Rua X	Maringá	Rodrigo	Rua X	Maringá
Rua X	Maringá	Cristina	Rua XTZ	Maringá
Rua X	Maringá	Sofia	Rua X	Maringá
Rua X	Maringá	Ricardo	Rua AAA	Itambé



# Solução

## ◆ Terceiro passo

- eliminar informações indesejadas

$$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}}(\text{temp}_2)$$

- relação resultado  $\text{temp}_3$

$\text{temp}_1.\text{rua}$	$\text{temp}_1.\text{cidade}$	cliente_nome	cliente.rua	cliente.cidade
Rua X	Maringá	Márcia	Rua X	Itambé
Rua X	Maringá	Cristina	Rua XTZ	Maringá
Rua X	Maringá	Sofia	Rua X	Maringá
Rua X	Maringá	Ricardo	Rua AAA	Itambé

# Solução

## ◆ Quarto passo

- exibir as informações solicitadas

$\pi_{\text{cliente\_nome}} (\sigma_{\text{temp}_1.\text{rua} = \text{cliente.rua} \wedge \text{temp}_1.\text{cidade} = \text{cliente.cidade} (\text{temp}_3))$

- relação resultado

cliente_nome
Sofia

# Discussão

## ◆ Solução proposta

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\sigma_{\text{temp}_1.\text{rua} = \text{cliente.rua} \wedge \text{temp}_1.\text{cidade} = \text{cliente.cidade}} (\text{temp}_3))$

## ◆ Operação de atribuição

$\text{temp}_1(\text{rua\_rodrigo}, \text{cidade\_rodrigo}) \leftarrow$

$\pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\sigma_{\text{rua\_rodrigo} = \text{rua} \wedge \text{cidade\_rodrigo} = \text{cidade}} (\text{temp}_3))$

# Discussão

## ❖ Operação renomear (1)

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \rho_{(\text{rua\_rodrigo}, \text{cidade\_rodrigo})} (\text{temp}_1) \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\sigma_{\text{rua\_rodrigo} = \text{rua} \wedge \text{cidade\_rodrigo} = \text{cidade}} (\text{temp}_3))$

## ❖ Operação renomear (2)

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \rho_{\text{dados\_rodrigo}(\text{rua\_rodrigo}, \text{cidade\_rodrigo})} (\text{temp}_1) \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\sigma_{\text{rua\_rodrigo} = \text{rua} \wedge \text{cidade\_rodrigo} = \text{cidade}} (\text{temp}_3))$

# Junção

- ◆ Concatena tuplas relacionadas de duas relações em tuplas únicas
- ◆ Simplifica consultas que requerem produto cartesiano
  - forma um produto cartesiano dos argumentos

# Junção

relação argumento 1  $\bowtie$  condição\_junção relação argumento 2

relação



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

vendedor (cod\_vend, nome\_vend)

cod_vend	nome_vend
1	Adriana
2	Roberto

Cliente  Vendedor  
cliente.cod\_vend = vendedor.cod\_vend

nro_cli	nome_cli	end_cli	saldo	cod_vend	cod_vend	nome_vend
1	Márcia	Rua X	100,00	1	1	Adriana
2	Cristina	Avenida 1	10,00	1	1	Adriana
3	Manoel	Avenida 3	234,00	1	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: entre  
zero e o (número de  
tuplas de cliente \* número  
de tuplas de vendedor)

# Junção

## ◆ Condição de junção

- $\langle \text{condição} \rangle \wedge \langle \text{condição} \rangle \wedge \dots \wedge \langle \text{condição} \rangle$

## ◆ $A_i \theta B_j$

- $A_i$ : atributo da relação argumento 1
- $B_j$ : atributo da relação argumento 2
- $A_i$  e  $B_j$  tem o mesmo domínio
- $\theta$  (theta) é um operador de comparação  $\{=, <, \leq, >, \geq, \neq\}$
- existem diversas variações sobre a operação de junção



# Junção

## ◆ Junção theta

- na qual pode ser usada qualquer operador  $\theta$  válido no domínio dos atributos comparados
  - ◆  $\{=, <, \leq, >, \geq, \neq\}$
- os atributos envolvidos na comparação aparecem ambos na relação resultado (gerando pares de atributos), mas tuplas com valores nulos nos atributos envolvidos na junção não aparecem no resultado
- variação mais genérica

# Junção

## ◆ Junção theta

cliente (nro\_cli, nome\_cli, end\_cli, saldo, vendedor)

vendedor (cod\_vend, nome\_vend)

cliente ⋈<sub>vendedor ≠ cod\_vend</sub> vendedor

grau da relação resultante é igual a 7

# Junção

## ◆ Equi-Junção

- uso mais comum de junção
- o operador  $\theta$  é a igualdade
  - ◆  $\{=\}$
- os atributos envolvidos na comparação aparecem ambos na relação resultado (gerando pares de atributos com valores idênticos), mas tuplas com valores nulos nos atributos envolvidos na comparação não aparecem no resultado

# Junção

## ◆ Equi-Junção

cliente (nro\_cli, nome\_cli, end\_cli, saldo, vendedor)

vendedor (cod\_vend, nome\_vend)

cliente ⋈<sub>vendedor = cod\_vend</sub> vendedor

grau da relação resultante é igual a 7

# Junção

## ◆ Junção Natural – $R * S$

- semelhante à operação de Equi-Junção
- porém, dos atributos da junção, apenas os originários de uma das relações operadas aparecem na relação resultado
  - ◆ requer que os atributos comparados tenham **nomes iguais** nas duas relações
- tuplas com valores nulos nos atributos envolvidos na comparação também não aparecem no resultado

# Junção

## ◆Junção Natural

cliente (nro\_cli, nome\_cli, end\_cli, saldo, cod\_vend)

vendedor (cod\_vend, nome\_vend)

vendedor \* cliente

grau da relação resultante é igual a 6

# Junção

## ◆Junção Natural

cliente (nro\_cli, nome\_cli, end\_cli, saldo, vendedor)

vendedor (cod\_vend, nome\_vend)



vendedor \*  $\rho_{(nro\_cli, nome\_cli, end\_cli, saldo, cod\_vend)}$  cliente

grau da relação resultante é igual a 6

# Junção

R

A	B	C
1	a	x
2	b	y
3	a	y
4	c	y

S

A	D
1	d
2	d
5	e

$R \bowtie S$

A	B	C	A	D
1	a	x	1	d
2	b	y	2	d

## ◆ Interna (inner join)

- somente as tuplas de R que têm tuplas correspondentes em S, e vice-versa, aparecem no resultado



# Junção

R

A	B	C
1	a	x
2	b	y
3	a	y
4	c	y

S

A	D
1	d
2	d
5	e

$R \bowtie S$

A	B	C	A	D
1	a	x	1	d
2	b	y	2	d
3	a	y	Null	Null
4	c	y	Null	Null

## ◆ Externa à esquerda (left outer join)

- mantém cada tupla de R em  $R \bowtie S$
- preenche com valores nulos os atributos de S que não correspondem às tuplas em R

# Junção

R			S		$R \bowtie S$				
A	B	C	A	D	A	B	C	A	D
1	a	x	1	d	1	a	x	1	d
2	b	y	2	d	2	b	y	2	d
3	a	y	5	e	Null	Null	Null	5	e
4	c	y							

## ◆ Externa à direita (right outer join)

- mantém cada tupla de S em  $R \bowtie S$
- preenche com valores nulos os atributos de R que não correspondem às tuplas em S

# Junção

**R**

A	B	C
1	a	x
2	b	y
3	a	y
4	c	y

**S**

A	D
1	d
2	d
5	e

**$R \bowtie S$**

A	B	C	A	D
1	a	x	1	d
2	b	y	2	d
3	a	y	Null	Null
4	c	y	Null	Null
Null	Null	Null	5	e

## ◆ Externa completa (full outer join)

- mantém as tuplas de R e S em  $R \bowtie S$
- preenche com valores nulos os atributos que não correspondem à coluna de junção

# Exercicio 1

◆ Considere as seguintes relações

- usuário ( cliente\_nome, gerente\_nome )
- cliente ( cliente\_nome, rua, cidade )

usuário

cliente_nome	gerente_nome
Márcia	Manoel
Rodrigo	Maria

cliente

cliente_nome	rua	cidade
Márcia	Rua X	Itambé
Rodrigo	Rua X	Maringá

◆ Liste o nome de todos os usuários atendidos pelo gerente Manoel, assim como as cidades nas quais eles vivem

# Exemplo 1

- ◆ Considere as seguintes relações
  - usuário ( cliente\_nome, gerente\_nome )
  - cliente ( cliente\_nome, rua, cidade )
- ◆ Liste o nome de todos os usuários atendidos pelo gerente Manoel, assim como as cidades nas quais eles vivem

$\text{temp}_1 \leftarrow \pi_{\text{cliente\_nome}} (\sigma_{\text{gerente\_nome} = \text{"Manoel"}} (\text{usuário}))$

$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{temp1.cliente\_nome} = \text{cliente.cliente\_nome}} (\text{temp}_2)$

$\pi_{\text{temp1.cliente\_nome}, \text{cidade}} (\text{temp}_3)$

# Exemplo 1

- ◆ Considere as seguintes relações
  - usuário ( cliente\_nome, gerente\_nome )
  - cliente ( cliente\_nome, rua, cidade )
- ◆ Liste o nome de todos os usuários atendidos pelo gerente Manoel, assim como as cidades nas quais eles vivem

$\text{temp}_1 \leftarrow \pi_{\text{cliente\_nome}} (\sigma_{\text{gerente\_nome} = \text{"Manoel"}} (\text{usuário}))$

$\text{temp}_2 \leftarrow \text{temp}_1 * \text{cliente}$

$\pi_{\text{temp1.cliente\_nome}, \text{cidade}} (\text{temp2})$

# Exemplo 2

◆ Considere a seguinte relação

■ cliente ( cliente\_nome, rua, cidade )

cliente_nome	rua	cidade
Márcia	Rua X	Itambé
Rodrigo	Rua X	Maringá
Cristina	Rua XTZ	Maringá
Sofia	Rua X	Maringá
Ricardo	Rua AAA	Itambé

◆ Liste o nome dos clientes que moram na mesma rua e na mesma cidade que Rodrigo

# Exemplo 2

- ◆ Considere a seguinte relação
  - cliente ( cliente\_nome, rua, cidade )
- ◆ Liste o nome dos clientes que moram na mesma rua e na mesma cidade que Rodrigo

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \text{temp}_1 \times \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\sigma_{\text{temp}_1.\text{rua} = \text{cliente.rua} \wedge \text{temp}_1.\text{cidade} = \text{cliente.cidade}} (\text{temp}_3))$



# Exemplo 2

◆ Considere a seguinte relação

■ cliente ( cliente\_nome, rua, cidade )

◆ Liste o nome dos clientes que moram na mesma rua e na mesma cidade que Rodrigo

$\text{temp}_1 \leftarrow \pi_{\text{rua}, \text{cidade}} (\sigma_{\text{cliente\_nome} = \text{"Rodrigo"}} (\text{cliente}))$

$\text{temp}_2 \leftarrow \text{temp}_1 * \text{cliente}$

$\text{temp}_3 \leftarrow \sigma_{\text{cliente\_nome} \neq \text{"Rodrigo"}} (\text{temp}_2)$

$\pi_{\text{cliente\_nome}} (\text{temp}_3)$

# Operações sobre Conjuntos

## ◆ Operações

- união
- intersecção
- diferença

## ◆ Características

- atuam sobre relações compatíveis
- eliminam tuplas duplicadas da relação resultado

Duas relações são compatíveis quando:

- possuem o mesmo grau
- seus atributos possuem os mesmos domínios  
(os domínios dos  $i$ -ésimos atributos de cada relação são os mesmos)

# União de Conjuntos

- ◆ Une duas relações R e S compatíveis em uma relação que contém todas as tuplas pertencentes a R, a S, ou a ambas R e S

relação argumento 1  $\cup$  relação argumento 2



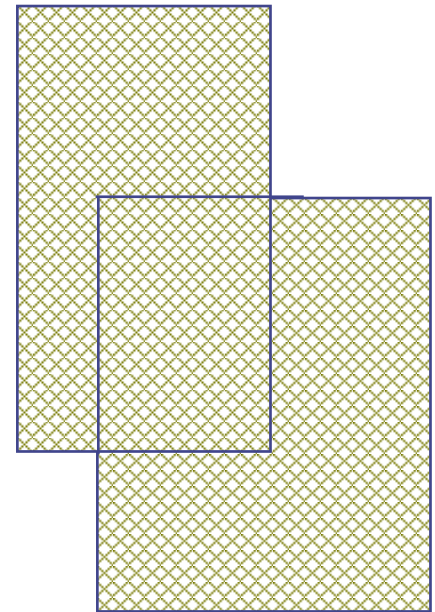
- relação
- resultado de alguma operação da álgebra relacional

# União de Conjuntos

- relações cujos domínios dos atributos são iguais, na mesma ordem de definição das colunas

**Notação:**

**$\langle \text{relação1} \rangle \cup \langle \text{relação2} \rangle$**



# Intersecção de Conjuntos

- ◆ Une duas relações R e S compatíveis em uma relação que contém todas as tuplas pertencentes tanto a R quanto a S

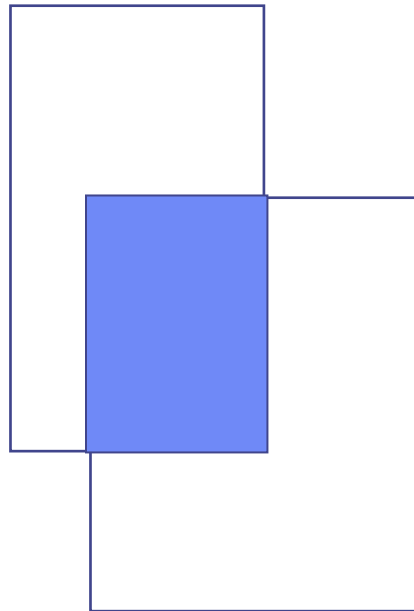
relação argumento 1  $\cap$  relação argumento 2

- relação
- resultado de alguma operação da álgebra relacional

# Intersecção de Conjuntos

Notação:

$\langle \text{relação1} \rangle \cap \langle \text{relação2} \rangle$



# Diferença de Conjuntos

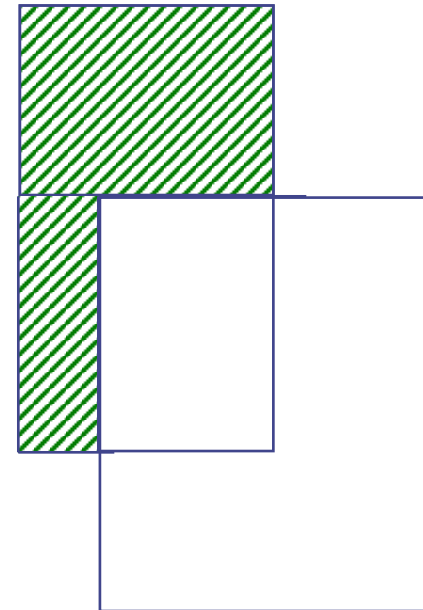
- ◆ Une duas relações R e S compatíveis em uma relação que contém todas as tuplas pertencentes a R que não pertencem a S

relação argumento 1 – relação argumento 2

- relação
- resultado de alguma operação da álgebra relacional

# Diferença de Conjuntos

Notação:  $\langle \text{relação1} \rangle - \langle \text{relação2} \rangle$



Exemplo : Listar os vendedores que não atendem nenhum cliente, ou seja, que estão na tabela Vendedor mas que não estão na tabela de “Clientes”



# Relações Cliente e Pedido

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

## vendedor (cod\_vend, nome\_vend)

cod_vend	nome_vend
1	Adriana
2	Roberto

## pedido (nro\_ped, data, nro\_cliente)

nro_ped	data	nro_cliente
1	10/12/2004	1
2	11/12/2004	4

# Consultas 5, 6 e 7

- ◆ Liste os números dos clientes que
  5. ou têm pedido, ou foram atendidos pelo vendedor 2, ou ambos
  6. têm pedido, e que foram atendidos pelo vendedor 2
  7. têm pedido, mas que não foram atendidos pelo vendedor 2

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

pedido (nro\_ped, data, nro\_cliente)

nro_ped	data	nro_cliente
1	10/12/2004	1
2	11/12/2004	4

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

pedido (nro\_ped, data, nro\_cliente)

nro_ped	data	nro_cliente
1	10/12/2004	1
2	11/12/2004	4

## Consultas 5, 6 e 7

- ◆ Liste os números dos clientes que têm pedido.

$$\text{Temp}_1 \leftarrow \pi_{\text{nro\_cliente}}(\text{pedido})$$

- ◆ Liste os números dos clientes que foram atendidos pelo vendedor 2.

$$\text{temp}_2 \leftarrow \pi_{\text{nro\_cli}}(\sigma_{\text{cod\_vend} = 2}(\text{cliente}))$$

# Consulta 5

- ◆ Liste os números dos clientes que ou têm pedido, ou foram atendidos pelo vendedor 2, ou ambos.

temp<sub>1</sub>

nro_cliente
1
4

temp<sub>2</sub>

nro_cli
4

temp<sub>1</sub>  $\cup$  temp<sub>2</sub>

nro_cliente
1
4

# Consulta 6

- ◆ Liste os números dos clientes que têm pedido, e que foram atendidos pelo vendedor 2 .

temp<sub>1</sub>

nro_cliente
1
4

temp<sub>2</sub>

nro_cli
4

temp<sub>1</sub> ∩ temp<sub>2</sub>

nro_cliente
4

# Consulta 7

- ◆ Liste os números dos clientes que têm pedido, mas que não foram atendidos pelo vendedor 2.

temp<sub>1</sub>

nro_cliente
1
4

temp<sub>2</sub>

nro_cli
4

temp<sub>1</sub> – temp<sub>2</sub>

nro_cliente
1

# Exercício

◆ Considere a seguinte relação

- conta ( nro\_conta, saldo )

<b>nro_conta</b>	<b>saldo</b>
01-010101-01	100,00
01-020202-02	200,00
01-030303-03	300,00
01-040404-04	400,00

◆ Liste o maior saldo

# Solução

## ◆ Primeiro passo

- realizar o produto cartesiano da relação conta com ela mesma

$$\text{temp}_1 \leftarrow \text{conta} \times \rho_{\text{conta2}}(\text{conta})$$

- relação resultado  $\text{temp}_1$



<b>conta.nro_conta</b>	<b>conta.saldo</b>	<b>conta2.nro_conta</b>	<b>conta2.saldo</b>
01-010101-01	100,00	01-010101-01	100,00
01-010101-01	100,00	01-020202-02	200,00
01-010101-01	100,00	01-030303-03	300,00
01-010101-01	100,00	01-040404-04	400,00
01-020202-02	200,00	01-010101-01	100,00
01-020202-02	200,00	01-020202-02	200,00
01-020202-02	200,00	01-030303-03	300,00
01-020202-02	200,00	01-040404-04	400,00
01-030303-03	300,00	01-010101-01	100,00
01-030303-03	300,00	01-020202-02	200,00
01-030303-03	300,00	01-030303-03	300,00
01-030303-03	300,00	01-040404-04	400,00
01-040404-04	400,00	01-010101-01	100,00
01-040404-04	400,00	01-020202-02	200,00
01-040404-04	400,00	01-030303-03	300,00
01-040404-04	400,00	01-040404-04	400,00

# Solução

## ◆ Segundo passo

- listar os saldos que não são os mais altos

$$\text{temp}_2 \leftarrow \pi_{\text{conta.saldo}} (\sigma_{\text{conta.saldo} < \text{conta2.saldo}} (\text{temp}_1))$$

- relação resultado  $\text{temp}_2$

<b>conta.saldo</b>
100,00
200,00
300,00

# Solução

## ◆ Terceiro passo

- listar todos os saldos da relação conta

$\text{temp}_3 \leftarrow \pi_{\text{saldo}}(\text{conta})$

- relação resultado  $\text{temp}_3$

saldo
100,00
200,00
300,00
400,00

# Solução

## ◆ Quarto passo

- fazer a diferença entre “todos os saldos da relação conta” e “os saldos que não são os mais altos”

$$\text{temp}_3 - \text{temp}_2$$

- relação resultado

saldo
400,00

# Exercício

## ◆ Considere as seguintes relações

- Aluno = (Nome, Idade, Curso)
- Professor = (Nome, Idade, Depto)
- Matricula = (NomeA, Disciplina, Nota)
- Aulas = (NomeProf, Disciplina)

1. Listar a idade e o nome dos alunos e professores do banco de dados
2. Listar as disciplinas em que os alunos de Computação se matricularam
3. Listar o departamento dos professores que não ministram a disciplina de Banco de Dados

# Exercício

1) Listar a idade e o nome dos alunos e professores do banco de dados

$Aux1 \leftarrow \pi_{\{Nome, Idade\}}(Aluno)$

$Aux2 \leftarrow \pi_{\{Nome, Idade\}}(Professor)$

$Pessoas \leftarrow Aux1 \cup Aux2$

2) Listar as disciplinas em que os alunos de computação se matricularam

$\pi_{\{Disciplina\}}(\sigma_{(curso = "computação")}(Aluno *_{\rho(Nome, Disciplina, Nota)} Matricula))$

3) Listar o departamento dos professores que não ministram a disciplina de Banco de Dados

$Aux1 \leftarrow \pi_{Nome}(\sigma_{disciplina = "Banco de Dados"}(Professor *_{\rho(Nome, Disciplina)} Aulas))$

$Aux2 \leftarrow \pi_{Nome}(Professor)$

$Departamento \leftarrow \pi_{Depto}(Aux2 - Aux1) * Professor$

# Atividades complementares

## ◆ Leitura para casa

- Capítulo 6, “Álgebra e cálculo relacional”  
do livro: Elmasri, Ramez; Navathe, Shamkant B.  
Sistemas de banco de dados, 6ª. edição (2011).



**OBRIGADO A TODOS**

**DÚVIDAS**

