

SCC-240/540/640 Bases de Dados

Prof. Robson L. F. Cordeiro

Álgebra Relacional – Parte 2

Material original editado: Profa. Elaine Parros Machado de Sousa

Exemplo

- Ex: “*Listar as disciplinas em que os alunos de computação se matricularam*”
 - Aluno = {Nome, Idade, Curso}
 - Matricula = {NomeA, Disciplina, Nota}

$\pi_{(Disciplina)}(\sigma_{(Curso = \text{"computação"})}(\sigma_{(Nome = NomeA)} (Matricula \times Aluno)))$

②

①

Exemplo

$$\pi_{\text{(Disciplina)}}(\sigma_{\text{(curso = "computação")}}(\sigma_{\text{(Nome = NomeA)}}(\text{Matricula X Aluno})))$$

- Mais eficiente \Rightarrow gravar somente as tuplas que atendem ao critério de seleção aplicado ao Produto Cartesiano
- Algoritmos eficientes para a combinação de operações
 - ex: Seleção + Produto Cartesiano

Operações Relacionais Binárias

Junção

- JUNÇÃO (“Join”) – $R \bowtie_{(\textit{condição da junção})} S$

$\pi_{(\textit{Disciplina})}(\sigma_{(\textit{Curso} = \textit{“computação”})}(\sigma_{(\textit{Nome} = \textit{NomeA})}(\textit{Matricula X Aluno})))$

$(\textit{Matricula} \bowtie \textit{Aluno})$
 $(\textit{NomeA} = \textit{Nome})$

Como representar a mesma consulta, usando junção, de maneira ainda mais eficiente?

Junção

■ **JUNÇÃO ("Join")** – $R \bowtie_{(\textit{condição da junção})} S$

$\pi_{(\textit{Disciplina})} (\textit{Matricula} \bowtie_{(\sigma_{(\textit{Curso} = \textit{"computação"})} \textit{Aluno}))} (\textit{NomeA} = \textit{Nome}))$

Em SQL

```
select Disciplina
from Aluno, Matricula
where Nome = NomeA
      and Curso = 'computacao'
```

Junção

- **JUNÇÃO ("Join")** – $R \bowtie_{(condição\ da\ junção)} S$

```
select Disciplina from Aluno, Matricula  
where Nome = NomeA  
and Curso = 'computacao'
```

Usando operador JOIN

```
select Disciplina  
from Aluno join Matricula on Nome = NomeA  
where Curso = 'computacao'
```

Junção

- **$R \bowtie_{\text{(condição da junção)}} S$**
 - condição da Junção:
 - $\langle \text{condição} \rangle \text{ AND } \langle \text{condição} \rangle \text{ AND } \dots \langle \text{condição} \rangle$
 - $\langle \text{condição} \rangle$: comparação entre atributos, ou conjunto de atributos:
 - **$\text{Atrib}_R \theta \text{Atrib}_S$**
 - Atrib_R - atributo da relação R
 - Atrib_S - atributo da relação S
 - Atrib_R e Atrib_S são **atributos de junção** - mesmo domínio
 - θ - operador de comparação válido no domínio desses atributos

Tipos de Junção

- Junções Internas (*inner joins*)
 - junção theta
 - equi-junção
 - junção natural
- Junções externas (*outer joins*)
 - *left outer join*
 - *right outer join*
 - *full outer join*

Junções Internas

- **Junção Theta (θ -join)**
 - **θ é qualquer operador** válido no domínio dos atributos de junção
 - atributos de junção das duas relações aparecem na relação resultado
 - variação mais genérica

Exemplo: Junção- θ

Aluno = {Nome, Idade, Curso} Matricula = {NomeA, Disciplina, Nota}

{<Zeca, 25, computação>,
<Zico, 18, eletrônica>,
<Juca, 21, odontologia>,
<Tuca, 18, computação> }

{<Zeca, SCC-125, 8.5>,
<Zico, SCC-148, 5.2>,
<Juca, SCC-125, 6.0>,
<Juca, SCC-148, 7.0 >}

Aluno \bowtie Matricula
(Nome = NomeA)

{**Nome**, Idade, Curso, **NomeA**, Disciplina, Nota}

{<**Zeca**, 25, computação, **Zeca**, SCC-125, 8.5>,
<**Zico**, 18, eletrônica, **Zico**, SCC-148, 5.2>,
<**Juca**, 21, odontologia, **Juca**, SCC-125, 6.0>,
<**Juca**, 21, odontologia, **Juca**, SCC-148, 7.0 >}

Junções Internas

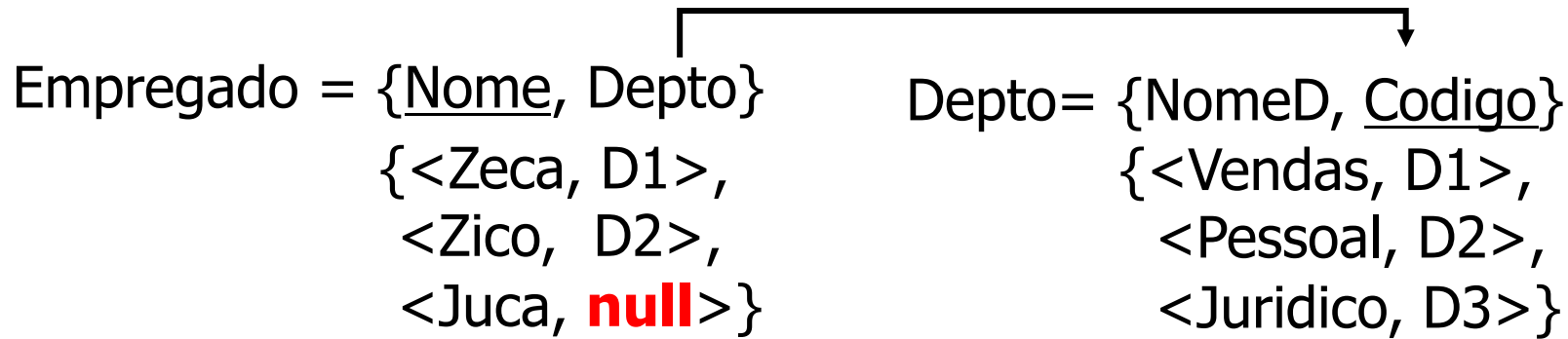
- **Equi-Junção (*Equi-join*)**
 - θ é um operador de **igualdade**
 - os atributos de junção das duas relações aparecem na relação resultado

Duas maneiras de simbolizar *equi-join*:

$$R \bowtie_{(\text{AtribR} = \text{AtribS})} S \quad \underline{\text{ou}}$$

$$R \bowtie_{(\text{AtribR}, \text{AtribS})} S$$

Exemplo: Equi-Junção



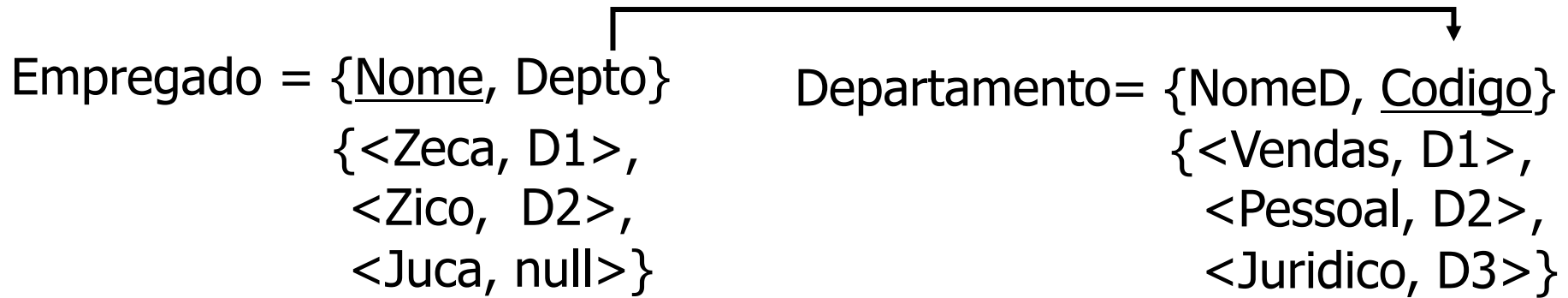
Empregado ⋈ Depto
(Depto, Codigo)

{Nome, **Depto**, NomeD, **Codigo**}
{<Zeca, **D1**, Vendas, **D1**>,
<Zico, **D2**, Pessoal, **D2**>}

Junções Internas

- **Junção Natural - $R * S$**
 - semelhante à Equi-Junção
 - apenas os atributos de junção de uma das relações aparece na relação resultado
 - requer que os atributos de junção tenham **nomes iguais** nas duas relações, ou sejam renomeados

Exemplo: Junção Natural



Empregado * $\rho_{(\text{NomeD}, \text{Depto})}$ Departamento

{Nome, **Depto**, NomeD}
{<Zeca, D1, Vendas>,
<Zico, D2, Pessoal>}

Junções Internas

- Resumindo... 3 tipos de Junção Interna (*inner joins*)
 - Junção - $\theta \rightarrow R \bowtie_{(\text{condição de junção})} S$
 - Equi-Junção $\rightarrow R \bowtie_{(\text{Atrib}_R, \text{Atrib}_S)} S$
 - Junção Natural $\rightarrow R * S$

Em SQL?

Junções Externas

- Junções externas (*outer joins*)
 - *Left Outer Join*
 - *Right Outer Join*
 - *Full Outer Join*

Junções Externas

- ***Left Outer Join*** – $R \bowtie_{\text{(condição de junção)}} S$
 - resultado:
 - tuplas que atendem à condição de junção
 - +
 - tuplas de R que não têm correspondentes em S

Exemplo: *Left Join*

Aluno = {Nome, Idade, Curso} Matricula = {NomeA, Disciplina, Nota}

{<Zeca, 25, computação>,
<Zico, 18, eletrônica>,
<Juca, 21, odontologia>,
<Tuca, 18, computação> }

{<Zeca, SCC-125, 8.5>,
<Zico, SCC-148, 5.2>,
<Juca, SCC-125, 6.0>,
<Juca, SCC-148, 7.0 >}

“Selecionar as informações de todos os alunos e, para os que estão matriculados, os códigos e notas das disciplinas que cursam.”

Aluno ⋈ Matricula
(Nome = NomeA)

{Nome, Idade, Curso, NomeA, Disciplina, Nota}

{<Zeca, 25, computação, Zeca, SCC-125, 8.5>,
<Zico, 18, eletrônica, Zico, SCC-148, 5.2>,
<Juca, 21, odontologia, Juca, SCC-125, 6.0>,
<Juca, 21, odontologia, Juca, SCC-148, 7.0 >,
<**Tuca, 18, computação, null, null, null**> }

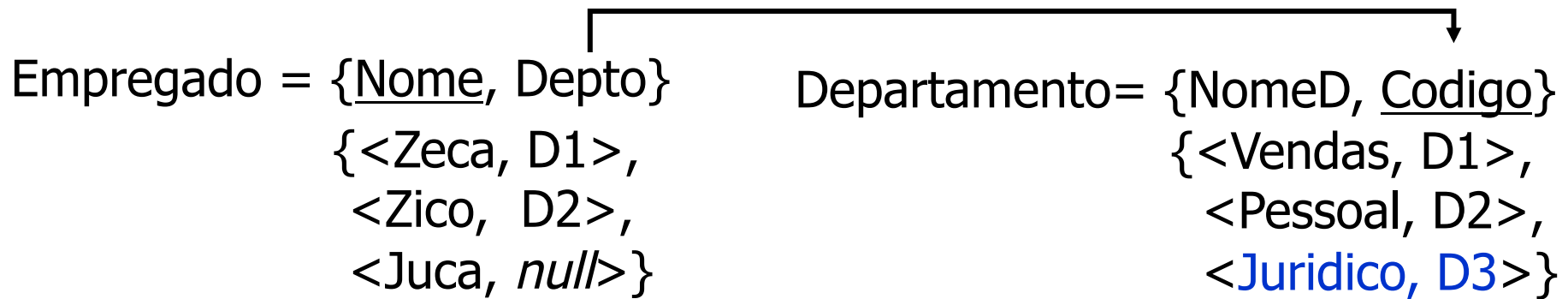
junção interna

junção externa

Junções Externas

- **Right Outer Join** – $R \boxtimes_{\text{(condição de junção)}} S$
 - resultado:
 - tuplas que atendem à condição de junção
 - +
 - tuplas de S que não têm correspondentes em R

Exemplo: *Right Join*



“Selecionar as informações de todos os departamentos e, se houver, dos empregados que trabalham neles.”

Empregado ⋈ Departamento
(Depto = Codigo)

{Nome, Depto, NomeD, Código}
{<Zeca, D1, Vendas, D1>,
<Zico, D2, Pessoal, D2>,
<**null, null, Juridico, D3**>}

Junções Externas

- ***Full Outer Join*** – $R \bowtie_{(\text{condição de junção})} S$
 - resultado:
 - tuplas que atendem à condição de junção
 - +
 - tuplas de R que não têm correspondentes em S
 - +
 - tuplas de S que não têm correspondentes em R

Exemplo: *Full Join*

Empregado = {Nome, Depto}
{<Zeca, D1>,
<Zico, D2>,
<Juca, *null*> }
Departamento = {NomeD, Codigo}
{<Vendas, D1>,
<Pessoal, D2>,
<Juridico, D3>}

Empregado \bowtie Departamento
(Depto = Codigo)

{Nome, Depto, NomeD, Código}
{<Zeca, D1, Vendas, D1>,
<Zico, D2, Pessoal, D2>,
<*Juca*, *null*, *null*, *null*>,
<*null*, *null*, *Juridico*, *D3*>}

Junções Externas em SQL

- *Left Outer Join*

```
SELECT <atributos> FROM tabela1 T1  
LEFT [OUTER] JOIN tabela2 T2 ON T1.atrib1 = T2.atrib2
```

- *Right Outer Join*

```
SELECT <atributos> FROM tabela1 T1  
RIGHT [OUTER] JOIN tabela2 T2 ON T1.atrib1 = T2.atrib2
```

- *Full Outer Join*

```
SELECT <atributos> FROM tabela1 T1  
FULL [OUTER] JOIN tabela2 T2 ON T1.atrib1 = T2.atrib2
```

Leitura recomendada

- R. Elmasri, S. Navathe: *Fundamentals of Database Systems* – 4th Edition
 - capítulo 6

Exercícios

empregado = {nomeEmpregado, CPE, rua, cidade,
telefone, idade}

trabalha = {empregado, companhia, salário}

companhia = {nome, CNPJ}

filial = { CNPJ, cidade}

gerente = {Empregado, Gerente}

Exercícios

- **Q1:** Liste nome e cidade de todos os empregados da IBM que ganham mais de dez mil dólares por mês
- **Q2:** Liste os nomes de todos os empregados que não trabalham para a IBM.
- **Q3:** Liste os nomes de todos os empregados que moram numa cidade onde há filial da companhia em que trabalham.
- **Q4:** Liste o CNPJ de todas as companhias com filiais em **todas** as cidades onde haja unidades da IBM
- **Q5:** Liste nome e CPF de todos os empregados e, para os que tiverem trabalhando, liste o CNPJ e o nome da companhia em que trabalham.