

Subconsultas ou Consultas Aninhadas

- Forma alternativa de especificar consultas envolvendo relacionamentos entre tabelas
- Otimização
 - filtragens prévias de dados na subconsulta
 - apenas tuplas/atributos de interesse são combinados com dados da(s) tabela(s) da consulta externa
- Cláusulas de subconsulta
 - *nome_atributo* [NOT] IN (*consulta_SQL*)
 - *nome_atributo* [< | <= | > | >= | < > | !=] ANY (*consulta_SQL*)
 - *nome_atributo* [< | <= | > | >= | < > | !=] ALL (*consulta_SQL*)

Subconsultas com IN

- Testam a relação de pertinência ou não-pertinência elemento-conjunto

```
select lista_atributos  
from tabela1 [...]  
where atributo_ou_expressão [NOT] IN  
      (consulta_SQL)
```

- Mapeamento para a álgebra relacional

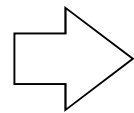
select a_1, \dots, a_n

from t_1

where $c \text{ IN}$

(**select** x **from** t_2

where $d > 5$)



$\pi_{a_1, \dots, a_n} (t_1 \theta X (\pi_x (\sigma_{d > 5} (t_2))))$

$\theta = t_1.c = t_2.x$

Exemplos

Álgebra

$\pi_{\text{nome}} ($
 $(\text{Médicos } \theta X$

$\theta = \text{Médicos.codm} = \text{Consultas.codm}$

$(\pi_{\text{codm}} (\sigma_{\text{data} = '06/11/13'}$
 $(\text{Consultas})))))$

$(\pi_{\text{CPF}} (\text{Funcionários})) \text{ — }$
 $(\pi_{\text{CPF}} (\text{Pacientes}))$

$(\pi_{\text{CPF}} (\text{Médicos})) \cap$
 $(\pi_{\text{CPF}} (\text{Pacientes}))$

SQL

```
Select nome  
From Médicos  
Where codm in  
(select codm  
  from Consultas  
  where data = '06/11/13')
```

```
Select CPF  
From Funcionários  
Where CPF not in  
(select CPF  
  from Pacientes)
```

```
Select CPF  
From Médicos  
Where CPF in  
(select CPF  
  from Pacientes)
```

Subconsultas com ANY

- Permitem outras comparações do tipo elemento-conjunto
 - testa se um valor é $>$, $<$, $=$, ... que *algum* valor em um conjunto

```
select lista_atributos
```

```
from tabela1 [, ...]
```

```
where atributo_ou_expressão [=|<|<=|>|>=|<>|!=] ANY  
(consulta_SQL)
```

- Mapeamento para a álgebra relacional

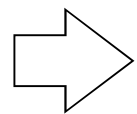
```
select a1, ..., an
```

```
from t1
```

```
where c > ANY
```

```
(select x from t2
```

```
where d > 5)
```


$$\pi_{a_1, \dots, a_n} (t_1 \theta X (\pi_x (\sigma_{d > 5} (t_2))))$$
$$\theta = t_1.c > t_2.x$$

Exemplos

Álgebra

$\pi_{\text{nome}} ($
 $(\text{Médicos } \theta X$

$\theta = \text{Médicos.codm} = \text{Consultas.codm}$

$(\pi_{\text{codm}} (\sigma_{\text{data} = '06/11/13'}$
 $(\text{Consultas})))))$

$\pi_{\text{Funcionários.idade}} ($
 $((\pi_{\text{idade}} (\text{Funcionários})) \theta X$

$\theta = \text{Funcionários.idade} < f2.idade$

$(\pi_{\text{idade}} (\rho_{f2} (\text{Funcionários})))$

SQL

```
Select nome
From Médicos
Where codm = any (ou in)
(select codm
 from Consultas
 where data = '06/11/13')
```

```
Select nome
From Funcionários
Where idade < any (
Select idade from
Funcionários)
```

Subconsultas com ALL

- Realiza uma comparação de igualdade ou desigualdade de um elemento com todos os elementos de um conjunto

```
select lista_atributos  
from tabela1 [, ...]  
where atributo_ou_expressão [=|<|<=|>|>=|<>|!=]  
      ALL(consulta_SQL)
```

- Não tem mapeamento para a álgebra relacional
 - não é equivalente a divisão
 - na divisão existe apenas comparação de igualdade
 - dividendo deve ter mais atributos que o divisor
 - não filtra automaticamente atributos do dividendo

Exemplos

```
Select nome  
From Funcionários  
Where salário > all  
    (Select salário  
     From Funcionários  
     Where departamento = 'contábil')
```

```
Select nome  
From Funcionários  
Where CPF < > all (or not in)  
    (Select CPF  
     From Pacientes)
```

Comparações Elemento-Elemento

- Casos em que a subconsulta retorna apenas **um elemento como resultado**
 - cardinalidade da subconsulta = 1
 - não é utilizada nenhuma cláusula de subconsulta neste caso

```
select lista_atributos  
from tabela1 [, ...]  
where atributo_ou_expressão [=|<|<=|>|>=|  
    <>|!=] (consulta_SQL com um único  
    elemento)
```


Exemplos

```
Select nome
From Funcionários
Where salário >
      (Select salário
       From Funcionários
       Where CPF = 22000200002)
```

```
select nome, CPF
from Médicos
where CPF < > 10000100001
and especialidade =
      (select especialidade
       from Médicos
       where CPF = 10000100001)
```

Subconsultas com EXISTS

- Quantificador existencial do cálculo relacional
 - testa se um predicado é V ou F na subconsulta
 - para cada tupla da consulta externa a ser analisada, a subconsulta é executada

```
select lista_atributos  
from tabela1 [, ...]  
where [NOT] EXISTS (consulta_SQL)
```

- Mapeamento para o cálculo relacional

select a_1, \dots, a_n

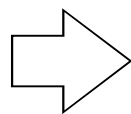
from T_1

where **EXISTS**

(select * from T_2

where $d > 5$

and $T_2.x = T_1.c$)



$\{t_1.a_1, \dots, t_1.a_n \mid t_1 \in T_1 \wedge \exists t_2 \in T_2$
 $(t_2.d > 5 \wedge t_2.x = t_1.c) \}$

Exemplos

Cálculo

$\{m.nome \mid m \in \text{Médicos} \wedge$
 $\exists c \in \text{Consultas}$
 $(c.data = '06/11/13' \wedge$
 $c.codm = m.codm)\}$

$\{f.nome \mid f \in \text{Funcionários} \wedge$
 $f.depto = \text{'pessoal'} \wedge$
 $\neg \exists p \in \text{Pacientes}$
 $(p.CPF = f.CPF)\}$

SQL

```
Select nome
From Médicos m
Where exists
  (Select *
   From Consultas
    Where data = '06/11/13'
     and codm = m.codm)
```

```
Select f.nome
From Funcionários f
Where f.depto = 'pessoal'
and not exists
  (Select *
   From Pacientes
    Where CPF = f.CPF)
```