

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE  
SÃO PAULO**

**DANIELA OLIVEIRA COSTA**

**CÁLCULO RELACIONAL**

**CAMPOS DO JORDÃO  
2025**

## INTRODUÇÃO

Este trabalho tem como objetivo explorar e explicar os conceitos fundamentais do cálculo relacional, uma das linguagens essenciais para consultas em bancos de dados relacionais. O cálculo relacional, desenvolvido por Edgar F. Codd na década de 1970, desempenha um papel crucial no modelo relacional, permitindo a descrição de consultas de forma declarativa, ou seja, sem a necessidade de especificar os métodos exatos para a execução das operações. O estudo será dividido em duas abordagens principais: o Cálculo Relacional de Tupla (TRC) e o Cálculo Relacional de Domínio (DRC). Ambas as abordagens têm como base a lógica matemática e a lógica de predicados de primeira ordem. Através desta análise, o trabalho visa detalhar as diferenças entre essas duas formas de cálculo e ilustrar como elas são aplicadas na prática para obter resultados específicos a partir das tabelas de um banco de dados.

## 1. CÁLCULO RELACIONAL

### 1.1 Definição

A linguagem foi criada por E. F. Codd no início da década de 1970, durante o desenvolvimento do modelo relacional de bancos de dados. A intenção de Codd era usar princípios da matemática para organizar e acessar dados de forma mais clara e confiável. Ele queria uma maneira de trabalhar com os dados que fosse mais lógica e menos dependente de como eles estavam armazenados fisicamente nos computadores.

O cálculo relacional, juntamente com a álgebra relacional, fazem parte do conjunto de linguagens fundamentais do modelo relacional de dados. Ambas as linguagens foram criadas antes da linguagem SQL, a qual se consolidou como padrão para sistemas gerenciadores de banco de dados relacionais (SGBDR).

O cálculo relacional é uma linguagem declarativa, ou seja, seu foco está na descrição do que se deseja obter como resultado de uma escolha, e não na forma como essa consulta é realizada.

Essa linguagem é importante por dois motivos principais. O primeiro é que ela se baseia na lógica matemática, mas especificamente na lógica de predicados de primeira ordem. O segundo é que a linguagem de consulta padrão (SQL) para SGBDRs tem alguns de seus alicerces no cálculo relacional de tupla.

O cálculo relacional apresenta duas variantes principais:

- Cálculo Relacional de Tuplas (TRC – Tuple Relational Calculus)
- Cálculo Relacional de Domínio (DRC – Domain Relational Calculus)

No cálculo relacional de tuplas, as variáveis manipuladas representam tuplas completas, ou seja, linhas inteiras das tabelas. Já no cálculo relacional de domínio, as variáveis representam valores individuais de atributos, isto é, valores das colunas de uma ou mais relações.

Ambas as formas representam as mesmas consultas, embora adotem diferentes formas sintáticas.

## 2. CÁLCULO RELACIONAL DE TUPLA

### 2.1 O que são

O cálculo relacional de tuplas é uma das formas de linguagem declarativa utilizada para realizar consultas no modelo relacional. Se baseia na utilização de variáveis que representam tuplas inteiras, ou seja, linhas de uma tabela. Cada variável de tupla percorre uma relação (tabela) específica no banco de dados e pode assumir os valores das tuplas dessa relação. A linguagem não define como as tuplas são encontradas, apenas quais tuplas devem aparecer no resultado com base nos critérios estabelecidos.

Representação do cálculo relacional de tupla:

$$\{t \mid \text{COND}(t)\}$$

Onde  $t$  é uma variável de tupla e  $\text{COND}(t)$  é uma condição que deve ser verdadeira para que a tupla  $t$  seja incluída no resultado.

#### Exemplo 1:

Retornar os funcionários com idade superior a 25 anos.

$$\{t \mid \text{FUNCIONARIO}(t) \wedge t.\text{idade} > 25\}$$

Nesse exemplo  $t$  é a variável de tupla,  $\text{FUNCIONARIO}(t)$  indica que a variável  $t$  percorre a relação  $\text{FUNCIONARIO}$  e  $t.\text{idade} > 25$  é a condição lógica que deve ser atendida. Será retornado todas as tuplas da relação  $\text{FUNCIONARIO}$  nas quais o valor do atributo idade forem superior a 25.

#### Exemplo 2:

Recuperar a data de nascimento e o endereço de um funcionário específico.

$$\{t.\text{data\_nasc}, t.\text{endereco} \mid \text{FUNCIONARIO}(t) \wedge t.\text{Primeiro\_nome} = \text{'Pedro'} \wedge t.\text{Ultimo\_nome} = \text{'Costa'}\}$$

A consulta retornará dois atributos: *data\_nasc* e *endereco* e as condições serão da relação *FUNCIONARIO* cujo primeiro nome é “Pedro” e o último nome “Costa”.

## 2.2 Fórmulas no Cálculo Relacional de Tupla

As consultas são definidas com fórmulas, que descrevem as condições que as tuplas devem cumprir para gerar o resultado. Essas fórmulas são compostas por átomos de cálculo predicado, conectados por operadores lógicos. A consulta retorna os atributos das tuplas que atendem a essas condições.

Representação:

$$\{(t1, A1, t2, A2 \mid COND(t1, t2...))\}$$

Onde *t1*, *t2*, são variáveis de tupla e *A1*, *A2* são atributos. *COND(...)* representa a fórmula lógica.

Átomos são as unidade mínimas de informação lógica e os principais tipos usados nas expressões são:

**\* Pertinência (Associação a uma Relação):** Indica que uma variável de tupla está associada a uma relação (tabela) do banco de dados:

$$t \in R$$

Esse átomo afirma que a variável de tupla *t* representa uma tupla (linha) válida da relação *R*.

**Exemplo:**

$$f \in FUNCIONARIO$$

A variável  $t$  representa uma tupla (linha) da tabela *FUNCIONARIO*. Esse átomo diz que  $t$  deve pertencer à relação *FUNCIONARIO*, ela percorre cada linha dessa tabela.

**\* Comparação de Atributos:** Compara valores de atributos entre si ou com constantes:

Comparação com valores constantes:

**$t.atributo > constante$**

Comparação entre atributos de tuplas distintas:

**$t1.atributo1 = t2.atributo2$**

Essas comparações permitem restringir o conjunto de tuplas com base nos valores armazenados nos atributos.

**Exemplo:**

**$t.idade > 30$**

Esse átomo compara o valor do atributo idade da tupla  $t$  com a constante 30. A condição será verdadeira para todas as tuplas em que o valor de idade seja maior que 30.

No cálculo relacional, os predicados relacionais (como  $=$ ,  $\neq$ ,  $<$ ,  $>$ ,  $\leq$ ,  $\geq$ ) são usados para comparar valores entre atributos ou com constantes. Já os conectivos lógicos (AND, OR, NOT) permitem combinar essas comparações, tornando as condições das consultas mais completas e precisas.

## **2.3 Quantificadores existenciais e universais**

Através dos quantificadores existenciais e universais é possível construir consultas mais complexas.

### 2.3.1 Quantificador existencial

É representado como  $(\exists)$  e pode ser lida como 'existe ao menos uma'.

Sintaxe:

$$\{ t.a \mid R(t) \wedge \exists s (S(s) \wedge \text{condição}(t, s)) \}$$

- $t$  e  $s$  são variáveis de tupla
- $t.a$  é o atributo que será retornado da tupla  $t$
- $R(t)$  e  $S(s)$  são predicados relacionais
- $\exists s (...)$  indica que existe uma tupla  $s$  em  $S$  que satisfaz a condição
- $\text{condição}(t, s)$  é uma condição lógica

**Exemplo:**

$$\{ p.nome \mid PROFESSOR(p) \wedge \exists d (DISCIPLINA(d) \wedge d.id\_prof = p.id\_prof) \}$$

A variável  $p$  representa uma tupla da relação *PROFESSOR*. A variável  $d$  representa uma tupla da relação *DISCIPLINA*.  $\exists d(...)$  significa que existe pelo menos uma disciplina  $d$  tal que  $d.id\_prof = p.id\_prof$ . O  $p.nome$  é o atributo que retornará o resultado.

### 2.3.2 Quantificador universal

Há uma classe de consultas que podem ser resolvidas de forma mais simples através do quantificar universal. Lido como 'para todo' ou 'para cada' e expressado como  $(\forall)$ .

Sintaxe:

$$\{ t.a \mid R(t) \wedge \forall s (S(s) \rightarrow \text{condição}(t,s)) \}$$

- $t.a$  é atributo que será retornado
- $R(t)$  a tupla  $t$  pertence à relação  $R$
- $\forall s (...)$  para toda tupla  $s$  da relação  $S$

$\neg S(s) / \text{condição}(t, s)$  se  $s$  pertence a  $S$ , então a condição entre  $t$  e  $s$  deve ser verdadeira

**Exemplo:**

$\{ p.\text{nome} \mid \text{PROFESSOR}(p) \wedge \forall d (\text{DISCIPLINA}(d) \rightarrow d.\text{id\_prof} = p.\text{id\_prof}) \}$

Variável  $p$  para uma tupla da relação *PROFESSOR*. Variável  $d$  para uma tupla da relação *DISCIPLINA*. *PROFESSOR*( $p$ ):  $p$  é uma tupla válida da relação *PROFESSOR*.  $\forall d (...)$ : para toda tupla  $d$  da relação *DISCIPLINA*, a condição deve ser verdadeira. *DISCIPLINA*( $d$ )  $\rightarrow d.\text{id\_prof} = p.\text{id\_prof}$ : se  $d$  é uma disciplina, então seu professor ( $d.\text{id\_prof}$ ) deve ser o mesmo que  $p.\text{id\_prof}$ .

### 2.3.3 Transformação de um quantificador

É possível transformar um quantificador universal e existencial e vice-versa para obter uma expressão equivalente. Pode ser escrita através da combinação do quantificador existencial com uma negação.

### 2.3.4 Tipos de variáveis

No cálculo relacional de tuplas, as variáveis podem ser livres ou limitadas, de acordo com sua posição em relação aos quantificadores lógicos. Uma variável de tupla livre é aquela que não está dentro de um quantificador (como  $\exists$  ou  $\forall$ ). Ela representa os valores que serão retornados pela consulta. Por outro lado, uma variável limitada aparece dentro do escopo de um quantificador e não faz parte do resultado, sendo usada apenas para verificar alguma condição lógica. Para garantir que a expressão seja segura, é necessário que toda variável livre esteja limitada ao universo de uma relação.

## 3. CÁLCULO RELACIONAL DE DOMÍNIO

### 3.1 O que são



As variáveis percorrem valores isolados dos domínios de atributos. Para formar uma relação para um resultado de pesquisa é necessário ter  $n$  variáveis de domínio, uma para cada atributo.

Representação do cálculo relacional de domínio:

$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid \text{COND}(x_1, x_2, \dots, x_n) \}$$

Onde  $x_1, x_2$  são variáveis de domínio que percorrem os atributos e  $\text{COND}$  é a condição ou fórmula do cálculo relacional.

#### Exemplo 1:

Retornar os funcionários com idade superior a 25 anos.

$$\{ \langle id, nome \rangle \mid \exists i (\text{FUNCIONARIO}(id, nome, i) \wedge i > 25) \}$$

Neste exemplo será retornado o  $id$  e o  $nome$  dos funcionários. É usado a variável  $i$  para representar a idade dos funcionários. A condição parte da relação  $\text{FUNCIONARIO}$ , indica que os valores de  $id$ ,  $nome$  e  $i$  pertencem a essa relação. Em seguida, é aplicado o filtro  $i > 25$ .

#### Exemplo 2:

Recuperar data de nascimento e endereço de um funcionário com nome específico.

$$\{ \langle d, e \rangle \mid \exists i \exists pn \exists un (\text{FUNCIONARIO}(i, pn, un, d, e) \wedge pn = \text{'Pedro'} \wedge un = \text{'Costa'}) \}$$

A consulta retorna  $\langle d, e \rangle$ , ou seja, a data de nascimento ( $d$ ) e o endereço ( $e$ ) do funcionário. As variáveis  $i$  ( $ID$ ),  $pn$  ( $\text{Primeiro Nome}$ ) e  $un$  ( $\text{Último Nome}$ ) são usadas para identificar o funcionário na relação  $\text{FUNCIONARIO}(i, pn, un, d, e)$ . A condição  $pn = \text{'Pedro'} \wedge un = \text{'Costa'}$  filtra apenas o funcionário com esse nome.

### 3.2 Fórmulas no Cálculo Relacional de Domínio

A expressão que pertence ao cálculo relacional de domínio é:

$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid P(x_1, x_2, \dots, x_n) \}$$

$\langle x_1, x_2, \dots, x_n \rangle$  são variáveis que representam valores individuais de atributos. A parte  $P(\dots)$  é uma fórmula lógica que pode conter predicados relacionais (como  $R(x, y)$ ), condições (ex:  $x > 30$ ), conectivos lógicos ( $\wedge$ ,  $\vee$ ,  $\neg$ ) e quantificadores ( $\exists$ ,  $\forall$ ) usados para limitar ou relacionar os valores.

As fórmulas são ligeiramente diferentes do calculo de tupla:

**\*Pertinência (Associação a uma Relação):** Indica que uma variável de tupla está associada a uma relação (tabela) do banco de dados:

$$R(x_1, x_2, \dots, x_n)$$

Esse átomo indica que os valores  $x_1, x_2, \dots, x_n$  pertencem à relação (tabela)  $R$

**\* Comparação de Atributos:** Compara valores de atributos entre si ou com constantes:

Comparação com valores constante:

$$\{ \langle x_1, x_2 \rangle \mid \exists x_3 (R(x_1, x_2, x_3) \wedge x_3 > 30) \}$$

Será retornado as variáveis  $x_1$  e  $x_2$  de uma relação  $R$ . A variável  $x_3$  é associada à relação  $R$  e deve satisfazer a condição  $x_3 > 30$ , ou seja,  $x_3$  é maior que 30.

**\*Booleanos:** São usados como base lógica para criar expressões mais complexas, como:

- True ou false
- Negação ( $\neg$ )
- Conjunção ( $\wedge$ )
- Disjunção ( $\vee$ )

## CONCLUSÃO

O cálculo relacional, tanto na versão de tuplas quanto na de domínio, é essencial para a compreensão das consultas em bancos de dados relacionais. Essas duas formas de cálculo, apesar de apresentarem sintaxes diferentes, cumprem a mesma função de expressar de maneira declarativa as condições para a recuperação de dados. Através do uso de quantificadores e operadores lógicos, é possível realizar consultas complexas e precisas. O conhecimento do cálculo relacional fornece uma base sólida para o entendimento do funcionamento interno das consultas SQL, e contribui para o desenvolvimento de sistemas de banco de dados mais eficientes e eficazes. Ao estudar essas técnicas, é possível aprofundar o entendimento do modelo relacional e das linguagens que o sustentam, tornando-se uma ferramenta essencial para profissionais da área de banco de dados.

## REFERÊNCIAS

HEUSER, Carlos Alberto. **Banco de Dados Relacional: Conceitos , SQL e Administração** - 1ª edição. Porto Alegre: O Autor, 2019.

ELMASRI, Ramez; NAVATHE, Shamkant B.. **Sistemas de Banco de Dados** – 7ª edição. São Paulo: Pearson Education do Brasil, 2018.

CODD, Edgar F. A Relational Model of Data for Large Shared Data Banks. **Communications of the ACM**, v.13, n.6, p.377-387, jun. 1970.

DURANT, Kathleen. **Relational Algebra & Relational Calculus**. Disponível em: <https://www.khoury.northeastern.edu/home/kathleen/classes/cs3200/4-RAAndRC.pdf>. Acesso em: 12 abr 2025.

ROCHA, Ricardo. **Bases de Dados: Parte IV: Álgebra e Cálculo Relacional**. Disponível em: <https://www.dcc.fc.up.pt/~ricroc/aulas/0910/bd/apontamentos/partelV.pdf>. Acesso em 13 abr 2025.