Capítulo 7-? Cálculo Relacional

O Cálculo Relacional (CR) é uma linguagem de consulta formal. Utilizando-se de uma expressão declarativa pode-se especificar uma consulta.

Uma expressão de cálculo permite a descrição da consulta desejada sem especificar os procedimentos para obtenção dessas informações, ou seja, é não-procedural. Contudo, tal consulta deve ser capaz de descrever formalmente a informação desejada, com exatidão.

Existem dois tipos: Cálculo Relacional de Tuplas (CRT) e Cálculo Relacional de Domínio (CRD). Eles são subconjuntos simples de lógica de primeira ordem.

No Cálculo Relacional existem variáveis, constantes, operadores lógicos, de comparação e quantificadores. As expressões de Cálculo são chamadas de fórmulas. Uma tupla de respostas é essencialmente uma atribuição de constantes às variáveis que levam a fórmula a um estado verdadeiro.

Em CRT, as variáveis são definidas sobre tuplas. Já em CRD, variáveis são definidas sobre o domínio dos elementos (ou seja, sobre os valores dos campos).

Todas as expressões de consulta descritas em CR possuem equivalentes em Álgebra Relacional.

Cálculo Relacional de Tuplas

É baseado na especificação de um número de variáveis de tuplas. Cada variável tupla pode assumir como seu valor qualquer tupla da relação especificada.

Uma consulta em CRT é especificada da seguinte forma:

{variável tupla | predicado}

O resultado de tal consulta é o conjunto de todas as variáveis tuplas para as quais o predicado é indicado como verdadeiro.

Uma expressão genérica do cálculo relacional de tuplas tem a forma$\{t_1.A_1, t_2.A_2,..., t_n.A_n \mid predicado(t_1, t_2,..., t_n, t_{n+1}, t_{n+2}, ...,t_{n+m})\}$

onde $t_1, t_2,..., t_n, t_{n+1}, t_{n+2}, ...,t_{n+m}$ são variáveis de tuplas, cada A_i é um atributo da relação na qual t_i se encontra . O predicado é uma fórmula do cálculo relacional de tuplas.

As fórmulas atômicas de cálculo de predicados podem ser uma das seguintes:

1-) Uma fórmula atômica R(ti), onde R é o nome de uma relação e ti é uma variável de tupla. Este átomo identifica a extensão da variável de tupla ti como a relação cujo nome seja R.

2-) Uma fórmula atômica ti.A op tj.B, onde op é um dos operadores de comparação no conjunto {=, >, <, ...}, ti e tj são variáveis de tuplas, A é um atributo da relação na qual ti se encontra, B é um atributo da relação na qual tj se encontra.

3-) Um fórmula atômica ti.A op c ou c op tj.B, onde op é um dos operadores de comparação no conjunto {=, >, <, ...}, ti e tj são variáveis de tuplas, A é um atributo da relação na qual ti se encontra, B é um atributo da relação na qual tj se encontra e c é um valor constante.

Cada uma das fórmulas atômicas anteriormente especificadas tem seu valor verdade avaliado como TRUE ou FALSE para uma combinação específica de tuplas. Para fórmulas do tipo 1, caso a variável de tupla seja atribuída a uma tupla da relação R dada, esta assume valor TRUE; caso contrário, FALSE. Já nas fórmulas do tipo 2 e 3, se as variáveis de tupla forem designadas de forma que os valores dos atributos especificados satisfaçam o predicado, esta assumirá valor verdade TRUE.

Todas as variáveis tuplas abordadas até então são consideradas variáveis livres[1], uma vez que estas não aparecem quantificadas. Contudo, além das definições acima, quantificadores (universal ($\forall$) ou existencial ($\exists$)) podem aparecer nas fórmulas. Neste caso, as variáveis que os sucedem são denominadas variáveis limite.

Uma fórmula é definida, de forma recursiva, por uma ou mais fórmulas atômicas, conectadas por operadores lógicos (AND, OR, NOT), como segue:

1-) Qualquer fórmula atômica.

2-) Se F1 e F2 são fórmulas atômicas, então (F1 AND F2), (F1 OR F2), NOT (F1) e NOT (F2) também o são, tendo seus valores verdade derivados a partir de F1 e F2, da seguinte forma:

a-) (F1 AND F2) será TRUE apenas se ambos, F1 e F2, forem TRUE;

b-) (F1 OR F2) será TRUE quando uma das duas fórmulas F1 e F2, for TRUE;

c-) NOT(F1) será TRUE quando F1 for FALSE;

c-) NOT(F2) será TRUE quando F2 for FALSE.

3-) Se F1 é uma fórmula atômica, então ($\exists$ t[2])(F1) também o é, e seu valor verdade apenas será TRUE se a fórmula F for avaliada como verdadeira para pelo menos uma tupla atribuída para ocorrências livres de t em F.

4-) Se F1 é uma fórmula atômica, então ($\forall$ t)(F1) também o é, e seu valor verdade apenas será TRUE se a fórmula F for avaliada como verdadeira para todas as tuplas atribuídas para ocorrências livres de t em F.

É possível escrever expressões equivalentes manipulando operadores e quantificadores. Alguns casos dessa manipulação podem ser declarados da seguinte forma:

1-) F1 AND F2 $\equiv$ NOT(NOT F1 OR NOT F2).
2-) ($\forall$ t) $\in$ r (F1(t)) $\equiv$ NOT ($\exists$ t) $\in$ r (NOT F1(t)).
3-) F1 $\Rightarrow$ F2 $\equiv$ NOT F1 OR F2.
4-) ($\forall$ x) (F(x)) $\equiv$ NOT ($\exists$ x) (NOT (F(x))
5-) ($\exists$ x) (F(x)) $\equiv$ NOT ($\forall$ x) (NOT (F(x))

---

[1] As únicas variáveis de tupla livres em uma expressão de cálculo relacional devem ser aquelas à esquerda da barra (|).

[2] t é uma variável de tupla.

6-) $(\forall x) (F(x) \text{ AND } P(x)) \equiv \text{NOT } (\exists x) (\text{NOT } (F(x)) \text{ OR NOT } (P(x)))$
7-) $(\forall x) (F(x) \text{ OR } P(x)) \equiv \text{NOT } (\exists x) (\text{NOT } (F(x)) \text{ AND NOT } (P(x)))$
8-) $(\exists x) (F(x) \text{ OR } P(x)) \equiv \text{NOT } (\forall x) (\text{NOT } (F(x)) \text{ AND NOT } (P(x)))$
9-) $(\exists x) (F(x) \text{ AND } P(x)) \equiv \text{NOT } (\forall x) (\text{NOT } (F(x)) \text{ OR NOT } (P(x)))$

Abaixo seguem exemplos de consultas em CRT.

1-) Encontre todos os empregados cujos salários estejam acima de R$3.500,00.

{t | EMPREGADO(t) AND t.SALARIO > 3500}

2-) Dê apenas os nomes e sobrenomes dos empregados cujos salários estejam acima de R$3.500,00.

{t.NOME, t.SOBRENOME | EMPREGADO(t) AND t.SALARIO > 3500}

3-) Selecione o nome e o endereço dos empregados que trabalham para o departamento de 'Informática'.

{t.NOME, t.SOBRENOME, t.ENDERECO | EMPREGADO(t) AND (∃ D) (DEPARTAMENTO (d) AND d.NOMED = 'Informática' AND d.NUMERODEP = t.NUD)}

4-) Para cada projeto localizado em 'São Paulo', liste o número do mesmo, o nome do departamento proponente, bem como sobrenome, data de nascimento e endereço do gerente responsável.

{p.NUMEROP, p.NUMD, m.SOBRENOME, m.DATANASCIMENTO, m.ENDERECO | PROJETO(p) AND EMPREGADO(m) AND p.LOCALIZACAO = 'São Paulo' AND ((∃ d) (DEPARTAMENTO(d) AND p.NUMD = d.NUMERODEP AND d.NSSGER = m.NSS))}

5-) Encontre os nomes dos empregados que trabalham em todos os projetos controlados pelo departamento de número 5.

{e.SOBRENOME, e.NOME | EMPREGADO(e) AND ((∀ x) (NOT(PROJETO(x)) OR NOT(x.NUMD = 5) OR ((∃ w) (TRABALHA_EM(w) AND w.NSSE = e.NUMEROP AND x.NUMEROP = w.NUMP))))}

Expressões Seguras

Uma expressão em CRT pode gerar uma infinidade de relações. Para a expressão

{t | NOT (R(t))}

pode existir uma infinidade de tuplas que não estão em R, de forma que esta é não-segura. A maioria dessas tuplas contém valores que não estão no banco de dados, logo, não são desejáveis como resultados.

Uma expressão segura no CR é uma expressão que garante a produção de um número finito de tuplas como resultado.

Para melhor definir expressão segura, o conceito de domínio pode ser utilizado. O domínio de uma expressão P é o conjunto de todos os valores referenciados por P. Isso inclui os valores mencionados em P propriamente dito, assim como os valores que aparecem na tupla de uma relação referenciada por P. Assim, o domínio de P é um conjunto de valores que aparecem explicitamente em P ou que aparecem em uma ou mais relações cujos nomes aparecem em P.

Cálculo Relacional de Domínio

A diferença básica entre CRT e CRD é que neste último as variáveis estendem-se sobre valores únicos de domínios de atributos. Para formar uma relação de grau n para um resultado de consulta, faz-se necessário criar n variáveis de domínio, uma para cada atributo.

Uma expressão genérica do cálculo relacional de tuplas tem a forma

$\{x_1, x_2, ..., x_n \mid predicado(x_1, x_2, ..., x_n, x_{n+1}, x_{n+2}, ..., x_{n+m})\}$

onde $x_1, x_2, ..., x_n, x_{n+1}, x_{n+2}, ..., x_{n+m}$ são variáveis de domínio aplicadas sobre o domínio dos atributos requeridos na consulta e predicado é uma fórmula atômica do CRD, que pode ser especificada em uma das formas que segue:

1-) Uma fórmula atômica $R(x_1, x_2, ..., x_n)$, onde R é o nome de uma relação de grau j e cada x_1, $1 \leq i \leq j$, é uma variável de domínio. Isto implica que uma lista de valores de $<x_1, x_2, ..., x_j>$ deve ser uma tupla na relação R, onde x_i é o valor do i-ésimo valor de atributo da tupla.

2-) Uma fórmula atômica xi op xj, onde op é um operador de comparação {=, <, >, ...} e xi e xj são variáveis de domínio.

3-) Uma fórmula atômica xi op c ou c xj, onde op é um operador de comparação {=, <, >, ...} e xi e xj são variáveis de domínio e c é um valor constante qualquer.

Como em CRT, as fórmulas são avaliadas em valores verdade para um conjunto específico de valores. Para a fórmula do tipo 1, o valor verdade será TRUE apenas se houver valores de domínio correspondentes a uma tupla de R atribuídos às variáveis de domínio que representam. Para os casos 2 e 3, o valor verdade será TRUE caso as variáveis de domínio possuam valores que satisfaçam o predicado.

Abaixo, para fins de comparação, seguem em CRD os mesmos exemplos de consultas já escritos em CRT.

1-) Encontre todos os empregados cujos salários estejam acima de R$3.500,00.

{qrstuvwxyz | (∃ x) EMPREGADO(qrstuvwxyz) AND x > 3500}

2-) Dê apenas os nomes e sobrenomes dos empregados cujos salários estejam acima de R$3.500,00.

{qs | (∃ x) EMPREGADO(qrstuvwxyz) AND x > 3500}

3-) Selecione o nome e o endereço dos empregados que trabalham para o departamento de 'Informática'.

{qsv | (∃ z) (∃ l) (∃ m) (EMPREGADO(qrstuvwxyz) AND DEPARTAMENTO(lmno) AND l = 'Informática' AND m = z)}

4-) Para cada projeto localizado em 'São Paulo', liste o número do mesmo, o nome do departamento proponente, bem como sobrenome, data de nascimento e endereço do gerente responsável.

{iksuv | (∃ j) (∃ m) (∃ n) (∃ t) (PROJETO(hijk) AND EMPREGADO(qrstuvwxyz) AND DEPARTAMENTO(lmno) AND k = m AND n = t AND j = 'São Paulo')}

5- exercício) Encontre os nomes dos empregados que trabalham em todos os projetos controlados pelo departamento de número 5.

Expressões Seguras

Uma expressão em CRD é dita segura se:

1-) Todos os valores que aparecem nas tuplas da expressão são valores dentro do domínio da mesma.

2-) Para todas as subfórmulas $(\exists\ x)\ (P(x))$, a subfórmula é verdadeira se, e somente se, existir um valor x no domínio de P tal que P(x) seja verdadeiro.

3-) Para toda subfórmula $(\forall\ x)\ (P(x))$, a subfórmula é verdadeira se, e somente se, P(x) for verdadeiro para todos os valores de x dentro do domínio de P.

As proposições acima garantem que possamos testar todas as subfórmulas "existe um" e "para todo" sem a necessidade de testar todas as suas infinitas possibilidades de ocorrência.