Projeto de Bases de Dados, 4º Entrega

2017/2018



**Grupo 1**

**Membros:**

* **84718** – Francisco Aguiar, 33.33% - 9 Horas
* **84719** – Gonçalo Marques, 33.33% - 9 Horas
* **84740** – Manuel Sousa, 33.33% - 9 Horas

**Professor:** Tomás Almeida e Silva Martins Alves

**Turno:** Quinta-Feira, 8:30 – 10:00 (BD81795L08)

**Restrições de Integridade**

**a)** O fornecedor (primário) de um produto não pode existir na relação fornece\_sec para o mesmo produto.

**CREATE** **OR** **REPLACE** **FUNCTION** PrimaryExistsOnSecondary(nifIN **VARCHAR**(9),eanIN **VARCHAR**(25))

**RETURNS** **BOOLEAN** **AS** $$

**DECLARE** ans **BOOLEAN**;

**BEGIN**

**SELECT** **TRUE** **INTO** ans

**FROM** Fornecedor\_secundario **AS** F

**WHERE** nif = nifIN **AND** ean = eanIN;

**IF** ans **IS** **NULL** **THEN** ans := **FALSE**; **END** **IF**;

**RETURN** ans;

**END**; $$ **LANGUAGE** plpgsql;

**CREATE** **OR** **REPLACE** **FUNCTION** SecondaryExistsOnPrimary(nifIN **VARCHAR**(9),eanIN **VARCHAR**(25))

**RETURNS** **BOOLEAN** **AS** $$

**DECLARE** ans **BOOLEAN**;

**BEGIN**

**SELECT** **TRUE** **INTO** ans

**FROM** Produto **AS** F

**WHERE** forn\_primario = nifIN **AND** ean = eanIN;

**IF** ans **IS** **NULL** **THEN** ans := **FALSE**; **END** **IF**;

**RETURN** ans;

**END**; $$ **LANGUAGE** plpgsql;

**ALTER** **TABLE** Produto

**ADD** **CONSTRAINT** cantExist **CHECK**(PrimaryExistsOnSecondary(forn\_primario,ean) != **TRUE**);

**ALTER** **TABLE** Fornecedor\_secundario

**ADD** **CONSTRAINT** cantExist **CHECK**(SecondaryExistsOnPrimary(nif,ean) != **TRUE**);

**b)** O instante mais recente de reposição tem de ser sempre anterior ou igual à data atual.

**ALTER** **TABLE** EventoReposicao

**ADD** **CONSTRAINT** RI\_EA3 **CHECK**(instante <= **CURRENT\_TIMESTAMP**);

**Índices**

1. Indique, justificando, que tipo de índice(s), sobre que atributo(s) e sobre que tabela(s) faria sentido criar de modo a acelerar a execução destas interrogações. Analise cada uma das consultas separadamente.

Após analise da query 1, concluímos que:

* *B-Tree index* primário(clustered) em Fornecedor.nif
* *B-Tree index* primário(clustered) em Produto.fornec\_primario

Este conjunto de índices primários vai fazer com que o join das duas tabelas seja O(N\_fornecedores) o que é bastante mais eficiente que O(N\_produtos \* N\_fornecedores) que seria a complexidade do join se não tivesse estes índices.

Secundary Hashed index na categoria = 'fruta':

Ao ter os produtos indexados por categoria , seria muito mais eficiente encontrar os produtos da categoria “fruta”, supondo que para fazer a procura de um produto da categoria

fruta teríamos de percorrer a tabela toda então a complexidade seria

O(N\_prods) o que é extremamente lento quando existem muitos

produtos, mas tendo todos os produtos indexados por categoria teríamos apenas uma procura na ordem dos O(1).

B-Tree primary index em Fornecedor\_secundario.ean

B-Tree primary index em Produto.ean

Este conjunto de índices primários vai fazer com que o join das duas tabelas

seja O(N\_produtos) o que é bastante mais eficiente que O(N\_produtos \* N\_fornecedores\_secundarios)

que seria a complexidade do join se não tivesse estes índices.

**Modelo Multidimensional**

**DROP** **TABLE** d\_produto CASCADE;

**DROP** **TABLE** d\_tempo CASCADE;

**DROP** **TABLE** info\_reposicao CASCADE;

**CREATE** **TABLE** d\_produto(

cean **VARCHAR**(25) **NOT** **NULL**,

categoria **VARCHAR**(50) **NOT** **NULL**,

nif\_fornecedor\_principal **VARCHAR**(9) **NOT** **NULL**,

**PRIMARY** **KEY** (cean)

);

**CREATE** **TABLE** d\_tempo(

dia **INT** **NOT** **NULL**,

mes **INT** **NOT** **NULL**,

ano **INT** **NOT** **NULL**,

**PRIMARY** **KEY**(dia,mes,ano)

);

**CREATE** **TABLE** info\_reposicao(

cean **VARCHAR**(25) **NOT** **NULL**,

dia **INT** **NOT** **NULL**,

mes **INT** **NOT** **NULL**,

ano **INT** **NOT** **NULL**,

numero\_reposicoes **INT** **NOT** **NULL**,

**FOREIGN** **KEY**(dia, mes, ano) **REFERENCES** d\_tempo,

**FOREIGN** **KEY**(cean) **REFERENCES** d\_produto,

**PRIMARY** **KEY**(dia,mes,ano,cean)

);

**INSERT** **INTO** d\_produto

**SELECT** **DISTINCT** R.ean **AS** cean, categoria, forn\_primario **AS** nif\_fornecedor\_principal

**FROM** Produto P , Reposicao R

**WHERE** P.ean = R.ean;

**INSERT** **INTO** d\_tempo

**SELECT** **DISTINCT** **EXTRACT**(**DAY** **FROM** instante)**AS** dia,**EXTRACT**(**MONTH** **FROM** instante)**AS** mes,**EXTRACT**(**YEAR** **FROM** instante) **AS** ano

**FROM** Reposicao;

**INSERT** **INTO** info\_reposicao

**SELECT** **DISTINCT** cean, dia, mes, ano, nro **AS** numero\_reposicoes

**FROM** Reposicao, d\_produto, d\_tempo

**WHERE** **EXTRACT**(**DAY** **FROM** instante) = dia **AND**

**EXTRACT**(**MONTH** **FROM** instante) = mes **AND**

**EXTRACT**(**YEAR** **FROM** instante) = ano **AND**

cean = ean;

**Data Analytics**

**SELECT** categoria, **SUM**(numero\_reposicoes)

**FROM** info\_reposicao,d\_produto

**WHERE** nif\_fornecedor\_principal = '123455678'

**GROUP** **BY** ROLLUP(categoria, ano, mes);