



## Projeto de Bases de Dados, Parte 4

83533	Mihail Brinza	.%	h
83557	Ricardo Brancas	.%	h
83883	David Nunes	.%	h

Grupo 37, turno BD2251795L10  
Professor Miguel Amaral

# 1 Restrições de Integridade

## 2 Índices

### 2.1 Índice 1

Esta *query* contém um **JOIN** cuja condição é de igualdade; a forma mais eficiente de o realizar é percorrendo uma das tabelas (ou um dos índices a ela associados) na totalidade e depois verificar (através de um índice de **HASH**) se existe um registo correspondente na outra tabela. No entanto uma das tabelas (*fornecedor*) já contém um índice sobre a coluna correta (*produto*). Apesar de este índice ser do tipo **BTREE**, muito provavelmente não compensa criar um novo índice do tipo **HASH** (dependendo, no entanto, da situação particular).

Para acelerar a condição **WHERE** devemos ainda criar um índice, também de **HASH**, sobre a coluna *produto.categoria*. Apresentamos os índices em questão:

```
-- CREATE INDEX fornecedor_nif_idx          ON fornecedor USING HASH (nif);
CREATE INDEX produto_categoria_idx         ON produto      USING HASH  (categoria);
```

### 2.2 Índice 2

Em relação à optimização do **JOIN**, estamos numa situação idêntica à da secção 2.1, não sendo necessário criar nenhum índice adicional (devido à existência da *primary key*).

Para otimizar o **GROUP BY** seria útil criar um índice do tipo **BTREE** sobre a coluna *produto.ean* (ou *fornecedor\_sec.ean*) mas, mais uma vez, já existe um índice com essas características associado à *primary key*. Como tal, não é necessário criar nenhum índice adicional para otimizar esta *query*.

## 3 Modelo Multidimensional

```
DROP TABLE IF EXISTS reposition_facts;
DROP TABLE IF EXISTS d_produto;
DROP TABLE IF EXISTS d_tempo;

CREATE TABLE d_produto (
    cean          NUMERIC(13),
    categoria     VARCHAR(80),
    nif_fornecedor_principal NUMERIC(9),
    PRIMARY KEY (cean)
);

CREATE TABLE d_tempo (
    dia SMALLINT,
    mes SMALLINT,
    ano INT,
    PRIMARY KEY (dia, mes, ano),
    CHECK (dia > 0 AND dia <= 31),
    CHECK (mes > 0 AND mes <= 12),
    CHECK (ano > 0)
);

CREATE TABLE reposition_facts (
    cean NUMERIC(13),
    dia SMALLINT,
    mes SMALLINT,
    ano INT,
    PRIMARY KEY (cean, dia, mes, ano),
    FOREIGN KEY (cean) REFERENCES d_produto (cean),
    FOREIGN KEY (dia, mes, ano) REFERENCES d_tempo (dia, mes, ano)
);

INSERT INTO d_tempo
    SELECT DISTINCT date_part('day', instante),
                   date_part('month', instante),
                   date_part('year', instante)
    FROM evento_reposicao;

INSERT INTO d_produto
```

```
SELECT DISTINCT ean, categoria, forn_primario
FROM reposicao NATURAL JOIN produto;

INSERT INTO reposition_facts
SELECT ean,
       date_part('day', instante),
       date_part('month', instante),
       date_part('year', instante)
FROM reposicao;
```

## 4 *Data Analytics*