

Agência *Veritatis*

UC de Bases de Dados (Agência de Detetives)

Ana Cerqueira	A104188
Humberto Gomes	A104348
Ivo Vieira	A103999
José Lopes	A104541
José Matos	A100612

Departamento de Informática – Escola de Engenharia – Universidade do Minho
Licenciatura em Engenharia Informática



31 de maio de 2024

- 1 Definição de Sistema e Requisitos, Modelações Concetual e Lógica
- 2 Implementação Física
- 3 Utilizadores e Privilégios
- 4 Povoamento da Base de Dados
- 5 Cálculo do Espaço da Base de Dados
- 6 Vistas de Utilização
- 7 Tradução de Interrogações
- 8 Indexação
- 9 Procedimentos, Funções e Gatilhos
- 10 Conclusão

Implementação Física – Ordenação

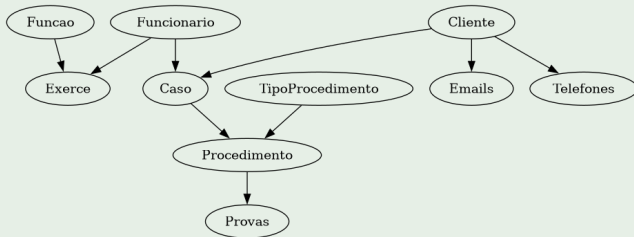
Depois da criação do esquema (CREATE DATABASE ...) ...

Problema

Determinar a ordem de criação de tabelas, visto que chaves estrangeiras exigem que a tabela referenciada já tenha sido criada.

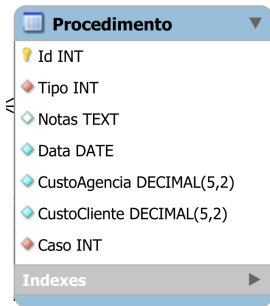
Solução

Criar as tabelas de acordo com uma ordem topológica do grafo abaixo:



Grafo de dependências de criação entre tabelas

Implementação Física – Exemplo



```
CREATE TABLE Procedimento(  
    Id INT NOT NULL AUTO_INCREMENT,  
    Tipo INT NOT NULL,  
    Notas TEXT NULL,  
    Data DATE NOT NULL,  
    CustoAgencia DECIMAL (5,2) NOT NULL DEFAULT 0 CHECK (CustoAgencia >= 0),  
    CustoCliente DECIMAL (5,2) NOT NULL DEFAULT 0 CHECK (CustoCliente >= 0),  
    Caso INT NOT NULL,  
  
    PRIMARY KEY (Id),  
    FOREIGN KEY (Tipo) REFERENCES TipoProcedimento (Id),  
    FOREIGN KEY (Caso) REFERENCES Caso (Id)  
);
```

Utilizadores e Privilégios – Criação de papéis

- Os privilégios de um utilizador estão associados ao seu cargo na empresa (conclusão da leitura dos requisitos de controlo);
- Criaram-se papeis (*roles*), conjuntos de privilégios possivelmente atribuídos a mais do que um utilizador.

```
CREATE ROLE
    'administrativo'@'localhost',
    'detetive'@'localhost',
    'administrador'@'localhost';
```

Utilizadores e Privilégios – Atribuição de privilégios

- Foram atribuídos dos privilégios (INSERT, UPDATE, SELECT e DELETE) conforme os requisitos de manipulação (todos os utilizadores) e de controlo (utilizadores em particular).
- Para interrogações, foram consultados, nas expressões de álgebra relacional desenvolvidas, os atributos de consulta necessária.

Exemplo (RM7):

$$\tau_{\text{DataInicio asc}} \left(\pi_{\text{Id, Designacao, DataInicio}} \left(\sigma_{\text{Cliente} = ?} (\text{Caso}) \right) \right)$$

```
GRANT SELECT(Id, Designacao, DataInicio, Cliente) ON Caso TO
'administrativo'@'localhost',
'detetive'@'localhost',
'administrador'@'localhost';
```

Utilizadores e Privilégios – Criação de utilizadores

Foram criados os utilizadores ...

```
CREATE USER
```

```
'elias.ribeiro'@'localhost' IDENTIFIED BY 'donodistotudo',  
'orlando.feio'@'localhost' IDENTIFIED BY 'alterego',  
'jacinto.fonseca'@'localhost' IDENTIFIED BY 'fonmolhada';
```

...e foram-lhes atribuídos os papéis:

```
GRANT 'administrador'@'localhost' TO 'elias.ribeiro'@'localhost';  
SET DEFAULT ROLE 'administrador'@'localhost' TO  
'elias.ribeiro'@'localhost';
```

```
GRANT 'detetive'@'localhost' TO 'orlando.feio'@'localhost';  
SET DEFAULT ROLE 'detetive'@'localhost' TO  
'orlando.feio'@'localhost';
```

```
GRANT 'administrativo'@'localhost' TO 'jacinto.fonseca'@'localhost';  
SET DEFAULT ROLE 'administrativo'@'localhost' TO  
'jacinto.fonseca'@'localhost';
```

Várias operações de inserção pela ordem de criação das tabelas, para não se referenciar registros inexistentes:

```
INSERT INTO Funcao(Id, Designacao) VALUES  
  (1, 'Gestor financeiro'),  
  (2, 'Gestor de recursos humanos'),  
  (3, 'Administrativo'),  
  (4, 'Detetive'),  
  (5, 'Assistente operacional');
```


- Especificação um formato de dados da BD, que contém também metadados (nomes dos atributos e ordem de povoamento de relações).
- Especificação e implementação de um programa que lê dados no formato especificado, que se conecta ao MySQL e executa as operações de inserção descritas anteriormente.

Cálculo do Espaço da Base de Dados – Método teórico

- Foi consultado o tamanho de todos os tipos de dados usados nos atributos;
- Para cada relação, foi calculado o tamanho de um registo (soma dos tamanhos dos atributos);
- Sabendo o número de registos por tabela, foi calculado o tamanho de cada relação;
- Foi calculado o tamanho da base de dados (soma dos tamanhos das tabelas).

∴ Os requisitos de armazenamento são baixos, mesmo quando se considera o tamanho da base de dados após alguns anos a crescer.

Cálculo do Espaço da Base de Dados – Método prático

O método teórico tem problemas como:

- Não contabiliza o tamanho ocupado pelos índices;
- Não tem em conta o formato de armazenamento dos registos.

Devido à grande complexidade do InnoDB, um método prático foi adotado: a inserção de milhares de registos e o cálculo do tamanho médio por registo com base no tamanho da base de dados medido.

Cálculo do Espaço da Base de Dados – Método prático

O tamanho da base de dados e os requisitos de armazenamento, mesmo maiores, mantêm-se baixos:

Relação	Tamanho do registo	Número de registos	Tamanho
Funcao	41.8	5	209.0
Funcionario	175.3	4	702.1
⋮	⋮	⋮	⋮
Provas	3454.5	6	20727.0
			Total: 59.9 kB

Vistas de Utilização – Um uso possível

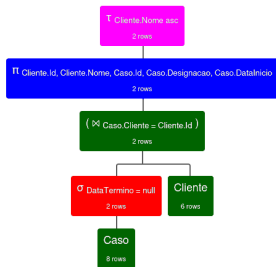
- Não se viu a necessidade de implementação de vistas de utilização, mas requisitos hipotéticos poderiam exigir o seu uso.
- Se um detetive apenas pudesse aceder a alguns atributos dos seus próprios casos:

```
CREATE VIEW vwCasosDetetive2 AS
  SELECT Id,
         Designacao,
         DataInicio,
         DataTermino,
         Cliente,
         Detetive
  FROM Caso
  WHERE Detetive = 2;
```

```
GRANT SELECT ON vwCasosDetetive2 TO 'orlando.feio'@'localhost';
```

Tradução de Interrogações

Trata-se de uma mera mudança da sintaxe da linguagem de álgebra relacional para SQL. Exemplo (RM2, listagem de clientes com casos atualmente em aberto):



```
SELECT Cliente.Id,  
       Cliente.Nome,  
       Caso.Id,  
       Caso.Designacao,  
       Caso.DataInicio  
FROM Caso INNER JOIN Cliente  
      ON Caso.Cliente = Cliente.Id  
WHERE Caso.DataTermino IS NULL  
ORDER BY Cliente.Nome ASC;
```

Nova adição – cores nas árvores de expressões de álgebra relacional.

- Não foi necessária, devido ao bom desempenho resultante do baixo número de registos;
- Foram desenvolvidos índices que ajudariam o desempenho de algumas interrogações, caso a base de dados aumentasse muito de tamanho.

Exemplo (RM2):

```
Caso INNER JOIN Cliente ON Caso.Cliente = Cliente.Id
```



```
CREATE INDEX idxCasoCliente ON Caso(Cliente);
```

- Melhoraria o desempenho porque, para cada cliente, se procuram identificar os casos com base no valor do atributo "Cliente";

Procedimentos, Funções e Gatilhos – Procedimento

Inserção de milhares de registos na base de dados:

```
DELIMITER $$
CREATE PROCEDURE EncheBD(IN nrRegistos INT)
BEGIN
    DECLARE dt DATE DEFAULT CURDATE();

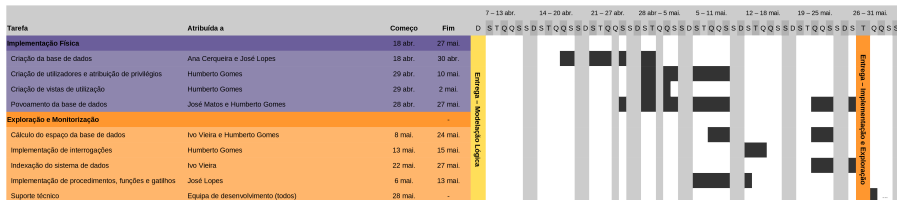
    START TRANSACTION;
    REPEAT
        INSERT INTO Funcao(Id, Designacao) VALUES
            (nrRegistos, REPEAT('A', 20));
        INSERT INTO Funcionario(Id, Nome, NIF, Salario, SeguroVida, Email, Telefone) VALUES
            (nrRegistos, REPEAT('A', 30), 100000000 + nrRegistos, 0000.00, 111111111,
            LPAD(nrRegistos, 30, '0'), LPAD(nrRegistos, 9, '0'));
        -- ...

        IF nrRegistos % 10 = 0 THEN
            INSERT INTO TipoProcedimento
                (Id, Designacao, CustoAgencia, CustoCliente, Descricao) VALUES
                (nrRegistos, REPEAT('A', 20), 000.00, 000.00, REPEAT('A', 2000));
            -- ...
        END IF;

        SET nrRegistos = nrRegistos - 1;
    UNTIL nrRegistos = 0
    END REPEAT;
    COMMIT;
END $$
DELIMITER ;
```


Conclusão

- Realizou-se com correção a maioria do ciclo de uma base de dados, restando apenas a monitorização;
- As tarefas originalmente planificadas não coincidem com as tarefas realizadas. No entanto, nesta fase, foi possível realizar mais tarefas em paralelo do que na fase anterior:



- Algumas decisões tomadas durante a implementação física deveriam ter acontecido anteriormente.

Agência *Veritatis*

UC de Bases de Dados (Agência de Detetives)

Ana Cerqueira	A104188
Humberto Gomes	A104348
Ivo Vieira	A103999
José Lopes	A104541
José Matos	A100612

Departamento de Informática – Escola de Engenharia – Universidade do Minho
Licenciatura em Engenharia Informática



31 de maio de 2024