Análise de Dados

3.ª Aula Prática Laboratorial

Mestrado Integrado em Engenharia Informática

Ano Letivo 2019/2020

Marisa Esteves

11 de Outubro de 2019

Universidade do Minho

Plano de Aula

1. Contextualização sobre os processos de ETL (*Extract, Transform, Load*) e de data warehousing;
2. Continuação da resolução da 2.ª ficha prática laboratorial pelos alunos em grupo.

Processo ETL

Definição

O processo ETL (*Extract, Transform, Load*) é um conjunto de processos que inclui a extração de dados de fontes de informação internas e externas, podendo estar em diferentes formatos, a transformação dos dados de acordo com as necessidades da organização e, finalmente, o carregamento dos mesmos numa estrutura de dados, como por exemplo um data mart ou um data warehouse.

Processo ETL

Definição

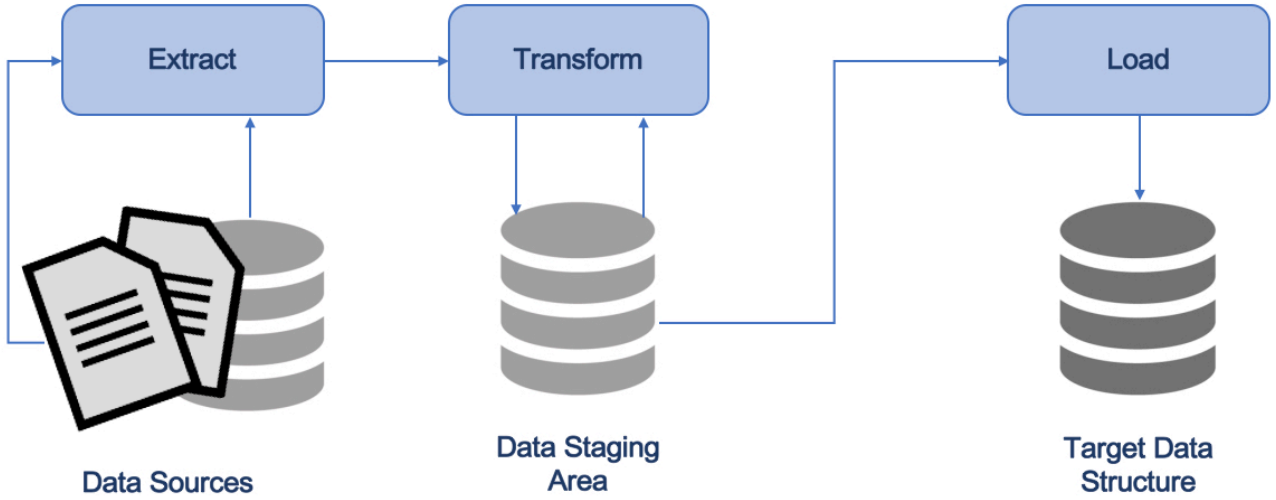

Figura 1 – Esquema do processo ETL.

Processo ETL

Porquê?

Os dados estão espalhados por diferentes localizações

Os dados estão armazenados em diferentes tipos de formato

O volume de dados continua a aumentar

Os dados podem estar estruturados, semi-estruturados ou não estruturados

Data Warehousing

Definição

O processo de data warehousing enfatiza à recolha de dados de diversas fontes através do processo ETL (*Extract, Transform, Load*), correspondendo à construção de data warehouses e/ou data marts, para aceder e analisar a informação de forma útil. Os dados extraídos são processados, formatados e consolidados numa estrutura de dados única para facilitar essencialmente a análise de dados.

Data Warehousing

Definição

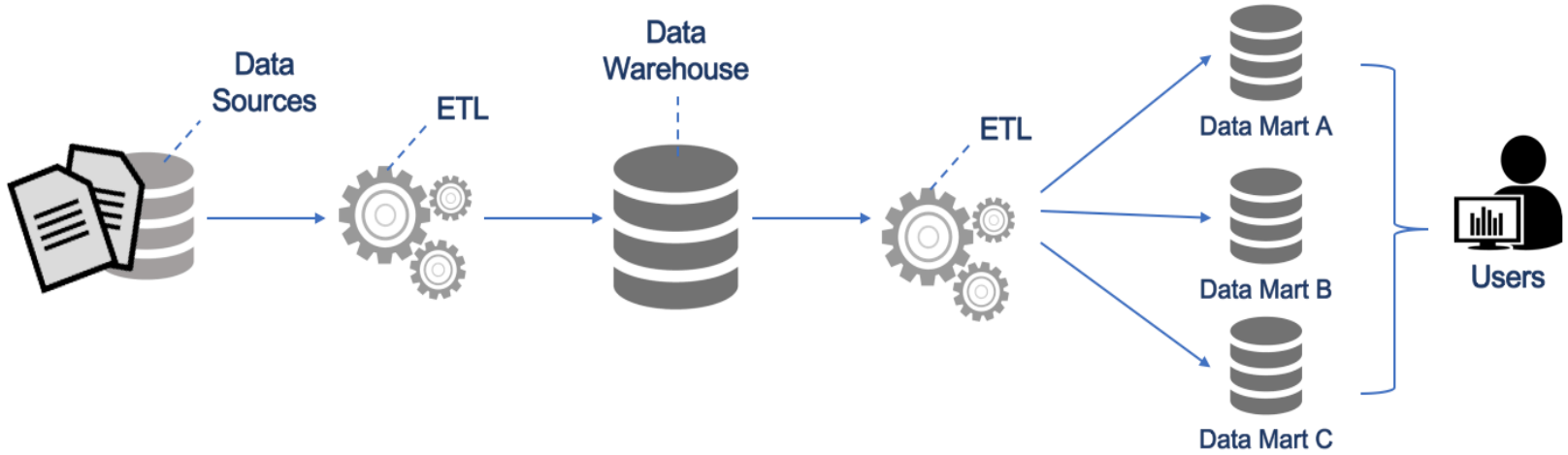

Figura 2 – Esquema do processo de data warehousing.

Data Warehousing

Data Warehouse vs. Data Mart

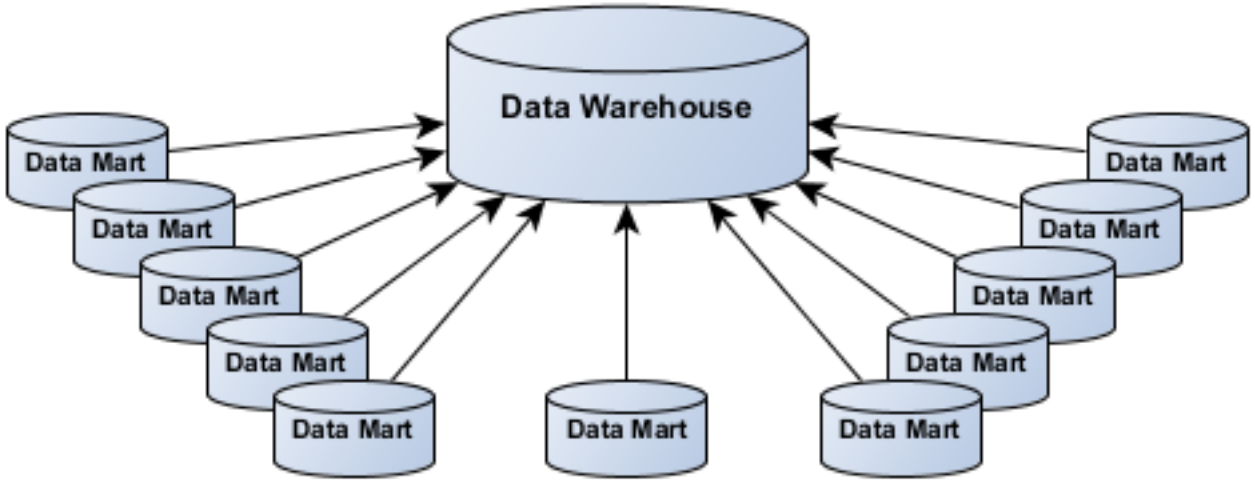

Figura 3 – Data warehouse vs. Data marts.

Data Warehousing

Modelo Dimensional – Esquema em Estrela vs. Esquema em Floco de Neve

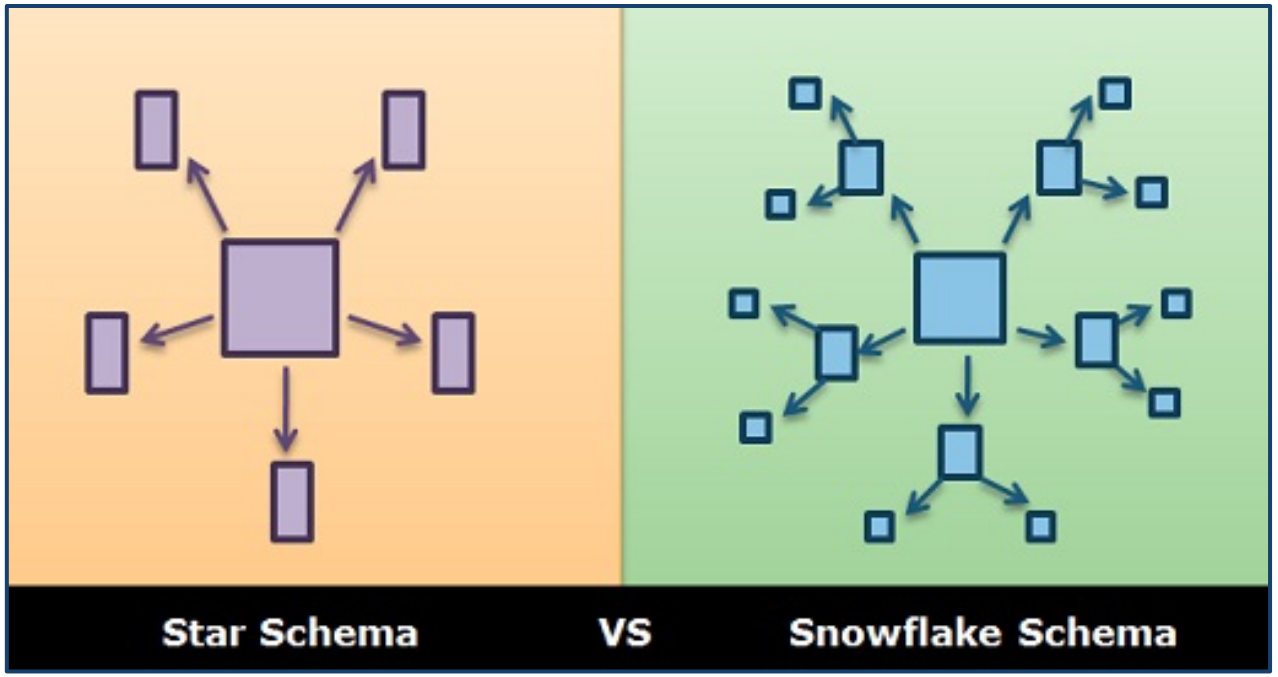

Figura 4 – Esquema em Estrela vs. Esquema em Floco de Neve.

Data Warehousing

Modelo Dimensional – Esquema em Constelação de Factos

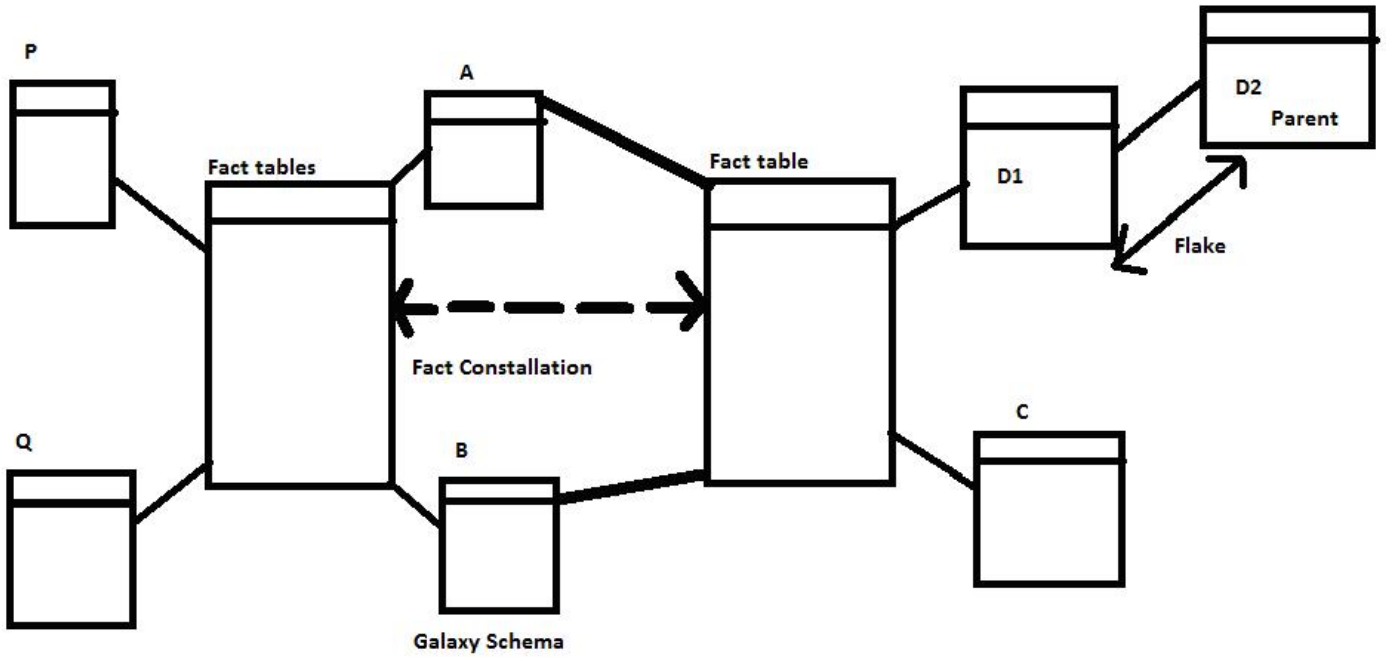

Figura 5 – Esquema em Constelação de Factos.

OLTP vs. OLAP

Definição

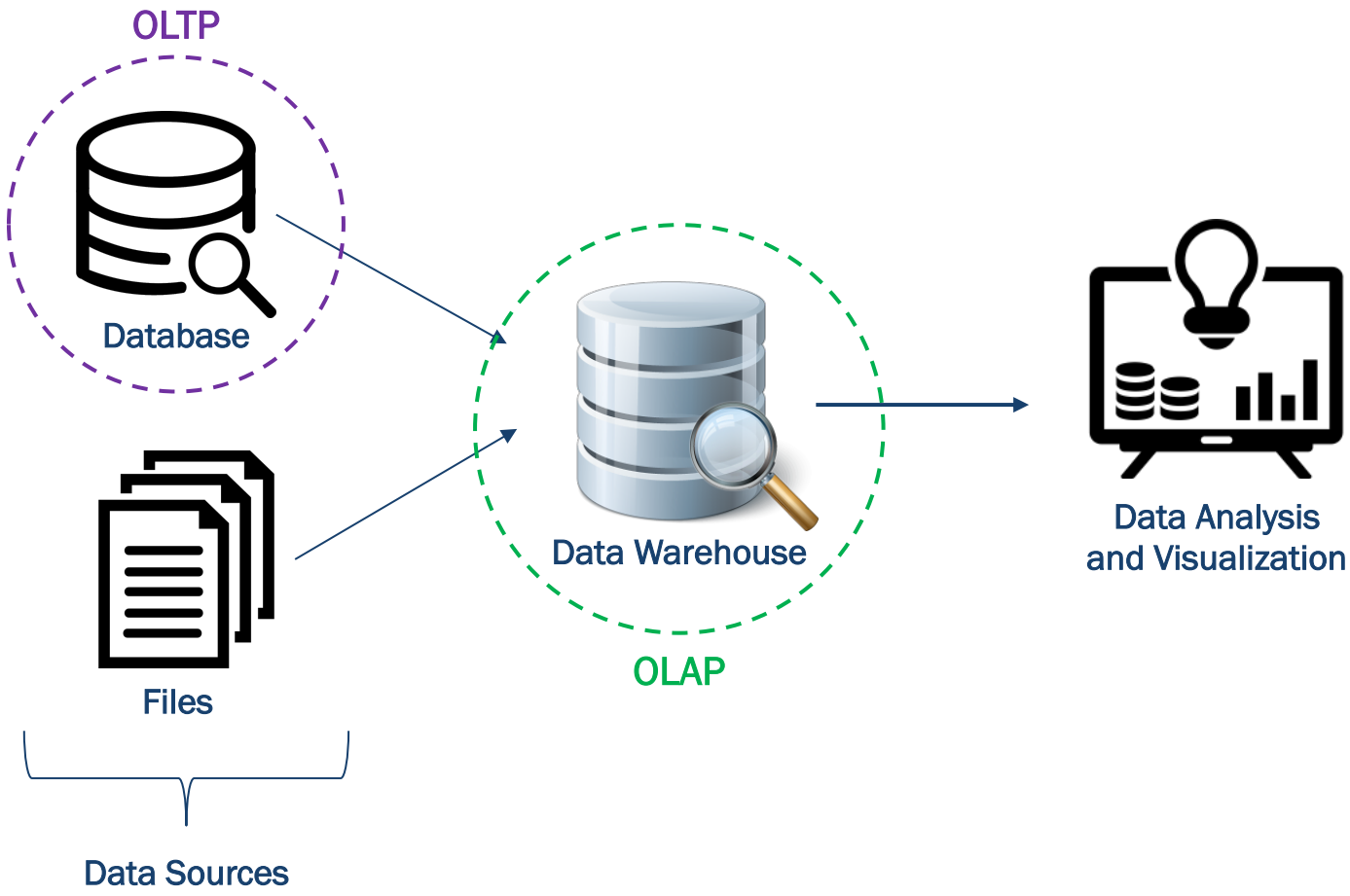

Figura 6 – OLTP (*Online Transaction Processing*) vs. OLAP (*Online Analytical Processing*).

OLTP vs. OLAP

Definição

Relational Database (OLTP)	Analytical Data Warehouse (OLAP)
Contains current data	Contains historical data
Useful in running the business	Useful in analysing the business
Based on Entity Relationship Model	Based on Star, Snowflake or Fact Constellation Schema
Provides primitive and highly detailed data	Provides summarized and consolidated data
Used for writing into the database	Used for reading data from the data warehouse
Database size ranges from 100 MB to 1 GB	Data warehouse ranges from 100 GB to 1 TB
Fast and it provides high performance	Highly flexible but it is not fast
Number of records accessed is in tens	Number of records accessed is in millions
Example: all bank transactions made by a customer	Example: bank transactions made by a customer at a particular time

Figura 7 – Diferenças entre OLTP e OLAP.

MySQL

INSERT INTO SELECT FROM

Permite copiar dados de uma tabela e os inserir noutra tabela.
No entanto, este comando SQL requer que os tipos de dados na tabela de origem (table1) e na tabela destino (table2) sejam iguais.

- **INSERT INTO** *table2* (*column1*, *column2*, *column3*, ...)
  SELECT *column1*, *column2*, *column3*, ...
  FROM *table1*
  WHERE *condition*

MySQL

Cursores

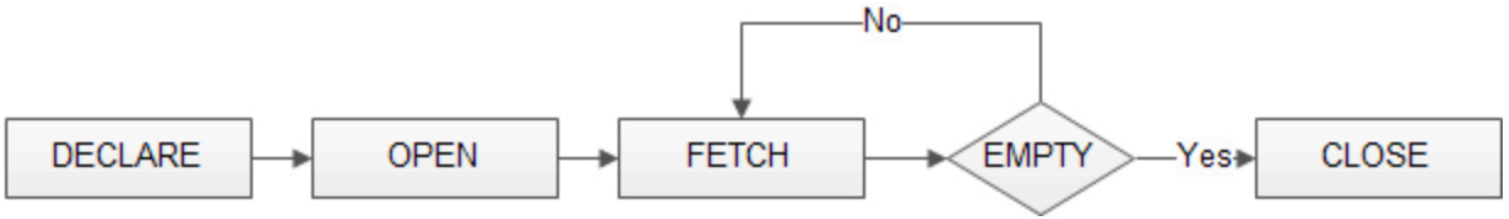

Figura 8 – Modo de funcionamento de cursores em MySQL.

MySQL

Cursores

```
1  CREATE PROCEDURE curdemo()
2  BEGIN
3    DECLARE done INT DEFAULT FALSE;
4    DECLARE a CHAR(16);
5    DECLARE b, c INT;
6    DECLARE cur1 CURSOR FOR SELECT id,data FROM test.t1;
7    DECLARE cur2 CURSOR FOR SELECT i FROM test.t2;
8    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
9
10   OPEN cur1;
11   OPEN cur2;
12
13   read_loop: LOOP
14     FETCH cur1 INTO a, b;
15     FETCH cur2 INTO c;
16     IF done THEN
17       LEAVE read_loop;
18     END IF;
19     IF b < c THEN
20       INSERT INTO test.t3 VALUES (a,b);
21     ELSE
22       INSERT INTO test.t3 VALUES (a,c);
23     END IF;
24   END LOOP;
25
26   CLOSE cur1;
27   CLOSE cur2;
28 END;
```

Figura 9 – Exemplo de um procedimento com cursores em MySQL.

Resolução da 2.ª Ficha Prática Laboratorial

1 Modelo Dimensional de Eventos dos Jogos Olímpicos

O ficheiro disponibilizado juntamente com esta ficha prática laboratorial, nomeadamente athlete_events.csv, contém dados reais históricos sobre os Jogos Olímpicos modernos, incluindo todos os Jogos Olímpicos desde os Jogos Olímpicos de Atenas de 1896 até aos Jogos Olímpicos de Rio de 2016. Os dados de 1000 registos foram extraídos para um ficheiro no formato .csv a partir de um *dataset* inicial com 271116 linhas.

No ficheiro athlete_events.csv, cada linha corresponde a um atleta competindo num determinado evento olímpico. A informação representada inclui 14 colunas, nomeadamente: id (identificador único de cada evento), id_athlete (identificador único do atleta), name_athlete, sex (*M* ou *F*), age, height (em centímetros), weight (em quilogramas), team (nome da equipa), games (ano e temporada), ano, temporada (*Summer* ou *Winter*), cidade (cidade anfitriã), evento (especificação do desporto) e medalha (*Gold*, *Silver*, *Bronze* ou *NA*).

Note-se que os Jogos de Inverno e os Jogos de Verão foram realizados no mesmo ano até 1992. Depois de 1992, os Jogos Olímpicos foram escalonados de tal forma que os Jogos de Inverno ocorrem cada quatro anos começando em 1994, depois os Jogos de Verão em 1996, depois os Jogos de Inverno em 1998, e assim por diante. Um erro comum que poderia cometer ao analisar este *dataset* seria assumir que os Jogos de Inverno e os Jogos de Verão sempre foram escalonados.

Assim, este conjunto de dados representa uma oportunidade única para fazer questões sobre como os Jogos Olímpicos evoluíram ao longo do tempo, incluindo sobre a participação e o desempenho de mulheres, das diferentes nações e dos diferentes desportos e eventos, entre outros.

Os profissionais de tecnologias de informação de uma empresa pretendem remodelar a organização da informação em questão num modelo dimensional baseado no esquema em estrela.

Resolução da 2.ª Ficha Prática Laboratorial

Com base no caso apresentado, pretende-se que:

1. Crie um novo *schema* no MySQL Workbench denominado "Ficha2".

2. Faça o *import* dos dados no ficheiro athlete_events.csv para uma nova tabela no *schema* criado na alínea anterior. No processo, uma tabela denominada "athlete_events" deverá ser criada e povoada corretamente com os dados do ficheiro (Table Data Import Wizard).

3. Analise a estrutura da tabela athlete_events e, consequentemente, define um modelo dimensional no formato de esquema em estrela. O modelo deverá ter uma tabela de factos e as respetivas tabelas de dimensão ligadas à tabela de factos.

4. Construa o modelo dimensional definido na alínea anterior no MySQL Workbench (*EER diagram*).

5. Faça a conversão do modelo lógico criado para o respetivo modelo físico para o *schema* Ficha2 (Database > Forward Engineer).

6. Povoe todas as tabelas do modelo dimensional (tabela de factos e tabelas de dimensão) em SQL a partir da tabela athlete_events. É de relembrar que as tabelas de dimensão deverão ser povoadas antes da tabelas de factos.