

Conceitos Iniciais e Modelo de Dados

O Particular e o Geral

Pastor Alemão
Boxer
Setter
Dobermann
Fox-Terrier

Fiat
Renault
Toyota
Volkswagen
Ferrari

António Silva
Isabel Sousa
Mário Gomes
João Lopes
Pedro Nunes

ABSTRACÇÃO

CÃO

AUTOMÓVEL

EMPREGADO
DONO DO CÃO

Depende do Sistema

Definição de Entidades por Abstracção

- Entidade:
 - Abstracção para a descrição de objectos ou conceitos que possuam um conjunto de características comuns
- Para cada CÃO é necessário conhecer algumas características:
 - o seu nome
 - a raça
 - a data de nascimento
 - a temperatura
 - o nome do seu dono
- Atributo:
 - Característica comum aos objectos ou conceitos que a Entidade retracta

Diferentes perspectivas da Entidade CÃO

- O CÃO visto pelo Veterinário:

CAO

nome	raca	genero	dataDeNascimento	temperatura	peso	numeroDeBIDoDono

DONO_DO_CAO

numeroDeBI	nome	endereco	balancoDaConta

- O CÃO visto pela Administração Municipal:

CAO

numeroDaLicenca	dataDaLicenca	nome	raca	genero	numeroDeBIDoDono

DONO_DO_CAO

numeroDeBI	nome	endereco

Associação

- Existe uma “ligação” entre:
 - CAO e
 - DONO_DO_CAO

CAO

nome	raca	genero	dataDeNascimento	temperatura	peso	numeroDeBI DoDono

DONO_DO_CAO

numeroDeBI	nome	endereço	balancoDaConta

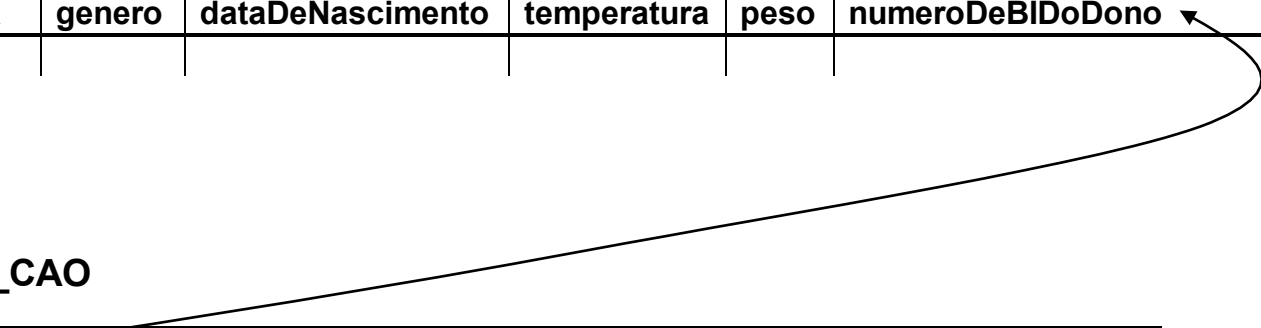
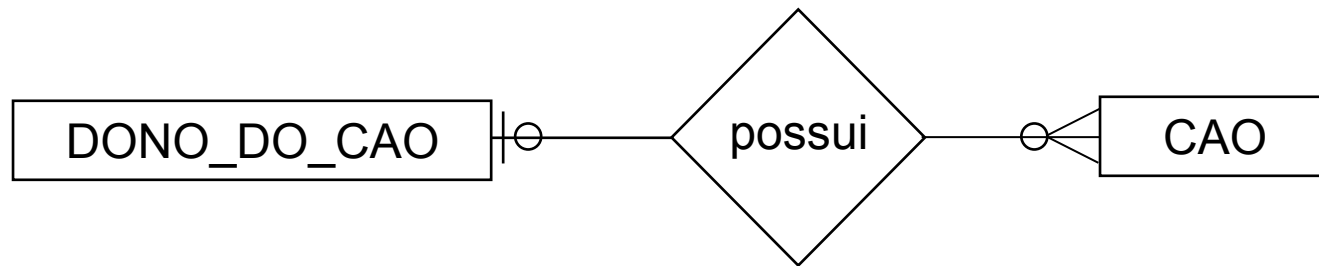


Diagrama Entidade - Associação



Modelo Relacional – a referência dos modelos de dados

- Introduzido por *Codd* em 1970.
- Tem um sólido fundamento teórico.
- Baseia-se na utilização de conceitos abstractos tais como a noção matemática de Relação.
- Representa a base de dados como uma colecção de Relações e Restrições sobre essas Relações.
- Define operações para manipulação das Relações.
- Deste modo a estrutura de conceitos adquire um nível de abstracção suficientemente distante do nível físico permitindo:
 - Atingir um elevado nível de Independência Física.

Elementos da Estrutura de Dados

- Os elementos principais da estrutura de dados da abordagem Relacional são o de:
 - Domínio
 - Atributo
 - Esquema de Relação
 - Relação
- Para exemplificar a apresentação destes conceitos, considere-se a seguinte informação sobre cada “Empregado” (representada em forma de uma tabela):

EMPREGADO

numero	nome	departamento	categoria
31445	António Silva	Contabilidade	5
30442	Isabel Sousa	Armazém	2
27710	Mário Gomes	Vendas	3
35561	João Lopes	Armazém	5
27734	Pedro Nunes	Publicidade	1

Domínio

- Designa-se por Domínio um conjunto de valores “atômicos”.
- Por “atômico” entende-se que cada valor do Domínio é indivisível (na perspectiva do Modelo Relacional).
- Exemplos de Domínios:
 - conjunto dos 5 dígitos válidos para os números de empregado
 - conjunto dos nomes de todos os empregados
 - conjunto dos nomes de todos os departamentos
 - categorias dos empregados - valores entre 1 e 5 (inclusive)

Esquema de Relação e Atributo

- Um Esquema de Relação $R (A_1, A_2, \dots, A_n)$ é constituído pelo:
 - nome do Esquema de Relação: R
 - uma lista (ordenada) de Atributos: A_1, A_2, \dots, A_n
- Cada Atributo A_i é:
 - o nome do papel representado por determinado Domínio no Esquema de Relação R .
- EMPREGADO (numero, nome, departamento, categoria) é um Esquema de Relação onde,
 - o nome do Esquema de Relação é: EMPREGADO
 - a lista de Atributos é: “numero”, “nome”, “departamento”, “categoria”
- O Esquema de Relação é utilizado para “descrever” (ou representar) uma Relação

Relação

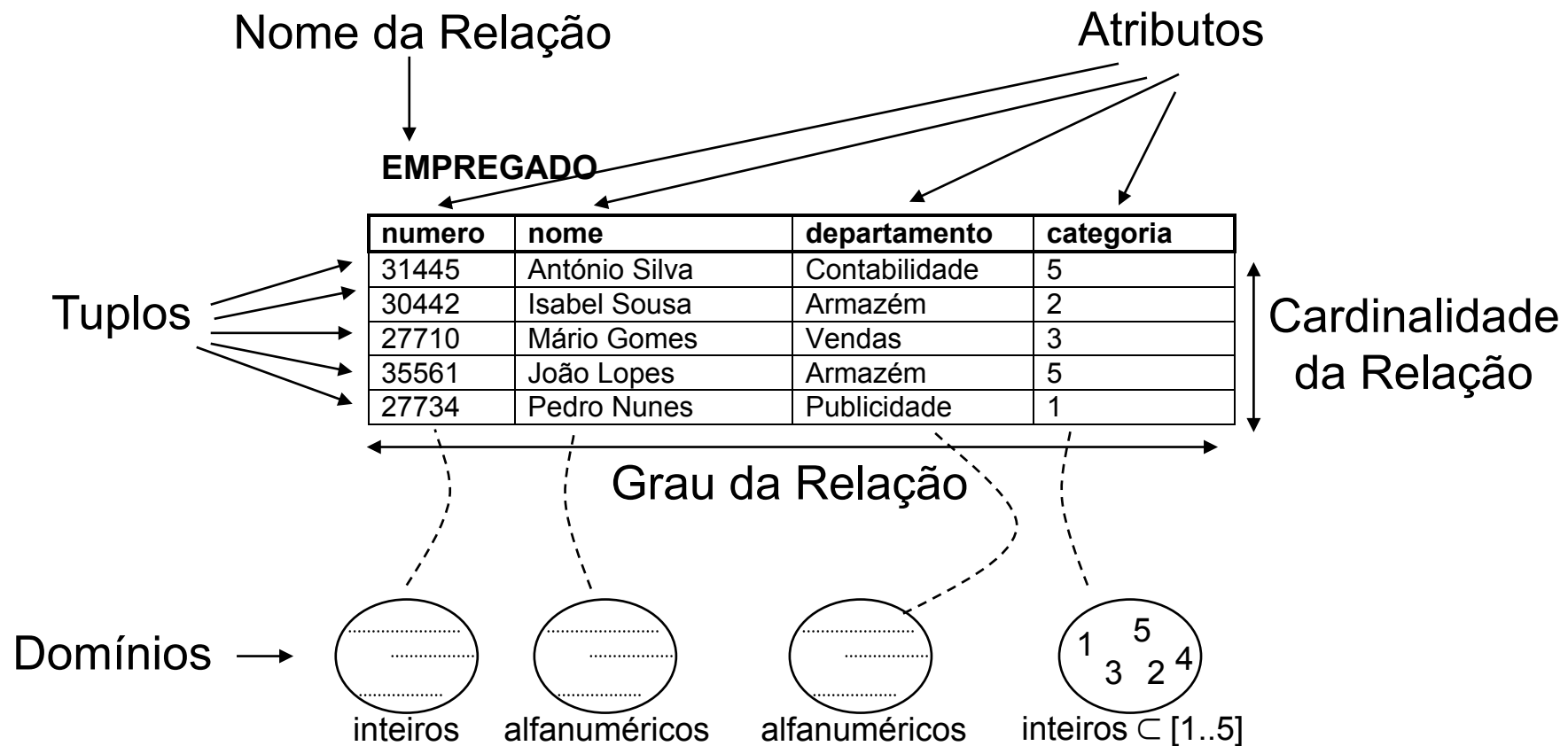
- Uma Relação r de um Esquema de Relação $R (A_1, A_2, \dots, A_n)$, é denotada por $r(R)$ e consiste no:
 - conjunto de tuplos $r = \{t_1, t_2, \dots, t_m\}$, onde
 - cada tuplo t_i é uma lista (ordenada) $\langle v_1, v_2, \dots, v_n \rangle$
 - ◊ onde, para cada v_i : $1 \leq i \leq n$,
 - ◊ $v_i \in D_i$, (Domínio do Atributo i) ou $v_i = \text{NULL}$
- NULL representa a ausência de valor:
 - por não ser conhecido ou,
 - por realmente não existir para determinado atributo em algum tuplo
- A Relação $r(\text{EMPREGADO})$ consiste no conjunto:
 - { $\langle 31445, \text{António Silva}, \text{Contabilidade}, 5 \rangle$,
 - $\langle 30442, \text{Isabel Sousa}, \text{Armazém}, 2 \rangle$,
 - $\langle 27710, \text{Mário Gomes}, \text{Vendas}, 3 \rangle$,
 - $\langle 35561, \text{João Lopes}, \text{Armazém}, 5 \rangle$,
 - $\langle 27734, \text{Pedro Nunes}, \text{Publicidade}, 1 \rangle$ }

Grau e Cardinalidade

- Designa-se por Grau o número de Atributos do Esquema de Relação
- Designa-se por Cardinalidade o número de tuplos da Relação
- O Esquema de Relação:
 - EMPREGADO (numero, nome, departamento, categoria)
 - tem Grau 4
- A Relação:
 - $r(\text{EMPREDADO}) =$
 { <31445, António Silva, Contabilidade, 5>,
 <30442, Isabel Sousa, Armazém, 2>,
 <27710, Mário Gomes, Vendas, 3>,
 <35561, João Lopes, Armazém, 5>,
 <27734, Pedro Nunes, Publicidade, 1> }
 - tem Cardinalidade 5

Apresentação da Relação

- A Relação é geralmente apresentada como uma Tabela, onde:
 - cada tuplo corresponde a uma linha e
 - cada cabeçalho de coluna indica o papel dos valores nessa coluna



As Linhas da Tabela

- A ordem pela qual aparecem as Linhas (Tuplos) na Tabela (Relação) não é importante
 - pode ser alterada sem que isso mude o significado da Relação
- Exemplos da mesma Relação:

EMPREGADO

numero	nome	departamento	categoria
31445	António Silva	Contabilidade	5
30442	Isabel Sousa	Armazém	2
27710	Mário Gomes	Vendas	3
35561	João Lopes	Armazém	5
27734	Pedro Nunes	Publicidade	1

EMPREGADO

numero	nome	departamento	categoria
35561	João Lopes	Armazém	5
27710	Mário Gomes	Vendas	3
31445	António Silva	Contabilidade	5
30442	Isabel Sousa	Armazém	2
27734	Pedro Nunes	Publicidade	1

Esquema Relacional e Base de Dados

- Esquema Relacional
 - Conjunto de Esquemas de Relação que representam determinado sistema
- Base de Dados
 - Conjunto de Relações de determinado Esquema Relacional
- Instância da Base de Dados
 - Base de Dados num determinado instante no tempo

Os “valores atômicos” dos Domínios

- Considere-se o Esquema de Relação:
 - DISCIPLINA_DO_ALUNO (numero, nome, disciplina)
- Considerem-se as Relações,
 - Incorrecta: **DISCIPLINA_DO_ALUNO**

numero	nome	disciplina
1234567	João Alves	Inglês, Português, Matemática
8901234	Marta Guedes	Química, Física, Matemática
5678901	Pedro Lopes	Química, Pintura

- Correcta: **DISCIPLINA_DO_ALUNO**

numero	nome	disciplina
1234567	João Alves	Inglês
1234567	João Alves	Português
1234567	João Alves	Matemática
8901234	Marta Guedes	Química
8901234	Marta Guedes	Física
8901234	Marta Guedes	Matemática
5678901	Pedro Lopes	Química
5678901	Pedro Lopes	Pintura

Superchave

- Uma Relação é um conjunto de tuplos.
- Todos os elementos de um conjunto têm que ser distintos entre si.
- Assim, todos os tuplos de uma Relação têm que ser distintos entre si
 - ou seja, não podem existir dois tuplos com a mesma combinação de valores para todos os seus atributos.
- Designa-se por Superchave o conjunto de Atributos para os quais nunca existem dois tuplos com os mesmos valores.
 - Os valores dos Atributos de uma Superchave permitem identificar univocamente todos os tuplos de uma Relação
 - Um exemplo de Superchave de qualquer Esquema de Relação é o conjunto de todos os seus Atributos
 - Uma Superchave pode ter Atributos “redundantes”

Chave

- Considerando os Esquemas de Relação:
 - EMPREGADO (numero, nome, departamento, categoria)
 - DISCIPLINA_DO_ALUNO (numero, nome, disciplina)
- Podemos considerar,
 - Superchave de EMPREGADO: {numero, nome, departamento}
 - Superchave de DISCIPLINA_DO_ALUNO: {numero, nome, disciplina}
- A Chave K de um Esquema de Relação é uma Superchave com a seguinte condição adicional:
 - retirando qualquer atributo de K resulta um conjunto que já não é uma Superchave
- Para os exemplos, teríamos:
 - Chave de EMPREGADO = {numero}
 - Chave de DISCIPLINA_DO_ALUNO = {numero, disciplina}

Chave Candidata

- Considerando o Esquema de Relação:
 - AUTOMOVEL (numeroMatricula, numeroMotor, modelo, ano)
 - e admitindo que cada automóvel só pode ter um único número de motor
- Possíveis Chaves:
 - numeroMatricula
 - numeroMotor
- Designa-se por Chave Candidata cada uma das possíveis Chaves de um Esquema de Relação.

Chave Primária

- A Chave Primária consiste na Chave Candidata elegida (escolhida) para identificar os tuplos da Relação
- Quando um Esquema de Relação tem várias Chaves Candidatas, em teoria a escolha da Chave Primária pode ser arbitrária.
- No entanto, na prática é usual escolher a Chave Candidata que:
 - tiver um maior significado no sistema em questão, ou
 - que tiver o menor número de Atributos
- No exemplo do AUTOMOVEL, se o sistema em questão fosse o da gestão de um parque de estacionamento poderíamos eleger:
 - Chave Primária: numeroMatricula

Convenção

- Iremos usar a convenção de sublinhar os Atributos que constituem a Chave Primária de um Esquema de Relação.
- No exemplo do AUTOMOVELO teríamos:
 - AUTOMOVELO (numeroMatricula, numeroMotor, modelo, ano)
- No exemplo do EMPREGADO teríamos:
 - EMPREGADO (numero, nome, departamento, categoria)
- No exemplo da DISCIPLINA_DO_ALUNO teríamos:
 - DISCIPLINA_DO_ALUNO (numero, disciplina, nome)

O EMPREGADO e a sua CATEGORIA

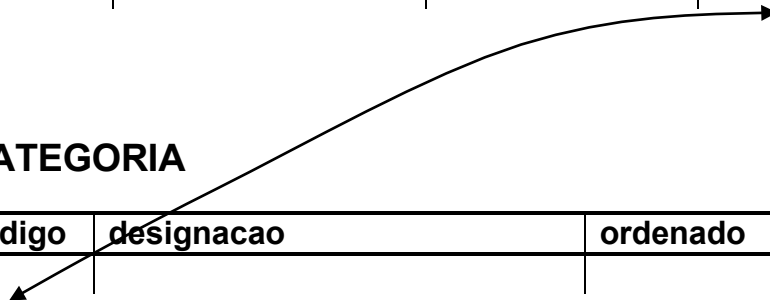
- Considerando os Esquemas de Relação:
 - EMPREGADO (numero, nome, departamento, categoria)
 - CATEGORIA (codigo, designacao, ordenado)
- Tem que ser possível identificar cada tuplo de cada Esquema
- Cada tuplo de uma Relação está “ligado” a um tuplo da outra

EMPREGADO

numero	nome	departamento	categoria

CATEGORIA

codigo	designacao	ordenado



Restrições de Integridade

- Uma Restrição de Integridade:
 - consiste numa condição imposta ao Esquema Relacional
 - restringe os dados que podem existir nas instâncias da Base de Dados
- As Restrições de Integridade:
 - São especificadas quando o Esquema Relacional é definido
 - São verificadas sempre que qualquer Relação é modificada
- As Restrições de Integridade a considerar serão:
 - Integridade de Entidades
 - Integridade Referencial
 - Integridade de Domínio
 - Integridade de Colunas
 - Integridade de Utilizador

Integridade de Entidades e Integridade Referencial

- A condição imposta pela Restrição de Integridade de Entidades é:
 - os valores da Chave Primária não podem ser NULL
 - (NULL na Chave Primária quereria dizer que não seria possível identificar alguns tuplos)
- Quanto à Restrição de Integridade Referencial a questão que se coloca é a seguinte:
 - se um dos tuplos de CATEGORIA for apagado o que acontece ao tuplo, ou tuplos, correspondentes em EMPREGADO ?

Integridade Referencial

- A Restrição de Integridade Referencial é imposta:
 - entre duas Relações
 - é usada para manter a consistência entre os tuplos das duas Relações
- Informalmente, a condição imposta pela Restrição de Integridade de Referencial é:
 - um tuplo numa Relação apenas pode referir outro tuplo que realmente exista noutra Relação

Integridade Referencial (Cont.)

- Considerando os Esquemas de Relação:
 - EMPREGADO (numero, nome, departamento, categoria)
 - CATEGORIA (codigo, designacao, ordenado)

EMPREGADO

numero	nome	departamento	categoria
31445	António Silva	Contabilidade	5
30442	Isabel Sousa	Armazém	2
27710	Mário Gomes	Vendas	3
35561	João Lopes	Armazém	5
27734	Pedro Nunes	Publicidade	1

CATEGORIA

codigo	designacao	ordenado
1	Estagiário	100
2	Técnico	140
3	Responsável do Grupo	200
4	Chefe de Projecto	250
5	Director do Departamento	300



Integridade Referencial e Chave Estrangeira

- A definição mais formal da Restrição de Integridade Referencial leva ao conceito de Chave Estrangeira.
- Um conjunto de Atributos FK num Esquema de Relação R1 é Chave Estrangeira de R1, se
 - Os Atributos em FK têm o mesmo Domínio que os da Chave Primária PK de um outro Esquema de Relação R2
 - O valor de FK num tuplo t1 de R1, ou ocorre como valor de PK para algum tuplo t2 ou é NULL
 - (os Atributos FK dizem-se referências para o Esquema de Relação R2)
- Note-se que uma Chave Estrangeira pode referir o seu próprio Esquema de Relação.
- As Restrições de Integridade Referencial derivam normalmente das Associações existentes entre Entidades representadas pelos Esquemas de Relação.

Integridade de Domínio

- A condição imposta pela Restrição de Integridade de Domínio é:
 - o valor de cada Atributo tem que ser um valor atómico retirado do Domínio desse Atributo
- Os tipos de dados (*data types*) relativos a Domínios, incluem tipicamente valores:
 - numéricos (possivelmente formatados DECIMAL(i, j))
 - cadeia de caracteres (de dimensão fixa ou variável)
 - cadeia de *bits* (de dimensão fixa ou variável)
 - data (com os componentes YEAR, MONTH e DAY)
 - hora (com os componentes HOUR, MINUTE e SECOND)

Integridade de Coluna

- A Integridade de Coluna consiste um refinamento da Integridade de Domínio
- Considerando os Esquemas de Relação:
 - EMPREGADO (numero, nome, departamento, categoria)
 - Os Atributos,
 - ◊ numero
 - ◊ categoria
 - tem como Domínio o conjunto dos valores numéricos
 - e como Restrições de Integridade de Coluna,
 - ◊ “numero” deve ser: valor positivo
 - ◊ “categoria” deve pertencer ao conjunto {1, 2, 3, 4, 5}

Integridade de Utilizador

- A Integridade de Utilizador (ou definida pelo Utilizador) consiste em qualquer outra regra a que as ocorrências de uma determinada Base de Dados deverão obedecer e que não é abrangida pelas restrições atrás mencionadas
- Por exemplo:
 - Um empregado nunca poderá baixar de categoria
 - Não poderá existir nenhum ordenado mais alto do que o correspondente ao da categoria de “Director de Departamento” do departamento de “Direcção”

Verificação das Restrições

- Se um dos tuplos de CATEGORIA for apagado o que acontece ao tuplo, ou tuplos, correspondentes em EMPREGADO ?
- Existem três possibilidades:
 - Apagar automaticamente os tuplos correspondentes em EMPREGADO
 - Inserir NULL nos campos correspondentes à Chave Estrangeira das ocorrências de EMPREGADO correspondentes ao tuplo apagado em CATEGORIA
 - Não permitir apagar qualquer tuplo de CATEGORIA enquanto os tuplos correspondentes em EMPREGADO não forem apagados

Verificação das Restrições (Cont.)

- A cada operação efectuada na Base de Dados o SGBD deve garantir sempre a Integridade de Entidade
 - Sempre que, por exemplo, é acrescentado um novo tuplo, o SGBD deve verificar se o(s) valor(es) presente(s) no(s) campo(s) correspondente(s) à Chave Primária são unívocos.
 - Na caso dos valores não serem unívocos ou serem nulos, deve ser gerada uma mensagem de erro e recusada a operação

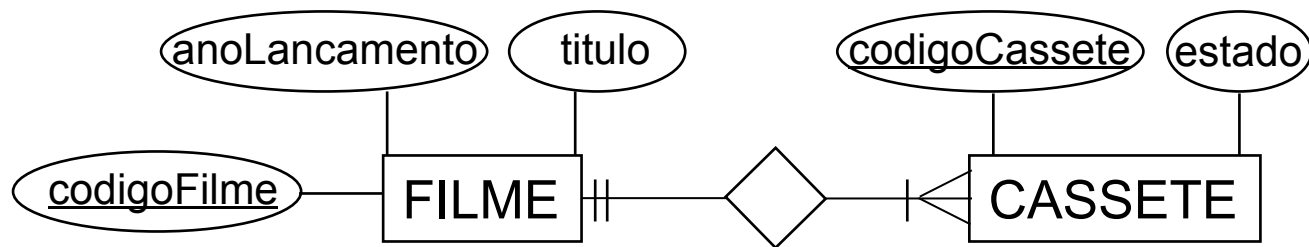
Exemplo (1ª parte): Fornecedores de Filmes de Vídeo

- A Base de Dados tem a seguinte informação acerca de uma empresa de aluguer de cassetes de filmes de vídeo
 - Informação acerca dos filmes existentes na empresa
 - ◊ essa informação inclui o código do filme, que se considera único, o título do filme assim como o ano em que foi lançado
 - Informação sobre cada uma das cassetes que existem na empresa
 - ◊ essa informação inclui o código da cassete (único), indicação de qual o filme gravado na cassete e o estado em que a cassete se encontra (alugado, disponível, perdido ou estragado)

Exemplo (1ª parte): Entidades, Associações e Atributos

- Entidades:
 - FILME
 - CASSETE
- Associações:
 - Saber quais as cassetes de cada filme, implica associar,
 - FILME e CASSETE
- Atributos:
 - código do filme, título, ano de lançamento
 - código da cassete, estado

Exemplo (1ª parte): Modelo Entidade - Associação



Exemplo (1ª parte): Esquema da Base de Dados

- Esquema de Relação FILME
 - FILME (codigoFilme, titulo, anoLancamento)
- Esquema de Relação CASSETE
 - CASSETE (codigoCassete, estado, codigoFilme)
- Chaves Estrangeiras:
 - No Esquema de Relação CASSETE,
 - ◊ existe uma RESTRIÇÃO FK_CASSETE sobre *codigoFilme* que é uma REFERÊNCIA para a Chave Primária de FILME (*codigoFilme*)

Exemplo (1ª parte): Esquema da Base de Dados (Cont.)

- Atributos e tipo
 - codigoFilme: caracter (6)
 - titulo: caracter (255)
 - anoLancamento: inteiro
 - codigoCassete: caracter (6)
 - estado: caracter (10)
- Refinamentos
 - O Atributo *estado* apenas pode ter os seguintes valores:
 - ◊ ‘alugado’, ‘disponível’, ‘perdido’, ‘estragado’
 - O Atributo *anoLancamento* apenas ter tomar valores superiores a 1900

Exemplo (1ª parte): Definição do Esquema da BD

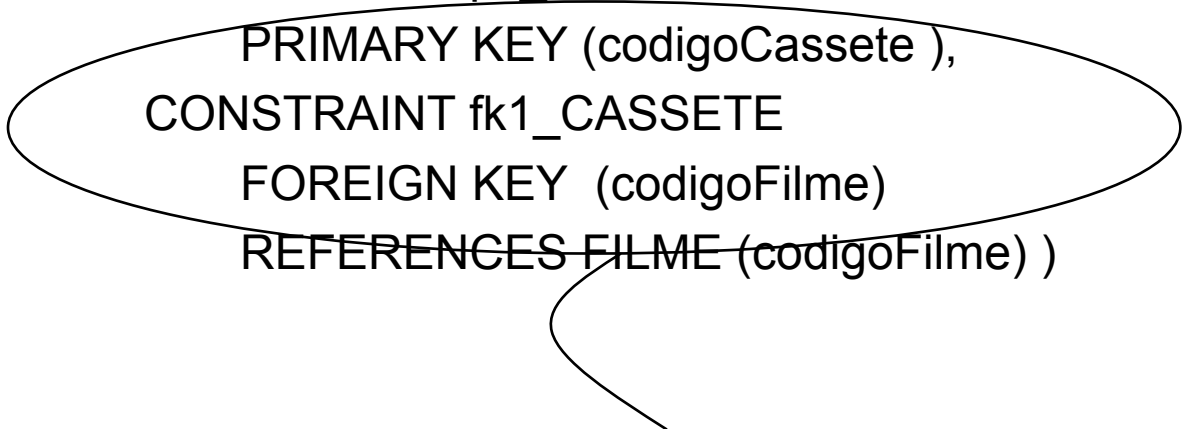
```
CREATE TABLE FILME (  
    codigoFilme CHAR(6) NOT NULL,  
    titulo VARCHAR(255) NOT NULL,  
    anoLancamento INT NOT NULL,  
    CONSTRAINT ck1_FILME  
        CHECK (anoLancamento > 1900),  
    CONSTRAINT pk_FILME  
        PRIMARY KEY (codigoFilme) )
```

Chave Primária

Integridade de Coluna

Exemplo (1ª parte): Definição do Esquema da BD (Cont.)

```
CREATE TABLE CASSETE (  
    codigoCassete CHAR(6) NOT NULL,  
    estado VARCHAR(10) NOT NULL,  
    codigoFilme CHAR(6) NOT NULL,  
    CONSTRAINT ck1_CASSETE  
        CHECK (estado IN ( 'alugado' , 'disponivel' , 'perdido' ,  
            'estragado' )),  
    CONSTRAINT pk_CASSETE  
        PRIMARY KEY (codigoCassete ),  
    CONSTRAINT fk1_CASSETE  
        FOREIGN KEY (codigoFilme)  
        REFERENCES FILME (codigoFilme) )
```



Chave Estrangeira

Exemplo (1ª parte): Base de Dados

FILME

<u>codigoFilme</u>	titulo	anoLancamento
F00001	Gone with the wind	1933
F00002	Terminator	1993

CASSETTE

<u>codigoCassete</u>	estado	codigoFilme
C00001	alugado	F00001
C00002	alugado	F00001
C00003	perdido	F00001
C00004	estragado	F00001
C00005	disponivel	F00002

Interrogar a Base de Dados

- Pretende-se interrogar a Base de Dados para saber:
 - Quais as cassetes do filme 'Terminator' que estão disponíveis ?

```
SELECT CASSETE.codigoCassete
FROM FILME, CASSETE
WHERE
    FILME.titulo = 'Terminator' AND
    FILME.codigoFilme = CASSETE.codigoFilme AND
    CASSETE.estado = 'disponivel'
```

- Como resposta a uma interrogação à Base de Dados obtém-se,
 - uma nova Relação

<u>codigoCassete</u>
C00005

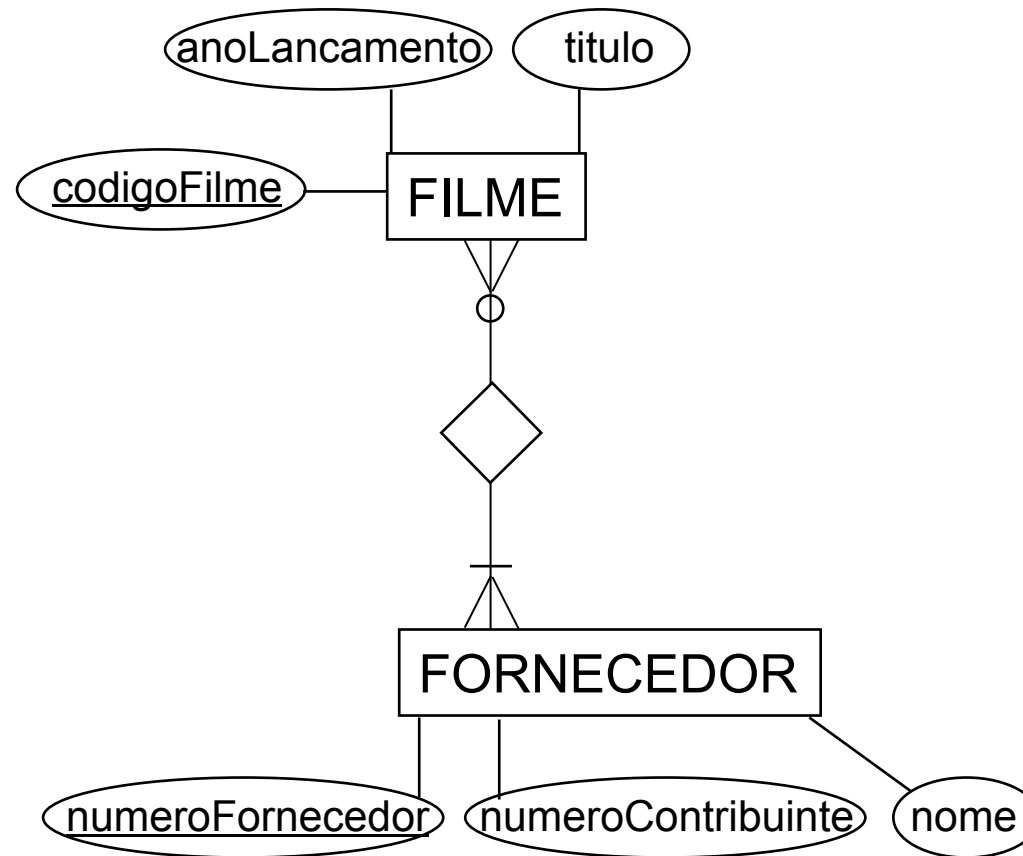
Exemplo (2ª parte): Fornecedores de Filmes de Vídeo

- Pretende-se agora que a Base de Dados já construída inclua:
 - Informação sobre cada um dos fornecedores da empresa
 - ◊ essa informação inclui um número de fornecedor (único) o nome da empresa e número de contribuinte
 - Informação sobre quais os filmes fornecidos por cada fornecedor

Exemplo (2ª parte): Entidades, Associações e Atributos

- Entidades:
 - FORNECEDOR
- Associações:
 - Saber quais os filmes fornecidos por cada fornecedor, implica associar,
 - FORNECEDOR e FILME
- Atributos:
 - numero do fornecedor, numero de contribuinte, nome

Exemplo (2ª parte): Modelo Entidade - Associação



Exemplo (2ª parte): Base de Dados

FORNECEDOR

<u>numeroFornecedor</u>	<u>numeroContribuinte</u>	nome
22222	123456789	Filmes & Comp
11111	987654321	Ibero Fimes
15151	123454321	Só Filmes

FORNECEDOR_FILME

<u>numeroFornecedor</u>	<u>codigoFilme</u>
22222	F00001
22222	F00002

FILME

<u>codigoFilme</u>	titulo	anoLancamento
F00001	Gone with the wind	1933
F00002	Terminator	1993

Exemplo (2ª parte): Esquema da Base de Dados

- Esquema de Relação FORNECEDOR
 - FORNECEDOR (numeroFornecedor, numeroContribuinte, nome)
 - ◊ numeroFornecedor e numeroContribuinte são Chaves Candidatas
- Esquema de Relação FORNECEDOR_FILME
 - FORNECEDOR_FILME (numeroFornecedor, codigoFilme)

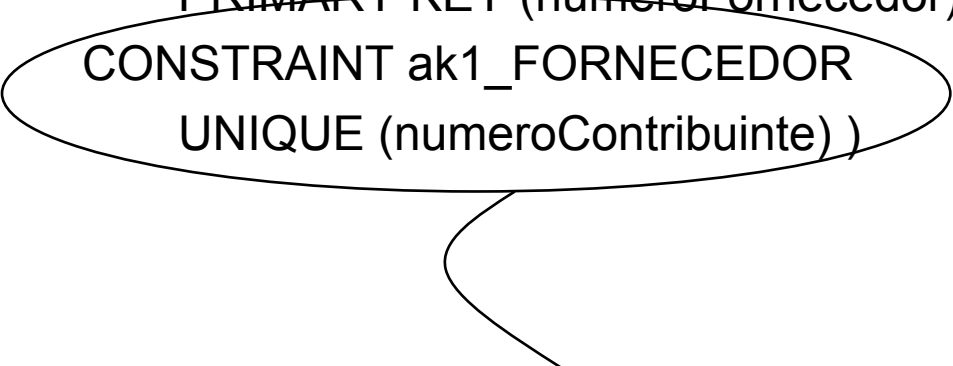
Chave Primária Concatenada

Exemplo (2ª parte): Esquema da Base de Dados (Cont.)

- Chaves Estrangeiras:
 - No Esquema de Relação FORNECEDOR_FILME,
 - ◊ existe uma RESTRIÇÃO FK1_FORNECEDOR_FILME sobre *numeroFornecedor* que é uma REFERÊNCIA para a Chave Primária de FORNECEDOR (*numeroFornecedor*)
 - ◊ existe uma RESTRIÇÃO FK2_FORNECEDOR_FILME sobre *codigoFilme* que é uma REFERÊNCIA para a Chave Primária de FILME (*codigoFilme*)
- Atributos e tipo
 - numeroFornecedor: inteiro
 - numeroContribuinte: caracter (10)
 - nome: caracter (100)

Exemplo (2ª parte): Definição do Esquema da BD

```
CREATE TABLE FORNECEDOR (  
    numeroFornecedor: INT NOT NULL,  
    numeroContribuinte: CHAR(6) NOT NULL,  
    CONSTRAINT pk_FORNECEDOR  
    PRIMARY KEY (numeroFornecedor),  
    CONSTRAINT ak1_FORNECEDOR  
    UNIQUE (numeroContribuinte) )
```



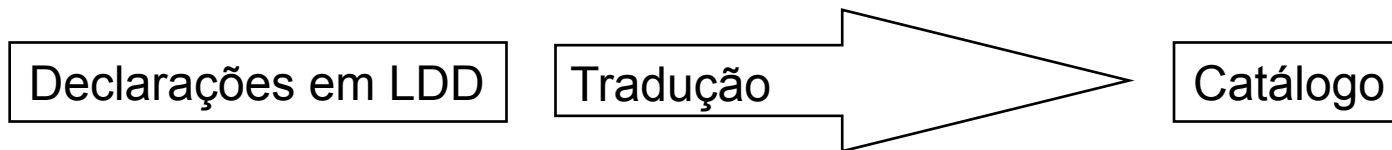
Chave Alternativa

Exemplo (2ª parte): Definição do Esquema da BD (Cont.)

```
CREATE TABLE FORNECEDOR_FILME (  
    numeroFornecedor INT,  
    codigoFilme CHAR(6),  
    CONSTRAINT pk PRIMARY KEY (numeroFornecedor, codigoFilme),  
    CONSTRAINT fk1_FORNECEDOR_FILME  
        FOREIGN KEY (numeroFornecedor)  
        REFERENCES FORNECEDOR (numeroFornecedor),  
    CONSTRAINT fk2_FORNECEDOR_FILME  
        FOREIGN KEY (codigoFilme)  
        REFERENCES FILME (codigoFilme) )
```

Linguagem de Definição de Dados (LDD)

- Do Inglês “Data Definition Language” (DDL).
- Notação usada para definir o esquema da base de dados.
- Tradutor (compilador) de LDD gera um conjunto de tabelas que são armazenadas no Catálogo.
- O Catálogo contém informação sobre os dados (meta informação).



- Alguns autores designam o Catálogo por “*Dicionário de Dados*”

Catálogo

- Consiste num repositório, geralmente gerido pelo SGBD, que contém informação relativa a:
 - Bases de Dados geridas pelo SGBD.
 - Modelos lógicos de dados das Bases de Dados.
 - Utilizadores das bases de dados e seus direitos de acesso.
 - Toda a restante informação necessária à definição e organização dos dados dentro de cada base de dados:
 - ◇ definição das tabelas,
 - ◇ definição dos atributos das tabelas (colunas e seus tipos),
 - ◇ mecanismo de chave para cada tabela,
 - ◇ regras de integridade para as tabelas.
- Um modelo lógico completo consiste não só em diagramas, mas também em especificações descritivas. Estas especificações são guardadas no Dicionário de Dados e podem consistir em simples regras de negócio.

Linguagem de Manipulação de Dados (LMD)

- Do Inglês “Data Manipulation Language” (DML).
- Linguagem que permite:
 - Aceder à informação armazenada na base de dados.
 - Inserir nova informação na base de dados.
 - Eliminar informação na base de dados.
- LMDs procedimentais e não-procedimentais:
 - Procedimentais: o utilizador deve especificar os dados necessários e como obtê-los.
 - Não-procedimentais: o utilizador especifica apenas os dados necessários, sem especificar como obtê-los (sua localização).
- LMDs textuais ou gráficas.
- LMDs *stand-alone* ou embutidas.

SQL (Structured Query Language)

- Unifica a LDD e a LMD numa única linguagem.
- Engloba ainda aspectos como os relacionados com a administração da base de dados.
- É uma linguagem não-procedimental.
- Existem diversas extensões em que os comandos SQL podem ser embutidos em linguagem de programação tradicionais (C, C++, etc).
- O SQL é um standard da ISO e da ANSI.