

Modelo Relacional

Base de Dados - 2023/24

Carlos Costa

Introdução

- Modelo proposto por Edgar F. Codd em 1970
 - garante uma grande independência de dados.

Information Retrieval

P. BAXENDALE, Editor

A Relational Model of Data for Large Shared Data Banks

E. F. CODD

IBM Research Laboratory, San Jose, California

Future users of large data banks must be protected from

— — — — —

In contrast, the problems treated here are those of *data independence*—the independence of application programs and terminal activities from growth in data types and changes in data representation—and certain kinds of *data inconsistency* which are expected to become troublesome even in nondeductive systems.

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for non-inferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

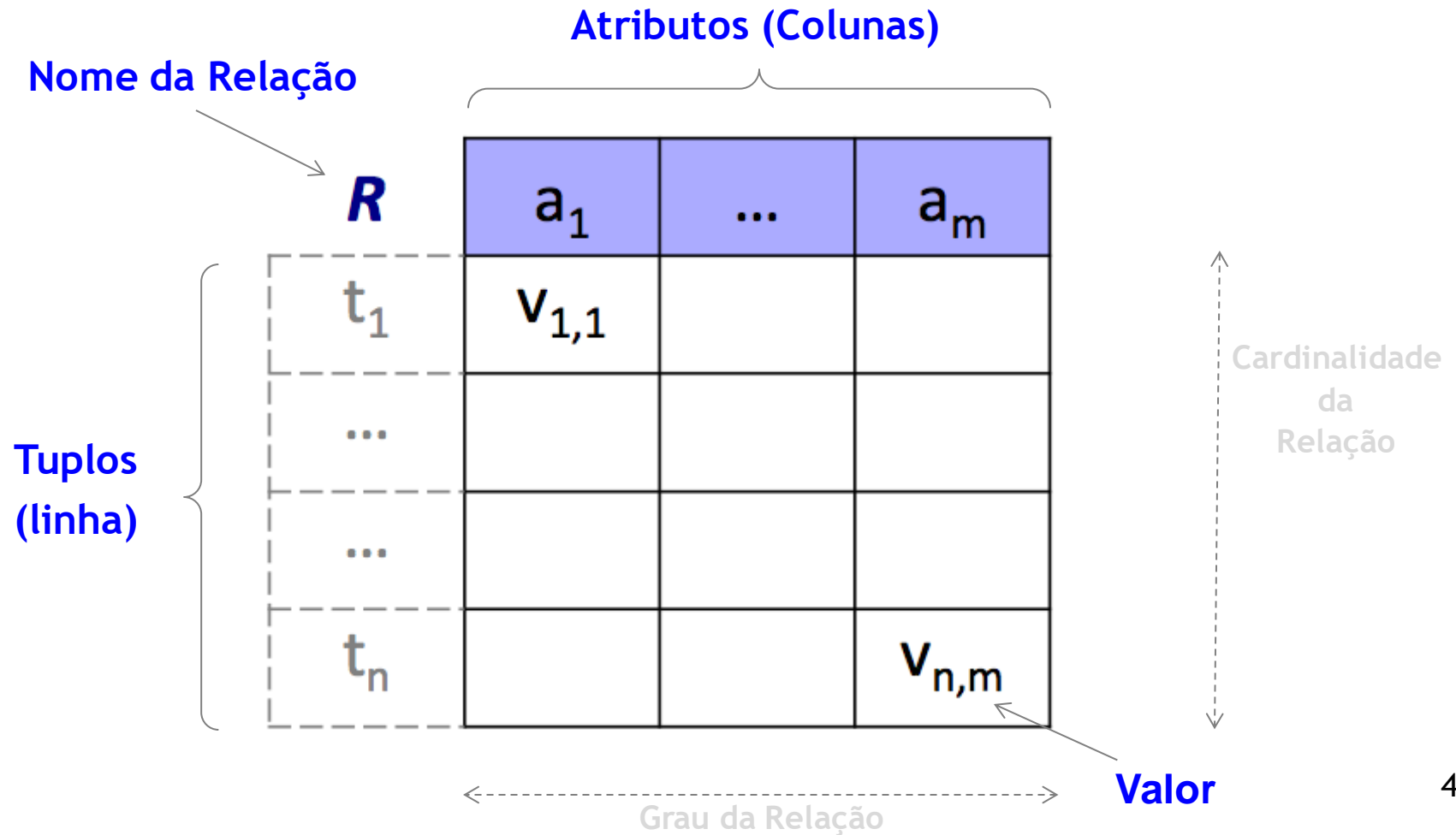
closely associated with the hardware-determined ordering of addresses. For example, the records of a file concerning parts might be stored in ascending order by part serial number. Such systems normally permit application programs to assume that the order of presentation of records from such a file is identical to (or is a subordering of) the

Modelo Relacional - Introdução

- Modelo baseado na Teoria dos Conjuntos.
 - Modelo matemático rigoroso
 - Anteriores evoluíram das técnicas de processamento de ficheiros
- Baseado na noção matemática de “Relação”, representadas por Tabelas.
- Dispõem de um sistema formal de manipulação das relações - Álgebra Relacional (próximas aulas).
- Utilização comercial no início dos anos 80.
 - Devido a restrições de hardware e linguagem de programação
- Contribuiu para a massificação das tecnologias de base de dados.

Conceitos (1/4)

- Base do Modelo Relacional - **Relação** (**Tabela**)



Conceitos (2/4)

- **Atributo** (A_1, A_2, \dots, A_n)
 - Representam o tipo de dados a armazenar.
 - O número de atributos de uma relação define o grau da relação.
 - Os atributos de uma relação devem ter nomes distintos.
- **Domínio** (D_1, D_2, \dots, D_n)
 - Tipo de dados
 - Gama de valores possíveis para determinado atributo.
 - Sexo {'M', 'F'}
 - Cidade {Porto, Aveiro, Coimbra, ...}
 - Nome {Maria, João, Ana, Sofia, ...}
 - Valores desconhecidos ou não existentes.
NULL

Conceitos (3/4)

- **Esquema da Relação** - $R(A_1, A_2, \dots, A_n)$
 - *Relational Schema*
 - Nome do esquema e lista de atributos,
 $Pessoa(nome, bi, idade)$
 - Opcionalmente: inclui o tipo dos atributos
 $Pessoa(nome:string, bi:integer, idade:integer)$
- **Relação** - $r(R)$
 - Estrutura bidimensional com determinado esquema e zero ou mais instâncias (tuplos).
 $r = \{t_1, t_2, \dots, t_m\}$
 - Formalmente é um subconjunto do produto cartesiano
 $r(R) \subseteq (\text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n))$

Conceitos (4/4)

- **Tuplo**

- Linha de uma relação.
 $t = \langle v_1, v_2, \dots, v_n \rangle$
- Devem ser distintos (numa relação) -> *Set*
- A ordem das linhas é indiferente.
- O número de tuplos define a cardinalidade da relação.

- **Atomicidade**

- O valor de um atributo num tuplo é atómico (não é composto/multi-valor).

- **Esquema da Base de Dados (*Database Schema*)**

- conjunto de todos os esquemas da relação da BD.
 $D = \{R_1(X_1), \dots, R_n(X_n)\}$

Relação - Exemplo 1

Relação Estudante

Atributos

| NMec | Nome | Curso |
|-------|--------------------|-------|
| 65022 | João Manuel Sousa | MIECT |
| 65023 | Maria Isabel Costa | TIC |
| 65024 | José Luis Pereira | LMAT |
| ... | ... | ... |

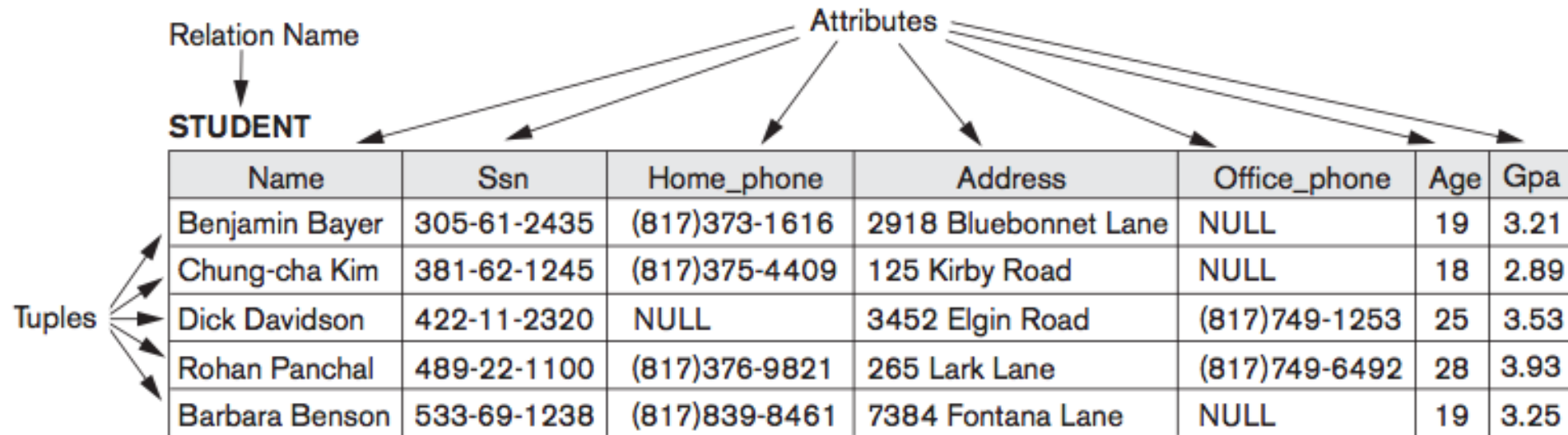
Cardinalidade
da
Relação

Grau da Relação (3)

Esquema de Relação

Estudante(Nmec, Nome, Curso)

Relação - Exemplo 2



- Esquema Relação

STUDENT(Name, Ssn, Home_phone, Address, Office_phone, Age, Gpa)

STUDENT(Name: string, Ssn: string, Home_phone: string, Address: string, Office_phone: string, Age: integer, Gpa: real)

- Tuplo da Relação

$t = \langle \text{(Name, Dick Davidson)}, (\text{Ssn, 422-11-2320}), (\text{Home_phone, NULL}), (\text{Address, 3452 Elgin Road}), (\text{Office_phone, (817)749-1253}), (\text{Age, 25}), (\text{Gpa, 3.53}) \rangle$

Relação - Chaves

- **Superchave** (*superkey*): conjunto de atributos que identificam de forma única os tuplos da relação.
- **Chave Candidata** (*candidate key*): subconjunto de atributos de uma superchave que não pode ser reduzido sem perder essa qualidade de superchave.
- **Chave Primária** (*primary key*): chave principal selecionada de entre as chaves candidatas.
- **Chave Única** (*unique key*): chave candidata não eleita como primária.
- **Chave Estrangeira** ou importada (*foreign key*): conjunto de um ou mais atributos que é chave primária noutra relação. ₁₀

SuperChaves e Chaves Candidatas

- Cada relação tem pelo menos uma superchave
 - Conjunto de todos os atributos

Exemplo

Estudante(Nome, Email, NMec, Curso)

Superchaves:

{Nome, Email, NMec, Curso},
 {Nome, Email, NMec},
 {Nome, Email},
 {Nome, NMec},
 {Email, NMec},
 {Email},
 {NMec}

Lista não exaustiva

Chaves
Candidatas ?

{Email}
{NMec}

Chave Primária

- A **escolha** da **chave primária** (de entre as candidatas) é **arbitrária**.
- As chaves candidatas não eleitas (primária) designam-se como **chaves únicas**.
- A chave primária **não pode** ter valor **NULL**.
- Recomendação: ter critério na escolha da chave primária. Por exemplo:
 - Elemento “natural” de identificação
 - Atributo cujo valor nunca (raramente) é alterado.

No exemplo do slide anterior, qual das chaves candidatas devo escolher para chave primária? Email ou NMec?

Mais razões... ?

Estudante(Nome, Email, NMec, Curso)

Chaves - Exemplo

CAR

| <u>License_number</u> | Engine_serial_number | Make | Model | Year |
|-----------------------|----------------------|------------|---------|------|
| Texas ABC-739 | A69352 | Ford | Mustang | 02 |
| Florida TVP-347 | B43696 | Oldsmobile | Cutlass | 05 |
| New York MPO-22 | X83554 | Oldsmobile | Delta | 01 |
| California 432-TFY | C43742 | Mercedes | 190-D | 99 |
| California RSK-629 | Y82935 | Toyota | Camry | 04 |
| Texas RSK-629 | U028365 | Jaguar | XJS | 04 |

Duas chaves candidatas:

- Licence_number e Engine_serial_number

Escolhemos com chave primária:

- Licence_number

Chaves - Relacionamento entre Tabelas

EMPLOYEE

| Fname | Minit | Lname | <u>Ssn</u> | Bdate | Address | Sex | Dno |
|----------|-------|---------|------------|------------|--------------------------|-----|-----|
| John | B | Smith | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | M | 5 |
| Franklin | T | Wong | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | M | 5 |
| Alicia | J | Zelaya | 999887777 | 1968-01-19 | 3321 Castle, Spring, TX | F | 4 |
| Jennifer | S | Wallace | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | F | 4 |
| Ramesh | K | Narayan | 666884444 | 1962-09-15 | 975 Fire Oak, Humble, TX | M | 5 |
| Joyce | A | English | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | F | 5 |
| Ahmad | V | Jabbar | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | M | 4 |
| James | E | Borg | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | M | 1 |

Chave
Estrangeira

Resumo:

- Temos a relação EMPLOYEE (Funcionário) e a relação DEPARTMENT (Departamento)
- Um EMPLOYEE trabalha num DEPARTMENT
- Dnumber é **chave primária** na relação DEPARTMENT
- Dno é **chave estrangeira** na relação EMPLOYEE

DEPARTMENT

| Dname | <u>Dnumber</u> |
|----------------|----------------|
| Research | 5 |
| Administration | 4 |
| Headquarters | 1 |

Chave Primária

Restrições de Integridade

- São regras que visam garantir a integridade dos dados.
 - Devem ser garantidas pelo próprio SGBD.

Tipos:

- **Domínio** - dos **atributos**. Forma mais elementar de integridade. Os campos devem obedecer ao tipo de dados e às restrições de valores admitidos para um atributo.
- **Entidade** - cada **tuplo** deve ser identificado de forma única com recurso a uma **chave primária** que não se repete e não pode ser null (condição de *set*).
- **Referencial** - o valor de uma **chave estrangeira** ou é **null** ou contém um valor que é **chave primária** na relação de onde foi importada.

Regras de Codd - 1

- Como definir (verificar se) um SGBD é ou não relacional?
- Codd estabeleceu uma lista de 12 regras* que definem/avaliam um sistema de modelo relacional.
- Vários autores (próprio Codd) reconhecem ser difícil encontrar implementações que, à luz das 12 regras, possam ser consideradas completamente relacional.
- No entanto foram muito importantes para combater posicionamentos proprietários da indústria de SGBD.

*Codd, E. (1985). "Is Your DBMS Really Relational?" and "Does Your DBMS Run By the Rules?"
ComputerWorld, October 14 and October 21.

Regras de Codd - 2

1. Representação da Informação

- Numa base de dados relacional, todos os dados, incluindo o próprio dicionário de dados, são representados de uma só forma, em tabelas bidimensionais.

2. Acesso garantido

- Cada elemento de dados fica bem determinado pela combinação do nome da tabela onde está armazenado, valor da chave primária e respectiva coluna (atributo).

3. Suporte sistemático de valores nulos (NULL)

- Valores NULL são suportados para representar informação não disponível ou não aplicável, independentemente do domínio dos respectivos atributos.

4. Catálogo activo e disponível

- Os metadados são representados e acedidos da mesma forma que os próprios dados.¹⁷

Regras de Codd - 3

5. Linguagem completa

- Apesar de um sistema relacional poder suportar várias linguagens, deverá existir pelo menos uma linguagem com as seguintes características:
 - Manipulação de dados, com possibilidade de utilização interativa ou em programas de aplicação.
 - Definição de dados.
 - Definição de views.
 - Definição de restrições de integridade.
 - Definição de acessos (autorizações).
 - Manipulação de transações (commit, rollback, etc.).

6. Regra da atualização de vistas (view)

- Numa vista, todos os dados modificados (em atributos actualizáveis) devem ver essas modificações traduzidas nas tabelas base.

7. Operações de alto-nível

- Capacidade de tratar uma tabela (base ou virtual) como se fosse um simples operando (ou seja, utilização de uma linguagem set-oriented), tanto em operações de consulta como de atualização ou eliminação.

Regras de Codd - 4

8. Independência física dos dados

- Alterações na organização física dos ficheiros da base de dados ou nos métodos de acesso a esses ficheiros (nível interno) não devem afectar o nível lógico.

9. Independência lógica dos dados

- Alterações no esquema da base de dados (nível lógico), que não envolvam remoção de elementos, não devem afectar o nível externo.

10. Restrições de integridade

- As restrições de integridade devem poder ser especificadas numa linguagem relacional, independentemente dos programas de aplicação, e armazenadas no dicionário de dados.

11. Independência da localização

- O facto de uma base de dados estar centralizada numa máquina, ou distribuída por várias máquinas, não deve repercutir-se ao nível da manipulação dos dados.

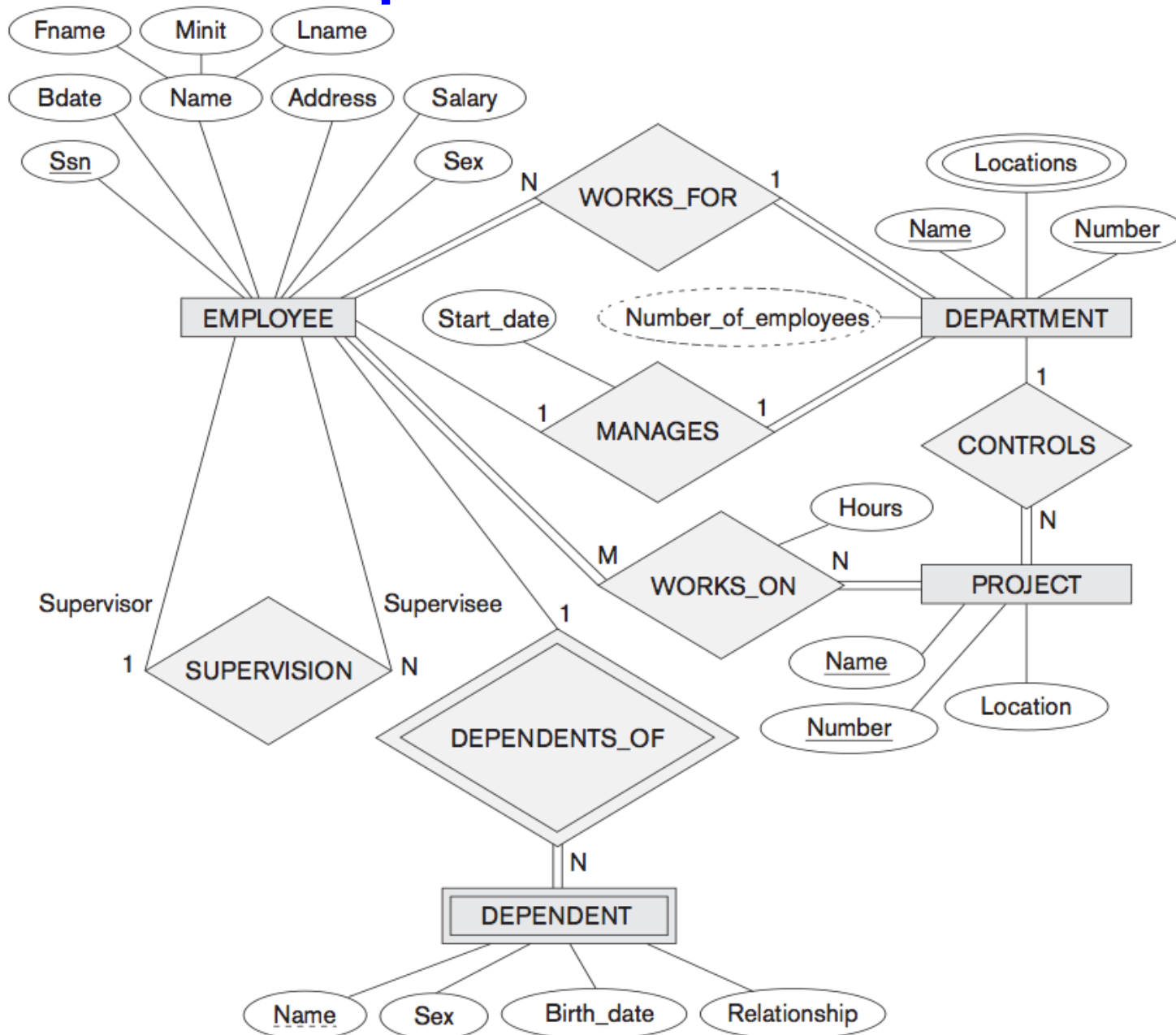
12. Não subversão

- Se existir no sistema uma linguagem de mais baixo-nível (tipo record-oriented), ela¹⁹ não deverá permitir ultrapassar as restrições de integridade e segurança.

Conversão do DER em Modelo Relacional

- Um desenho conceptual de uma base de dados, utilizando DER, pode ser representado por intermédio de um conjunto de relações (tabelas)
- Cada conjunto de entidades e relações do DER vai gerar uma única relação (tabela) com o nome do respectivo conjunto.
- *Mapping Process*
 - Vamos seguir um conjunto de regras.
- Caso Estudo: DER da Empresa

DER Empresa - Caso de Estudo



DER para Relacional - passo 1

- Para cada **entidade regular** E do esquema ER, criar uma relação (tabela) R e inclui todos os atributos de E.
- Incluir os atributos compostos como elementos singulares.
- Selecionar uma das chaves de E para chave primária de R.

EMPLOYEE

| | | | | | | | |
|-------|-------|-------|------------|-------|---------|-----|--------|
| Fname | Minit | Lname | <u>Ssn</u> | Bdate | Address | Sex | Salary |
|-------|-------|-------|------------|-------|---------|-----|--------|

DEPARTMENT

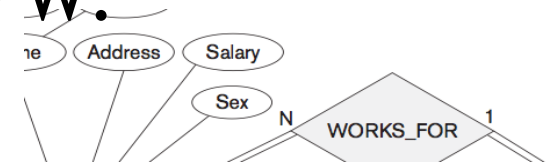
| | |
|-------|----------------|
| Dname | <u>Dnumber</u> |
|-------|----------------|

PROJECT

| | | |
|-------|----------------|-----------|
| Pname | <u>Pnumber</u> | Plocation |
|-------|----------------|-----------|

DER para Relacional - passo 2

- Cada **entidade fraca** W do esquema ER é representada por uma relação (tabela) R que inclui os seu atributos, assim como a chave primária da entidade dominante E que passará a ser chave estrangeira em R.
- Incluir os atributos compostos de W, caso existam, como elementos singulares.
- A chave primária de R é a combinação da chave primária de E e da chave parcial de W.

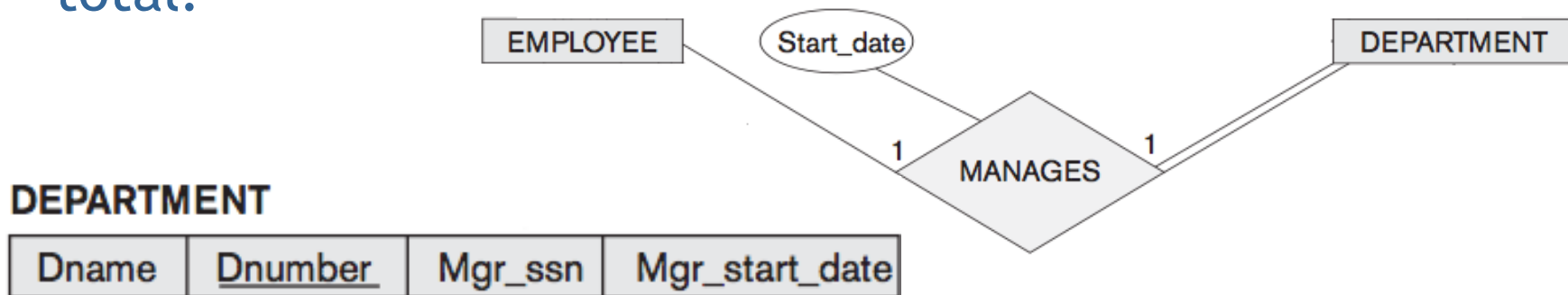


DEPENDENT

| <u>Essn</u> | <u>Dependent_name</u> | Sex | Bdate | Relationship |
|-------------|-----------------------|-----|-------|--------------|
|-------------|-----------------------|-----|-------|--------------|

DER para Relacional - passo 3

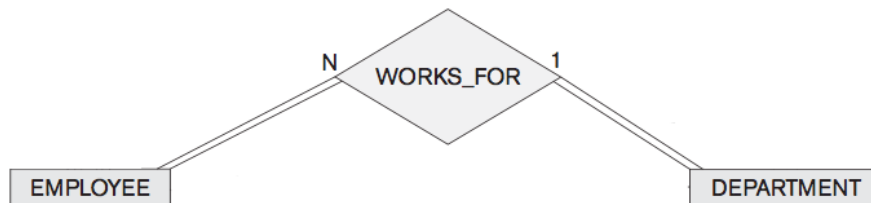
- Para cada **relacionamento 1:1** do esquema ER, envolvendo as relações S e T:
 - escolher uma das relações, digamos S, e incluir como chave estrangeira, a chave primária da outra relação.
 - incluir em S eventuais atributos do relacionamento.
 - devemos escolher como S uma relação com participação total.



Escolhemos com S a relação DEPARTMENT e incluímos a chave primária de EMPLOYEE como chave estrangeira.

DER para Relacional - passo 4

- Para cada **relacionamento 1:N** do esquema ER, envolvendo as relações S e T:
 - escolher como S a relação que representa a entidade do lado N e como T a que representa a entidade do lado 1.
 - incluir em S, como chave estrangeira, a chave primária da relação T.
 - incluir os atributos do relacionamento em S.

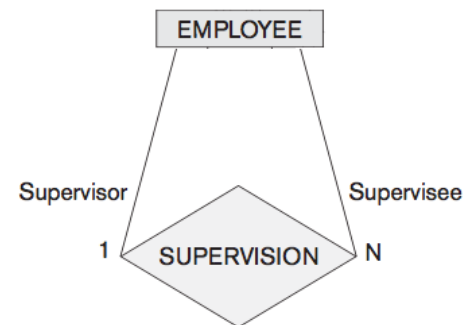
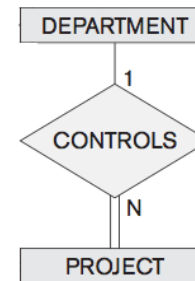


PROJECT

| | | | |
|-------|----------------|-----------|------|
| Pname | <u>Pnumber</u> | Plocation | Dnum |
|-------|----------------|-----------|------|

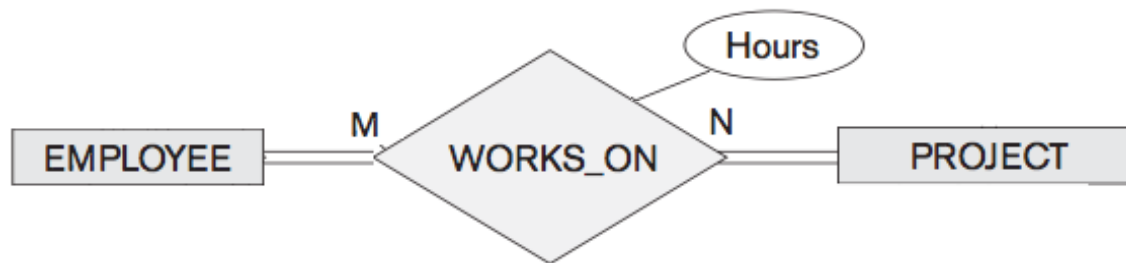
EMPLOYEE

| | | | | | | | | | |
|-------|-------|-------|------------|-------|---------|-----|--------|-----------|-----|
| Fname | Minit | Lname | <u>Ssn</u> | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|-------|-------|-------|------------|-------|---------|-----|--------|-----------|-----|



DER para Relacional - passo 5

- Para cada **relacionamento N:M** do esquema ER, criar uma nova relação (tabela) R.
 - incluir como chave estrangeira as chaves primárias das relações que participam em R. Estas chaves combinadas formarão a chave primária da relação R.
 - incluir os atributos do relacionamento em R.

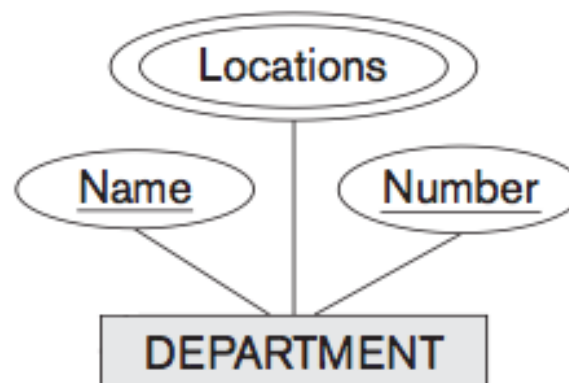


WORKS_ON

| | | |
|-------------|------------|-------|
| <u>Essn</u> | <u>Pno</u> | Hours |
|-------------|------------|-------|

DER para Relacional - passo 6

- Para cada **atributo multi-valor** A do esquema ER, criar uma nova relação (tabela) R.
 - incluir um atributo correspondendo a A.
 - incluir a chave primária K da relação que tem A como atributo.
 - a chave primária de R é a combinação de A e K.



DEPT_LOCATIONS

| | |
|----------------|------------------|
| <u>Dnumber</u> | <u>Dlocation</u> |
|----------------|------------------|

DER para Relacional - passo 7

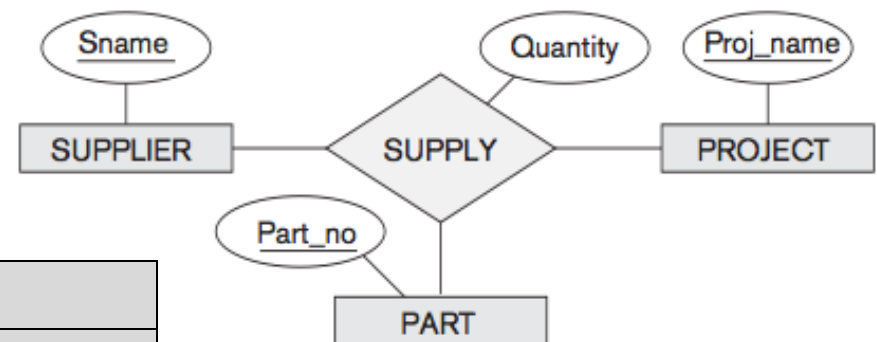
- Para cada **relacionamento n-ário** ($n > 2$):
 - criar uma nova relação (tabela) R
 - incluir, como chaves estrangeiras, as chaves primárias das relações que representam as entidades participantes
 - incluir os eventuais atributos do relacionamento
 - a chave primária de R é normalmente a combinação das chaves estrangeiras

| SUPPLIER | |
|--------------|-----|
| <u>Sname</u> | ... |

| PROJECT | |
|------------------|-----|
| <u>Proj_name</u> | ... |

| PART | |
|----------------|-----|
| <u>Part_no</u> | ... |

| SUPPLY | | | |
|--------------|------------------|----------------|----------|
| <u>Sname</u> | <u>Proj_name</u> | <u>Part_no</u> | Quantity |



Empresa - Esquema da BD

EMPLOYEE

| Fname | Minit | Lname | <u>Ssn</u> | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|-------|-------|-------|------------|-------|---------|-----|--------|-----------|-----|
|-------|-------|-------|------------|-------|---------|-----|--------|-----------|-----|

DEPARTMENT

| Dname | <u>Dnumber</u> | Mgr_ssn | Mgr_start_date |
|-------|----------------|---------|----------------|
|-------|----------------|---------|----------------|

DEPT_LOCATIONS

| <u>Dnumber</u> | <u>Dlocation</u> |
|----------------|------------------|
|----------------|------------------|

PROJECT

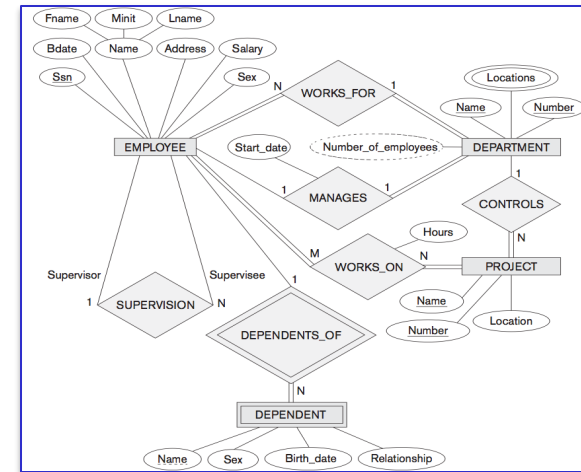
| Pname | <u>Pnumber</u> | Plocation | Dnum |
|-------|----------------|-----------|------|
|-------|----------------|-----------|------|

WORKS_ON

| <u>Essn</u> | <u>Pno</u> | Hours |
|-------------|------------|-------|
|-------------|------------|-------|

DEPENDENT

| <u>Essn</u> | <u>Dependent_name</u> | Sex | Bdate | Relationship |
|-------------|-----------------------|-----|-------|--------------|
|-------------|-----------------------|-----|-------|--------------|



DER

Instância da BD Empresa - Exemplo

EMPLOYEE

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|----------|-------|---------|-----------|------------|--------------------------|-----|--------|-----------|-----|
| John | B | Smith | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | M | 30000 | 333445555 | 5 |
| Franklin | T | Wong | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | M | 40000 | 888665555 | 5 |
| Alicia | J | Zelaya | 999887777 | 1968-01-19 | 3321 Castle, Spring, TX | F | 25000 | 987654321 | 4 |
| Jennifer | S | Wallace | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | F | 43000 | 888665555 | 4 |
| Ramesh | K | Narayan | 666884444 | 1962-09-15 | 975 Fire Oak, Humble, TX | M | 38000 | 333445555 | 5 |
| Joyce | A | English | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | F | 25000 | 333445555 | 5 |
| Ahmad | V | Jabbar | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | M | 25000 | 987654321 | 4 |
| James | E | Borg | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | M | 55000 | NULL | 1 |

WORKS_ON

| Essn | Pno | Hours |
|-----------|-----|-------|
| 123456789 | 1 | 32.5 |
| 123456789 | 2 | 7.5 |
| 666884444 | 3 | 40.0 |
| 453453453 | 1 | 20.0 |
| 453453453 | 2 | 20.0 |
| 333445555 | 2 | 10.0 |
| 333445555 | 3 | 10.0 |
| 333445555 | 10 | 10.0 |
| 333445555 | 20 | 10.0 |
| 999887777 | 30 | 30.0 |
| 999887777 | 10 | 10.0 |
| 987987987 | 10 | 35.0 |
| 987987987 | 30 | 5.0 |
| 987654321 | 30 | 20.0 |
| 987654321 | 20 | 15.0 |
| 888665555 | 20 | NULL |

DEPARTMENT

| Dname | Dnumber | Mgr_ssn | Mgr_start_date |
|----------------|---------|-----------|----------------|
| Research | 5 | 333445555 | 1988-05-22 |
| Administration | 4 | 987654321 | 1995-01-01 |
| Headquarters | 1 | 888665555 | 1981-06-19 |

DEPT_LOCATIONS

| Dnumber | Dlocation |
|---------|-----------|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

DEPENDENT

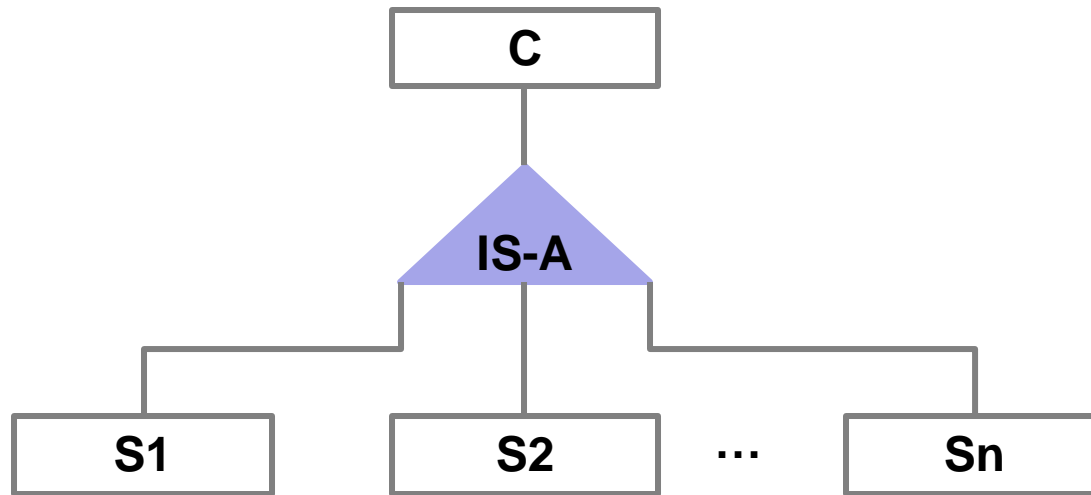
| Essn | Dependent_name | Sex | Bdate | Relationship |
|-----------|----------------|-----|------------|--------------|
| 333445555 | Alice | F | 1986-04-05 | Daughter |
| 333445555 | Theodore | M | 1983-10-25 | Son |
| 333445555 | Joy | F | 1958-05-03 | Spouse |
| 987654321 | Abner | M | 1942-02-28 | Spouse |
| 123456789 | Michael | M | 1988-01-04 | Son |
| 123456789 | Alice | F | 1988-12-30 | Daughter |
| 123456789 | Elizabeth | F | 1967-05-05 | Spouse |

PROJECT

| Pname | Pnumber | Plocation | Dnum |
|-----------------|---------|-----------|------|
| ProductX | 1 | Bellaire | 5 |
| ProductY | 2 | Sugarland | 5 |
| ProductZ | 3 | Houston | 5 |
| Computerization | 10 | Stafford | 4 |
| Reorganization | 20 | Houston | 1 |
| Newbenefits | 30 | Stafford | 4 |

DER para Relacional - Especialização

- Várias aproximações possíveis... vamos apresentar duas usuais.



superclasse $C \{k, a_1, \dots, a_n\}$, k é chave primária
 n subclasses $\{S_1, S_2, \dots, S_n\}$

DER para Relacional - Especialização

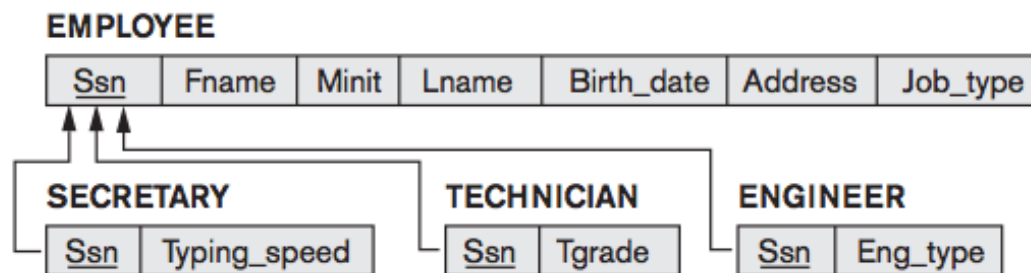
Método 1

- Formar uma relação (tabela) L para a entidade de maior nível (C)

$\text{Attrs}(L) = \{k, a_1, \dots, a_n\}$ e $\text{PK}(L) = k$

- Criar uma relação L_i para cada entidades de nível inferior. Incluir em cada uma destas relações a chave primária de C e os atributos locais.

$\text{Attrs}(L_i) = \{k\} \cup \{\text{attributes of } S_i\}$ e $\text{PK}(L_i) = k$



DER para Relacional - Especialização

Método 2

- Criar uma relação Li para cada entidade de nível inferior. Incluir os atributos da superclasse e os atributos locais.

$\text{Attrs}(Li) = \{\text{attributes of } Si\} \cup \{k, a_1, \dots, a_n\}$ e $\text{PK}(Li) = k$

CAR

| | | | | |
|-------------------|------------------|-------|-----------|------------------|
| <u>Vehicle_id</u> | License_plate_no | Price | Max_speed | No_of_passengers |
|-------------------|------------------|-------|-----------|------------------|

TRUCK

| | | | | |
|-------------------|------------------|-------|-------------|---------|
| <u>Vehicle_id</u> | License_plate_no | Price | No_of_axles | Tonnage |
|-------------------|------------------|-------|-------------|---------|

Só funciona com especialização total.

Só se recomenda em especializações disjuntas pois nas sobrepostas há duplicação de informação da mesma entidade por várias relações (tabelas).

DER para Relacional - Resumo

ER MODEL

Entity type

1:1 or 1:N relationship type

M:N relationship type

n -ary relationship type

Simple attribute

Composite attribute

Multivalued attribute

Value set

Key attribute

RELATIONAL MODEL

Entity relation

Foreign key (or *relationship* relation)

Relationship relation and *two* foreign keys

Relationship relation and n foreign keys

Attribute

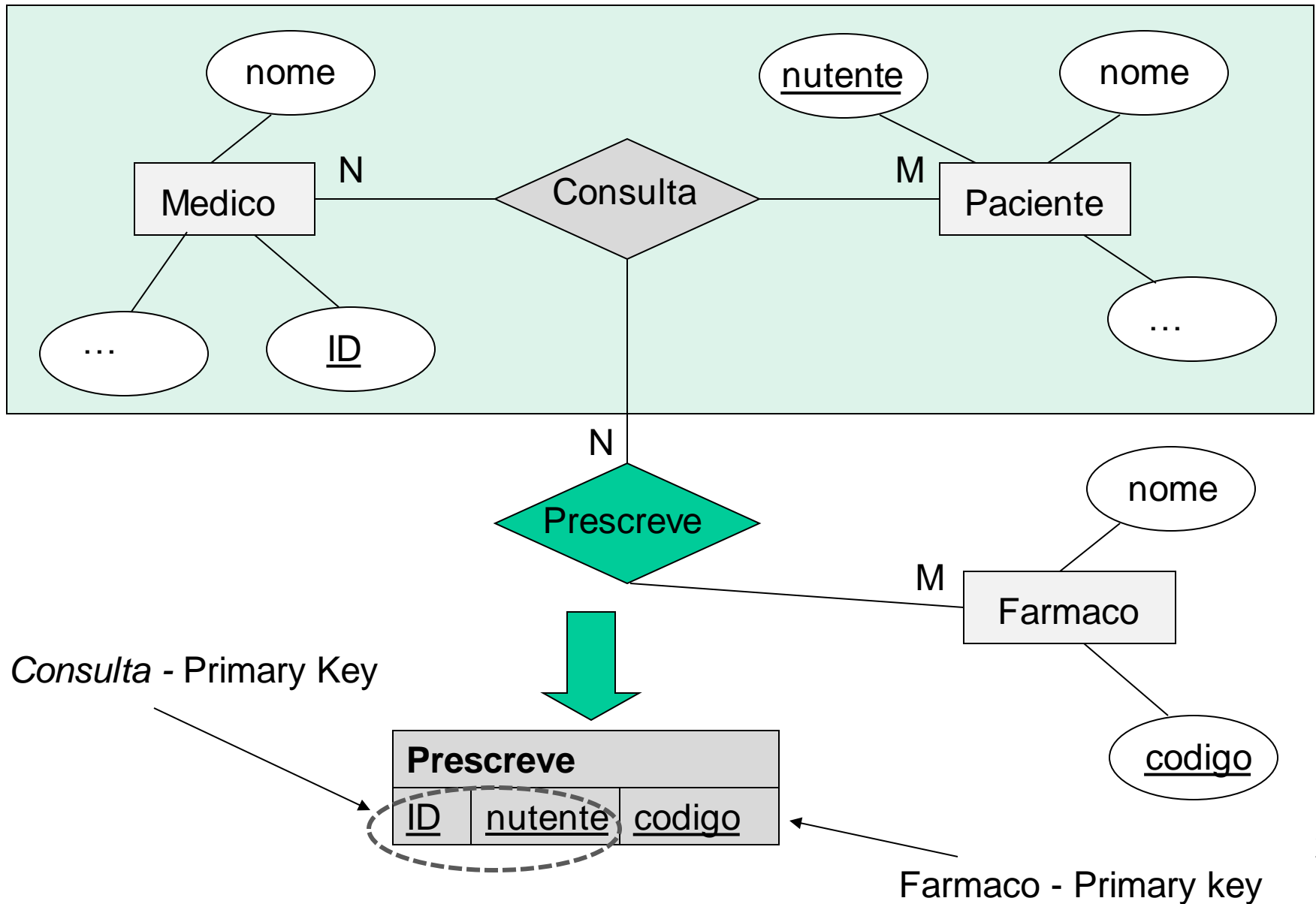
Set of simple component attributes

Relation and foreign key

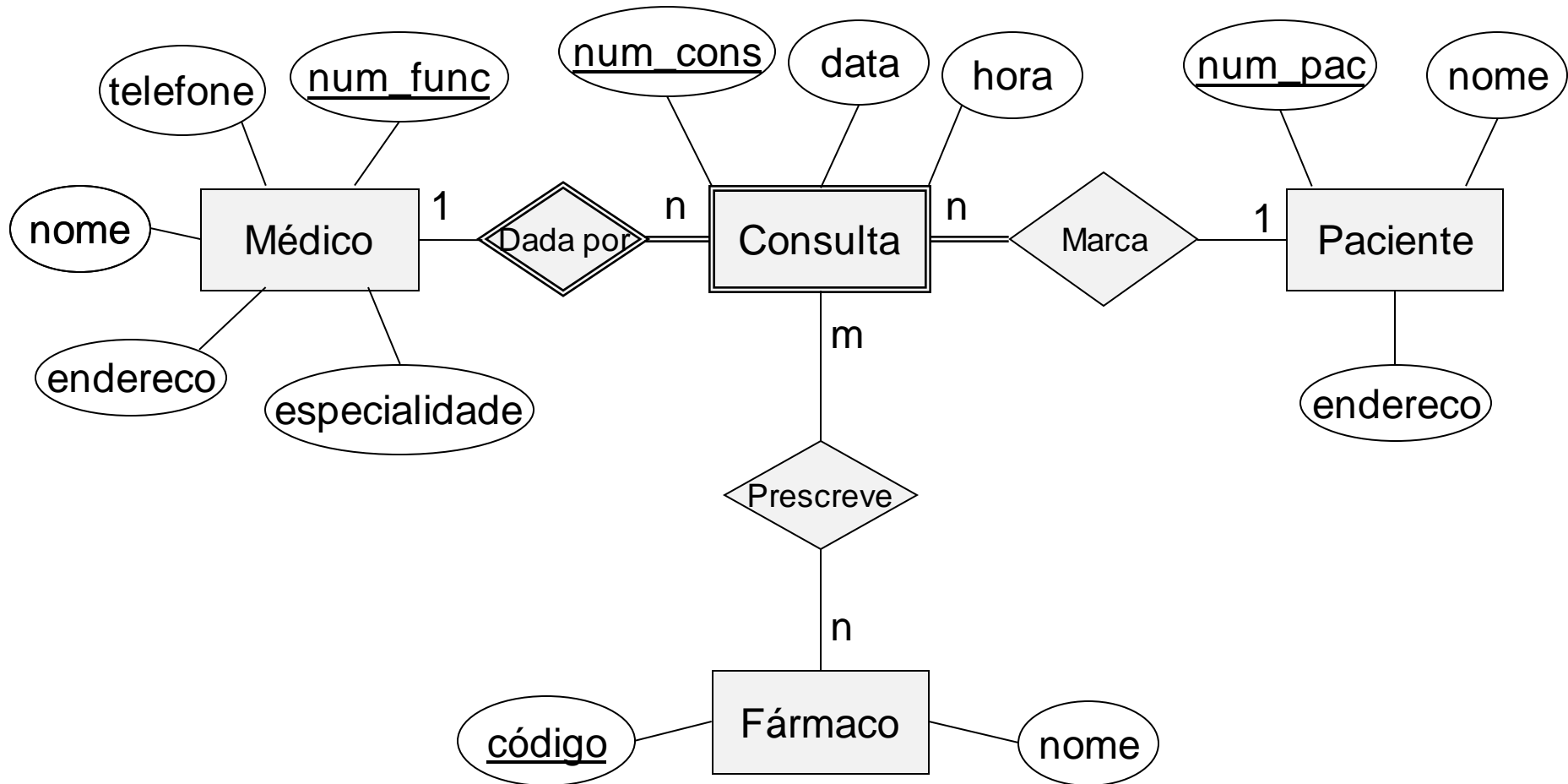
Domain

Primary (or secondary) key

DER para Relacional - Agregação



DER para Relacional - Exemplo da Clínica



DER para Relacional - Exemplo da Clínica

- Passo 1 (entidades regulares)

| Médico | | | | |
|----------------------|------|----------|----------|---------------|
| <u>num_func</u> (PK) | nome | telefone | endereco | especialidade |

| Paciente | | |
|---------------------|------|----------|
| <u>num_pac</u> (PK) | nome | endereco |

| Fármaco | |
|--------------------|------|
| <u>codigo</u> (PK) | nome |

- Passo 2 (entidades fracas)

| Consulta | | | |
|------------------------|--------------------------|------|------|
| <u>medico</u> (FK)(PK) | <u>num_consulta</u> (PK) | data | hora |

DER para Relacional - Exemplo da Clínica

- Passo 3 (rel. 1:1)
 - Não se aplica
- Passo 4 (rel. 1:N)

| Consulta | | | | |
|--------------------------|--------------------------|----------------|------|------|
| <u>medico</u> (FK1) (PK) | <u>num_consulta</u> (PK) | paciente (FK2) | data | hora |

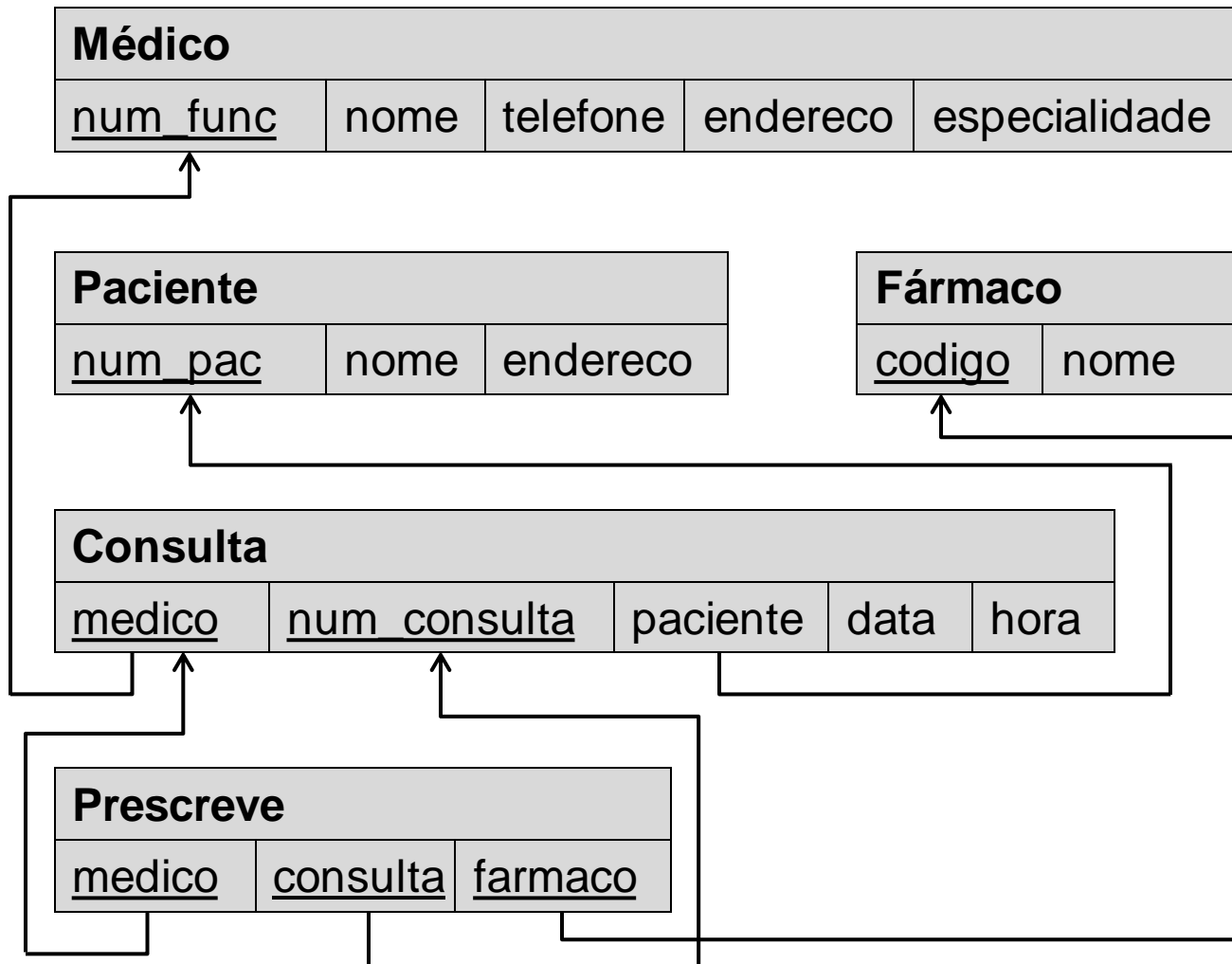
- Passo 5 (rel. N:M)

| Prescreve | | |
|-------------------------|---------------------------|--------------------------|
| <u>medico</u> (FK1)(PK) | <u>consulta</u> (FK1)(PK) | <u>farmaco</u> (FK2)(PK) |

- Passo 6 e 7
 - Não se aplicam

DER para Relacional - Exemplo da Clínica

Esquema Relacional da Base de Dados

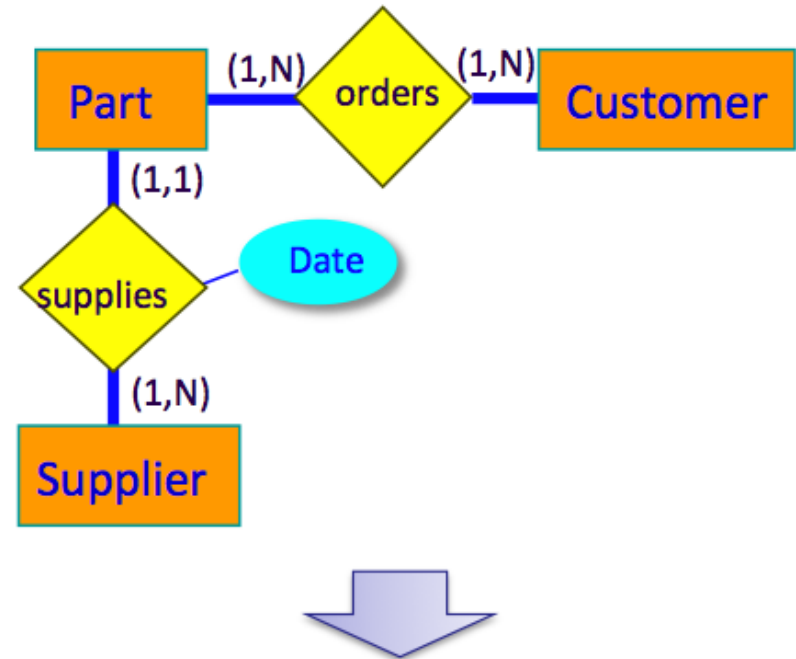


A Seguir?

The “real world”



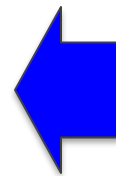
The E/R Model (Conceptual Model)



SQL – describe Database Schema

```

CREATE TABLE DEPARTMENT
( Dname          VARCHAR(15)    NOT NULL,
  Dnumber        INT           NOT NULL,
  Mgr_ssn        CHAR(9)       NOT NULL,
  Mgr_start_date DATE,
  PRIMARY KEY (Dnumber),
  UNIQUE (Dname),
  FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn) );
  
```



The Relational Schema

Part (Name,Description,Part#)
Supplier (Name, Addr)
Customer (Name, Addr)
Supplies (Name,Part#, Date)
Orders (Name,Part#)

Resumo

- Desenho Lógico de BD
- Modelo Relacional
- Restrições de Integridade
- Conversão de Diagramas Entidade-Relação para Esquema Relacional
- Casos de Estudo