

O Modelo ER

Bases de Dados (CC2005)

**Departamento de Ciência de Computadores
Faculdade de Ciências da Universidade do Porto**

Eduardo R. B. Marques — DCC/FCUP

Modelação conceptual de BDs

■ De requisitos a modelo conceptual ...

- Tendo identificado os requisitos do universo de uma BD torna-se útil a sua modelação conceptual.
- A modelação conceptual tem por propósito definir um modelo para a BD **independente** do tipo de base de dados e SGBD específico que depois se empregue na fase de implementação.

■ Modelo Entidade-Relacionamento (ER)

- Modelo usado para desenho conceptual de uma BD, empregando os conceitos de **entidades**, **atributos** e **relacionamentos**.
- Tem associadas uma **sintaxe textual** e também uma **sintaxe visual na forma de diagramas ER**.

Conceitos do modelo ER

■ Entidades

- **Objetos ou conceitos do mundo real com uma existência independente.**
- ALUNO PESSOA CURSO CARRO EMPRESA FACULDADE PRODUTO ...

■ Atributos

- **Propriedades que caracterizam as entidades.**
- Atributos exemplo para um ALUNO: NumMec Nome Sexo DataNasc

■ Relacionamentos

- **Representam ligações entre duas ou mais entidades.**
- Relacionamento exemplo ESTUDA: um ALUNO **estuda** em uma FACULDADE.

Entidades e atributos

Entidade-tipo

- **Entidade-tipo:** Modelo para um conjunto de entidades que partilham a mesma estrutura, definido por um **nome** e uma **lista de atributos**.
- **ALUNO(NumMec, Nome, Sexo, DataNasc)**
 - **Nome:** ALUNO
 - **Atributos:** NumMec, Nome, Sexo, DataNasc
 - Entidade-tipo modelando o universo de alunos com os seguintes atributos: n.º mecanográfico, nome, sexo e data de nascimento.
- Convenções
 - **NOME** em maiúsculas e singular.
 - Atributos: **PrimeiraLetraDeCadaPalavra** maiúscula. Abreviaturas são comuns (como em NumCC ou DataNasc).

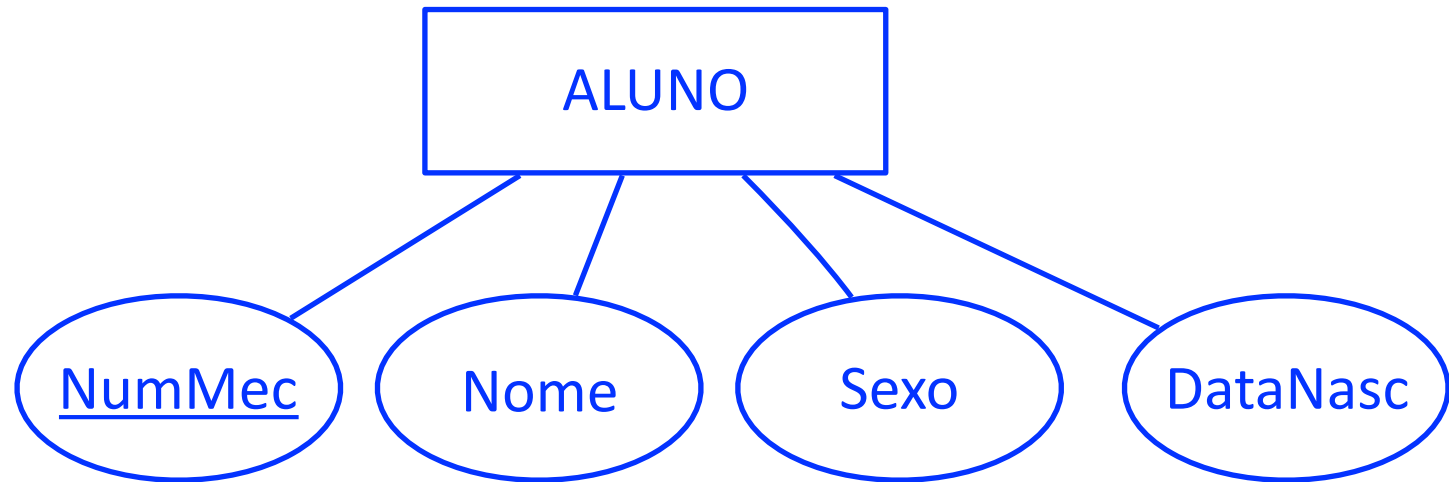
Entidade

- Uma **entidade** é uma instância concreta do esquema modelado por uma entidade-tipo.
- Para **ALUNO(NumMec, Nome, Sexo, DataNasc)** podemos ter por exemplo as seguintes entidades:
 - **ALUNO₁(193942, 'José Silva', M, 17-11-2000)**
 - **ALUNO₂(212334, 'Maria Carvalho', F, 08-02-1976)**
 - **ALUNO₃(112345, 'Rita Assunção', F, 09-02-1987)**
 - **ALUNO₄(244556, 'Rita Assunção', F, 08-02-1987)**
- Dois alunos podem ter o mesmo nome, sexo, e (ainda menos provável mas possível) até a mesma data de nascimento ...
- O número mecanográfico - **NumMec** - **identifica de forma única** uma entidade **ALUNO**.

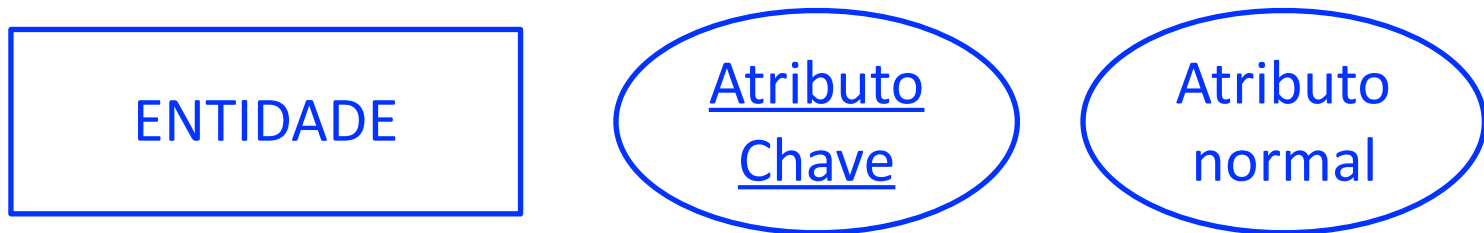
Atributo chave

- **Atributo chave:** atributos que identifica de forma única cada entidade.
- **ALUNO(NumMec, Nome, Sexo, DataNasc)** tem apenas um atributo-chave: NumMec.
- A convenção sintáctica é que um atributo chave apareça sublinhado.
- Supondo que um aluno era adicionalmente caracterizado por um n.º de cartão de cidadão **NumCC** então teríamos dois atributos chave **ALUNO(NumMec, NumCC, Nome, Sexo, DataNasc)**.

Entidades-tipo – diagrama ER



- Além da representação textual o modelo ER emprega uma notação visual na forma de **diagrama**.
- Sintaxe visual:

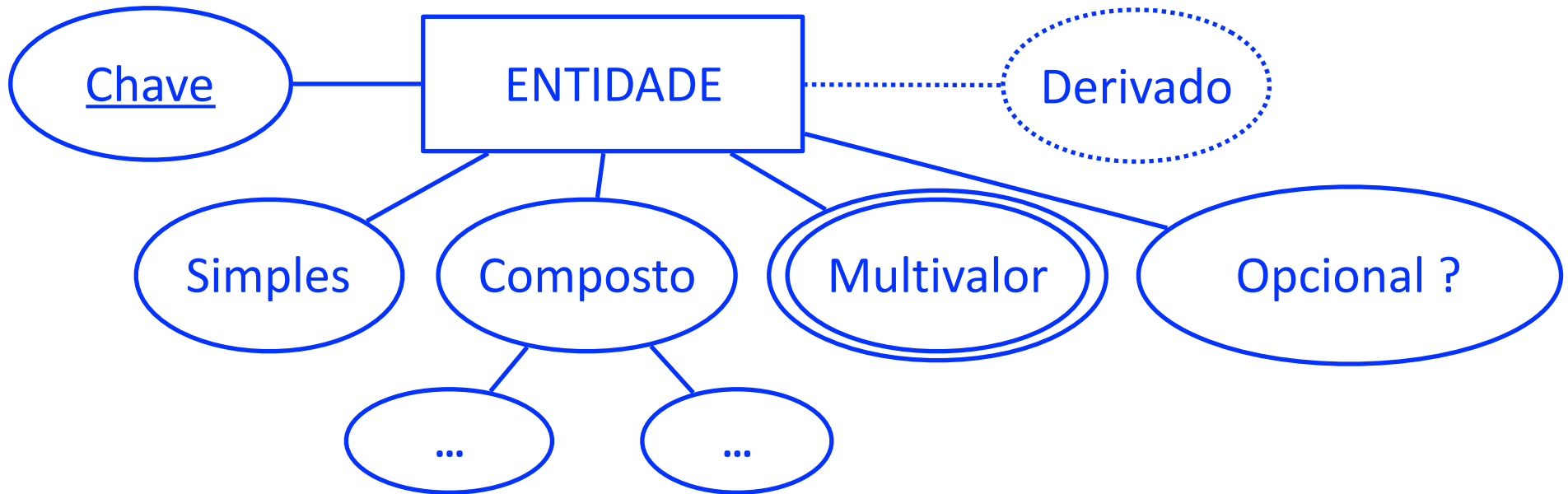


Domínios de atributos

- O **domínio de um atributo** é o conjunto de valores que um atributo pode tomar.
- Dependendo do domínio em causa o valor de um atributo pode ser:
 - **Definido** por valor concreto OU possivelmente **indefinido** se opcional para a entidade, designado nesse caso por **NULL**.
 - **Simples** OU **composto** por vários sub-atributos.
 - **Derivado** se derivado do valor de outros atributos ou informação no modelo.
 - Ter **valor único** OU ser **multi-valor** (conjunto de valores)
 - **Complexo** se formado pela combinação de vários atributos multi-valor e/ou compostos.

Entidades – sintaxe adicional

Sintaxe visual



Sintaxe textual

ENTIDADE(Chave, Simples, Composto(Sub-atributos),
{ Multivalor }, Opcional?, [Derivado])

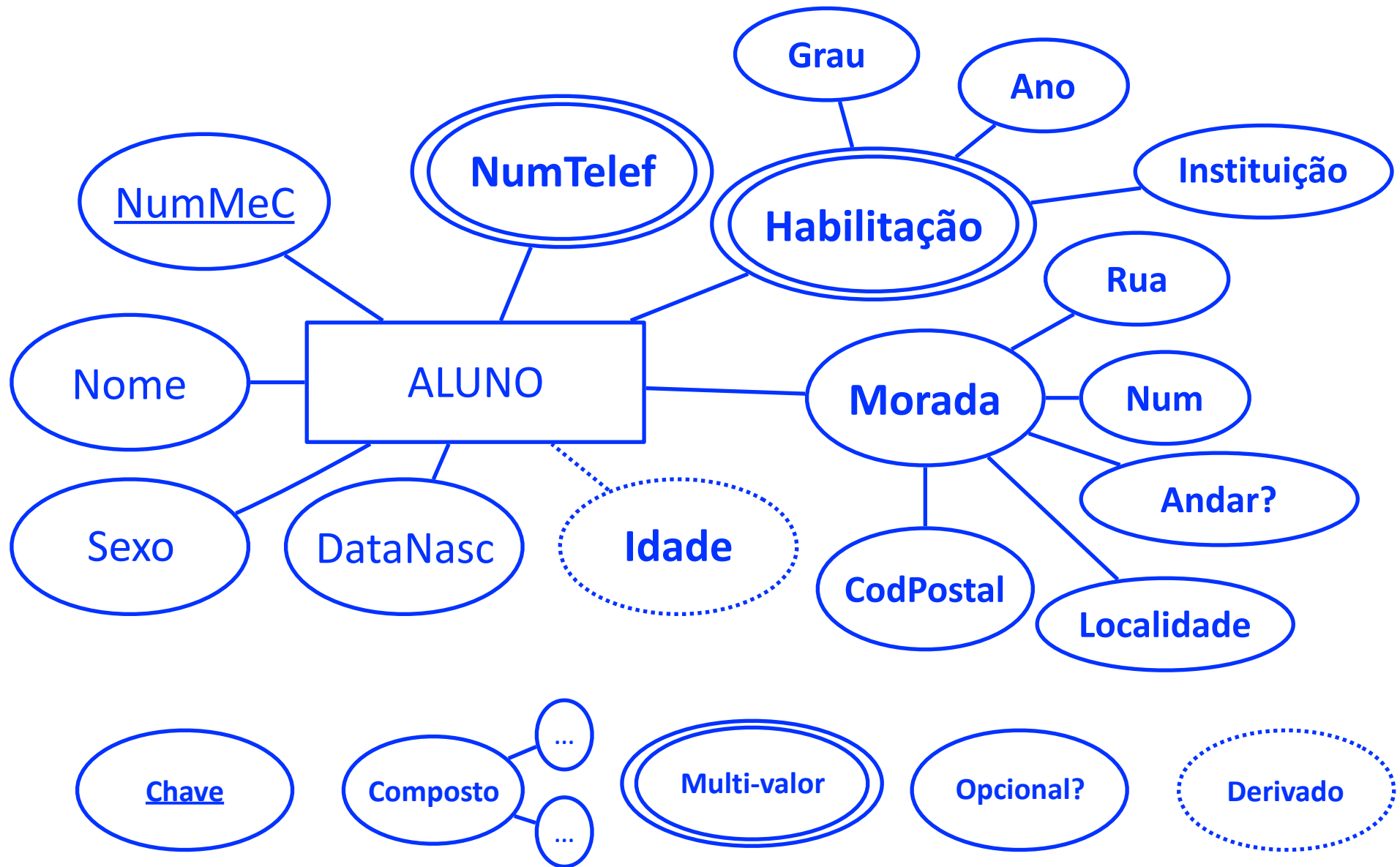
Exemplo

■ Para

`ALUNO (NumMec, Nome, Sexo, DataNasc, [Idade],
Morada(Rua,Num,Andar?,Localidade,CodPostal), {NumTelefone},
{Habilitação(Grau,Ano, Instituição)})`

- **Idade** é um atributo **derivado** de **DataNasc**
- **Morada** é um atributo **composto** por sub-atributos **Rua, Num, Andar, Localidade, CodPostal**. Numa modelação alternativa poderíamos também decompor **DataNasc** em dia, mês e ano (usualmente datas são no entanto tratadas como valores simples) ou **Nome** em nome principal e apelidos.
- **Andar** é um sub-atributo opcional de **Morada**.
- **{NumTelefone}** é **multi-valor**: considera-se que um aluno pode ter mais do que um n.º de telefone (ex. n.º telefone fixo, n.º telemóvel).
- **{Habilitação(Grau,Ano,Instituição)}** é um atributo **complexo**, pois trata-se de um atributo multi-valor em que cada valor é por sua vez composto. Um aluno pode ter várias habilitações académicas, cada uma caracterizado por um grau, ano e instituição.

Exemplo (cont.)



Domínios de atributos (cont.)

Para

`Aluno(NumMeC, Nome, Sexo, DataNasc, Idade,
Morada(Rua,Num,Andar?,Localidade,CodPostal), {NumTelefone},
{Habilitação(Grau, Ano, Instituição)})`

podemos ter os seguintes exemplos de entidades:

`ALUNO1(19428771, 'José Silva', M, 17-11-2000, 18, ('Rua Fim do Mundo', 783,
'R/C', 'Finisterra', '4444-555'), {987654321, 222333444}, { ('Ens. Secundário',
2017, 'Escola Sec. Dr. Estranho-Amor') } }`

`ALUNO2(10447777, 'Maria Carvalho', F, 08-02-1976, 43, ('Rua das Bases de
Dados', 1555, NULL, 'Vila Nova de Informática', '4000-123'), {933933933},
{ ('Ens. Secundário', 1994, 'Escola Sec. Vila Nova de Informática'), ('Lic. Física',
1998, 'Fac. de Ciências Univ. Porto') } }`

Domínios de atributos (cont.)

ALUNO₁(19428771, 'José Silva', M, 17-11-2000, 18, ('Rua Fim do Mundo', 783, 'R/C', 'Finisterra', '4444-555'), {987654321, 222333444}, { ('Ens. Secundário', 2017, 'Escola Sec. Dr. Estranho-Amor') } }

- sub-atributo Andar definido
- mais do que um nº de telefone
- uma só habilitação

- sub-atributo Andar = NULL
- um só nº de telefone
- mais do que uma habilitação

ALUNO₂(10447777, 'Maria Carvalho', F, 08-02-1976, 43, ('Rua das Bases de Dados', 1555, NULL, 'Vila Nova de Informática', '4000-123'), {933933933}, { ('Ens. Secundário', 1994, 'Escola Sec. Vila Nova de Informática'), ('Lic. Física', 1998, 'Fac. de Ciências Univ. Porto') } }

Relacionamentos

Relacionamento – forma geral

- Um **relacionamento** exprime uma interação conceptual entre entidades.
- Forma geral:
NOME(ENTIDADE-TIPO₁, ... , ENTIDADE-TIPO_N, Atributo₁, ..., Atributo_k)
- **NOME** é o **nome** do relacionamento, por convenção predicado que faz uso de verbos/adjectivos e qualificadores (ex. **FILHO_DE**, **TRABALHA_PARA**).
- **ENTIDADE-TIPO₁, ... , ENTIDADE-TIPO_N** são as **entidades-tipo participantes**.
- **N** : **grau do relacionamento** (número de participantes)
- Um relacionamento pode (opcionalmente) ter também associados **atributos** **Atributo₁, ..., Atributo_k**

Relacionamentos – exemplo simples

- No universo (simplificado) de uma faculdade considerem-se as entidades-tipo

ALUNO(NumMec, Nome, Sexo, DataNasc, ...)

CURSO(IdCurso, Nome, ...)

DEPARTAMENTO(IdDep, Nome, ...)

PROFESSOR(NumFunc, Nome, ...)

que podem relacionar-se de várias formas, por ex. (entre várias outras possíveis):

RESPONSAVEL_POR(DEPARTAMENTO, CURSO)

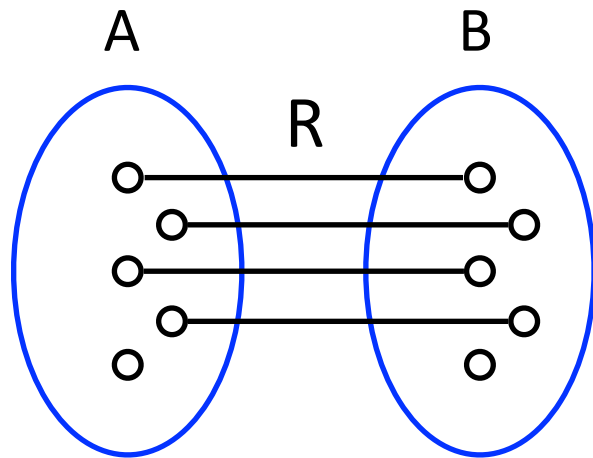
DIRECTOR_DE(PROFESSOR, DEPARTAMENTO)

TRABALHA_EM(PROFESSOR, DEPARTAMENTO)

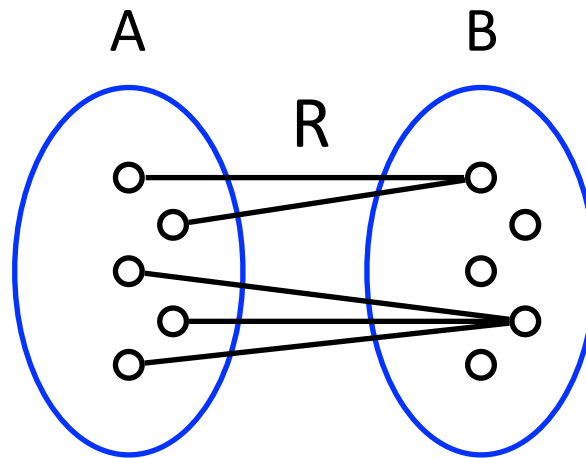
INSCRITO_EM(ALUNO, CURSO, AnoInsc, AnoConcl?)

- Os relacionamentos acima dizem-se **binários** porque associam duas entidades (têm grau 2). **INSCRITO_EM** tem também atributos.
- Veremos depois exemplos de relacionamentos de grau superior a 2, mas na maior parte dos casos relacionamentos binários são suficientes para a modelação de uma BD.

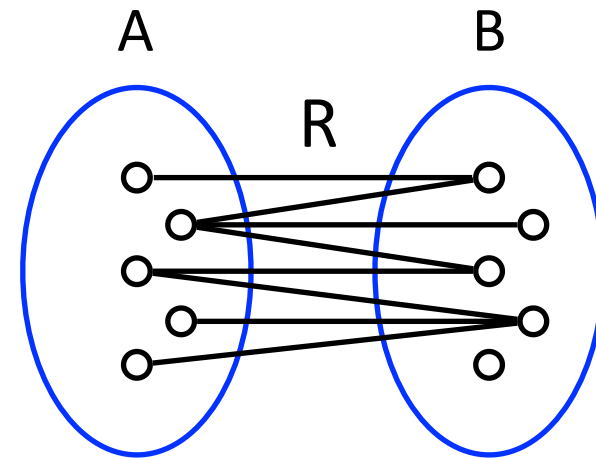
Restrições de cardinalidade



1:1



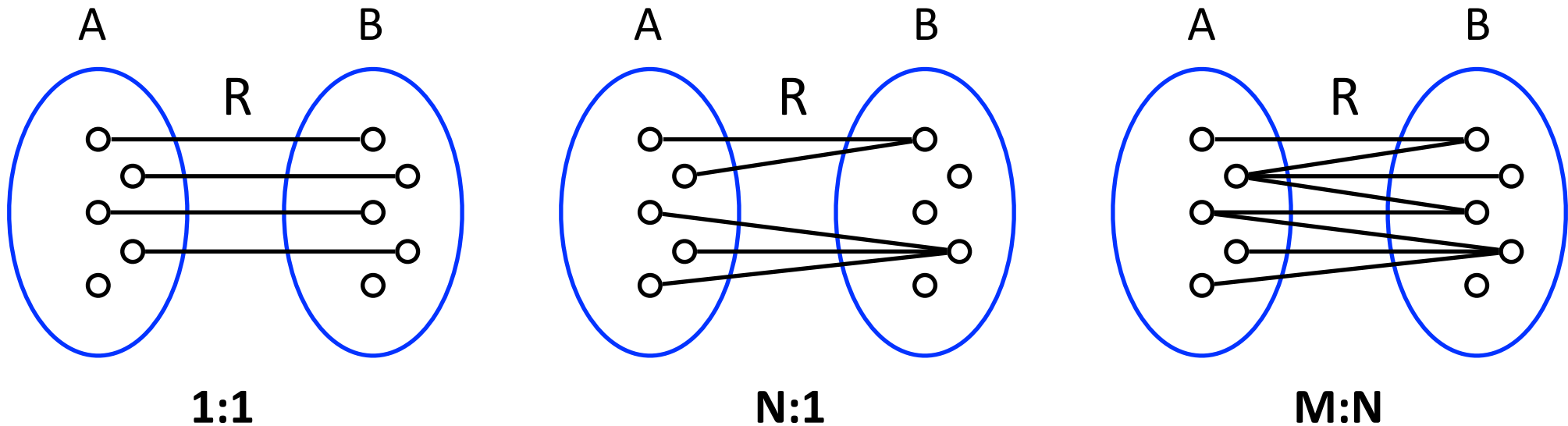
N:1



M:N

- A um relacionamento binário podemos ter associadas restrições de **cardinalidade**:
 - 1:1 — um-para-um
 - 1:N N:1 — um-para-muitos, muitos-para-um
 - M:N — muitos-para-muitos

Restrições de cardinalidade (cont.)



- Nos exemplos anteriores:

`DIRECTOR_DE(PROFESSOR, DEPARTAMENTO)` **1:1**

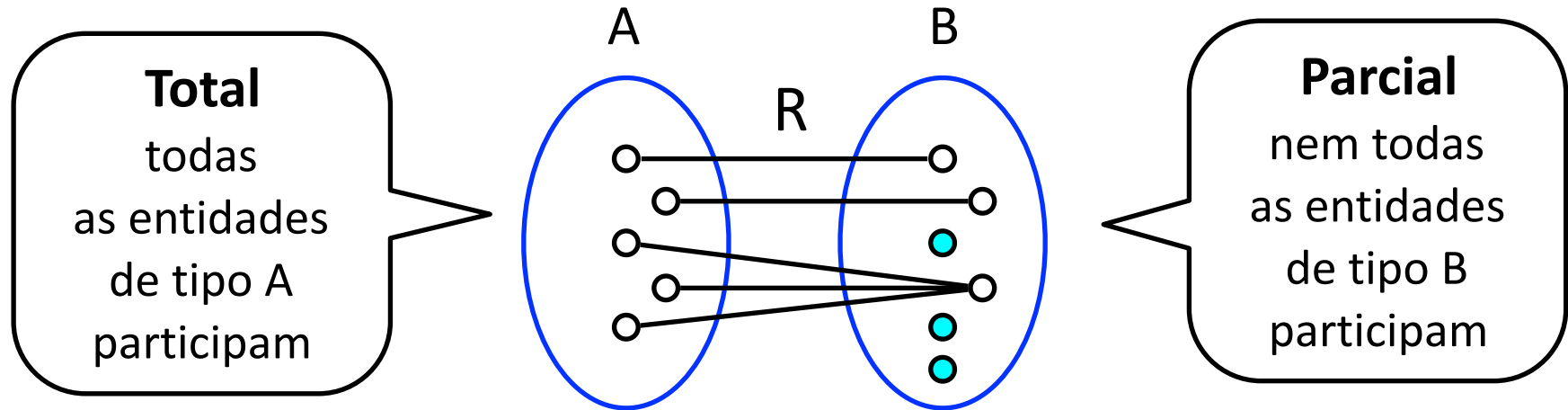
`TRABALHA_EM(PROFESSOR, DEPARTAMENTO)` **N:1**

`RESPONSAVEL_POR(DEPARTAMENTO, CURSO)` **1:N**

`INSCRITO_EM(ALUNO, CURSO, AnoInscrição, AnoConclusão?)` **M:N**

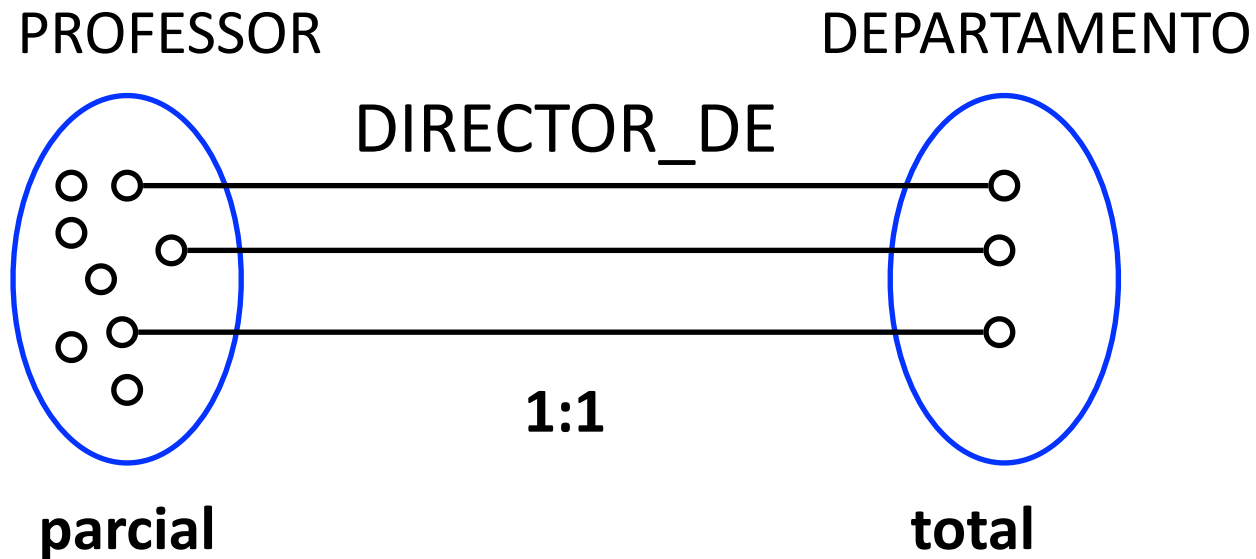
(obs.: um aluno pode fazer vários cursos: Lic. CC, Mestrado DS, ...)

Restrições de participação



- A um relacionamento podemos ter também associadas restrições de **participação**. A participação é **total** para uma entidade-tipo se a existência de uma entidade desse tipo obriga que a que participe no relacionamento, e **parcial** caso contrário.

Exemplos



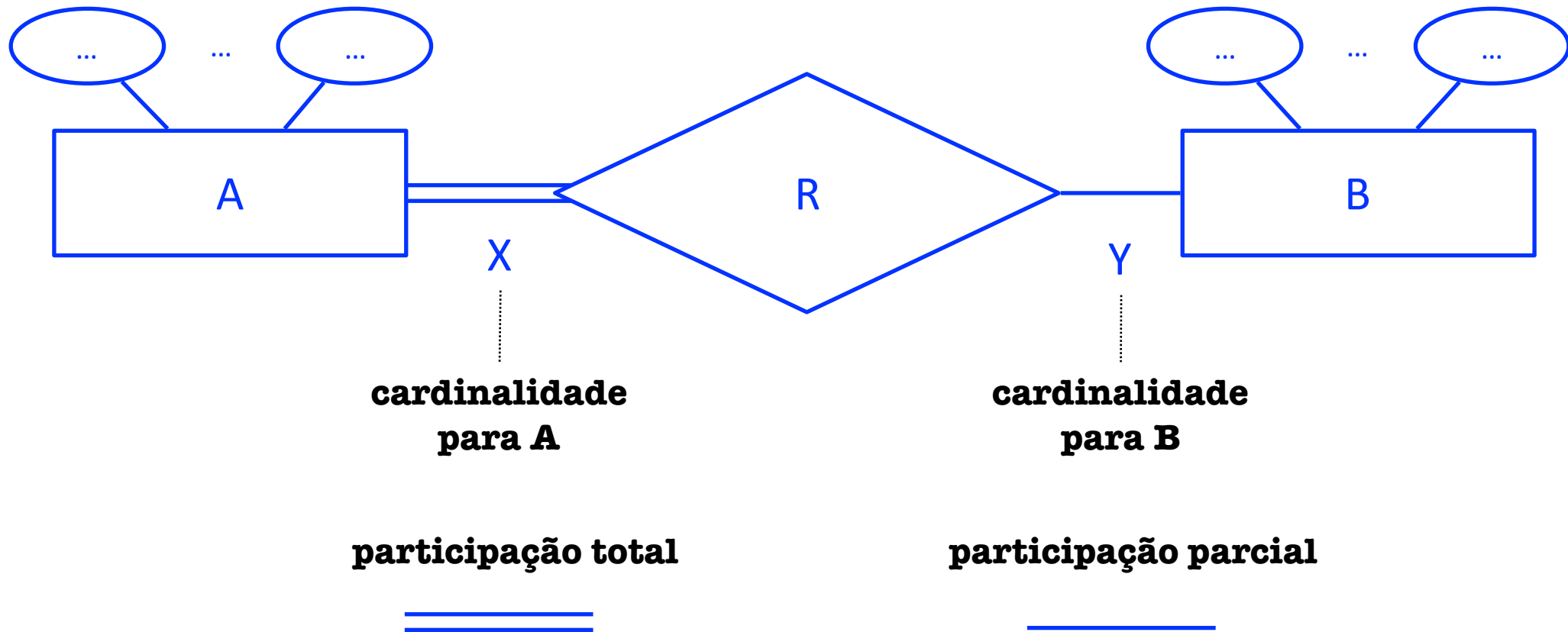
Só alguns
professores
são directores
de curso.

Todo o
departamento
tem um
director.

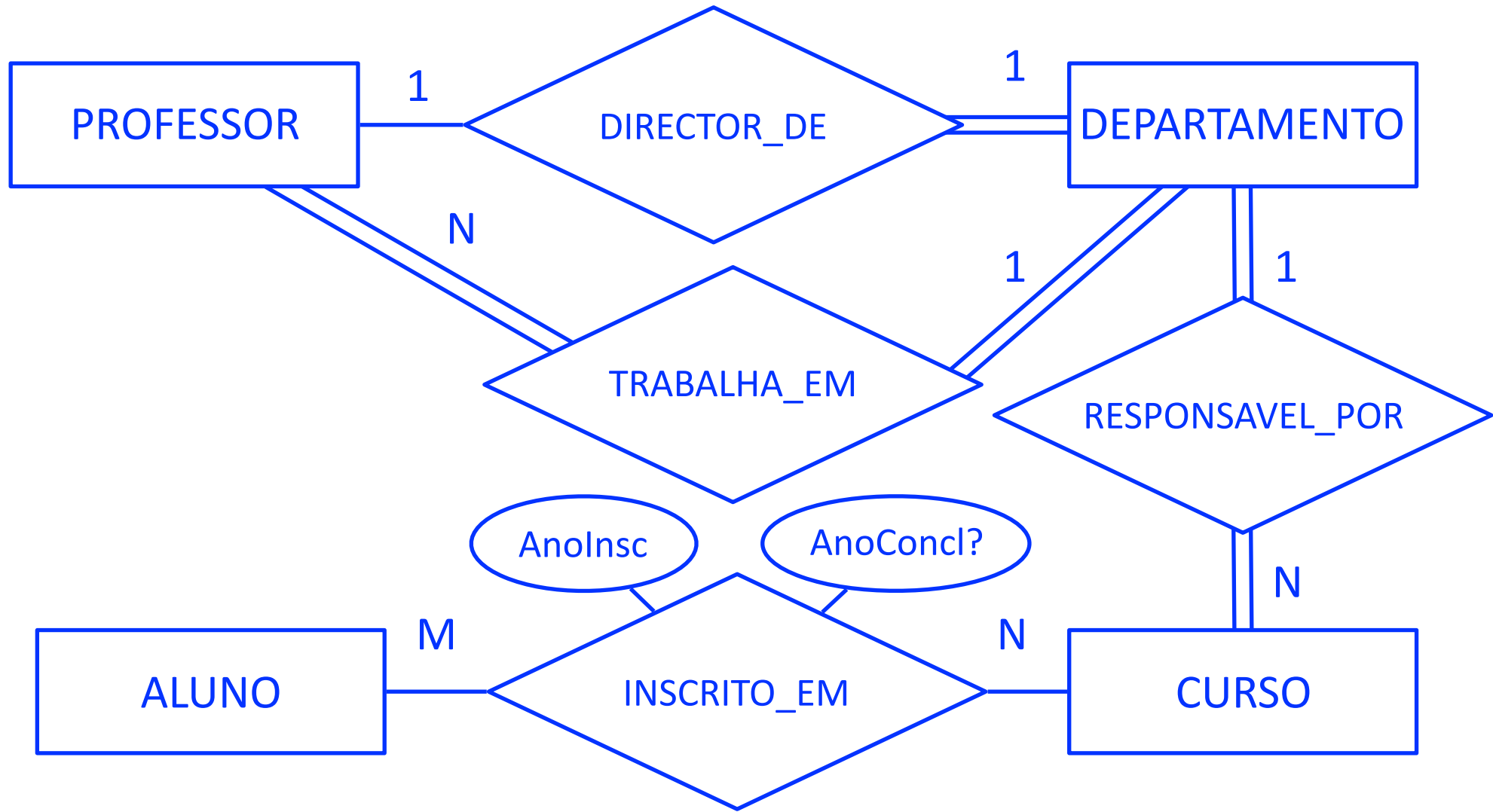
Exemplos

- **DIRECTOR_DE**(PROFESSOR, DEPARTAMENTO) — **parcial** para **PROFESSOR** (nem todos os professores são directores de curso), **total** para **DEPARTAMENTO** (todo o departamento tem um director).
- **TRABALHA_EM**(PROFESSOR, DEPARTAMENTO) — **total para as 2 entidades** assumindo que todo o professor está associado a um departamento e que um departamento tem necessariamente professores associados.
- **RESPONSAVEL_POR**(DEPARTAMENTO, CURSO) — **total para as 2 entidades** analogamente ao caso anterior (assumindo de forma razoável que cada departamento é responsável por pelo menos um curso).
- **INSCRITO_EM**(ALUNO, CURSO, AnoInscrição, AnoConclusão?) — **parcial para ambas as entidades** assumindo que poderão haver alunos (ainda) não inscritos num curso e que um curso pode (ainda) não ter tido alunos (ex. programas de doutoramento novos).

Relacionamentos – diagramas ER



Exemplos anteriores – diagrama ER



Nota: por simplicidade omitimos os atributos das entidades-tipo.

De requisitos a modelo ER

De requisitos a modelo ER

- A modelação de uma BD é o resultado da análise de requisitos para o universo em causa.
- **Requisitos $\mapsto$ Modelo ER**
 - Os requisitos devem ser identificados para o universo em causa, de forma tão clara quanto possível
 - O modelo ER resulta da interpretação (correcta) dos requisitos.
- Caso-de-estudo a seguir:
 - BD para uma empresa, com algumas semelhanças ao exemplo da faculdade que temos vindo a considerar.
 - Adaptado de “Fundamentals of Database Systems, 7th ed”, capítulo 3.

BD empresa

- Consideremos o universo de uma empresa em que temos as seguintes **entidades-tipo**, respectivos **atributos** e **relacionamentos implícitos**):
 - **FUNCIONÁRIO** com os seguintes atributos: n.º de CC, nome, email opcional, data de nascimento, salário, horas semanais de dedicação a projectos, **funcionário supervisor opcional**, e departamento definido a que pertence.
 - **DEPARTAMENTO**: com nome único, várias localizações possíveis associadas (moradas como anteriormente), um funcionário gestor, e um ou mais funcionários que trabalham para o departamento. Um funcionário pode ser gestor de apenas um departamento.
 - **PROJECTO**: com nome único, data de início, data de conclusão, departamento definido que controla o projecto, e funcionário definido para director do projecto. Poderão haver departamentos sem projectos associados.
 - Um funcionário trabalha em um ou mais projectos, sendo necessário identificar o número de horas semanais que cada funcionário dedica a cada projecto. Um projecto tem sempre funcionários que trabalham nele (além do director).

BD empresa — apenas entidades-tipo ?

Podemos tentar exprimir o universo anterior usando apenas entidades-tipo

FUNCIONÁRIO(NumCC, Nome, Email?, DataNasc, Salário,,
HorasProj, Supervisor?, Departamento)

DEPARTAMENTO(Nome, Gerente,
{ Localização(Rua, Num, Andar?, Localidade, CodPostal)})

PROJECTO(Nome, DataInício, DataFim, Departamento, Director,
{ Trabalho(Funcionário, Horas) })

mas ao termos “**referências**” entre entidades, a modelação usando apenas entidade-tipos não é adequada. Devemos usar ao invés relacionamentos. Que relacionamentos estão implícitos?

BD empresa – reformulação

Reformulando, podemos ter como entidades-tipo

FUNCIONÁRIO(NumCC, Nome, Email?, DataNasc, Salário, [HorasProj])

DEPARTAMENTO(Nome, Localização(Rua, Num, Andar?, Localidade, CodPostal))

PROJECTO(Nome, DataInício, DataFim)

e os seguintes relacionamentos

SUPERVISIONA(FUNCIONÁRIO, FUNCIONÁRIO)

TRABALHA_PARA(FUNCIONÁRIO, DEPARTAMENTO)

GERE(FUNCIONÁRIO, DEPARTAMENTO)

CONTROLA(DEPARTAMENTO, PROJECTO)

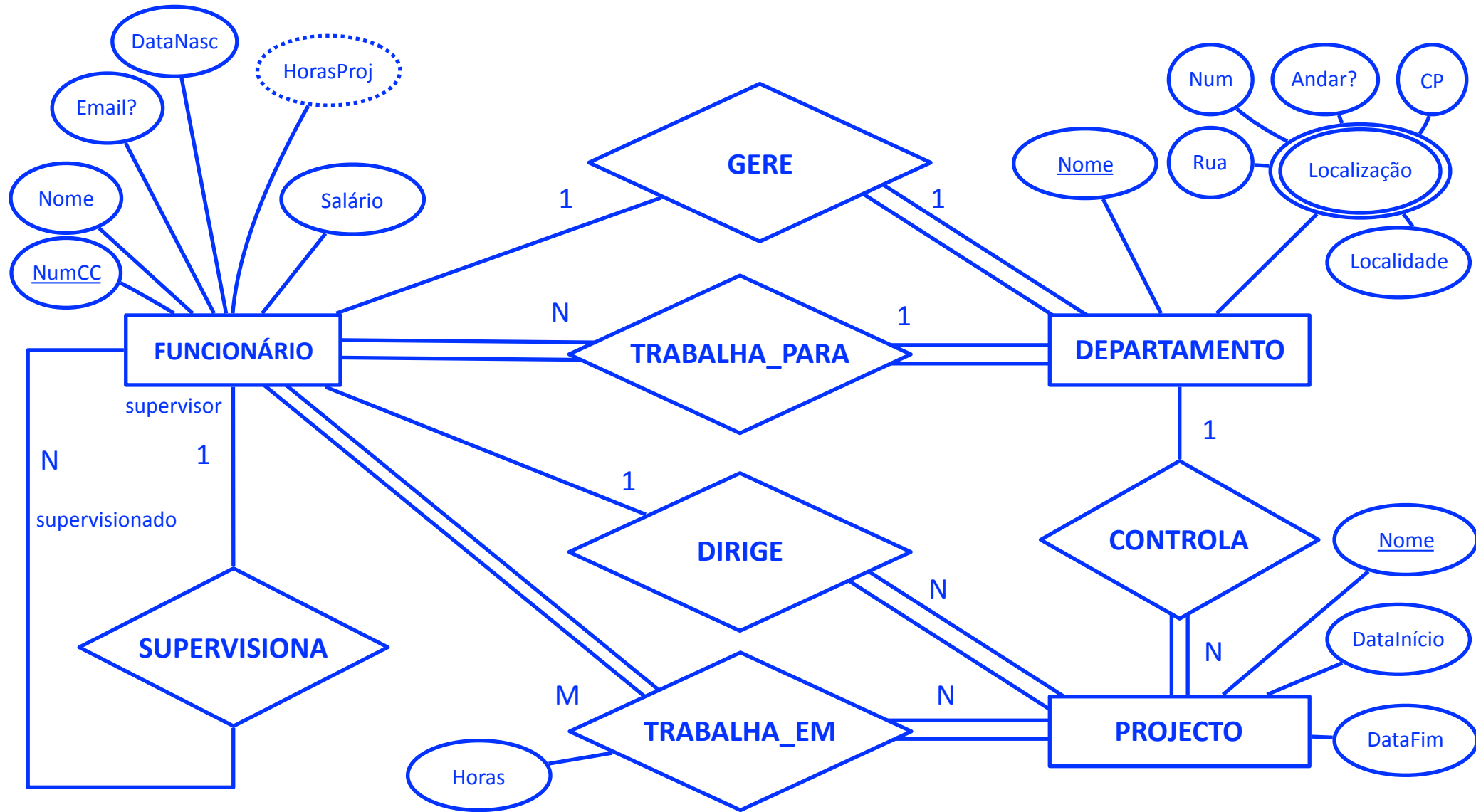
DIRIGE(FUNCIONÁRIO, PROJECTO)

TRABALHA_EM(FUNCIONÁRIO, PROJECTO, Horas)

BD empresa – cardinalidade e participação

Relacionamento	C	P
SUPERVISIONA (FUNCIONÁRIO, FUNCIONÁRIO)	1 <> N	parcial <> parcial
TRABALHA_PARA (FUNCIONÁRIO, DEPARTAMENTO)	N <> 1	total <> total
GERE (FUNCIONÁRIO, DEPARTAMENTO)	1 <> 1	parcial <> total
CONTROLA (DEPARTAMENTO, PROJECTO)	1 <> N	parcial <> total
DIRIGE (FUNCIONÁRIO, PROJECTO)	1 <> N	parcial <> total
TRABALHA_EM (FUNCIONÁRIO, PROJECTO, Horas)	M <> N	total <> total

BD empresa – diagrama ER



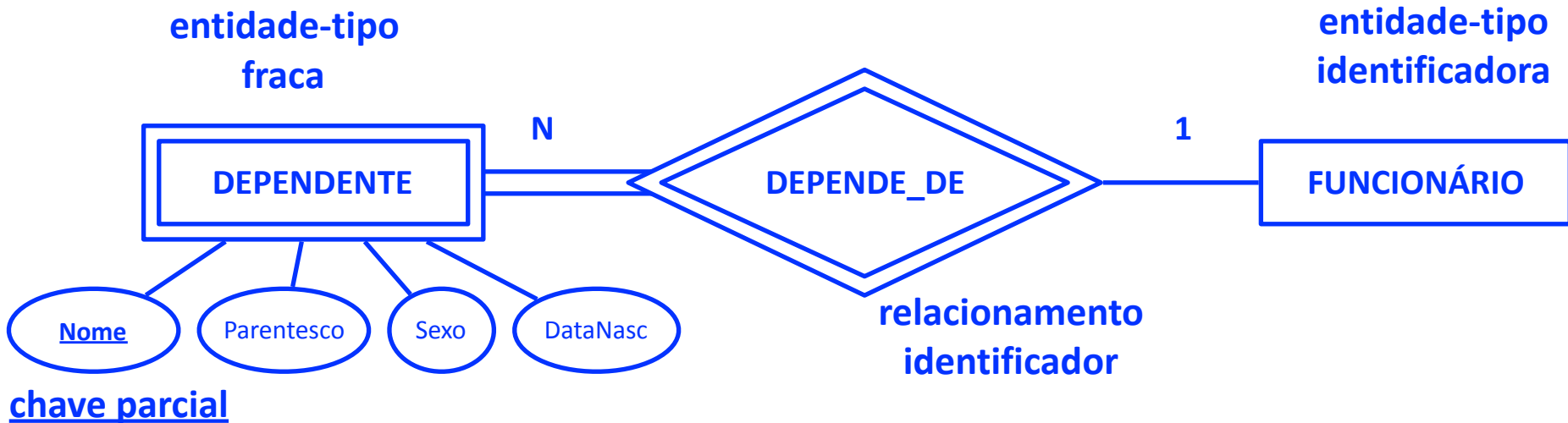
Aspectos complementares

Entidades-tipo fracas



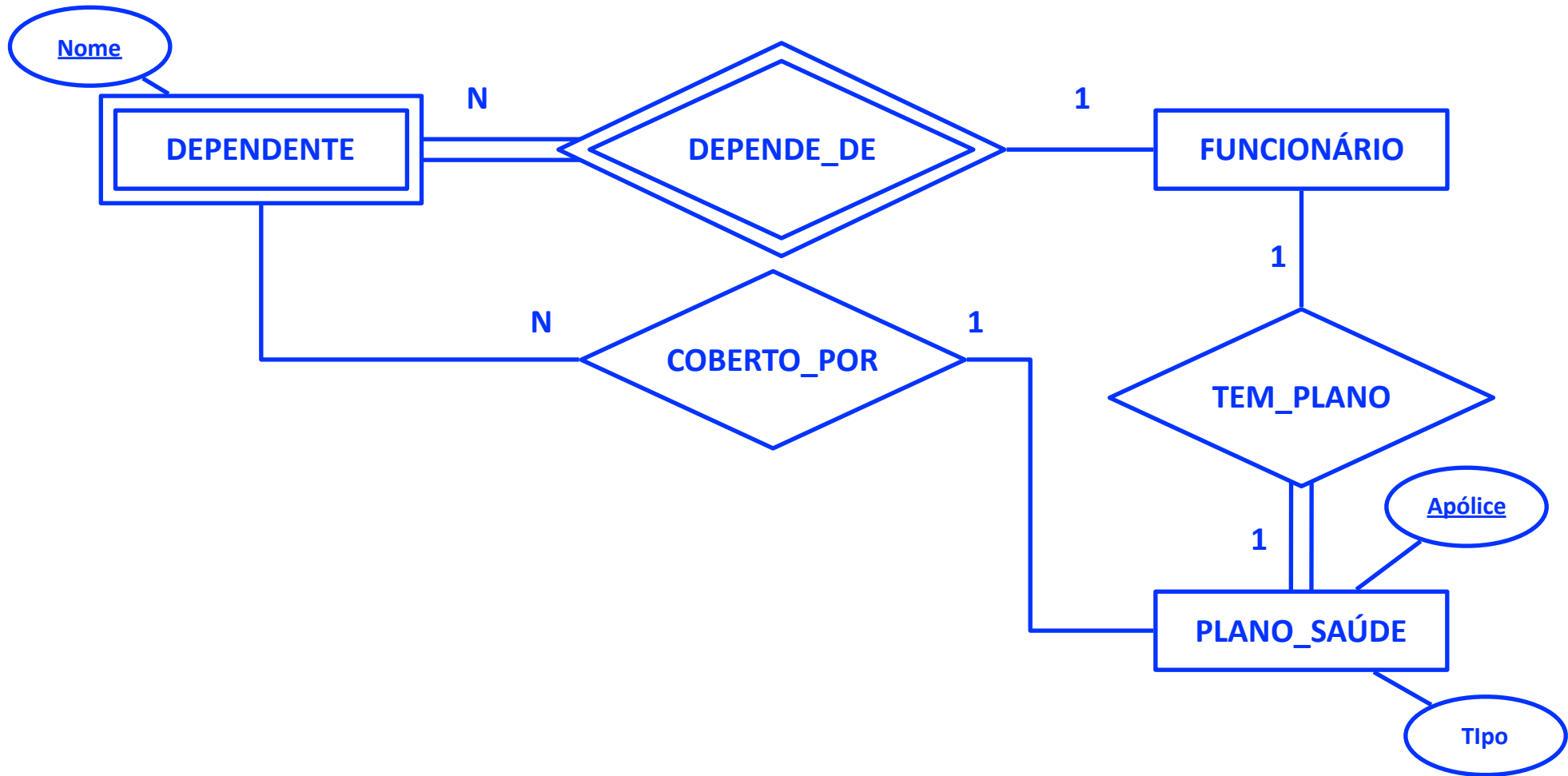
- são assim chamadas as entidades-tipo sem chave
- **dependência existencial: uma entidade fraca** — instância de entidade-tipo fraca — **está necessariamente ligada a uma única instância de uma entidade-tipo identificadora**, por meio de um **relacionamento identificador N:1 com participação total da entidade-tipo fraca** (entidade-tipo identificadora pode ter participação total ou parcial).
- entidades fracas podem ter uma **chave parcial**, que permita distinguir as instâncias da entidade-tipo fraca **apenas** no contexto da entidade identificadora

Entidades-tipo fracas – exemplo (cont.)



- No contexto da BD empresa poderíamos considerar a noção de pessoa dependente (parte do núcleo familiar) de um funcionário.
- **Nome** é a chave parcial de **DEPENDENTE**: os dependentes de um funcionário distinguem-se entre si pelo nome. Podem existir dependentes com o mesmo nome para funcionários distintos.

Entidades-tipo fracas (cont.)



- Entidades-tipo fraca podem envolver-se em outros relacionamentos para além do relacionamento identificador.

Relacionamentos de grau superior a 2

- Até agora vimos apenas relacionamentos de grau 2 (binários), os mais comuns, o nosso foco durante o semestre. Fica no entanto a nota que podemos ter relacionamentos de grau superior.
- No exemplo de uma empresa considere o fornecimento de produtos no contexto de um projecto em que:
 - estão envolvidas **3 entidades**: o **projecto**, o **produto** fornecido, e o **fornecedor** do produto;
 - um fornecimento é caracterizado por uma certa quantidade do produto em causa.

Relacionamentos de grau superior a 2 (cont.)

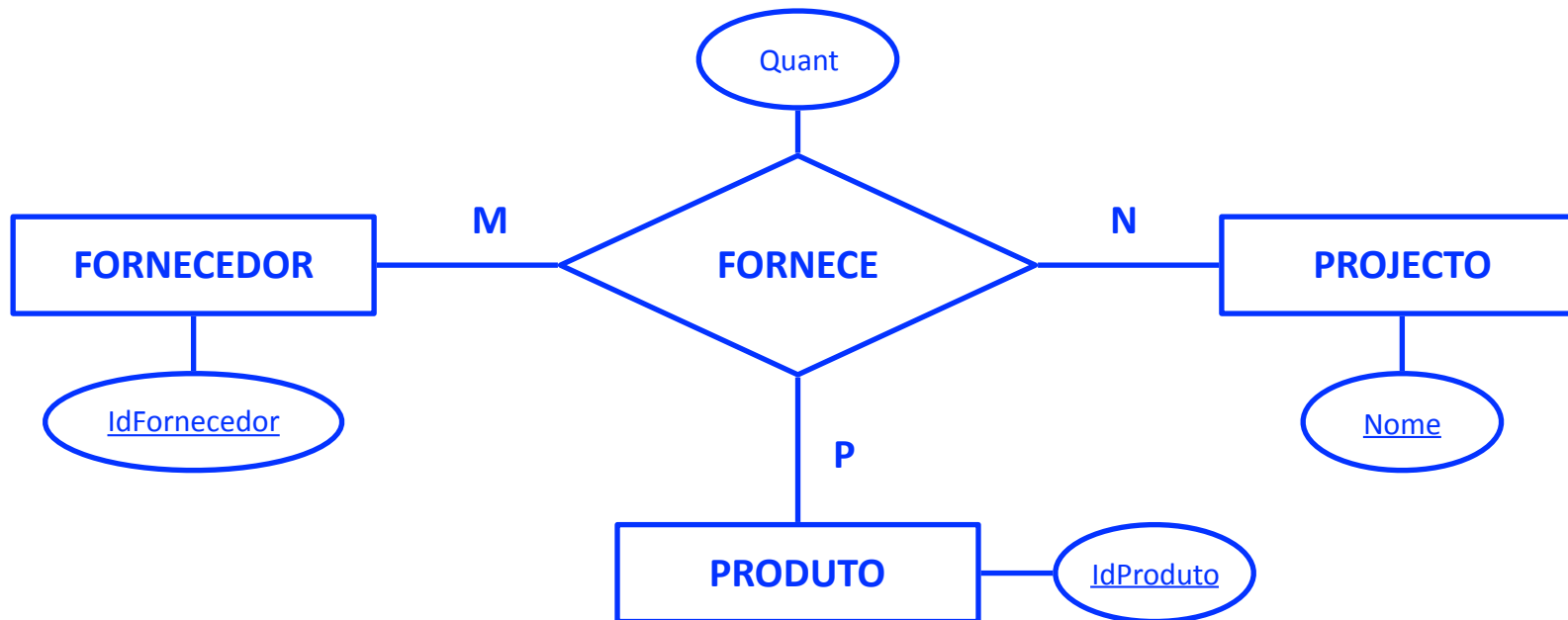
Para o exemplo anterior, assumindo que temos as entidade-tipos adicionais no modelo da empresa:

PRODUTO(IdProduto, ...)

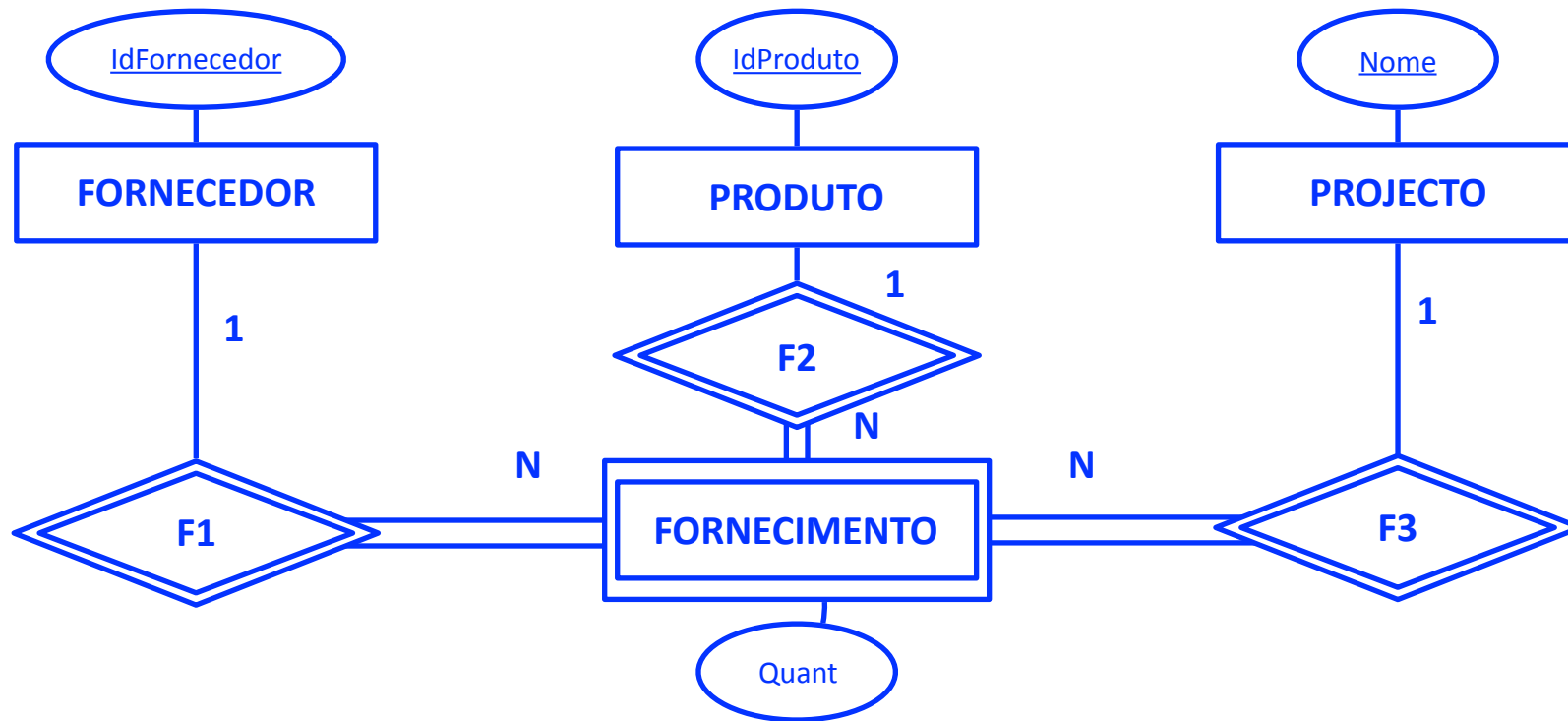
FORNECEDOR(IdFornecedor, ...)

podemos ter a relação ternária:

FORNECE(FORNECEDOR, PRODUTO, PROJECTO, Quant)



Relacionamentos de grau superior a 2 (cont.)



- Em alguns casos como no exemplo, podemos modelar o equivalente a um relacionamento de grau 3 (ou superior) recorrendo a entidades-tipo fracas com mais de uma entidade-tipo identificadora (no exemplo 3).