

Regras de Transformação – EA para Relacional

Abordagens

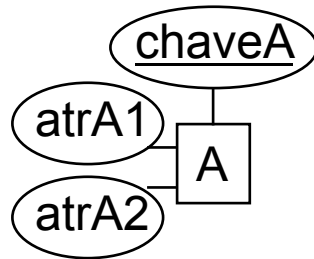
- Uma abordagem:
 - ◊ construir com base no Modelo Entidade - Associação um Esquema de Relação único (com todos os Atributos)
 - ◊ com base nas Dependências Funcionais extraídas da análise, normalizar esse Esquema de Relação até à Forma Normal desejada
- Outra abordagem:
 - ◊ aplicar um método sistemático de passagem do Modelo Entidade - Associação para Esquemas de Relação
 - ◊ verificar se os Esquemas se encontram na Forma Normal desejada e caso não estejam, normalizar até à Forma Normal desejada
- Vamos explorar esta última abordagem
 - ◊ para isso é necessário ter um método sistemático de passagem do Modelo Entidade - Associação para Esquemas de Relação

Regras gerais

- Cada Entidade corresponde a um Esquema de Relação com os mesmos Atributos e a mesma Chave
- Por cada Associação 1:1 um dos Esquemas de Relação deve ter como Chave Estrangeira a Chave do Esquema de Relação que representa a outra Entidade
- Por cada Associação 1:N o Esquema de Relação que representa a Entidade N deve ter como Chave Estrangeira a Chave do Esquema de Relação que representa a Entidade do lado 1
- Por cada Associação M:N deve ser criado um novo Esquema de Relação que tem como Chave, a Chave de cada uma das Entidades envolvidas na Associação
- Estas regras são gerais podendo haver interesse em adoptar regras específicas para determinadas situações

Entidade

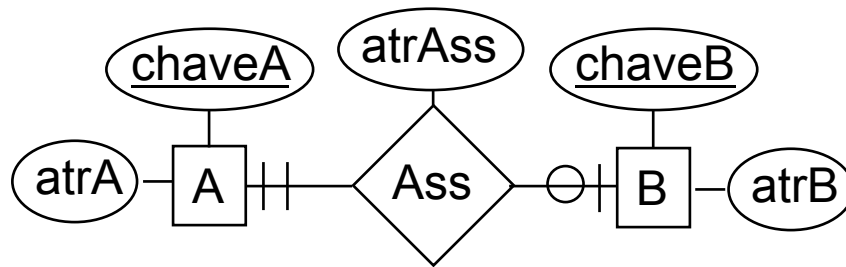
- Cada Entidade corresponde a um Esquema de Relação com os mesmos Atributos e a mesma Chave



◇ A(chaveA, atrA1, atrA2)
· Chaves Candidatas = { chaveA }

Associação binária 1:1

- Obrigatória apenas num dos lados



◇ A(chaveA, atrA)

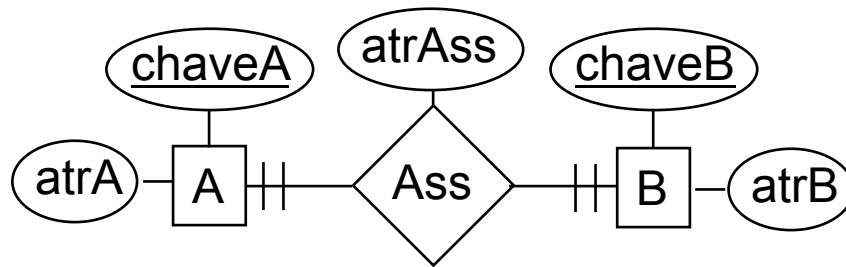
- Chaves Candidatas = { chaveA }

◇ B(chaveB, atrB, atrAss, chaveA)

- Chaves Candidatas = { chaveB } e { chaveA }
- Chaves Estrangeiras = { chaveA }

Associação binária 1:1 (cont. 1)

- Obrigatória em ambos os lados



◇ A(chaveA, atrA, atrAss, chaveB, atrB)

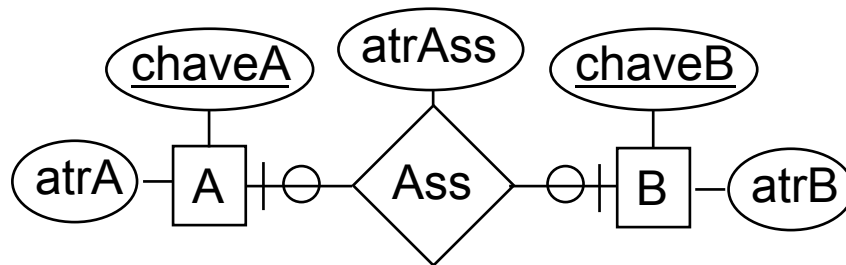
- Chaves Candidatas = { chaveA } e { chaveB }

- Esta solução é bastante limitativa

◇ A (ou B) podem, por si só, participar em alguma Associação com outra Entidade

Associação binária 1:1 (cont. 2)

- Nenhum dos lados é obrigatório



◇ A(chaveA, atrA)

- Chaves Candidatas = { chaveA }

◇ B(chaveB, atrB)

- Chaves Candidatas = { chaveB }

◇ A_B (chaveA, atrAss, chaveB)

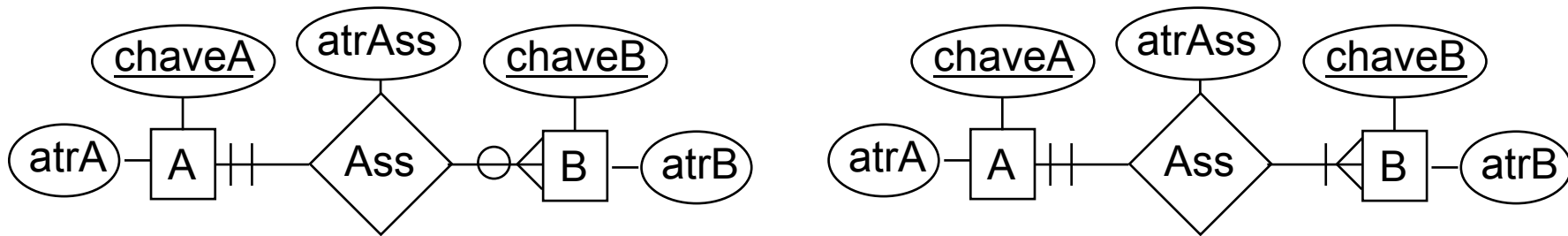
- Chaves Candidatas = { chaveA } e { chaveB }
- Chaves Estrangeiras = { chaveA } e { chaveB }

Associação binária 1:1 (cont. 3)

- Se nenhum dos lados da Associação é obrigatório foi visto que
 - ◇ a situação pode ser representada por três Esquemas de Relação
- No entanto,
 - ◇ a situação pode-se representar apenas por dois Esquemas de Relação.
 - ◇ será uma solução idêntica à da situação em que apenas um dos lados é obrigatório
- Para se optar por representar esta situação através de dois ou três Esquemas de Relação é necessário,
 - ◇ ter em conta a proporção entre o número de ocorrências da Associação e o número de ocorrências das Entidades que lhe estão associadas
 - ◇ se apenas uma percentagem reduzida das Entidades está de facto associada, pode-se optar por três Esquemas de Relação;
 - ◇ se a grande percentagem das Entidades está associada, a situação é idêntica aquela em que apenas um dos lados é obrigatório

Associação binária 1:N

- Obrigatória apenas do lado N (ou obrigatória em ambos os lados)



◇ A(chaveA, atrA)

- Chaves Candidatas = { chaveA }

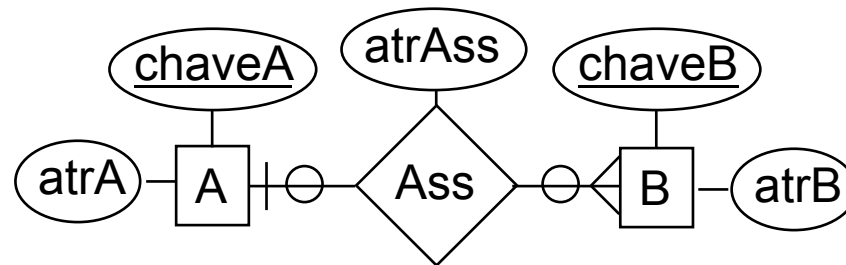
◇ B(chaveB, atrB, atrAss, chaveA)

- Chaves Candidatas = { chaveB }
- Chaves Estrangeiras = { chaveA }

- Notar que a garantia da participação obrigatória do lado 1 não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação binária 1:N (cont.)

- Nenhum dos lados é obrigatório



◇ A(chaveA, atrA)

- Chaves Candidatas = { chaveA }

◇ B(chaveB, atrB)

- Chaves Candidatas = { chaveB }

◇ A_B (chaveB, atrAss, chaveA)

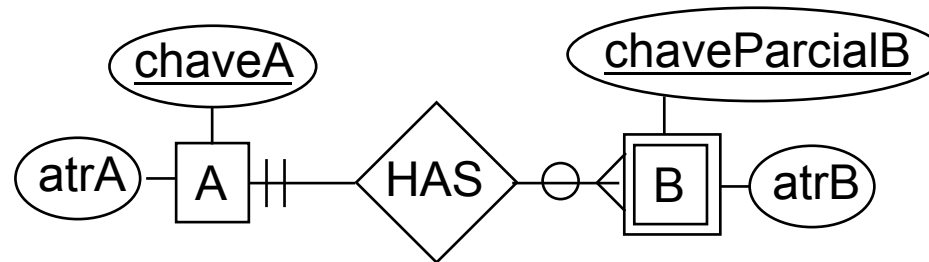
- Chaves Candidatas = { chaveB }
- Chaves Estrangeiras = { chaveB } e { chaveA }

Associação binária 1:N (cont. 1)

- Se nenhum dos lados da Associação é obrigatório foi visto que
 - ◇ a situação pode ser representada por três Esquemas de Relação
- No entanto,
 - ◇ a situação pode-se representar apenas por dois Esquemas de Relação.
 - ◇ será uma solução idêntica à da situação em que a Entidade do “lado N” é obrigatória
- Para se optar por representar esta situação através de dois ou três Esquemas de Relação é necessário,
 - ◇ ter em conta a proporção entre o número de ocorrências da Associação e o número de ocorrências da Entidade “do lado N”
 - ◇ se apenas uma percentagem reduzida das Entidades está de facto associada, pode-se optar por três Esquemas de Relação;
 - ◇ se a grande percentagem das Entidades está associada, a situação é idêntica aquela em que apenas o “lado N” é obrigatório

Entidade Fraca

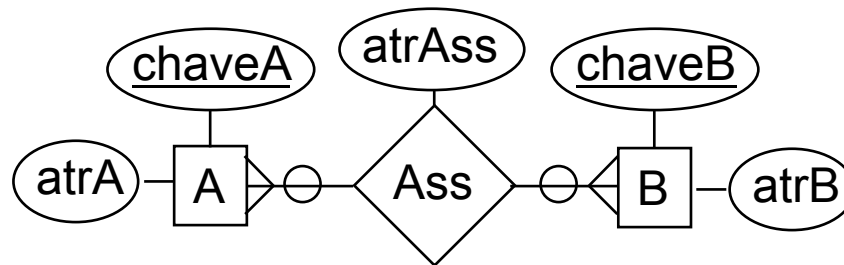
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ B(chaveParcialB, chaveA, atrB)
 - Chaves Candidatas = { chaveParcialB, chaveA }
 - Chaves Estrangeiras = { chaveA }

Associação binária M:N

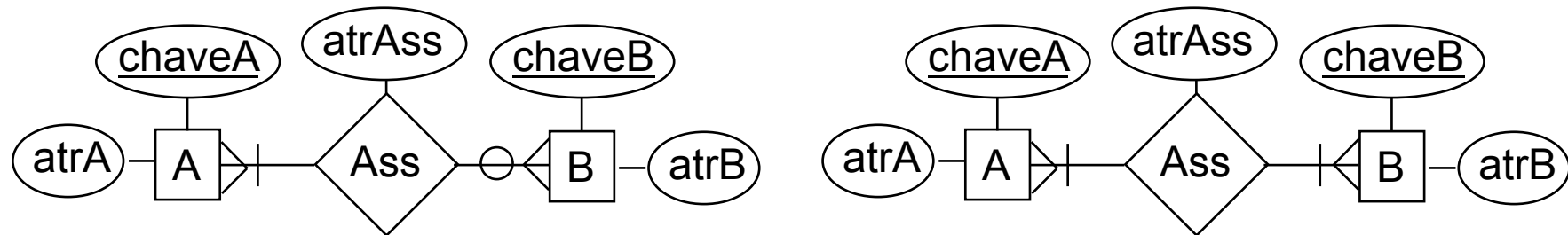
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ A_B (chaveA, chaveB, atrAss)
 - Chaves Candidatas = { chaveA, chaveB }
 - Chaves Estrangeiras = { chaveA } e { chaveB }

Associação binária M:N (cont.)

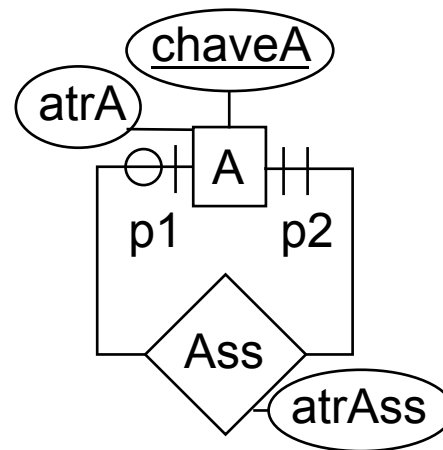
- Nos casos em que existe obrigatoriedade



- O Esquema Relacional é o mesmo que o do caso em que não existe obrigatoriedade em nenhum dos lados
- No entanto é preciso notar que a garantia da participação obrigatória não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação unária 1:1

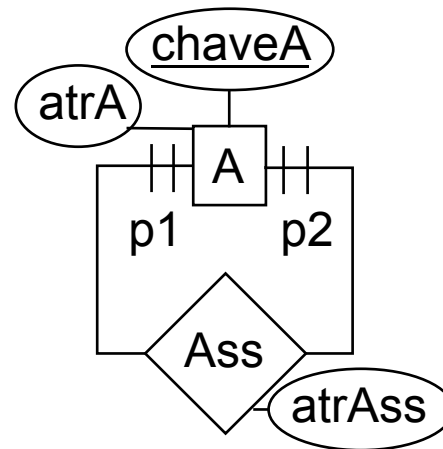
- Obrigatória apenas num dos lados



- ◇ $A(\text{chaveA_p1}, \text{atrA}, \text{atrAss}, \text{chaveA_p2})$
- Chaves Candidatas = { chaveA_p1 } e { chaveA_p2 }
 - Chaves Estrangeiras = a Chave Candidata que não for escolhida para Chave Primária
 - uma Entidade com o papel p2 que não esteja Associada a nenhuma outra com papel p1, terá o mesmo valor para os Atributos chaveA_p1 e chaveA_p2 e o Atributo atrAss poderá ser NULL

Associação unária 1:1 (cont.)

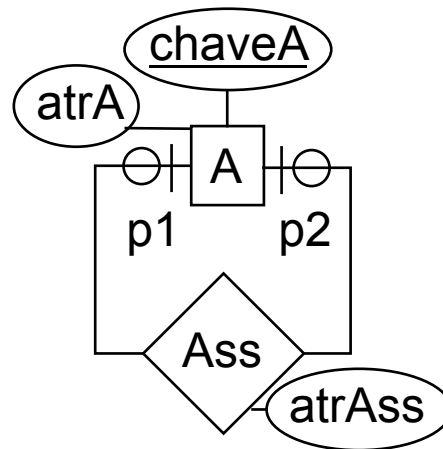
- Obrigatória em ambos os lados



- ◇ A(chaveA_p1, atrA_p1, atrAss, chaveA_p2, atrA_p2)
- Chaves Candidatas = { chaveA_p1 } e { chaveA_p2 }

Associação unária 1:1 (cont. 1)

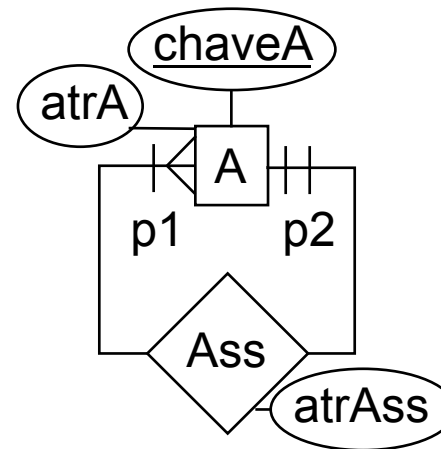
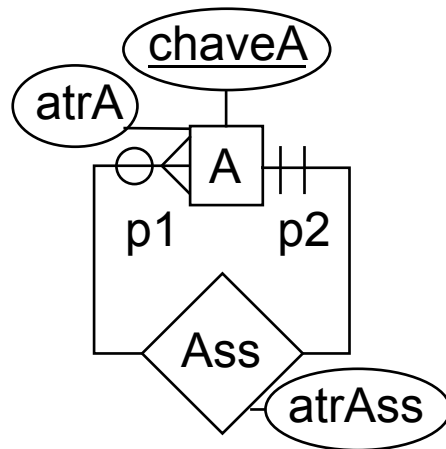
- Nenhum dos lados é obrigatório



- ◇ $A(\text{chaveA}, \text{atrA})$
 - Chaves Candidatas = { chaveA }
- ◇ $A_A(\text{chaveA_p1}, \text{chaveA_p2}, \text{atrAss})$
 - Chaves Candidatas = { chaveA_p1 } e { chaveA_p2 }
 - Chaves Estrangeiras = { chaveA_p1 } e { chaveA_p2 } (referem a Chave de A)

Associação unária 1:N

- Obrigatória apenas do lado N (ou obrigatória em ambos os lados)

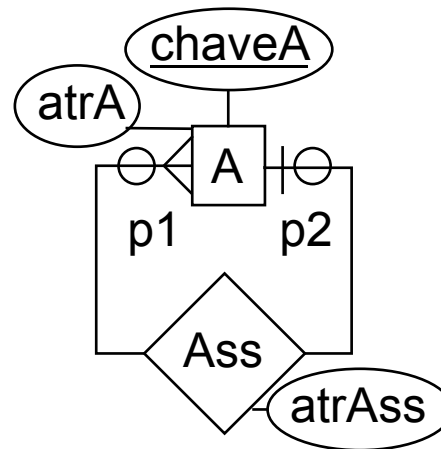


◇ A(chaveA_p1, atrA, atrAss, chaveA_p2)

- Chaves Candidatas = { chaveA_p1 }
 - Chaves Estrangeiras = { chaveA_p2 } (refere a Chave de A)
- Notar que a garantia da participação obrigatória do lado 1 não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação unária 1:N (cont.)

- Nenhum dos lados é obrigatório



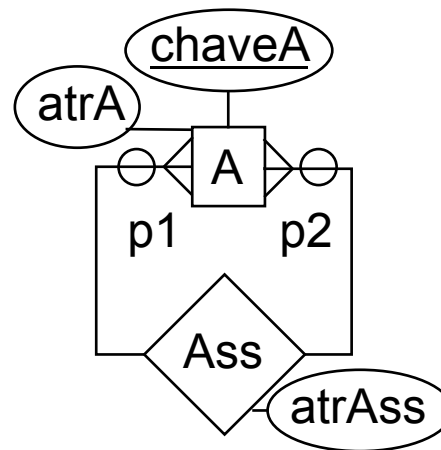
- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ A_A (chaveA_p1, atrAss, chaveA_p2)
 - Chaves Candidatas = { chaveA_p1 }
 - Chaves Estrangeiras = { chaveA_p1 } e { chaveA_p2 } (referem a Chave de A)

Associação unária 1:N (cont. 1)

- Se nenhum dos lados da Associação é obrigatório foi visto que
 - ◇ a situação pode ser representada por dois Esquemas de Relação
- No entanto,
 - ◇ a situação pode-se representar apenas por um Esquemas de Relação.
 - ◇ será uma solução idêntica à da situação em que a Entidade do “lado N” é obrigatória
- Para se optar por representar esta situação através de um ou dois Esquemas de Relação é necessário,
 - ◇ ter em conta a proporção entre o número de ocorrências da Associação e o número de ocorrências da Entidade “do lado N”
 - ◇ se apenas uma percentagem reduzida das Entidades está de facto associada, pode-se optar por dois Esquemas de Relação;
 - ◇ se a grande percentagem das Entidades está associada, a situação é idêntica aquela em que apenas o “lado N” é obrigatório

Associação unária M:N

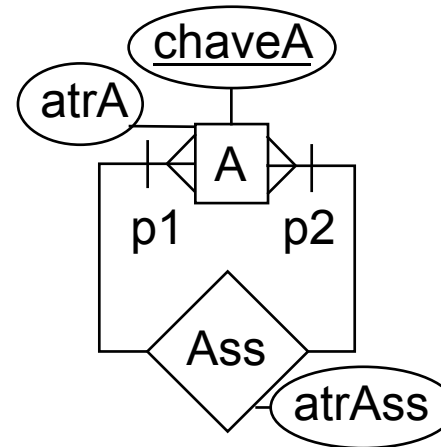
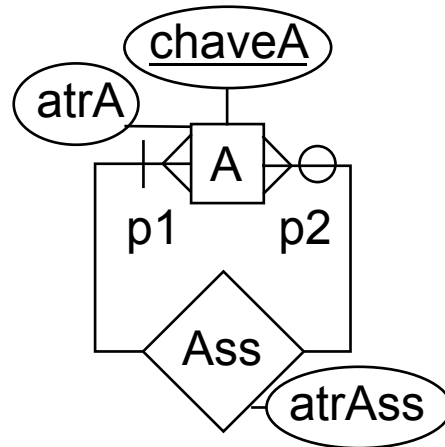
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ A_A (chaveA_p1, chaveA_p2, atrAss)
 - Chaves Candidatas = { chaveA_p1, chaveA_p2 }
 - Chaves Estrangeiras = { chaveA_p1 } e { chaveA_p2 } (referem a Chave de A)

Associação unária M:N (cont.)

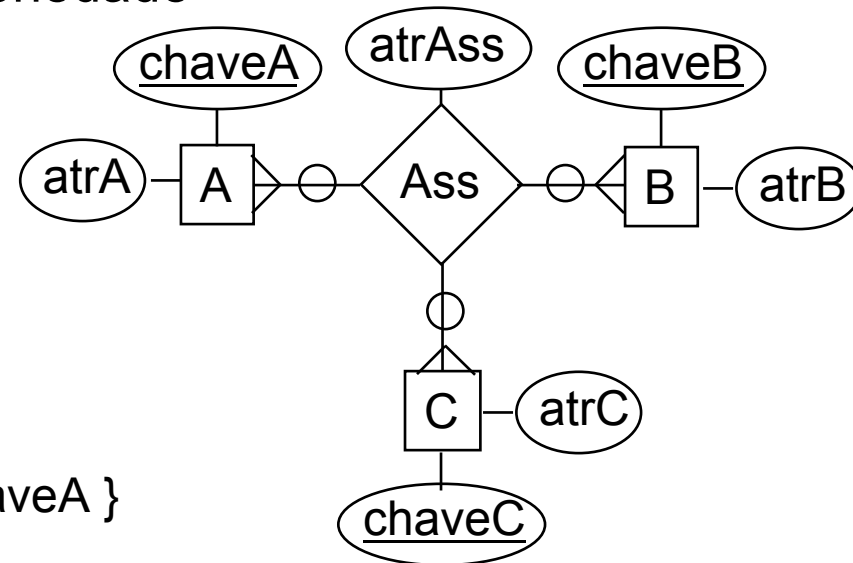
- Nos casos em que existe obrigatoriedade



- O Esquema Relacional é o mesmo que o do caso em que não existe obrigatoriedade em nenhum dos lados
- No entanto é preciso notar que a garantia da participação obrigatória não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação ternária M:N:P

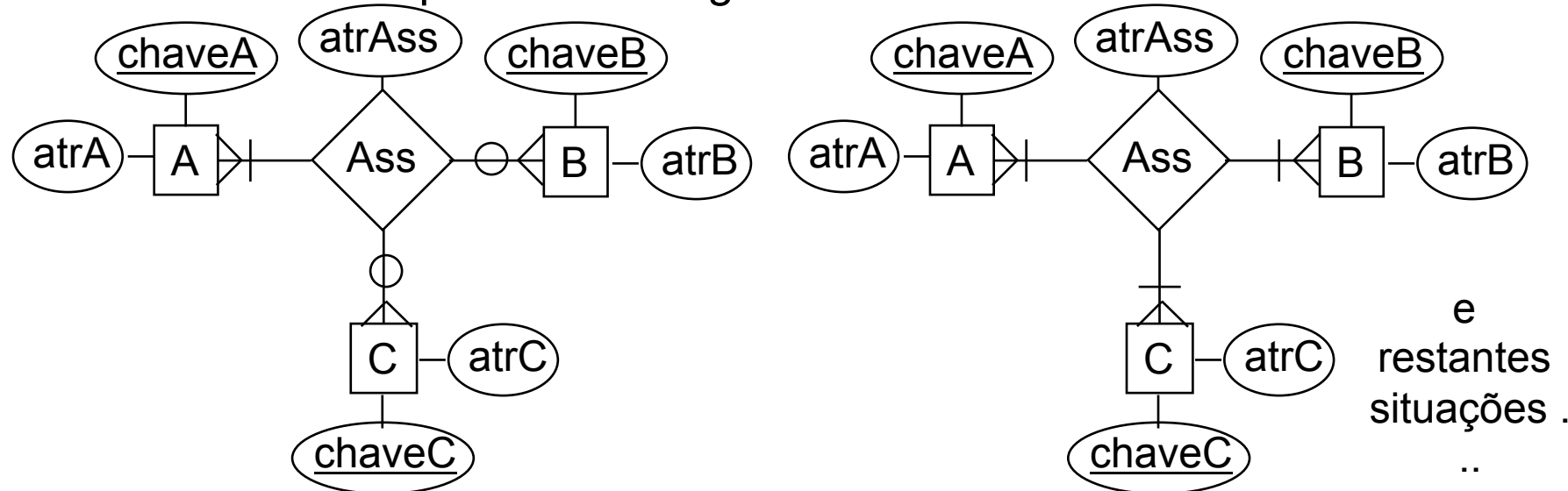
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ C(chaveC, atrC)
 - Chaves Candidatas = { chaveC }
- ◇ A_B_C (chaveA, chaveB, chaveC, atrAss)
 - Chaves Candidatas = { chaveA, chaveB, chaveC }
 - Chaves Estrangeiras = { chaveA } e { chaveB } e { chaveC }

Associação ternária M:N:P (cont.)

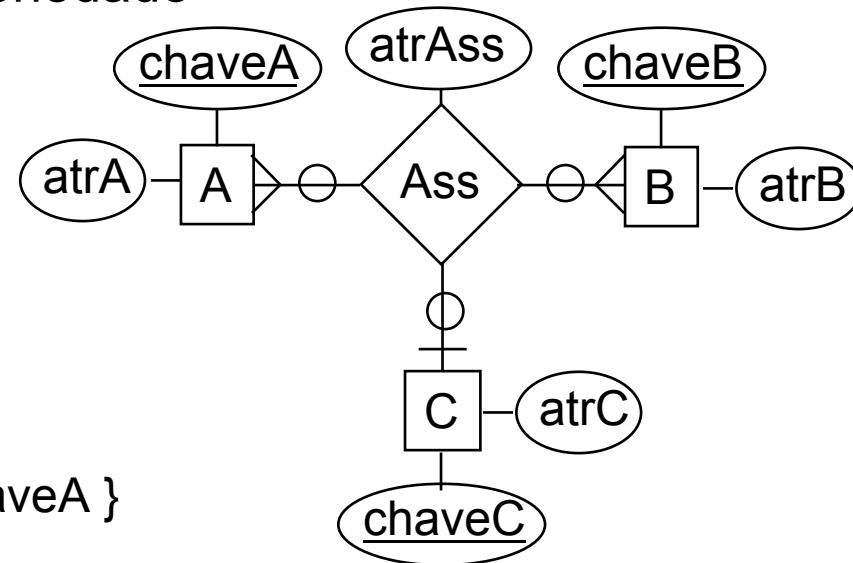
- Nos casos em que existe obrigatoriedade



- O Esquema Relacional é o mesmo que o do caso em que não existe obrigatoriedade em nenhum dos lados
- No entanto é preciso notar que a garantia da participação obrigatória não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação ternária M:N:1

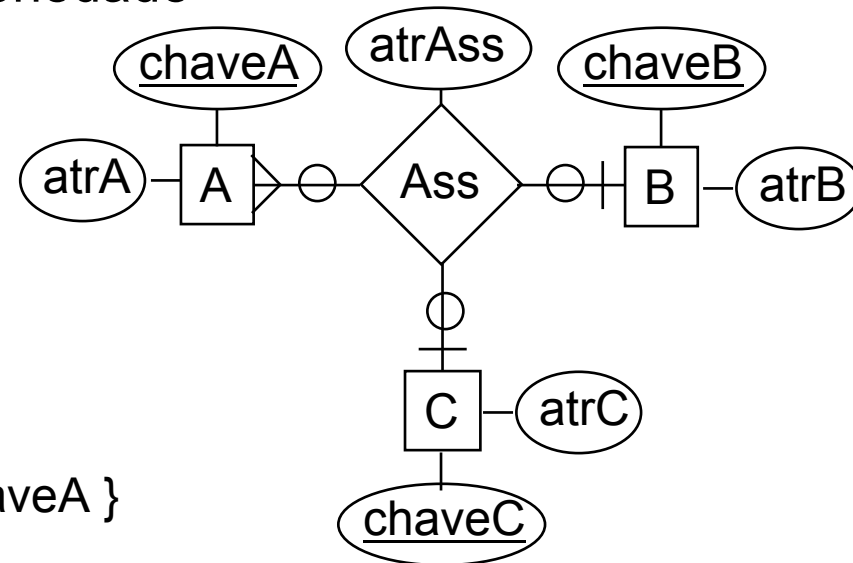
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ C(chaveC, atrC)
 - Chaves Candidatas = { chaveC }
- ◇ A_B_C (chaveA, chaveB, chaveC, atrAss)
 - Chaves Candidatas = { chaveA, chaveB }
 - Chaves Estrangeiras = { chaveA } e { chaveB } e { chaveC }

Associação ternária M:1:1

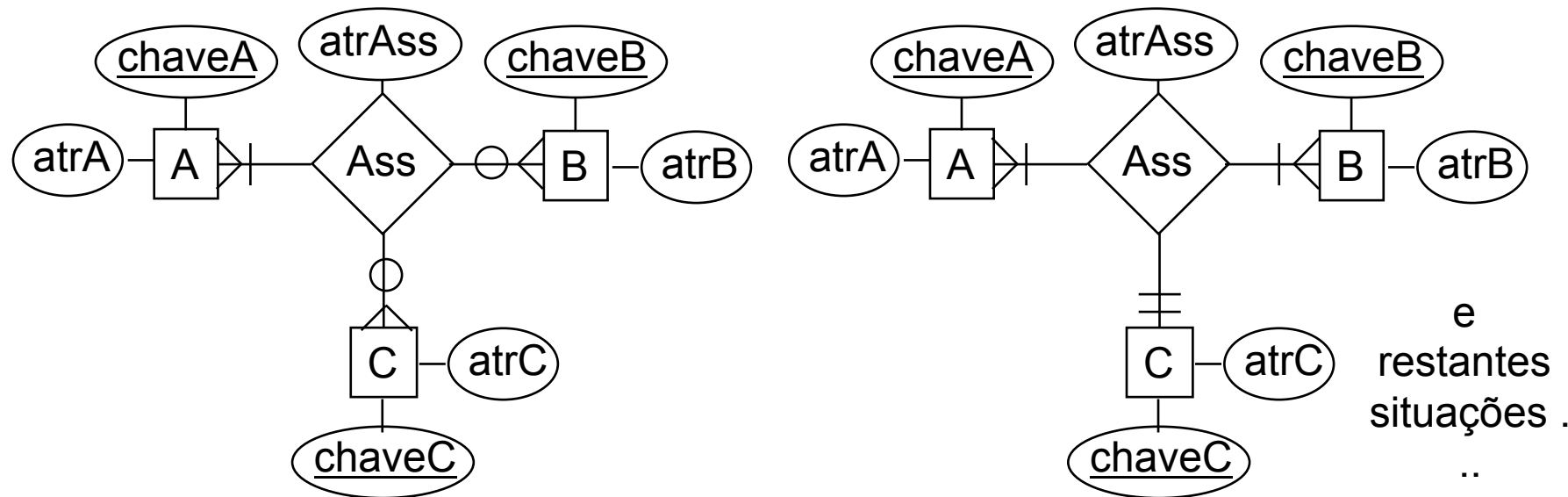
- Independentemente da obrigatoriedade



- ◇ A(chaveA, atrA)
 - Chaves Candidatas = { chaveA }
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ C(chaveC, atrC)
 - Chaves Candidatas = { chaveC }
- ◇ A_B_C (chaveA, chaveB, chaveC, atrAss)
 - Chaves Candidatas = { chaveA }
 - Chaves Estrangeiras = { chaveA } e { chaveB } e { chaveC }

Associação ternária M:N:P, M:N:1, M:1:1

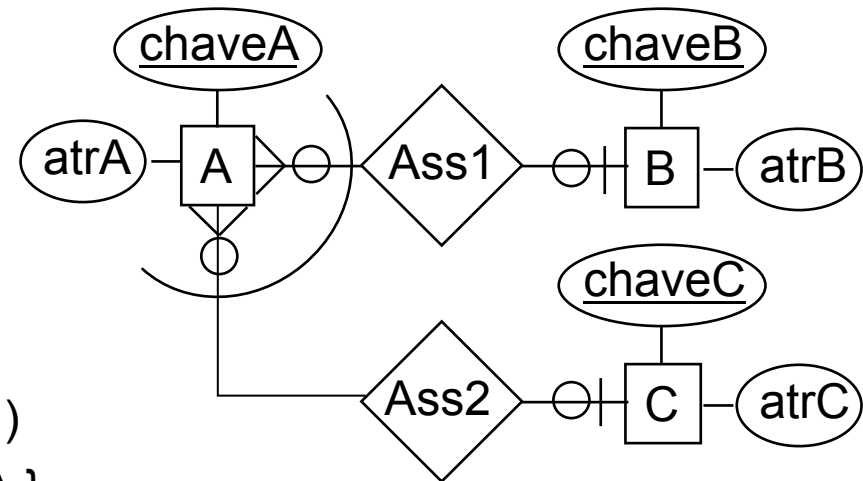
- Nos casos em que existe obrigatoriedade



- O Esquema Relacional obtido é sempre o mesmo que o do caso em que não existe obrigatoriedade em nenhum dos lados
- No entanto é preciso notar que a garantia da participação obrigatória não é conseguida directamente pelo Esquema Relacional (necessita processamento adicional)

Associação Exclusiva

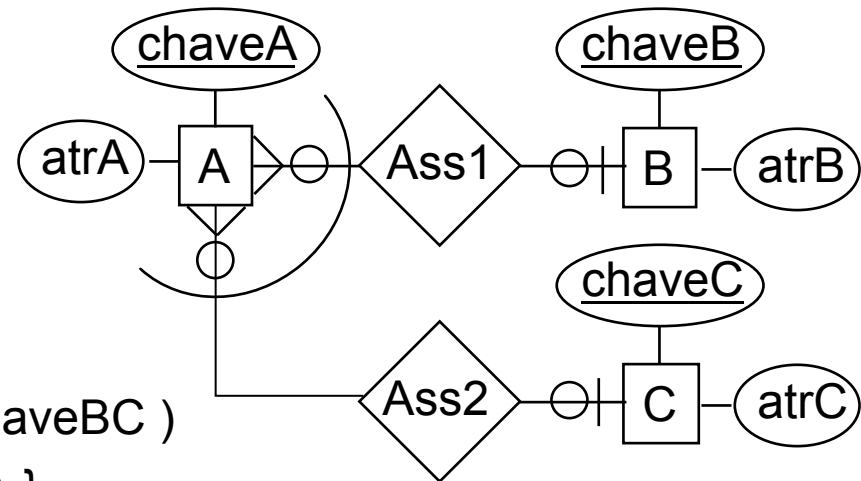
- As Chaves Primárias das Entidades mutuamente exclusivas não têm o mesmo Domínio



- ◇ A(chaveA, atrA, chaveB, chaveC)
 - Chaves Candidatas = { chaveA }
 - Chaves Estrangeiras = { chaveB } e { chaveC }
 - em cada tuplo de A, apenas uma das Chaves Estrangeiras pode ser diferente de NULL
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ C(chaveC, atrC)
 - Chaves Candidatas = { chaveC }

Associação Exclusiva (cont.)

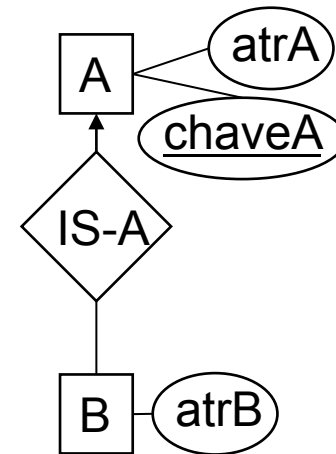
- Todas as Chaves Primárias das Entidades mutuamente exclusivas têm o mesmo Domínio



- ◇ A(chaveA, atrA, * atribDiscr, ** chaveBC)
 - Chaves Candidatas = { chaveA }
 - * Atributo discriminante que pode ter os valores “B”, “C” ou NULL
 - ** Atributo que irá ter o valor de chaveB ou chaveC consoante o valor de atribDiscr for respectivamente “B” ou “C”
- ◇ B(chaveB, atrB)
 - Chaves Candidatas = { chaveB }
- ◇ C(chaveC, atrC)
 - Chaves Candidatas = { chaveC }

Generalização

- Apenas 1 sub-Entidade e ela (B) tem a mesma Chave Primária que a Entidade Generalização (A)
- Se a Generalização for (p, e) ou (p, s)



◇ A(chaveA, atrA)

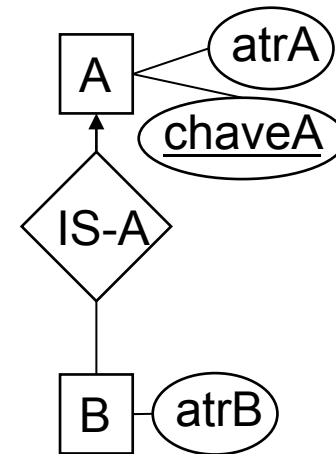
- Chaves Candidatas = { chaveA }

◇ B(chaveA, atrB)

- Chaves Candidatas = { chaveA }
- Chaves Estrangeiras = { chaveA } (refere a Chave de A)

Generalização (cont.)

- Apenas 1 sub-Entidade e ela (B) tem a mesma Chave Primária que a Entidade Generalização (A)
- Se a Generalização for (t, e) ou (t, s)



◇ A(chaveA, atrA, atrB)
· Chaves Candidatas = { chaveA }

Generalização (cont. 1)

- Mais do que 1 sub-Entidade e todas elas (B, C, ...) têm a mesma Chave Primária que a Entidade Generalização (A)

◇ A(chaveA, atrA, * atribDiscr)

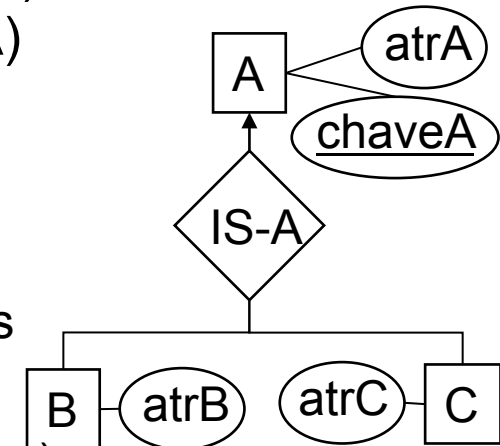
- Chaves Candidatas = { chaveA }
- * Atributo discriminante que pode ter os valores
 - “B”, “C” se a Generalização for (t, e)
 - “B”, “C”, NULL se a Generalização for (p, e)
 - “B”, “C”, “B_C” se a Generalização for (t, s)
 - “B”, “C”, “B_C”, NULL se a Generalização for (p, s)

◇ B(chaveA, atrB)

- Chaves Candidatas = { chaveA }
- Chaves Estrangeiras = { chaveA }

◇ C(chaveA, atrC)

- Chaves Candidatas = { chaveA }
- Chaves Estrangeiras = { chaveA }



Generalização (cont. 2)

- Mais do que 1 sub-Entidade e algumas (B, C, ...) poderão ter Chave Primária diferente da Entidade Generalização (A)

◇ A(chaveA, atrA, * atribDiscr)

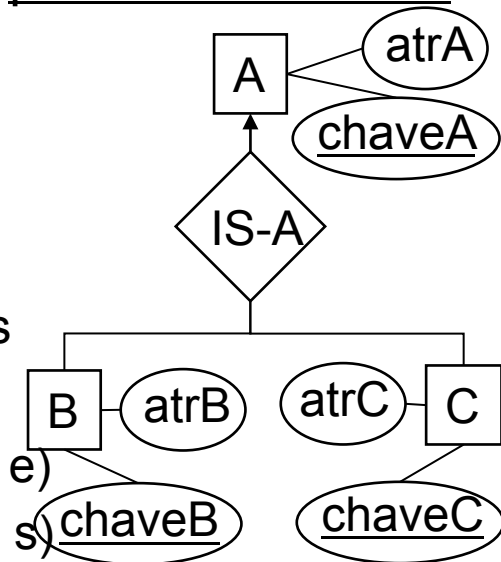
- Chaves Candidatas = { chaveA }
- * Atributo discriminante que pode ter os valores
 - “B”, “C” se a Generalização for (t, e)
 - “B”, “C”, NULL se a Generalização for (p, e)
 - “B”, “C”, “B_C” se a Generalização for (t, s)
 - “B”, “C”, “B_C”, NULL se a Generalização for (p, s)

◇ B(chaveB, atrB, chaveA)

- Chaves Candidatas = { chaveB } e { chaveA }
- Chaves Estrangeiras = { chaveA }

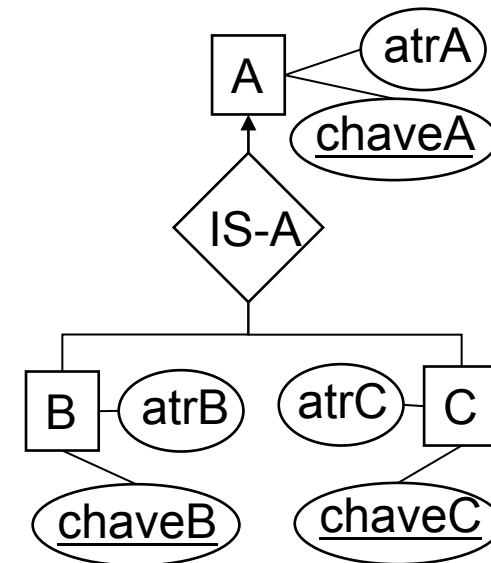
◇ C(chaveC, atrC, chaveA)

- Chaves Candidatas = { chaveC } e { chaveA }
- Chaves Estrangeiras = { chaveA }



Generalização (cont. 3)

- Se a Generalização tiver Cobertura Total com Sobreposição, ou seja se for do tipo (t, s),
 - ◇ podem-se ter Esquemas de Relação apenas para as sub-Entidades, replicando os Atributos da Entidade Generalização nos Esquemas de Relação das sub-Entidades



- ◇ B(chaveB, atrB, chaveA, atrA)
 - Chaves Candidatas = { chaveB } e { chaveA }
- ◇ C(chaveC, atrC, chaveA, atrA)
 - Chaves Candidatas = { chaveC } e { chaveA }

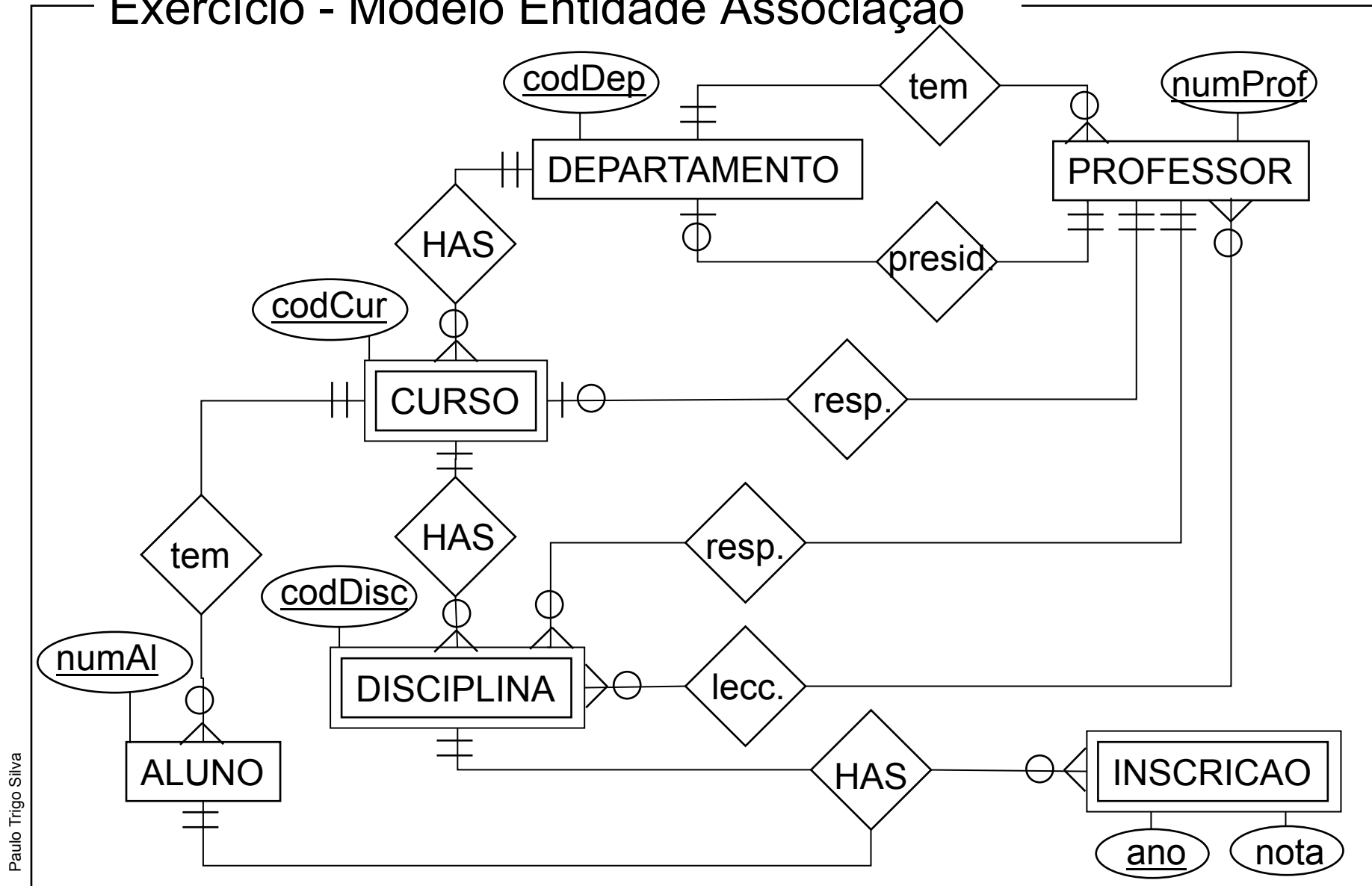
Exercício

- Departamentos: cada departamento é identificado por um código único dentro da escola. Tem um presidente que é professor desse departamento.
- Curso: cada curso depende de um departamento e tem uma identificação própria dentro do seu departamento. Tem um responsável que é professor desse departamento.
- Disciplina: cada disciplina depende do curso e, dentro deste, tem uma identificação própria. Tem um responsável que é professor do respectivo departamento. Cada disciplina é leccionada por um conjunto de professores.

Exercício (cont.)

- Alunos: cada aluno possui um número que o identifica em toda a escola e frequenta um único curso. Os alunos inscrevem-se anualmente em várias disciplinas do seu curso e têm uma nota em resultado dessa inscrição.
- Professores: cada professor tem um número único dentro da escola e pertence a um departamento. Os professores podem ser presidentes do departamento e/ou responsáveis por um curso (apenas um). Os professores podem ser responsáveis por disciplinas (várias). Cada professor lecciona um conjunto de disciplinas.
- Atendendo a esta descrição do sistema em análise,
 - ◊ Construa um Modelo Entidade - Associação que o represente
 - ◊ Do Modelo Entidade - Associação construa o Esquema Relacional

Exercício - Modelo Entidade Associação



Exercício - Modelo Relacional

- DEPARTAMENTO(codDep, numProf)
 ◇ { numProf } é também Chave Candidata
- PROFESSOR(numProf, codDep)
- CURSO(codCur, codDep, numProf)
 ◇ { numProf } é também Chave Candidata
- DISCIPLINA(codDisc, codCur, codDep, numProf)
- DISCIPLINA_PROFESSOR(codDisc, codCur, codDep, numProf)
- ALUNO(numAl, codCur)
- INSCRICAO(ano, codDisc, codCur, codDep, numAl, nota)
- Nota: Atributo(s) com o nome de Chave são Chaves Estrangeiras