

Banco de Dados

Módulo 3 - Modelo ER Conceitos Adicionais

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO



Tópicos

- Conceitos adicionais do Modelo ER
 - ▶ Entidade Fraca
 - ▶ Especialização / Generalização
 - ▶ Agregação e Relacionamentos sobre Relacionamentos
- Atualizações no Modelo ER
- Resumo

Entidades Fracas

■ Definição:

- ▶ um esquema de entidades E é *fraco* sse E não tem chave
- ▶ um esquema de entidades F é *forte* sse F tem chave

■ Problemas:

- ▶ como recuperar uma entidade fraca do BD?
 - através de consultas aos seus atributos
 - navegando para a entidade através de relacionamentos
- ▶ como identificar uma entidade fraca?
 - *atributo discriminador* +
chave *transferida* de uma ou mais entidades fortes
via um relacionamento *discriminador*

Entidades Fracas

- Relacionamento Discriminador:
 - ▶ Seja E um esquema de entidades fraco.
Seja R um esquema de relacionamentos sobre F_1, \dots, F_n .
 - ▶ R é um esquema de relacionamentos *discriminador* para E sse:
 - E ocorre em um único papel em R e todos os outros esquemas de entidades de R são fortes
 - R é total em E
 - E é um identificador de R

Entidades Fracas

Turma

A	L520	t1
B	L512	t2
A	L540	t3

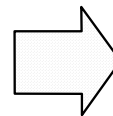
Pertence

t1	d1
t2	d1
t3	d2

Disciplina

d1	inf1731	BD
d2	inf1732	CSGBD

Pertence é total em Turma
Turma é identificador de Pertence
Código é chave de Disciplina



Pertence é discriminador de Turma

Entidades Fracas

■ Atributo Discriminador:

- ▶ Seja E um esquema de entidades fraco.
Suponha que E tenha um atributo A .

Seja R um esquema de relacionamentos sobre F_1, \dots, F_n .
Suponha que R seja discriminador para E e que $F_i = E$.

- ▶ A é um *atributo discriminador* para E em R sse:

para todo estado s do BD,

para todo (f_1, \dots, f_n) e (g_1, \dots, g_n) em $s(R)$

se $(f_1, \dots, f_{i-1}, f_{i+1}, \dots, f_n) = (g_1, \dots, g_{i-1}, g_{i+1}, \dots, g_n)$ e $s(A)(f_i) = s(A)(g_i)$

então $f_i = g_i$

Entidades Fracas

Turma

A	L520	t1
B	L512	t2
A	L540	t3

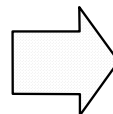
Pertence

t1	d1
t2	d1
t3	d2

Disciplina

d1	inf1731	BD
d2	inf1732	CSGBD

As turmas da mesma disciplina
tem letras distintas
(que podem ser reusadas
em disciplinas distintas)



Letra é atributo discriminador de Turma

Entidades Fracas

■ Entidade Fraca ou Atributo?

▶ Situação:

- E é um esquema de entidades fraco
- E não tem atributos
- R é um esquema de relacionamento binário entre E e F
- R é discriminador para E

▶ Simplificação:

- E pode ser modelado como atributo de F
- R removido
- os possíveis esquemas de relacionamento sobre E passam a ser sobre F

▶ Exemplo:

– Situação:

- Professor modelado como entidade fraca para Turma, ligado por um relacionamento Leciona
- Professor não tem atributos

– Simplificação:

- Professor pode ser modelado como atributo de Turma
- Leciona deixa de existir

Entidades Fracas

■ Entidade Fraca ou Atributo?

▶ Situação:

- E é um esquema de entidades fraco
- E não tem atributos
- R é um esquema de relacionamento entre F_1, \dots, F_n e $E = F_i$
- R é discriminador para E

▶ Simplificação:

- E pode ser modelado como atributo de R
- os possíveis esquemas de relacionamento sobre E passam a ser sobre R

Especialização / Generalização

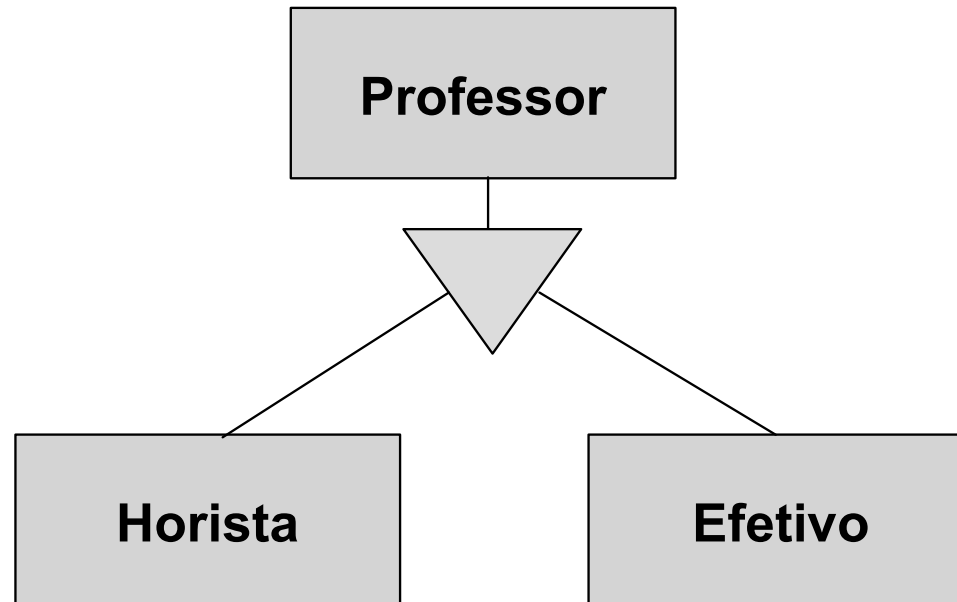
■ Definição:

- ▶ Sejam E e F dois esquemas de entidades
- ▶ E é uma especialização de F ou F é uma generalização de E sse
 - E herda todos os atributos de F
 - em todo estado s do BD, $s(E)$ é um subconjunto de $s(F)$

Especialização / Generalização

- Exemplo:

Horista e Efetivo são especializações de Professor



Especialização / Generalização

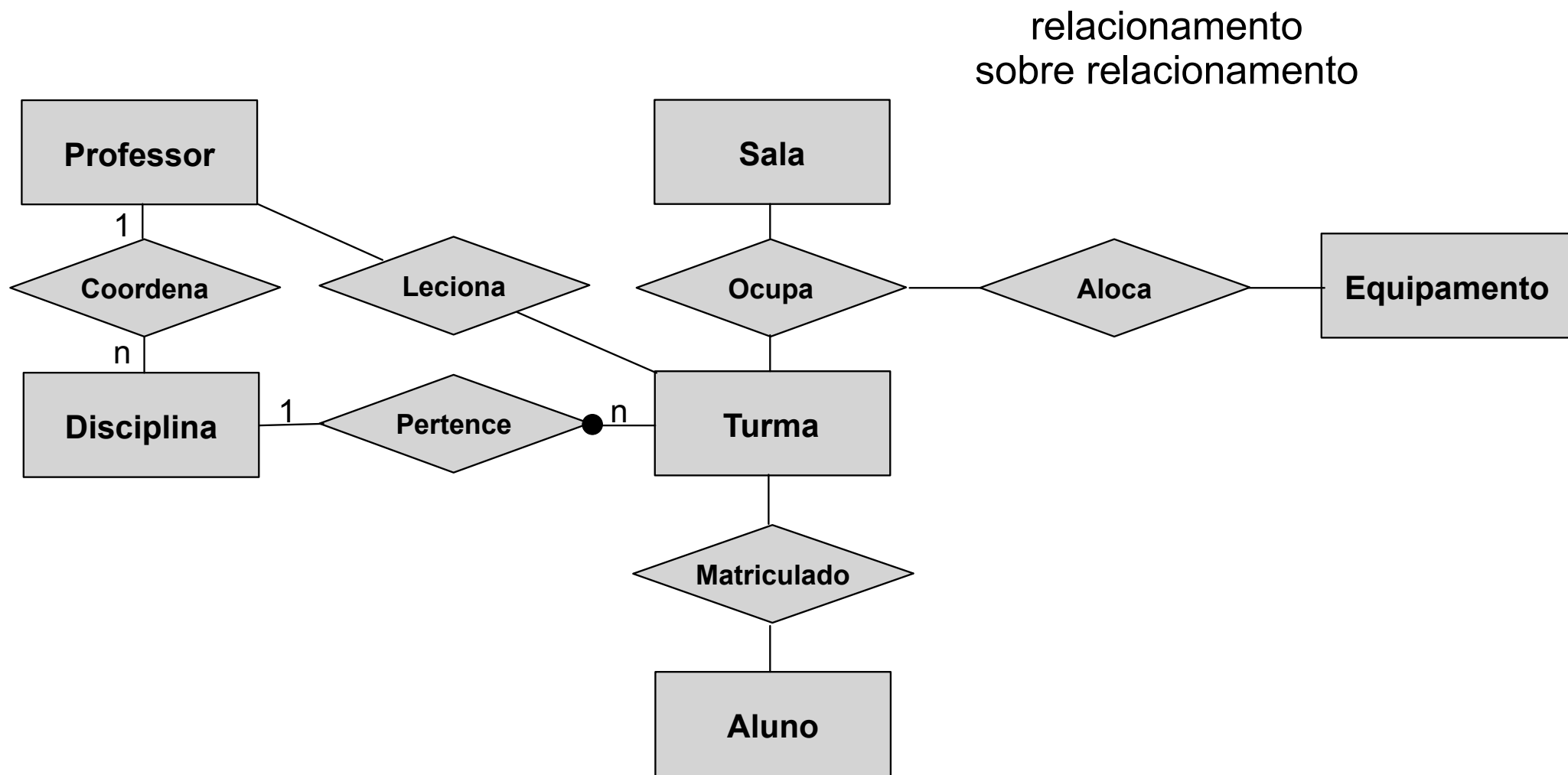
- Formas de definição:
 - manual: pertinência definida explicitamente pelo usuário
 - automática: pertinência definida por predicados

- Restrições de integridade sobre especializações:
 - disjuntas ou com superposição
 - total ou parcial

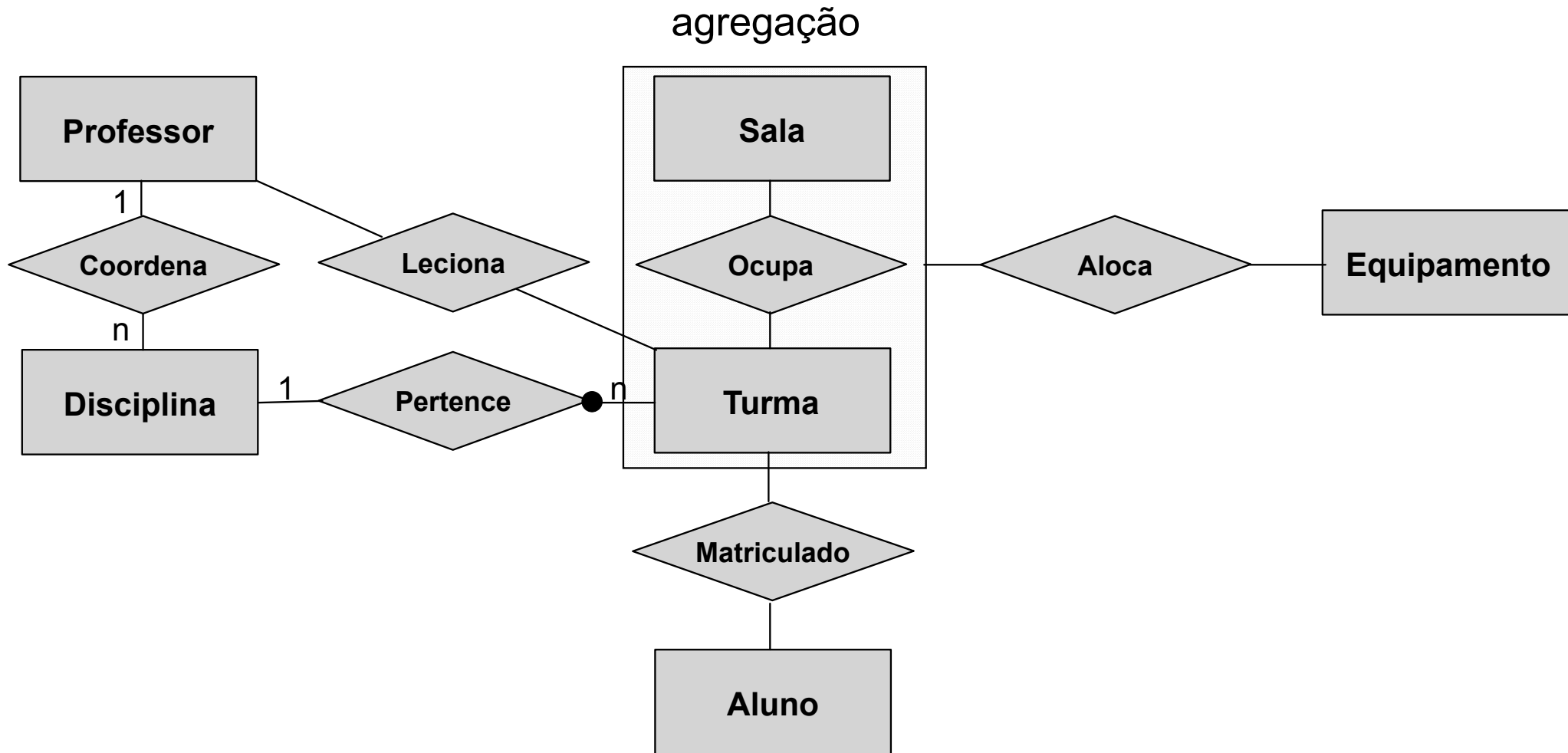
Agregação

- Agregação e relacionamentos sobre relacionamentos:
 - ▶ a noção de relacionamento pode ser estendida de tal forma a permitir que um relacionamento envolva entidades e também relacionamentos
 - ▶ a noção de agregação corresponde simplesmente a tratar um relacionamento e as entidades participantes como uma nova entidade
 - ▶ a noção de agregação é equivalente a permitir relacionamentos sobre relacionamentos

Aggregação



Agregação



Atualizações no Modelo ER

- Operações elementares:
 - ▶ **inserção** de uma entidade ou relacionamento
 - ▶ **remoção** de uma entidade ou relacionamento
 - ▶ **teste** de existência de uma entidade ou relacionamento
 - ▶ **recuperação** de atributos de entidade ou relacionamento
 - ▶ **recuperação** de entidades participantes de relacionamento

Atualizações no Modelo ER

- Transação (correta):
 - ▶ seqüência de operações elementares que preservam todas as restrições de integridade definidas no esquema ER

- Problema:
 - ▶ como construir transações corretas, com relação a:
 - chaves
 - identificadores
 - restrições de cardinalidade
 - *dependências de existência*

Atualizações no Modelo ER

- Dependências de existência:

- ▶ Seja R um esquema de relacionamento sobre F_1, \dots, F_n .
Seja s um estado do BD.

Para todo relacionamento (f_1, \dots, f_n) em $s(R)$,
 f_i existe em $s(F_i)$

- ▶ Seja R um esquema de relacionamento sobre F_1, \dots, F_n .
Suponha que R seja total sobre F_i .
Seja s um estado do BD.

Para todo e_i em $s(F_i)$,
existe (f_1, \dots, f_n) em $s(R)$ tal que $e_i = f_i$

Atualizações no Modelo ER

■ Dependências de existência (cont.):

- ▶ Seja F uma generalização de F_1, \dots, F_n .
Seja s um estado do BD.

todo f_i em $s(F_i)$ existe em $s(F)$

- ▶ Seja F uma generalização de F_1, \dots, F_n .
Suponha que F seja total.
Seja s um estado do BD.

para todo f em $s(F)$,
 f existe em $s(F_i)$, para algum i em $[1, n]$

Atualizações no Modelo ER

- Dependências de existência (cont.):
 - ▶ Seja F uma generalização de F_1, \dots, F_n .
Suponha que não haja superposição entre F_i e F_k .
Seja s um estado do BD.

para todo f em $s(F)$,
 f existe em $s(F_i)$ sse f não existe em $s(F_k)$

Atualizações no Modelo ER

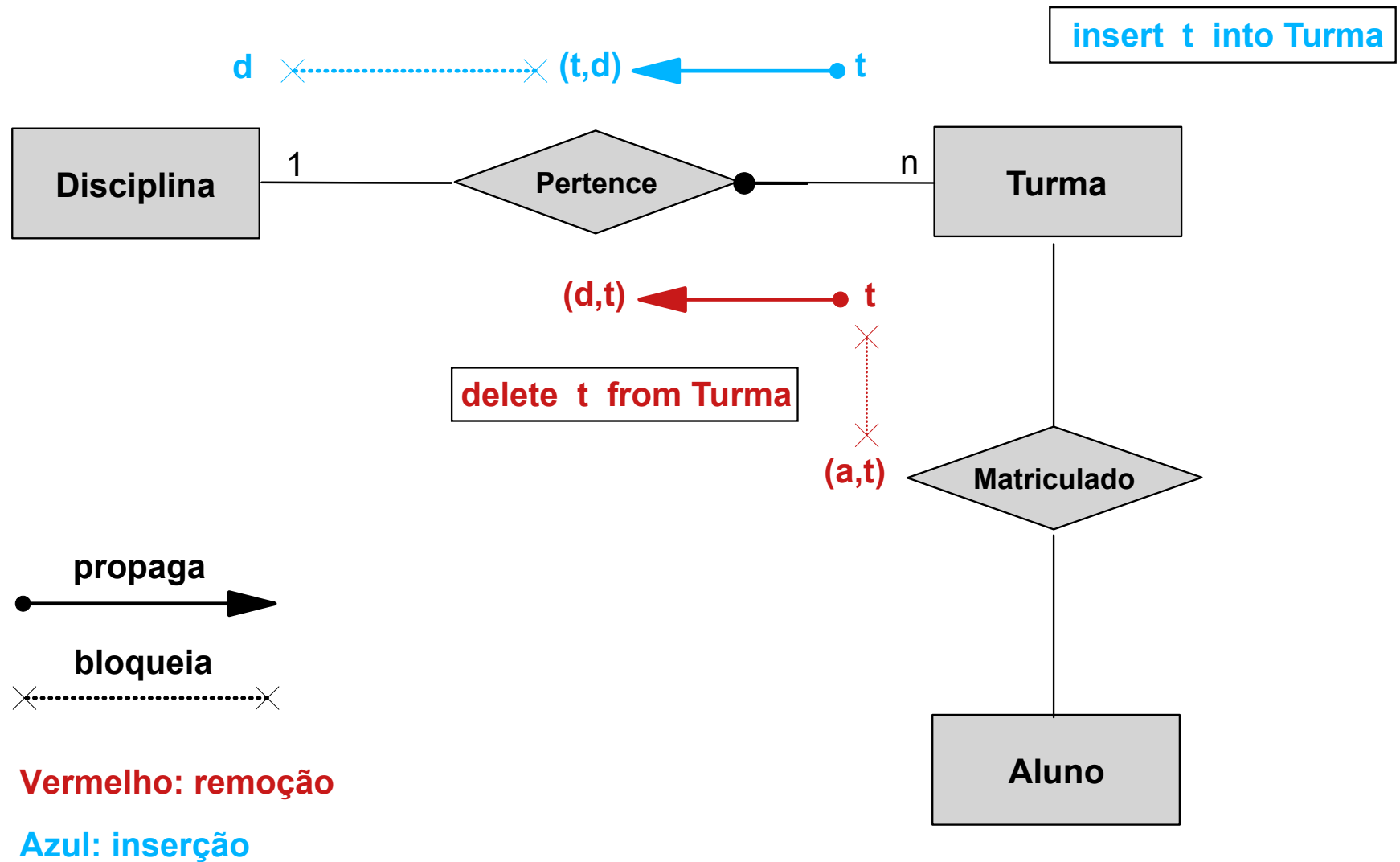
- Sub-problema:

- ▶ como construir transações corretas com relação a dependências de existência?

- Solução:

- ▶ definição de *assertivas* para bloquear inserções ou remoções que violariam as dependências de existência, ou
- ▶ definição de *triggers* para propagar inserções ou remoções que violariam as dependências de existência

Atualizações no Modelo ER



Atualizações no Modelo ER

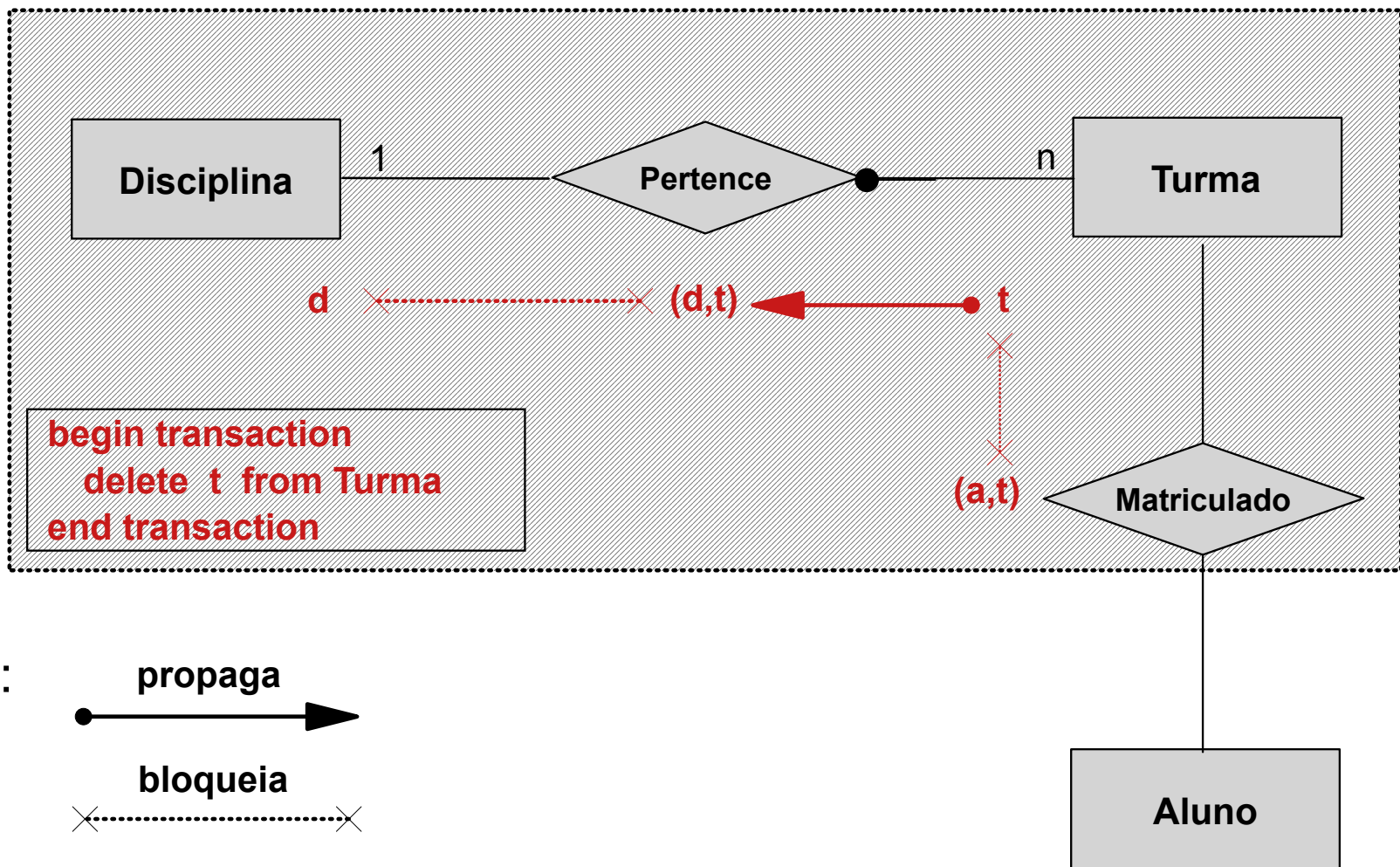
■ Problema:

- ▶ Com encapsular os conjuntos de entidades e relacionamentos de tal forma a preservar as restrições de integridade?

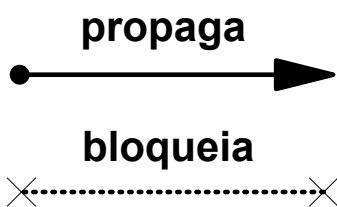
■ Solução:

- ▶ definir triggers para propagação de atualizações e assertivas para bloqueio de atualizações
- ▶ definir transações da seguinte forma:
 - transação inicia com uma inserção (ou remoção)
 - sistema dispara triggers e aplica assertivas
 - triggers podem disparar outros triggers
 - triggers podem ser bloqueados por assertivas
 - assertivas podem provocar o rollback de toda a transação

Atualizações no Modelo ER



Legenda:



Vermelho: remoção

Azul: inserção



Resumo

■ Entidade

- ▶ um objeto com ***existência independente*** deve ser modelado como entidade

(para que possa ser representado no BD incondicionamente, ou seja, sem que dependa da representação de outros objetos)

- ▶ um objeto com ***mais de uma propriedade própria*** deve ser modelado como uma entidade

(para evitar redundâncias)

Resumo

■ Relacionamento

- ▶ um **evento** que envolve vários objetos deve ser modelado como um **relacionamento**

■ Atributo

- ▶ uma **propriedade** de uma entidade ou relacionamento deve ser modelada como um atributo (da entidade ou do relacionamento)

■ Especialização / Generalização

- ▶ organiza conjuntos de entidades em hierarquias
- ▶ racionaliza a definição de atributos via herança