

cin.ufpe.br



UNIVERSIDADE FEDERAL DE PERNAMBUCO



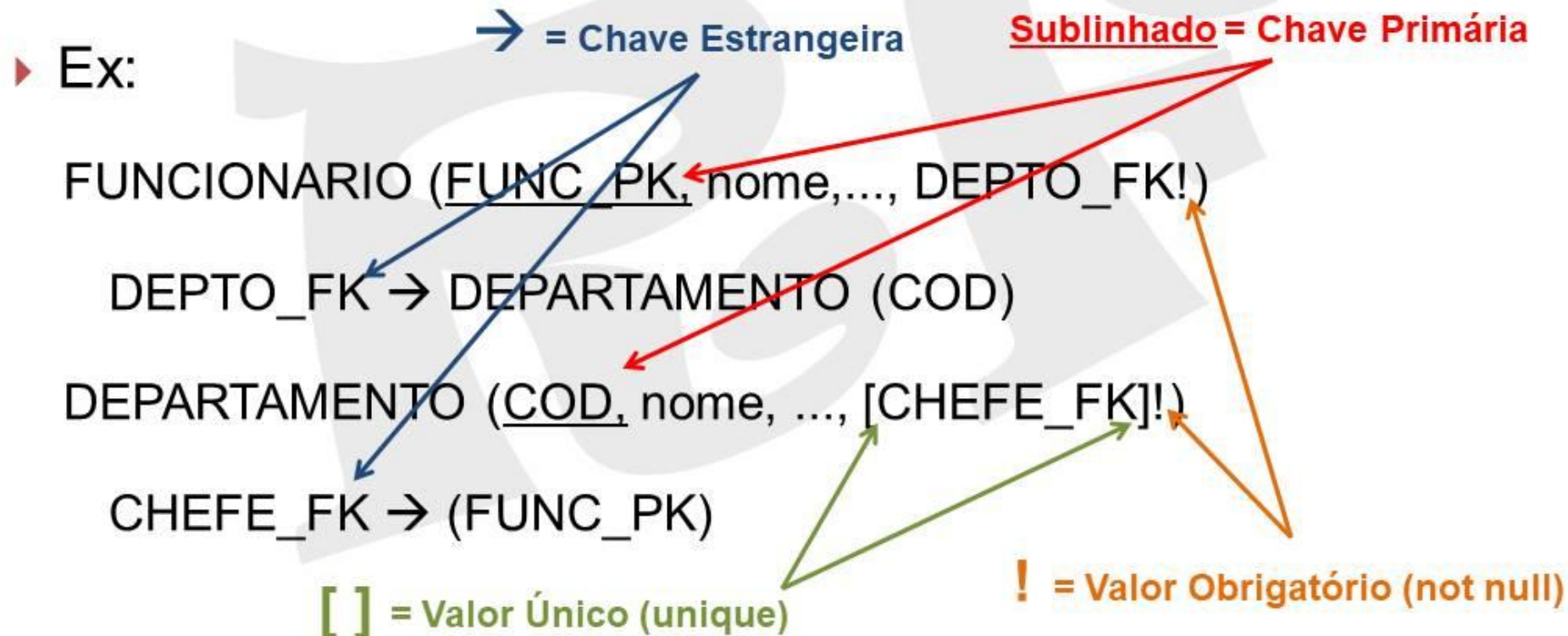
Projeto Lógico Relacional



Por: Prof. Robson do Nascimento Fidalgo

Notação Simplificada (reforçando)

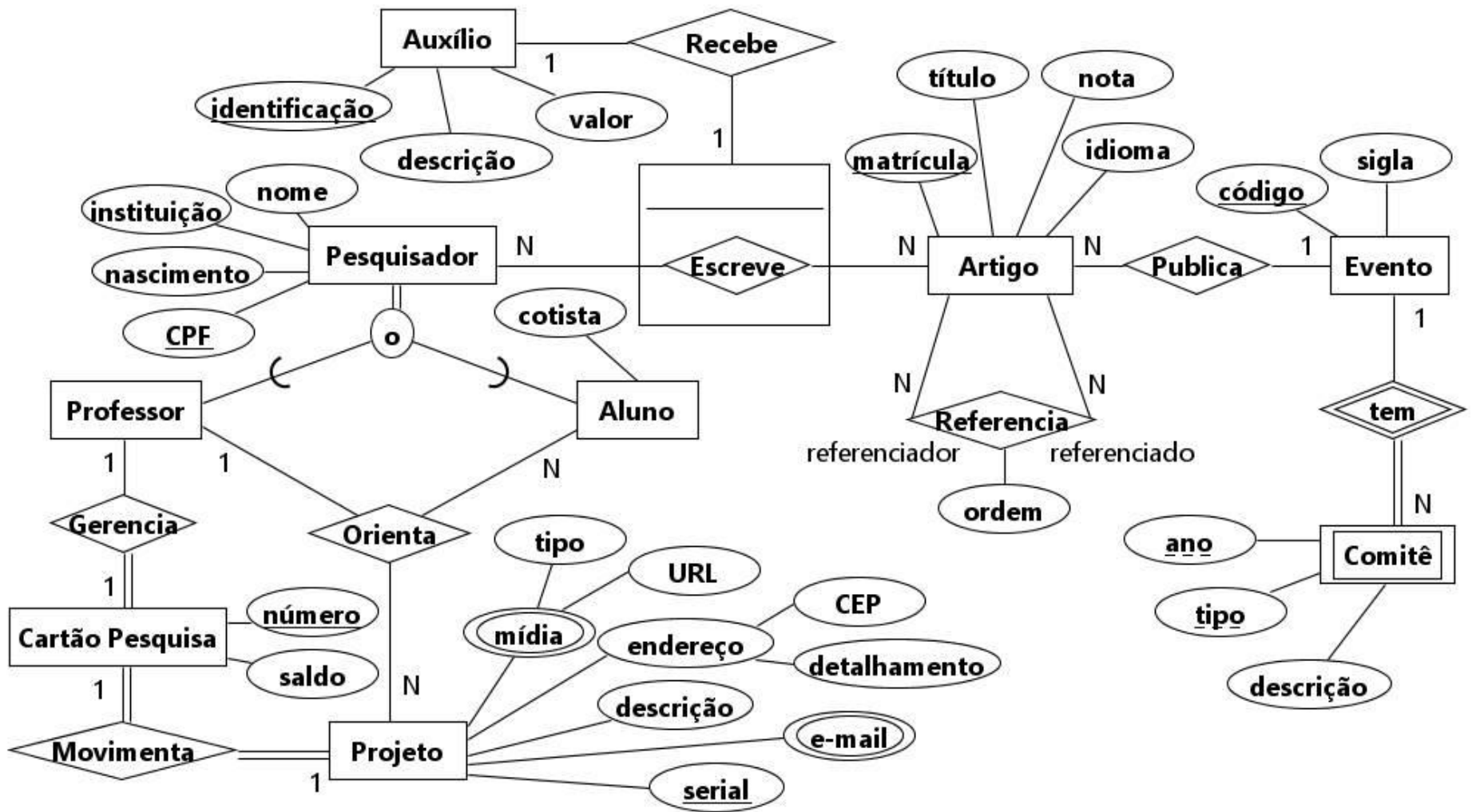
- ▶ Esquema Relacional = definição das tabelas
- ▶ Representação básica (incompleta mas compacta)



Mapeamento EER-Relacional

- ▶ 1 esquema EER pode gerar N esquemas Relacionais
 - ▶ Existem várias maneiras de mapear relacionamentos e heranças
- ▶ O mapeamento deve priorizar:
 - ▶ 1º) Evitar junções (consultas mais rápidas)
 - ▶ 2º) Diminuir o número de chaves (índices menores e mais rápidos)
 - ▶ 3º) Evitar campos opcionais (menos testes de qualidade dos dados)
- ▶ Dicas para nomear relações e atributos
 - ▶ Use nomes curtos
 - ▶ Elimine espaços em branco e caracteres especiais
 - ▶ Adote um padrão

Caso Prático



Passos para fazer o mapeamento

- ▶ 1º) Mapear as entidades regulares e seus atributos
- ▶ 2º) Mapear as entidades fracas e seus atributos
- ▶ 3º) Mapear as super/subentidades e seus atributos
- ▶ 4º) Mapear as entidades associativas
- ▶ 5º) Mapear os relacionamentos e seus atributos

Mapear as entidades regulares e seus atributos

- ▶ Cada entidade regular é mapeada para uma relação
 - ▶ A chave primária (PK) da relação é o atributo identificador da entidade mapeada
- ▶ Cada atributo multivalorado é mapeado para:
 - ▶ 1) N atributos (desde de que N seja pequeno) ou
 - ▶ 2) Uma relação cuja PK é o atributo multivalorado + a PK da relação origem
 - ▶ A PK que migrou da relação origem é chave estrangeira (FK)
- ▶ Atributos comum, composto ou derivado são mapeados para atributos da relação

Mapear as entidades regulares e seus atributos



Projeto(serial, descricao, end_CEP, end_detalhamento);

E-mail(serial, e-mail)

serial → Projeto(serial)

Midia(serial, tipo, URL)

serial → Projeto(serial)

OU

Projeto(serial, descricao, end_CEP, end_detalhamento,
e-mail1, e-mail2, e-mail3);

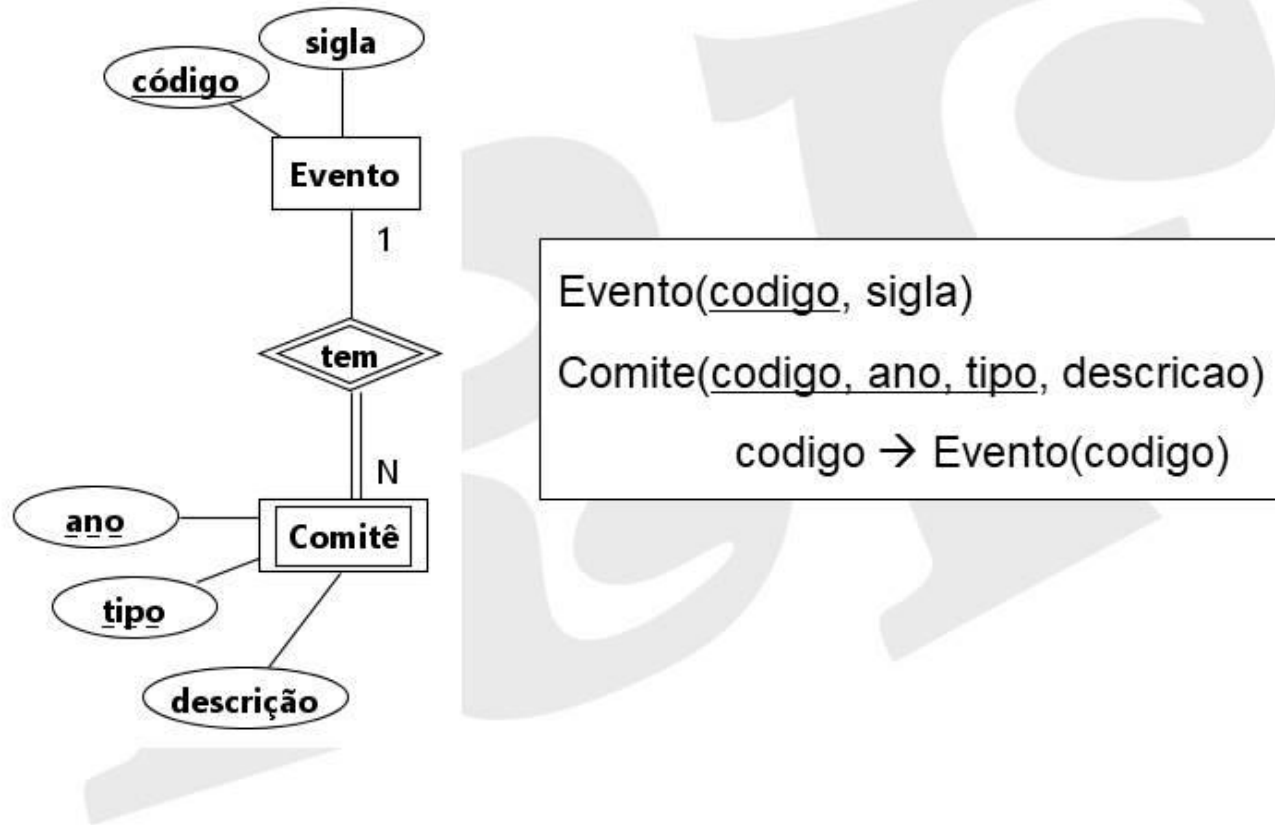
Midia(serial, tipo, URL)

serial → Projeto(serial)

Mapear as entidades fracas e seus atributos

- ▶ Cada entidade fraca é mapeada para uma relação
 - ▶ A PK que mapeia a entidade forte migra como FK
 - ▶ A PK da relação é formada pela FK + o discriminador (se existir)
- ▶ Atributos comum, composto, derivado ou multivalorado seguem os mesmos mapeamentos das entidades regulares

Mapear as entidades fracas e seus atributos



Mapear as super/subentidades e seus atributos

▶ 4 alternativas

- ▶ Uma relação para cada entidade da herança
- ▶ Uma relação para cada subentidade da herança total
- ▶ Uma única relação para toda herança disjunta ou direta
- ▶ Uma única relação para toda herança sobreposta

Uma relação para cada entidade da herança

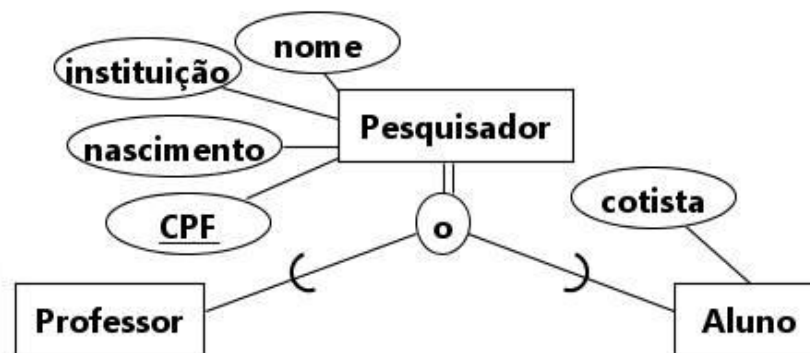
- ▶ Características fortes (+) e fracas (-)
 - ▶ Funciona bem para qualquer tipo de herança (+)
 - ▶ Reduz atributos opcionais (+)
 - ▶ Reduz testes para garantir a qualidade dos dados (+)
 - ▶ Exige junções (-)

Uma relação para cada entidade da herança

► Mapeamento

- Cada Super/subentidade é mapeada para uma relação
 - A PK de cada relação é o atributo identificador da superentidade mapeada
 - A PK de cada relação que mapeia uma subentidade será FK para a superentidade
- Atributos comum, composto, derivado ou multivalorado seguem os mesmos mapeamentos das entidades regulares

Uma relação para cada entidade da herança



Pesquisador(CPF, nascimento, instituicao, nome)

Professor(CPF)

$CPF \rightarrow Pesquisador(CPF)$

Aluno(CPF, cotista)

$CPF \rightarrow Pesquisador(CPF)$

Uma relação para cada subentidade da HT

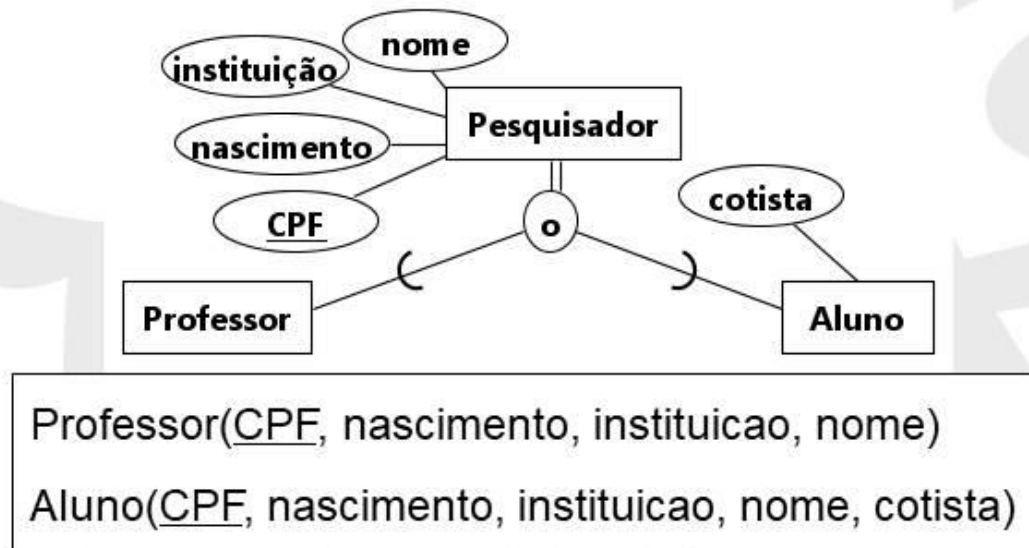
- ▶ Características fortes (+) e fracas (-)
 - ▶ Gera redundância de dados para heranças sobrepostas (-)
 - ▶ Reduz atributos opcionais (+)
 - ▶ Reduz testes para garantir a qualidade dos dados (+)
 - ▶ Reduz junções (+)

Uma relação para cada subentidade da HT

► Mapeamento

- Cada subentidade é mapeada para uma relação
 - A PK de cada relação é o atributo identificador da superentidade mapeada
- Os atributos e relacionamentos da superentidade migram para as relações que mapeiam as subentidades
- Atributos comum, composto, derivado ou multivalorado → igual ao apresentado no mapeamento de entidades regulares

Uma relação para cada subentidade da HT



Uma única relação para toda herança D ou D

- ▶ Características fortes (+) e fracas (-)
 - ▶ Só funciona com heranças definidas por um predicado/condição
 - ▶ Exige atributos opcionais (-)
 - ▶ Desaconselhada quando as subentidades têm muitos atributos ou relacionamentos
 - ▶ Exige testes para garantir a qualidade dos dados (-)
 - ▶ Testar o povoamento dos atributos e dos relacionamentos
 - ▶ Reduz junções (+)

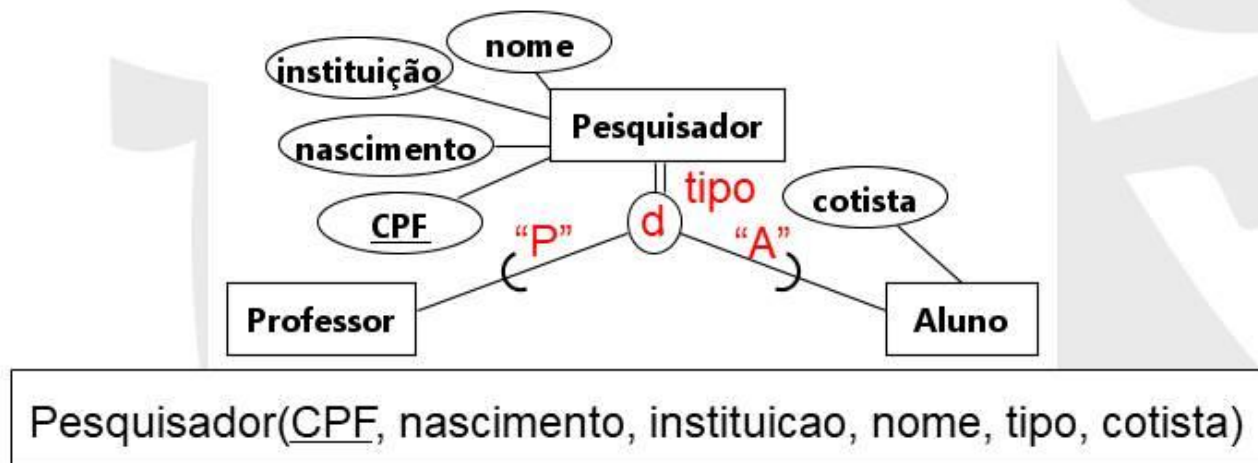
Uma única relação para toda herança D ou D

► Mapeamento

- A superentidade e as subentidades de uma herança são mapeadas para uma única relação
 - A PK da relação é o atributo identificador da superentidade mapeada
- O predicado/condição da herança torna-se um atributo da relação mapeada (seu domínio deve cobrir as subentidades)
- Os atributos e relacionamentos das subentidades migram para a relação mapeada
- Atributos comum, composto, derivado ou multivalorado → igual ao apresentado no mapeamento de entidades regulares

Uma única relação para toda herança D ou D

- ▶ (supondo que a herança é disjunta e definida por um predicado/condição)



Uma única relação para toda herança sobreposta

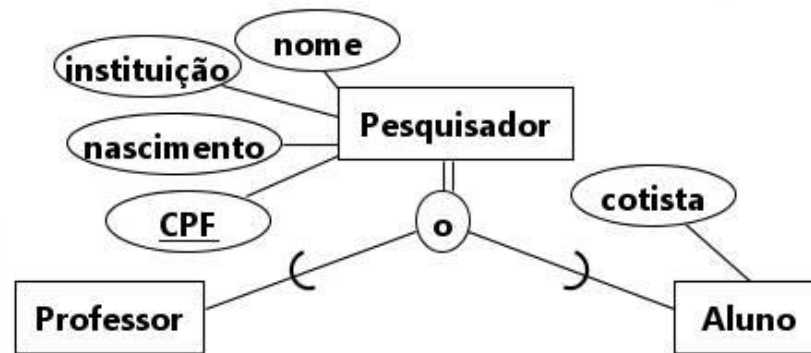
- ▶ Características fortes (+) e fracas (-)
 - ▶ Exige atributos opcionais (-)
 - ▶ Desaconselhada quando as subentidades têm muitos atributos ou relacionamentos
 - ▶ Exige testes para garantir a qualidade dos dados (-)
 - ▶ Testar o povoamento dos atributos e dos relacionamentos
 - ▶ Reduz junções (+)
 - ▶ Reduz redundância de dados (+)

Uma única relação para toda herança sobreposta

► Mapeamento

- A superentidade e as subentidades de uma herança são mapeadas para uma única relação
 - A PK da relação é o atributo identificador da superentidade mapeada
- Para cada subentidade, criar na relação mapeada um atributo booleano
- Os atributos e relacionamentos da superentidade e das subentidades migram para a relação mapeada
- Atributos comum, composto, derivado ou multivalorado → igual ao apresentado no mapeamento de entidades regulares

Uma única relação para toda herança sobreposta



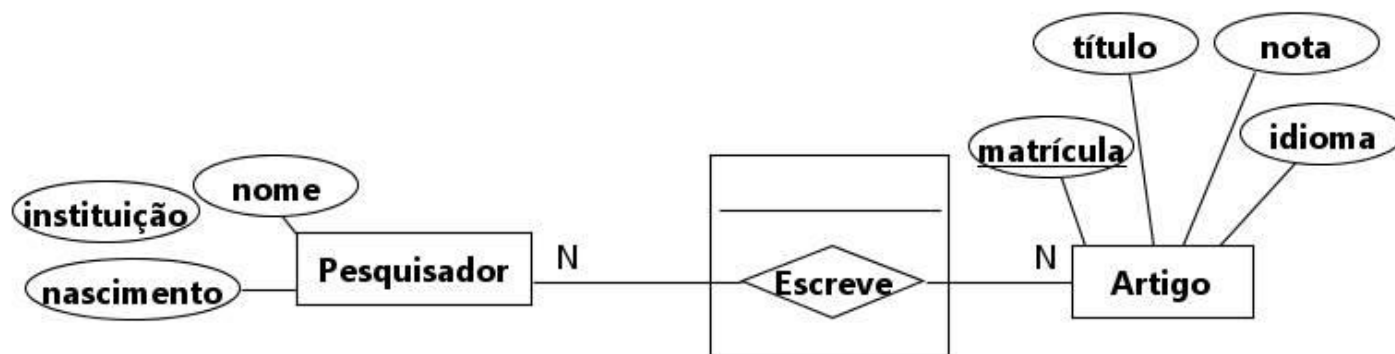
Pesquisador(CPF, nascimento, instituicao, nome, eh_prof, eh_alu, cotista)

Mapear as entidades associativas

▶ Mapeamento

- ▶ Cada Entidade Associativa é mapeada para uma relação
- ▶ As PK das relações envolvidas migram com FK obrigatórias
- ▶ Os atributos do relacionamento (se houverem) ficam na relação mapeada
- ▶ A PK da relação depende do grau e da cardinalidade do relacionamento
 - ▶ Usar as mesmas regras aplicadas em relacionamentos (ver próximos slides)

Mapear as entidades associativas



Escreve(CPF, mat)

CPF → Pesquisador(CPF)

mat → Artigo(matricula)

Mapear os relacionamentos e seus atributos

- ▶ 3 alternativas
 - ▶ Fusão de relações
 - ▶ Adição de chave estrangeira
 - ▶ Criação de relação

Fusão de relações

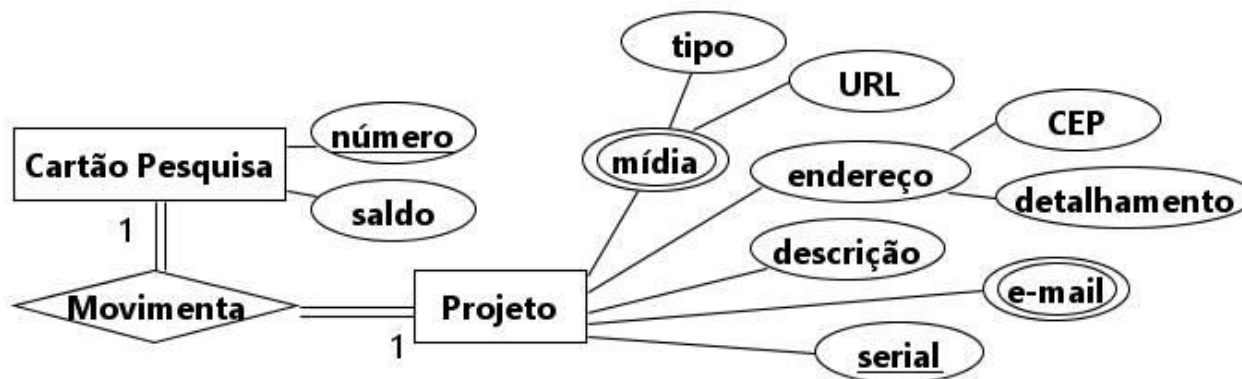
- ▶ Características fortes (+) e fracas (-)
 - ▶ Exige atributos opcionais (-)
 - ▶ Desaconselhada quando as entidades têm muitos atributos ou relacionamentos
 - ▶ Exige testes para garantir a qualidade dos dados (-)
 - ▶ Testar o povoamento dos atributos e dos relacionamentos
- ▶ Reduz junções (+)

Fusão de relações

- ▶ Mapeamento (Melhor caso 1:1 - Total/Total)
 - ▶ Fundir as relações em uma única relação
 - ▶ A PK da relação fundida deve ser uma das PK originais
 - ▶ Dar preferência para a PK que poderá ser mais consultada
 - ▶ Usar “[]” para definir a outra PK como chave alternativa (AK)
 - ▶ Usar “!” para definir a AK como obrigatória

Fusão de relações

- Mapeamento (Melhor caso 1:1 - Total/Total)



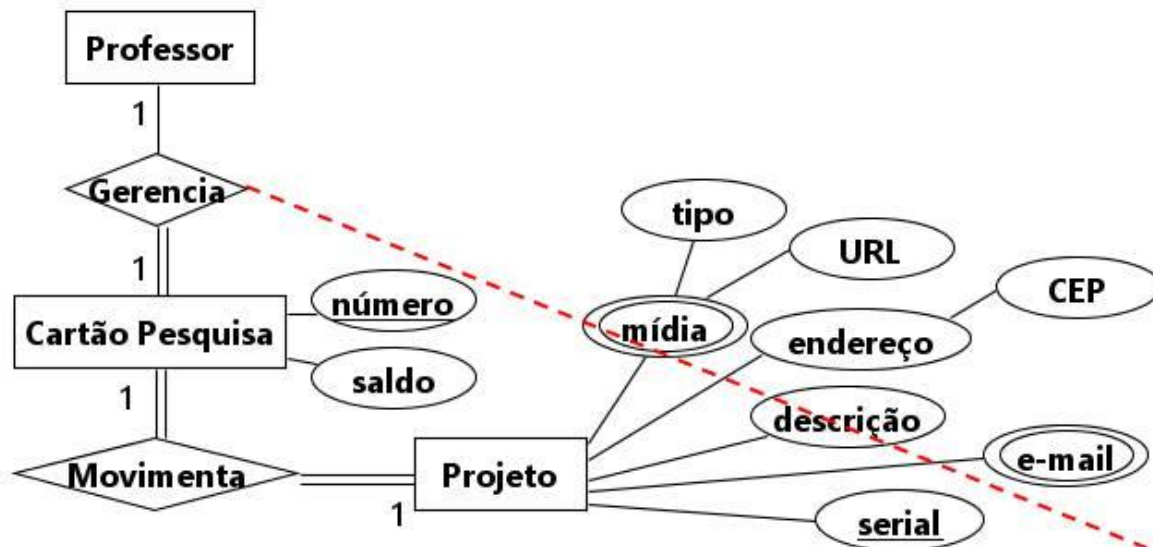
ProjetoCartao(serial, descricao, end_CEP, end_detalhamento, [numero]!, saldo)

Fusão de relações

- ▶ Mapeamento (Caso alternativo 1:1 - Total/Parcial)
 - ▶ Fundir as relações em uma única relação
 - ▶ A PK da relação fundida deve ser a PK da relação original que tem participação parcial
 - ▶ Usar “[]” para definir a outra PK como chave alternativa (AK)
- ▶ Avaliar Custo X Benefício!

Fusão de relações

- Mapeamento (Caso alternativo 1:1 - Total/Parcial)



ProfessorProjetoCartao(serial, descricao, end_CEP, end_detalhamento, numero, saldo, [CPF] !)

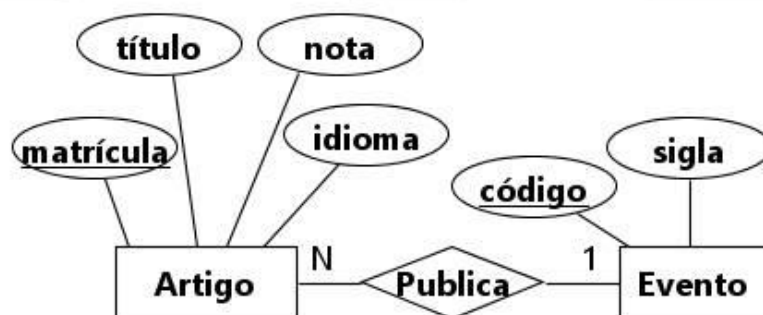
As semânticas da herança e do ternário são prejudicadas!

Adição de chave estrangeira

- ▶ Características fortes (+) e fracas (-)
 - ▶ Reduz atributos opcionais (+)
 - ▶ Reduz testes para garantir a qualidade dos dados (+)
 - ▶ Exige junção (-)

Adição de chave estrangeira

- ▶ Mapeamento (Melhor caso 1:N)
 - ▶ A PK da relação do lado 1 migra como FK para a outra relação
 - ▶ Caso o lado N seja total, usar “!” para definir a FK como obrigatória
 - ▶ Os atributos do relacionamento (se houverem) migram com a PK



Evento(codigo, sigla)

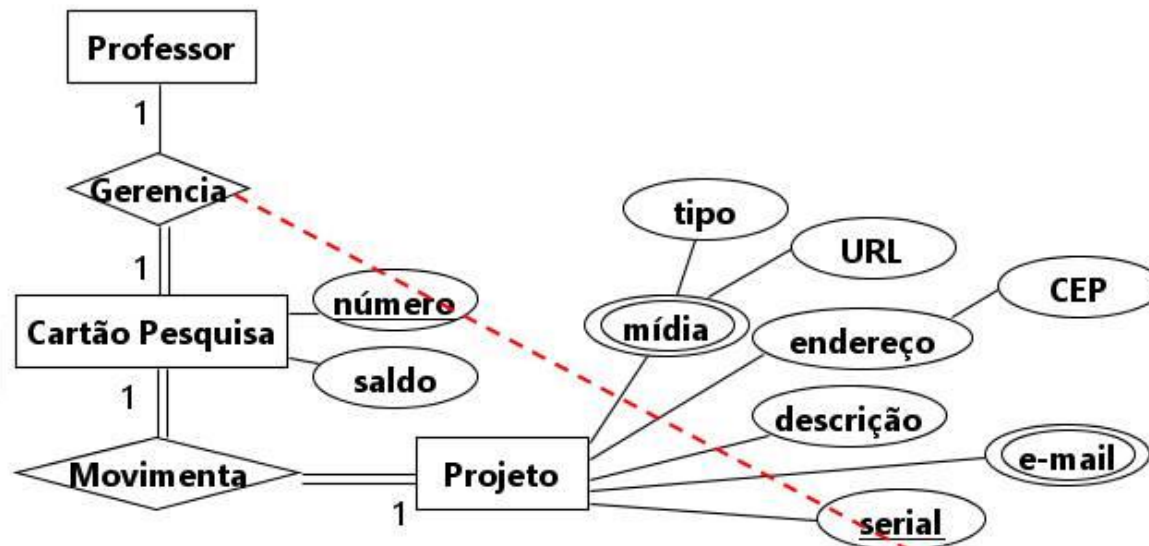
Artigo(matricula, titulo, nota, idioma, codigo)

codigo → Evento(codigo)

Adição de chave estrangeira

- ▶ Mapeamento (Caso alternativo 1:1)
 - ▶ Parcial/Parcial - a PK de qualquer uma das relações migra como FK única e opcional (usar “[]” para definir a FK como única)
 - ▶ Total/Parcial - a PK da relação do lado parcial migra como FK única e obrigatória (usar “[] + !” para definir a FK como única e obrigatória)
 - ▶ Total/Total - a PK de qualquer uma das relações migra como FK única e obrigatória (usar “[] + !” para definir a FK como única e obrigatória)
 - ▶ Os atributos do relacionamento (se houverem) migram com a PK

Adição de chave estrangeira (1:1)



Professor (CPF)

CPF → Pesquisador(CPF)

ProjetoCartao(serial, descricao, end_CEP, end_detalhamento, numero, saldo, [CPF] !)

CPF → Professor(CPF)

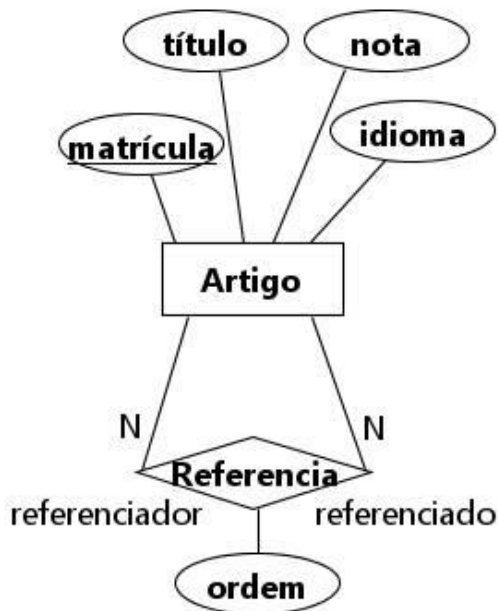
Criação de relação

- ▶ Características fortes (+) e fracas (-)
 - ▶ Reduz atributos opcionais (+)
 - ▶ Reduz testes para garantir a qualidade dos dados (+)
 - ▶ Exige junções (-)

Criação de relação

- ▶ Mapeamento (Melhor caso M:N)
 - ▶ Cada relacionamento M:N é mapeado para uma relação
 - ▶ As PK das relações envolvidas migram com FK
 - ▶ A composição das FK forma a PK da relação
 - ▶ Os atributos do relacionamento (se houverem) ficam na relação mapeada

Criação de relação



Referencia(referenciador, referenciado, ordem)
referenciador → Artigo(matrícula)
referenciado → Artigo(matrícula)

Criação de relação

- ▶ Mapeamento (Caso alternativo 1:N - evitar)
 - ▶ O relacionamento 1:N é mapeado para uma relação
 - ▶ As PK das relações envolvidas migram como FK
 - ▶ A FK do lado N torna-se PK
 - ▶ A outra FK torna-se obrigatória (usar “!” para definir a FK como obrigatória)
 - ▶ Os atributos do relacionamento (se houverem) ficam na relação mapeada

Criação de relação

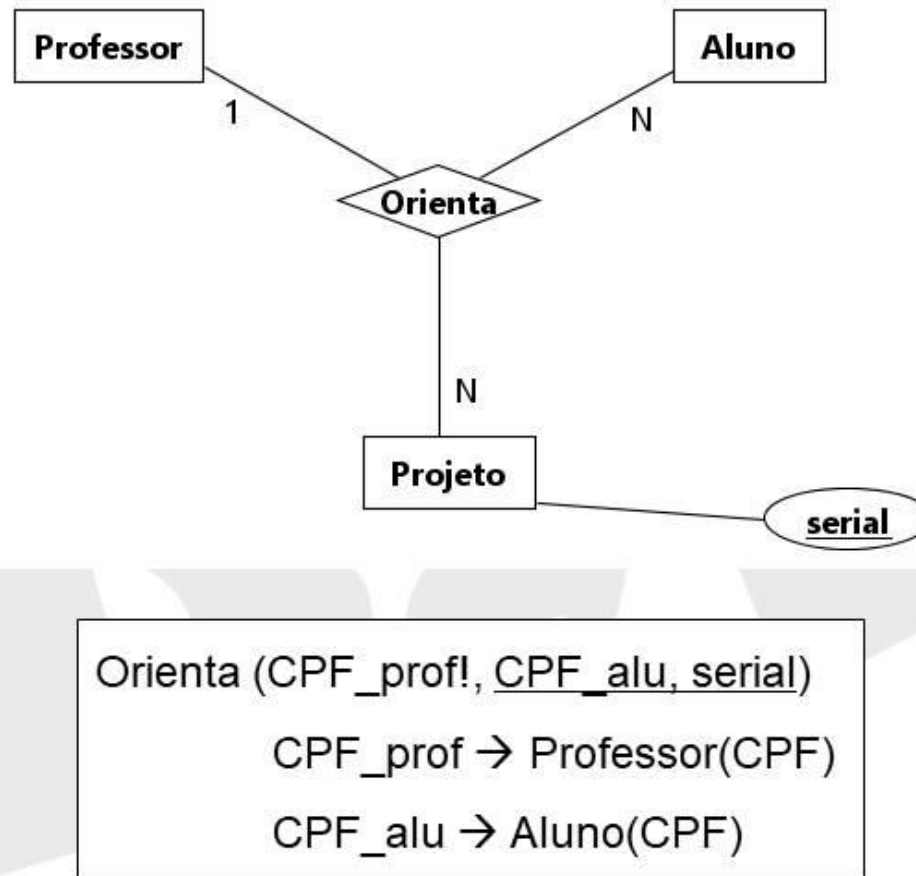
- ▶ Mapeamento (Caso alternativo 1:1 - evitar)
 - ▶ O relacionamento 1:1 é mapeado para uma relação
 - ▶ As PK das relações envolvidas migram como FK
 - ▶ A PK é definida a partir das participações do relacionamento
 - ▶ Parcial/Parcial - a PK pode ser qualquer uma das FK
 - ▶ Total/Parcial - a PK é a FK do lado parcial
 - ▶ Total/Total - a PK pode ser qualquer uma das FK
 - ▶ A outra FK torna-se única (AK) e obrigatória (usar “[] + !” para definir a FK como única e obrigatória)
 - ▶ Os atributos do relacionamento (se houverem) ficam na relação mapeada

Criação de relação (Relacionamentos N-ários)

► Mapeamento

- Cada relacionamento n-ário é mapeado para uma relação
- As PK das relações envolvidas migram com FK obrigatórias (usar “!” para definir a FK como única e obrigatória)
- Os atributos do relacionamento (se houverem) ficam na relação mapeada
- A PK da relação depende da cardinalidade do relacionamento
 - N:N:N → PK formada por todas as FK
 - 1:N:N → PK dupla formada pelas FK do lado N
 - 1:1:N → caso raro e complexo
 - 1:1:1 → caso raro e complexo

Criação de relação (Relacionamentos N-ários)



Caso Prático (Mapeando Entidades)

Projeto(serial, descricao, end_CEP, end_detalhamento);

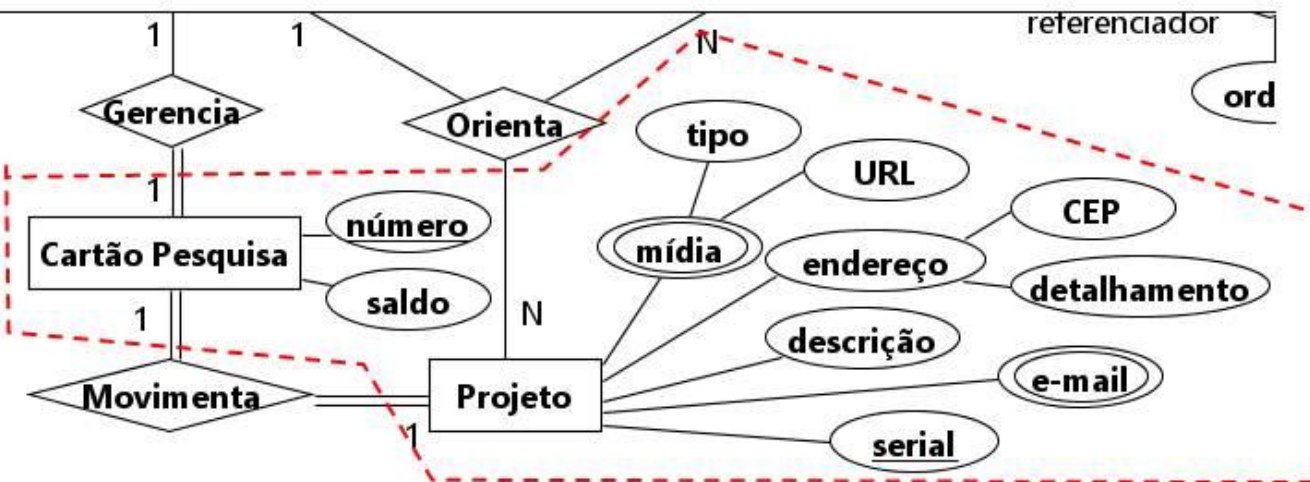
E-mail(serial, e-mail)

serial → Projeto(serial)

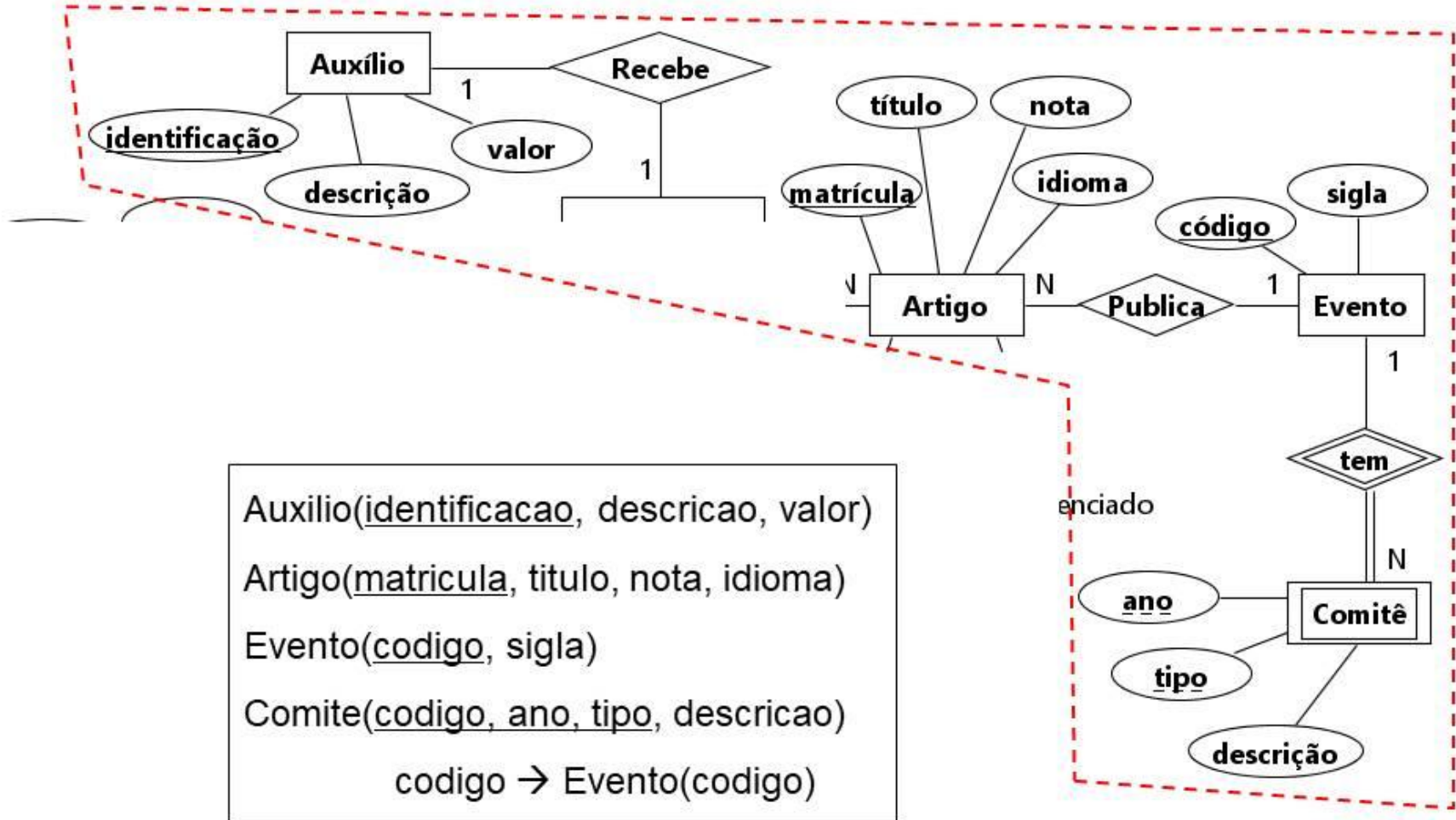
Midia(serial, tipo, URL)

serial → Projeto(serial)

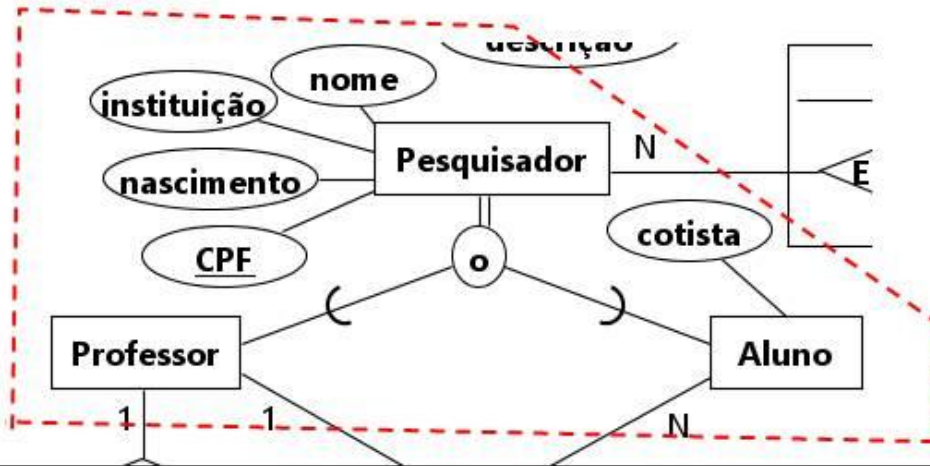
CartaoPesquisa(numero, saldo)



Caso Prático (Mapeando Entidades)



Caso Prático (Mapeando Herança)



Pesquisador(CPF, nascimento, instituicao, nome)

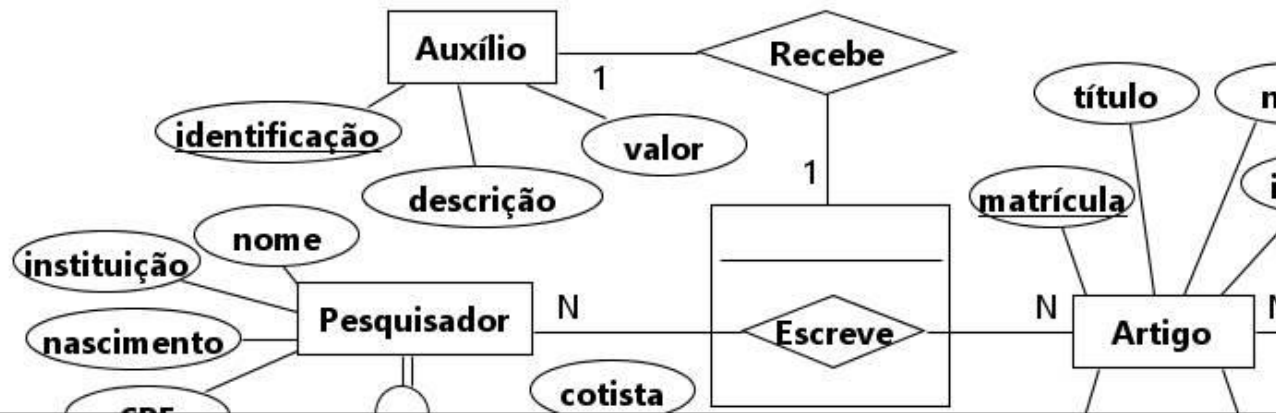
Professor(CPF)

CPF → Pesquisador (CPF)

Aluno(CPF, cotista)

CPF → Pesquisador (CPF)

Caso Prático (Mapeando E. Associativa)

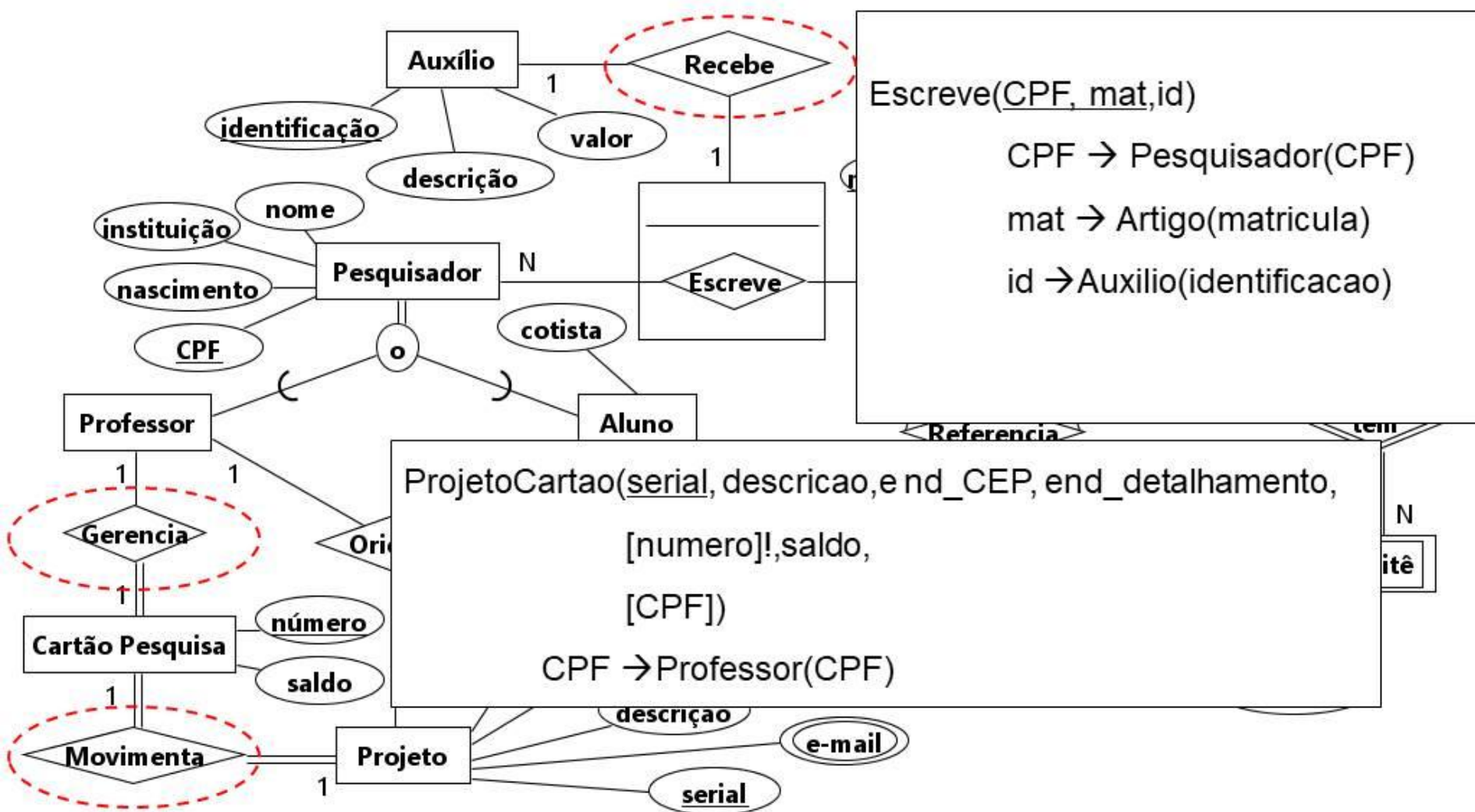


Escreve(CPF, matricula)

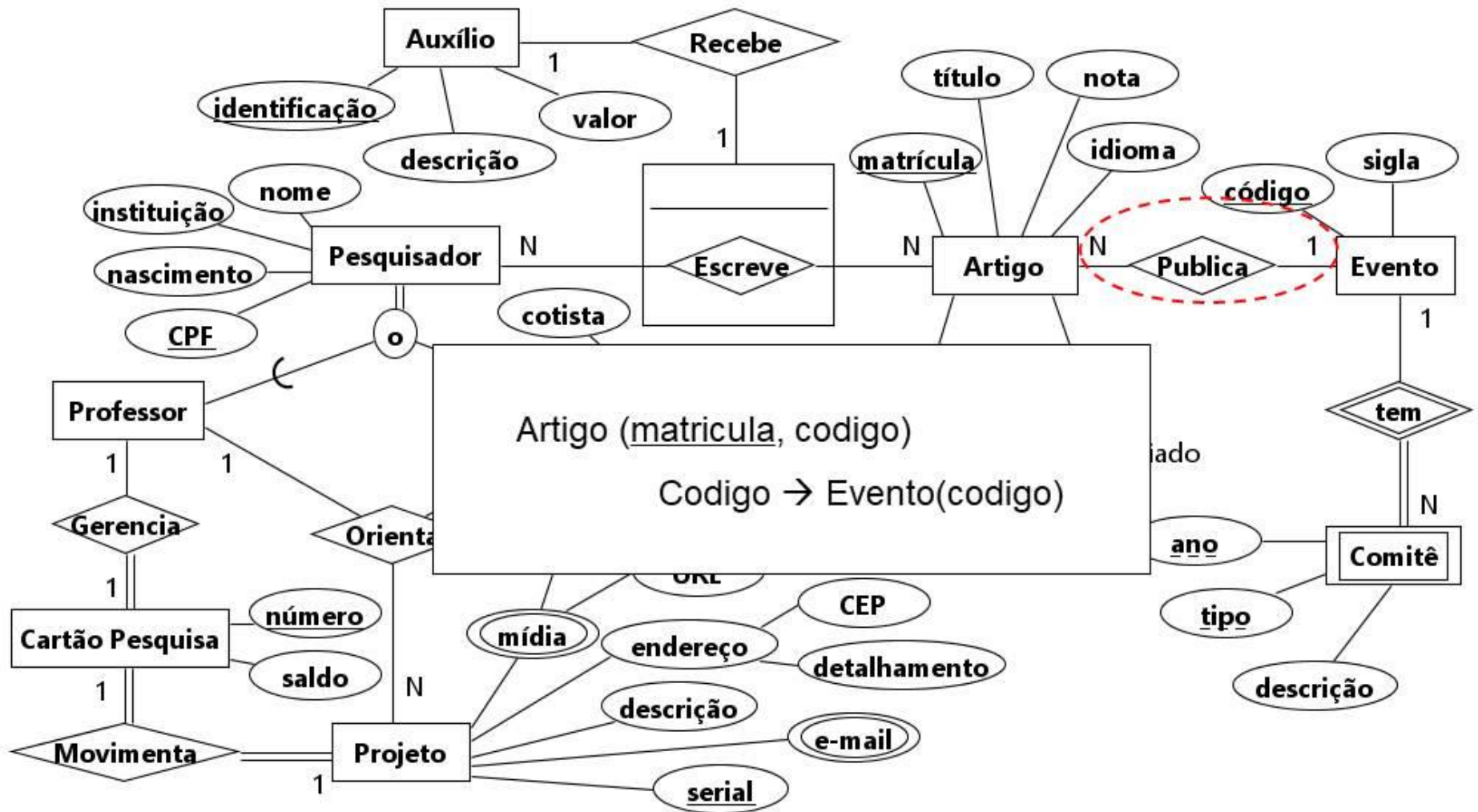
CPF → Pesquisador(CPF)

mat → Artigo(matricula)

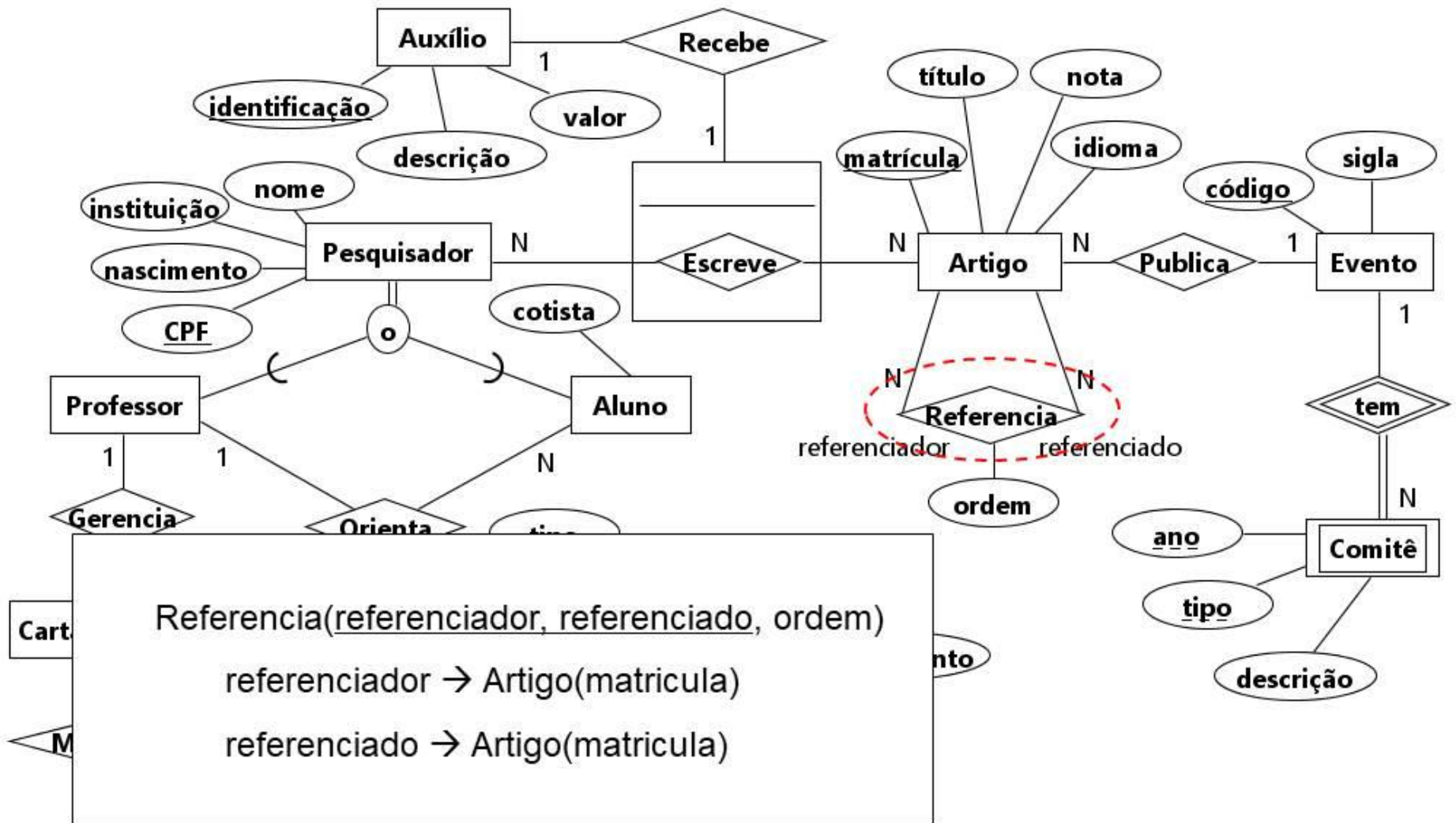
Caso Prático (Mapeando Relacionamento 1:1)



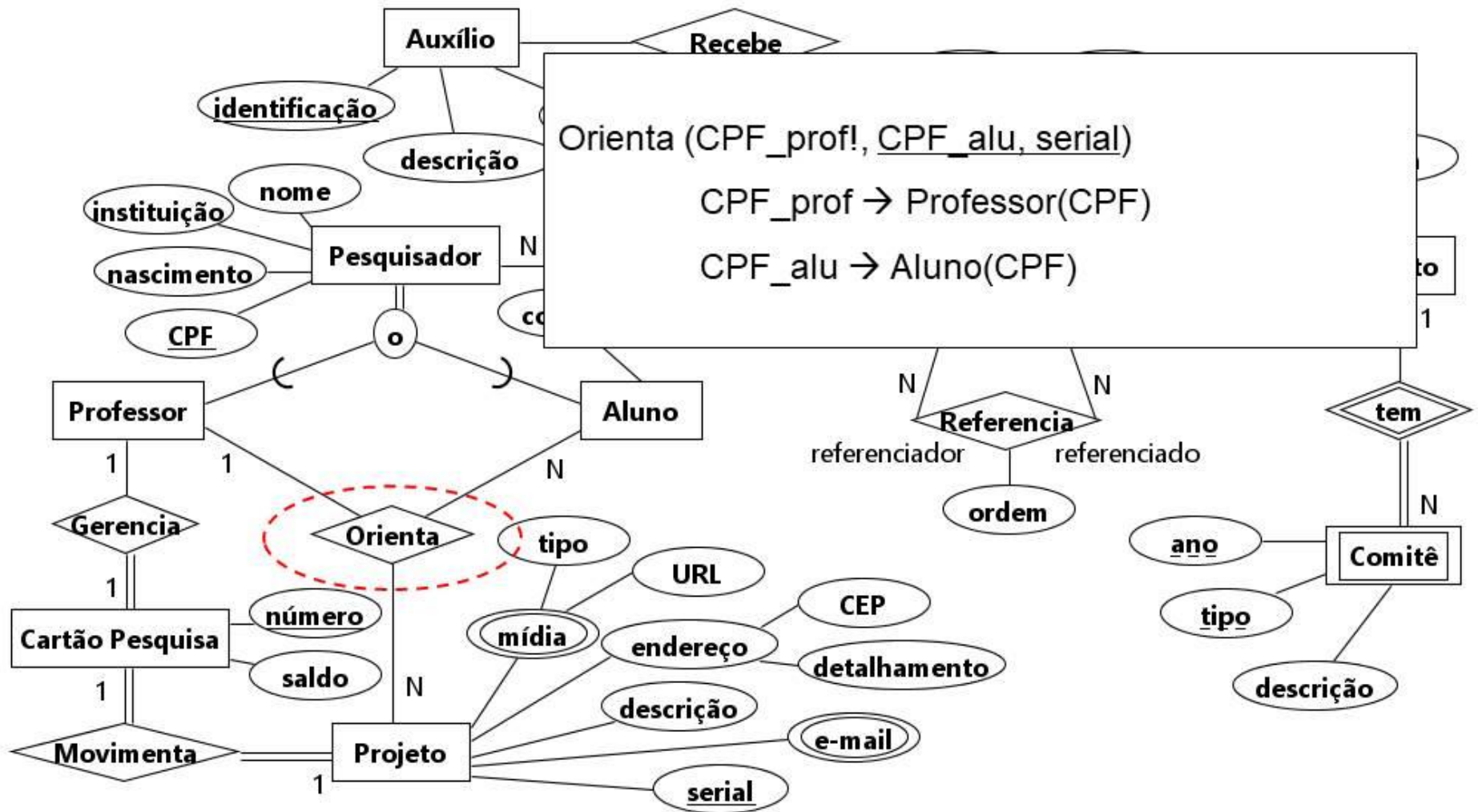
Caso Prático (Mapeando Relacionamento 1:N)



Caso Prático (Mapeando Relacionamento M:N)



Caso Prático (Mapeando Relacionamento N-ário)



cin.ufpe.br



UNIVERSIDADE FEDERAL DE PERNAMBUCO