

NExT – Nova Experiência de Trabalho

Módulo – Banco de Dados

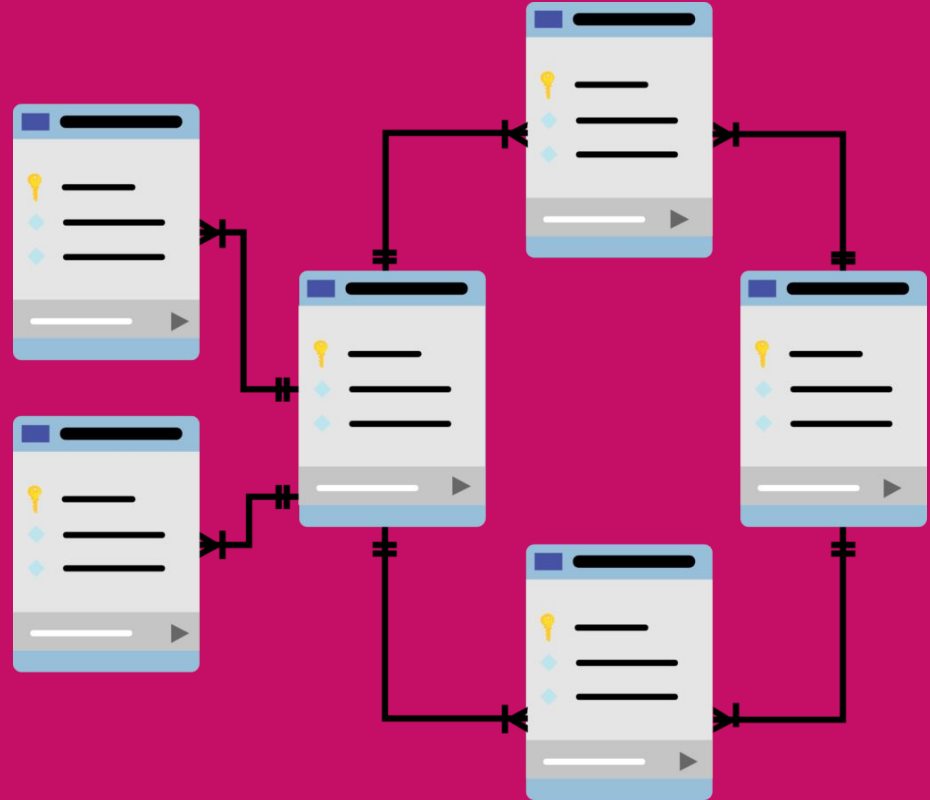
com add comentários!!!!

Gabrielle K. Canalle
gkc@cesar.school

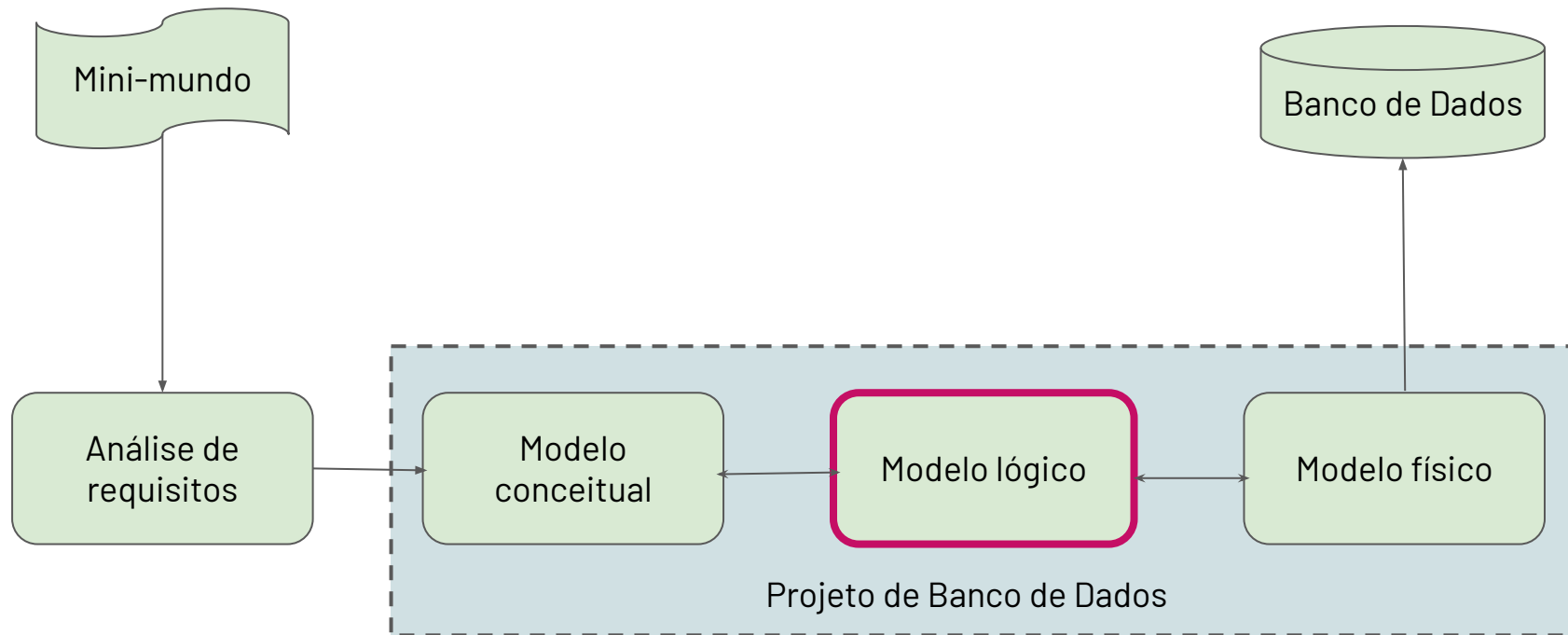
Natacha Targino
ntrsb@cesar.school



Modelo Lógico



Projeto de Banco de Dados



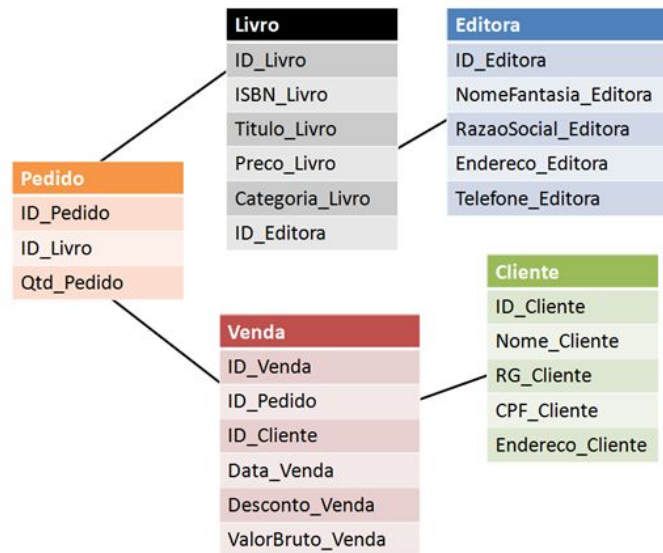
Modelagem Lógica

Abordagem Relacional

- O Banco de Dados Relacional é composto por tabelas ou relações, com **chaves primárias** e **estrangeiras**.

A chave primária é aquele atributo identificador

A chave estrangeira de uma tabela será sempre uma chave primaria de outra tabela



Modelagem Lógica

Tabela Livro

ID_Livro	ISBN	Título	Preço	Categoria	ID_Editora
00101	9786555442410	Ouvir, Agir e Encantar	41,70	Negócios e economia	3245
00102	9788588639171	Sistemas de Banco de Dados	232,99	Computação e Informática	2349
00103	9788576058816	Estruturas de Dados	159,38	Computação e Informática	2783

Chave Primária (PK)

Chave Estrangeira (FK)

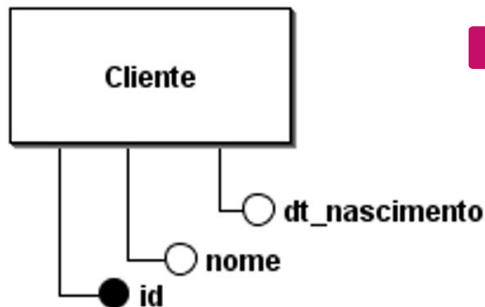
Linha
ou
Tupla
ou
Registro

Transformação do Modelo ER para o Relacional

Modelo ER	Modelo Relacional
Entidade	Tabela (Relação)
Instância de Entidade	Linha (Tupla)
Atributo	Coluna (Campo)
Atributo Multivalorado	Tabela Auxiliar
Atributo Identificador	Chave
Atributo Composto	Várias Colunas
Relacionamento	Ligações (ou Tabela)

Transformação Entidade e Atributos

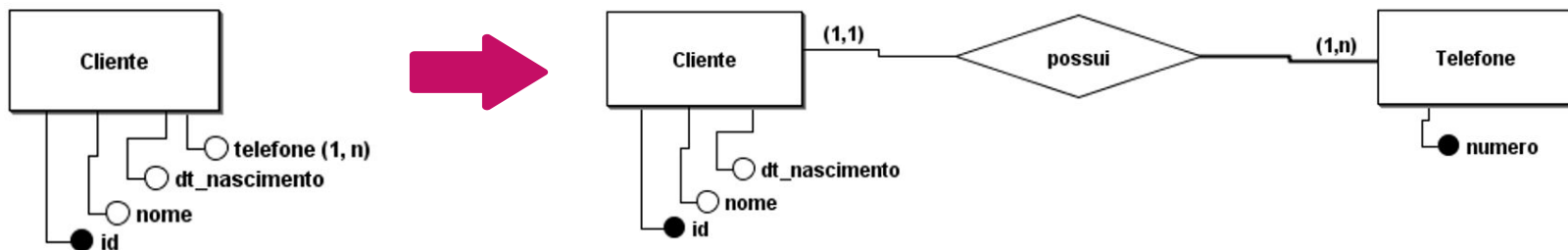
- Cada entidade é traduzida para uma **tabela**.
- Cada atributo (simples) da entidade define uma **coluna** da tabela.
- A coluna correspondente ao atributo identificador é **chave primária**



Cliente(id_cliente, nome, dt_nasc)

Transformação Atributos Multivalorados

1º Solução: Criar uma tabela para o atributo



Cliente (id_cliente, nome, dt_nasc)

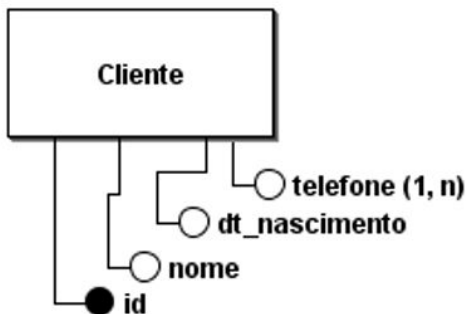
Telefone (id_cliente, id_numero, numero)

id_cliente referencia Cliente

Sempre lembrar de referenciar!

Transformação Atributos Multivalorados

2º Solução: Criar colunas para valores do atributo

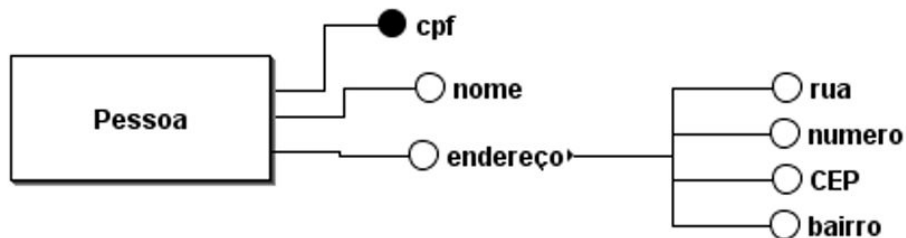


Indicada quando se sabe a quantidade máxima (**E também não permite muitos valores)

Cliente(id_cliente, nome, dt_nasc, num_telefone_01, num_telefone_02)

Transformação Atributos Compostos

Criar colunas para as partes do atributo



Pessoa(cpf, nome, rua, numero, cep, bairro)

Transformação de Relacionamentos

- Depende da cardinalidade das entidades envolvidas.
- Formas básicas de transformação:
 - **Tabela própria** para o relacionamento
 - **Colunas adicionais** dentro da tabela de entidade
 - **Fusão das entidades** em uma

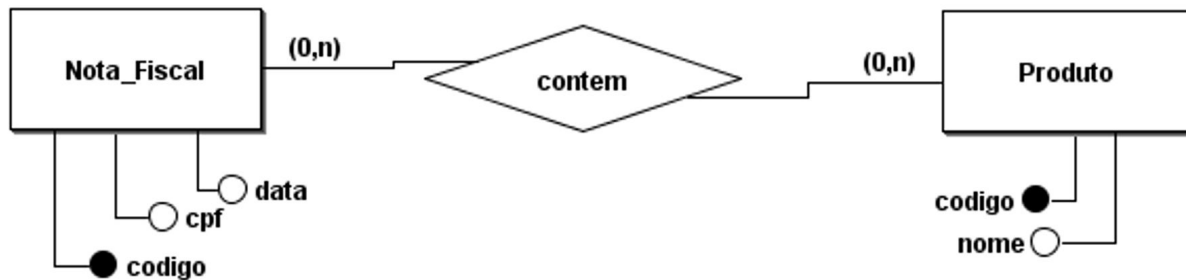
Transformação de Relacionamentos

Relacionamento N:N com ou sem Atributos

- **Criar sempre uma tabela própria**, as chaves estrangeiras das tabelas envolvidas formar a chave primária dessa nova tabela
 - Caso o relacionamento tenha **atributos**, são **criadas colunas** para esses atributos nessa nova tabela

Transformação de Relacionamentos

Relacionamento N:N SEM Atributos



NotaFiscal (cod_nf, cpf, data)

Produto (cod_pr, nome)

Contem (cod_nf, cod_pr)

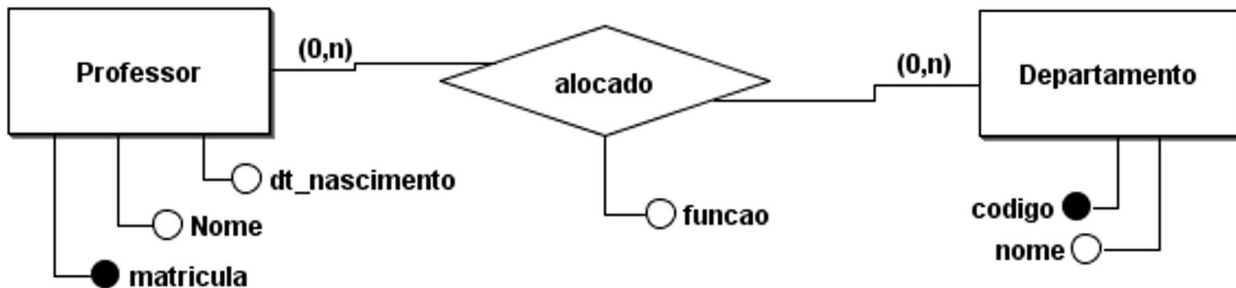
cod_nf referencia NotaFiscal

cod_pr referencia Produto

(nota fiscal contem varios produtos
e produtos podem estar contidos
em varias notas fiscais)

Transformação de Relacionamentos

Relacionamento N:N COM Atributos



Professor (matricula, nome, dt_nasc)

Departamento (codigo_d, nome)

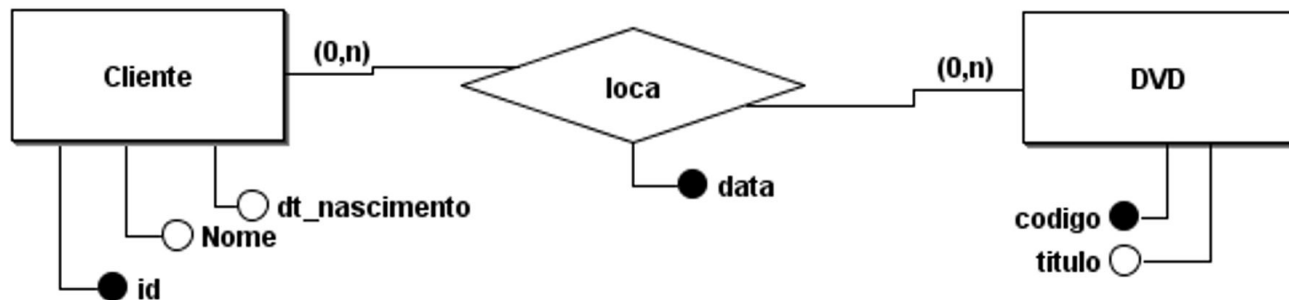
Alocação (matricula, codigo_d, funcao)

matricula referencia Professor

codigo_d referencia Departamento

Transformação de Relacionamentos

Relacionamento N:N COM Atributos



Cliente (id, nome, dt_nasc)

DVD (codigo, titulo)

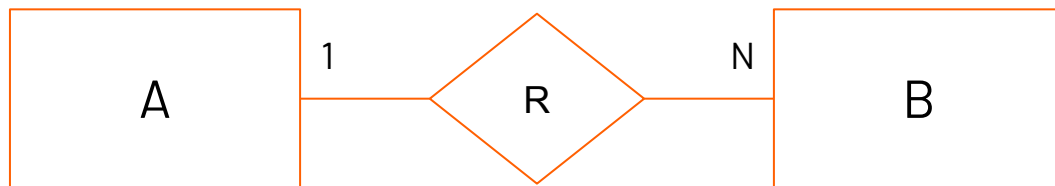
Locação (id, cod_dvd, data)

id referencia Cliente

cod_dvd referencia DVD(codigo)

Transformação de Relacionamentos

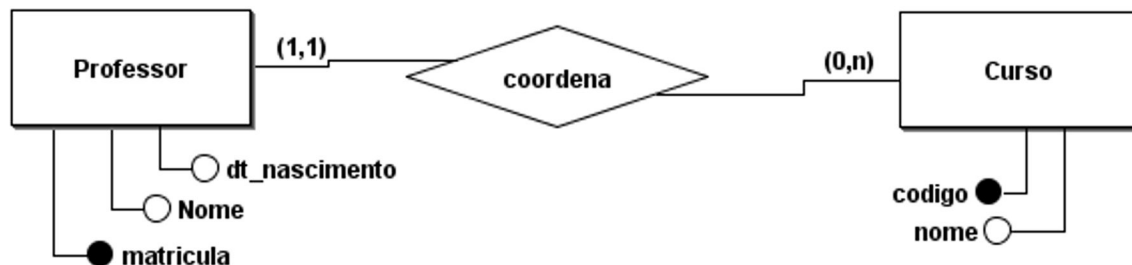
Relacionamento 1:N SEM Atributos



- **Adição de Coluna:** Acrescentar a chave primária da tabela A como chave estrangeira na tabela B
 - OBS.: a chave estrangeira na tabela B poderá ter ou não valor nulo, dependendo da opcionalidade

Transformação de Relacionamentos

Relacionamento 1:N SEM Atributos



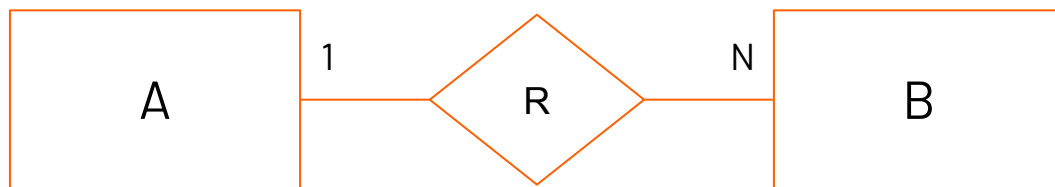
Professor (matricula, nome, dt_nasc)

Curso (codigo, nome, coordenador)

coordenador referencia Professor(matricula)

Transformação de Relacionamentos

Relacionamento 1:N COM Atributos

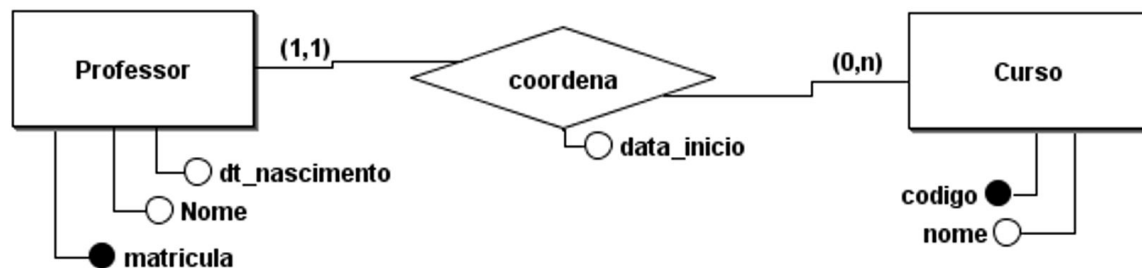


mais
comum

- **1º opção:** migrar os atributos do relacionamento para a tabela B
- **2º opção:** criar uma nova tabela para o relacionamento

Transformação de Relacionamentos

Relacionamento 1:N COM Atributos



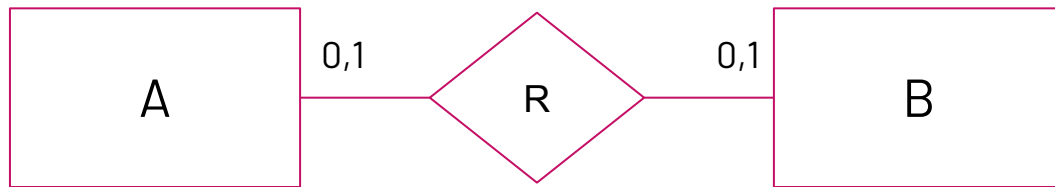
Professor(matricula, nome, dt_nasc)

Curso(codigo, nome, coordenador, data_inicio_coor)

coordenador referencia Professor(matricula)

Transformação de Relacionamentos

Relacionamento 0,1:0,1 SEM Atributos



- **1º opção:** Acrescentar a chave primária da tabela A como chave estrangeira da tabela B
- **2º opção:** Acrescentar a chave primária da tabela B como chave estrangeira da tabela A

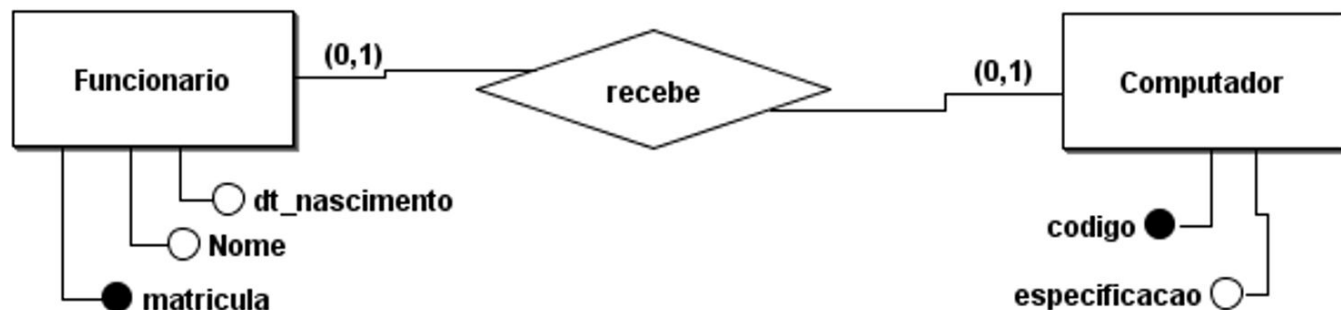
Transformação de Relacionamentos

Relacionamento 0,1:0,1 SEM Atributos

- Critérios para Escolha:
 - **1º Ver qual tabela nasce antes:** se A surge primeiro, então, migrar a chave primária de A para B
 - **2º Analisar qual entidade será mais manipulada, em nível de acesso:** se a tabela A será mais manipulada, colocar a chave primária de B nela
 - **3º Para desempate, observar qual a maior chave:** deverá ser migrada a menor

Transformação de Relacionamentos

Relacionamento 0,1:0,1 SEM Atributos



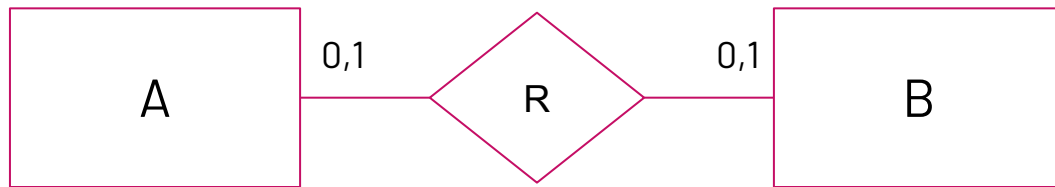
Computador (codigo, especificacao)

Funcionario (matricula, nome, dt_nasc, codigo_comp)

codigo_comp referencia Computador(codigo)

Transformação de Relacionamentos

Relacionamento 0,1:0,1 COM Atributos

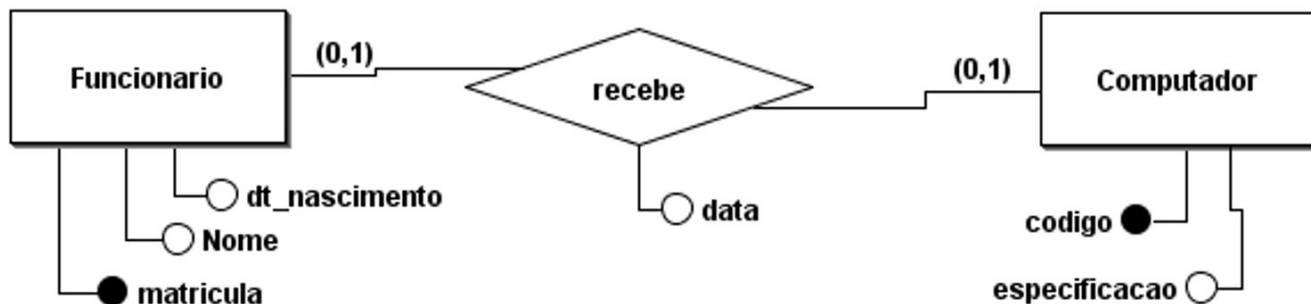


mais
comum

- **1º opção:** migrar os atributos do relacionamento para uma das tabelas
- **2º opção:** Criar uma nova tabela, agregando as chaves estrangeiras de A e B com os atributos do relacionamento

Transformação de Relacionamentos

Relacionamento 0,1:0,1 COM Atributos

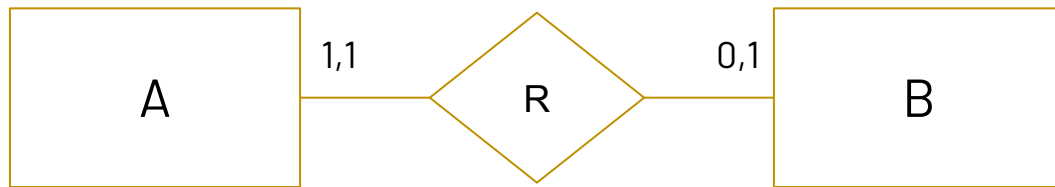


Computador(codigo, especificacao)

Funcionario(matricula, nome, dt_nasc, cod_comp, dt_comp)
cod_comp referencia Computador(codigo)

Transformação de Relacionamentos

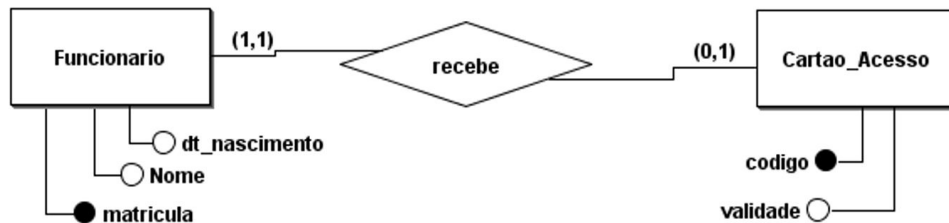
Relacionamento 1,1:0,1



- **Fusão das tabelas**
- **Adição de Coluna:** Acrescentar a chave primária da tabela A como chave estrangeira da tabela B

Transformação de Relacionamentos

Relacionamento 1,1:0,1



- **Esquemas Relacionais possíveis:**

a) **Funcionario**(matricula, nome, dt_nasc)

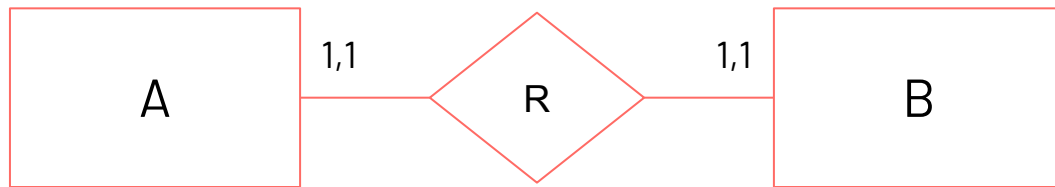
Cartao_Acesso(codigo, validade, matricula)

matricula referencia Funcionario

b) **Funcionario**(matricula, nome, dt_nasc, codigo_ct, validade_ct)

Transformação de Relacionamentos

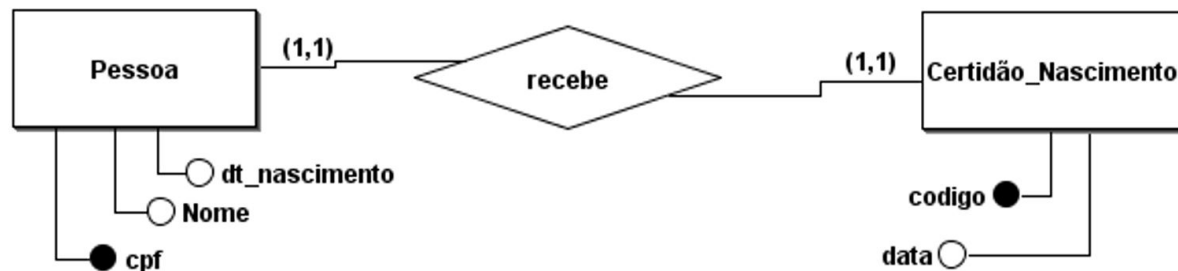
Relacionamento 1,1:1,1



- **Fusão das tabelas**

Transformação de Relacionamentos

Relacionamento 1,1:1,1



Pessoa(cpf, nome, dt_nasc, cod_certidão, data_certidão)

Transformação de Relacionamentos

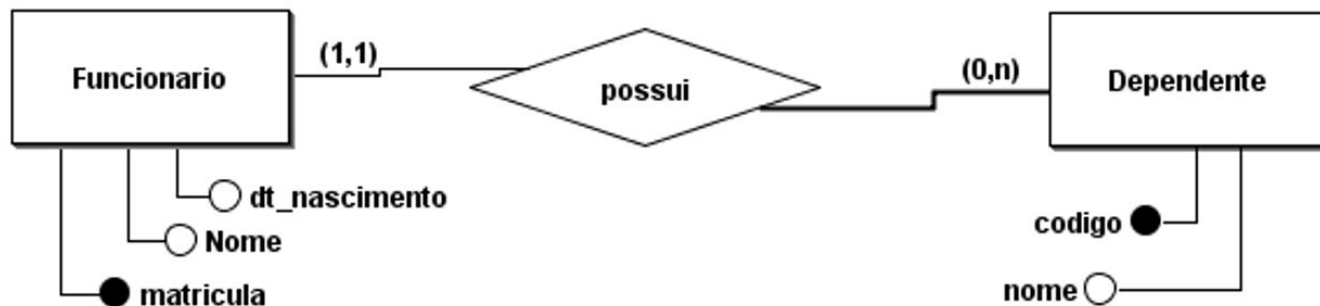
Resumo

Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição coluna	Fusão tabelas
Relacionamentos 1:1			
	±	✓	×
	±	±	✓
	±	±	✓
Relacionamentos 1:n			
	±	✓	×
	±	✓	×
	±	✓	×
	±	✓	×
Relacionamentos n:n			
	✓	×	×
	✓	×	×
	✓	×	×

✓ Alternativa preferida
 ± Pode ser usada – 1ª opção
 × Não usar
 ± Pode ser usada – 2ª opção

Fonte: Heuser, C. A. Projeto de Banco de Dados (6ª edição)

Entidade Fraca/Relacionamento Identificador



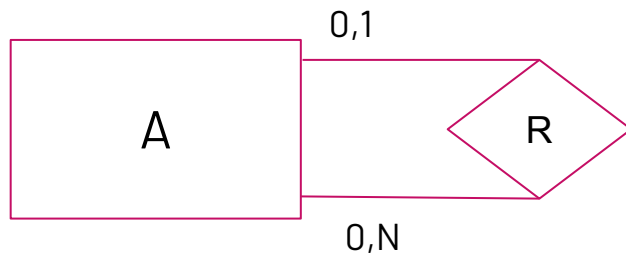
Funcionario (matricula, nome, dt_nasc)

Dependente (matricula, codigo, nome)


matricula referencia Funcionario

A chave primária
da entidade fraca
é **Composta**

Derivação de Auto Relacionamentos



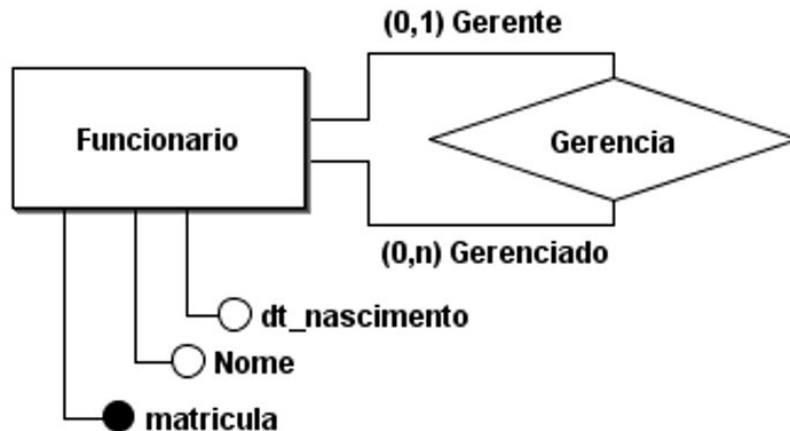
- Etapas:

- **1º:** Considere como se fosse um relacionamento entre duas entidades, ou seja: The diagram shows two separate rectangular entity boxes, both labeled 'A', connected by a diamond-shaped relationship box labeled 'R'. The line from the left 'A' to 'R' is labeled '0,1', and the line from 'R' to the right 'A' is labeled '0,N'.

- **2º:** Considere as regras de derivação, usando no papel das entidades A e B, a mesma entidade A

Derivação de Auto Relacionamentos

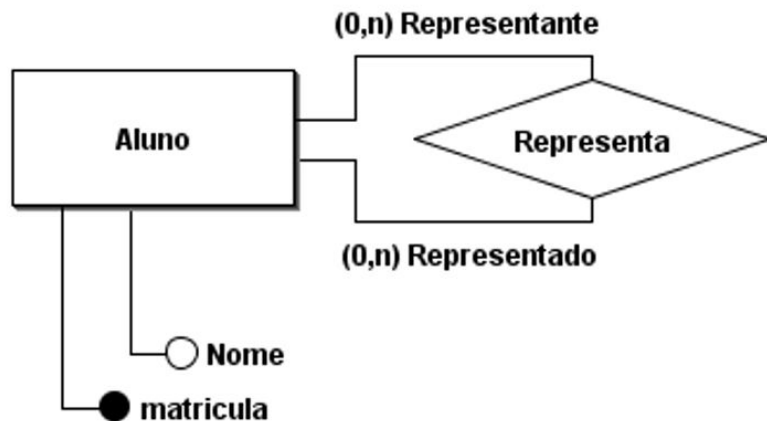
Auto Relacionamento 1:N



Funcionario(matricula, nome, dt_nasc, mat_gerente)
mat_gerente referencia Funcionario(matricula)

Derivação de Auto Relacionamentos

Auto Relacionamento N:N



Aluno (matricula, nome)

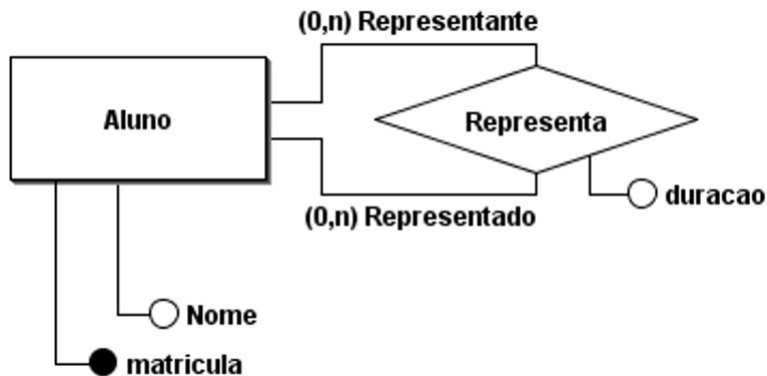
Representacao (mat_representante, mat_representado)

mat_representante referencia Aluno(matricula)

mat_representado referencia Aluno(matricula)

Derivação de Auto Relacionamentos

Auto Relacionamento N:N COM atributo



Aluno (matricula, nome)

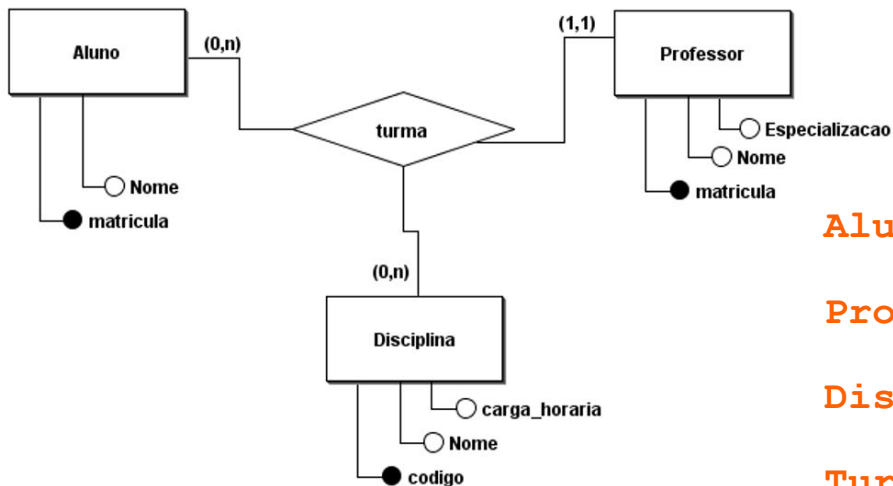
Representacao (mat_representante, mat_representado, duracao)

mat_representante referencia Aluno(matricula)

mat_representado referencia Aluno(matricula)

Derivação de Relacionamento com grau maior que 2

Relacionamentos Ternários - N:N:1



Uma tabela é criada para o relacionamento Ternário

Aluno (mat_alu, nome)

Professor (mat_prof, nome, especializacao)

Disciplina (codigo, nome, carga_horaria)

Turma (mat_alu, codigo, mat_prof)

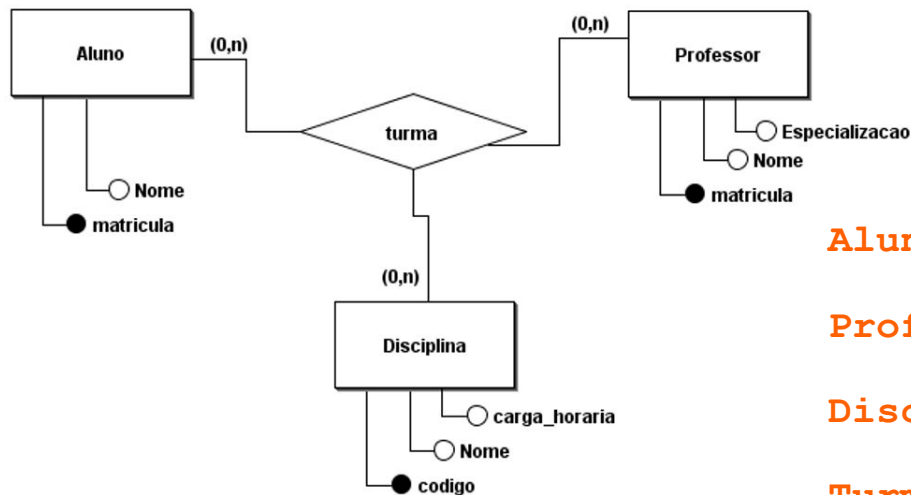
mat_alu referencia Aluno

codigo referencia Disciplina

mat_prof referencia Professor

Derivação de Relacionamento com grau maior que 2

Relacionamentos Ternários - N:N:N



Uma tabela é criada para o relacionamento Ternário

Aluno (mat_alu, nome)

Professor (mat_prof, nome, especializacao)

Disciplina (codigo, nome, carga_horaria)

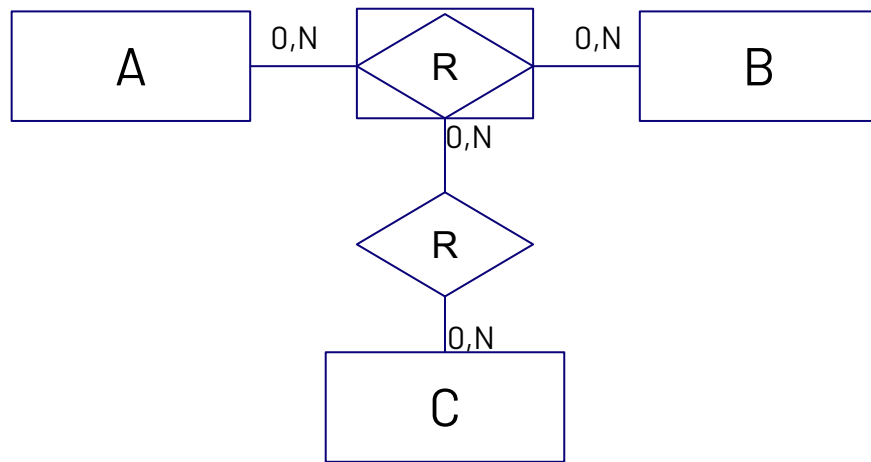
Turma (mat_alu, codigo, mat_prof)

mat_alu referencia Aluno

codigo referencia Disciplina

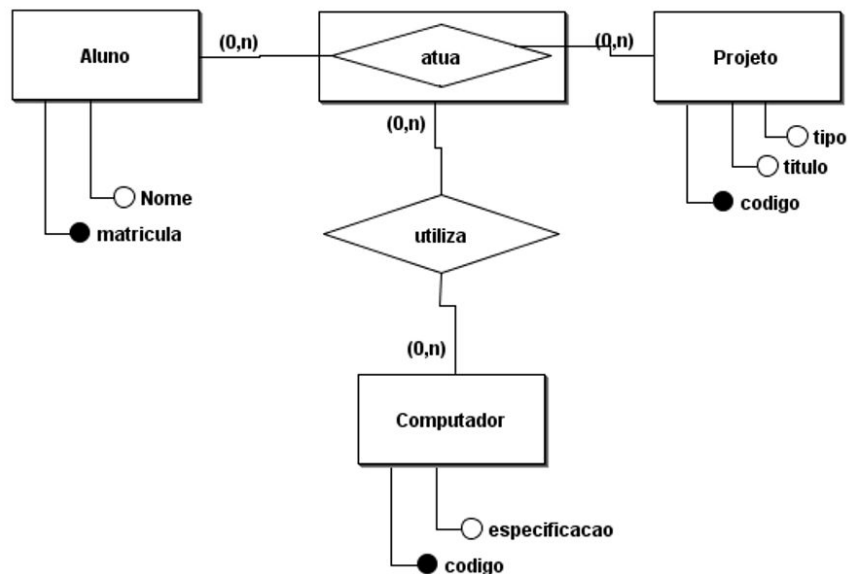
mat_prof referencia Professor

Derivação de Agregação/Entidade Associativa



- Etapas:
 - **1º:** Criar uma tabela referente ao relacionamento entre as tabelas A e B
 - **2º:** Analisar o relacionamento dessa tabela criada e a tabela C, para a relação essas duas tabelas considere as regras de derivação vistas

Derivação de Agregação/Entidade Associativa



Aluno(matricula, nome)

Projeto(cod_proj, titulo, tipo)

Computador(cod_comp, especificacao)

Atuacao(mat, cod_proj)

mat referencia Aluno(matricula)

cod_proj referencia projeto

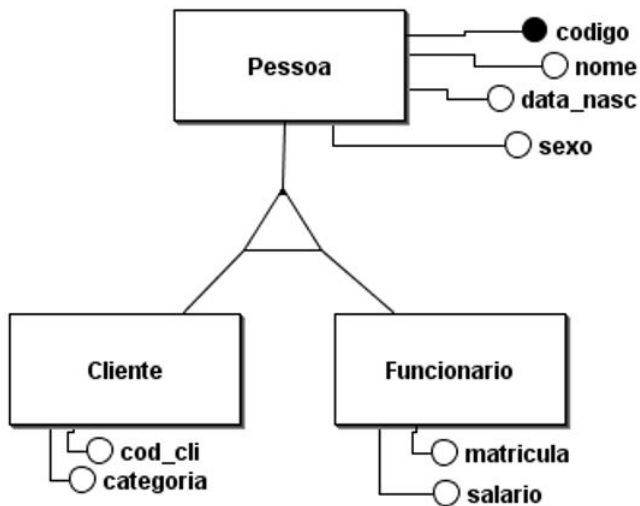
Uso(mat, cod_proj, cod_comp)

mat,cod_proj referencia Atuacao

cod_comp referencia Computador

Derivação de Estruturas de Generalização/Especialização

- 1ª opção: Criar uma tabela para a entidade generalizada e outra tabela para cada entidade especializada



Pessoa (codigo, nome, data_nasc, sexo)

Cliente (codigo, cod_cli, categoria)

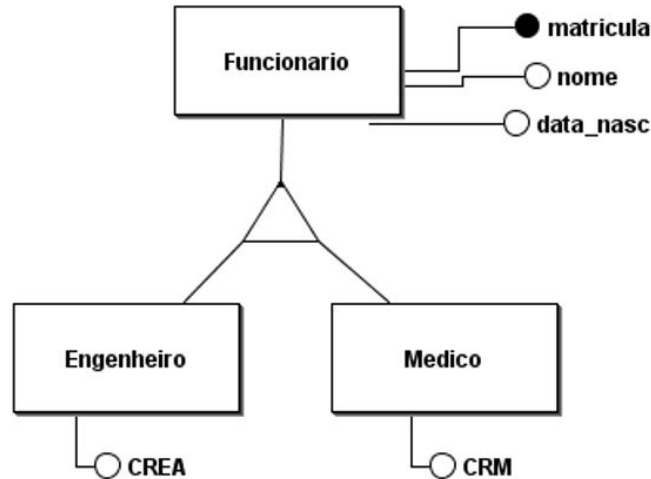
codigo referencia Pessoa

Funcionario (codigo, mat, salario)

codigo referencia Pessoa

Derivação de Estruturas de Generalização/Especialização

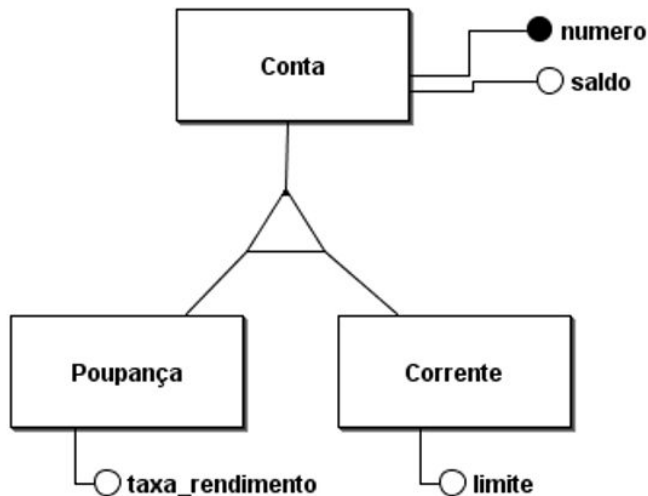
- **2ª opção: Criar somente uma tabela para a entidade generalizada e migrar todos os atributos e relacionamentos para essa tabela**



Funcionario (matricula, nome, data_nasc, CREA, CRM)

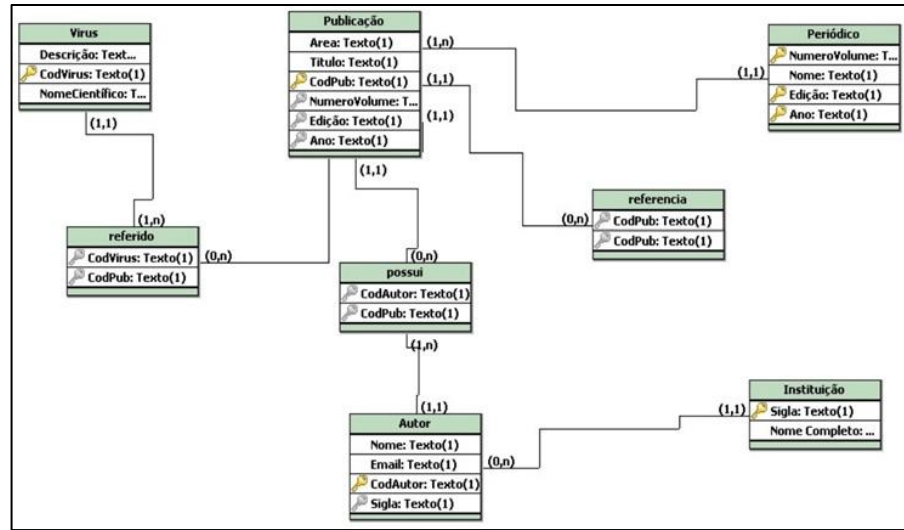
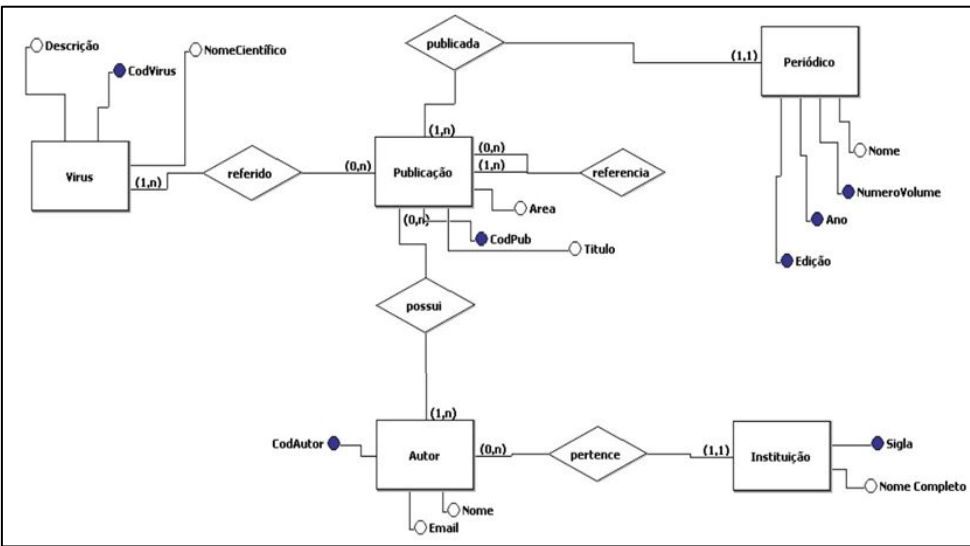
Derivação de Estruturas de Generalização/Especialização

- **3ª opção: Criar somente tabelas para as entidades especializadas** e migrar todos os atributos e relacionamentos generalizados para cada uma dessas tabelas

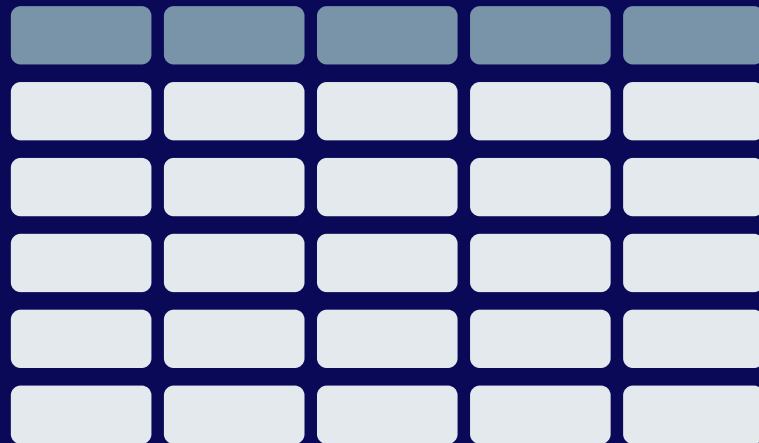


Conta_Poupança (num, saldo, taxa)

Conta_Corrente (num, saldo, limite)



Normalização



Normalização

- É o processo de **decomposição** de um esquema de relação em outros esquemas de relação
- Os esquemas resultantes devem **preservar a semântica original** (restrições de integridade, dados e relacionamentos)
- Tabelas normalizadas representam de maneira melhor uma realidade modelada e atendem ao modelo relacional.

Normalização

- **Objetivos:**

- Eliminar redundâncias
- Minimizar anomalias de inserção, remoção e atualização
- Garantir que as dependências entre os dados façam sentido
- Obtenção de um modelo ER

- **Benefícios:**

- O espaço de armazenamento dos dados diminui
- A descrição do BD será imediata
- A tabela pode ser atualizada com maior eficiência
- A reversão para o modelo conceitual é possível

Normalização

1FN: Uma tabela está na primeira forma normal, quando ela

NÃO CONTÉM tabelas aninhadas

(Tabela Aninhada: **Grupo repetido** ou **coluna não atômica** ou **coluna multi-valorada**, que não apresenta valores atômicos)

Normalização: 1FN

Cod_Disciplina	Nome_Disciplina	Cod_Prof	Nome_Prof	Créditos	Horas
IDC09	Banco de Dados I	IP032	Natacha Targino	06	100
		IP059	Gabrielle Canalle		
IDC12	Banco de Dados II	IP098	João Vicente	04	67
		IP059	Gabrielle Canalle		
		IP044	Vinícius Filho		
IDC07	Programação	IP098	João Vicente	04	67
IDC16	Estrutura de Dados	IP032	Natacha Targino	06	100
		IP029	Mariana Rachel		

Disciplina (Cod_Disciplina, Nome_Disciplina, (Cod_Prof, Nome_Prof),
Créditos, Horas)

Normalização: 1FN

Cod_Disciplina	Nome_Disciplina	Créditos	Horas
IDC09	Banco de Dados I	06	100
IDC12	Banco de Dados II	04	67
IDC07	Programação	04	67
IDC16	Estrutura de Dados	06	100

Cod_Disciplina	Cod_Prof	Nome_Prof
IDC09	IP032	Natacha Targino
IDC09	IP059	Gabrielle Canalle
IDC12	IP098	João Vicente
IDC12	IP059	Gabrielle Canalle
IDC12	IP044	Vinícius Filho
IDC07	IP098	João Vicente
IDC16	IP032	Natacha Targino
IDC16	IP029	Mariana Rachel

Disciplina (Cod_Disciplina, Nome_Disciplina, Créditos, Horas)

Disciplina_Prof (Cod_Disciplina, Cod_Prof, Nome_Prof)

Cod_Disciplina referencia Disciplina

Normalização

2FN: Uma tabela está na segunda forma normal, quando ela **está na 1NF** e todo atributo não-chave é plenamente dependente da chave primária, ou seja,

NÃO Contém Dependência Funcional Parcial

(Dependência Funcional Parcial: Ocorre quando uma coluna **depende** apenas de **parte** de uma **chave primária composta**)

Normalização: 2FN

Cod_Disciplina	Cod_Prof	Nome_Prof
IDC09	IP032	Natacha Targino
IDC09	IP059	Gabrielle Canalle
IDC12	IP098	João Vicente
IDC12	IP059	Gabrielle Canalle
IDC12	IP044	Vinícius Filho
IDC07	IP098	João Vicente
IDC16	IP032	Natacha Targino
IDC16	IP029	Mariana Rachel

Disciplina_Prof (Cod_Disciplina, Cod_Prof, Nome_Prof)

Cod_Disciplina referencia Disciplina

Normalização: 2FN

Cod_Prof	Nome_Prof
IP032	Natacha Targino
IP059	Gabrielle Canalle
IP098	João Vicente
IP044	Vinícius Filho
IP029	Mariana Rachel

Cod_Disciplina	Cod_Prof
IDC09	IP032
IDC09	IP059
IDC12	IP098
IDC12	IP059
IDC12	IP044
IDC07	IP098
IDC16	IP032
IDC16	IP029

Prof (Cod_Prof, Nome_Prof)

Disciplina_Prof (Cod_Disciplina, Cod_Prof)

Cod_Disciplina referencia Disciplina

Cod_Prof referencia Prof

Normalização

3FN: Uma tabela está na segunda forma normal, quando ela **está na 2NF** e nenhum atributo não-chave é transitivamente dependente da chave primária ou seja,

NÃO Contém Dependência Funcional Transitiva

(Dependência Funcional Transitiva: Ocorre quando uma coluna, além de depender da chave primária da tabela, **depende de outra coluna** ou conjunto de colunas da tabela)

Normalização: 3FN

Cod_Disciplina	Nome_Disciplina	Créditos	Horas
IDC09	Banco de Dados I	06	100
IDC12	Banco de Dados II	04	67
IDC07	Programação	04	67
IDC16	Estrutura de Dados	06	100

Disciplina (Cod_Disciplina, Nome_Disciplina, Créditos, Horas)

Normalização: 3FN

Créditos	Horas
06	100
04	67

Cod_Disciplina	Nome_Disciplina	Créditos
IDC09	Banco de Dados I	06
IDC12	Banco de Dados II	04
IDC07	Programação	04
IDC16	Estrutura de Dados	06

Creditos(creditos, horas)

Disciplina(Cod_Disicplina, Nome_Disciplina, Créditos)

Creditos referencia Creditos

Normalização

Antes

Cod_Disciplina	Nome_Disciplina	Cod_Prof	Nome_Prof	Créditos	Horas
IDC09	Banco de Dados I	IP032	Natacha Targino	06	100
		IP059	Gabrielle Canalle		
IDC12	Banco de Dados II	IP098	João Vicente	04	67
		IP059	Gabrielle Canalle		
		IP044	Vinícius Filho		
IDC07	Programação	IP098	João Vicente	04	67
IDC16	Estrutura de Dados	IP032	Natacha Targino	06	100
		IP029	Mariana Rachel		

X

Depois

Cod_Disciplina	Nome_Disciplina	Créditos
IDC09	Banco de Dados I	06
IDC12	Banco de Dados II	04
IDC07	Programação	04
IDC16	Estrutura de Dados	06

Cod_Prof	Nome_Prof
IP032	Natacha Targino
IP059	Gabrielle Canalle
IP098	João Vicente
IP044	Vinícius Filho
IP029	Mariana Rachel

Cod_Disciplina	Cod_Prof
IDC09	IP032
IDC09	IP059
IDC12	IP098
IDC12	IP059
IDC12	IP044
IDC07	IP098
IDC16	IP032
IDC16	IP029

Créditos	Horas
06	100
04	67

Dúvidas?

Gabrielle K. Canalle
gkc@cesar.school

Natacha Targino
ntrsb@cesar.school

