Resumo 2

Banco de dados 1

Marcos Geraldo Braga Emiliano 19.1.4012

Fonte: Slides das aulas

**Modelo relacional**

Introduzido por Codd em 1970 (pesquisa da IBM), modelo formal fundamentado nos conceitos de uma relação matemática (teoria de conjuntos), o modelo relacional representa o banco de dados como uma coleção de relações. Informalmente, cada relação é semelhante a uma tabela, cada linha da tabela representa uma coleção de valores de dados relacionados, que podem ser interpretados como fatos descrevendo uma entidade ou um relacionamento, o nome da tabela e os nomes das colunas são usados para ajudar na interpretação do significado dos valores em cada linha da tabela.

**Conceitos Básicos**

Na terminologia do modelo relacional, uma linha é chamada de uma tupla, um cabeçalho de coluna é chamado de um atributo, e uma tabela é chamada de uma relação. O tipo de dado que descreve os valores que um atributo pode ter é chamado de domínio. Um domínio D é um conjunto de valores atômicos (indivisíveis). Esquema de Relação: é uma expressão da forma R (A1, A2, ..., An), onde:

R: nome da relação;

Ai: nome de um atributo, cujo domínio em R é denotado por dom (Ai);

n: grau da relação;

Relação ou instância de uma relação: uma relação r de um esquema R, denotado por r(R), é um conjunto de tuplas: r = {t1, t2, ..., tn}, cada tupla é uma lista ordenada de n valores: t = <v1, v2, ..., vn>, onde cada vi, 1 ≤ i ≤ n, é um elemento do domínio dom (Ai) ou um valor especial nulo. Formalmente, uma relação r(R) é um subconjunto do produto cartesiano dos domínios que definem R: r(R) ⊆ (dom(A1) X dom(A2) X ... X dom (An)), o produto cartesiano especifica todas as possíveis combinações de valores dos domínios fundamentais.

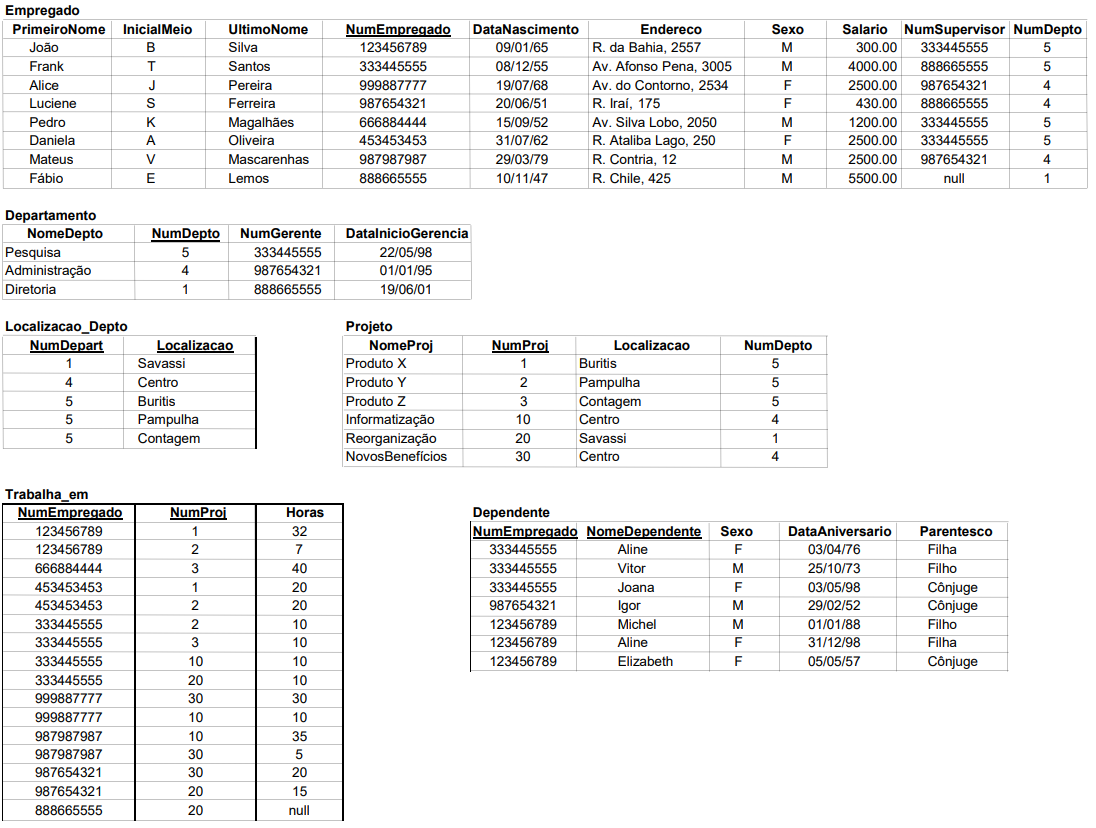
**Características das Relações**

A ordem das tuplas é irrelevante, a ordem dos valores dentro de uma tupla é relevante, a menos que se estabeleça uma correspondência entre esses valores e os atributos definidos. Exemplo: t = < (nome, BD), (código, 032), (cargaHorária, 72), (numCreditos, 4)>, o valor de cada atributo em uma tupla é atômico, atributos compostos e multi-valorados não são permitidos e as tuplas de uma relação são únicas.

**Esquema de um Banco de Dados Relacional**

Um esquema S de um banco de dados relacional define um conjunto de esquemas de relação R = {R1, R2, ..., Rn} e um conjunto de restrições de integridade I = {I1, I2, ..., I­m}. Uma instância BD de S é um conjunto de instâncias de relação BD = {r1,r2,...,rn}, tal que cada ri é uma instância de Ri e as relações ri satisfazem as restrições de integridade em I. Portanto, pode-se dizer que S = (R, I).

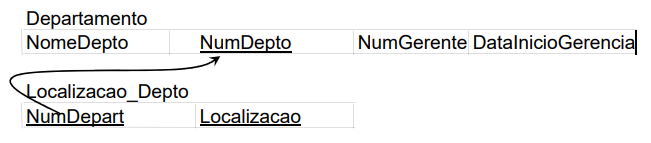
**Exemplo de Instância de um Banco de Dados**

****

**Restrições do Modelo Relacional**

**Restrição de Domínio**:

O valor de cada atributo deve ser um valor atômico no domínio do atributo, **Restrição de Chave**: uma chave é um conjunto mínimo de valores dos atributos que identifica unicamente uma tupla, se um esquema de relação tem mais de uma chave, cada uma é chamada **chave candidata**. Uma das chaves candidatas é arbitrariamente escolhida para ser a **chave primária**, e as outras são chamadas **chaves alternativas** ou **chaves secundárias**, é melhor escolher como chave primária aquela com o menor número de atributos, cada esquema de relação R deve ter uma chave primária que é indicada no esquema por um sublinhado. **Restrição de Integridade de Entidade**: a chave primária de um esquema de relação R não pode ter valor nulo. **Restrição de Integridade Referencial**: é uma restrição especificada entre duas relações, sendo usada para manter a consistência entre as tuplas das duas relações. Usa-se o conceito de chave estrangeira para definir tais restrições, um conjunto de atributos FK em um esquema de relação R1 é uma chave estrangeira de R1 que referência uma relação R2 se satisfaz as duas regras seguintes: os atributos de FK referenciam a chave primária PK de R2, tendo os mesmos domínios dos atributos de PK e um valor de FK na tupla t1 da instância r1(R1) ocorre como um valor de PK para alguma tupla t2 da instância r2(R2) ou é nulo. As restrições de integridade referencial são obtidas, geralmente, a partir dos relacionamentos entre as entidades representadas pelos esquemas de relação. As restrições de integridade referencial podem ser representadas graficamente usando-se setas que partem de uma chave estrangeira para a chave primária na relação referenciada.



**Integridade Referencial com Opção de Exclusão**

Notação: R1[fk] op→ R2[pk], onde "op" é a opção de exclusão, dentre as seguintes: bloqueio (restrict): se alguma tupla referência a tupla a ser excluída, através de uma chave estrangeira, a exclusão não é efetuada, propagação (cascade): todas as tuplas que referenciam a tupla a ser excluída, através de uma chave estrangeira, são excluídas também automaticamente e substituição por nulos (set null): todas as tuplas que referenciam a tupla a ser excluída, através de uma chave estrangeira, têm os valores dos atributos da chave estrangeira modificados para nulo (se for permitido nulo) e a exclusão é efetuada.

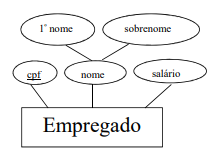
**Operações de Atualização sobre Relações**

Inserção (insert): insere novas tuplas em uma relação. pode violar qualquer dos quatro tipos de restrições discutidas, exclusão (delete): exclui tuplas de uma relação, pode violar somente restrição de integridade referencial e modificação (update ou modify): muda os valores de modificação (update ou modify): muda os valores de alguns atributos em tuplas existentes, modificar um atributo que não é chave primária nem chave estrangeira pode violar somente a restrição de domínio, modificar a chave primária é similar a excluir uma tupla e inserir uma outra no seu lugar; assim, pode violar qualquer das quatro restrições discutidas ou modificar um atributo de uma chave estrangeira pode violar a restrição de integridade referencial ou de domínio.

**Mapeamento ERE para Relacional**

**Entidade**:

Para cada tipo de entidade E no esquema ERE, crie uma relação R que inclua todos os atributos simples de E, inclua também os atributos simples componentes de um atributo composto de E na relação R, Escolha uma das chaves candidatas de E para ser a chave primária de R, exemplo: para o tipo de entidade Empregado, gere a relação:

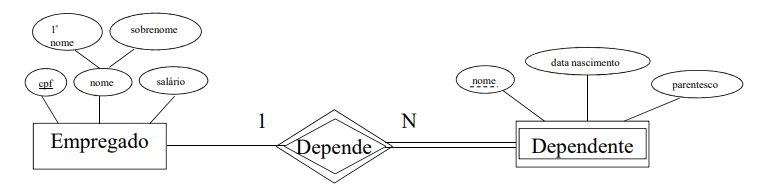


Empregado (cpf, primeironome, sobrenome, salario);

**Entidade Fraca**:

Para cada tipo de entidade fraca W no esquema ERE, crie uma relação R e inclua todos os atributos simples (ou componentes simples de atributos compostos) de W como atributos de R, inclua também como atributos de R todos os atributos componentes das chaves primárias de cada uma das entidades fortes de W, cada uma dessas inclusões corresponde a uma chave estrangeira de R, a chave primária de R é a combinação dos atributos das chaves primárias das entidades fortes de W mais a chave parcial de W. Exemplo:

para o tipo de entidade fraca Dependente, gere a relação:

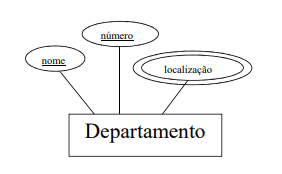


Dependente (cpfEmp, nome, dataNascimento, parentesco)

Dependente[cpfEmp] p→ Empregado[cpf]

**Atributo Multi-valorado:**

Para cada atributo multi-valorado A, crie uma nova relação R que inclua o atributo A mais a chave primária K (como chave estrangeira em R) da relação que representa o tipo de entidade ou o tipo de relacionamento que tem A como atributo. Se o atributo multi-valorado é composto, inclua seus componentes simples, a chave primária de R é a combinação de K e A. Exemplo:



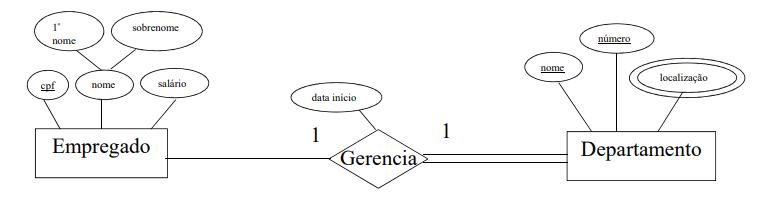
para o atributo Localização, gere a relação:

Localização\_Depto (número, localização)

Localização\_Depto[número] p→ Departamento[número]

**Relacionamento Binário 1:1**

Para cada tipo de relacionamento binário R 1:1 no esquema ERE, identifique as relações S e T que correspondem aos tipos de entidades participantes de R. Escolha uma das relações, S por exemplo, e inclua como chave estrangeira em S a chave primária de T. É melhor escolher um tipo de entidade com participação total em R no papel de S, inclua todos os atributos simples (ou componentes simples de atributos compostos) do tipo de relacionamento R como atributo de S. **Obs**: um mapeamento alternativo é juntar os dois tipos de entidades e o tipo de relacionamento em uma única relação. Isso é particularmente apropriado quando ambas as participações são totais e os tipos de entidades não participam de nenhum outro tipo de relacionamento. Exemplo: para o tipo de relacionamento Gerencia, adicione os atributos cpfGerente e dataInício à relação Departamento:

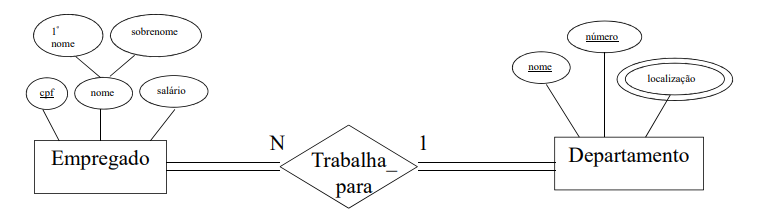


Departamento (número, nome, cpfGerente, dataInício)

Departamento[cpfGerente] b→ Empregado[cpf]

**Relacionamento Binário 1:N**

Para cada tipo de relacionamento binário R 1:N regular (não identificador) no esquema ERE, identifique a relação S que representa o tipo de entidade participante do lado N do tipo de relacionamento. Inclua como chave estrangeira em S a chave primária da relação T que representa o outro tipo de entidade participante de R. Isso ocorre porque cada instância do lado N está relacionada a, no máximo, uma instância do lado 1 do tipo de relacionamento, inclua todos os atributos simples (ou componentes simples de atributos compostos) do tipo de relacionamento R como atributos de S. Exemplo: para o tipo de relacionamento Trabalha\_para, adicione o atributo númeroDepto à relação Empregado:

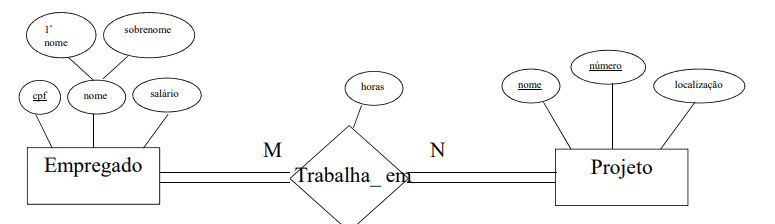


Empregado (cpf, primeiroNome, sobrenome, salario, númeroDepto)

Empregado[númeroDepto] b→ Departamento[número]

**Relacionamento Binário M:N**

Para cada tipo de relacionamento binário R M:N no esquema ERE, crie uma nova relação S para representar R. Inclua como chaves estrangeiras em S as chaves primárias das relações que representam os tipos de entidades participantes, também inclua todos os atributos simples (ou componentes simples de atributos compostos) do tipo de relacionamento R como atributos de S, a chave primária de S é a combinação de suas chaves estrangeiras. Exemplo: para o tipo de relacionamento Trabalha\_em, gere a relação:



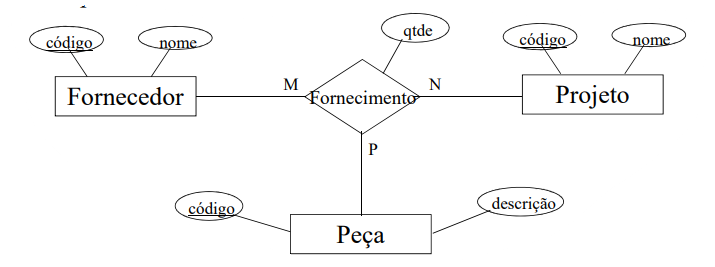
Trabalha\_em (cpf, númeroProjeto, horas)

Trabalha\_em[cpf] p→ Empregado[cpf]

Trabalha\_em[númeroProjeto] p→ Projeto[número]

Os tipos de relacionamentos 1:1 e 1:N podem ser mapeados de forma similar ao tipo de relacionamento M:N, essa alternativa é particularmente útil quando existem poucas instâncias relacionadas, a fim de evitar valores nulos nas chaves estrangeiras, neste caso, a chave primária da relação que representa o tipo de relacionamento é a chave estrangeira de somente uma das relações que representam os tipos de entidades participantes, para um tipo de relacionamento 1:N, a chave primária vem da relação representando o lado N. Para um tipo de relacionamento 1:1, qualquer lado pode ser escolhido, mas é preferível escolher o lado com restrição de participação total (se houver); no caso, a outra chave estrangeira deve ser definida como chave secundária.

**Relacionamento n-ário (n > 2)**

Para cada tipo de relacionamento n-ário R, onde n > 2, no esquema ERE, crie uma nova relação S para representar R. Inclua como atributos da chave estrangeira em S as chaves primárias das relações que representam os tipos de entidades participantes. Também inclua todos os atributos simples (ou componentes simples de atributos compostos) do tipo de relacionamento R como atributos de S. A chave primária de S é normalmente a combinação de suas chaves estrangeiras. Entretanto, se as restrições de cardinalidade de qualquer um dos tipos de entidades E participante de R é 1, então a chave primária de S não deve incluir a chave estrangeira que referencia a relação correspondente a E. Exemplo: para o tipo de relacionamento Fornecimento, gere a relação:

fornecimento (codForn, codProj, codPeça, qtde)

fornecimento[codForn] p→ Fornecedor[código]

fornecimento[codProj] p→ Projeto[código]

fornecimento[codPeça] p→ Peça[código]

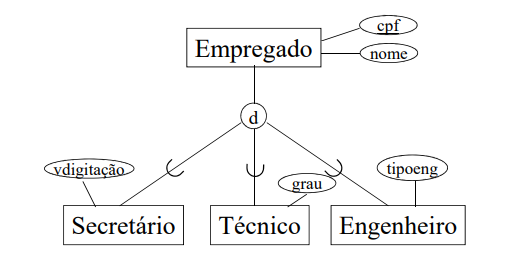
**Especialização / Generalização**

Opção 1: crie uma relação L para a superclasse C no esquema ERE com os atributos de C. A chave primária de L é uma chave de C. Crie também uma relação Li para cada subclasse Si. Cada Li inclui os atributos específicos de Si mais a chave primária de L, a qual torna-se também a chave primária de Li. Essa opção funciona para qualquer restrição i na especialização: disjunta/sobreposta, total/parcial.

Opção 2: crie uma relação Li para cada subclasse Si com os atributos da subclasse mais os atributos da superclasse. A chave primária de Li é uma chave da superclasse. Essa opção deve ser usada para restrições total e disjunta.

Opção 3: crie uma única relação L com todos os atributos da superclasse C e das subclasses Si, mais um atributo para indicar a qual subclasse cada tupla pertence. A chave primária de L é uma chave de C. Essa opção é para uma especialização cujas subclasses são disjuntas. Essa opção pode gerar um grande número de valores nulos.

Opção 4: crie uma única relação L com todos os atributos da superclasse C e das subclasses Si, mais um atributo lógico (flag) ti para cada subclasse para indicar se a tupla pertence à subclasse Si. A chave primária de L é uma chave de C. Essa opção é indicada para especialização cujas subclasses são sobrepostas (mas também funciona para especialização disjunta). Exemplos: para a especialização abaixo.



usando a opção 1, gere as relações:

Empregado (cpf, nome)

Secretário (cpf, vdigitação)

Secretário[cpf] p→ Empregado[cpf]

Técnico (cpf, grau)

Técnico[cpf] p→ Empregado[cpf]

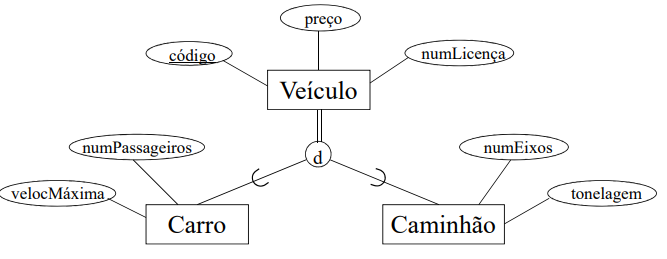
Engenheiro (cpf, tipoeng)

Engenheiro[cpf] p→ Empregado[cpf]

usando a opção 3, gere a relação:

Empregado (cpf, nome, tipodetrabalho, vdigitação, grau, tipoeng)

para a generalização abaixo:

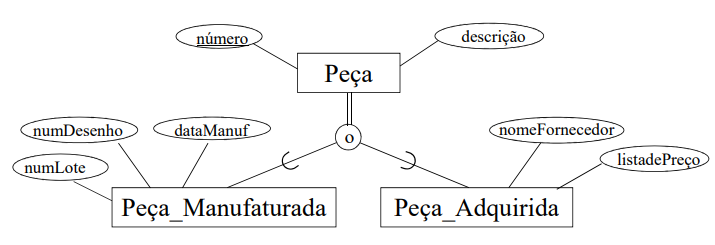


usando a opção 2, gere as relações:

Carro (código, numLicença, preço, numPassageiros, velocMáxima)

Caminhão (código, numLicença, preço, numEixos, tonelagem)

para a especialização abaixo:



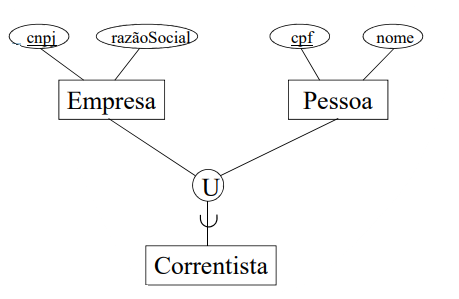
usando a opção 4, gere a relação:

Peça (número, descrição, Mflag, numDesenho, dataManuf, numLote, Aflag, nomeFornecedor, listadePreço)

**Tipo União ou Categoria**

Crie uma relação para representar a categoria e inclua todos os seus atributos. Para uma categoria cujas superclasses têm chaves diferentes, adicione um novo atributo chave, chamado “chave substituta”, para ser a chave primária da relação. Adicione este atributo como chave estrangeira em todas as relações correspondentes às superclasses da categoria, para especificar a correspondência de valores entre a chave substituta e as chaves de cada superclasse. Para uma categoria cujas superclasses têm as mesmas chaves, a chave substituta não é necessária. Adicione à relação representante da categoria o atributo chave de uma superclasse para ser a sua chave primária. Cada chave primária das relações correspondentes às superclasses são também chaves estrangeiras referenciando a relação da categoria. Exemplo:

para a categoria abaixo:



gere as relações:

Empresa (cnpj, razãosocial, numCorrentista)

Empresa[numCorrentista] n→ Correntista[numCorrentista]

Pessoa (cpf, nome, numCorrentista)

Pessoa[numCorrentista] n→ Correntista[numCorrentista]

Correntista (numCorrentista)