



Linguagem SQL

Aula 02 – Modelagem Relacional

Gustavo Bianchi Maia
gustavo.maia@faculdadeimpacta.com.br

Sumário

- Modelos de Banco de Dados
- Modelo Relacional
- Modelo Entidade Relacionamento
- Diagrama Entidade Relacionamento
- Modelagem de Dados
- Normalização
- Exercícios

Modelos de Banco de Dados

Todas as organizações, por menor que sejam, possuem quantidades cada vez maiores de dados e informações a armazenar. Todavia, a manipulação destas informações se tornou impossível de ser realizada manualmente (via papéis, principalmente), pois sua utilização além de demorada (devido a catalogação dos dados) é passível de erros principalmente ocasionados pelo desgaste do operador em conseguir resgatar informações requisitadas.

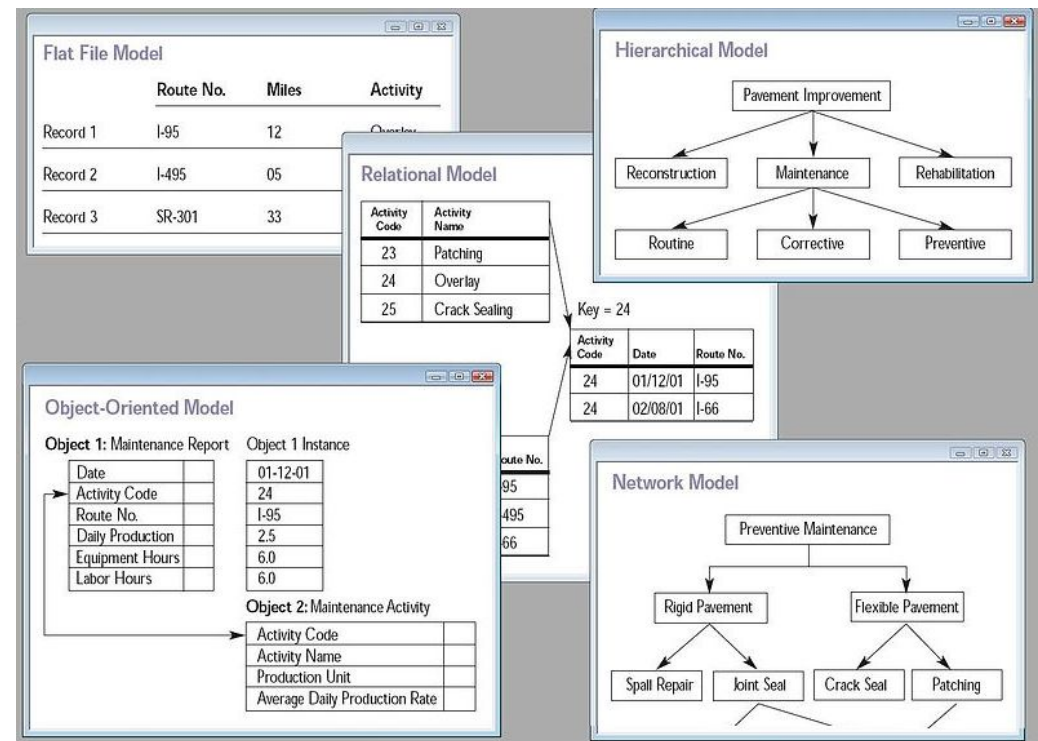
Nesse sentido, torna-se mais fácil encontrar a informação numa base de dados que recorre a uma das tecnologias de informação de maior sucesso e confiança. Ou seja, as bases de dados estendem a função do papel ao guardar a informação em computadores.

Modelos de Banco de Dados

Bancos de dados ou bases de dados são coleções organizadas de dados que se relacionam de forma a criar algum sentido (informação) e dar mais eficiência durante uma pesquisa ou estudo. São de vital importância para empresas, e há décadas se tornaram umas das principais peças dos sistemas de informação.

Existem vários modelos de base de dados:

- Modelo Plano (Flat File)
- Modelo em Rede
- Modelo Hierárquico
- Modelo Relacional
- Orientado a objetos
- Orientado a documentos
- Objeto-Relacional



Modelos de Banco de Dados

O SGBD disponibiliza uma interface para que seus clientes possam incluir, alterar ou consultar dados previamente armazenados.

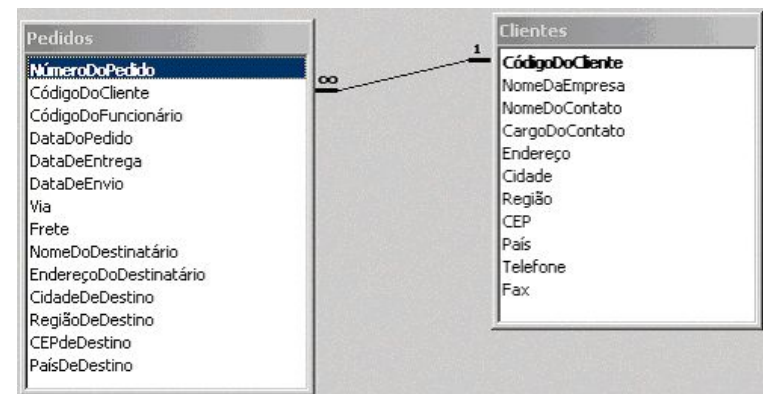
Em bancos de dados relacionais a interface é constituída pelas APIs (Application Programming Interface) ou drivers do SGBD, que executam comandos na linguagem SQL (Structured Query Language).

Um SGBD implica na criação e manutenção de bases de dados, elimina a necessidade de especificação de definição de dados, age como interface entre os programas de aplicação e os arquivos de dados físicos e separa as visões lógicas das concepções dos dados.

Modelo Relacional

O modelo relacional, proposto por E. F. Codd, é um modelo de dados, adequado a ser o modelo subjacente de um SGBD, que se baseia no princípio em que todos os dados estão guardados em tabelas.

Numa definição simplificada, o modelo baseia-se em **dois conceitos**: conceito de **entidade** e **relacionamento** - Uma entidade é um elemento caracterizado pelos dados que são recolhidos na sua identificação, vulgarmente designado por tabela. Na construção da tabela identificam-se os dados da entidade. A atribuição de valores a uma entidade constrói um registro da tabela. O relacionamento determina o modo como cada registro de cada tabela se associa a registros de outras tabelas.



Modelo Relacional

Em 1985, Edgar Frank Codd, criador do modelo relacional, publicou um artigo onde definia 13 regras para que um Sistema Gerenciador de Banco de Dados (SGBD) fosse considerado relacional:

- 1. Regra Fundamental**

Um SGBD relacional deve gerir os seus dados usando apenas suas capacidades relacionais

- 2. Regra da Informação**

Toda informação deve ser representada de uma única forma, como dados em uma tabela

Modelo Relacional

3. Regra da Garantia de Acesso

Esta regra é essencialmente uma reafirmação do requisito fundamental para chaves primárias. Todos os dados necessitam ser acessíveis e pode ser acessado logicamente (e unicamente - valor atômico) usando o nome da tabela, o valor da chave primária da linha e o nome da coluna.

4. Tratamento sistemático de valores nulos

Deve permitir que cada campo possa permanecer nulo, especificamente, deve suportar uma representação de "falta de informação e informações inaplicáveis" que é sistemática, diferente de todos os valores regulares (diferente do zero, string vazia, string de caracteres em branco). Existem para representar dados não existentes de forma sistemática e independente do tipo de dado.

Modelo Relacional

5. Catálogo Dinâmico On-line Baseado no Modelo Relacional

- Os metadados devem ser armazenados e gerenciados como dados comuns, ou seja, em tabelas no interior do Banco de Dados. Esses dados devem estar disponíveis aos usuários autorizados, utilizando a linguagem de consulta padrão de manipulação de dados.

6. Sub-linguagem Ampla de Dados

- O sistema necessita suportar ao menos uma linguagem relacional que
 - Possua uma sintaxe linear;
 - Possa ser utilizada seja interativamente, seja por meio de programas;
 - Suporte operações de definição de dados (incluindo definições de Visualizações);
 - Suporte operações de manipulação de dados (atualização, bem como recuperação), de segurança e restrições de integridade, transação (*begin, commit e rollback*), operações de gerenciamento.

Modelo Relacional

7. Atualização por meio de Visões

Todas as Visualizações que são teoricamente atualizáveis deve ser atualizáveis pelo sistema.

8. Inserção, atualização e eliminação de alto nível

A capacidade de manipular a relação base ou relações derivadas como um operador único não se aplica apenas a recuperação de dados, mas também a inserção, alteração e eliminação de dados.

9. Independência Física de Dados

Aplicativos e recursos ad hoc não são afetados logicamente quando os métodos de acesso ou as estruturas de armazenamento físico são alterados.

10. Independência Lógica de Dados

Aplicativos e recursos ad hoc não são afetados logicamente quando alterações de estruturas de tabela preservarem os mesmos valores originais da tabela (alteração da ordem ou inserção de colunas).

Modelo Relacional

11. Independência de Integridade

As relações de integridade específicas de um banco de dados relacional devem ser definidas em uma sublinguagem de dados e armazenadas no catálogo (e não em programas).

12. Independência de Distribuição

A linguagem de manipulação de dados deve possibilitar que as aplicações permaneçam inalteradas estejam os dados centralizados ou distribuídos fisicamente.

13. A Não-Transposição das Regras

Se o SGBD dá suporte a acesso de baixo nível aos dados, não deve haver um modo de negligenciar as regras de integridade do mesmo.

Modelo Entidade Relacionamento

Conjunto de conceitos e elementos de modelagem que o projetista de banco de dados precisa conhecer.

Definição

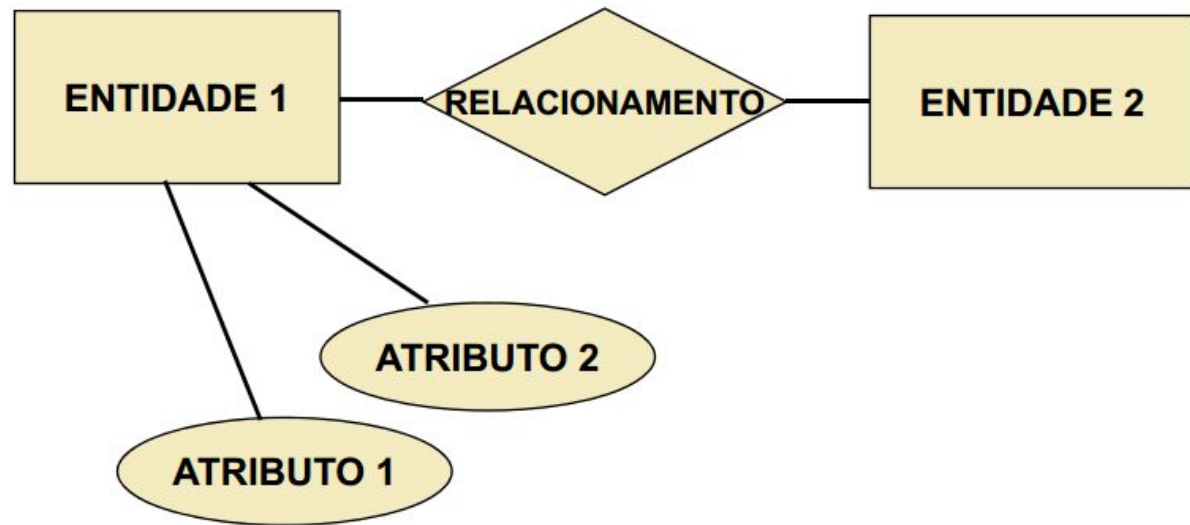
Consiste em mapear o mundo real do sistema em um modelo gráfico que irá representar o modelo e o relacionamento existente entre os dados. O Modelo de Entidade e Relacionamento (MER ou Modelo ER) é uma representação da realidade e pode ser representado por [entidades](#), [relacionamentos](#) e [atributos](#).

Objetivo

Facilitar o projeto de banco de dados, possibilitando a especificação da estrutura lógica geral do banco de dados

Modelo Entidade Relacionamento

- Componentes do Diagrama E-R (Peter Chen)
 - **Retângulos:** representam conjuntos-entidade
 - **Elipses:** representam atributos
 - **Losangos:** representam conjuntos-relacionamento
 - **Linhas:** ligam atributos a conjuntos-entidade e conjuntos e conjuntos-entidade a conjuntos-relacionamento



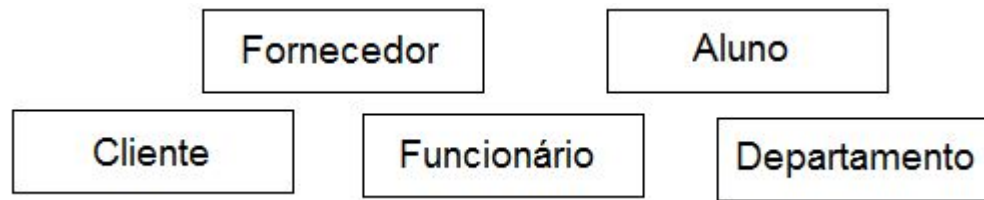
Modelo Entidade Relacionamento

A arquitetura de um banco de dados relacional pode ser descrita de maneira informal ou formal. Na descrição informal estamos preocupados com aspectos práticos da utilização e usamos os termos tabela, linha e coluna. Na descrição formal estamos preocupados com a semântica formal do modelo e usamos termos como relação (tabela), tupla (linha) e atributo(coluna).

- **Entidades** (tabelas - relações)

Conjunto de elementos que possuem características próprias e sobre os quais é preciso armazenar informações úteis. Uma entidade é um “objeto” no mundo real que pode ser identificada de forma única em relação aos outros objetos;

Ex:



Modelo Entidade Relacionamento

Podemos classificar as entidades segundo o motivo de sua existência:

- **Entidades fortes:** são aquelas cuja existência independe de outras entidades, ou seja, por si só elas já possuem total sentido de existir. Em um sistema de vendas, a entidade produto, por exemplo, independe de quaisquer outras para existir.

Candidato

Veículo

- **Entidades fracas:** ao contrário das entidades fortes, as fracas são aquelas que dependem de outras entidades para existirem, pois individualmente elas não fazem sentido (*quando a entidade precisa da chave primária de outra entidade para tornar-se única*). Mantendo o mesmo exemplo, a entidade venda depende da entidade produto, pois uma venda sem itens não tem sentido.

Transação

Venda

Modelo Entidade Relacionamento

- **Entidades associativas:** esse tipo de entidade surge quando há um relacionamento do tipo muitos para muitos (explicado a seguir). Nestes casos, é necessária a criação de uma entidade intermediária cuja identificação é formada pelas chaves primárias das outras duas entidades.

No contexto de uma aplicação de vendas, como precisamos relacionar vendas e produtos numa relação muitos para muitos, a entidade produto não pode referenciar diretamente a venda, nem o inverso, pois isso caracterizaria um relacionamento um para um ou um para muitos. Sendo assim, criamos uma entidade intermediária para representar os itens da venda, que tanto possuem a identificação do produto, quando da venda em que estão contidos. Neste caso específico, também caberiam a esta entidade informações como quantidade de itens e desconto unitário, por exemplo.

Modelo Entidade Relacionamento

O que não são entidades:

- Entidade com apenas 1 elemento
- Operações do sistema
- Saídas do sistema
- Pessoas que realizam trabalhos (usuários do sistema)
- Cargos de Direção

Modelo Entidade Relacionamento

Todos os dados de um banco de dados relacional são armazenados em entidades (tabelas). Uma tabela é uma simples estrutura de linhas e colunas. Em uma tabela, cada linha contém um mesmo conjunto de colunas.

*Ex: **Objetos concretos** (pessoa, automóvel, produto, ...) ou **abstrato** (departamento, projeto, segmento, ...)*

- **Instância** (registros / linhas - tuplas)

São elementos da entidade, ou seja, cada linha formada por uma lista ordenada de colunas representa uma instância, registro ou tupla. Os registros não precisam conter informações em todas as colunas, podendo assumir valores nulos quando assim se fizer necessário.

Modelo Entidade Relacionamento

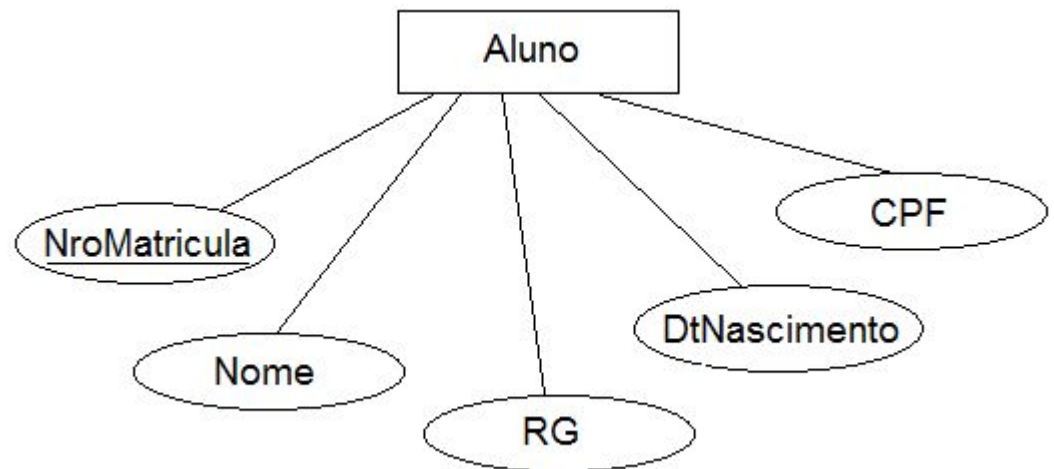
- **Atributos** (ou colunas/campos)

São informações que desejamos guardar sobre a instância da entidade. É uma qualidade associada a uma entidade. As colunas de uma tabela são também chamadas de Atributos. Ao conjunto de valores que um atributo pode assumir chama-se domínio. Por exemplo: em um campo do tipo numérico, serão somente armazenados números.

- **Atributo Obrigatório** - É aquele que para uma instância de uma entidade ou relacionamento deve possuir um valor (*NOT NULL*).
- **Atributo Opcional** - É aquele que para uma instância da entidade ou relacionamento pode possuir um valor (*NULL*).

Modelo Entidade Relacionamento

- **Atributo Monovalorado** – Um único valor para cada instância.
Ex: Nome
- **Atributo Multivalorado** – Mais de um valor para cada instância.
Ex: Dependentes
- **Atributo Derivado** – Quando mantém relação com outro atributo.
Ex: Idade (atributo derivado) da Data de Nascimento



Modelo Entidade Relacionamento

Os atributos podem ser classificados quanto à sua função da seguinte forma:

- **Descritivos:** representam característica intrínsecas de uma entidade, tais como nome ou cor.
- **Nominativos:** além de serem também descritivos, estes têm a função de definir e identificar um objeto. Nome, código, número são alguns exemplos.
- **Referenciais:** representam a ligação de uma entidade com outra em um relacionamento. Por exemplo, uma venda possui o CPF do cliente, que a relaciona com a entidade cliente.

Modelo Entidade Relacionamento

- **Tipos do Atributo (tipo de dado)**

É a natureza do dado associada ao domínio de um atributo.

Ex: Caractere ou string, Numérico, Data, ...

- **Domínio do Atributo**

É o universo de valores que um atributo pode armazenar, ou seja, o conjunto de valores possíveis do atributo. A simples escolha do tipo de dados já impõe um domínio no atributo, mas podemos restringir ainda mais, aceitando apenas alguns valores possíveis.

Ex:

Atributo	Domínio
Idade	de 0 a 120
Gênero	Masculino ou Feminino
Estado Civil	Solteiro, Casado, Divorciado, Viúvo

Modelo Entidade Relacionamento

Chave

Uma chave é um atributo ou conjunto de atributos que têm a propriedade de identificar unicamente uma ocorrência da entidade. As tabelas relacionam-se umas as outras através de chaves.

Tipos de chaves:

- **Chave Simples:** Formada por apenas um atributo.
Ex: Numa entidade de Funcionário, número de matrícula do funcionário.
- **Chave Composta:** Quando mais de um atributo compõem a chave.
Ex: Numa entidade de Item de Pedido, número do pedido e número do produto.

Modelo Entidade Relacionamento

- **Chave primária:** (*PK - Primary Key*) é a chave que identifica cada registro dando-lhe unicidade. Só pode haver uma chave primária por entidade. Pode ser simples ou composta, mas não pode conter atributos opcionais (NULL). A chave primária nunca se repetirá.

*Ex: Numa entidade de **Produto**, número do produto.*

- **Chave Candidata ou Alternativa** - Quando um atributo ou conjunto de atributos tem os mesmos requisitos de uma chave primária, ou seja, pode ser utilizada para identificar unicamente os registros, mas não foi escolhido como chave primária da entidade.

*Ex: Numa entidade de **Cliente**, número do cliente, CPF, RG, ...*

Modelo Entidade Relacionamento

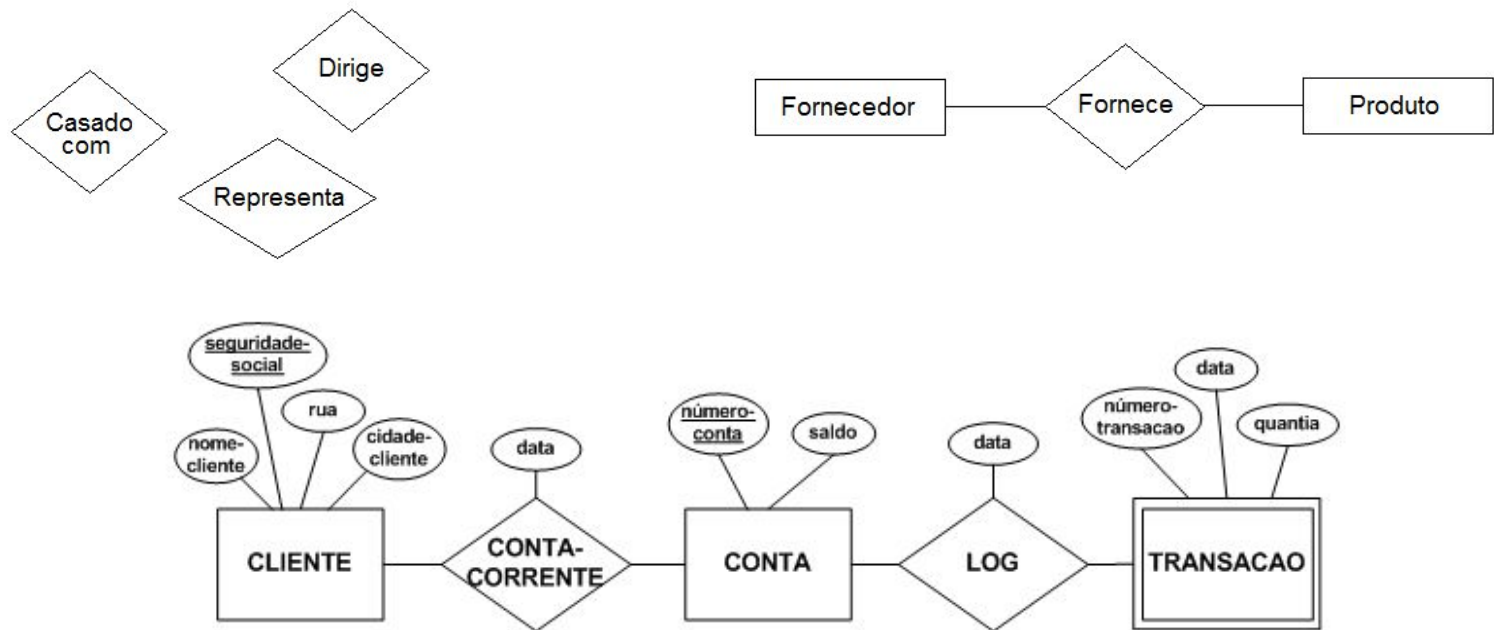
- **Chave Estrangeira:** (*FK - Foreign Key*) é a chave formada através de um relacionamento com a chave primária ou candidata de outra tabela. Define um relacionamento entre as tabelas e pode ocorrer repetidas vezes. Caso a chave primária seja composta na origem, a chave estrangeira também o será.

Modelo Entidade Relacionamento

Relacionamentos

As entidades são conectadas umas às outras através de relacionamentos. É uma associação entre instâncias de entidades devido a regras de negócio. Pode ocorrer entre instâncias de uma mesma entidade (auto-relacionamento).

Ex:



Modelo Entidade Relacionamento

Existem alguns tipos de relacionamentos possíveis no MER:

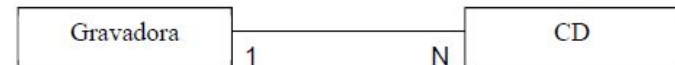
- Um para um 1..1 (um para um)

Cada uma das duas entidades envolvidas referenciam obrigatoriamente apenas uma unidade da outra.



- Um para muitos 1..n ou 1..* (um para muitos)

Uma das entidades envolvidas pode referenciar várias unidades da outra, porém, do outro lado cada uma das várias unidades referenciadas só pode estar ligada a uma unidade da outra entidade.

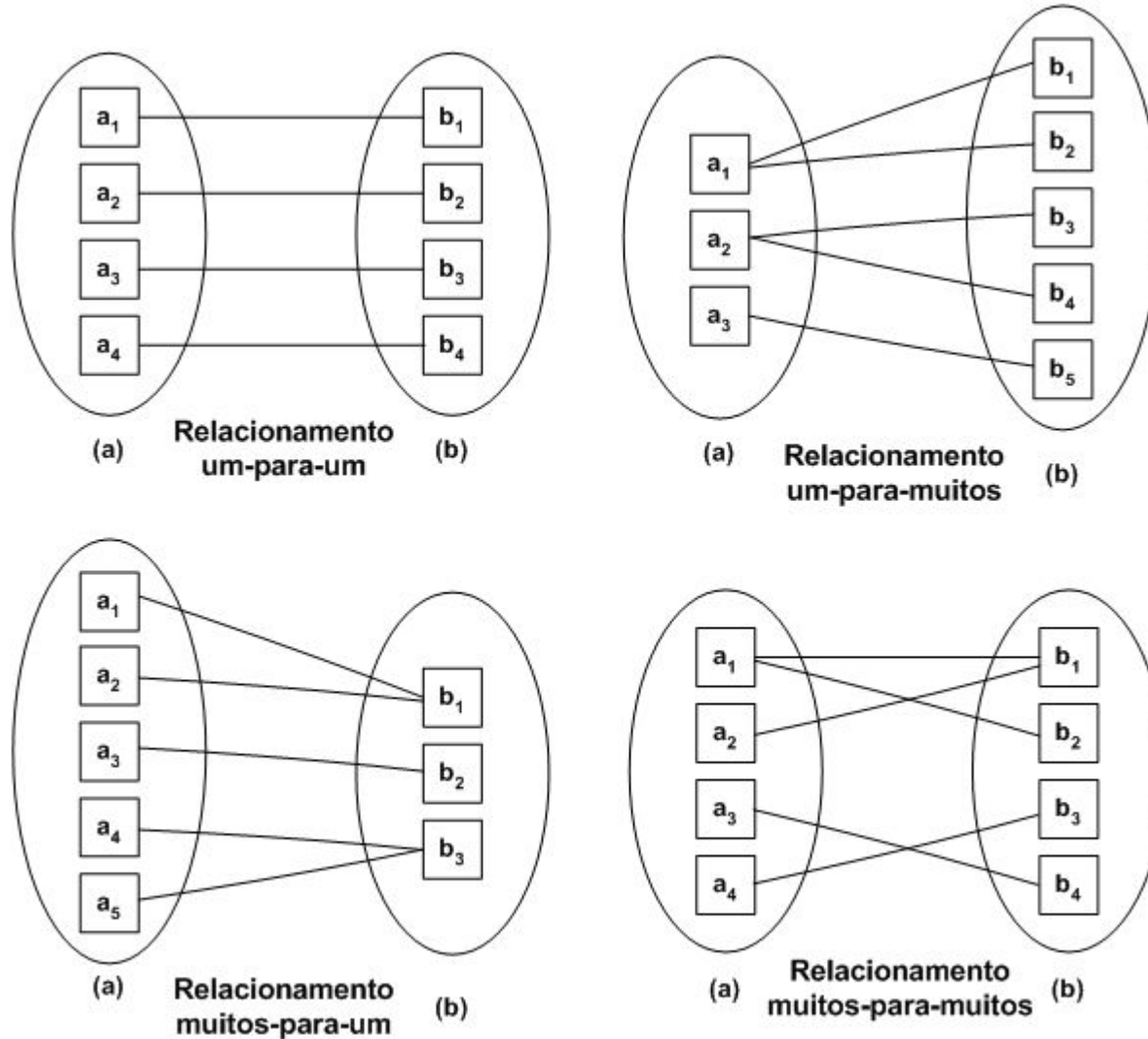


- Muitos para muitos n..n ou *.* (muitos para muitos)

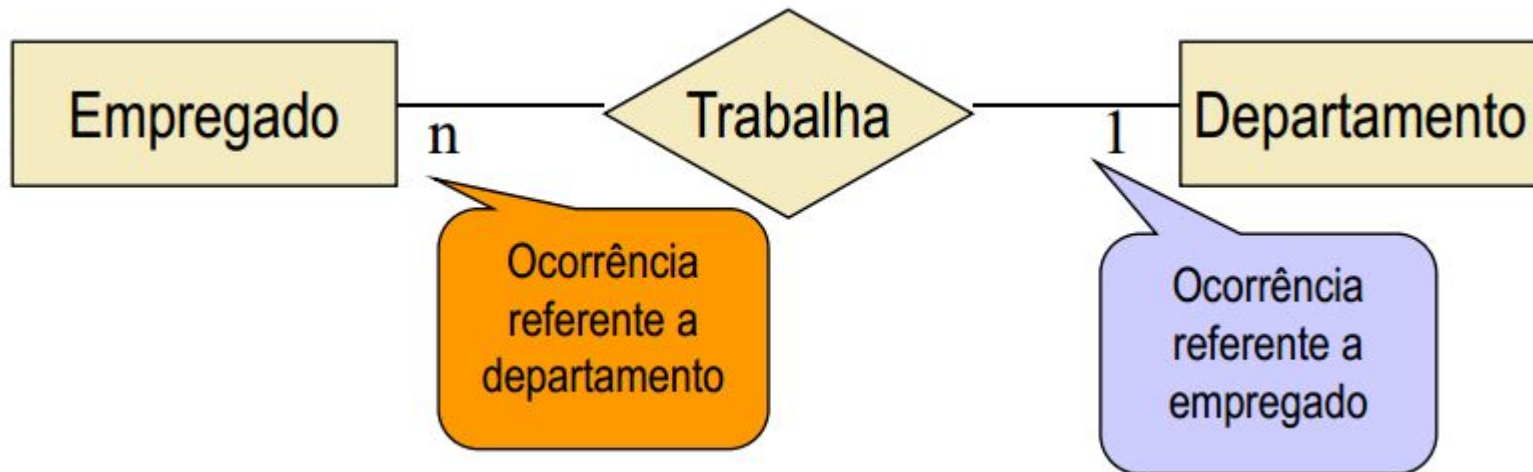
Neste tipo de relacionamento cada entidade, de ambos os lados, podem referenciar múltiplas unidades da outra.



Modelo Entidade Relacionamento



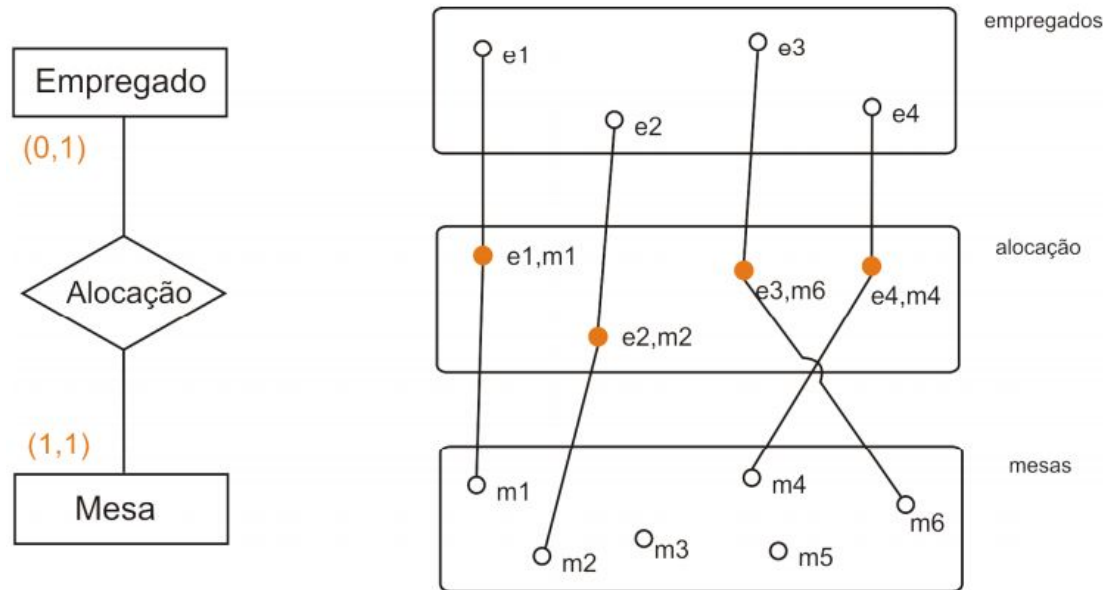
Modelo Entidade Relacionamento



Modelo Entidade Relacionamento



Com base na figura acima, construa um diagrama de ocorrências para o relacionamento apresentado para o relacionamento apresentado.



Modelo Entidade Relacionamento

Notações

Baseado no trabalho original de E. F. Codd, em março de 1976, que Peter Pin-Shan Chen publicou um trabalho intitulado “The Entity-Relationship Model”, no qual definia uma possível abordagem para o processo de modelagem dos dados. Na modelagem conceitual o modelo E-R proposto por Peter Chen, continua muito utilizado.

Na década de 80, James Martin propôs uma nova notação para engenharia da informação (Crow’s Foot – “Pé de Galinha”), baseada em símbolos e é comum a utilização desta notação na modelagem lógica e física.

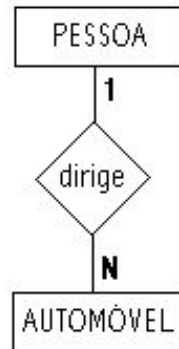
Modelo Entidade Relacionamento

Relacionamento

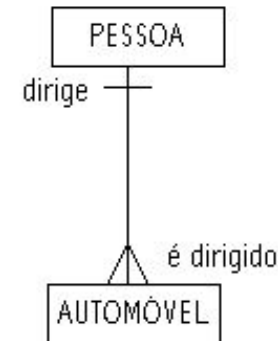
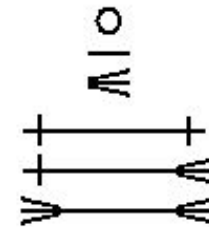
"Zero"
"Um"
"Muitos"
"Um para Um"
"Um para Muitos"
"Muitos para Muitos"

Notação Chen

0
1
N
1:1
1:N
M:N



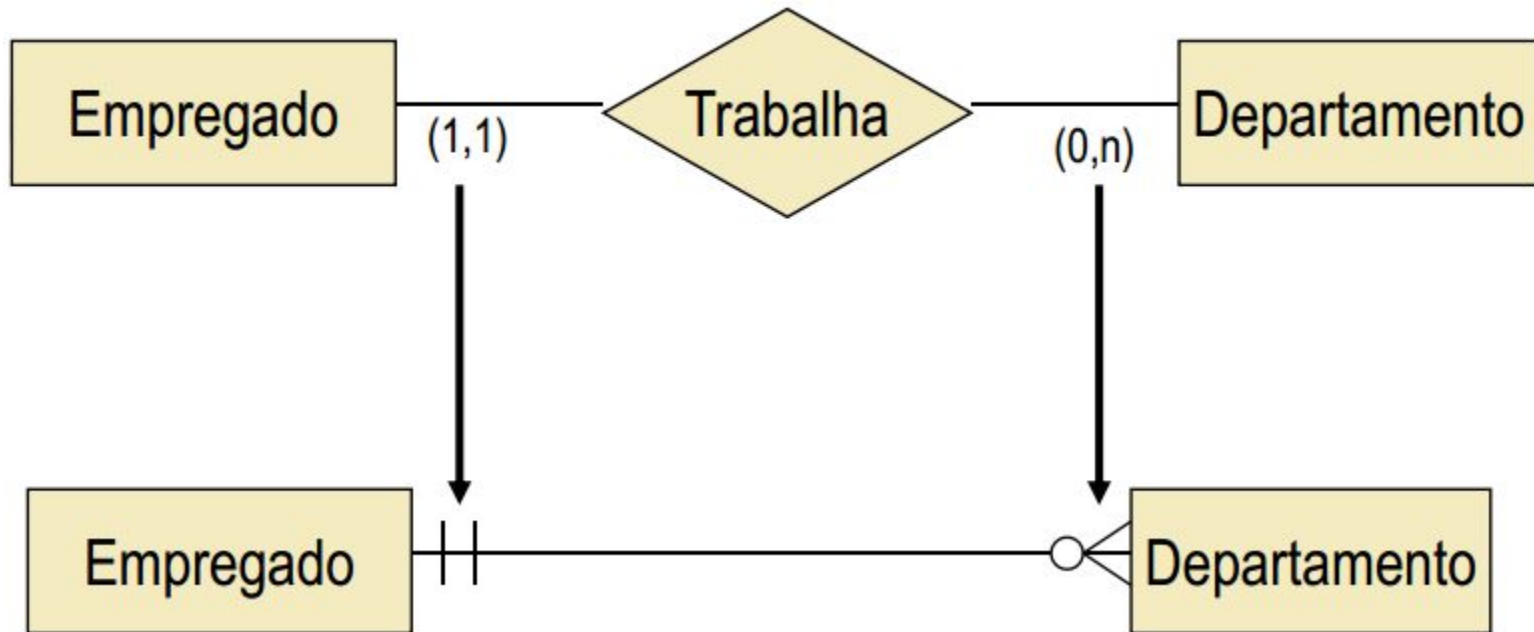
Notação Martin



Existem várias notações que foram sendo evoluídas ao longo do tempo, como por exemplo a IDEF1X que é muito utilizado em ferramentas CASE.

Modelo Entidade Relacionamento

Peter Chen



James Martin

Modelo Entidade Relacionamento

Cardinalidade

Define quantas ocorrências de uma entidade participam no mínimo e no máximo em um relacionamento com outra entidade.

- **Cardinalidade Máxima:** Indica a quantidade máxima de ocorrências de entidades que podem estar associadas a uma ocorrência da outra entidade (1 ou N).
- **Cardinalidade Mínima:** Especifica se a participação de todas as ocorrências das entidades no relacionamento é obrigatória ou opcional.

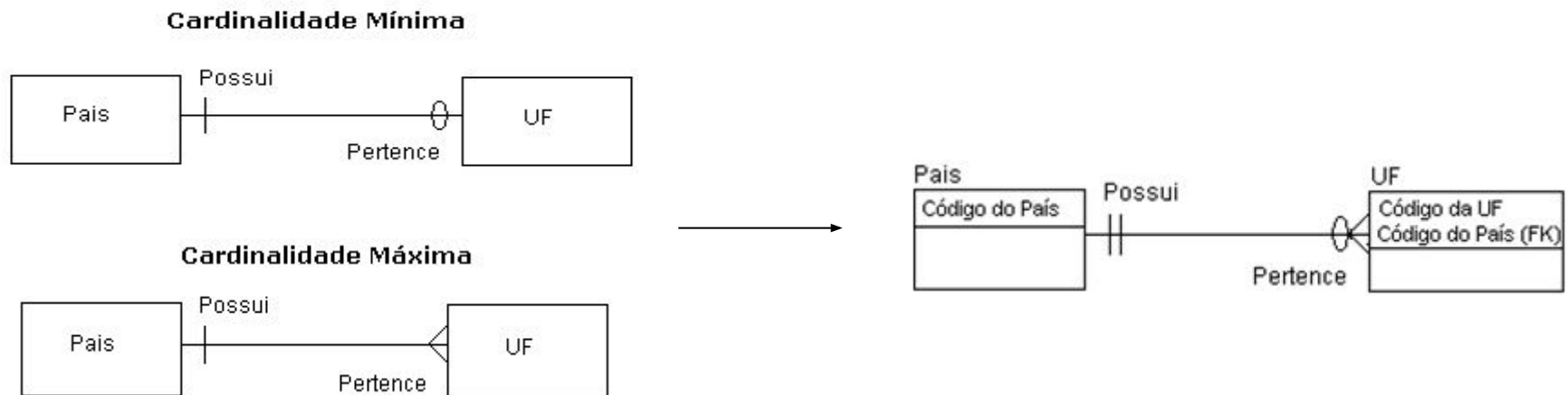


Diagrama Entidade Relacionamento

O Diagrama Entidade Relacionamento (Diagrama ER ou ainda DER) é a sua representação gráfica e principal ferramenta do Modelo Entidade Relacionamento.

É um modelo diagramático que descreve o modelo de dados de um sistema com alto nível de abstração e é usado para representar o modelo conceitual do negócio.

Em situações práticas, o diagrama é tido muitas vezes como sinônimo de modelo, uma vez que sem uma forma de visualizar as informações, o modelo pode ficar abstrato demais para auxiliar no desenvolvimento do sistema. Dessa forma, quando se está modelando um domínio, o mais comum é já criar sua representação gráfica, seguindo algumas regras.

Diagrama Entidade Relacionamento

O diagrama facilita ainda a comunicação entre os integrantes da equipe, pois oferece uma linguagem comum utilizada tanto pelo analista, responsável por levantar os requisitos, e os desenvolvedores, responsáveis por implementar aquilo que foi modelado.

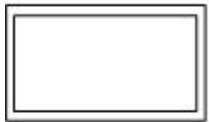
Em sua notação original, proposta por Peter Chen (idealizador do modelo e do diagrama), as entidades deveriam ser representadas por retângulos, seus atributos por elipses e os relacionamentos por losangos, ligados às entidades por linhas, contendo também sua cardinalidade (1..1, 1..N ou N..N).

Diagrama Entidade Relacionamento

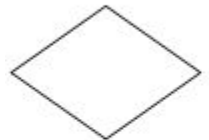
Representações Gráficas



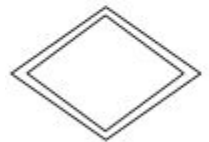
ENTIDADES (FORTE)



ENTIDADES FRACA



RELACIONAMENTOS



RELACIONAMENTOS DEPENDENTES



ATRIBUTOS



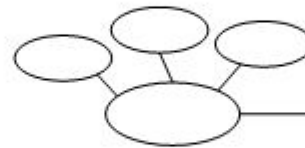
ATRIBUTO CHAVE



ATRIBUTO CHAVE PARCIAL (ENTIDADES FRACAS)



ATRIBUTO MULTIVALORADO



ATRIBUTO COMPOSTO



ATRIBUTO DERIVADO



TOTAL PARTICIPAÇÃO DE E2 EM R



CARDINALIDADE 1:N PARA E1:E2 EM R



RESTRIÇÃO DE CARDINALIDADE (min,máx) NA PARTICIPAÇÃO DE E EM R

Diagrama Entidade Relacionamento

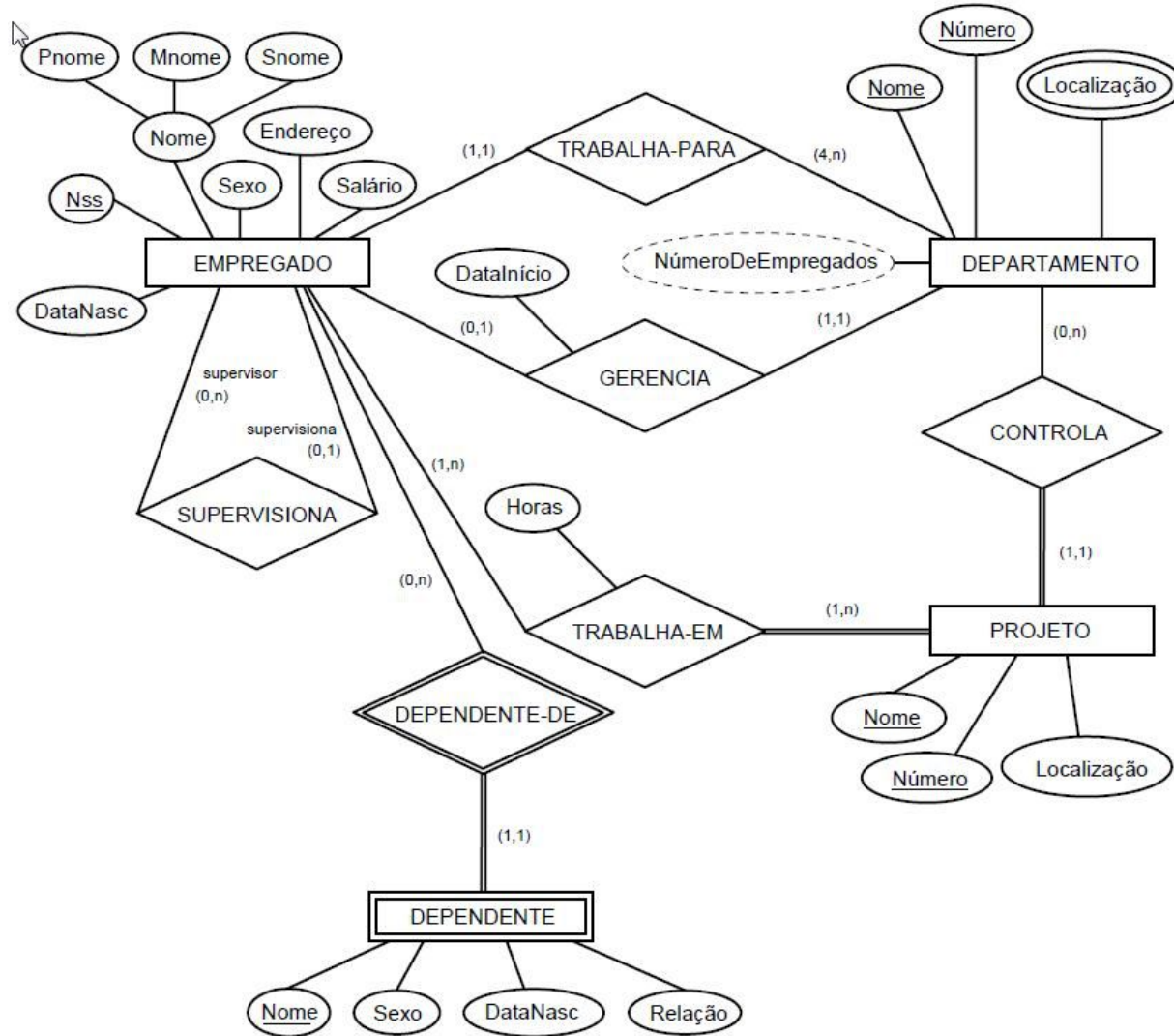


Diagrama Entidade Relacionamento

A notação original ainda é adotada, porém, notações mais modernas abandonaram o uso de elipses para atributos e passaram a utilizar o formato mais utilizado na UML, em que os atributos já aparecem listados na própria entidade. Essa forma torna o diagrama mais limpo e fácil de ser lido.

Neste diagrama podemos identificar alguns dos conceitos vistos:

- **Entidades fortes:** Usuário, Livro e Sessão;
- **Entidades fracas:** Empréstimo;
- **Relacionamentos:** um Usuário efetua vários Empréstimos, vários Empréstimos contêm vários Livros, vários Livros pertencem a uma Sessão.

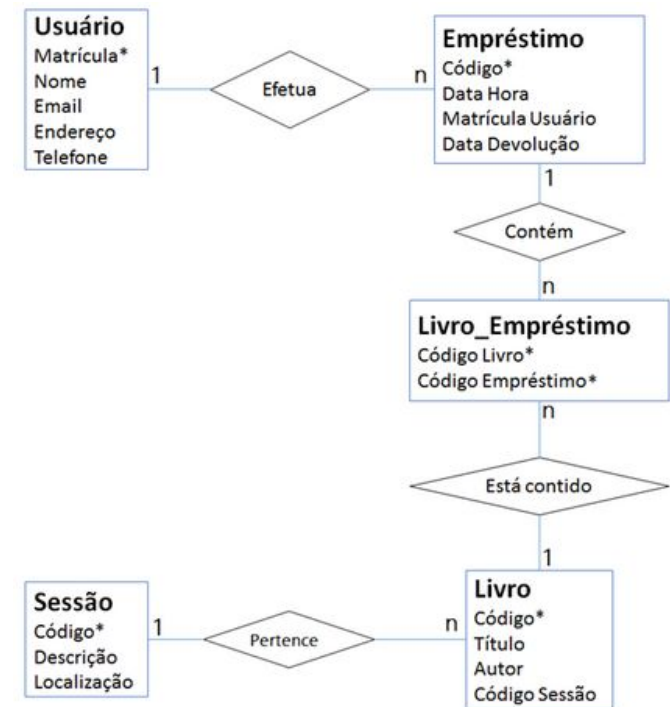


Diagrama Entidade Relacionamento

No domínio representado pelo diagrama abaixo temos as seguintes entidades e relacionamentos:

- Proprietário contata Corretor (um proprietário pode contatar vários corretores e um corretor pode ser contactado por vários proprietários).
- Corretor atende Inquilino (um corretor pode atender vários inquilinos e um inquilino pode ser atendido por vários corretores).
- Inquilino aluga Imóvel (um inquilino aluga um imóvel e um imóvel pode ser alugado por vários inquilinos).
- Proprietário possui Imóvel (um proprietário possui vários imóveis e um imóvel pertence a apenas um proprietário).

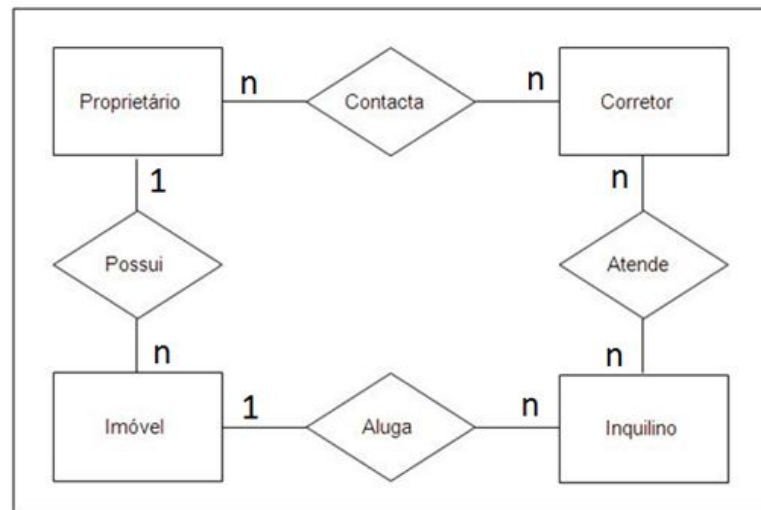


Diagrama Entidade Relacionamento de sistema de imobiliária

Diagrama Entidade Relacionamento

Outra variação mostra a cardinalidade de uma forma mais completa, deixando de forma mais explícita, o número de objetos envolvidos nos relacionamentos. Nesse modelo, em cada lado do relacionamento temos o formato (X,Y).

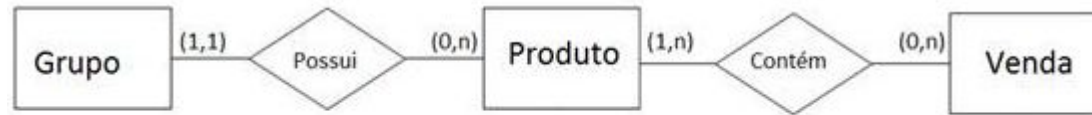


Diagrama Entidade Relacionamento

No diagrama, lemos os relacionamentos da seguinte forma:

- 1 ou 1 grupo possui 0 ou muitos produtos. Como de um lado temos “1 ou 1”, isso equivale a apenas “1”, pois não temos várias possibilidades. Já do lado do produto, indicamos que um grupo pode possuir nenhum produto, mas também pode possuir vários.
- 0 ou várias vendas contém 1 ou muitos produtos. Ou seja, um produto pode nunca ser vendido (0 vendas) como também pode ser vendido várias vezes (n vendas). Já uma venda deve conter 1 ou vários produtos, pois uma venda não pode estar vazia (0 produtos).

•

Modelagem de Dados

Um modelo é uma representação abstrata e simplificada de um sistema real, com a qual se pode explicar ou testar o seu comportamento, em seu todo ou em partes.

Modelar significa criar um modelo que explique as características de funcionamento e comportamento de um software a partir do qual ele será criado, facilitando seu entendimento e seu projeto, através das características principais que evitarão erros de programação, projeto e funcionamento. É uma parte importante do desenho de um sistema de informação.

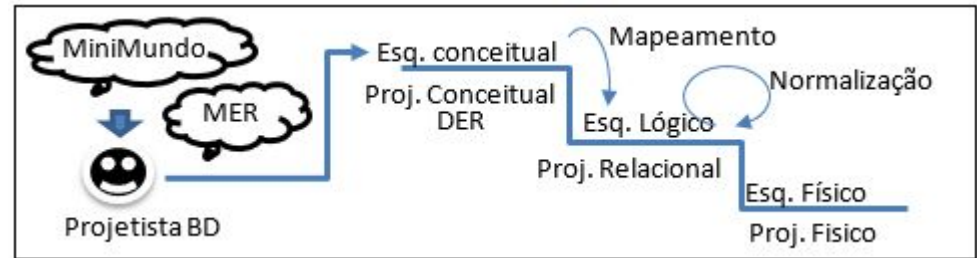
Modelagem de Dados

Um modelo de dados:

- Representar um ambiente observado
- Servir de instrumento para comunicação
- Favorecer o processo de verificação e validação
- Apoiar na discussão sobre alterações, correções e novos requisitos
- Servir como referencial para a geração de estruturas de dados
- Preservar a informação (documentação)
- Capturar aspectos de relacionamento entre os objetos observados

Modelagem de Dados

Níveis da Modelagem de Dados



- **Modelo Conceitual**

Representa as regras de negócio sem limitações tecnológicas ou de implementação por isto é a etapa mais adequada para o envolvimento do usuário que não precisa ter conhecimentos técnicos.

(Sem IFs, CASEs ou tabelas, apenas visão geral do negócio com entidades e atributos principais)

- **Modelo Lógico**

Leva em conta limites impostos por algum tipo de tecnologia de banco de dados (relacional, hierárquico, etc.).

(Ainda não sabemos qual o banco será, mas já documentamos entidades, atributos, nomenclaturas, chaves, relações e normalização)

- **Modelo Físico**

Leva em consideração limites impostos pelo SGBD (SQL Server, Oracle, Sybase, ..) e pelos requisitos não funcionais dos programas que acessam os dados.

(Aqui definimos quais entidades e relacionamentos se transformarão em tabelas, campos que terão índices, tamanho e tipo, etc.)

Mapeamento do modelo Conceitual para o Lógico

Passo 1:

Para cada entidade regular E no DER, criar uma relação R que inclua todos os atributos simples de E. Para um atributo composto, inclua apenas os atributos simples que compõem o atributo composto. Escolha um dos atributos-chave de E como sendo a chave-primária de R. Se a chave escolhida de E for composta, então o conjunto de atributos simples que o compõem irão formar a chave-primária de R.



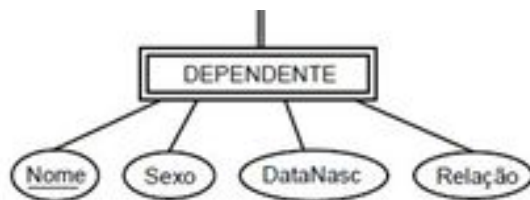
Empregado
<u>NSS</u> (PK) Pnome Mnome Snome Sexo Endereco Salário DataNasc

Mapeamento do modelo Conceitual para o Lógico

Passo 2:

Para cada tipo de entidade fraca W do DER com o tipo de entidade de identificação E, criar uma relação R e incluir todos os atributos simples (ou os componentes simples de atributos compostos) de W como atributos de R. Além disso, incluir como a chave estrangeira de R a chave-primária da relação que corresponde ao tipo de entidade de identificação; isto resolve o problema do tipo do relacionamento de identificação de W.

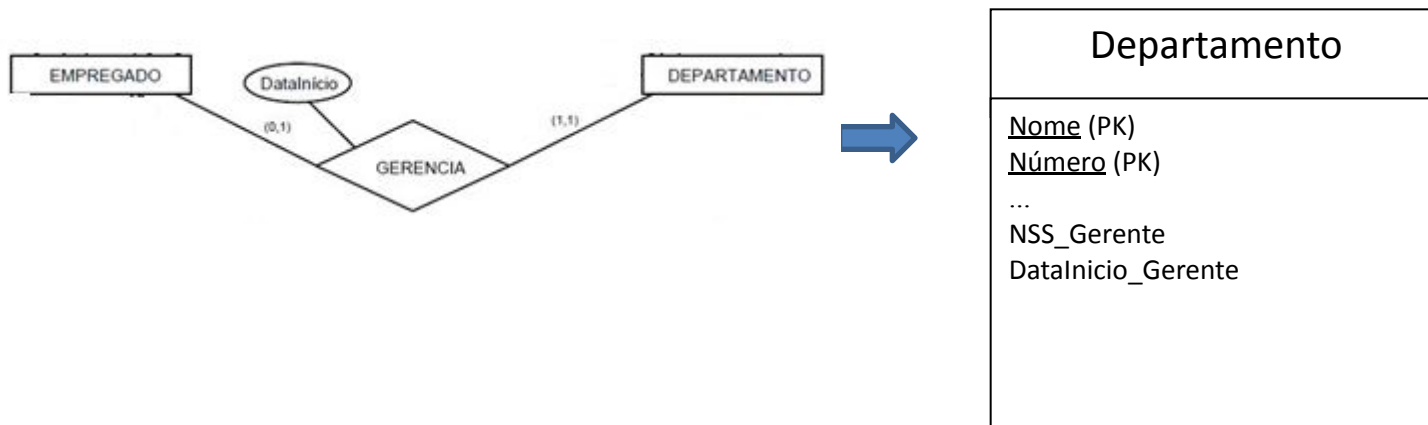
A chave-primária de R é a combinação da chave-primária do tipo de entidade de identificação e a chave-parcial do tipo de entidade fraca W.



Mapeamento do modelo Conceitual para o Lógico

Passo 3:

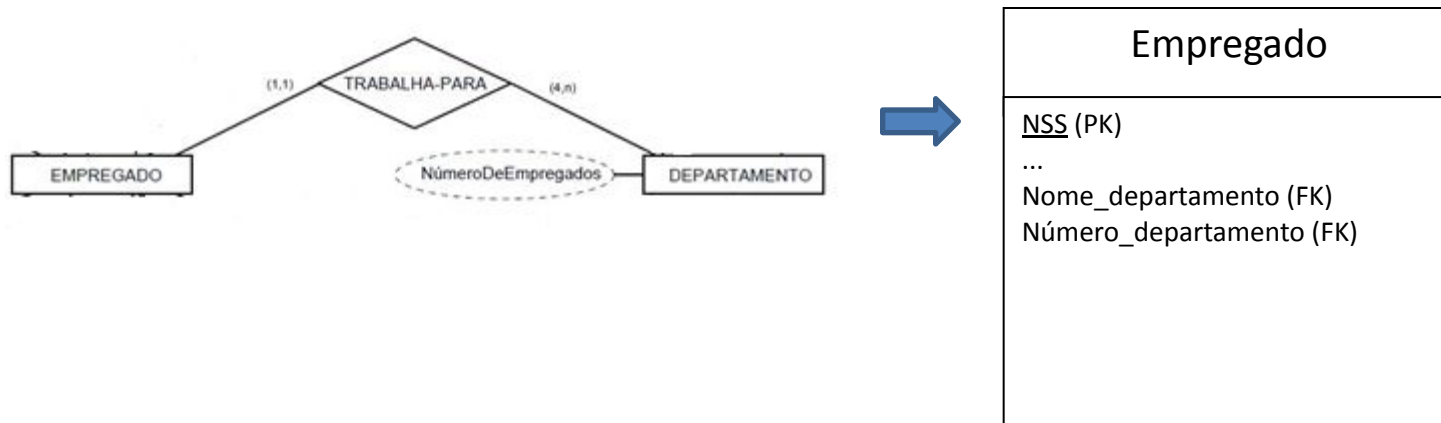
Para cada tipo de relacionamento binário 1:1 R do DER, criar as relações S e T que correspondem aos tipos de entidade participantes em R. Escolher uma das relações, por exemplo S, que inclua como chave-estrangeira de S a chave-primária de T. É melhor escolher o tipo de entidade com participação total em R como a relação S. Inclua todos os atributos simples (ou os componentes simples de atributos compostos) do tipo de relacionamento 1:1 R como atributos de S.



Mapeamento do modelo Conceitual para o Lógico

Passo 4:

Para cada tipo de relacionamento binário regular 1:N (não fraca) R, identificar a relação S que representa o tipo de entidade que participa do lado N do tipo de relacionamento. Inclua como chave-estrangeira de S a chave-primária da relação T que representa o outro tipo de entidade que participa em R; isto porque cada instância da entidade do lado N está relacionada a mais de uma instância de entidade no lado 1 do tipo de relacionamento. Inclua também quaisquer atributos simples (ou componentes simples de atributos compostos) do tipo de relacionamento 1:N como atributos de S.

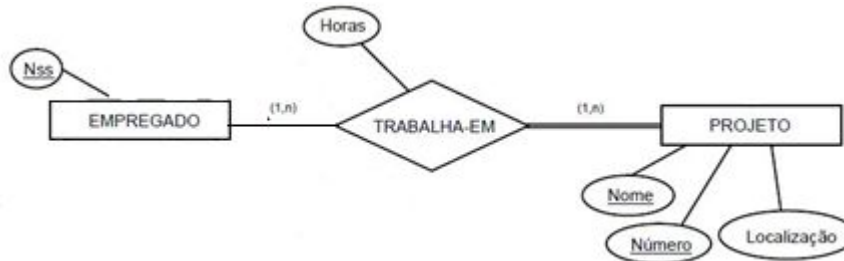


Mapeamento do modelo Conceitual para o Lógico

Passo 5:

Para cada tipo de relacionamento binário M:N R, criar uma nova relação S para representar R. Incluir como chave-estrangeira em S as chaves-primárias das relações que representam os tipos de entidade participantes; sua combinação irá formar a chave primária de S. Inclua também qualquer atributo simples do tipo de relacionamento M:N (ou componentes simples dos atributos compostos) como atributos de S. Note-se que não se pode representar um tipo de relacionamento M:N como uma simples chave-estrangeira em uma das relações participantes - como foi feito para os tipos de relacionamentos 1:1 e 1:N - por causa da razão de cardinalidade M:N.

Relacionamentos M:N sempre derivam uma nova relação, para o tipo relacionamento.



Mapeamento do modelo Conceitual para o Lógico

Passo 6:

Para cada atributo A multivalorado, criar uma nova relação R que inclua um atributo correspondendo a A e a chave-primária K da relação que representa o tipo de entidade ou o tipo de relacionamento que tem A como atributo. A chave-primária de R é a combinação de A e K. Se o atributo multivalorado é composto inclua os atributos simples que o compõem.



Localização
<u>Nome Departamento (PK) (FK)</u>
<u>Número Departamento (PK) (FK)</u>
...
Localização

Mapeamento do modelo Conceitual para o Lógico

Passo 7:

Para cada tipo de relacionamento n-ário R , $n > 2$, criar uma nova relação S para representar R . Inclua como chave-estrangeira em S as chaves-primárias das relações que representam os tipos de entidades participantes. Incluindo-se também qualquer atributo simples do tipo de relacionamento n-ário (ou componentes simples dos atributos compostos) como atributo de S . A chave-primária de S é normalmente uma combinação de todas as chaves-estrangeiras e referenciam as relações que representam os tipos de entidades participantes.

Porém, se a restrição de participação (min, max) de um dos tipos de entidades E que participa em R tiver $\text{max}=1$, então a chave-primária de S pode ser a chave-estrangeira que referencia a relação E' correspondente a E ; isto porque cada entidade e' em E irá participar em apenas uma instância de R e, portanto, pode identificar univocamente esta instância de relacionamento.



Aluguel
<u>Exported KEY 1</u> (PK) (FK)
<u>Exported KEY 2</u> (PK) (FK)
<u>Exported KEY 3</u> (PK) (FK)
...
<atributos do relacionamento>

Normalização na Prática

Introduzido pelo cientista britânico Dr. Codd, é um processo matemático, com fundamentos na teoria de conjuntos, que visa retirar grupos repetitivos (*atributos multivalorados*), redundância de dados e anomalias de atualização (insert, delete, update).

A aplicação desta técnica faz com que surjam novas entidades e relacionamentos, de forma que não se perca a integridade das informações iniciais.

O objetivo da normalização é elaborar um modelo de dados compacto e acoplado, facilitando posteriormente a manutenção do sistema e aumentando a velocidade de processamento das informações.

Normalização na Prática

Para abordar a metodologia da normalização, aliando a parte conceitual com aplicação prática, utilizaremos um sistema de pedidos tradicional, onde podemos observar diversas anomalias e onde a aplicação da normalização se faz indicado.

Código do Vendedor: 735

Prazo de Entrega: 20 dias

Nome Vendedor: Arthur Lima

Num. Pedido	Cliente	Endereço	CPF	RG	Cód. Produto	Unidade	Qty	Descrição	Valor Unitário	Total Produto	Total do Pedido
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	341	Kg	1	Pia Celite L12	R\$ 153,27	R\$ 153,27	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	127	Metro	20	Pedra	R\$ 5,00	R\$ 100,00	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	56	Litro	2	Querosene	R\$ 8,59	R\$ 17,18	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	42	Galão	5	Tinta	R\$ 150,00	R\$ 750,00	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	61	Unid.	1	Serra	R\$ 25,00	R\$ 25,00	R\$ 1.045,45
3413	Ana Beatriz Silva	Rua Luiz XVI, 123 - Itaim	5648565487	45757458	230	Saco	3	Estopa	R\$ 7,00	R\$ 21,00	R\$ 506,00
3413	Ana Beatriz Silva	Rua Luiz XVI, 123 - Itaim	5648565487	45757458	127	Metro	50	Pedra	R\$ 5,00	R\$ 250,00	R\$ 506,00
3413	Ana Beatriz Silva	Rua Luiz XVI, 123 - Itaim	5648565487	45757458	435	Unid.	1	Furadeira	R\$ 235,00	R\$ 235,00	R\$ 506,00
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	341	Kg	3	Pia Celite L12	R\$ 153,27	R\$ 459,81	R\$ 880,71
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	56	Litro	10	Querosene	R\$ 8,59	R\$ 85,90	R\$ 880,71
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	127	Metro	10	Pedra	R\$ 5,00	R\$ 50,00	R\$ 880,71
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	28	Kg	1	Prego	R\$ 10,00	R\$ 10,00	R\$ 880,71
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	65	Kg	2	Parafuso	R\$ 25,00	R\$ 50,00	R\$ 880,71
5210	Cássio Mendes	Av. Laerte Gusmão, 21 - Centro	9879879878	11556644	42	Galão	1	Tinta	R\$ 150,00	R\$ 150,00	R\$ 880,71
9437	Suelen Abrantes	Rua 7 de setembro, 32 - Bom Retiro	1261261261	28945865	61	Unid.	3	Serra	R\$ 25,00	R\$ 75,00	R\$ 395,99
9437	Suelen Abrantes	Rua 7 de setembro, 32 - Bom Retiro	1261261261	28945865	230	Saco	6	Estopa	R\$ 7,00	R\$ 42,00	R\$ 395,99
9437	Suelen Abrantes	Rua 7 de setembro, 32 - Bom Retiro	1261261261	28945865	56	Litro	8	Querosene	R\$ 8,59	R\$ 68,72	R\$ 395,99
9437	Suelen Abrantes	Rua 7 de setembro, 32 - Bom Retiro	1261261261	28945865	43	Galão	1	Massa	R\$ 57,00	R\$ 57,00	R\$ 395,99
9437	Suelen Abrantes	Rua 7 de setembro, 32 - Bom Retiro	1261261261	28945865	341	Kg	1	Pia Celite L12	R\$ 153,27	R\$ 153,27	R\$ 395,99

Normalização na Prática

- Primeira Forma Normal – 1FN

"Uma entidade está na 1FN, se e somente se, não possuir atributos multivalorados."

Às vezes encontramos informações que retratam ocorrências de um mesmo fato dentro de uma entidade (atributos multivalorados). A ocorrência de um fato deve ser atômica (única) e representada por uma única linha, vinculadas a sua chave primária.

A 1 FN diz que: cada ocorrência da chave primária deve corresponder a uma e somente uma informação de cada atributo, ou seja, a entidade não deve conter grupos multivalorados.

Para se obter entidades na 1FN, é necessário gerar chaves primárias para cada entidade e decompor cada entidade não normalizada em tantas entidades quanto for o número de conjuntos de atributos com vários valores. Nas novas entidades criadas, a chave primária é a concatenação da chave primária da entidade original mais o(s) atributo(s) do grupo repetitivo visualizado(s) como chave primária deste grupo.

Normalização na Prática

Verificando atributos multivalorados que geram linhas de uma mesma ocorrência na entidade.

Código do Vendedor: 735

Prazo de Entrega: 20 dias

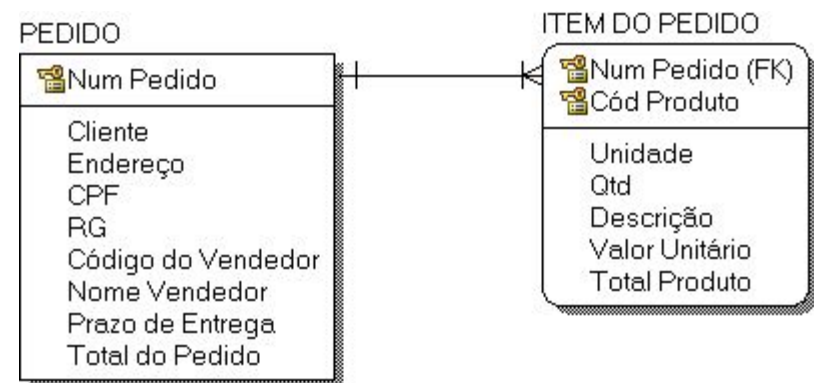
Nome Vendedor: Arthur Lima

Num. Pedido	Cliente	Endereço	CPF	RG	Cód. Produto	Unidade	Qty	Descrição	Valor Unitário	Total Produto	Total do Pedido
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	341	Kg	1	Pia Celite L12	R\$ 153,27	R\$ 153,27	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	127	Metro	20	Pedra	R\$ 5,00	R\$ 100,00	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	56	Litro	2	Querosene	R\$ 8,59	R\$ 17,18	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	42	Galão	5	Tinta	R\$ 150,00	R\$ 750,00	R\$ 1.045,45
1625	Flávio dos Santos	Rua 13 de Maio, 23 - Centro	7878787878	12345678	61	Unid.	1	Serra	R\$ 25,00	R\$ 25,00	R\$ 1.045,45

Aplicando a 1FN, obtemos:



Modelo Conceitual

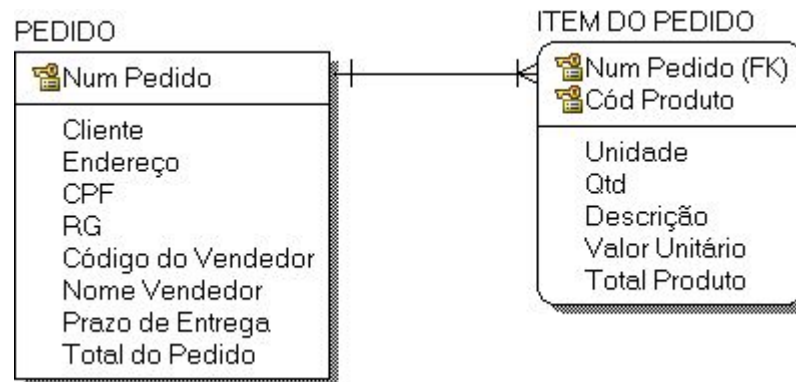


Modelo Lógico

Normalização na Prática

- Dependência Funcional

"Um atributo ou conjunto de atributos A é dependente funcional de um outro atributo B, na mesma entidade, se a cada valor de B existir nas linhas da entidade um único valor de A correspondente."

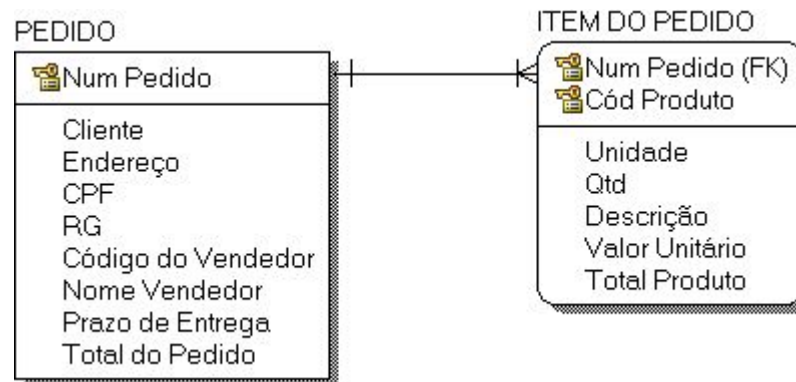


Dependência: *Prazo de Entrega* (A) depende funcionalmente do *Num Pedido* (B)

Normalização na Prática

- Dependência Funcional Total (ou Completa) e Parcial

*"Na ocorrência de uma **chave primária concatenada ou composta**, dizemos que um atributo ou conjunto de atributos depende de forma completa ou total da chave primária, se cada valor da chave (e não parte dela) está associado um valor para cada atributo. Se o atributo depender apenas de parte da chave, temos uma dependência parcial."*



Dependência Total: *QTD* depende de forma total de *Num Pedido* e *Cod Produto*

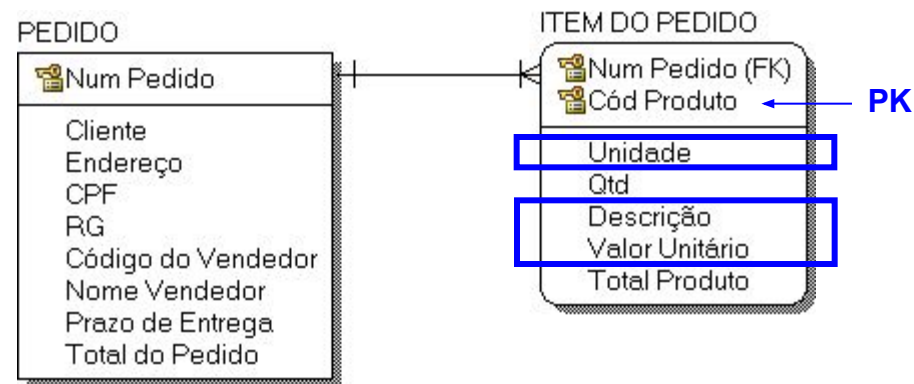
Dependência Parcial: *Descrição* depende de forma parcial de *Cod Produto*

Normalização na Prática

- Segunda Forma Normal – 2FN

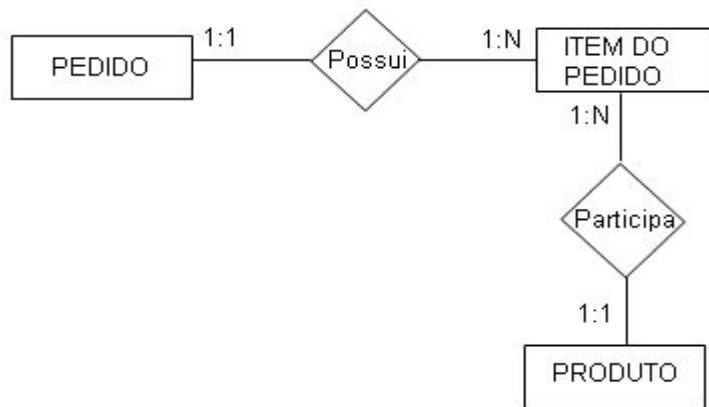
“Uma entidade está na 2FN, se e somente se, estiver na 1FN e cada atributo não-chave for dependente total da chave primária concatenada, isto é, cada atributo não-chave não poderá ser dependente parcial da chave primária concatenada.”

Geram novas entidades que herdarão a chave parcial e todos atributos que dependem desta chave parcial.

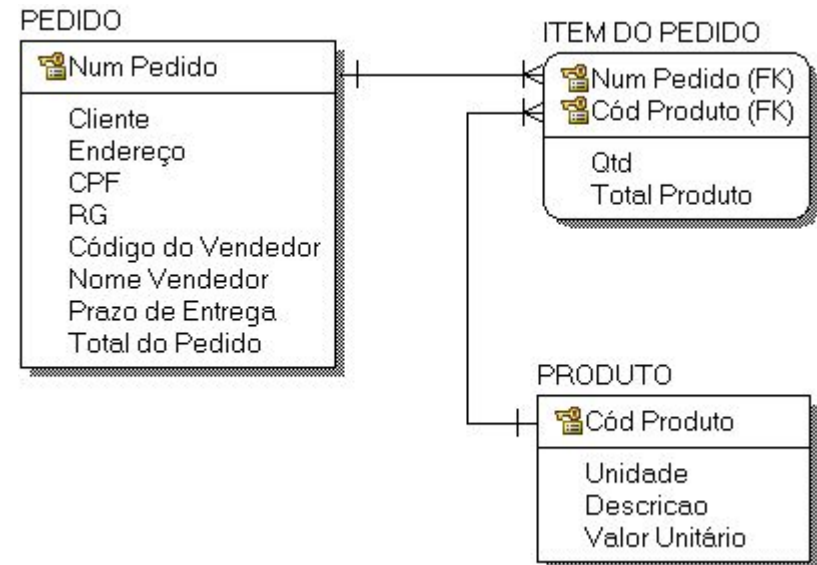


Normalização na Prática

Aplicando a 2FN, obtemos:



Modelo Conceitual

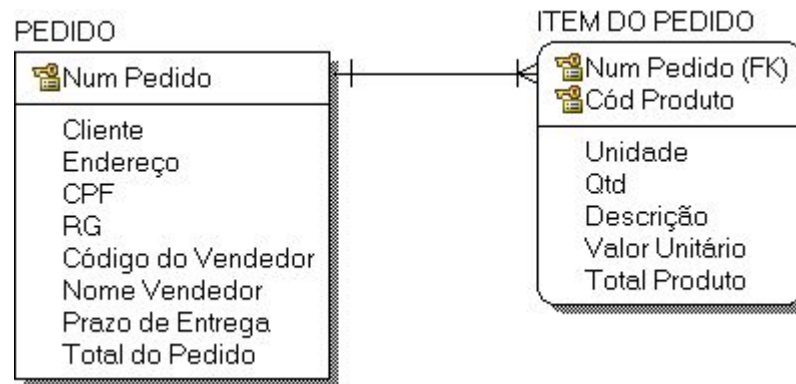


Modelo Lógico

Normalização na Prática

- Dependência Funcional Transitiva

*“Quando um atributo ou conjunto de atributos A depende funcionalmente de um atributo B, na mesma entidade, e B **não pertence a chave primária**, dizemos que A é dependente transitivo de B.”*



Dependência Transitiva: *Endereço, CPF, RG depende transitivamente de Cliente*

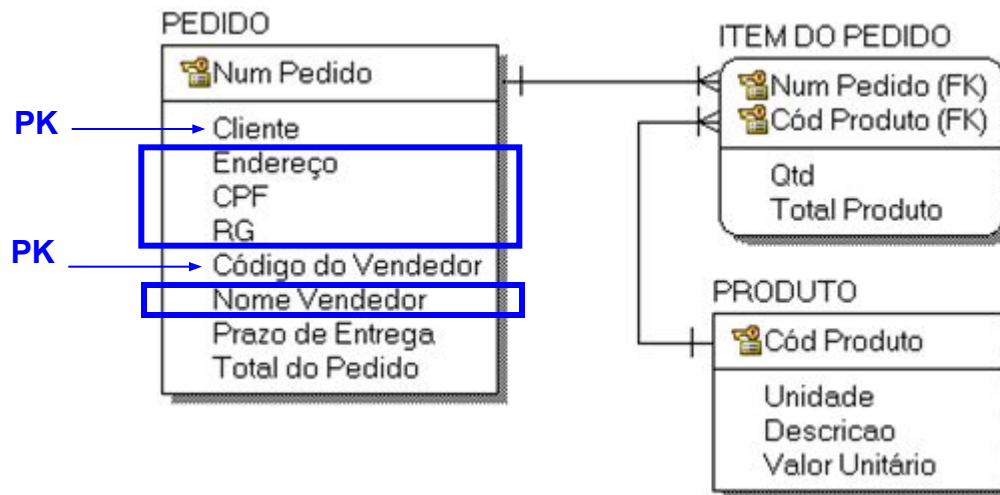
Normalização na Prática

Terceira Forma Normal – 3FN

“Uma entidade está na 3FN, se ela estiver na 2FN e cada atributo não chave não possuir dependência transitiva com outros atributos não chave da entidade.”

Atributos que representam **cálculos**, podem ser interpretados como dependência funcional e podem também ser retirados do modelo.

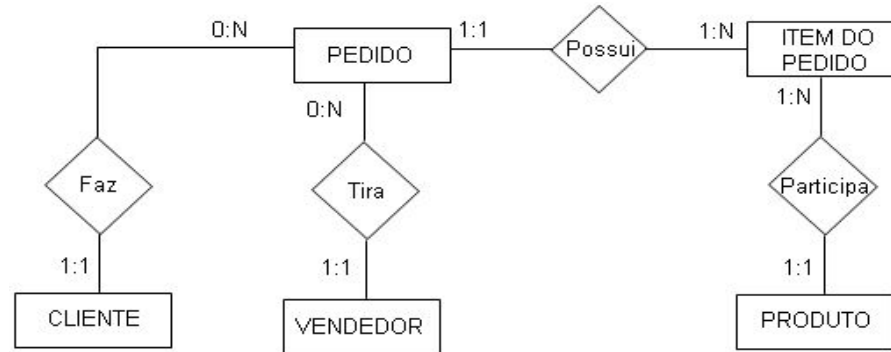
Ao retirar a dependência transitiva, devemos criar uma nova entidade que contenha os atributos que dependem transitivamente de outro e a sua chave primária é o atributo que causou esta dependência.



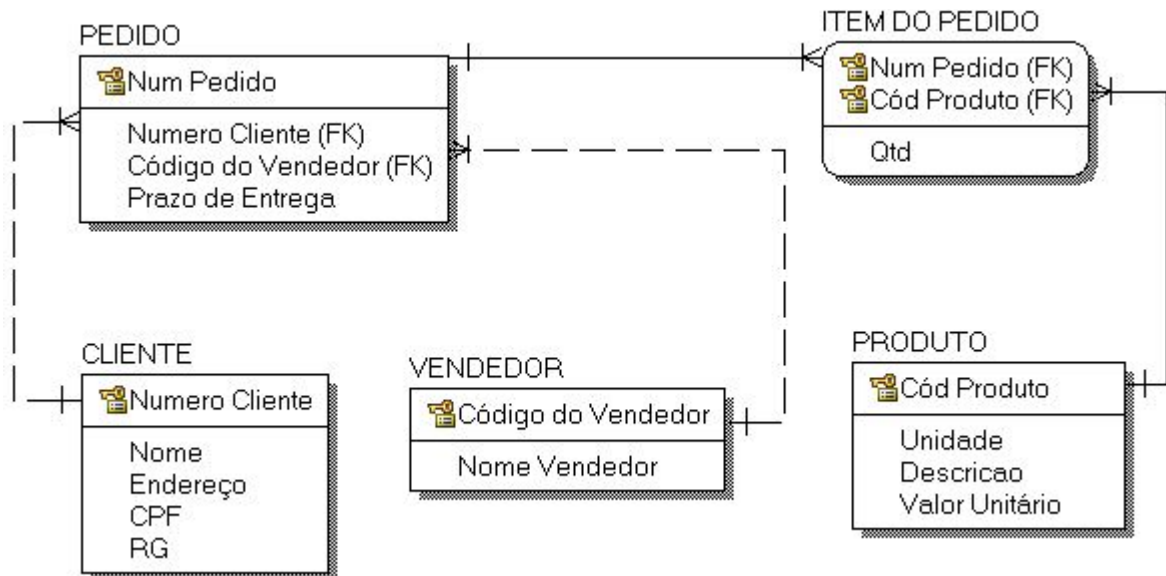
Normalização na Prática

Aplicando a 3FN:

Modelo Conceitual



Modelo Lógico



Normalização na Prática

- Existem outras formas normais como Forma Normal de Boyce-Codd (BCNF), 4FN, 5FN e a Forma Normal Domain-Key, mas a aplicação destas FN, deixam uma complexidade muito grande ao modelo. Na prática, um modelo na 3FN já elimina a maior parte das anomalias de redundância e atualização.



Obrigado !

Gustavo Bianchi Maia
gustavo.maia@faculdadeimpacta.com.br