



BANCO DE DADOS

- **Normalização**
- **Modelo lógico**

- As anomalias estão presentes em Bancos de Dados mal projetados
- Normalmente ocorrem excesso de dados armazenados nas tabelas.
- Causas principais são dependências parciais e dependências transitivas.

- **Anomalias de Inserção:** Ocorre quando se pretende inserir uma informação sem que uma outra da qual ela depende esteja inserida

Exemplo:

Inserir uma parcela para uma venda sem que a venda esteja cadastrada.

- **Anomalias de Exclusão:** Ocorre quando se tenta excluir uma informação que possui outras informações que são dependentes desta.

Exemplo:

Excluir uma venda sendo que esta possui parcelas cadastradas.

- **Anomalias de Modificação:** Ao se alterar dados em uma tabela, dados de outra tabela também devem sofrer modificações.

Exemplo:

Alterar o código da venda da tabela venda necessita da alteração do código da referida venda também na tabela parcela.

- Temos que projetar o banco de dados de forma a não ter presentes as anomalias de inserção, exclusão ou de modificação.
- Conseguimos eliminá-las através do processo de **normalização**

- Processo de análise de uma entidade de forma que esteja bem construída.
- Decompõe as entidades com anomalias em entidades menores e mais bem estruturadas de forma a inserir, excluir ou modificar dados sem gerar anomalias.

A normalização possui caráter organizativo e pode ocorrer durante a concepção do modelo conceitual, durante a derivação do modelo lógico para o relacional, ou após a derivação do modelo lógico.

As principais características de uma base de dados normalizada são:

- Geração de aplicações mais estáveis.
- Aumento do número de tabelas utilizadas.
- Diminuição dos tamanhos médios das tabelas.

- As formas normais de Boyce Codd consistem em testes aplicados às entidades de forma que satisfaça a uma **forma normal**.
- Foi proposto três formas normais que funcionam de forma interdependentes.

Primeira Forma Normal – 1FN

Diz-se que uma tabela está na primeira forma normal quando ela não contém tabelas aninhadas. A primeira forma normal assegura que não existam repetições de valores nos atributos nem grupos repetidos de atributos das entidades de um modelo de dados.

Em uma determinada realidade, às vezes encontramos algumas informações que se repetem, retratando ocorrências de um mesmo fato dentro de uma única linha e vinculada a sua chave primária.

O objetivo da primeira forma normal é eliminar o aninhamento de tabelas para que cada tabela tenha informações de um único assunto. Não podemos ter mais de um assunto em uma tabela.

Podemos dizer que uma tabela se encontra na Primeira Forma Normal se:

- Possui chave primária;
- Não possui grupos repetitivos;
- Todos os seus atributos são atômicos, ou seja, não precisa ser decomposto.

Exemplo de 1FN: Livros

Tabela de Livros: Estrutura original

<u>IdLivro</u>	Título	Assunto	Autor1	Autor2	Autor3
21237	Os Sertões	Ficção	E. Cunha		
33455	Eletricidade básica	Física	A. Silva	B. Santos	
12312	Atlas do Brasil	Geografia	IBGE		

Estrutura normalizada na 1FN:

Tabela de Livros

<u>IdLivro</u>	Título	Assunto
21237	Os Sertões	Ficção
33455	Eletricidade básica	Física
12312	Atlas do Brasil	Geografia

Tabela Autores_Livros

<u>IdLivro</u>	<u>Autor</u>
21237	E. Cunha
33455	A. Silva
33455	B. Santos
12312	IBGE

Exemplo de 1FN: Clientes

Tabela de Clientes: Estrutura original

<u>Codigo</u>	Nome	Telefone	Tipo_tel	Rua	No	Cidade
00001	Maria	3441 8566	Residencial	Contorno	2316	Belo Horizonte
00001	Maria	3215 8751	Serviço	Contorno	2316	Belo Horizonte
00001	Maria	9158 3239	Celular	Contorno	2316	Belo Horizonte
00002	Antônio	8874 5698	Celular	Afonso Pena	5693	Belo Horizonte

Estrutura normalizada na 1FN:

Tabela: Clientes

<u>Codigo</u>	Nome	Rua	No	Cidade
00001	Maria	Contorno	2316	Belo Horizonte
00002	Antônio	Afonso Pena	5693	Belo Horizonte

Tabela: Telefone_Clientes

<u>Codigo</u>	<u>Telefone</u>	Tipo_tel
00001	3441 8566	Residencial
00001	3215 8751	Serviço
00001	9158 3239	Celular
00002	8874 5698	Celular

Primeira Forma Normal – 2FN

É dito que uma tabela está na segunda forma normal se ela atende a todos os requisitos da primeira forma normal e se os registros na tabela, que não são chaves, dependam da chave primária em sua totalidade e não apenas parte dela.

Para isso, devemos localizar os valores que dependem parcialmente da chave primária e criar tabelas separadas para conjuntos de valores que se aplicam a vários registros e relacionar estas tabelas com uma chave estrangeira.

Primeira Forma Normal – 2FN

cd_locacao	cd_filme	titulo_filme	devolucao	cd_cliente
1010	201	The Matrix	2011-10-12	743
1011	302	O Grito	2011-12-10	549
1012	201	The Matrix	2011-12-30	362

Tabela 4: Tabela não está na segunda forma normal

cd_filme	titulo_filme
201	The Matrix
302	O Grito

Tabela 5: Tabela criada para armazenar os filmes

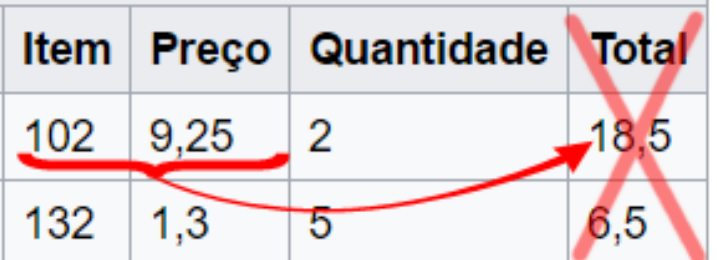
cd_locacao	cd_filme	devolucao	cd_cliente
1010	201	2011-10-12	743
1011	302	2011-12-10	549
1012	201	2011-12-30	362

Tabela 6: Tabela na segunda forma normal

Primeira Forma Normal – 3FN

Se analisarmos uma tupla e não encontrarmos um atributo **não chave** dependente de outro atributo **não chave**, podemos dizer que a entidade em questão está na terceira forma normal - contanto que esta não vá de encontro as especificações da primeira e da segunda forma normal.

Itens do pedido				
Pedido	Item	Preço	Quantidade	Total
15	102	9,25	2	18,5
15	132	1,3	5	6,5



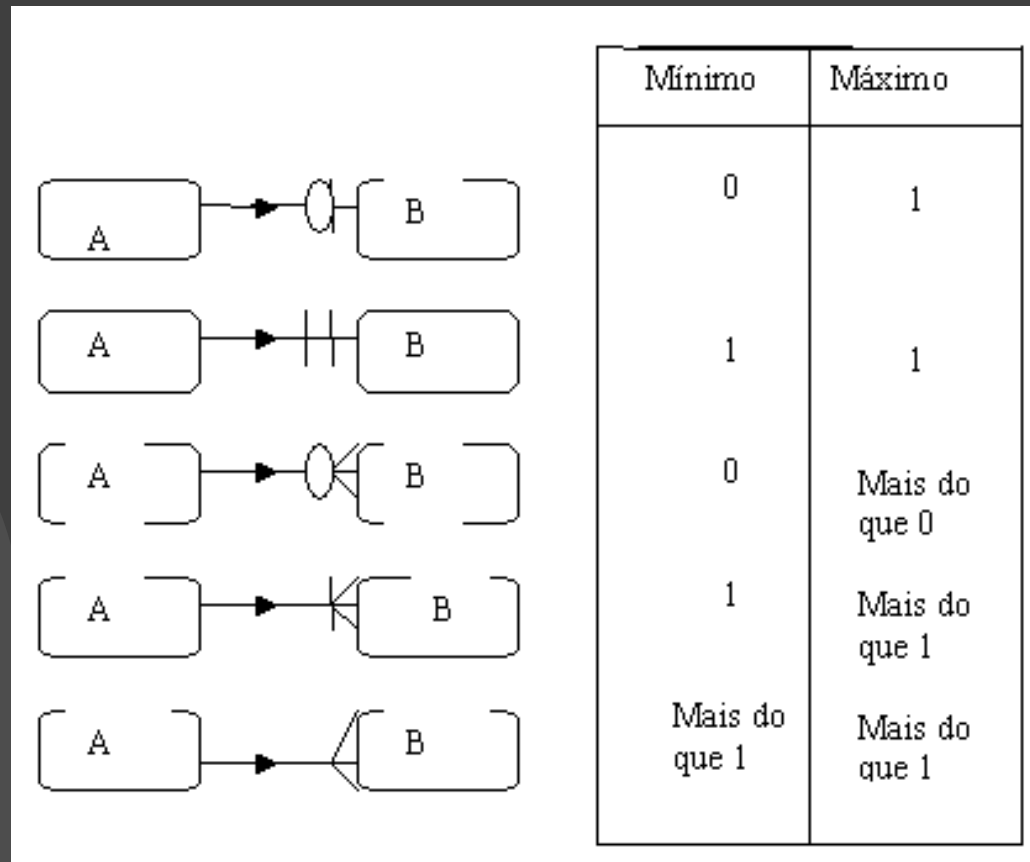
Itens do pedido			
Pedido	Item	Preço	Quantidade
15	102	9,25	2
15	132	1,3	5

- Consiste na representação gráfica dos Dados Modelados levando-se em consideração as restrições e quaisquer tipo de implicação com relação ao SGBD a ser implementado o modelo.
- O modelo conceitual é mapeado para o modelo lógico, realizando os ajustes e adaptações necessárias às condições de implementação impostas pelo SGBD.
- Uma maior riqueza de detalhes é informado ao modelo, como por exemplo tipo de dado, tamanho, índices, visões e outros tipos de objetos que se façam necessários ao modelo.

Nova nomenclatura utilizada no modelo lógico com relação ao modelo conceitual:

Entidade	→	Tabela
Atributo	→	Coluna
Identificador Único	→	Chave primária Chave única
Relacionamento	→	Chave Estrangeira
Domínio	→	Check Constraint

A cardinalidade e a ordinalidade são representadas pelo estilo de uma linha e sua extremidade, de acordo com o estilo de notação escolhido.



Conectividade	Peter Chan	James Martin
1:1		
1:N		
N:N		

Cardinalidade

Relacionamento

(0,1)



(1,1)



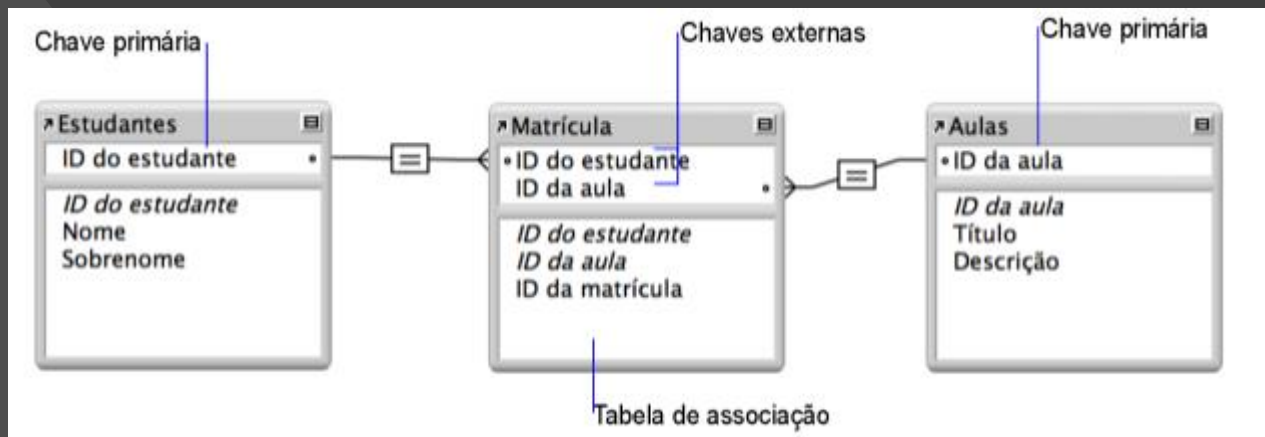
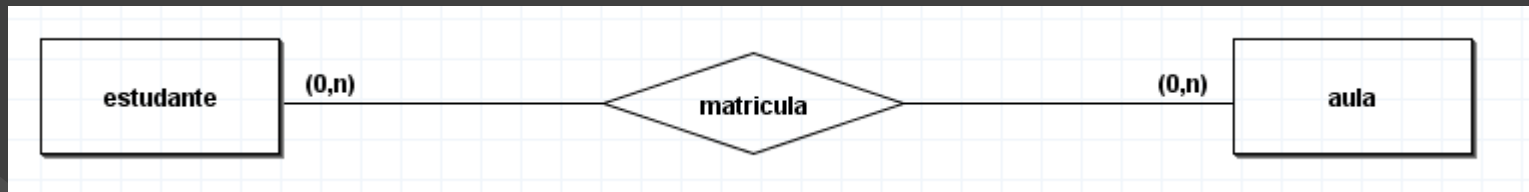
(0,N)



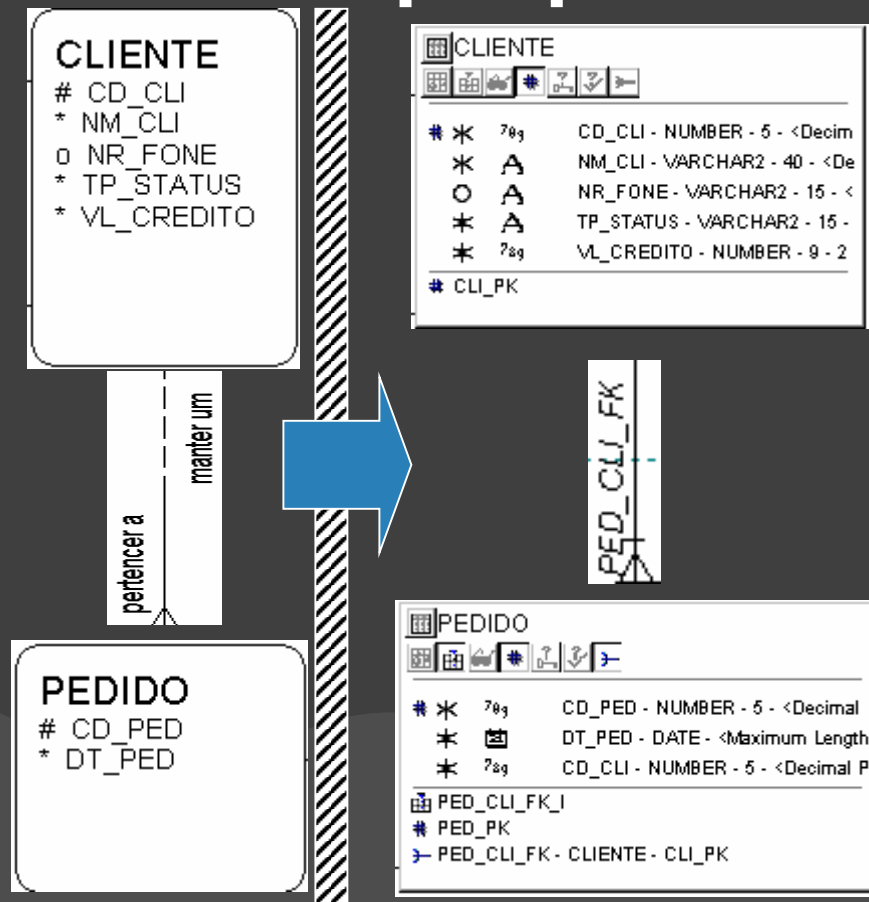
(1,N)



Um relacionamento muitos para muitos ocorre quando vários registros em uma tabela são associados a vários registros em outra tabela. Por exemplo, um relacionamento muitos para muitos existe entre clientes e produtos: clientes podem comprar vários produtos e produtos podem ser comprados por muitos clientes.



Exemplo prático:



CLIENTE

Entidade cliente para a tabela cliente
Atributos para colunas, agora com mais detalhes como tipo de dado e tamanho

Identificador único CD_CLI para chave primária CLI_PK

PEDIDO

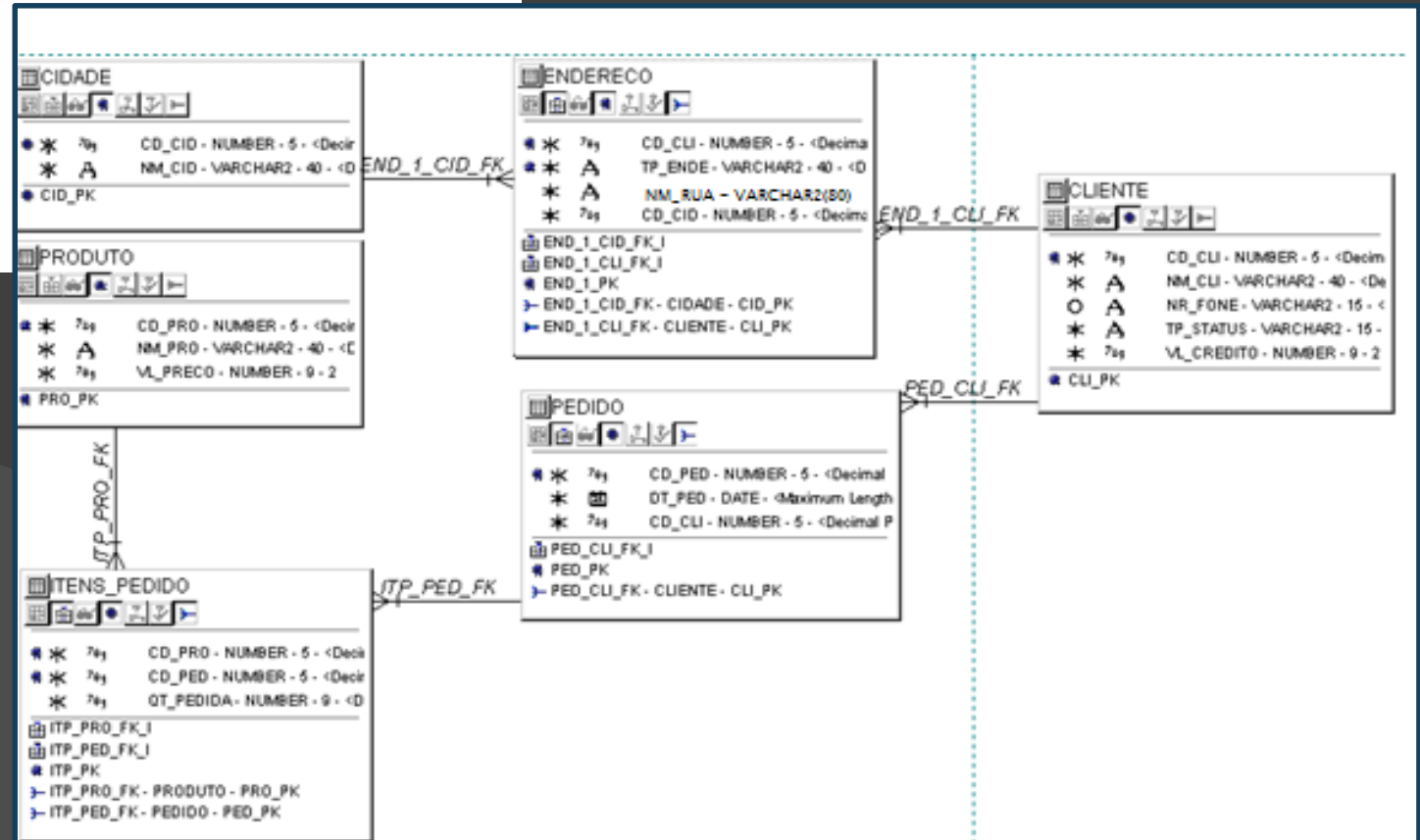
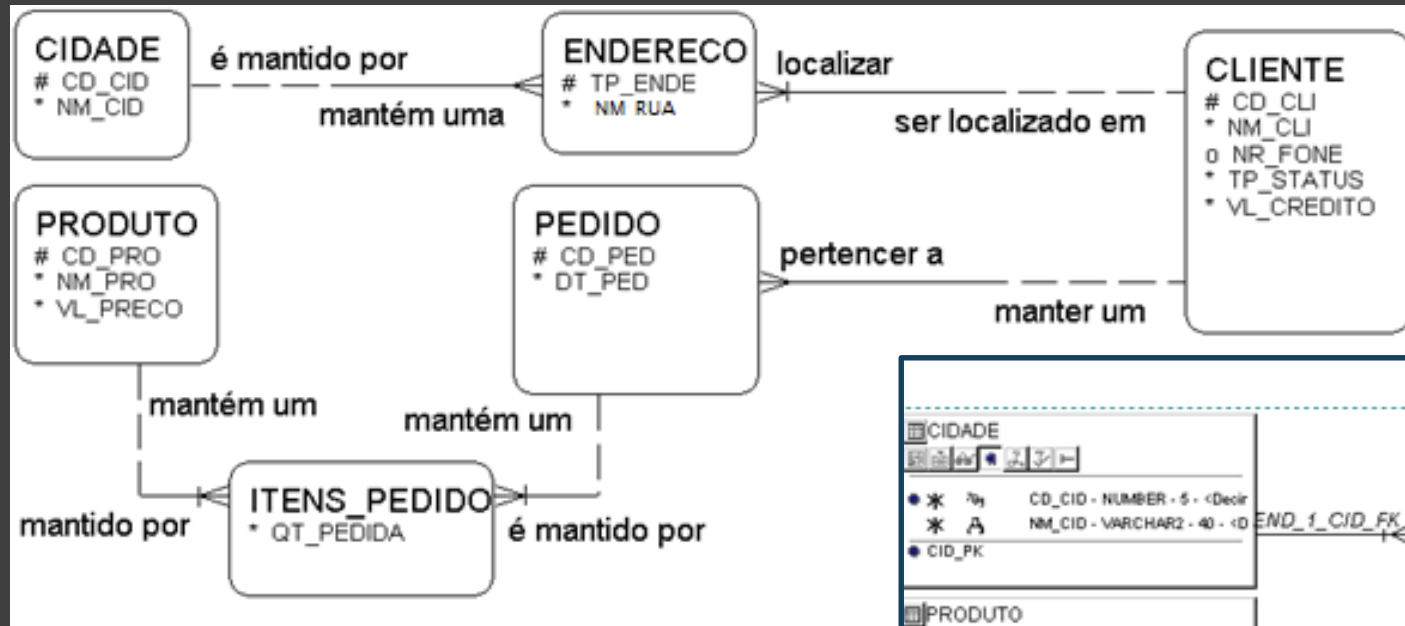
Entidade pedido para a tabela pedido
Identificador único CD_PED para chave primária PED_PK

Relacionamento para chave estrangeira ped_cli_fk

Índice para a chave estrangeira ped_cli_fk_i

Adição do atributo cd_cli obedecente ao relacionamento no modelo conceitual

Modelo E-R: Nível Lógico



Modelo E-R: Nível Lógico

