

Banco de Dados

04 – Normalização

Profª Cristina Verçosa Pérez Barrios de Souza

cristina.souza@pucpr.br





Tópicos

- › Projeto Lógico
 - Modelo Relacional
- › Conceitos iniciais
- › Normalização

PROJETO LÓGICO DE BANCO DE DADOS:

1ª Fase - Projeto Conceitual

Gera o **MODELO CONCEITUAL**

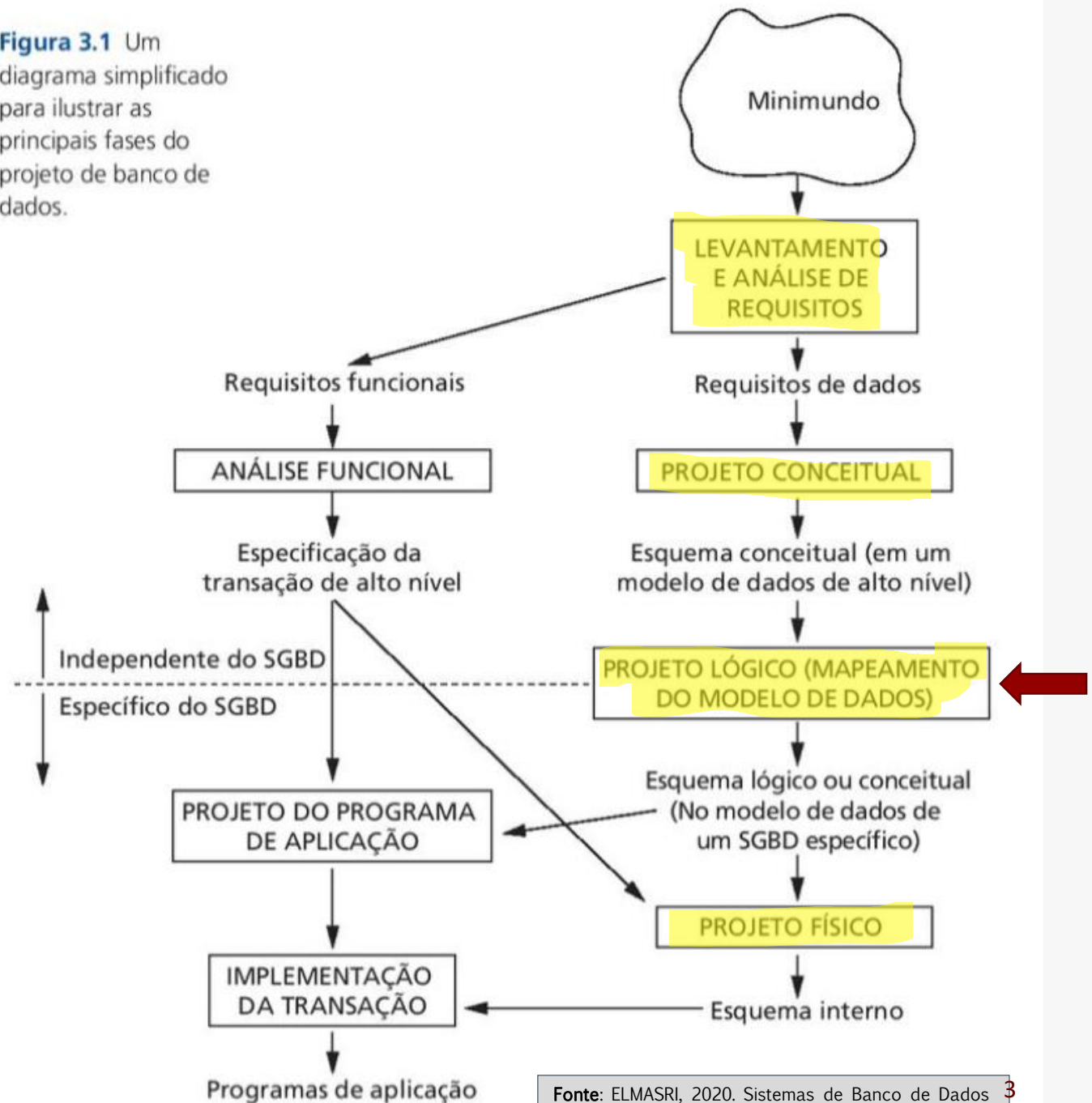
2ª Fase - Projeto Lógico

Gera o **MODELO LÓGICO**

3ª Fase - Implementação

Gera o **MODELO FÍSICO**

Figura 3.1 Um diagrama simplificado para ilustrar as principais fases do projeto de banco de dados.





Conceitos Iniciais



Contextualização

- › O objetivo do Projeto de um Banco de Dados Relacional
 - Gerar um **conjunto de esquemas de relações (tabelas)** que:
 - › Permita **armazenar informações sem redundância desnecessária** e
 - › Permita **recuperar informações facilmente**.
- › Em resumo, um **Projeto Lógico de Banco de Dados** deve:
 - Obter estruturas de dados que **visam sempre a integridade dos dados**.
 - Obter **independência de aplicação**
 - › Projetar como **os dados são e não como serão usados**, pois sempre haverá novos requisitos de aplicação.
 - Obter **independência do seu projeto físico**.



Anomalias de Inserção, Remoção e Atualização

› Considere o Esquema e a Instância a seguir

– Esquema:

› *AgenciaFuncionario* (ID_Func, Nome, Cargo, Salario, NumAg, Endereço, Telefone)

Esquema:

AgenciaEmpregado	
PK	ID_Func
	Nome
	Cargo
	Salário
	NumAg
	Endereço
	Telefone

– Instância:

ID_Func	Nome	Cargo	Salario	NumAg	Endereço	Telefone
12	José	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050
34	Luiz	Gerente	5000	2025	Rua Carlos Cavalcanti, 10	3026-8015
56	Marcela	Caixa	2000	3031	Av Silva Jardim, 321	3480-1818
78	Ruth	Gerente	5500	3031	Av Silva Jardim, 321	3480-1818
90	Antônio	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050

A tabela **AgenciaFuncionario**, armazena os dados dos **funcionários** e de **agências bancárias**.

Com isso, é possível identificar cada funcionário (**ID_Func**, **nome**, **cargo**, **salário**) e também onde ele trabalha (**agência**, **endereço**, **telefone**).

Conclusão: a tabela **AgenciaFuncionario** parece ser uma ótima opção para reduzir o número de tabelas e aumentar a velocidade das consultas, **entretanto, esta solução pode gerar várias anomalias....**



Anomalias de Inserção, Remoção e Atualização

› Exemplo: **Anomalia de Inserção**

- Ao se **INSERIR** um **novo funcionário** na **agência 1415**, devemos ter cuidado para inserir os mesmos **endereço e telefone** já cadastrados para **NumAg 1415** na tabela

Esquema:

AgenciaEmpregado	
PK	ID_Func
	Nome
	Cargo
	Salário
	NumAg
	Endereço
	Telefone

ID_Func	Nome	Cargo	Salario	NumAg	Endereço	Telefone
12	José	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050
34	Luiz	Gerente	5000	2025	Rua Carlos Cavalcanti, 10	3026-8015
56	Marcela	Caixa	2000	3031	Av Silva Jardim, 321	3480-1818
78	Ruth	Gerente	5500	3031	Av Silva Jardim, 321	3480-1818
90	Antônio	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050
97	Manoel	Caixa	2000	1415	Rua Quinze de Novembro - 100	32435050

ANOMALIA DE INSERÇÃO:

Inconsistência de Dados devido à REDUNDÂNCIA:

- Existe diferença nesses dados?
- Qual dos dados é o correto?



Anomalias de Inserção, Remoção e Atualização

› Exemplo: **Anomalia de Atualização**

- Ao se **ATUALIZAR** a **funcionário 56, Marcela, Caixa**, devemos ter cuidado para inserir os mesmos **endereço e telefone** já cadastrados para **NumAg 3031** na tabela

ID Func	Nome	Cargo	Salario	NumAg	Endereço	Telefone
12	José	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050
34	Luiz	Gerente	5000	2025	Rua Carlos Cavalcanti, 10	3026-8015
56	Marcela	Gerente	2000	3031	Avenida Silva Jardim, no. 321	34801818
78	Ruth	Gerente	5500	3031	Av Silva Jardim, 321	3480-1818
90	Antônio	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050

Esquema:

AgenciaEmpregado	
PK	ID Func
	Nome
	Cargo
	Salário
	NumAg
	Endereço
	Telefone

Redundância não controlada aumenta consumo de espaço em disco e dificulta a manutenção da integridade de dados em caso de **inserções e atualizações**.

ANOMALIA DE ATUALIZAÇÃO

Inconsistência de Dados devido à REDUNDÂNCIA:

- Existe diferença nesses dados?
- Qual dos dados é o correto?



Anomalias de Inserção, Remoção e Atualização

› Exemplo: **Anomalia de Exclusão**

- Ao se **EXCLUIR** o **funcionário 34, Luiz, Gerente**, também perderemos os dados da agência **2025** (Endereço e Telefone)

ID Func	Nome	Cargo	Salario	NumAg	Endereço	Telefone
12	José	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050
34	Luiz	Gerente	5000	2025	Rua Carlos Cavalcanti, 10	3026-8015
56	Marcela	Caixa	2000	3031	Av Silva Jardim, 321	3480-1818
78	Ruth	Gerente	5500	3031	Av Silva Jardim, 321	3480-1818
90	Antônio	Caixa	2000	1415	Rua XV de Novembro, 111	3243-5050

Esquema:

AgenciaEmpregado	
PK	ID Func
	Nome
	Cargo
	Salário
	NumAg
	Endereço
	Telefone

ANOMALIA DE EXCLUSÃO:

Perda de informação

- A exclusão de um funcionário pode eliminar também os dados de agência



Dependência Funcional

› Definição

- Seja uma tabela **T**, e sejam **X** e **Y** seus atributos.
- Dizemos que:
 - › **Y é funcionalmente dependente de X**, ou
 - › **X determina funcionalmente Y**, ou
 - › **$X \rightarrow Y$** , se e somente se:
 - › Cada valor de **X** em **T** está associado com exatamente um valor de **Y** em **T**.

› Exemplo:

- Dado o esquema **Tab_Pedidos(ID_Pedido, Descrição, PrazoEntrega)**
- O **PrazoEntrega** de um pedido depende da identificação **ID_Pedido** do pedido considerado:

ID_Pedido \rightarrow PrazoEntrega

atributo determinante

atributo dependente

IMPORTANTE: uma **chave primária** em uma **relação / tabela** determina funcionalmente todos os outros atributos não-chave na linha.



NORMALIZAÇÃO



Primeira Forma Normal (1FN)

Uma tabela está na **Primeira Forma Normal (1FN)** se, e somente se, todos os seus atributos forem **atômicos**. Ou seja, cada atributo pode ter apenas um valor por registro (tupla ou linha).

> **Como deixar na 1FN:**

1. Crie uma **nova tabela** para conter os dados do **atributo não atômico**;
2. Crie na nova tabela **um atributo** para conter o **atributo não atômico** da **tabela original**;
3. Crie na nova tabela **um atributo** para conter a **chave primária (PK)** da **tabela original**;
4. Defina uma **chave estrangeira (FK)** para garantir a relação entre a **nova tabela** e a **tabela original**;
5. Defina a **chave primária** da **nova tabela**;
6. Remova o **atributo não atômico** da **tabela original**.

> Exemplo:

ESQUEMA: *TAB_Pessoa = (ID_pessoa, nome, endereço, telefones)*

INSTÂNCIA:

ID_pessoa	nome	endereço	telefones
10	Maria	Rua das Flores, 1000	3026-8877, 3026-1415
12	Sofia	Av Joaquim Navarro, 837	4003-1819, 3401-5599
23	Eugênio	Rua Dr. Muricy, 186	99988-0022

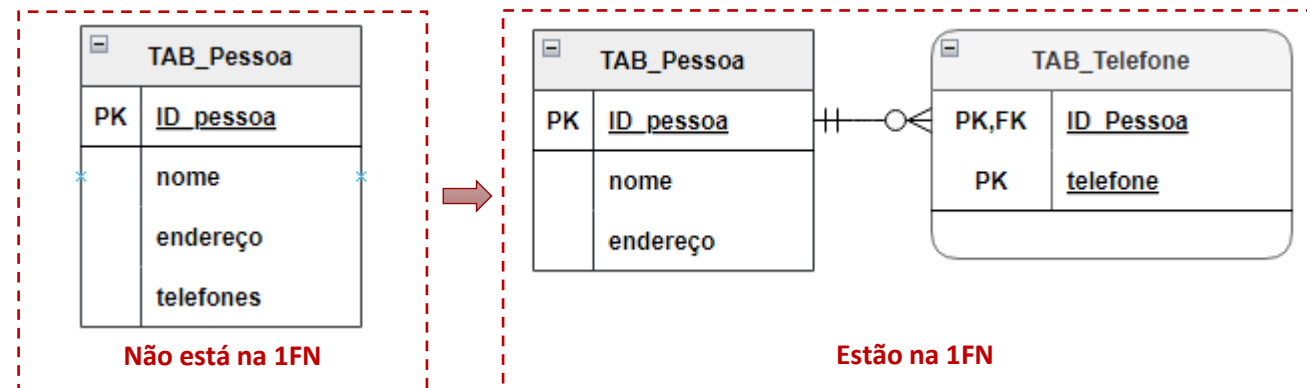
Atributo **NÃO ATÔMICO**,
pois é **MULTIVALORADO**.
Logo, não está na 1FN.



Primeira Forma Normal (1FN)

Uma tabela está na **Primeira Forma Normal (1FN)** se, e somente se, todos os seus atributos forem **atômicos**. Ou seja, cada atributo pode ter apenas um valor por registro (tupla ou linha).

> Como deixar na 1FN:



> Exemplo modificado:

ID_pessoa	nome	endereço
10	Maria	Rua das Flores, 1000
12	Sofia	Av Joaquim Navarro, 837
23	Eugênio	Rua Dr. Muricy, 186

+

ID_pessoa	telefone
10	3026-8877
10	3026-1415
12	4003-1819
12	3401-5599
23	99988-0022

Novas tabelas estão na 1FN.



Segunda Forma Normal (2FN)

Uma tabela está na **Segunda Forma Normal (2FN)** se, e somente se, estiver na 1FN e todos os atributos não chaves da tabela dependerem unicamente da **chave primária** (não podendo depender apenas de parte dela).

› **Como deixar na 2FN:**

1. Identifique as **colunas que não são funcionalmente dependentes da chave primária** da tabela
2. **Remova essa(s) coluna(s) da tabela original** e crie **uma nova tabela** com esses dados.
3. Defina uma **chave estrangeira (FK)** para garantir a relação entre a **nova tabela** e a **tabela original**;
4. Defina a **chave primária** da **nova tabela**.

› Exemplo:

ESQUEMA: *TAB_Aluno_Disciplina = (ID_aluno, ID_disciplina, nota, nome_disciplina)*

INSTÂNCIA:

PK composta

<u>ID_aluno</u>	<u>ID_disciplina</u>	nota	nome_disciplina
80432	4002	8,7	Matemática
81542	4007	5,3	Língua Portuguesa
83654	4008	9,4	Química

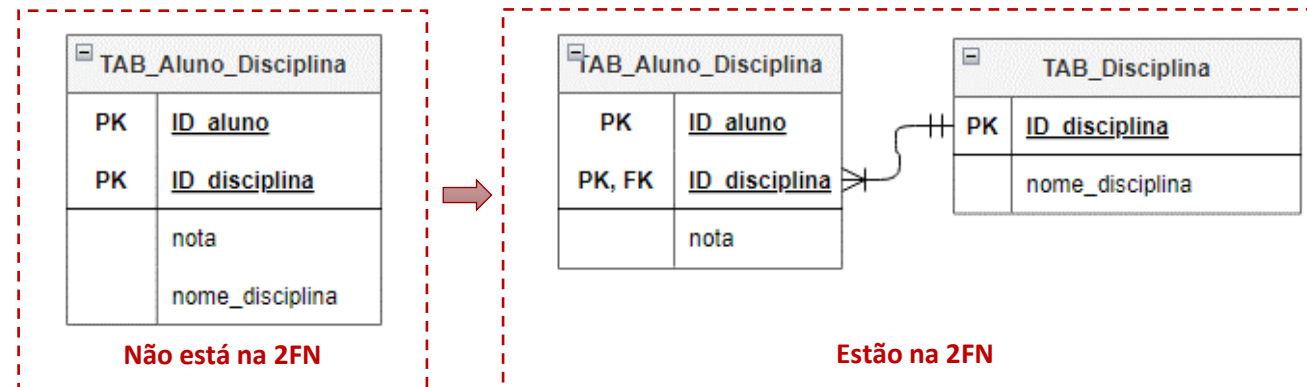
Atributo depende apenas de parte da PK.
Logo, não está na 2FN.



Segunda Forma Normal (2FN)

Uma tabela está na **Segunda Forma Normal (2FN)** se, e somente se, estiver na 1FN e todos os atributos não chaves da tabela dependerem unicamente da **chave primária** (não podendo depender apenas de parte dela).

> Como deixar na 2FN:



> Exemplo modificado:

ID_aluno	ID_disciplina	nota
80432	4002	8,7
81542	4007	5,3
83654	4008	9,4

+

ID_disciplina	nome_disciplina
4002	Matemática
4007	Língua Portuguesa
4008	Química

Novas tabelas estão na 2FN.



* Um atributo **B** é funcionalmente dependente de **A** (ou $A \rightarrow B$) se, e somente se, para cada valor de **A** só existe um valor de **B**.

Terceira Forma Normal (3FN)

Uma tabela está na **Terceira Forma Normal (3FN)** se, e somente se, estiver na 2FN e os **atributos não chave** devem ser **mutuamente independentes** e **dependentes única e exclusivamente da chave primária***.

> Como deixar na 3FN:

1. Identifique as **colunas que são funcionalmente dependentes das outras colunas não chave**
2. **Remova essa(s) coluna(s) da tabela original e crie uma nova tabela** com esses dados.
3. Defina uma **chave estrangeira (FK)** para garantir a relação entre a **nova tabela** e a **tabela original**;
4. Defina a **chave primária** da **nova tabela**.

> Exemplo:

ESQUEMA: **TAB_Funcionario** = (ID_funcionario, nome, ID_cargo, nome_cargo)

INSTÂNCIA:

<u>ID_funcionario</u>	nome	ID_cargo	nome_cargo
1937	Marta	4012	Auxiliar
1856	Gabriel	0783	Gerente
2088	Oliver	5088	Contador

PK

Atributo não depende da PK.
Logo, não está na 3FN.

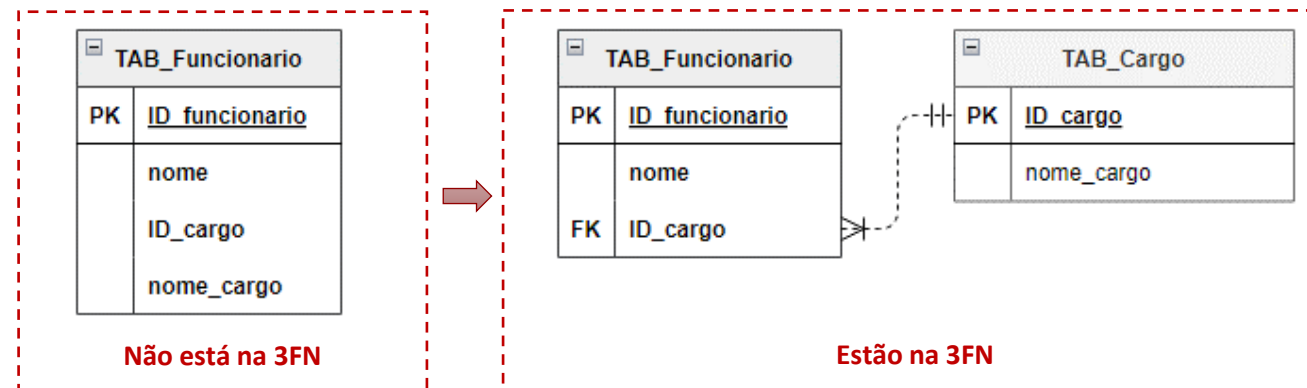


* Um atributo **B** é funcionalmente dependente de **A** (ou $A \rightarrow B$) se, e somente se, para cada valor de **A** só existe um valor de **B**.

Terceira Forma Normal (3FN)

Uma tabela está na **Terceira Forma Normal (3FN)** se, e somente se, estiver na 2FN e os **atributos não chave** devem ser **mutuamente independentes** e **dependentes única e exclusivamente da chave primária***.

> Como deixar na 2FN:



> Exemplo modificado:

ID_funcionario	nome	ID_cargo
1937	Marta	4012
1856	Gabriel	0783
2088	Oliver	5088

+

ID_cargo	nome_cargo
4012	Auxiliar
0783	Gerente
5088	Contador

Novas tabelas estão na 3FN.



Resumo da Normalização

- › O **objetivos gerais do processo de normalização** são:
 - Evitar **redundâncias indesejadas**;
 - Evitar **anomalias de inserção, atualização e exclusão**;
 - Produzir um projeto que seja uma "**boa representação do mundo real**" – isto é, que seja **intuitivamente fácil de entender** e uma **boa base para crescimento futuro**;
 - **Simplificar** a implementação de **restrições de integridade**.