



Instituto Federal de Educação, Ciência e Tecnologia da Bahia - IFBA
Departamento de Informática
Integrado / Análise e Desenvolvimento de Sistemas / Licenciatura em Computação

Normalização de Relações

André L. R. Madureira <andre.madureira@ifba.edu.br>
Doutorando em Ciência da Computação (UFBA)
Mestre em Ciência da Computação (UFBA)
Engenheiro da Computação (UFBA)

Normalização de relações

- Processo de analisar esquemas de relação usando suas dependências funcionais (DFs) e atributos visando:
 - Minimizar redundâncias de dados
 - Minimizar anomalias de atualização
- *“Processo de melhoria na qualidade de um banco de dados”*
 - Otimização de banco de dados (desempenho e espaço necessário para armazenar os dados)

Normalização de relações

- Consiste em testes realizados sobre uma relação para verificar se ela satisfaz uma **forma normal**
 - Os testes são baseados nas dependências funcionais e atributos de uma relação
- Inicialmente Edgar Codd propôs 3 formas normais em 1972:
 - 1ª forma normal (1FN)
 - 2ª forma normal (2FN)
 - 3ª forma normal (3FN)

Normalização de relações

- Consiste em testes realizados sobre uma relação para verificar se ela satisfaz uma **forma normal**
 - Os testes são baseados nas dependências funcionais e atributos de uma relação
- Inicialmente Edgar Codd propôs 3 formas normais em 1972:
 - 1ª forma normal (1FN)
 - 2ª forma normal (2FN)
 - 3ª forma normal (3FN)

Relações que não satisfazem à uma forma normal são decompostas em relações menores, que atendam aos testes

Forma normal

- Uma relação **está** em uma forma normal se ela satisfaz TODOS os critérios associados a esta forma normal
 - Uma relação que atende a formas normais de grau maior, também atende às formas normais de grau menor
 - **Ex:** se uma relação está na 3FN, ela também está na 2FN e na 1FN
 - **Grau de normalização:** forma normal mais alta da relação

1ª Forma Normal (1FN)

- Não aceita **atributos multivalorados**, e nem **atributos repetidos**
 - Domínio dos atributos inclui apenas:
 - Valores simples, atômicos e indivisíveis
 - **Ex:** CPF, RG, ano_nascimento, primeiro_nome_mae

Pessoa				
id	<u>cpf</u>	<u>rg</u>	<u>ano_nasc</u>	<u>primeiro_nome_mae</u>
1	415.254.784-00	01354021-67	1992	Carla
2	661.457.745-87	85410234-87	1985	Katia
3	325.471.154-64	74520132-95	1970	Jussara

Exemplo de relação que NÃO pertence a 1FN

Pessoa			
id	<u>cpf</u>	<u>ano_nasc</u>	<u>endereco</u>
1	415.254.784-00	1992	Rua Castelo 74
2	661.457.745-87	1985	Av ACM 85
3	325.471.154-64	1970	Rua de Baixo 101

Exemplo de relação que NÃO pertence a 1FN

Pessoa			
id	<u>cpf</u>	<u>ano_nasc</u>	<u>endereço</u>
1	415.254.784-00	1992	Rua Castelo 74
2	661.457.745-87	1985	Av ACM 85
3	325.471.154-64	1970	Rua de Baixo 101

Endereço é um **atributo multivalorado**

A mesma pessoa pode ter múltiplos endereços diferentes

Exemplo de relação que NÃO pertence a 1FN

Pessoa			
id	<u>cpf</u>	<u>ano_nasc</u>	<u>endereço</u>
1	415.254.784-00	1992	Rua Castelo 74
2	661.457.745-87	1985	Av ACM 85
3	325.471.154-64	1970	Rua de Baixo 101

Endereço é um **atributo multivalorado**

A mesma pessoa pode ter múltiplos endereços diferentes

Solução:

Colocar endereço em uma relação à parte, e associar ela a Pessoa

Exemplo de adequação de relação para 1FN

PK Pessoa		
<u>id</u>	cpf	ano_nasc
1	415.254.784-00	1992
2	661.457.745-87	1985
3	325.471.154-64	1970

Endereço FK		
rua	numero	id_pessoa
Rua Castelo	74	1
Av ACM	85	2
Rua de Baixo	101	3



Relação
“Pessoa” agora
está na 1FN

Exemplo de relação que NÃO pertence a 1FN

Funcionario		
<u>id</u>	cpf	cargo
1	415.254.784-00	Padeiro
2	661.457.745-87	Secretaria
3	325.471.154-64	Padeiro

Exemplo de relação que NÃO pertence a 1FN

Funcionario		
<u>id</u>	cpf	cargo
1	415.254.784-00	Padeiro
2	661.457.745-87	Secretaria
3	325.471.154-64	Padeiro

Cargo é um **atributo repetido**

Exemplo de relação que NÃO pertence a 1FN

Funcionario		
<u>id</u>	cpf	cargo
1	415.254.784-00	Padeiro
2	661.457.745-87	Secretaria
3	325.471.154-64	Padeiro

Cargo é um **atributo repetido**

Existem pessoas com o mesmo cargo

Exemplo de relação que NÃO pertence a 1FN

Funcionario		
<u>id</u>	cpf	cargo
1	415.254.784-00	Padeiro
2	661.457.745-87	Secretaria
3	325.471.154-64	Padeiro

Cargo é um **atributo repetido**

Existem pessoas com o mesmo cargo

Solução:

Transformar “cargo” em uma relação de “cadastro”.
Criar uma tabela de junção que associa Funcionario a Cargo

Exemplo de adequação de relação para 1FN

Funcionario	
<u>id</u>	cpf
1	415.254.784-00
2	661.457.745-87
3	325.471.154-64

Cargo	
<u>id</u>	nome
1	Padeiro
2	Gerente
3	Secretaria

Exemplo de adequação de relação para 1FN

Funcionario	
<u>id</u>	cpf
1	415.254.784-00
2	661.457.745-87
3	325.471.154-64

Cargo	
<u>id</u>	nome
1	Padeiro
2	Gerente
3	Secretaria

Funcionario_Cargo	
id_func	id_cargo
1	1
2	3
3	1

Exemplo de adequação de relação para 1FN

Funcionario	
<u>id</u>	cpf
1	415.254.784-00
2	661.457.745-87
3	325.471.154-64

PK

Cargo	
<u>id</u>	nome
1	Padeiro
2	Gerente
3	Secretaria

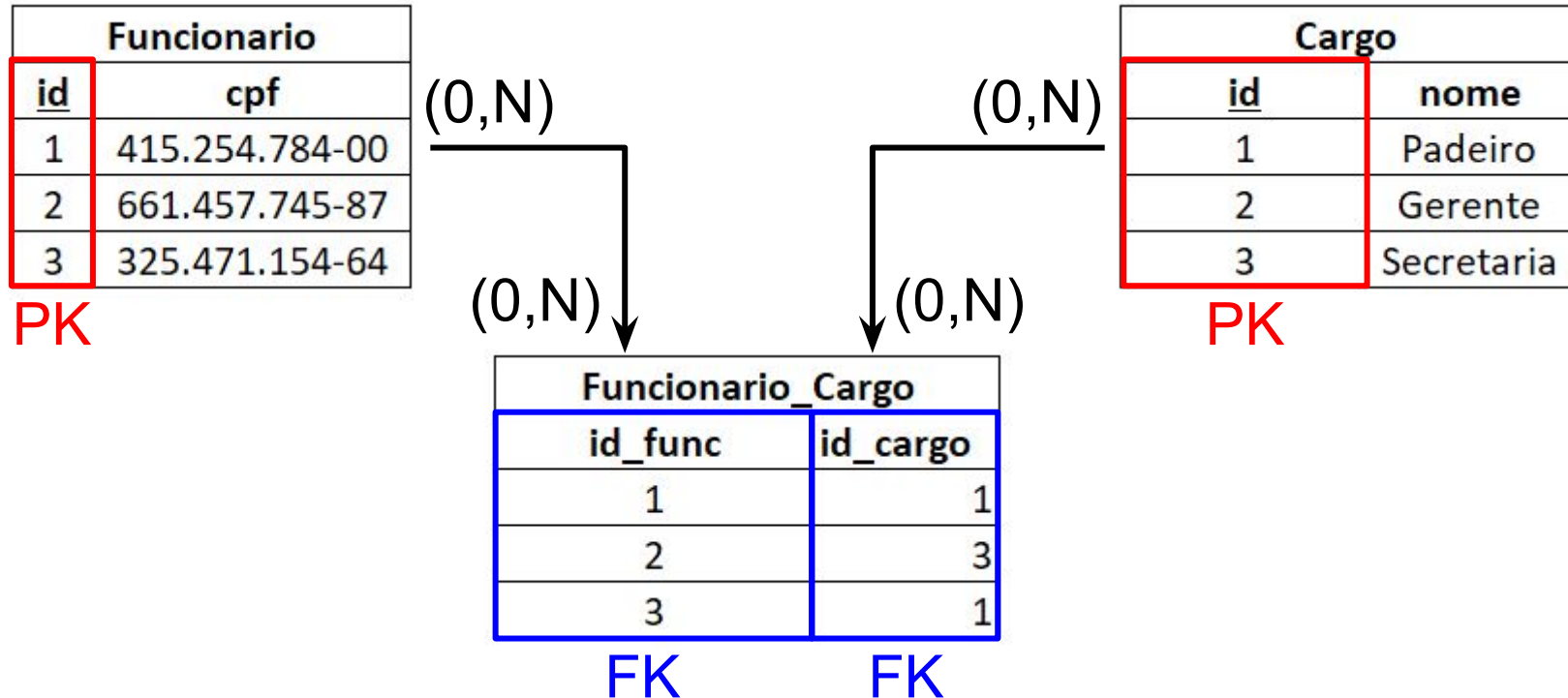
PK

Funcionario_Cargo	
id_func	id_cargo
1	1
2	3
3	1

FK

FK

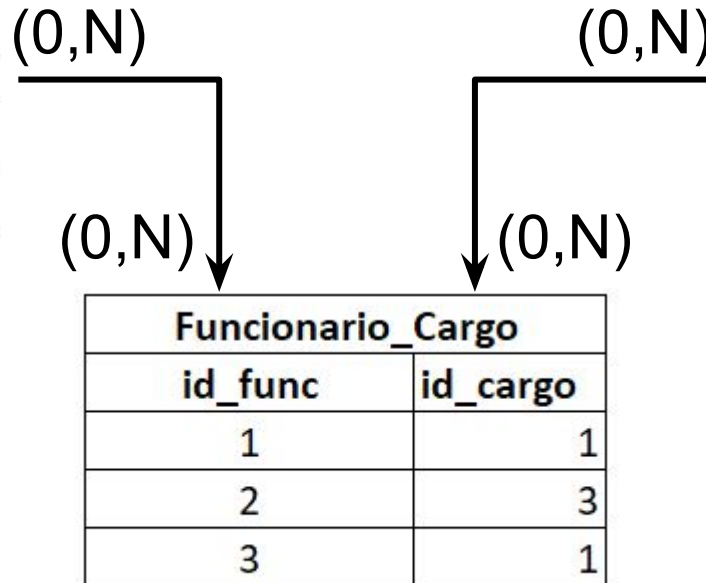
Exemplo de adequação de relação para 1FN



Exemplo de adequação de relação para 1FN

Funcionario	
<u>id</u>	cpf
1	415.254.784-00
2	661.457.745-87
3	325.471.154-64

Cargo	
<u>id</u>	nome
1	Padeiro
2	Gerente
3	Secretaria



Relação
“*Funcionario*”
agora está na 1FN

Exercício - Coloque seu projeto lógico na 1FN

- Ajuste o modelo lógico do seu projeto para estar na 1FN
 - Faça os ajustes necessários
 - Considere as boas práticas de bancos de dados
 - Semântica clara (sem ambiguidade)
 - Evite valores NULL
 - Evite tuplas falsas
 - Reduza informações redundantes nas tuplas

Dependência Funcional Total

- Uma DF $X \twoheadrightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

Dependência Funcional Total

- Uma DF $X \rightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

Dependência Funcional Total

- Uma DF $X \rightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{horas_trab}$

Dependência Funcional Total

- Uma DF $X \rightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{horas_trab}$

Sera que podemos dizer que:

$\text{cpf_func} \rightarrow \text{horas_trab}$

OU

$\text{proj_nome} \rightarrow \text{horas_trab}$

?

Dependência Funcional Total

- Uma DF $X \rightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{horas_trab}$

Sera que podemos dizer que:

$\text{cpf_func} \rightarrow \text{horas_trab}$

ou

$\text{proj_nome} \rightarrow \text{horas_trab}$

?

Dependência Funcional Total

- Uma DF $X \rightarrow Y$ é uma DF Total se a remoção de qualquer atributo de X significar que a dependência não se mantém mais

DF Total

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

(cpf_func, proj_num) \rightarrow horas_trab
é DF Total

Sera que podemos dizer que:

cpf_func \rightarrow horas_trab

ou

proj_nome \rightarrow horas_trab

?

Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{proj_nome}$

Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{proj_nome}$

Sera que podemos dizer que:

$\text{cpf_func} \rightarrow \text{proj_nome}$

OU

$\text{proj_nome} \rightarrow \text{proj_nome}$

?

Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{proj_nome}$

~~$\text{cpf_func} \rightarrow \text{proj_nome}$~~

Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

DF

Funcionario_Projeto				
<u>cpf_func</u>	<u>proj_num</u>	proj_nome	horas_trab	func_nome

PK

(cpf_func, proj_num) \rightarrow proj_nome

~~cpf_func \rightarrow proj_nome~~

proj_num \rightarrow proj_nome



Dependência Funcional Parcial

- Uma DF $X \rightarrow Y$ é uma DF Parcial se algum atributo de X puder ser removido e a DF ainda se mantenha

DF Parcial

Funcionario_Projeto				
cpf_func	proj_num	proj_nome	horas_trab	func_nome

PK

$(\text{cpf_func}, \text{proj_num}) \rightarrow \text{proj_nome}$
é **DF Parcial**

~~$\text{cpf_func} \rightarrow \text{proj_nome}$~~

$\text{proj_num} \rightarrow \text{proj_nome}$



2ª Forma Normal (2FN)

- Uma relação **R** está na 2FN, se:
 - Ela também estiver na 1FN
 - Cada atributo não principal **A** em **R** for DF total da chave primária de **R**

2ª Forma Normal (2FN)

- Uma relação **R** está na 2FN, se:
 - Ela também estiver na 1FN
 - Cada atributo não principal **A** em **R** for DF total da chave primária de **R**

Atributos não principais são todos aqueles que não pertencem a chave primária

2ª Forma Normal (2FN)

- Uma relação **R** está na 2FN, se:
 - Ela também estiver na 1FN
 - Cada atributo não principal **A** em **R** for DF total da chave primária de **R**

Atributos não principais são todos aqueles que não pertencem a chave primária



Exemplo de relação que NÃO está na 2FN

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

Exemplo de relação que NÃO está na 2FN

DF

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

(cpf_func, proj_numero) -> horas_trab
é DF Total ?

Exemplo de relação que NÃO está na 2FN

DF

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

cpf_func -> horas_trab

proj_numero -> horas_trab

Exemplo de relação que NÃO está na 2FN

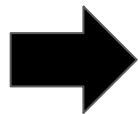
DF Total

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

~~cpf_func → horas_trab~~

~~proj_numero → horas_trab~~



(cpf_func, proj_numero) → horas_trab
é DF Total

Exemplo de relação que NÃO está na 2FN

DF



Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

(cpf_func, proj_numero) -> func_nome
é DF Total ?

Exemplo de relação que NÃO está na 2FN

DF

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

cpf_func -> func_nome

proj_numero -> func_nome

Exemplo de relação que NÃO está na 2FN

DF

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

cpf_func -> func_nome



~~proj_numero -> func_nome~~

Exemplo de relação que NÃO está na 2FN

DF Parcial

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

cpf_func -> func_nome

~~proj_numero -> func_nome~~



(cpf_func, proj_numero) -> func_nome
é **DF Parcial**

Exemplo de relação que NÃO está na 2FN

DF Parcial

Funcionario_Projeto				
cpf_func	proj_numero	horas_trab	func_nome	proj_nome
777.888.999-00	205	40	Carlos	Represa de Itaibé
111.222.333-44	206	32	Amanda	Placas solares
000.555.777-11	207	44	Kécia	Restaurante

PK

Como existem atributos que **dependem parcialmente** da chave primária (*cpf_func*, *proj_numero*), a relação **NÃO ESTÁ** na 2FN

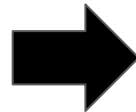
Exemplo adequação de relação para 2FN

DF Total

Funcionario_Projeto		
<u>cpf_func</u>	<u>proj_numero</u>	horas_trab
777.888.999-00	205	40
111.222.333-44	206	32
000.555.777-11	207	44

PK

cpf_func → horas_trab
proj_numero → horas_trab



(cpf_func, proj_numero) → horas_trab
é DF Total

Exemplo adequação de relação para 2FN

Funcionario_Projeto		
<u>cpf_func</u>	<u>proj_numero</u>	horas_trab
777.888.999-00	205	40
111.222.333-44	206	32
000.555.777-11	207	44

DF Total

Funcionario	
<u>cpf</u>	nome
777.888.999-00	Carlos
111.222.333-44	Amanda
000.555.777-11	Kécia

PK

Não é preciso testar se **cpf** -> **nome** é DF Total.
Isto se deve ao fato de termos apenas um atributo principal (*cpf*).

Exemplo adequação de relação para 2FN

Funcionario_Projeto		
<u>cpf_func</u>	<u>proj_numero</u>	horas_trab
777.888.999-00	205	40
111.222.333-44	206	32
000.555.777-11	207	44

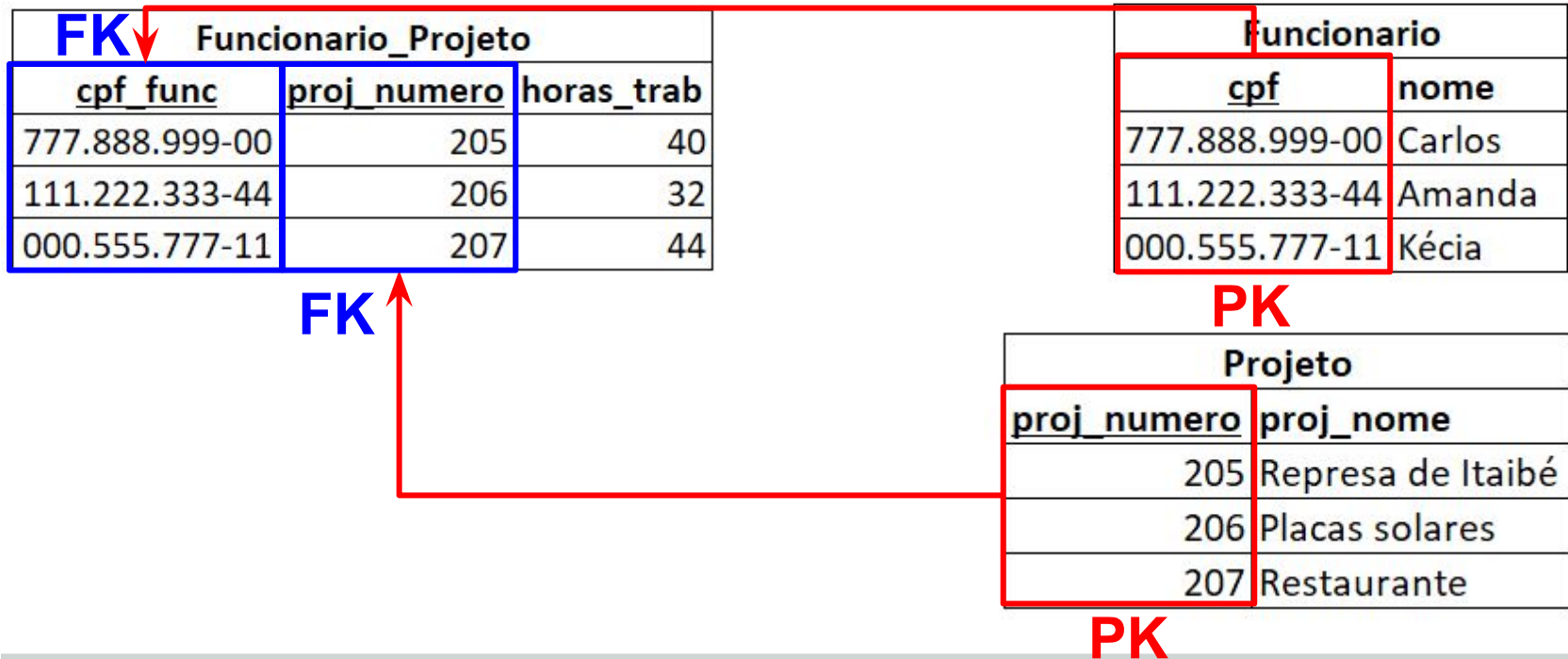
Funcionario	
<u>cpf</u>	nome
777.888.999-00	Carlos
111.222.333-44	Amanda
000.555.777-11	Kécia

DF Total

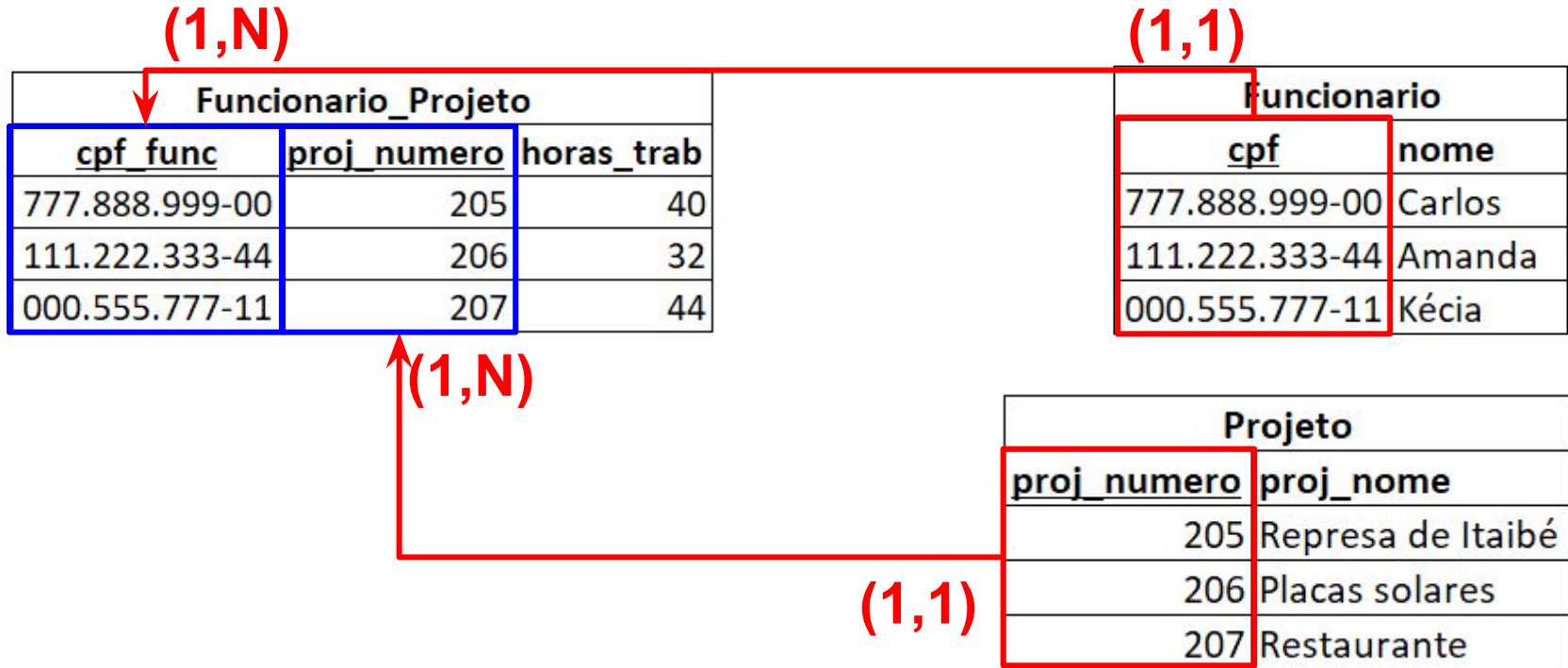
Projeto	
<u>proj_numero</u>	proj_nome
205	Represa de Itaibé
206	Placas solares
207	Restaurante

PK

Exemplo adequação de relação para 2FN



Exemplo adequação de relação para 2FN



Exercício - Coloque seu projeto lógico na 2FN

- Ajuste o modelo lógico do seu projeto para estar na 2FN
 - Faça os ajustes necessários
 - Considere as boas práticas de bancos de dados
 - Semântica clara (sem ambiguidade)
 - Evite valores NULL
 - Evite tuplas falsas
 - Reduza informações redundantes nas tuplas

Dependência Funcional Transitiva

- Uma DF $X \rightarrow Y$ é uma DF transitiva se existe:
 - $X \rightarrow W$
 - $W \rightarrow Y$
- Onde **W** é um atributo não principal (“*W não é chave primária*”)

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

Dependência Funcional Transitiva

- Uma DF $X \rightarrow Y$ é uma DF transitiva se existe:
 - $X \rightarrow W$
 - $W \rightarrow Y$
- Onde W é um atributo não principal (*“W não é chave primária”*)

DF

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

PK

Dependência Funcional Transitiva

- Uma DF $X \rightarrow Y$ é uma DF transitiva se existe:
 - $X \rightarrow W$
 - $W \rightarrow Y$
- Onde W é um atributo não principal (“ W não é chave primária”)

DF

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

PK

DF

Dependência Funcional Transitiva

DF

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

PK

DF

Temos:

cpf_func -> num_depto
num_depto -> cpf_gerente

Dependência Funcional Transitiva

DF

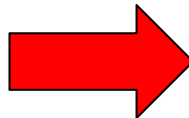
Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

PK

DF

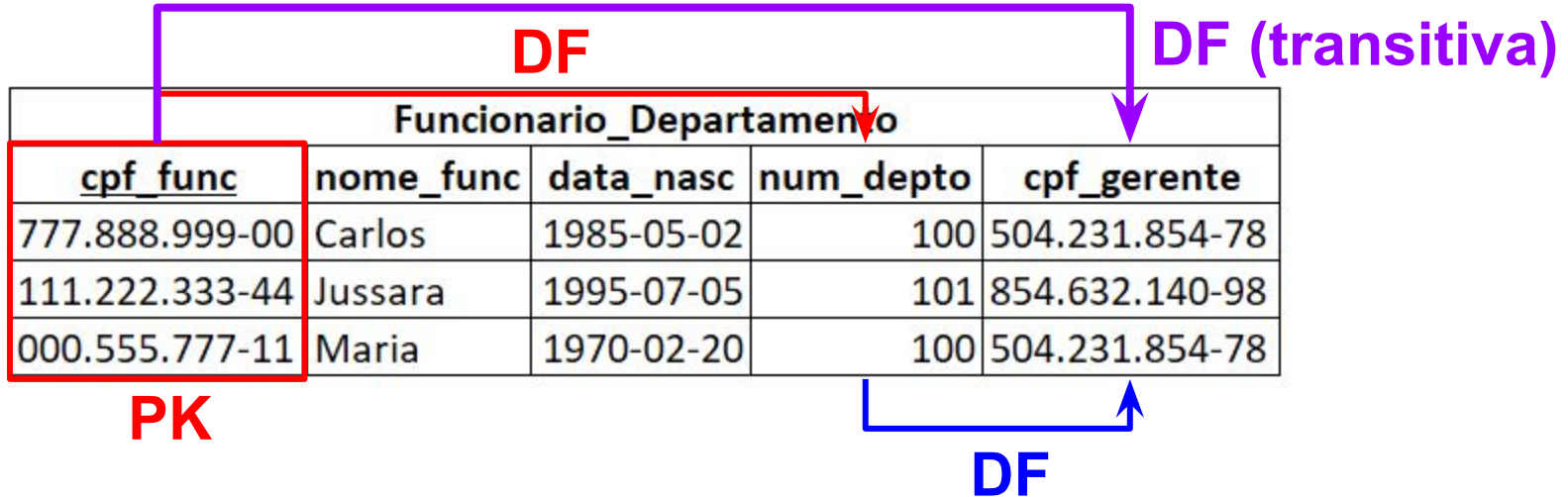
Temos:

cpf_func -> num_depto
num_depto -> cpf_gerente



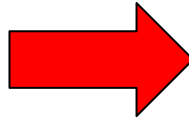
cpf_func -> cpf_gerente
existe e é **DF transitiva**

Dependência Funcional Transitiva



Temos:

cpf_func -> num_depto
num_depto -> cpf_gerente



cpf_func -> cpf_gerente
existe e é **DF transitiva**

3ª Forma Normal (3FN)

- Uma relação **R** está na 3FN, se:
 - Ela também estiver na 2FN
 - Nenhum atributo não principal (que não pertence a chave primária) de **R** for **DF transitiva** da chave primária
 - Isto é, atributos não devem depender de outros atributos não principais

Exemplo de relação que NÃO está na 3FN

DF

<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-12	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

Temos as seguintes DFs:

cpf_func -> **num_depto** -> **cpf_gerente**

Exemplo adequação de relação para 3FN

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

Exemplo adequação de relação para 3FN

Funcionario_Departamento				
<u>cpf_func</u>	nome_func	data_nasc	num_depto	cpf_gerente
777.888.999-00	Carlos	1985-05-02	100	504.231.854-78
111.222.333-44	Jussara	1995-07-05	101	854.632.140-98
000.555.777-11	Maria	1970-02-20	100	504.231.854-78

Pertence a tabela **Funcionário**

Pertence a tabela **Departamento**

Exemplo adequação de relação para 3FN

Funcionario		
<u>cpf</u>	nome	data_nasc
777.888.999-00	Carlos	1985-05-02
111.222.333-44	Jussara	1995-07-05
000.555.777-11	Maria	1970-02-20

Departamento	
<u>numero</u>	cpf_gerente
100	504.231.854-78
101	854.632.140-98

Exemplo adequação de relação para 3FN

Funcionario		
<u>cpf</u>	nome	data_nasc
777.888.999-00	Carlos	1985-05-02
111.222.333-44	Jussara	1995-07-05
000.555.777-11	Maria	1970-02-20

Departamento	
<u>numero</u>	cpf_gerente
100	504.231.854-78
101	854.632.140-98

Funcionario_Departamento	
cpf_func	num_depto
777.888.999-00	100
111.222.333-44	101
000.555.777-11	100

Exemplo adequação de relação para 3FN

Funcionario		
<u>cpf</u>	nome	data_nasc
777.888.999-00	Carlos	1985-05-02
111.222.333-44	Jussara	1995-07-05
000.555.777-11	Maria	1970-02-20

Departamento	
<u>numero</u>	cpf_gerente
100	504.231.854-78
101	854.632.140-98

(1,1)
↓
(1,N)

Funcionario_Departamento	
cpf_func	num_depto
777.888.999-00	100
111.222.333-44	101
000.555.777-11	100

(1,1)
↓
(1,N)

Resumo das Formas Normais

- Primeira Forma Normal (1FN)
 - Não há **atributos multivalorados**, e nem **repetidos**
- Segunda Forma Normal (2FN)
 - Não há **DF parcial entre** atributos chaves e não chaves
- Terceira Forma Normal (3FN)
 - Não há **DF transitiva** entre atributos não chaves

Exercício - Coloque seu projeto lógico na 3FN

- Ajuste o modelo lógico do seu projeto para estar na 3FN
 - Faça os ajustes necessários
 - Considere as boas práticas de bancos de dados
 - Semântica clara (sem ambiguidade)
 - Evite valores NULL
 - Evite tuplas falsas
 - Reduza informações redundantes nas tuplas

Referencial Bibliográfico

- KORTH, H.; SILBERSCHATZ, A.; SUDARSHAN, S. **Sistemas de bancos de dados**. 5. ed. Rio de Janeiro: Ed. Campus, 2006.
- DATE, C. J. **Introdução a sistemas de bancos de dados**. Rio de Janeiro: Ed. Campus, 2004. Tradução da 8ª edição americana.