

# Engenharia Reversa e Normalização

Banco de Dados I

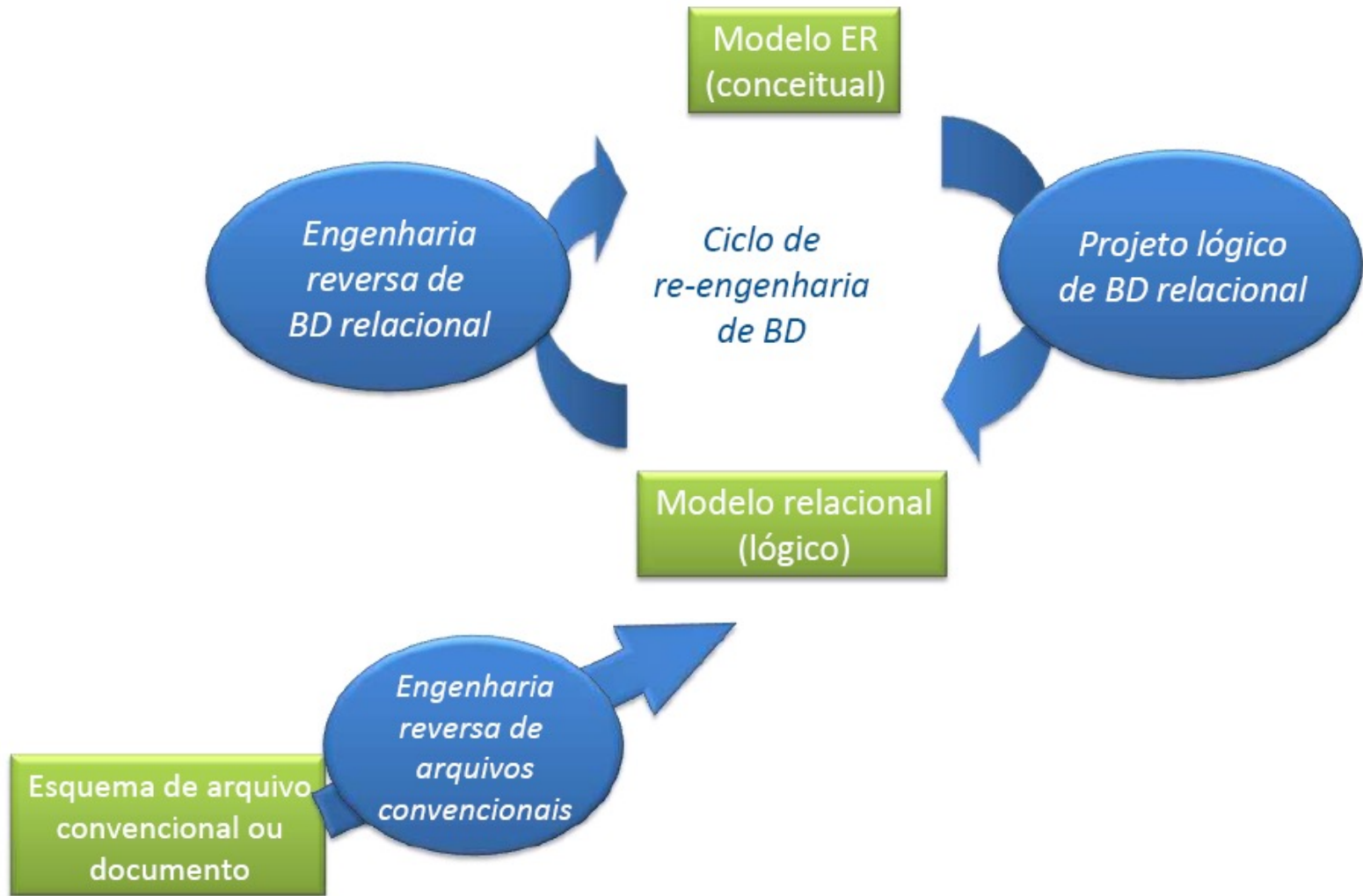


JESUÍTAS BRASIL



Somos infinitas possibilidades

# Engenharia Reversa



# Engenharia Reversa

## Normalização

Processo que transforma um esquema de dados qualquer em um modelo relacional



Normalização é executada para todos esquemas de documentos disponíveis

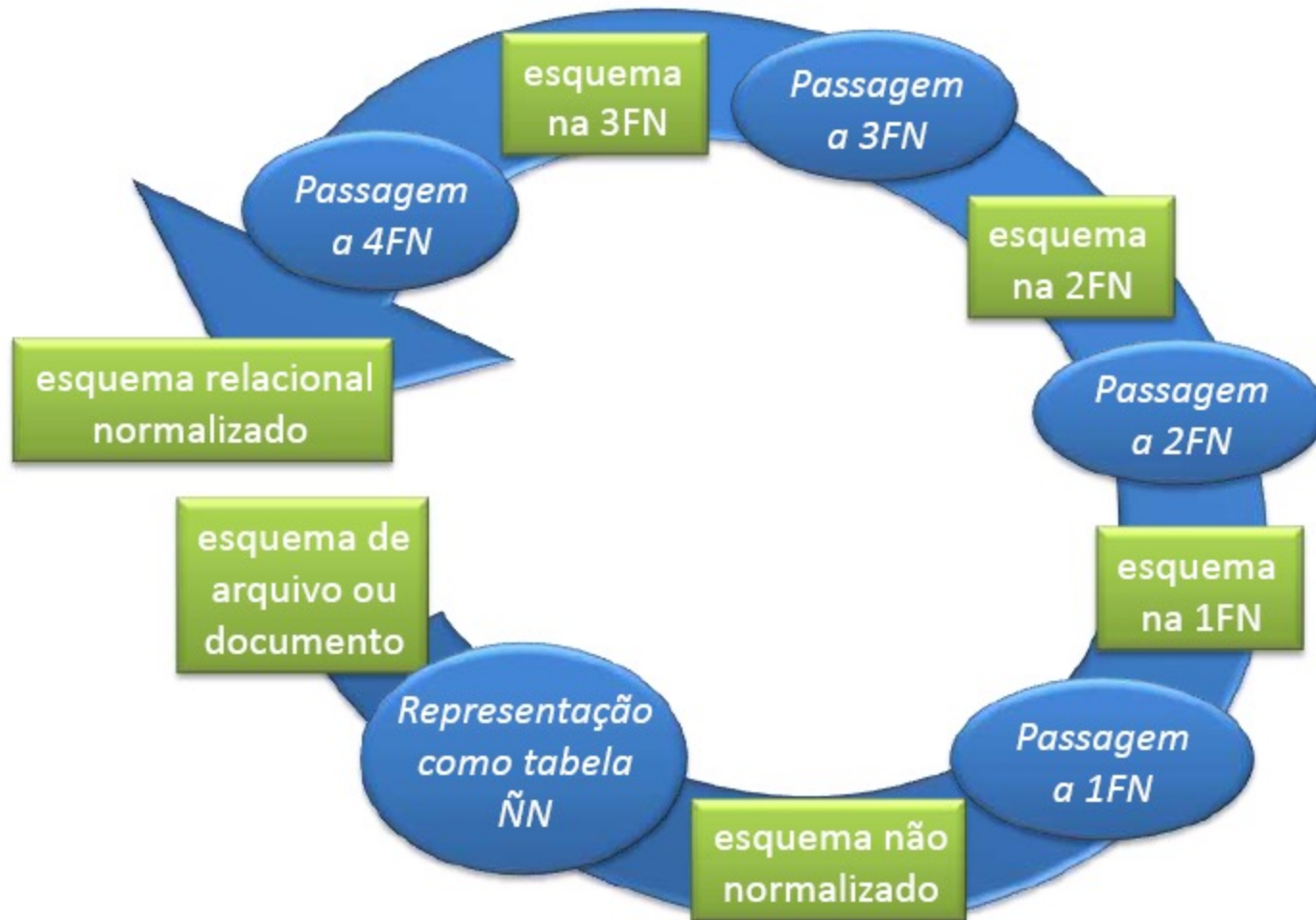
# Normalização

## Objetivos

- Eliminar **redundâncias** de dados
- Eliminar **estruturas inexistentes** (atributos multivalorados)
- Eliminar **anomalias**
  - Inserção
  - Deleção
  - Alteração

# Normalização

## Esquema de normalização



# Normalização

## Documento Exemplo

RELATÓRIO DE ALOCAÇÃO A PROJETO					
CÓDIGO DO PROJETO: LSC001			TIPO: Novo Desenv.		
DESCRIÇÃO: Sistema de Estoque					
CÓDIGO DO EMPREGADO	NOME	CATEGORIA FUNCIONAL	SALÁRIO	DATA DE INÍCIO NO PROJETO	TEMPO ALOCADO AO PROJETO
2146	João	A1	4	1/11/91	24
3145	Sílvio	A2	4	2/10/91	24
6126	José	B1	9	3/10/92	18
1214	Carlos	A2	4	4/10/92	18
8191	Mário	A1	4	1/11/92	12
CÓDIGO DO PROJETO: PAG02			TIPO: Manutenção		
DESCRIÇÃO: Sistema de RH					
CÓDIGO DO EMPREGADO	NOME	CATEGORIA FUNCIONAL	SALÁRIO	DATA DE INÍCIO NO PROJETO	TEMPO ALOCADO AO PROJETO
8191	Mário	A1	4	1/05/93	12
4112	João	A2	4	4/01/91	24
6126	José	B1	9	1/11/92	12

# Normalização

- **Primeiro passo (ÑN)**

Tabela não-normalizada

Tabela não-primeira-forma-normal

Possui uma ou mais tabelas aninhadas

Utilizaremos a notação ÑN

- **Tabela aninhada**

Coluna que, ao invés de conter valores atômicos, contém tabelas aninhadas

Grupo repetido

Coluna multi-valorada

Coluna não-atômica

# Normalização

Exemplo de tabela **ÑN** com tabela aninhada

CódProj	Tipo	Descr	Emp					
			CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	Novo Desenv.	Sistema de Estoque	2146	João	A1	4	1/11/91	24
			3145	Sílvio	A2	4	2/10/91	24
			6126	José	B1	9	3/10/92	18
			1214	Carlos	A2	4	4/10/92	18
			8191	Mário	A1	4	1/11/92	12
PAG02	Manutenção	Sistema de Pagamento	101	Mário	A1	4	1/05/93	12
				João	A2	4	4/01/91	24
				José	B1	9	1/11/92	12

tabela  
aninhada

**Esquema textual**

Proj(CodProj, Tipo, Descr, (CodEmp, Nome, Cat, Sal, DataIni, TempAl))



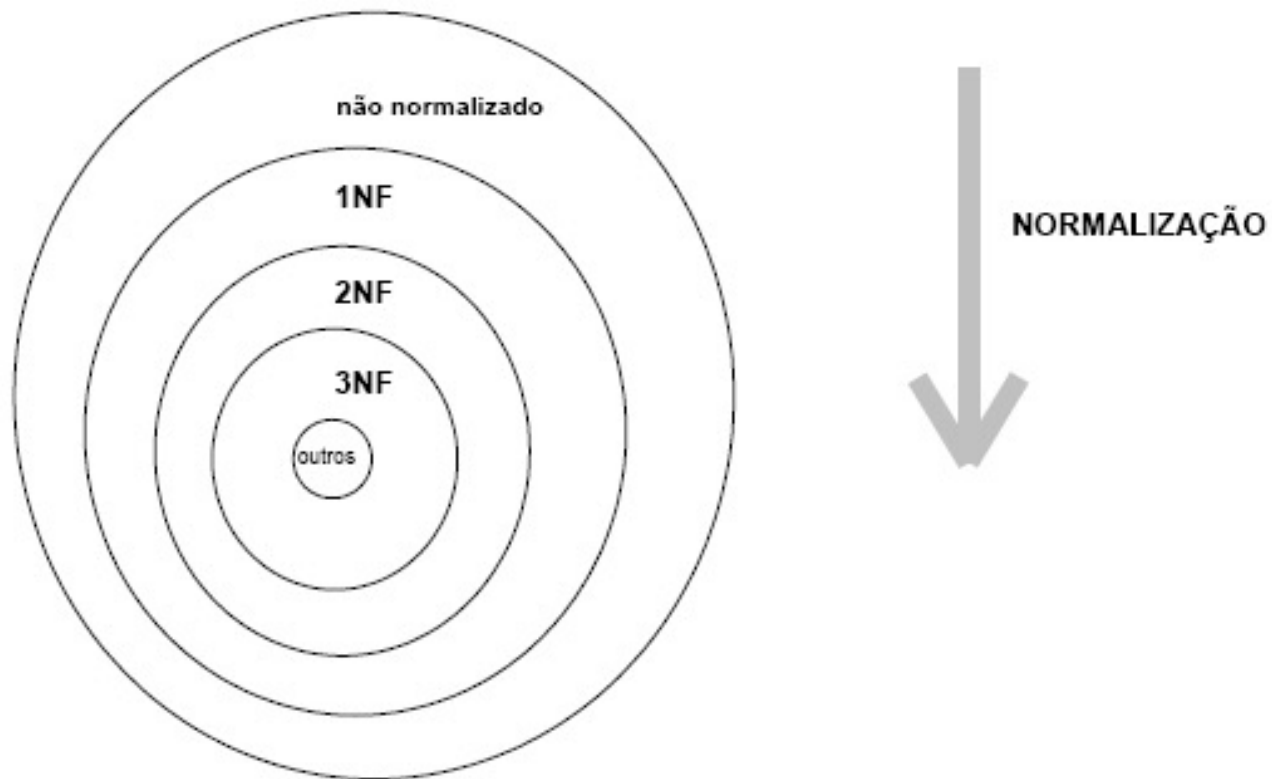
# Normalização

Há diversas formas normais, cada vez mais rígidas, para verificar tabelas relacionais:

- 1. primeira forma normal (1FN),**
- 2. segunda forma normal (2FN),**
- 3. terceira forma normal (3FN);**

# Normalização

Representação gráfica da normalização

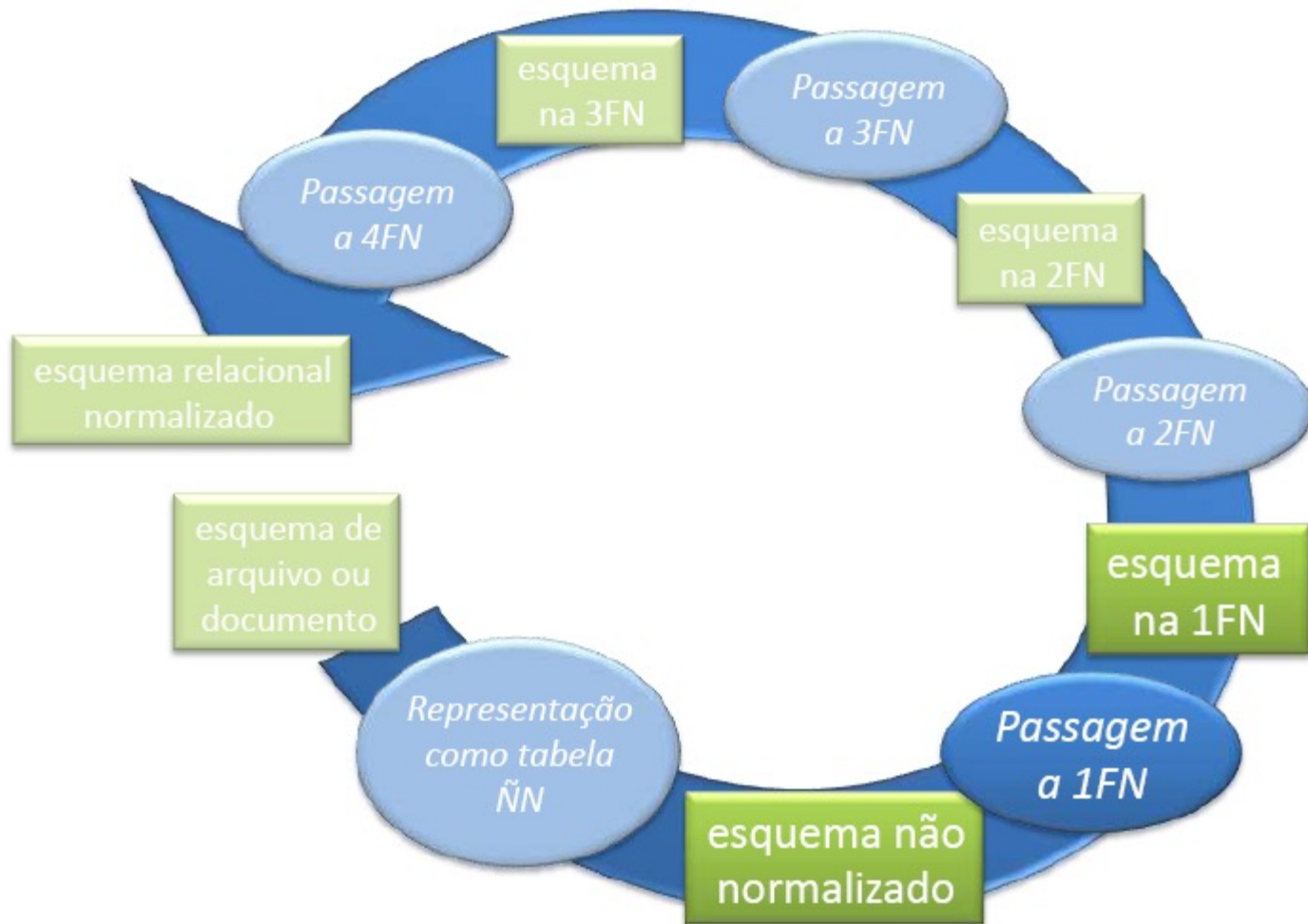


# Normalização

Conceito para cada forma normal

Tipo de normalização	Conceito básico / regra
1FN	Domínio multivalorado, atributo determinante ou chave
2FN	Dependência funcional, atributo determinante
3FN	Dependência funcional transitiva

# Normalização (1FN)



# Normalização (1FN)

## Primeira forma normal (1FN)

Uma tabela está na primeira forma normal quando ela

- **não contém tabelas aninhadas**
- **não possui colunas multi-valoradas**

(apenas colunas simples e indivisíveis)

# Normalização (1FN)

Exemplo de **1FN** utilizando uma tabela para cada tabela aninhada considerando sua chave primária, bem como as colunas de cada tabela

ÑN:

```
Proj (CodProj, Tipo, Descr,  
      (CodEmp, Nome, Cat, Sal, DataIni, TempAl)  
      )
```



1FN:

```
Proj (CodProj, Tipo, Descr)
```

```
ProjEmp (CodProj,CodEmp, Nome, Cat, Sal, DataIni, TempAl)
```

# Normalização (1FN)

Utilizando o segundo método, obtemos:

ÑN:

(CodProj, Tipo, Descr,  
(CodEmp, Nome, Cat, Sal, DataIni, TempAl))



1FN:

(CodProj, Tipo, Descr)  
(CodProj, CodEmp, Nome, Cat,  
Sal, DataIni, TempAl)

# Normalização (1FN)

Tabelas após a **1FN**

Proj:

<u>CódProj</u>	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

<u>CódProj</u>	<u>CodEmp</u>	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146	João	A1	4	1/11/91	24
LSC001	3145	Sílvio	A2	6	2/10/91	24
LSC001	6126	José	B1	9	3/10/92	18
LSC001	1214	Carlos	A2	6	4/10/92	18
LSC001	8191	Mário	A1	4	1/11/92	12
PAG02	8191	Mário	A1	4	1/05/93	12
PAG02	4112	João	A2	6	4/01/91	24
PAG02	6126	José	B1	9	1/11/92	12



# Normalização (1FN)

## Exercício 1

Esta tabela representa um arquivo que armazena informações sobre um concurso vestibular.

Apresentar a 1FN:

ÑN:

```
Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,  
                (Cod-Cand, Nome-Cand, Escore-Cand)  
                )
```

# Normalização (1FN)

## Exercício 1

ÑN:

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,  
(Cod-Cand, Nome-Cand, Escore-Cand)  
)

### 1 FN:

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato\_Curso (#Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

# Normalização (2FN e 3FN)

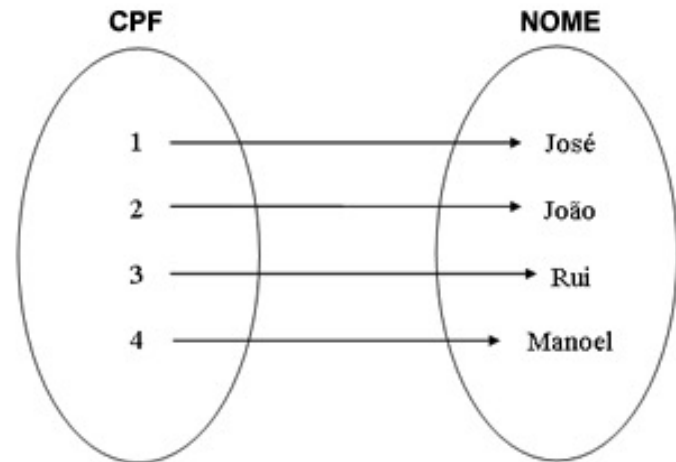
Para entender 2FN e 3FN:

É necessário compreender o conceito de ***dependência funcional***.

# Normalização (2FN e 3FN)

## Dependência funcional

Observe o conjunto:



**Existe uma dependência entre os valores dos conjuntos**, que pode ser expressa pela função  $f(\text{CPF})=\text{nome}$ . Ou seja, nome é função do CPF, ou seja, se existir um número de CPF, será encontrado o nome da pessoa correspondente. Esta dependência é expressa no Modelo Relacional da seguinte maneira:

- **CPF -> NOME**

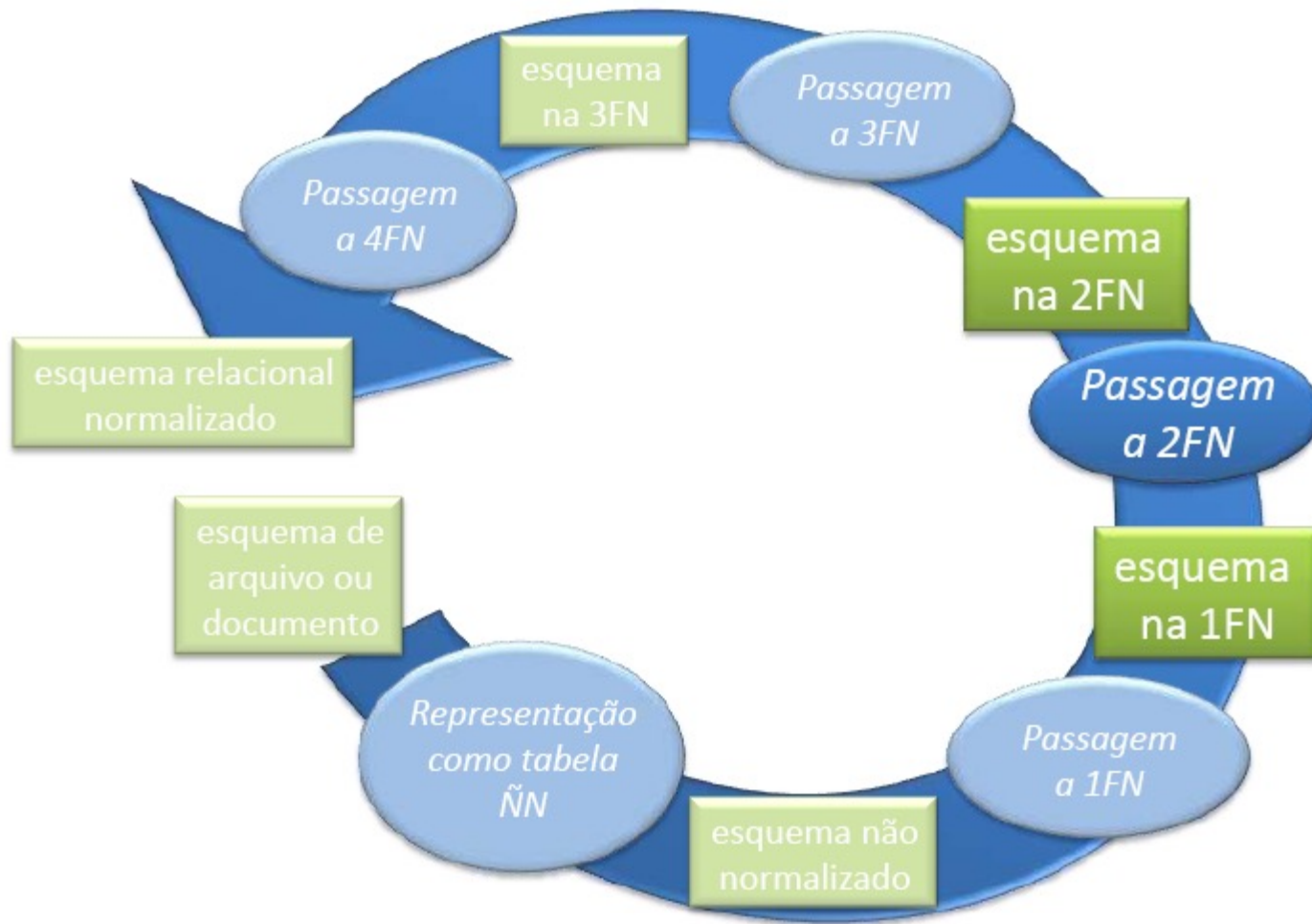
Com um número de CPF é possível encontrar o nome da pessoa, ou ainda: nome depende funcionalmente do CPF.

# Normalização (2FN e 3FN)

## Dependência funcional parcial

Uma dependência (funcional) parcial ocorre quando uma coluna depende **apenas de parte de uma chave primária composta**

# Normalização (2FN)



# Normalização (2FN)

## Segunda forma normal (2FN)

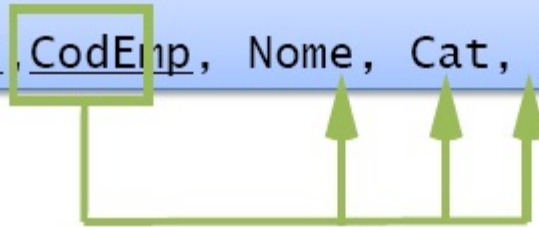
Uma tabela encontra-se na segunda forma normal, quando, além de **estar na 1FN**, **não contém dependências parciais**.

# Normalização (2FN)

## Dependência Funcional Parcial

1FN:

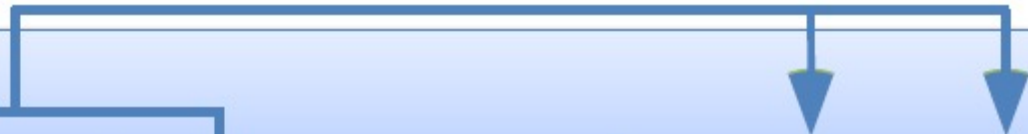
ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)



## Dependência Funcional

1FN:

ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)





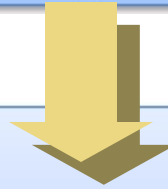
# Normalização (2FN)

## Passagem à 2FN

1FN:

Proj (CodProj, Tipo, Descr)

ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)



2FN:

Proj (CodProj, Tipo, Descr)

ProjEmp (CodProj, CodEmp, DataIni, TempAl)

Emp (CodEmp, Nome, Cat, Sal)

Tabelas com apenas **uma coluna** de chave primária ou com **todas as colunas** fazendo parte da chave primária **já estão automaticamente na 2FN**

# Normalização (2FN)

## Tabelas após a 2FN

Proj:

<u>CódProj</u>	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

<u>CódProj</u>	<u>CodEmp</u>	DataIni	TempAl
LSC001	2146	1/11/91	24
LSC001	3145	2/10/91	24
LSC001	6126	3/10/92	18
LSC001	1214	4/10/92	18
LSC001	8191	1/11/92	12
PAG02	8191	1/05/93	12
PAG02	4112	4/01/91	24
PAG02	6126	1/11/92	12

Emp:

<u>CodEmp</u>	Nome	Cat	Sal
2146	João	A1	4
3145	Sílvio	A2	6
1214	Carlos	A2	6
8191	Mário	A1	4
4112	João	A2	6
6126	José	B1	9

# Normalização (2FN)

## Exemplo 1

### 1 FN:

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

### **Apresentar a 2FN:**

# Normalização (2FN)

## Exemplo 1

### 1 FN:

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

### Apresentar a 2FN:

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, #Cod-Cand, Escore-Cand)

Candidato (Cod-Cand, Nome-Cand)

# Normalização (3FN)

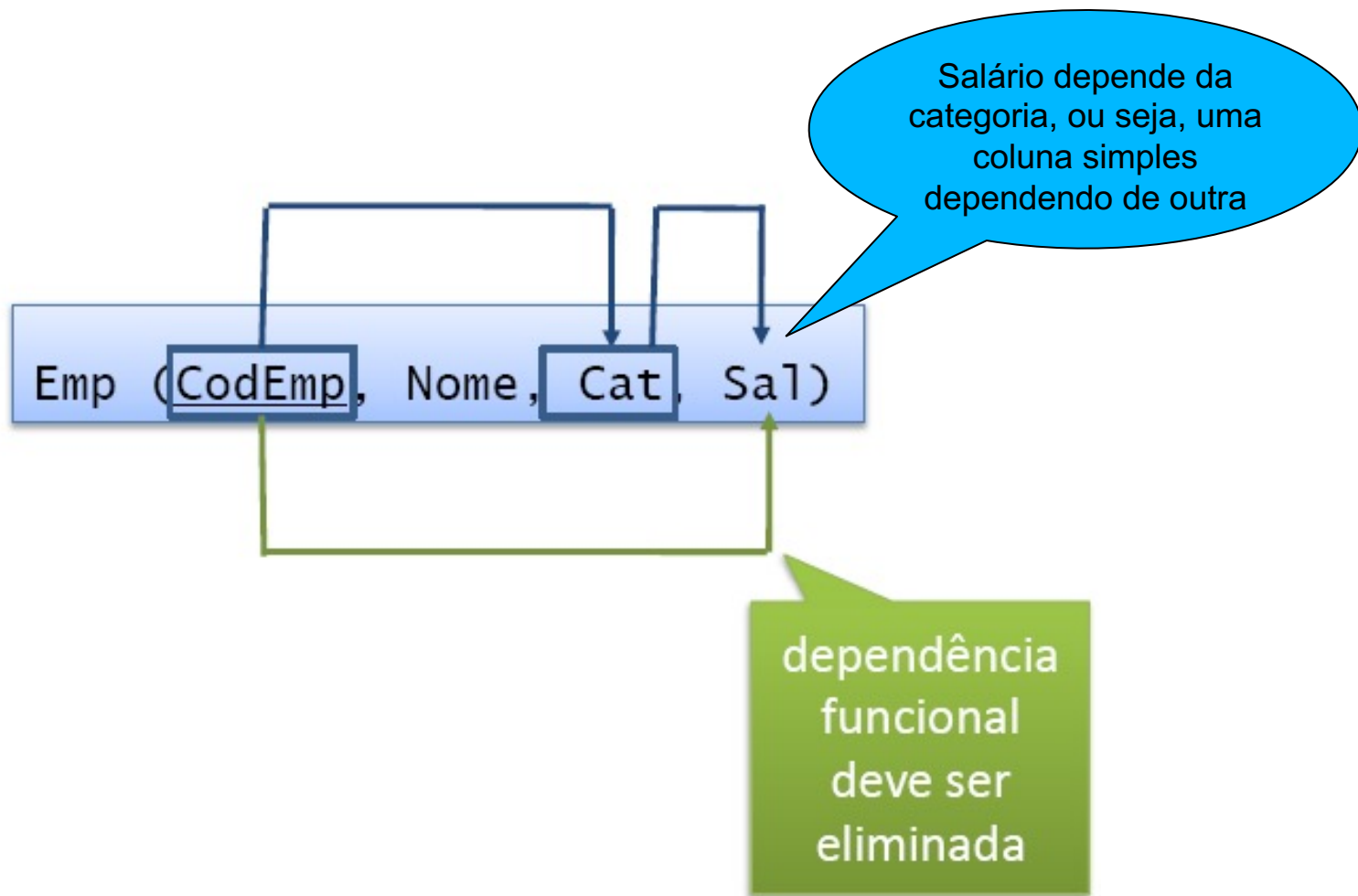


# Normalização (3FN)

## Terceira forma normal (3FN)

Uma tabela encontra-se na terceira forma normal, quando, além de **estar na 2FN**, **não contém dependências transitivas**.

# Normalização (3FN)



# Normalização (3FN)

2FN:

Emp (CodEmp, Nome, Cat, Sal)

Emp (CodEmp, Nome, Cat, Sal)

Colunas que dependem da  
chave permanecem na tabela  
original

3FN:

Emp (CodEmp, Nome, #Cat)

Colunas que dependem de  
coluna não chave vão para  
outra tabela

3FN:

Cat(Cat, Sal)



# Normalização (3FN)

2FN:

Proj (CodProj, Tipo, Descr)

ProjEmp (CodProj, CodEmp, DataIni, TempAl)

Emp (CodEmp, Nome, Cat, Sal)



3FN:

Proj (CodProj, Tipo, Descr)

ProjEmp (CodProj, CodEmp, DataIni, TempAl)

Emp (CodEmp, Nome, Cat)

Cat (Cat, Sal)

# Exemplo Normalizado

ÑN:

```
Proj (CodProj, Tipo, Descr,  
      (CodEmp, Nome, Cat, Sal, DataIni, TempAl)  
      )
```

1FN:

```
(CodProj, Tipo, Descr)  
(CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)
```

2FN:

```
Proj (CodProj, Tipo, Descr)  
ProjEmp (CodProj,CodEmp, DataIni, TempAl)  
Emp (CodEmp, Nome, Cat, Sal)
```

3FN:

```
Proj (CodProj, Tipo, Descr)  
ProjEmp (CodProj, CodEmp, DataIni, TempAl)  
Emp (CodEmp, Nome, Cat)  
Cat (Cat, Sal)
```

# Exemplo Normalizado

## Tabelas após a 3FN

Proj:

<u>CódProj</u>	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

<u>CódProj</u>	<u>CodEmp</u>	DataIni	TempAl
LSC001	2146	1/11/91	24
LSC001	3145	2/10/91	24
LSC001	6126	3/10/92	18
LSC001	1214	4/10/92	18
LSC001	8191	1/11/92	12
PAG02	8191	1/05/93	12
PAG02	4112	4/01/91	24
PAG02	6126	1/11/92	12

Emp:

<u>CodEmp</u>	Nome	Cat
2146	João	A1
3145	Sílvio	A2
1214	Carlos	A2
8191	Mário	A1
4112	João	A2
6126	José	B1

Cat:

<u>Cat</u>	Sal
A1	4
A2	6
B1	9

# Normalização (3FN)

## Exercício 1

ÑN:

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,  
(Cod-Cand, Nome-Cand, Escore-Cand)  
)

**1 FN:**

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

**2FN:**

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, #Cod-Cand, Escore-Cand)

Candidato (Cod-Cand, Nome-Cand)

**Apresentar a 3FN:**

# Normalização (3FN)

## Exercício 1

ÑN:

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,  
(Cod-Cand, Nome-Cand, Escore-Cand)  
)

**1 FN:**

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

**2FN:**

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, #Cod-Cand, Escore-Cand)

Candidato (Cod-Cand, Nome-Cand)

**3FN:**

Curso (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidato-Curso (#Cod-Curso, #Cod-Cand, Escore-Cand)

Candidato (Cod-Cand, Nome-Cand)

# Normalização

Para a maioria dos documentos e arquivos a decomposição até a **3FN é suficiente**

Na literatura, aparecem outras formas normais:

- FNBC (Forma normal de Boyce/Codd).

Utiliza a Chave candidata.

Substitui a 3FN (não abordaremos!)

- 4FN
- 5FN

# Referências Bibliográficas

- Material originalmente elaborado por Prof. Gilberto Irajá Müller. Material autorizado e cedido pelo autor. Revisado e atualizado por Prof. João Tavares.
- HEUSER, Carlos Alberto. **Projeto de Banco de Dados**. 6. ed. Bookman Companhia Ed, 2009.
- SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Sistemas de Banco de Dados**. 3<sup>a</sup>. Ed. São Paulo: Makron Books, 2010.