Departamento de Ciência da Computação

ciência da computação

Universidade Federal de Juiz de Fora

# DCC060 – Banco de Dados

MATERIAL DE APOIO

## Normalização de Banco de Dados Relacionais

PROF. TARCÍSIO DE SOUZA LIMA

- Tabelas de BDs relacionais, derivadas dos modelos ER ou UML, às vezes são afetados por anomalias de atualização e integridade e por problemas de desempenho.
- Quando o BD como um todo é definido como uma única tabela, isso pode resultar em uma grande quantidade de dados redundantes e pesquisas demoradas.
- Pode resultar também em atualizações longas e dispendiosas e as exclusões em particular podem provocar a eliminação de dados úteis com um efeito colateral indesejado.
- Exemplo:

  BD Vendas (em uma única tabela) – próximo slide -->

## Vendas

| nome-produto | num-pedido | nome-cliente | end-cliente | crédito | data | nome-vendedor |
|---|---|---|---|---|---|---|
| aspirador de pó | 1458 | David Barreto | Andrelândia | 6 | 03/01/03 | Carlos |
| computador | 2730 | Solange Silva | Petrópolis | 10 | 15/04/05 | Tássio |
| geladeira | 2460 | Michel Souza | Anápolis | 8 | 12/09/04 | Daniel |
| DVD player | 519 | Pedro Ramalho | Divinópolis | 3 | 05/12/04 | Fred |
| rádio | 1986 | Carlos Antonelli | Campos | 7 | 10/05/05 | Rui |
| CD player | 1817 | Carmem Reis | Muriaé | 8 | 03/08/02 | Paulo |
| aspirador de pó | 1865 | Carlos Antonelli | Campos | 7 | 01/10/04 | Carlos |
| aspirador de pó | 1885 | Beth Correia | Divinópolis | 8 | 19/04/99 | Carlos |
| geladeira | 1943 | David Barreto | Andrelândia | 6 | 04/01/04 | Daniel |
| televisão | 2315 | Sara Pimentel | Parati | 6 | 15/03/04 | Fred |

- Produtos, vendedores, clientes e pedidos são todos armazenados na tabela Vendas.
- Informações de produto e cliente são armazenados de forma redundante, desperdiçando espaço de armazenamento.
- Consultas como “quais clientes pediram aspirador de pó no mês passado?” exigiria uma pesquisa na tabela inteira.
- A atualização de um endereço, p.ex., o de David Barreto, exigiria a alteração de múltiplas linhas.
- A exclusão de um pedido de um cliente, p. ex., o de Solange Silva (que comprou um computador caro), se esse for o seu único pedido pendente, excluirá a única cópia do seu endereço e da sua avaliação de crédito como um efeito colateral, sendo difícil (ou impossível) recuperar esta informação.

- Objetivo: tornar o BD seria mais eficiente e confiável.
- Solução: desmembrar a tabela em tabela menores para que esses tipos de problemas sejam eliminados.
- Classes de esquemas de BDs relacionais ou definições de tabelas, chamadas de formas normais, normalmente são usadas para realizar esse objetivo.
- A criação de tabelas de BDs em uma dada forma normal é chamada de normalização. A normalização é feita analisando-se as interdependências entre atributos individuais associados a essas tabelas e tomando-se projeções (subconjuntos de colunas) de tabelas maiores para formar tabelas menores.
- Dependendo do nível de refinamento em que uma tabela se encontre, ela estará numa determinada forma normal.
- Existem regras, baseadas nos conceitos de dependência funcional, que determinam as características necessárias a uma tabela para que ela esteja numa determinada forma normal.

- **Superchave:** é um conjunto de um ou mais atributos que, quando tomados coletivamente, nos permite identificar exclusivamente uma ocorrência de uma entidade ou linha de uma tabela.
- **Chave candidata:** qualquer subconjunto de atributos de uma superchave que também seja uma superchave e que não possa ser reduzido a outras superchaves.
- **Chave primária:** é selecionada arbitrariamente, a partir do conjunto de chaves candidatas, para que ela seja usada como índice dessa tabela; quando constituída
- **Chave Estrangeira:** campos de uma tabela que sejam chave primária numa outra tabela qualquer.

**Exemplos:** Suponha as tabelas

**Emp = {Cód-Depto, Matrícula-Depto, Nome, Salário, Cargo}**

**Depto = {Cód-Depto, NomeDepto, Ramais}**

- na tabela **Depto** são chaves candidatas os campos: **Cód-Depto** (chave primária escolhida) e **NomeDepto**;
- na tabela **Emp**, a composição dos campos **Cód-Depto** e **Matrícula-Depto** forma a sua chave primária (chave primária composta);
- o campo **Cód-Depto** na tabela **Emp** é um exemplo de chave estrangeira.
- na tabela **Depto** são superchaves: (**Cód-Depto**), (**Cód-Depto**, **NomeDepto**) e (**Cód-Depto**, **NomeDepto**, **Ramais**), que é a superchave trivial (isto é, todos os campos da tabela)

**Exemplos:** Suponha as tabelas definidas nos exemplos de chaves

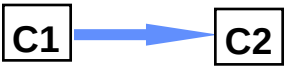
- na tabela **Depto** não podem haver registros cujo código do departamento seja nulo (integridade de entidade);
- na tabela **Emp**:
  - o campo **Matrícula-Depto** não pode se nulo (integridade de entidade);
  - o campo **Cód-Depto**:
    a) por ser chave estrangeira, poderia assumir o valor nulo ou o de um departamento qualquer, desde que este esteja cadastrado na tabela **Depto**;
    b) por ser um componente da chave primária, não pode assumir o valor nulo.

- **Integridade de Entidade:**
  - Nenhum campo componente de uma chave primária deve poder assumir um valor nulo;
  - Decorre do fato de que todas as entidades que se quer representar devem ser distingüíveis entre si.
- **Integridade Referencial:**
  - Um campo que seja chave estrangeira só pode assumir:
    - valor nulo;
    - valor para o qual exista uma tupla na tabela onde ela é chave primária.
  - Útil na manutenção da integridade dos dados tanto na sua inclusão como na sua modificação e exclusão.

***Observação:*** Dependendo do SGBD com que se está trabalhando, deve-se verificar que regras de integridade e em que nível elas podem ser por ele implementadas, ficando sob a responsabilidade dos programas a sua implementação complementar.

- Dada uma relação R diz-se que o atributo C2 de R é **funcionalmente dependente** do atributo C1 de R, sse cada valor de C1 em R, tem associado a ele precisamente, e a qualquer instante, um único valor de C2 em R

Esquematicamente:

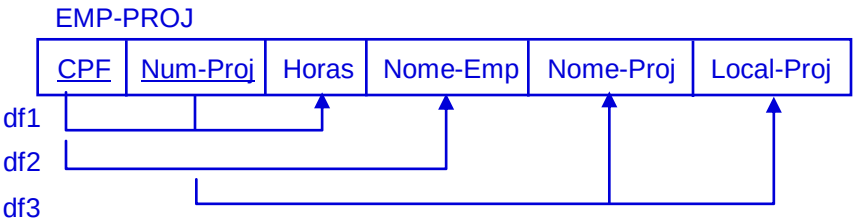



- Deve ser derivada a partir do conhecimento conceitual que se tem sobre o mundo real que se quer representar.
- O exame dos dados pode sugerir, mas não garantir, dependências funcionais em potencial.

**Exemplo:** Cód-Depto → NomeDepto

**X → Y**

Os valores dos atributos do conjunto X determinam os valores dos atributos do conjunto Y ou, inversamente, os valores dos atributos do conjunto Y dependem funcionalmente dos valores dos atributos do conjunto X.




df1 : {CPF, Num-Proj} → {Horas}

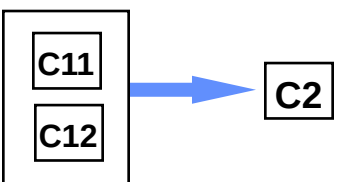
df2 : {CPF} → {Nome-Emp}

df3 : {Num-Proj} → {Nome-Proj, Local-Proj}

DF deve ser explicitamente definida por alguém que conheça a ***semântica*** dos atributos de uma relação. Não basta deduzir a DF pelas instâncias.

- O conceito de dependência funcional pode ser estendido para o caso em que C1, C2 ou ambos são domínios compostos.
- Sendo C1 uma chave primária composta pelos atributos C11 e C12 diz-se que C2 é **completamente funcionalmente dependente** de C1 sse, a cada valor de C1 (e não a parte deles!), está associado um único valor de C2.

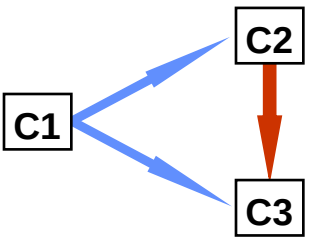
Esquematicamente:




**Exemplo:** (Cód-Depto, Matrícula-Depto) → Nome, Salário, Cargo

- Dados os atributos C1, C2 e C3 de uma relação, sendo C1 sua chave primária, diz-se que C2 e C3 são **mutuamente dependentes** sse forem funcionalmente dependentes de C1, além de existir uma dependência funcional entre eles.

Esquematicamente:




***Obs.:*** os atributos são *mutuamente independentes* se nenhum deles for funcionalmente dependente do outro

**Exemplo:** Supondo uma estrutura de salários padronizados por cargo

$$\begin{cases} \text{(Cód-Depto, Matrícula-Depto)} \longrightarrow \text{Nome, Salário, Cargo} \\ \text{Cargo} \longrightarrow \text{Salário} \end{cases}$$

Uma relação (um arquivo) está na **1FN** sse todos os domínios básicos contiverem somente valores atômicos (simples, indivisíveis) e todos os atributos que não pertencem à chave primária são funcionalmente dependentes dela.

Assim sendo:

- existe uma chave primária
- não existem itens de grupo
- não existem atributos (campos) multivalorados
- relação (arquivo) tem representação bidimensional

Para solucionar o problema da multivaloração devem ser geradas múltiplas tuplas (registros) nas quais:

- cada tupla (registro) contém um único valor do atributo (campo) multivalorado;
- todos os outros campos devem ser copiados da tupla (registro) original

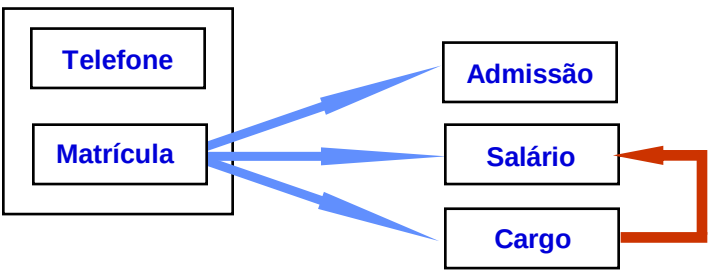
Relação **Empregado** (não normalizada)

| Matrícula | Salário | Data Admissão | Cargo | Telefones |
|---|---|---|---|---|
| 1807 | 13.017,00 | 12/10/93 | Eng2 | 264-1092, 257-3196 |
| 2264 | 5.912,50 | 01/03/96 | Prg1 | 291-1267 |
| 2462 | 5.912,50 | 20/07/92 | Prg1 | 551-6322, 245-6734 |

Relação **Empregado** (em **1FN**)

| Matrícula | Salário | Data Admissão | Cargo | Telefone |
|---|---|---|---|---|
| 1807 | 13.017,00 | 12/10/93 | Eng2 | 264-1092 |
| 1807 | 13.017,00 | 12/10/93 | Eng2 | 257-3196 |
| 2264 | 5.912,50 | 01/03/96 | Prg1 | 291-1267 |
| 2462 | 5.912,50 | 20/07/92 | Prg1 | 551-6322 |
| 2462 | 5.912,50 | 20/07/92 | Prg1 | 245-6734 |

Esquema da Relação **Empregado** (em **1FN**):




Porém:

- Existe **dependência funcional não completa** e
- Existe **dependência funcional mútua**

que precisam ser resolvidas (seguindo o processo da normalização)

Relação **Fornecedor** (não normalizada)

| Código | Nome | Escritórios | Telefone | Bancos |
|---|---|---|---|---|
| 1632 | Cia ABC Ltda. | Rio de Janeiro | 265-2133 | BCMT, BNAA |
| | | São Paulo | 597-1642 | |
| | | Belém | 227-6331 | |
| 1512 | XYZ Ind e Com | Belo Horizonte | 395-1672 | BRR |
| | | Recife | 222-1227 | |
| 2364 | FFF S/A | Curitiba | 262-1394 | BMTG, BNAA |

- Passar para a **1FN** significaria “aplainar” a relação (arquivo) acima
- Isto só é possível fazendo-se, para cada tupla (registro) da relação (arquivo) não normalizada, todas as combinações entre escritórios/telefones e bancos

Relação **Fornecedor** (em **1FN**)

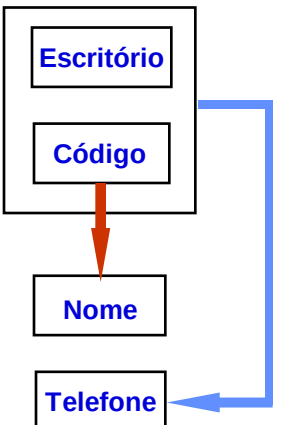
| Código | Nome | Escritório | Telefone | Banco |
|---|---|---|---|---|
| 1632 | Cia ABC Ltda. | Rio de Janeiro | 265-2133 | BCMT |
| 1632 | Cia ABC Ltda. | São Paulo | 597-1642 | BCMT |
| 1632 | Cia ABC Ltda. | Belém | 227-6331 | BCMT |
| 1632 | Cia ABC Ltda. | Rio de Janeiro | 265-2133 | BNAA |
| 1632 | Cia ABC Ltda. | São Paulo | 597-1642 | BNAA |
| 1632 | Cia ABC Ltda. | Belém | 227-6331 | BNAA |
| 1512 | XYZ Ind e Com | Belo Horizonte | 395-1672 | BRR |
| 1512 | XYZ Ind e Com | Recife | 222-1227 | BRR |
| 2364 | FFF S/A | Curitiba | 262-1394 | BMTG |
| 2364 | FFF S/A | Curitiba | 262-1394 | BNAA |

Relação **Fornecedor**

- O resultado em **1FN** é uma relação (arquivo) onde cada tupla (registro) representa cada Banco, para cada Escritório, e Telefone do fornecedor.
- Considerando-se que cada fornecedor tem apenas um escritório com um telefone por cidade, a chave primária da relação (arquivo) normalizada fica sendo: **Código, Escritório, Banco**
- Existem uma série de anomalias na atualização de uma relação (arquivo) como a proposta
- Estas anomalias decorrem do fato de se ter combinado os múltiplos valores de Banco com os múltiplos valores de Escritório (com seu Telefone)
- A observação de que *Bancos* e *Escritórios* são independentes uns dos outros sugere que a relação (arquivo) seja desmembrada em duas outras, também em **1FN**.

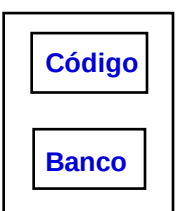
Relação **Escritório** (em **1FN**)

| Código | Nome | Escritório | Telefone |
|---|---|---|---|
| 1632 | Cia ABC Ltda. | Rio de Janeiro | 265-2133 |
| 1632 | Cia ABC Ltda. | São Paulo | 597-1642 |
| 1632 | Cia ABC Ltda. | Belém | 227-6331 |
| 1512 | XYZ Ind e Com | Belo Horizonte | 395-1672 |
| 1512 | XYZ Ind e Com | Recife | 222-1227 |
| 2364 | FFF S/A | Curitiba | 262-1394 |




Relação **Banco**

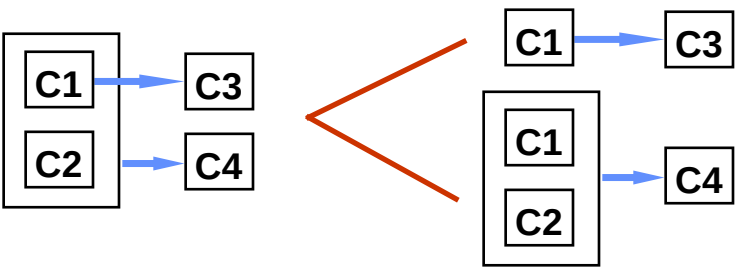
| Código | Banco |
|---|---|
| 1632 | BCMT |
| 1632 | BNAA |
| 1512 | BRR |
| 2364 | BMTG |
| 2364 | BNAA |



Uma relação (um arquivo) está na **2FN** sse, além de estar na **1FN**, todo atributo não-primo, isto é, não seja membro da chave, for completamente funcionalmente dependente dela, isto é, for totalmente dependente da chave primária.

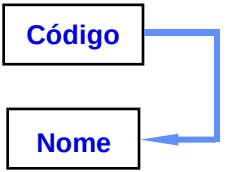
Assim sendo:

- atributos (campos) que não sejam dependentes exatamente da mesma chave devem ser separados em diferentes relações (arquivos)



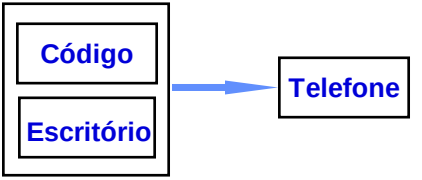
Relação **Fornecedor**

| Código | Nome |
|---|---|
| 1632 | Cia ABC Ltda. |
| 1512 | XYZ Ind e Com |
| 2364 | FFFS/A |




Relação **Escritório** (em **2FN**)

| Código | Escritório | Telefone |
|---|---|---|
| 1632 | Rio de Janeiro | 265-2133 |
| 1632 | São Paulo | 597-1642 |
| 1632 | Belém | 227-6331 |
| 1512 | Belo Horizonte | 395-1672 |
| 1512 | Recife | 222-1227 |
| 2364 | Curitiba | 262-1394 |

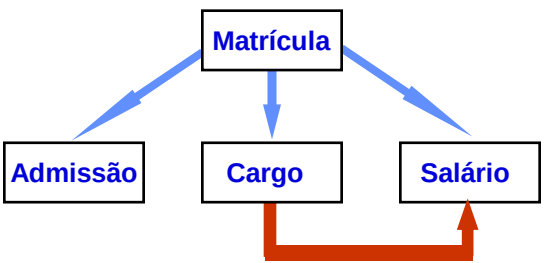


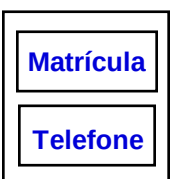
Relação **Empregado** (em **2FN**)

| Matrícula | Salário | Data Admissão | Cargo |
|---|---|---|---|
| 1807 | 13.017,00 | 12/10/93 | Eng2 |
| 2264 | 5.912,50 | 01/03/96 | Prg1 |
| 2462 | 5.912,50 | 20/07/92 | Prg1 |

Relação **Telefone**

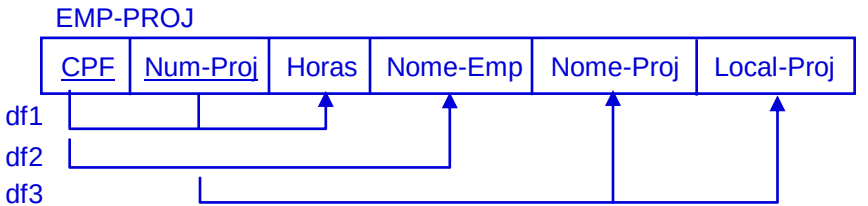
| Matrícula | Telefone |
|---|---|
| 1807 | 264-1092 |
| 1807 | 257-3196 |
| 2264 | 291-1267 |
| 2462 | 551-6322 |
| 2462 | 245-6734 |

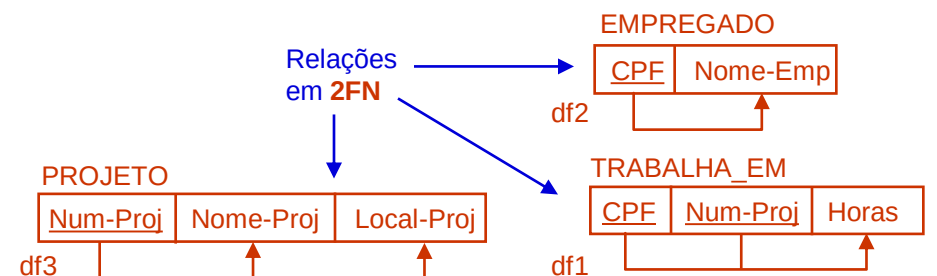



***Obs.:*** *Dependência mútua ainda precisa ser eliminada*

**Outro exemplo:** relação em **1FN** que não está em **2FN**



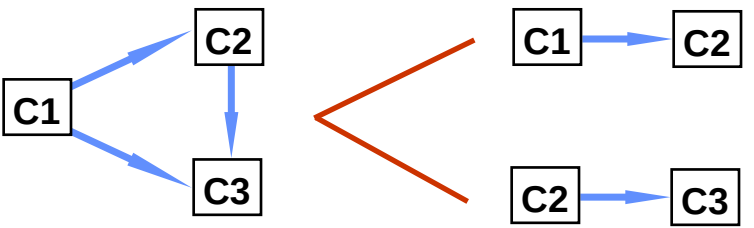
Uma relação (um arquivo) está na **3FN** sse, além de estar na **2FN**, nenhum atributo não-primo for transitivamente dependente da chave primária, isto é, forem mutuamente independentes.

Em outras palavras, para toda DF $X \rightarrow A$, uma das duas condições seguintes devem valer:

- X é uma superchave ou
- A é membro de uma chave candidata

Assim sendo:

- atributos (campos) que estejam envolvidos numa dependência funcional mútua devem ser transferidos para uma outra relação (arquivo)



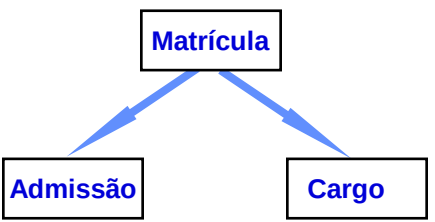
## Exemplo

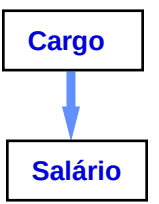
Relação **Empregado** (em **3FN**)

| Matrícula | Data Admissão | Cargo |
|---|---|---|
| 1807 | 12/10/93 | Eng2 |
| 2264 | 01/03/96 | Prg1 |
| 2462 | 20/07/92 | Prg1 |

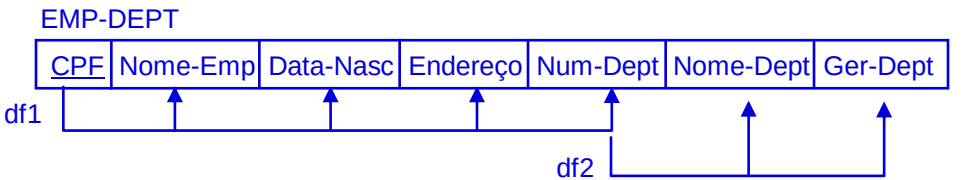
Relação **Cargo**

| Cargo | Salário |
|---|---|
| Eng2 | 13.017,00 |
| Prg1 | 5.912,50 |



**Outro exemplo:** relação em **2FN** que não está em **3FN**




Relações em **3FN**


- Otimiza desempenho da atualização
- Ajuda a diminuir dúvidas de projeto pois, por vezes, com o objetivo de melhorar o desempenho das funções de acesso, o projetista toma decisões que comprometem as funções de atualização. Seguindo estritamente o processo de normalização, isso nunca aconteceria.
- Grupa dados de tal forma que em uma relação cada dado é um dado dependente da chave, de toda a chave e nada mais que a chave principal da relação.
- É um meio formal de se estruturar a informação de tal forma que fique claro exatamente que tipos de dados existem e que dependências funcionais eles satisfazem.

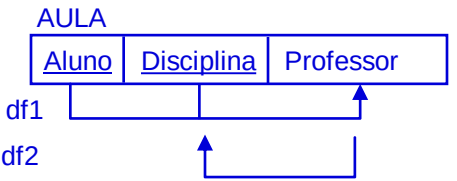
***Observação:***

*O processo de normalização segue com outras formas (**FNBC – Forma Normal Boyce-Codd**, além da **4FN** e da **5FN**). Entretanto, a normalização até 3FN atende a maioria das situações práticas.*

Uma tabela **R** está na *Forma Normal de Boyce-Codd (FNBC)* se, para cada DF não trivial X → A, X for uma superchave de **R**.

- É uma forma mais restritiva de **3FN**, isto é toda relação em **FNBC** está também em **3FN**; entretanto, uma relação em **3FN** não está necessariamente em **FNBC**.

**Exemplo:** Relação em **3FN** que não está em **FNBC**




df2 : {Professor} → {Disciplina}
{Professor} não é uma superchave.

AULA

| Aluno | Disciplina | Professor |
|---|---|---|
| Carlos | Química | Ana |
| Carlos | Física | Antonio |
| Marta | Química | Ana |
| Marta | Português | Maria |
| João | Português | Manoel |

Anomalia de exclusão: Se Carlos sair da aula de Física, não teremos nenhum registro de que Antonio leciona Física.

Relações em **FNBC**

R1

| Aluno | Professor |
|---|---|
| Carlos | Ana |
| Carlos | Antonio |
| Marta | Ana |
| Marta | Maria |
| João | Manoel |

R2

| Professor | Disciplina |
|---|---|
| Ana | Química |
| Antonio | Física |
| Maria | Português |
| Manoel | Português |

Note, entretanto, que a dependência funcional

$$df1 : \{\text{Aluno, Disciplina}\} \rightarrow \{\text{Professor}\}$$

foi perdida na decomposição.

Considere, por exemplo, a inserção de (Marta, Manoel) em R1.

## Decomposição/Junção sem perda

- Conclusão: alguns esquemas não podem ser normalizados em **FNBC** e ao mesmo tempo preservar todas as dfs.
- Solução: redundância parcial.

AULA

| Aluno | Disciplina | Professor |
|---|---|---|
| Carlos | Química | Ana |
| Carlos | Física | Antonio |
| Marta | Química | Ana |
| Marta | Português | Maria |
| João | Português | Manoel |

R2

| Professor | Disciplina |
|---|---|
| Ana | Química |
| Antonio | Física |
| Maria | Português |
| Manoel | Português |

Relações em **FNBC**, com redundância parcial
e todas as dependências funcionais preservadas

- *Baseada em dependências multivaloradas* (DMVs)

Uma relação **R** está em **4FN** se e somente se estiver em **FNBC** e, caso exista alguma **DMV**

X --->> Y,

a **DMV** é trivial (i.e., $Y \subset X$ ou $X \cup Y = R$)

ou

X é uma superchave de **R**.

EMPREGADO

| Nome-Emp | Nome-Proj | Nome-Depend |
|---|---|---|
| Prado | X | João |
| Prado | Y | Ana |
| Prado | X | Ana |
| Prado | Y | João |
| Borba | W | José |
| Borba | X | José |
| Borba | Y | José |
| Borba | Z | José |
| Borba | W | Joana |
| Borba | X | Joana |
| Borba | Y | Joana |
| Borba | Z | Joana |
| Borba | W | Beto |
| Borba | X | Beto |
| Borba | Y | Beto |
| Borba | Z | Beto |

Relação EMPREGADO não está em **4FN**.
2 DMVs não triviais:
Nome-Emp --->> Nome-Proj
Nome-Emp --->> Nome-Depend

EMP-PROJ

| Nome-Emp | Nome-Proj |
|---|---|
| Prado | X |
| Prado | Y |
| Borba | W |
| Borba | X |
| Borba | Y |
| Borba | Z |

EMP-DEP

| Nome-Emp | Nome-Depend |
|---|---|
| Prado | João |
| Prado | Ana |
| Borba | José |
| Borba | Joana |
| Borba | Beto |

As relações EMP-PROJ e EMP-DEP estão em **4FN**.

- Tipos adicionais de dependências:
  - dependências de junção e de inclusão, que levam a formas normais mais restritas (**Quinta Forma Normal**, **Forma Normal de Domínio-Chave**).
- A utilidade prática destas formas normais é limitada, porque num banco de dados real com muitos atributos, é muito difícil (e praticamente irrelevante) descobrir tais dependências e restrições.

## Normalização como Ferramenta para Validação da Qualidade de um Esquema

- As formas normais até **FNBC** são baseadas em dependências funcionais, exceto a **1FN**, que faz parte da definição do modelo relacional.
- O design conceitual baseado nos modelos ER ou OO tende naturalmente a produzir esquemas normalizados, a menos da **1FN**.
- A separação de conceitos é o resultado natural do design conceitual bem feito.
- Na prática, esquemas que violam a normalização são exemplos de esquemas mal projetados.