

MC536



Normalização

Normalização

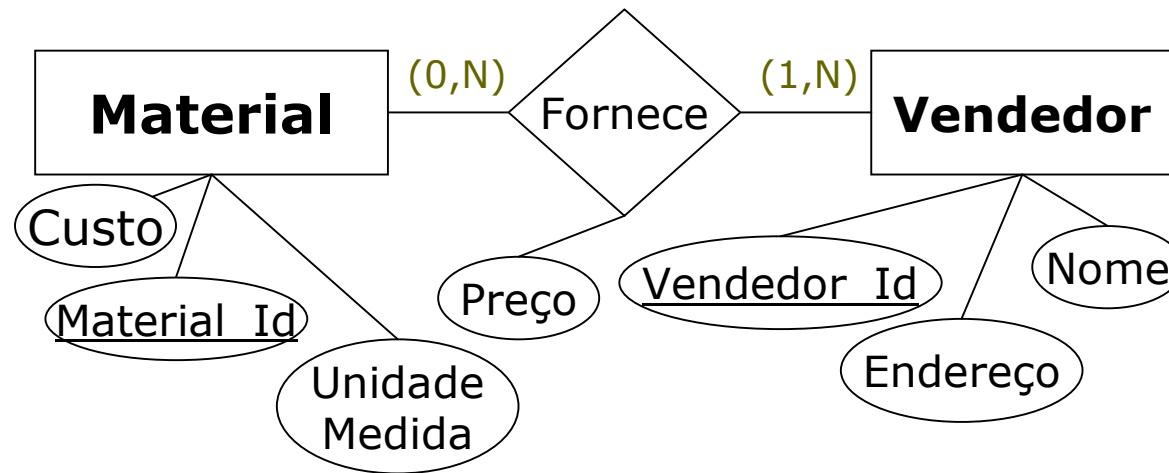
- Processo que decompõe
 - esquemas de relações com anomalias
 - em
 - esquemas de relações menores que satisfaçam certas restrições

- Motivação
 - melhora o projeto lógico evitando duplicação desnecessária de dados
 - produz relações menores mais bem-estruturadas

Dependência Funcional

- É uma restrição entre dois atributos ou dois conjuntos de atributos
- Advém do conhecimento da semântica dos dados armazenados
- Para qualquer relação **R**,
 - um atributo **B** é funcionalmente dependente de um atributo **A** se, para toda instância de **R**, um valor de **A** determina univocamente o valor de **B**
 - ou
 - **A** → **B**
 - **{A,B}** → **{C,D}** ≈ **AB** → **CD**
- Determinante: atributo do lado esquerdo da flecha

Dependência Funcional



Material

<u>Material Id</u>	Custo	Unidade_de_Medida
--------------------	-------	-------------------

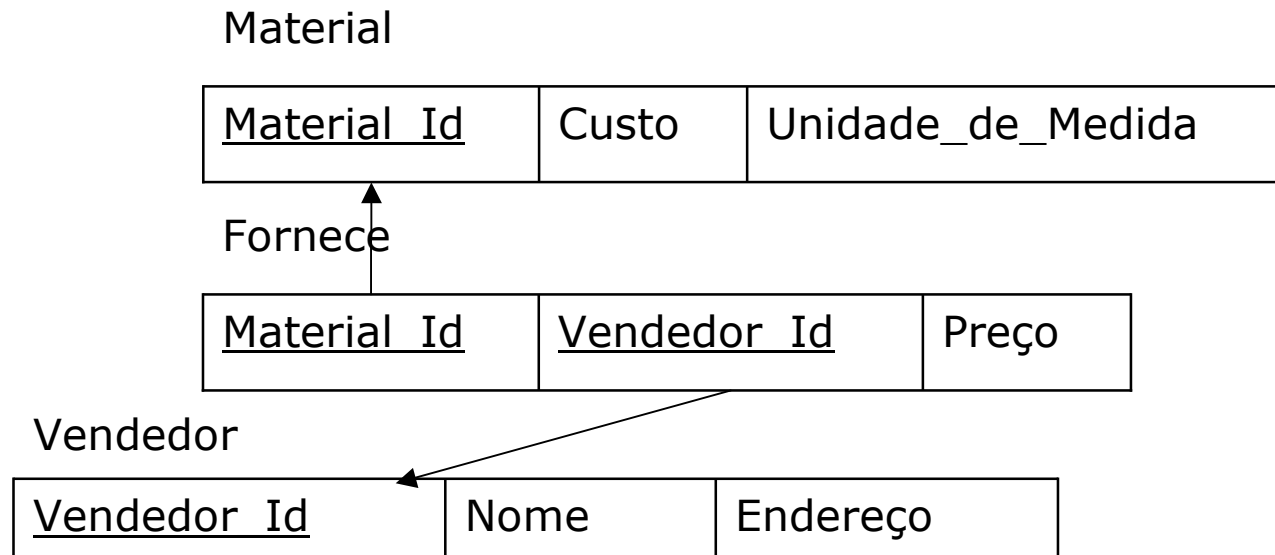
Fornece

<u>Material Id</u>	<u>Vendedor Id</u>	Preço
--------------------	--------------------	-------

Vendedor

<u>Vendedor Id</u>	Nome	Endereço
--------------------	------	----------

Dependência Funcional



- ❑ $\text{Material_Id} \rightarrow \{\text{Custo}, \text{Unidade_de_Medida}\}$
- ❑ $\text{Vendedor_Id} \rightarrow \{\text{Nome}, \text{Endereço}\}$
- ❑ $\{\text{Material_Id}, \text{Vendedor_Id}\} \rightarrow \text{Preço}$
- ❑ $\{\text{Vendedor_Id}, \text{Nome}\} \rightarrow \text{Endereço}$

Regras de Inferência de Armstrong

- **A1 Reflexividade.** Se X contém Y então $X \rightarrow Y$
Uma df obtida por este axioma é chamada de df trivial
- **A2 Aumentativa.** Se $X \rightarrow Y$ então para qualquer Z , $XZ \rightarrow YZ$
- **A3 Transitividade.** Se $X \rightarrow Y$ e $Y \rightarrow Z$ então $X \rightarrow Z$
- $\{A1, A2, A3\}$
 - Sólido (*sound*): só levam a deduções de dependências funcionais verdadeiras
 - Completo: dado um conjunto de dfs F válidas para uma relação R , permitem obter todas as dfs válidas para R que podem ser inferidas a partir de F

Regras de Inferência de Armstrong

- **A4 União.** Se $X \rightarrow Y$ e $X \rightarrow Z$, então $X \rightarrow YZ$
 - $X \rightarrow XY$ por A2; como $X \rightarrow Z$, $XY \rightarrow ZY$ por A2 então $X \rightarrow ZY$ por A3
 - (Note que $ZY=YZ$)
- **A5 Decomposição.** Se $X \rightarrow Y$ e Y contém Z então $X \rightarrow Z$
 - $Y \rightarrow Z$ por A1 e como $X \rightarrow Y$, então $X \rightarrow Z$ por A3
 - Se $X \rightarrow A_1, A_2, \dots, A_n$ então $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$
 - Sem perda de generalidade pode-se supor que todas as dfs são do tipo $X \rightarrow A_i$, onde A_i é um atributo simples
 - Forma canônica
- **A6 Pseudo-transitividade.** Se $X \rightarrow Y$, $WY \rightarrow Z$, então $WX \rightarrow Z$

Fecho (Closure)

- ❑ O **Fecho** de um subconjunto F de dfs é o conjunto F^+ de todas as dfs que podem ser inferidas a partir de F
- ❑ O **Fecho** de um conjunto de atributos X com relação a F é o conjunto X^+ de todos os atributos que são determinados funcionalmente por X
- ❑ X^+ pode ser calculado aplicando os axiomas A1, A2, A3 repetidamente usando as dfs em F

Fecho (Closure)

- Fecho de X : X^+
- **Algoritmo:**
 - $X^+ := X$
 - Repita
 - $\text{old}X^+ := X^+$
 - Para toda DF $Y \rightarrow Z$ em F
 - Se X^+ contém Y então
 - $X^+ := X^+ \cup Z$
 - Até que ($\text{old}X^+ = X^+$)

Fecho (Closure)

- $F = \{RG \rightarrow \text{Nome_Func},$
 $\text{Num_Proj} \rightarrow \{\text{Nome_Proj}, \text{Loc_Proj}\},$
 $\{RG, \text{Num_Proj}\} \rightarrow \text{Horas}\}$
 - $\{RG\}^+ = \{RG, \text{Nome_Func}\}$
 - $\{\text{Num_Proj}\}^+ = \{\text{Num_Proj}, \text{Nome_Proj},$
 $\text{Loc_Proj}\}$
 - $\{RG, \text{Num_Proj}\}^+ = \{RG, \text{Nome_func},$
 $\text{Num_Proj}, \text{Nome_Proj}, \text{Loc_Proj}, \text{Horas}\}$

Fecho (Closure): usos

- Determinar se uma df $X \rightarrow Y$ é uma df válida.
 - Calcular X^+ . Se X^+ **contém** Y , então $X \rightarrow Y$ é uma df válida.
- Determinar se um dado subconjunto W é chave de $R=(A_1, A_2, \dots, A_n)$
 - Calcular W^+ e verificar se $W^+ = \{A_1, A_2, \dots, A_n\}$. Se isto ocorre, $W \rightarrow A_1, W \rightarrow A_2, \dots, W \rightarrow A_n$ e, portanto, W é superchave de R

Dependência Funcional: chave

- Atributo, ou combinação de atributos que identifica univocamente uma linha de uma relação
 - **Identificação única:** todo atributo não-chave é dependente funcionalmente da chave
 - **Não-redundância:** nenhum atributo na chave pode ser eliminado sem destruir a propriedade de identificação única

Dependência Funcional: chave

- A_i, A_j, \dots, A_k é uma chave de uma relação R se e somente se:
 - A_i, A_j, \dots, A_k determina todos os outros atributos de R
 - Nenhum subconjunto de A_i, A_j, \dots, A_k determina funcionalmente os atributos restantes de R
- **Atributo não-primo (ANP)** é um atributo que não está contido em nenhuma chave candidata de R. Todo atributo que pertence a uma chave candidata é denominado **primo**
- Uma **superchave** de R é um subconjunto dos atributos de R que contém uma chave
 - Toda chave é uma superchave, mas nem toda superchave é chave

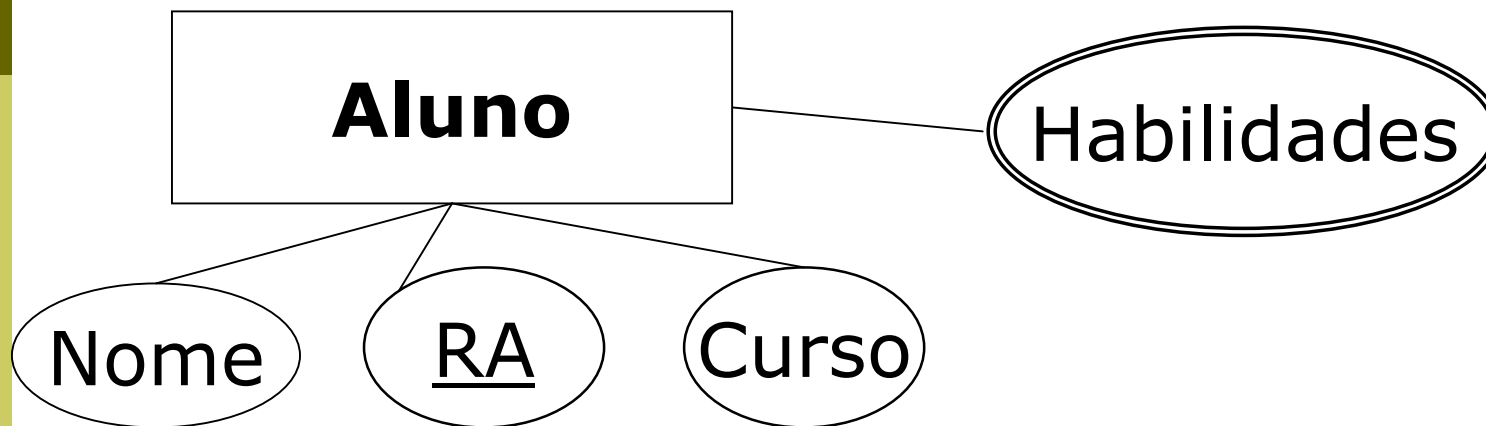
Forma Normal

- ❑ Estado de uma relação que resulta da aplicação de regras simples com respeito a dependências funcionais envolvendo atributos desta relação
- ❑ Conjunto de testes para “certificar” se uma dada relação pertence a uma forma normal
- ❑ Uma relação está em uma forma normal quando ela satisfaz um número de características desejáveis

Primeira Forma Normal (1FN)

- Uma relação não pode conter atributo multivalorado nem composto
 - O domínio dos atributos deve incluir valores atômicos
 - O valor de qualquer atributo deve ser um único valor do domínio daquele atributo

Primeira Forma Normal (1FN)



Aluno

<u>RA</u>	Nome	Curso	Habilidade
000001	João	Letras	{Tupi}
000002	Maria	Matemática	{Inglês, Espanhol}
963127	Ricardo	Computação	{Inglês, Francês}

Primeira Forma Normal (1FN)

□ Com redundância

Aluno

<u>RA</u>	Nome	Curso	Habilidade
000001	João	Letras	Tupi
000002	Maria	Matemática	Inglês
000002	Maria	Matemática	Espanhol
963127	Ricardo	Computação	Inglês
963127	Ricardo	Computação	Francês

Primeira Forma Normal (1FN)

Aluno

<u>RA</u>	Nome	Curso
000001	João	Letras
000002	Maria	Matemática
963127	Ricardo	Computação

Habilidades_Aluno

<u>RA</u>	<u>Habilidade</u>
000001	Tupi
000002	Inglês
000002	Espanhol
963127	Inglês
963127	Francês

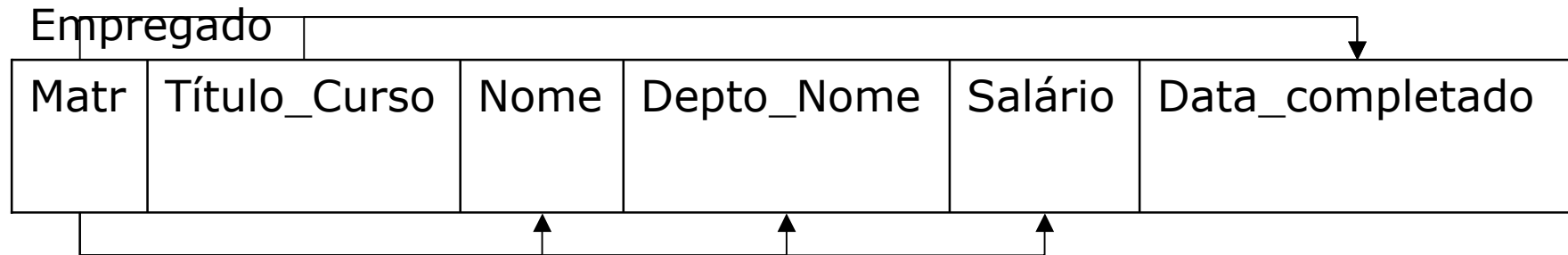
Segunda Forma Normal (2FN)

□ Dependência Parcial

- Uma dependência funcional na qual um ou mais atributos não-chave são dependentes de parte da chave primária
- Um atributo A é dependente funcional parcial de um atributo X , se $X \rightarrow A$ e existe um subconjunto próprio Y de X tal que $Y \rightarrow A$
 - Chama-se **subconjunto próprio** de um conjunto A , todo conjunto B , tal que B está contido em A e $B \neq A$

Segunda Forma Normal (2FN)

▣ Dependência Parcial



Segunda Forma Normal (2FN)

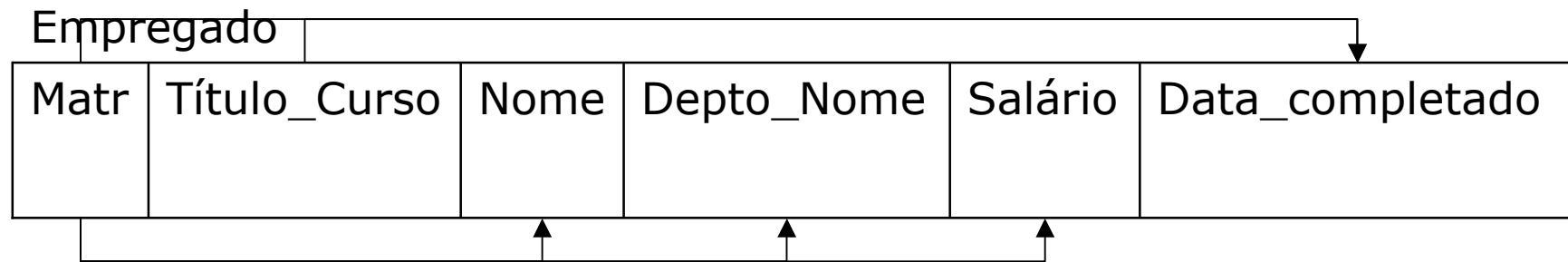
- Uma relação R está em 2FN se está em 1FN e todo atributo não-chave é totalmente dependente funcionalmente da chave primária
- Uma relação R está em 2FN se nenhum ANP de R é dependente funcional parcial de qualquer chave de R

Segunda Forma Normal (2FN)

- Uma relação em 1FN está em 2FN:
 - Se a chave primária consiste de apenas um atributo
 - Nenhum atributo não-chave existe na relação (todos os atributos na relação são componentes da chave primária)
 - Todo atributo não-chave é dependente funcionalmente de todo o conjunto de atributos da chave primária

Segunda Forma Normal (2FN)

□ Dependência Parcial



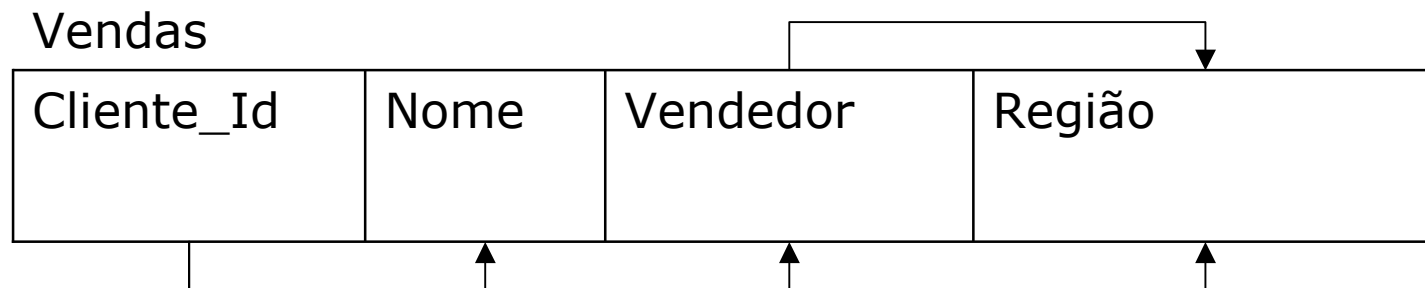
□ Solução: decompor em duas relações

- Empregado(Matr, Nome, Depto_Nome, Salário)
- EmpCurso(Matr, Título_Curso, Data_Completado)

Terceira Forma Normal (3FN)

□ Dependência Transitiva

- Dependência entre dois ou mais atributos não-chaves
- Dadas as dfs $X \rightarrow Y$, $Y \rightarrow X$, e $Y \rightarrow Z$, dizemos que Z depende transitivamente de X



Terceira Forma Normal (3FN)

Vendas

Cliente_Id	Nome	Vendedor	Região
8023	Anderson	Sandro	Sul
9167	Bruno	Humberto	Oeste
7929	Horácio	Sandro	Sul
6837	Tom	João	Leste
8596	Emerson	Humberto	Oeste
7018	Arnaldo	Fabricio	Norte

Terceira Forma Normal (3FN)

□ Possíveis Anomalias

■ Inserção

- Um novo vendedor não pode ser inserido a menos que um cliente tenha sido alocado para um vendedor

■ Deleção

- Quando um cliente é deletado (6837) da tabela, perde-se a informação de que João está alocado para a região Leste

■ Atualização

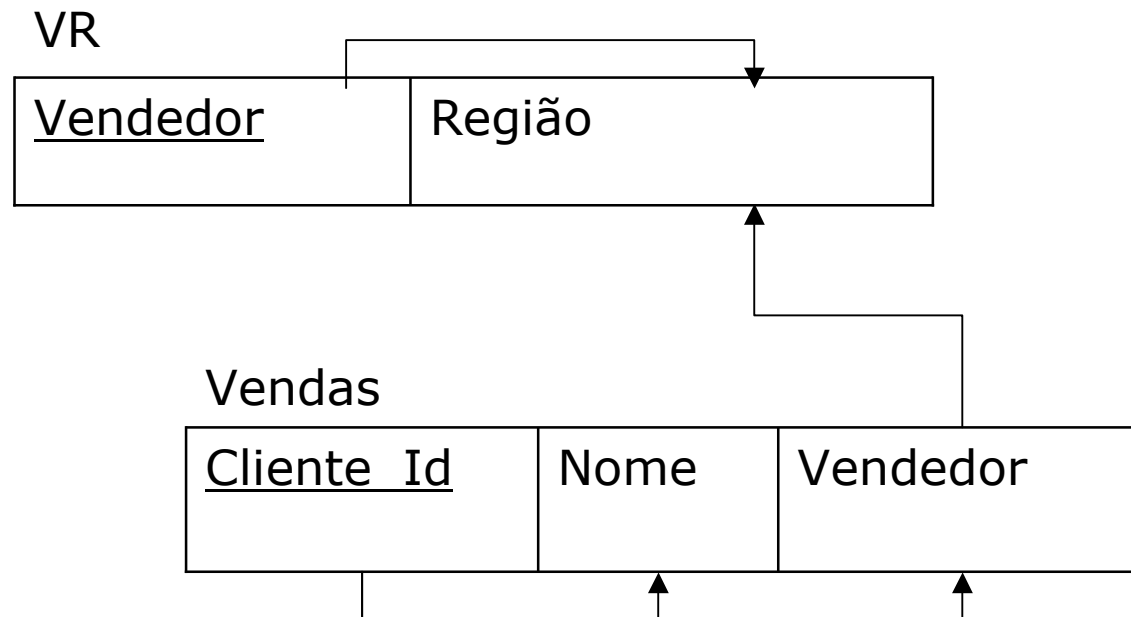
- Se um vendedor muda de região, várias linhas deverão ser modificadas para refletir este fato

Vendas

Cliente_Id	Nome	Vendedor	Região
8023	Anderson	Sandro	Sul
9167	Bruno	Humberto	Oeste
7929	Horácio	Sandro	Sul
6837	Tom	João	Leste
8596	Emerson	Humberto	Oeste
7018	Arnaldo	Fabricio	Norte

Terceira Forma Normal (3FN)

- Solução: decompor a relação em duas
 - Economiza espaço: região não repetida para cada cliente



Terceira Forma Normal (3FN)

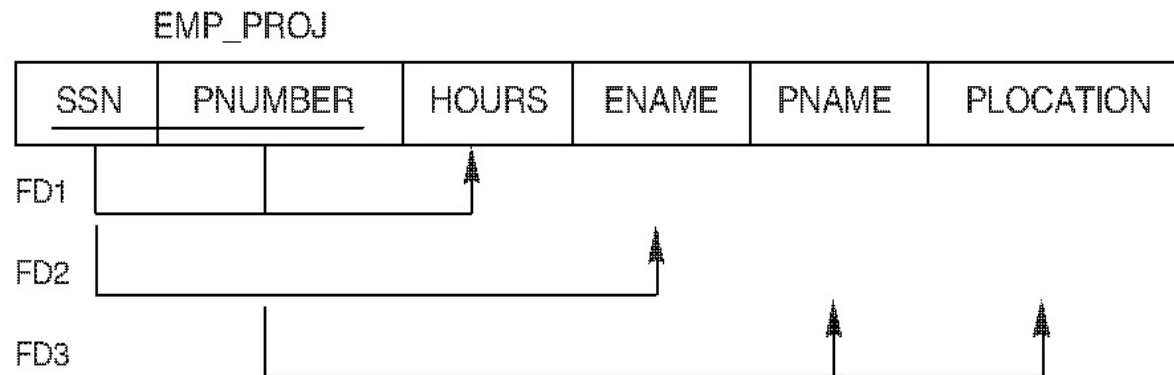
- Uma relação em 2FN que não tem nenhuma dependência transitiva
- Uma relação R está em 3FN se nenhum ANP de R depende transitivamente de qualquer chave de R

Terceira Forma Normal (3FN)

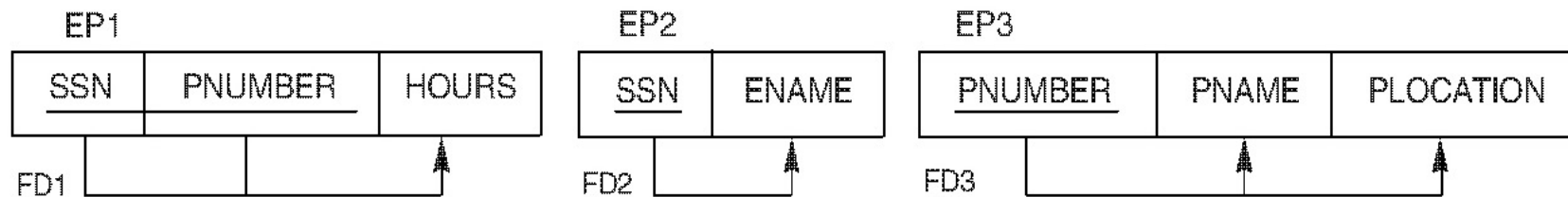
- Uma relação R está em 3FN se, para toda df não trivial $X \rightarrow Y$, válida para R, então
 - (a) X é uma superchave ou
 - (b) Y é atributo primo
- Se uma relação R viola 3FN, tanto a condição (a) quanto a condição (b) é violada
 - Violar (b): Y é um atributo não-primo
 - Violar (a): X não é um superconjunto de nenhuma chave de R
 - X poderia ser não-primo
 - Viola 3FN (transitividade)
 - X poderia ser subconjunto de uma chave de R
 - Viola 2FN e 3FN: dependência parcial da chave

Exemplos (1)

(a)

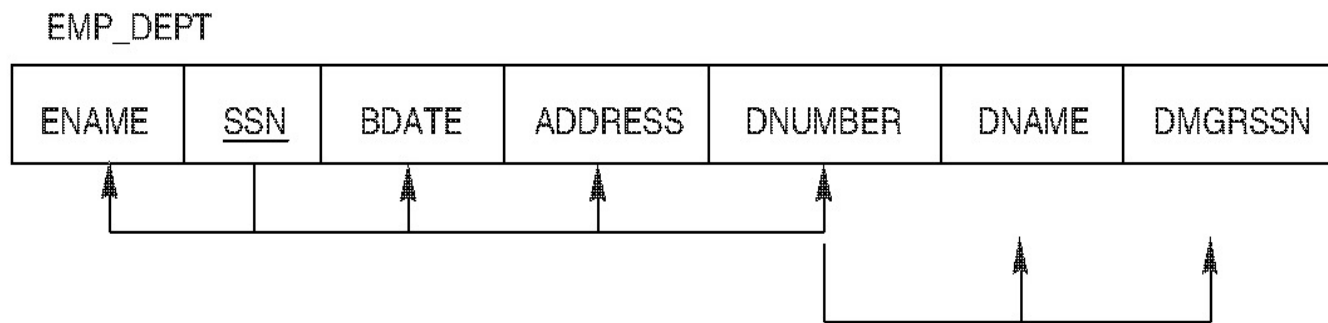


2NF NORMALIZATION

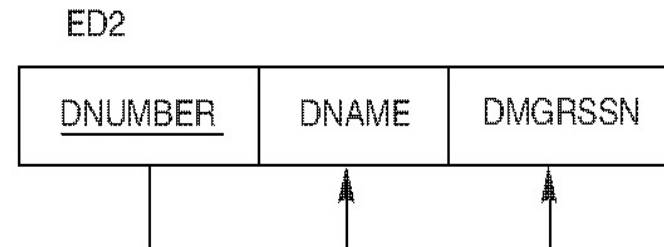
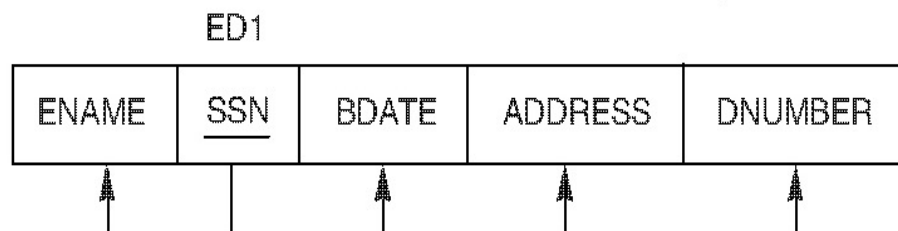


Exemplos (1)

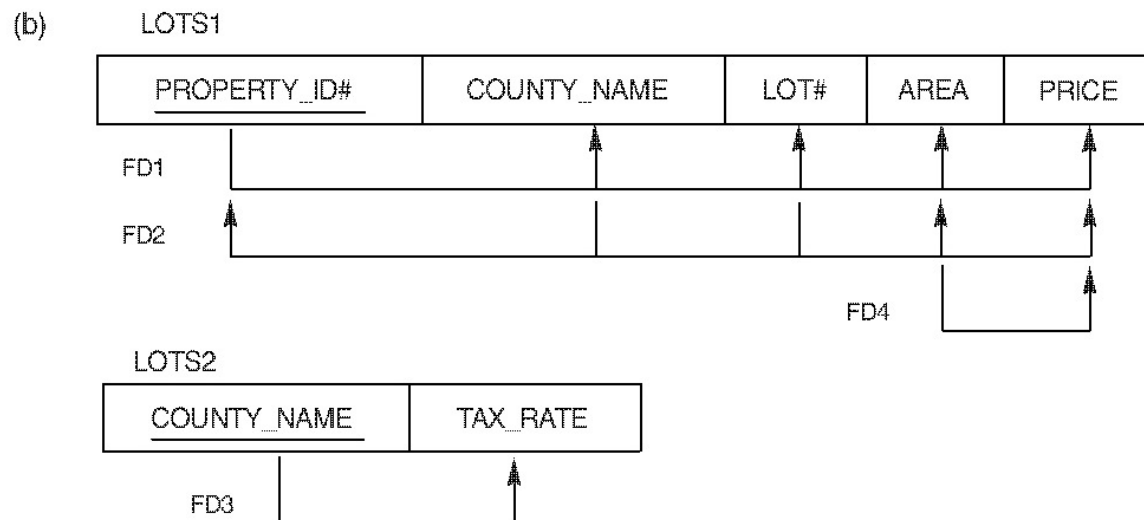
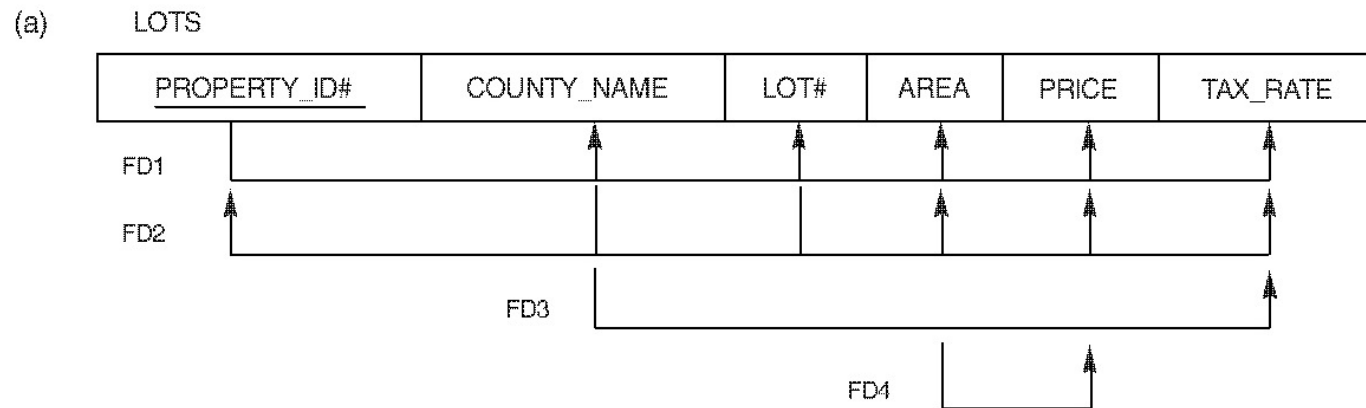
(b)



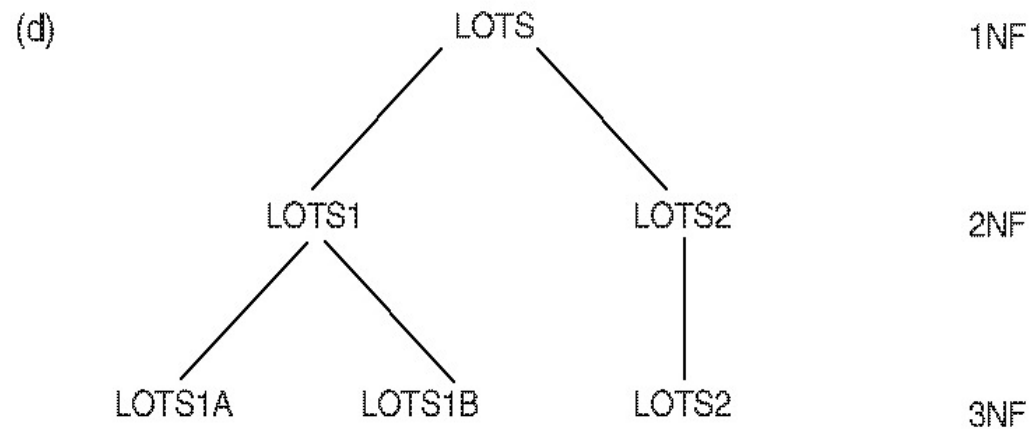
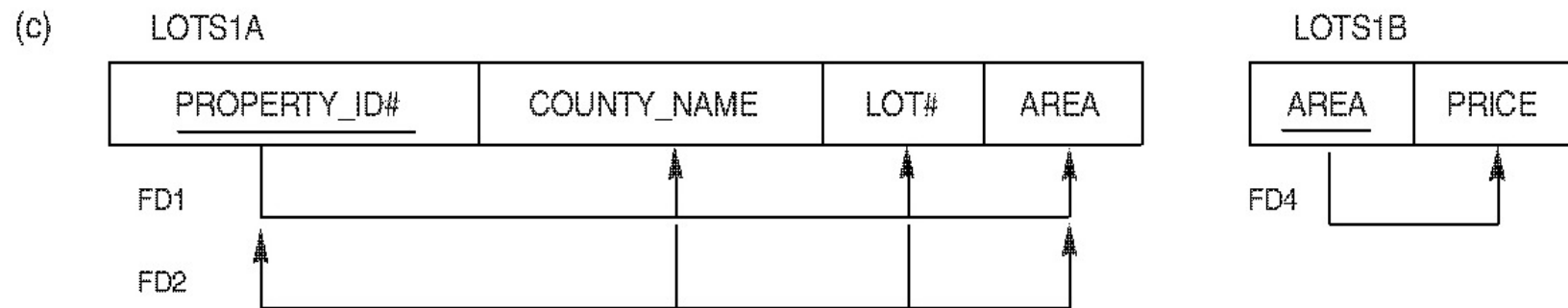
3NF NORMALIZATION



Exemplos (2)



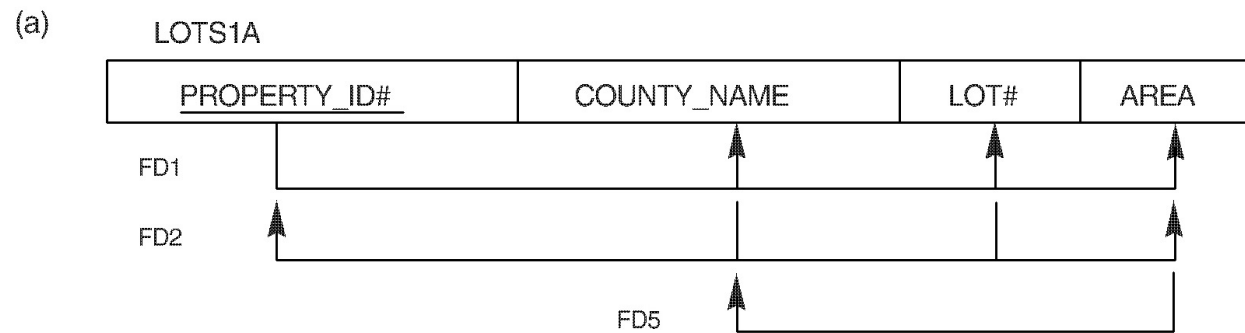
Exemplos (2)



Forma Normal de Boyce-Codd (FNBC)

- Uma relação R está em FNBC se, para toda fd não trivial $X \rightarrow Y$, válida para R , X é uma superchave de R

Forma Normal de Boyce-Codd (FNBC)



BCNF Normalization

