



Fundamentos de Banco de Dados

Normalização





- Objetivo
 - Apresentar uma abordagem de projeto de banco de dados, denominada de Normalização, a qual permite analisar a qualidade das relações, bem como elevar a sua qualidade.
- Principais tópicos
 - Anomalias
 - Tuplas espúrias
 - Abordagens de Projeto de Banco de Dados
 - Dependências Funcionais
 - Regras de Inferência para DFs





- Principais tópicos (*Continuação*)
 - Formas Normais com base em Chaves Primárias
 - Definição Geral de Formas Normais
 - BCNF (Boyce-Codd Normal Form)
 - Dependências Multivaloradas
 - Quarta Forma Normal (4FN)





- Top-down
 - Iniciar com o agrupamento dos atributos obtidos a partir do projeto conceitual de mapeamento
 - Isso é chamado de projeto por análise
- Bottom-up
 - Considerar os relacionamentos entre atributos
 - Construir as relações
 - Isso é chamado projeto pela síntese
- Nossa Abordagem
 - Utilizar a abordagem Top-down para obter as relações
 - Utilizar a abordagem Bottom-up para melhorar a qualidade das relações obtidas anteriormente

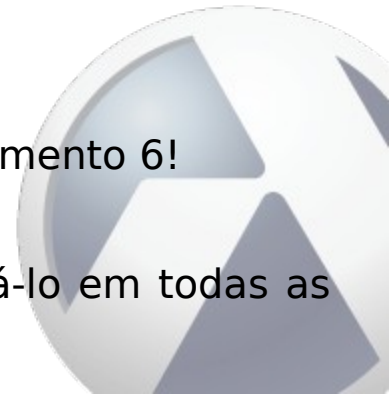


- Cuidado com redundância de informação

EMP_DEP

<u>NSS</u>	NOME	DTANIV	DNUMERO	DNOME	GERENTE
21	AA	-	5	CV	91
22	BB	-	5	CV	91
23	CC	-	6	TS	93
24	DD	-	7	OS	94
25	EE	-	7	OS	94

- Anomalias de Inserção:
 - Como inserir novo departamento sem que exista empregados?
 - Inserir empregados é difícil quando informações de departamento devem ser inseridas corretamente.
- Anomalias de Remoção:
 - O que acontece quando removemos CC? Perdemos o departamento 6!
- Anomalias de Alteração:
 - Se mudarmos o gerente do departamento 5, devemos mudá-lo em todas as tuplas com DNUMERO = 5.





Tuplas Espúrias

- Não quebre uma relação em relações que possam gerar tuplas espúrias

DNUMERO	NOME	PNOME	PLOCALIZAÇÃO
123	XX	Compras	São Paulo
123	XX	Vendas	Rio de Janeiro
124	YY	Logística	São Paulo

- A relação pode ser quebrada em

DNUMERO	NOME	PNOME	DNUMERO	PLOCALIZAÇÃO
123	XX	Compras	123	São Paulo
123	XX	Vendas	123	Rio de Janeiro
124	YY	Logística	124	São Paulo

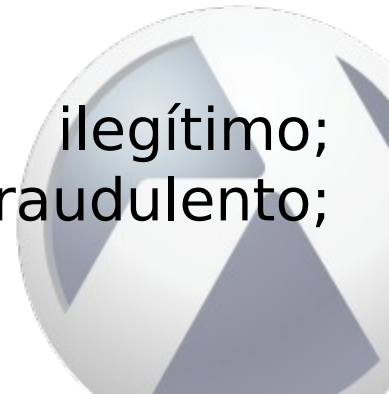
- Quando fazemos o Join, obtemos NOVAS TUPLAS!

<u>DNUMERO</u>	NOME	PNOME	PLOCALIZAÇÃO
123	XX	Compras	São Paulo
123	XX	Compras	Rio de Janeiro
123	XX	Vendas	São Paulo
123	XX	Vendas	Rio de Janeiro
124	YY	Logística	São Paulo

Após o Join, o resultado não foi a relação original. Assim, houve perda de informações. Conclui-se que houve uma decomposição com perdas.



- Projetos incorretos de BDRs podem gerar resultados inválidos em certas operações Join
- A propriedade de “junção sem perdas” é usada para garantir resultados corretos em operações Join
- As relações devem ser projetadas para satisfazer a condição de junção sem perdas. Nenhuma tupla espúria deve ser gerada ao fazer um join natural de qualquer relação
 - **Espúrio:** Falso; suposto; adulterado; ilegítimo; bastardo; ilegal; desonesto; ilícito; fraudulento; viciado; impuro; incorreto; ilídimo.





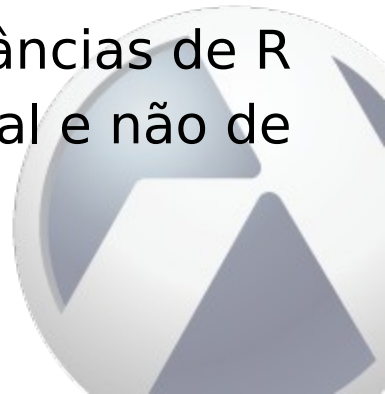
Dependências Funcionais

- Dependências funcionais (DFs) são usadas para medir formalmente a qualidade do projeto relacional
- As DFs e chaves são usadas para definir formas normais de relações
- As DFs são restrições que são derivadas do significado dos atributos e do seus inter-relacionamentos
- Um conjunto de atributos X determina funcionalmente um conjunto de atributos Y se o valor de X determinar um único valor Y
 - $X \rightarrow Y$



Dependências Funcionais

- $X \rightarrow Y$
 - Se duas tuplas tiverem o mesmo valor para X , elas devem ter o mesmo valor para Y . Ou seja:
 - Se $X \rightarrow Y$ então, para quaisquer tuplas $t1$ e $t2$ de $r(R)$:
 - Se $t1[X] = t2[X]$, então $t1[Y] = t2[Y]$
- Se K é uma chave de R , então K determina funcionalmente todos os atributos de R
 - Isso porque, nunca teremos duas tuplas distintas com $t1[K] = t2[K]$
- Importante
 - $X \rightarrow Y$ especifica uma restrição sobre todas as instâncias de R
 - As DFs são derivadas das restrições do mundo real e não de uma extensão específica da relação R





Exemplos de Restrições de DF

- O número do seguro social determina o nome do empregado
 - NSS → ENOME
- O número do projeto determina o nome do projeto e a sua localização
 - PNUMERO → { PNOME, PLOCALIZACAO }
- O nss de empregado e o número do projeto determinam as horas semanais que o empregado trabalha no projeto
 - { NSS, PNUMERO } → HORAS





Regras de Inferência para DFs

- Regras de inferência de Armstrong:
 - RI1. (Reflexiva) Se $Y \subseteq X$ (é subconjunto de), então $X \rightarrow Y$
 - (Isso também é válido quando $X=Y$)
 - RI2. (Aumentativa) Se $X \rightarrow Y$, então $XZ \rightarrow YZ$
 - (Notação: XZ significa $X \cup Z$)
 - RI3. (Transitiva) Se $X \rightarrow Y$ e $Y \rightarrow Z$, então $X \rightarrow Z$
- RI1, RI2 e RI3 formam um conjunto completo de regras de inferência





Regras de Inferência para DFs

- Algumas regras de inferência úteis:
 - (Decomposição) Se $X \rightarrow YZ$, então $X \rightarrow Y$ e $X \rightarrow Z$
 - (Aditiva) Se $X \rightarrow Y$ e $X \rightarrow Z$, então $X \rightarrow YZ$
 - (Pseudotransitiva) Se $X \rightarrow Y$ e $WY \rightarrow Z$, então $WX \rightarrow Z$
- As três regras de inferência acima, bem como quaisquer outras regras de inferência, podem ser deduzidas a partir de RI1, RI2 e RI3 (propriedade de ser completa)





Formas Normais



Formas Normais com base em Chaves Primárias

- Normalização de Relações
- Uso prático de Formas Normais
- Definições de Chaves e de Atributos que participam de Chaves
- Primeira Forma Normal
- Segunda Forma Normal
- Terceira Forma Normal



- Normalização
 - Processo de decompor relações “ruins” dividindo seus atributos em relações menores e “melhores”
- Forma Normal
 - Indica o nível de qualidade de uma relação
- 1FN
 - Definição de relação. Atributos atômicos (indivisíveis).
- 2FN, 3FN, BCNF
 - Baseiam-se em chaves e DFs de uma relação esquema
- 4FN e 5FN
 - Baseiam-se em chaves e dependências multivaloradas





Uso Prático das Formas Normais

- Na prática, a normalização é realizada para obter projetos de alta qualidade
- Os projetistas de bancos de dados não precisam normalizar na maior forma normal possível.
- Desnormalização
 - Processo de armazenar junções de relações de forma normal superior como uma relação base que está numa forma normal inferior

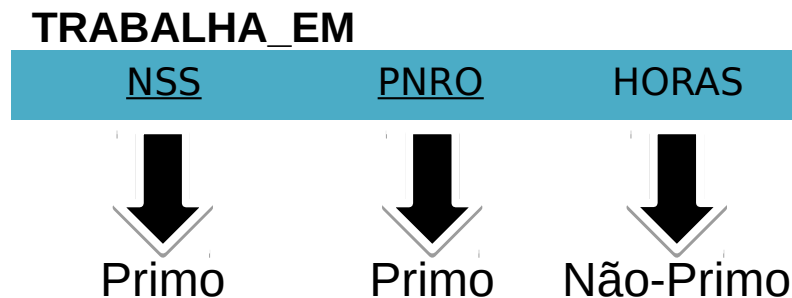




- Uma superchave, S , de uma relação esquema
 - $R = \{A_1, A_2, \dots, A_n\}$
- é um conjunto de atributos, subconjunto de R , com a propriedade de que $t_1[S] \neq t_2[S]$ para qualquer extensão $r(R)$
- Uma superchave, K , é uma chave se K é uma superchave mínima
- Se uma relação esquema tiver mais de uma chave, cada chave será chamada de chave-candidata. Uma das chaves-candidatas é arbitrariamente escolhida para ser a chave-primária e as outras são chamadas de chaves-secundárias



- Um atributo primo (ou primário) é membro de alguma chave-candidata
- Um atributo não-primo é um atributo que não é primo – isto é, não é membro de qualquer chave-candidata





Primeira Forma Normal

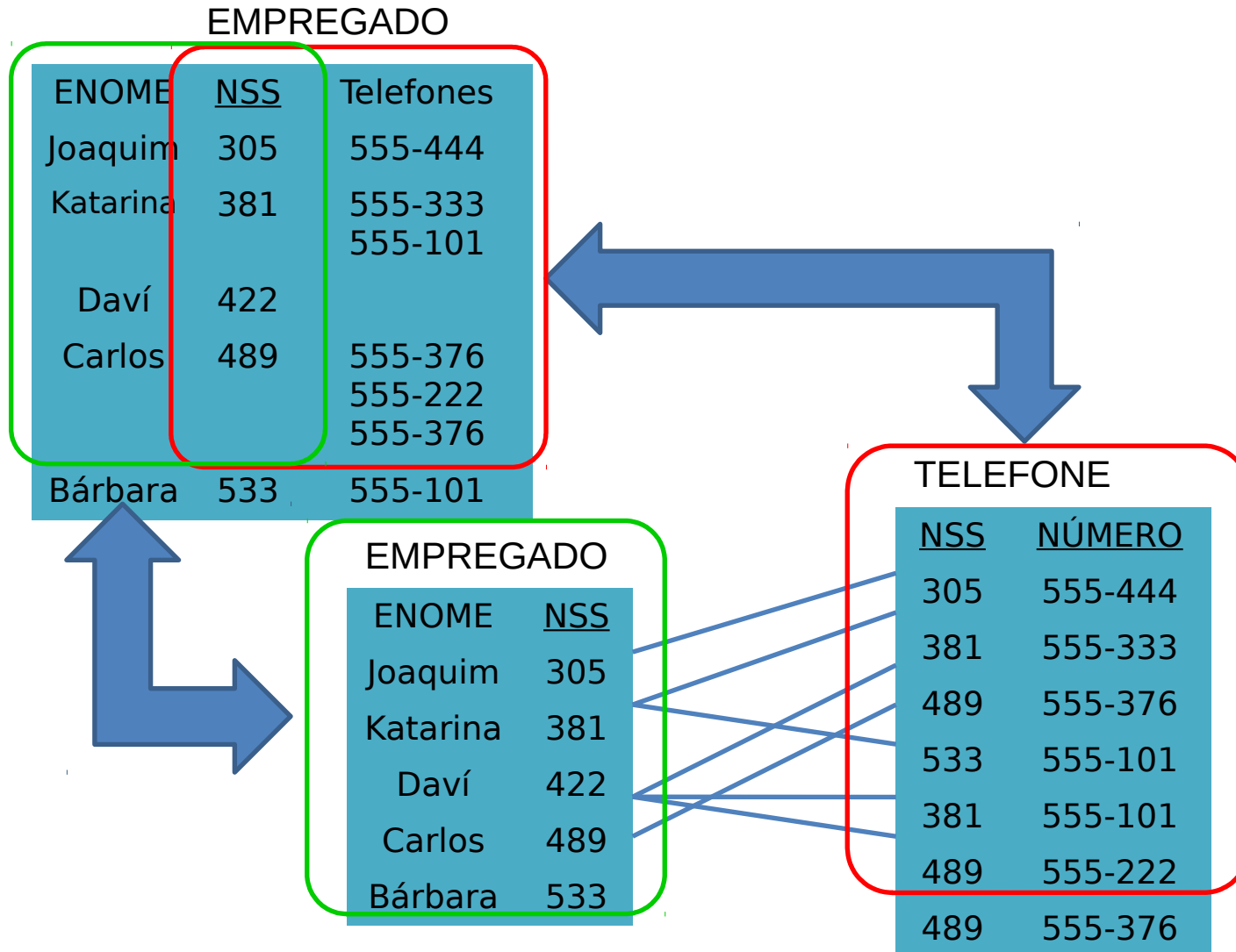


- Proíbe que relações tenham
 - Atributos compostos
 - Atributos multivalorados
 - Relações aninhadasOu seja
 - Permite apenas atributos que sejam atômicos
- Considerado como parte da definição de relação





Normalização na 1 FN





Normalização de Relações com Atributos Compostos para a 1 FN

EMPREGADO_PROJETO

ENOME	NSS	PROJETOS	
		PNUMERO	HORAS
Joaquim	305	1	32
Joaquim	305	2	8
Daví	422	3	40
Carlos	489	1	20
Carlos	489	3	20

EMPREGADO

ENOME	<u>NSS</u>
Joaquim	305
Daví	422
Carlos	489

PROJETO

<u>NSS</u>	<u>PNUMERO</u>	HORAS
305	1	32
305	2	8
422	3	40
489	1	20
489	3	20



Segunda Forma Normal





Segunda Forma Normal

- Para entender a 2FN precisamos entender:
 - Dependência Funcional
 - Chave-primária
 - Atributo Não-Primo
 - Dependência funcional total
 - Uma DF, $Y \rightarrow Z$, onde a remoção de qualquer atributo de Y invalida a DF. Exemplos:
 - $\{ \text{NSS}, \text{PNUMERO} \} \rightarrow \text{HORAS}$ é dependente totalmente de
 - $\{ \text{NSS}, \text{PNUMERO} \}$, uma vez que
 - NSS não determina HORAS e nem PNUMERO determina HORAS
 - $\{ \text{NSS}, \text{PNUMERO} \} \rightarrow \text{ENOME}$ não é dependente totalmente de
 - $\{ \text{NSS}, \text{PNUMERO} \}$; ENOME é dependente parcialmente de
 - $\{ \text{NSS}, \text{PNUMERO} \}$, pois $\text{NSS} \rightarrow \text{ENOME}$



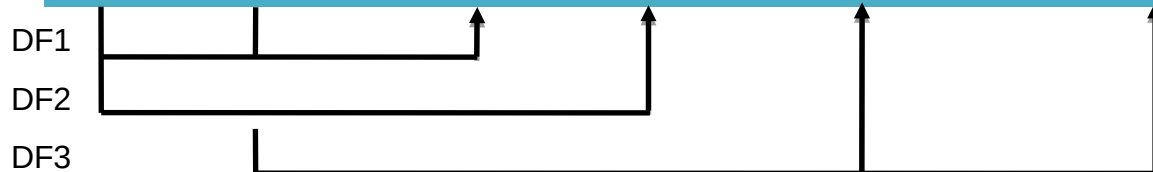
- Uma relação esquema R está na 2FN se
 - Estiver na 1FN e
 - Todos os atributos não-primos A de R forem totalmente dependentes da chave-primária
- R pode ser decomposto em relações que estejam na 2 FN através do processo de normalização
- Visa a diminuição da redundância e o desagrupamento de informações. Na 2FN, uma tabela representa um quantidade menor de entidades.



Normalização para a 2FN e 3FN

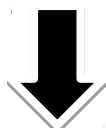
EMPREGADO_PROJETO

NSS PNUMERO HORAS ENOME PNUMO PLOCALIZACAO



EMPREGADO_PROJETO

NSS PNUMERO HORAS



Não-Primo

**Depende totalmente
da chave-primária**

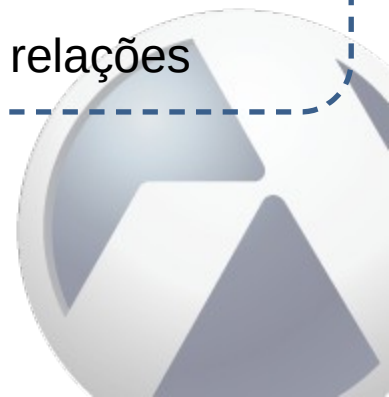
EMPREGADO

NSS ENOME

PROJETO

PNUMERO PNUMO PLOCALIZACAO

O mesmo ocorre com as outras relações





Terceira Forma Normal





Terceira Forma Normal

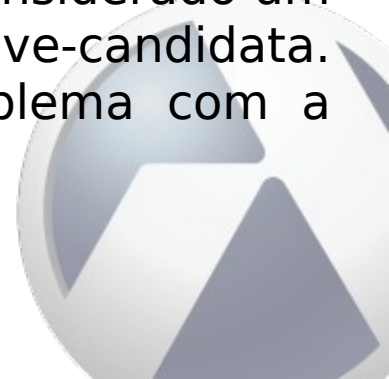
- Para entender a 3FN precisamos entender:
 - 2FN
 - Atributo Não-Primo
 - Dependência funcional transitiva
 - Se $X \rightarrow Y$ e $Y \rightarrow Z$ então $X \rightarrow Z$





Terceira Forma Normal

- Uma relação esquema R está na 3FN se:
 - Ela estiver na 2FN e
 - Nenhum atributo não-primo, A, for transitivamente dependente da chave-primária
- R pode ser decomposto em relações que estejam na 3FN via o processo de normalização
- Visa a diminuição da redundância eliminando a transitividade
- NOTA:
 - Em $X \rightarrow Y$ e $Y \rightarrow Z$, sendo X a chave-primária, pode ser considerado um problema se, e somente se, Y não for uma chave-candidata. Quando Y é uma chave-candidata, não existe problema com a dependência transitiva
 - Por exemplo, considere EMP (NSS, Emp#, Salario).
 - Aqui, $NSS \rightarrow Emp\# \rightarrow Salario$ e Emp# é uma chave-candidata





Terceira Forma Normal

Exemplo

EMPREGADO_PROJETO

ENOME NSS DTNASC ENDERECO DNUMERO DNOME PLOCALIZACAO



EMPREGADO

ENOME NSS DTNASC ENDERECO DNUMERO



PROJETO

DNUMERO DNOME PLOCALIZACAO





Terceira Forma Normal

Exemplo

Itens do Pedido				
Pedido	Item	Preço	Quantidade	Total
15	102	9,25	2	18,5
15	132	1,3	5	6,5





Definição Geral de Formas Normais

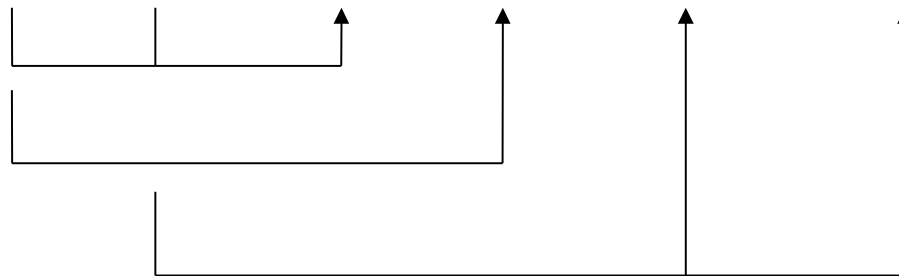
- As definições anteriores consideravam somente a chave-primária
- As próximas definições levarão em consideração as várias chaves candidatas





Definição Geral de Formas Normais

- Redefinição da 2FN:
 - Uma relação esquema R está na 2FN se todos os atributos não-primos, A, forem totalmente dependentes de todas as chaves de R
- Teste:
 - Verifique que EMP_PROJ não está na 2FN
 - EMP_PROJ (nss, pnúmero, horas, enome, pnome, plocalizacao)





Faculdade
IMPACTA
TECNOLOGIA

Exercícios



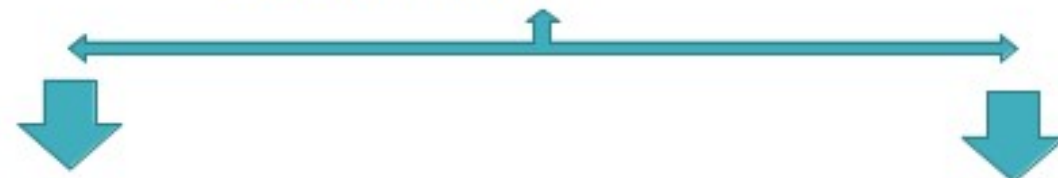


<u>CódForn</u>	Nome	Status	Cidade	<u>CódPeça</u>	Preço	Qtde	Valor
F1	Fornecedor 1	20	Londres	P1	R\$ 10,00	300	R\$ 3.000,00
F1	Fornecedor 1	20	Londres	P2	R\$ 20,00	200	R\$ 4.000,00
F1	Fornecedor 1	20	Londres	P3	R\$ 15,00	400	R\$ 6.000,00
F1	Fornecedor 1	20	Londres	P4	R\$ 25,00	200	R\$ 5.000,00
F1	Fornecedor 1	20	Londres	P5	R\$ 12,00	100	R\$ 1.200,00
F1	Fornecedor 1	20	Londres	P6	R\$ 5,00	100	R\$ 500,00
F2	Fornecedor 2	10	Paris	P1	R\$ 10,00	300	R\$ 3.000,00
F2	Fornecedor 2	10	Paris	P2	R\$ 20,00	400	R\$ 8.000,00
F3	Fornecedor 3	10	Paris	P2	R\$ 20,00	200	R\$ 4.000,00
F4	Fornecedor 4	20	Londres	P2	R\$ 20,00	200	R\$ 4.000,00
F4	Fornecedor 4	20	Londres	P4	R\$ 25,00	300	R\$ 7.500,00
F4	Fornecedor 4	20	Londres	P5	R\$ 12,00	400	R\$ 4.800,00





Chave Primária

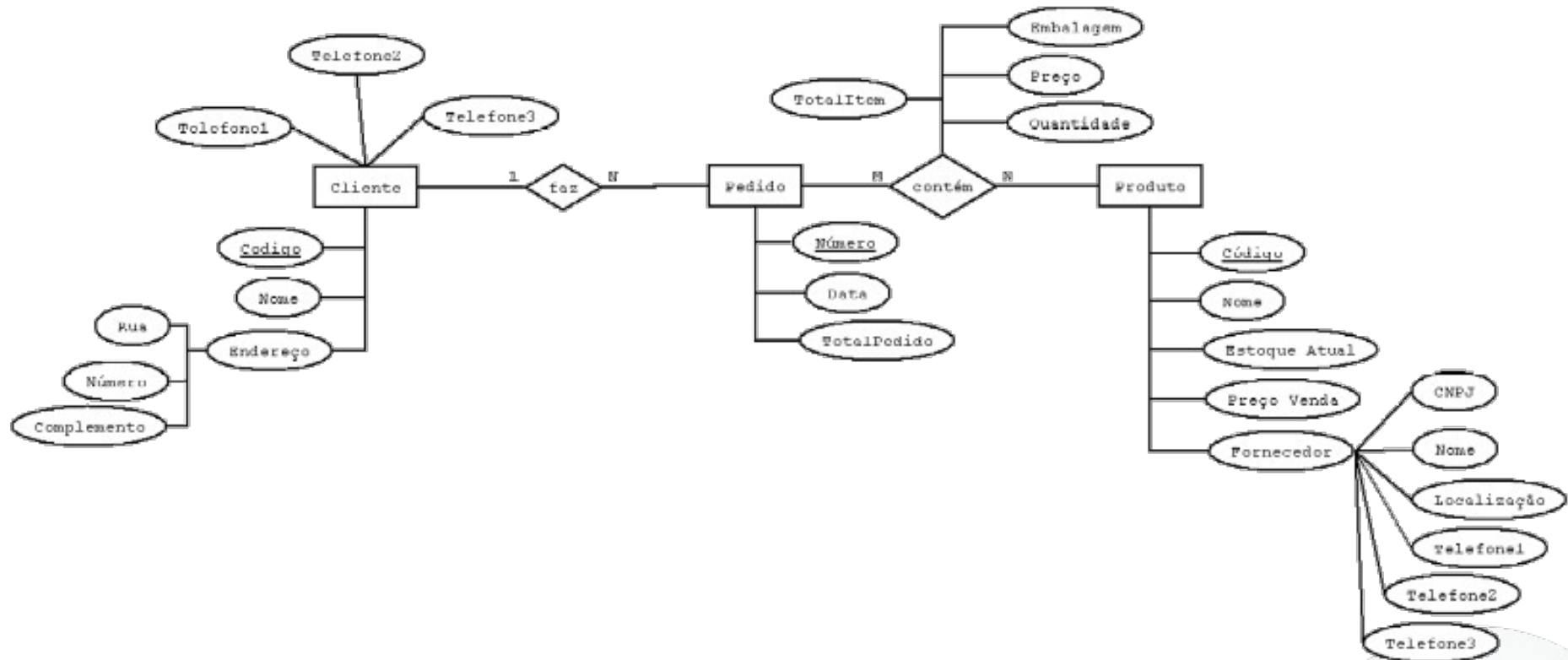


NroNota	Data	CodForn	Nome	Telefone	Endereço	CodItem	CodProd	Produto	Qtda	Preço	TotalItem
1	15/6/2003	22	Empresa Excalibur	3335-5255	Rua Itamaracá, 15	1	CA	Chapas de Aço	35	R\$ 15,00	R\$ 525,00
						2	BB	Bobina	20	R\$ 15,00	R\$ 300,00
						3	TC	Tábuas Corridas	50	R\$ 20,00	R\$ 1.000,00
2	20/7/2003	17	Cia Silva	3334-4787	Rua Cardoso, 145	1	TC	Tábuas Corridas	50	R\$ 20,00	R\$ 1.000,00
						2	CM	Compensado	30	R\$ 15,00	R\$ 450,00
						3	PM	Ripas de Madeira	300	R\$ 2,00	R\$ 600,00
3	22/8/2003	22	Empresa Excalibur	3335-5255	Rua Itamaracá, 15	1	CA	Chapas de Aço	50	R\$ 15,00	R\$ 750,00
						2	BB	Bobina	20	R\$ 15,00	R\$ 300,00





Mapeie e Normalize





Forma Normal de Boyce-Codd



BCNF (Boyce-Codd Normal Form)

- Definição de BCNF:
 - Uma relação esquema R está na BCNF se, sempre que houver uma DF $X \rightarrow A$ em R, então X é uma superchave de R
- Cada FN engloba a FN anterior:
 - Toda relação em 2FN está na 1FN
 - Toda relação em 3FN está na 2FN
 - Toda relação em BCNF está na 3FN
- Existem relações que estão na 3FN mas não em BCNF
- A meta é alcançar a BCNF ou 3FN em todas as relações





Boyce-Codd normal form

R

<u>ESTUDANTE</u>	<u>CURSO</u>	INSTRUTOR
Nair	Banco de dados	Marcos
Silas	Banco de dados	Nico
Silas	Sistemas Operacionais	Altair
Silas	Teoria	Saulo
Wilson	Banco de Dados	Marcos
Wilson	Sistemas Operacionais	Álvaro
Wellington	Banco de Dados	Carlos
Zenaide	Banco de Dados	Nico

Relação em 3FN mas não em BCNF

Uma relação esquema R está na **3FN** se, sempre que houver uma DF $X \rightarrow A$, então:

- X é uma superchave de R
- ou**
- A é atributo primo de R.

Uma relação esquema R está na **BCNF** se, sempre que houver uma DF $X \rightarrow A$, então:

- X é uma superchave de R





Alcançando a BCNF pela Decomposição

- Existem duas DF em relação:
 - df1: { estudante, curso } \rightarrow instrutor
 - df2: instrutor \rightarrow curso
 - Se a relação tivesse apenas df1, a relação estaria na BCNF.
 - Mas em df2, instrutor não é uma superchave, e, portanto, viola a BCNF, mas não a 3FN, pois curso é primo.
- Uma relação que não esteja na BCNF deve ser decomposta para atender a esta propriedade, mas abdica da preservação das dependências funcionais nas relações decompostas





Alcançando a BCFN pela Decomposição

- Três possíveis decomposições para relação:
 - $R1(\text{estudante}, \text{instrutor})$ e $R2(\text{estudante}, \text{curso})$
 - $R1(\text{curso}, \text{instrutor})$ e $R2(\text{curso}, \text{estudante})$
 - $R1(\text{instrutor}, \text{curso})$ e $R2(\text{instrutor}, \text{estudante})$
- Todas as três decomposições perdem a $df1$.
 - Temos que conviver com este sacrifício, mas não podemos sacrificar a propriedade não-aditiva após a decomposição.
- Das três, apenas a terceira decomposição não gera tuplas espúrias após a junção (join), e, assim, mantém a propriedade não-aditiva.



Alcançando a BCFN pela Decomposição

R1

INSTRUTOR	ESTUDANTE
Marcos	Nair
Nico	Silas
Altair	Silas
Saulo	Silas
Marcos	Wilson
Álvaro	Wilson
Carlos	Wellington
Nico	Zenaide

R2

INSTRUTOR	CURSO
Marcos	Banco de dados
Nico	Banco de dados
Altair	Sistemas Operacionais
Saulo	Teoria
Álvaro	Sistemas Operacionais
Carlos	Banco de Dados

X
(JOIN)

Relação original: R



ESTUDANTE	CURSO	INSTRUTOR
Nair	Banco de dados	Marcos
Silas	Banco de dados	Nico
Silas	Sistemas Operacionais	Altair
Silas	Teoria	Saulo
Wilson	Banco de Dados	Marcos
Wilson	Sistemas Operacionais	Álvaro
Wellington	Banco de Dados	Carlos
Zenaide	Banco de Dados	Nico

Note que para as
outras possíveis
decomposições,
isso não
acontece.



Decomposição sem perdas

- A decomposição de R em X e Y é sem perdas se e somente se, pelo menos uma das duas DF for válida
 - $X \cap Y \rightarrow X$ ou (Intersecção)
 - $X \cap Y \rightarrow Y$
- Caso especial
 - Se $U \rightarrow V$, então a decomposição de R em UV e $R - V$ é sem perdas.

Verifique que a decomposição de R satisfaz esta condição!

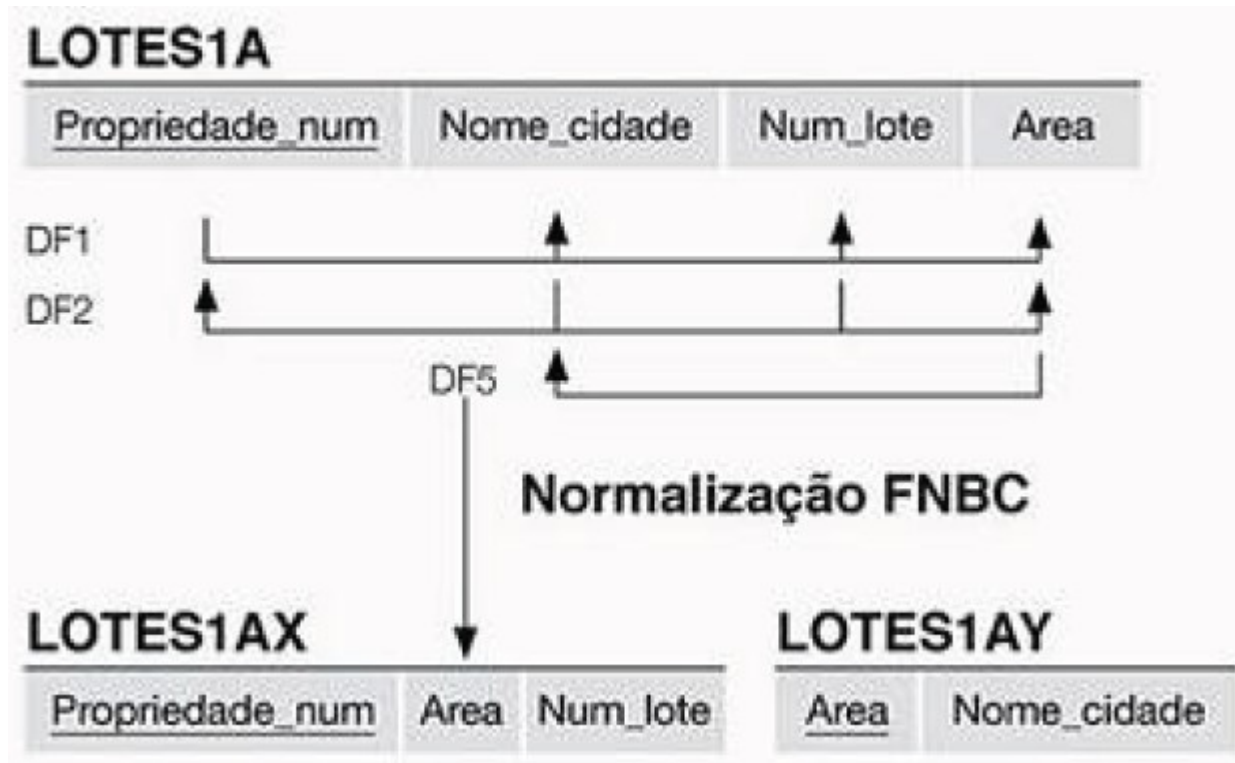
- https://pt.wikipedia.org/wiki/Tabela_de_s%C3%ADmbolos_matem%C3%A1ticos



Algoritmo de Decomposição BCNF

- Considere uma relação R e suas DFs associadas.
 - Se $X \rightarrow Y$ violar a FNBC, decomponha R em XY e $R - Y$.
- Aplicando esta idéia repetidamente, obteremos uma decomposição sem perdas de R em uma coleção de relações na BCNF.
- Em geral, mais de uma DF pode violar a BCNF. Dependendo da ordem em que as dependências são tratadas, podemos obter decomposições diferentes (e mesmo assim corretas).

FNBC: Exemplo





Quarta Forma Normal





- As dependências multivaloradas são consequência da 1FN, a qual não aceita atributos multivalorados.
 - Considere, por exemplo, a relação ACERVO abaixo:

ISBN	AUTOR	CÓPIAS
85-7323-169-6	Dantas	1, 2
0-13031-995-3	Molina, Ulman, Widom	1, 2

- Relação Normalizada para BCNF (note que não há DFs)

ISBN	AUTOR	CÓPIAS
85-7323-169-6	Dantas	1
85-7323-169-6	Dantas	2
0-13031-995-3	Molina	1
0-13031-995-3	Molina	2
0-13031-995-3	Ulman	1
0-13031-995-3	Ulman	2
0-13031-995-3	Widom	1
0-13031-995-3	Widom	2

Mas ainda temos
redundâncias por que?

Porque existem dependências
multivaloradas!

ISBN → AUTOR
ISBN → CÓPIAS



- Sempre que $X \twoheadrightarrow Y$ ocorrer, dizemos que X multidetermina Y .
- Devido a semetria da definição, sempre que $X \twoheadrightarrow Y$ ocorrer em R , também ocorre $X \twoheadrightarrow Z$.
- Por isso, $X \twoheadrightarrow Y$ implica $X \twoheadrightarrow Z$; por isso, às vezes é escrito como $X \twoheadrightarrow Y \mid Z$.
- Então, na relação ACERVO do exemplo anterior:
 - ISBN \twoheadrightarrow AUTOR \mid CÓPIAS





- Elimina redundâncias provocadas pelas dependências multivaloradas (MVD).
- Uma relação está na 4FN se não contiver mais de uma MVD.
 - Mas porque é tão ruim ter uma tabela com múltiplas dependências multivaloradas?
 - Em ACERVO
 - Para inserir mais uma cópia do ISBN 0-13031-995-3, será necessário inserir 3 tuplas, uma para cada autor.

ISBN	AUTOR	CÓPIAS
85-7323-169-6	Dantas	1
85-7323-169-6	Dantas	2
0-13031-995-3	Molina	1
0-13031-995-3	Molina	2
0-13031-995-3	Ulman	1
0-13031-995-3	Ulman	2
0-13031-995-3	Widom	1
0-13031-995-3	Widom	2

Alterações e
Remoções carecem
com mesmo
problema



- A solução é decompor a relação ACERVO em duas

De acordo com as MVD

ISBN \rightarrow AUTOR
ISBN \rightarrow CÓPIAS

ISBN	AUTOR
85-7323-169-6	Dantas
0-13031-995-3	Molina
0-13031-995-3	Ulman
0-13031-995-3	Widom

ISBN	CÓPIAS
85-7323-169-6	1
85-7323-169-6	2
0-13031-995-3	1
0-13031-995-3	2

- A MVD desejável é aquele cujo determinante é superchave da relação

Caso especial

Se R tiver as seguintes MVD

$A \rightarrow B$ e $B \rightarrow C$

Neste caso, R estará na 4FN se, e somente se, B e C são dependentes um do outro.





Quinta Forma Normal





5FN e Dependência de Junção

- Algumas vezes uma relação não pode ser decomposta sem perdas em duas relações, mas pode ser decomposta em três ou mais.
- A 5FN capta a idéia de que uma relação esquema deve ter alguma decomposição sem perda (dependência de junção).
- Encontrar casos reais da 5FN é difícil.





5FN e Dependência de Junção

- Um pequeno exemplo

AEP

AGENTE	EMPRESA	PRODUTO
Smith	Ford	Carro
Smith	Ford	Caminhão
Smith	GM	Carro
Smith	GM	Caminhão
Jones	Ford	Carro

Regra:

Se um AGENTE vende um certo
PRODUTO e este AGENTE
representa uma EMPRESA que faz
este PRODUTO

então

O AGENTE deve vender o
PRODUTO para a EMPRESA.

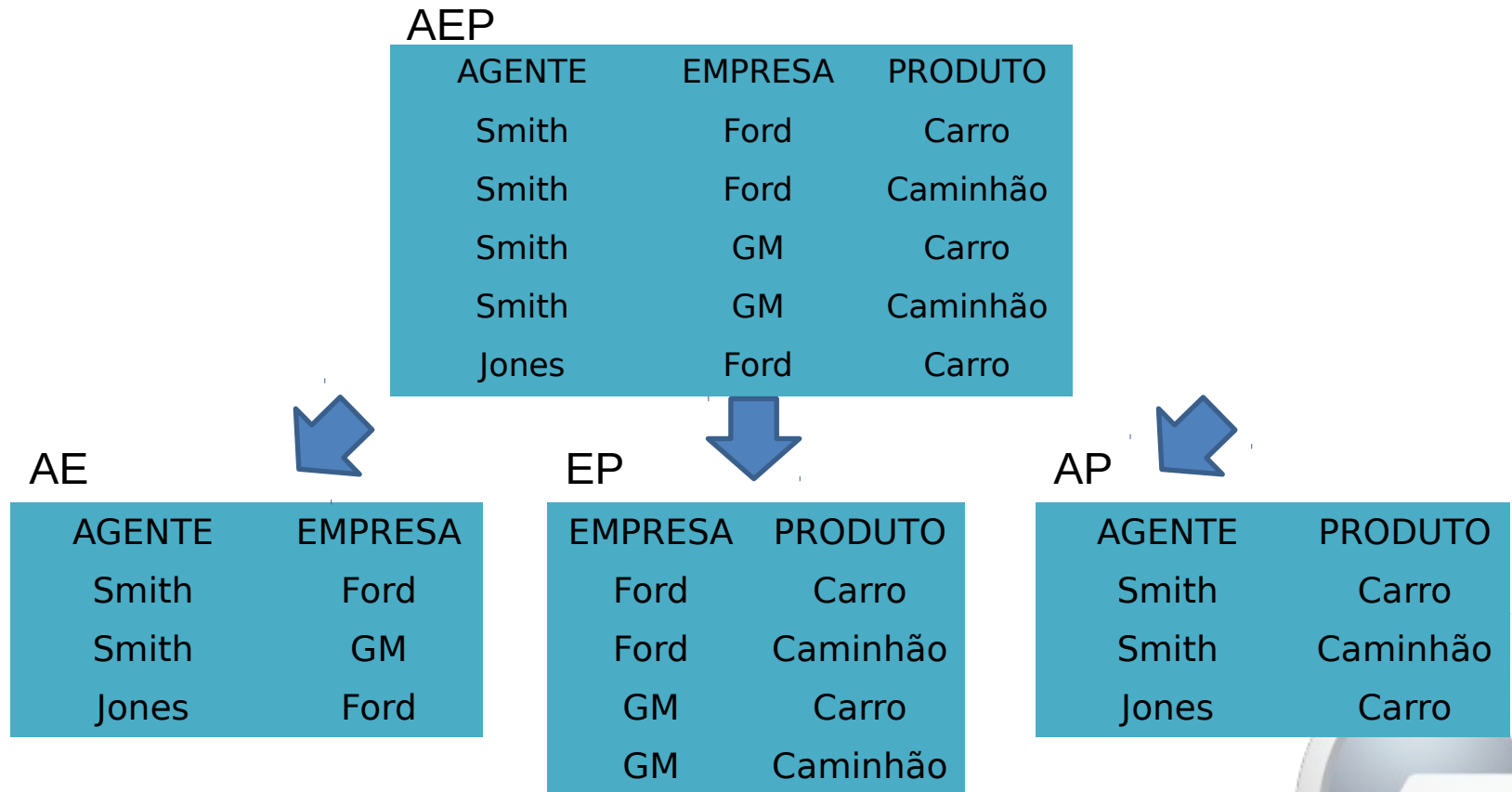
AGENTES representam EMPRESAS
EMPRESAS fazem PRODUTOS
AGENTES vendem PRODUTOS





5FN e Dependência de Junção

- Um pequeno exemplo



$$AEP = AE * EP * AP$$



5FN e Dependência de Junção

- Dependência de Junção
 - Uma relação R satisfaz a dependência de junção
 - JD (R1, R2, ..., Rn) se
 - $R = R1 * R2 * ... * Rn$
 - onde R1, R2, ..., Rn são subconjuntos dos atributos de R.
 - Note que uma dependência multivalorada é um caso especial de dependência de junção (n=2).



5FN e Dependência de Junção

- Uma relação R está na 5FN se, e somente se, ela estiver na 4FN e todas as suas dependências de junção forem determinadas pelas chaves candidatas.
- A descoberta de DJs em bancos de dados reais com centenas de atributos é praticamente impossível. Isso poderá ser feito apenas contando com um grande grau de intuição sobre os dados por parte do projetista. Por isso, a prática atual de projeto de banco de dados dá pouca atenção a elas (Elmasri & Navathe 4ª. Edição).



- Existem outras formas normais, porém elas estão fora do escopo desta disciplina, pois são formas pouco utilizadas em projetos de banco de dados devido a sua dificuldade de aplicação prática.





- Dado o DER de uma locadora de vídeo (próximo slide), e o mapeamento realizado para o esquema do BD Relacional, verifique a qualidade das relações obtidas (qual forma normal atingida) e, se necessário, normalize todos os esquemas de relações para a 3FN ou, se possível, para a BCNF.





Exemplo





Esquema de Relações do BD

- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun, NomeMun, SiglaEst, NomeEst)
- FoneEmailCliente (CodCli-CE, Email, Telefone)
- Venda (NumVda, DataVda)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodFab-CE, CodCol-CE, CodMol-CE)
- Estoque (CodProd-CE, Numero, Cor, Quantidade, CodBarra, Preço)
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade, TotalItem, CodCli-CE)



Primeira Forma Normal

Eliminar Atributos Não Atômicos





1FN: Primeira Forma Normal

- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun, NomeMun, SiglaEst, NomeEst)
- FoneEmailCliente (CodCli-CE, Email, Telefone)
- Venda (NumVda, DataVda)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodFab-CE, CodCol-CE, CodMol-CE)
- Estoque (CodProd-CE, Numero, Cor, Quantidade, CodBarra, Preço)
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade, TotalItem, CodCli-CE)



Segunda Forma Normal

Eliminar Dependência Parcial da Chave Primária





2FN: Segunda Forma Normal

- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun, NomeMun, SiglaEst, NomeEst)
- FoneEmailCliente (CodCli-CE, Email, Telefone)
- Venda (NumVda, DataVda, CodCli-CE)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodBarra, Preço, CodFab-CE, CodCol-CE, CodMol-CE)
- ~~Estoque (CodProd-CE, Numero, Cor, Quantidade, CodBarra, Preço) Fere a 2FN~~
 - CodProd → CodBarra, Preço, portanto temos uma dependência parcial da PK
- ~~Estoque (CodProd-CE, Número, Cor, Quantidade)~~
- ~~Tem (NumVda-CE, CodProd-CE, Preço, Quantidade, TotalItem, CodCli-CE) Fere 2FN~~
 - NumVda-CE, CodProd-CE → Preço, Quantidade, CodCli-CE
 - NumVda-CE → CodCli-CE
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade, TotalItem)





Terceira Forma Normal Eliminar Dependência por Transitividade da Chave Primária





3FN: Terceira Forma Normal

- ~~Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun, NomeMun, SiglaEst, NomeEst)~~ Fere a 3FN
 - CodCli → CodMun
 - CodMun → NomeMun, SiglaEst
 - SiglaEst → NomeEst, portanto temos 2 dependências por transitividade!
- Estado (SiglaEst, NomeEst)
- Municipio (CodMun, NomeMun, SiglaEst-CE)
- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun-CE)
- FoneEmailCliente (CodCli-CE, Email, Telefone)
- Venda (NumVda, DataVda, CodCli-CE)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodBarra, Preço, CodFab-CE, CodCol-CE, CodMol-CE)
- Estoque (CodProd-CE, Número, Cor, Quantidade)
- ~~Tem (NumVda-CE, CodProd-CE, Preço, Quantidade, TotalItem, CodCli-CE)~~ Fere a 3FN
 - NumVda-CE, CodProd-CE → Preço, Quantidade
 - Preço, Quantidade → TotalItem, portanto temos aqui transitividade!
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade)





Quarta Forma Normal Eliminar Múltiplas Redundâncias Multivaloradas





4FN: Quarta Forma Normal

- Estado (SiglaEst, NomeEst)
- Municipio (CodMun, NomeMun, SiglaEst-CE)
- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun-CE)
- ~~FoneEmailCliente (CodCli-CE, Email, Telefone)~~ Fere a 4FN
 - CodCli →→ Email
 - CodCli →→ Telefone, portanto temos mais de uma dependência multivalorada!
- TelefoneCLiente (CodCli-CE, Telefone)
- EmailCLiente (CodCli-CE, Email)
- Venda (NumVda, DataVda, CodCli-CE)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodBarra, Preço, CodFab-CE, CodCol-CE, CodMol-CE)
- Estoque (CodProd-CE, Número, Cor, Quantidade)
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade)





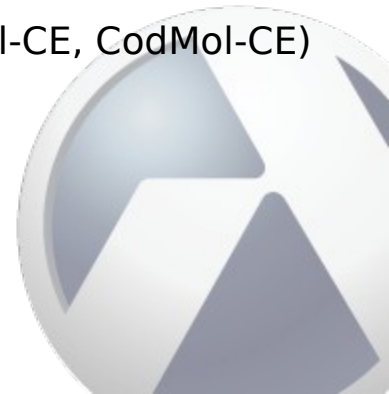
Esquema de Relações do Banco de Dados Normalizado





4FN: Quarta Forma Normal

- Estado (SiglaEst, NomeEst)
- Municipio (CodMun, NomeMun, SiglaEst-CE)
- Cliente (CodCli, CPF, NomeCli, Rua, Numero, Complemento, CEP, CodMun-CE)
- TelefoneCLiente (CodCli-CE, Telefone)
- EmailCLiente (CodCli-CE, Email)
- Venda (NumVda, DataVda, CodCli-CE)
- Fabricante (CodFab, NomeFab)
- Modelo (CodMod, NomeMod)
- Colecao (CodCol, NomeCol)
- Produto (CodProd, NomeProd, Genero, CodBarra, Preço, CodFab-CE, CodCol-CE, CodMol-CE)
- Estoque (CodProd-CE, Número, Cor, Quantidade)
- Tem (NumVda-CE, CodProd-CE, Preço, Quantidade)





Forma Normal de Boyce-Codd





FNBC: Forma Normal de Boyce-Codd

- Na Impacta e na grande maioria das faculdades, uma disciplina pode ser lecionada por mais de um professor.
- Então, desta maneira temos as seguintes dependências funcionais:
 - CodTurma -> CodProfessor, CodDisciplina
- E a relação Professor_Disciplina_Turma:
 - Professor_Disciplina_Turma (CodTurma, CodProfessor, CodDisciplina)





FNBC: Forma Normal de Boyce-Codd

- Entretanto, existem universidades (geralmente públicas) onde uma disciplina é lecionada por somente um professor!
- Nesta realidade, teríamos as seguintes dependências funcionais:
 - CodTurma -> CodDisciplina
 - CodDisciplina -> CodProfessor





FNBC: Forma Normal de Boyce-Codd

- Desta forma, seria necessária normalizar a relação Professor_Disciplina_Turma, pois ela fere a FNBC, uma vez que CodDisciplina não é uma superchave da seguinte forma:
 - Disciplina_Professor (CodDisciplina, CodProfessor)
 - Disciplina_Turma (CodTurma, CodDisciplina)





Leituras Recomendadas





Leituras Recomendadas

- ELMASRI, Ramez; NAVATHE, Shamkant B.. **Sistemas de banco de dados**. 6. ed. São Paulo: Pearson Education, 2011. 788 p.
 - Capítulos 15 e 16
- ou
- ELMASRI, Ramez; NAVATHE, Shamkant B.. **Sistemas de banco de dados**. 4. ed. São Paulo: Pearson Education, 2005. 724 p.
 - Capítulos 10 e 11
- RANGEL, Alexandre Leite et al. Unidade 5 – Normalização e Desempenho em Sistema de Banco de Dados Relacional. In: RANGEL, Alexandre Leite et al. **Banco de Dados**. Batatais: Claretiano, 2014. Cap. 5. p. 197-216.





Referências





■ Referências Bibliográficas

1. ELMASRI, Ramez; NAVATHE, Shamkant B.. **Sistemas de banco de dados**. 6. ed. São Paulo: Pearson Education, 2011. 788 p.
2. Korth, H.; Silberschatz, A. **Sistemas de Bancos de Dados**. 3a. Edição, Makron Books, 1998.
3. RAMAKRISHNAN, Raghu; GEHRKE, Johannes. **Sistemas de gerenciamento de banco de dados**. 3. ed. São Paulo: Mc Graw Hill, 2008. 884 p.
4. Teorey, T.; Lightstone, S.; Nadeau, T. **Projeto e modelagem de bancos de dados**. Editora Campus, 2007.





Obrigado

Prof. Dr. Alexandre L. Rangel
alexandre.leite@faculdadeimpacta.edu.br
www.alexandreirangel.blogspot.com.br

