

Normalização de Dados e Formas Normais

Prof: Aldelir Fernando Luiz

Disciplina: Banco de Dados I

Turma: 3º semestre

Curso: Bacharelado em Ciência da Computação

Semestre 01/



Tópicos

- Conceitos em Normalização de Dados
- Etapas da Normalização
- 1^a, 2^a e 3^a Formas Normais
- Estudos de caso



Normalização de Dados: Conceitos Básicos

- A ideia de **normalização** tem por princípio prover elementos para produzir relações bem estruturadas/definidas
- Ela visa evitar problemas causados pela **redundância**, isto é, o armazenamento repetido de dados, o qual pode causar problemas de:
 - **Mantenabilidade**: modificação ou exclusão de ocorrências/tuplas/registros implica acessos a diversas partes da base de dados, incorrendo em dificuldade de manter a coerência;
 - **Custo**: a redundância implica diretamente no custo de armazenamento, uma vez que o espaço é proporcional ao número de vezes que os mesmos os dados aparecem nas ocorrências;
 - **Desempenho**: operações de acesso ao subsistema de armazenamento (i.e., o disco) são as que mais incidem impactos sobre o desempenho dos sistemas.



Objetivos da Normalização de Dados

- Visa prover orientações para as especificação das relações de um banco de dados
- Evitar problemas provocados por anomalias de inclusão, exclusão e modificação/atualização de dados
 - Elas causam inconsistências de informações
- Produzir um conjunto de relações “bem definidas” e com as propriedades/atributos desejados, a partir dos requisitos de negócio identificados
- Substituir o conjunto de relações existentes por outro, a partir de um rigoroso processo de examinação dos atributos das relações analisadas



Princípio(s) da Normalização de Dados

- A normalização é um procedimento que examina rigorosamente os atributos de uma entidade
 - Como já dito, com o objetivo de evitar anomalias que possam ocorrer na inclusão, na exclusão ou na alteração de ocorrência(s) específica(s) em uma relação/entidade/tabela
- A normalização é um processo através do qual se cria relações “normalizadas” (i.e., bem formadas/definidas), com vistas para evitar a maioria dos problemas que surgem durante a concepção de um sistema informático baseado na teoria/modelo relacional de dados
- Também pode ser vista como um processo de engenharia reversa que visa obter um modelo lógico relacional a partir de especificações de bancos de dados não relacionais



Princípio(s) da Normalização de Dados

- Em suma, o resultado da aplicação da técnica da normalização é um banco de dados relacional “bem definido”
- A normalização também ajuda a compreender/identificar aspectos importantes como redundância de dados e chaves estrangeiras, ao passo que permite responder perguntas como:
 - Que atributos minha entidade deve ter?
 - Quais relacionamentos devem existir?
 - Como identificar a(s) cardinalidade(s)?
- Também organiza os atributos em tabelas, de modo que a redundância seja eliminada, isto é, remove grupos repetidos de informações (p. ex.: atributos compostos, multivalorados, etc.)

- A manutenibilidade de sistema de banco de dados depende de alguns aspectos observados durante o projeto de banco de dados:
 - A existência de grupos repetitivos
 - A existência de atributos multivalorados
 - Redundâncias de dados;
 - Perdas de Informações
 - Falhas quanto à sincronização de dados idênticos, em diferentes relações;
 - Existência de atributos que dependem apenas de parte de uma chave primária;
 - Existência de dependências transitivas entre atributos
- A normalização permite evitar a ocorrência de tais problemas

- O processo de normalização é realizado a partir da examinação dos atributos existentes nas relações
- Assim, para melhor compreender a especificação do *schema* de banco de dados é importante utilizar de um padrão de nomenclatura para os atributos
- Tal procedimento é adequado também, porque a maioria dos SGBDs comerciais limita em 32 caracteres a denominação de objetos, atributos e outras estruturas de dados pertinentes
- Neste sentido, ao adotar-se um padrão de nomenclatura deve-se identificar o objeto/atributo de maneira clara e descritiva



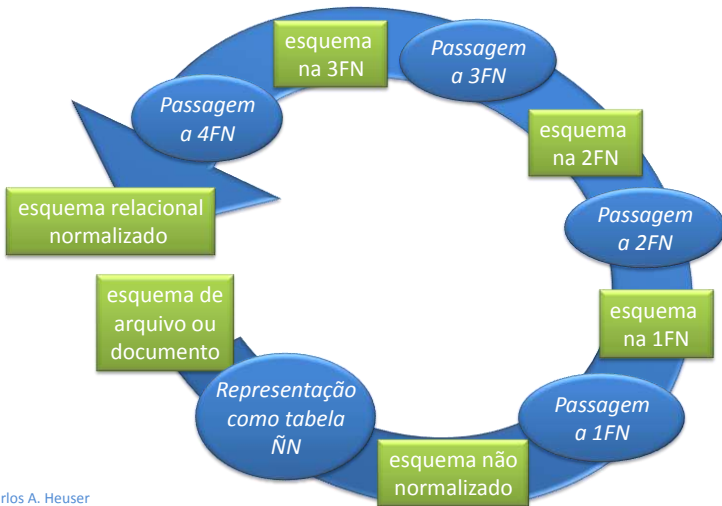
Processo de Normalização de Dados

- Sugestão para abreviação de atributos é a seguinte:

Sigla	Significado
cod	Código
des	Descrição
den	Denominação
txt	Texto
nom	Nome
num	Número
vlr	Valor
ind	Indicador
qtd	Quantidade
img	Imagem
dat	Data
hor	Hora
log	Lógico



Processo de Normalização de Dados (Passos)





Processo de Normalização de Dados (Passos)

DADOS NÃO NORMALIZADOS

Grupos de repetição



Decomposição de todas as estruturas de dados não planas em relações bi-dimensionais

PRIMEIRA FORMA NORMAL

Sem Grupos de repetição



Em relações com chave compostas, assegurar que todos os outros atributos dependem integralmente da mesma

SEGUNDA FORMA NORMAL

Dependência total



Eliminar as dependências transitivas, isto é, os atributos que dependem da chave primária por meio de outro(s) atributo(s)

TERCEIRA FORMA NORMAL

Dependência transitiva

- Abaixo, um exemplo de documento não normalizado

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 RH	8191	Mário	A1	4	1/05/93	12
			4112	João	A2	4	4/01/91	24
			6126	José	B1	9	1/11/92	12

- Abaixo, um exemplo de documento não normalizado

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



Processo de Normalização de Dados

- Analisemos a relação apresentada a seguir

Produto	Pedido	Cliente	Endereço	Crédito	Data	Vendedor
TV	1546	João	Rua A	6	01-03-10	Marcelo
DVD	2345	Miguel	Av. B	5	12-09-09	Felipe
Rádio	2466	Carlos	Trav. C	9	05-05-10	Marcelo
CD	0987	João	Rua A	11	11-12-08	Antonio
Geladeira	8576	Davi	Rua 9	4	15-04-10	Lidio



Processo de Normalização de Dados

- Visto que as informações de produtos, clientes e vendedores estão aglomeradas numa mesma relação, é notório que há uma redundância de informações (p. ex.: o endereço de João)
 - Isso incorre num desperdício de espaço de armazenamento, de modo que toda a tabela/relação precisa ser percorrida no caso de uma consulta ou atualização
- Para corrigir o problema é necessário decompor essa relação em relações menores, de modo que cada uma armazene apenas um único domínio de dados (p. ex.: produtos, clientes, vendedores, vendas...)



Conceito de Dependência Funcional

- O processo de decomposição de relações requer uma compreensão sobre o conceito Dependência Funcional
- Dependência funcional pode ser entendida como uma restrição existente entre dois conjuntos de **atributos** X e Y de uma **relação** R , de forma que o valor de um atributo identifique o valor para cada um dos outros atributos
 - Y é funcionalmente dependente de X ou
 - X determina Y ou
 - Y depende de X
 - se para cada valor distinto de X estiver associado um, e somente um valor de Y
- Isto é, Y é funcionalmente dependente de X se e somente se para cada valor de X em R existe um e somente um valor de Y na relação R



Conceito de Dependência Funcional

- A dependência funcional de dois atributos é denotada pelo símbolo \Rightarrow , de modo que, se Y é funcionalmente dependente de X , então $X \Rightarrow Y$ (X é determinante!)
- Considere o exemplo a seguir:

Pedido			
Produto	Preço (R\$)	Tributos (R\$)	Total (R\$)
TV LED	400,00	10,00	410,00
DVD Player	125,00	7,50	132,50
MicroSystem	300,00	8,00	308,00

- Note que os valores do atributo **Total** correspondem a soma dos campos referentes aos atributos **Preço** e **Taxas**
- Logo:** $\{\text{Preço}, \text{Taxas}\} \Rightarrow \text{Total}$



Conceito de Dependência Funcional

- Considere o exemplo a seguir:

Funcionário		
Matrícula	Nome	Setor
1020	Perovânio	A001
1021	Veneravez	A002
1022	Zenovevo	A002
1023	Ambrósia	A001
1024	Tenório	A002

- Como identificar as dependências funcionais presentes na tabela que representa a relação **Funcionário**?



Conceito de Dependência Funcional

- Considerando a relação **Funcionário**:
 - Setor \Rightarrow Matrícula?
R: Não, pois o setor 1001 identifica mais de uma matrícula
 - Matrícula \Rightarrow Setor?
R: Sim, pois se conhecido o funcionário é possível identificar qual seu setor de lotação
 - Nome \Rightarrow Matrícula?
R: Não, pois pode haver mais de um funcionário com o mesmo nome
 - Matrícula \Rightarrow Nome?
R: Sim, pois uma matrícula está associada a apenas um único nome
- **Logo:**
 - Matrícula \Rightarrow {Nome, Setor}



Conceito de Dependência Funcional Parcial

- Analisemos um exemplo de dependência funcional parcial:

Orçamento		
Comércio	Produto	Preço (R\$)
Jhyacy	Papel Higiênico	15,25
Hanjelonny	Escova Dental	8,75
Bystekah	Cerveja	2,99
Loirão	Papel Higiênico	14,88

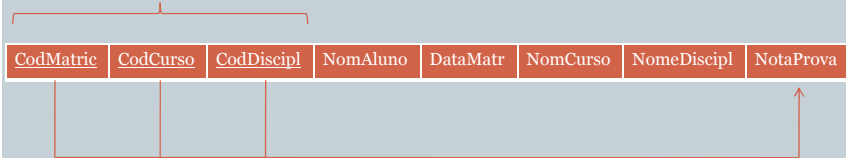
- Produto \Rightarrow Preço?
R: Não, pois o mesmo produto pode ter preços distintos praticados por diferentes comércios
- Comércio \Rightarrow Preço?
R: Não, pois para cada comércio há tantos preços quanto os produtos por ele comercializados
- **Logo:** {Comércio, Produto} \Rightarrow Preço



Conceito de Dependência Funcional Parcial

- Analisemos outro exemplo de dependência funcional parcial:

Chave Composta

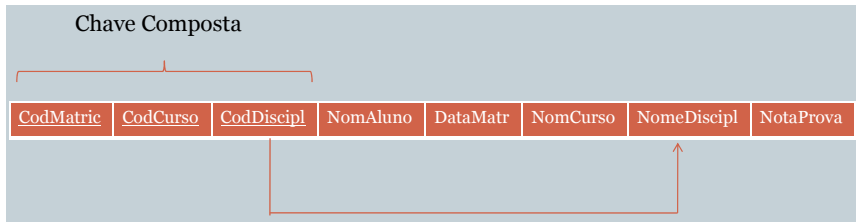


- Note que o atributo **NotaProva** só é identificado corretamente quando a chave está completa – porque a nota depende de aluno, curso e disciplina, e não de cada parte da chave, individualmente
- No caso, o atributo **NotaProva** é completamente dependente da chave composta



Conceito de Dependência Funcional Parcial

- Por outro lado, quando um atributo depende apenas de parte de uma chave composta, é dito que há uma dependência parcial



- É o caso do atributo **NomeDiscipl**, que depende apenas do atributo **CodDiscipl** – que é parte da chave primária composta



Conceito de Dependência Funcional Transitiva

- Em determinadas circunstâncias, pode ocorrer casos em que um atributo de uma relação não seja diretamente dependente da chave primária daquela relação ou de parte dela, mas sim de outro atributo daquela relação
- Neste caso, quando o(s) valor(es) de um atributo depende de valor(es) de atributo(s) que não é(são) chave ou não fazem parte de uma chave, é dito que há uma dependência transitiva

Funcionário				
Matrícula	Nome	IdCargo	Cargo	Salário
1001	Marivânia	1	Professor	750,00
1002	Anacleto	2	Secretária	997,25
1003	Gumercindo	3	Técnico	1.077,34
1004	Babilônio	1	Professor	750,00
1005	Tibéria	2	Secretária	997,25



Conceito de Dependência Funcional Transitiva

- Considerando a relação **Funcionário** do exemplo anterior:
 - Matricula \Rightarrow Nome?
R: Sim, pois uma matrícula está associada a apenas um único nome
 - Matricula \Rightarrow IdCargo?
R: Sim, pois se conhecido o funcionário é possível identificar qual seu cargo
 - Nome \Rightarrow Cargo?
R: Não, pois a denominação do cargo é determinada pelo atributo IdCargo
 - Nome \Rightarrow Salário?
R: Não, pois o valor do salário é determinado pelo cargo em exercício
- **Logo:**
 - Matrícula \Rightarrow {Nome, IdCargo}
 - IdCargo \Rightarrow {Cargo, Salário}



Conceito de Dependência Funcional Transitiva

- Note que o atributo **IdCargo** (i.e., identificador do cargo) – que não é chave primária – determina a denominação do cargo e o valor do salário, e estes atributos por sua vez, não dependem diretamente do atributo matrícula
- Tal situação é caracterizada como uma dependência funcional **transitiva**, isto é, um atributo que depende da chave primária por intermédio de outro atributo não chave primária
- Neste caso particular
 - Matrícula \Rightarrow IdCargo \Rightarrow {Cargo, Salário}
- Note que este tipo de dependência incorre numa redundância de informações (p. ex.: os funcionários Marivânia e Babilônio são professores e possuem os mesmos valores para atributos Cargo e Salario)



Etapas para a Consecução da Normalização de Dados

- O processo de normalização dar-se-á por meio da aplicação de “formas normais”
 - Formas normais consistem em regras, que tornam-se mais especializadas, na medida em que se avança o processo de normalização
- Originalmente, em 1972, Edgar Frank Codd definiu três formas normais, as quais denominou 1ª Forma Normal (FN), 2ª FN e 3ª FN
- ma definição mais forte da 3ª NF foi apresentada posteriormente por Boyce-Codd, a qual é conhecida como Forma Normal de Boyce-Codd (FNBC)
- Embora de aplicação mais rara, há também a 4ª FN e a 5ª FN



Etapas para a Consecução da Normalização de Dados

- Na maioria dos casos, a colocação das relações na terceira forma normal, já evita a maioria dos problemas comuns encontrados na concepção de projetos de bancos de dados relacionais mau especificados
- Muitos autores afirmam que a aplicação das três formas normais definidas por Codd é o suficiente para evitar redundâncias e inconsistências
- Essencialmente, as formas normais são aplicadas na ordem 1ª FN, 2ª FN e 3ª FN
- Já que as formas normais FNBC, 4ª NF e 5ª FN lidam com situações especiais que surgem ocasionalmente e, se necessário for, pode-se aplica-las na ordem FNBC, 4ª FN e 5ª FN

- Considere as ocorrências de dados anteriormente apresentadas

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 RH	8191	Mário	A1	4	1/05/93	12
			4112	João	A2	4	4/01/91	24
			6126	José	B1	9	1/11/92	12



Etapas para a Consecução da Normalização de Dados

- Como primeiro passo, deve-se transformar o documento a ser analisado em uma relação, nos termos do modelo relacional
- Assim, para o exemplo apresentado na lâmina anterior temos:

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

- Note que os parênteses aninhados identificam grupos de atributos cujos valores se repetem, a cada ocorrência



Etapas para a Consecução da Normalização de Dados

- Para outro exemplo de aplicação da normalização, considere as ocorrências de dados abaixo apresentadas:

Numer o do vende dor	Numer o do produt o	Nome do vende dor	Porcen tagem de comiss ão	Ano de contra tação	Numer o do depart ament o	Nome do gerent e	Nome do produt o	Preço Unitár io	Quanti dade
137	1000 1010 1030	João	3	1995	20	Carlos	Bola Cola Goma	8,00 15,00 2,00	4 6 8
221	2000 3010 4020	Marco	5	2000	10	Pedro	Lápis Régua Pincel	1,00 3,00 2,00	9 4 1
456	3000 9999 0348	Maria	3	1995	20	Carlos	Copo Jarro Palito	4,00 20,00 0,50	2 6 7



Etapas para a Consecução da Normalização de Dados

- O primeiro passo, antes de proceder com a aplicação das formas normais, consiste em especificar os identificados no artefato analisado, para a forma compatível com o modelo relacional
- No caso, para o exemplo apresentado na lâmina anterior temos:
Venda (num_vendedor, nom_vendedor, vlr_pct_comissao, dat_contratacao, num_depto, nom_gerente, (num_produto, nom_produto, vlr_preco_unitario, qtd_produto))
- Note que os parênteses aninhados identificam grupos de atributos cujos valores se repetem, a cada ocorrência

- A primeira forma normal visa eliminar os grupos de repetição (ou tabelas/relações aninhadas), que são observados no artefato objeto de normalização
- Neste caso, a 1ª Forma Normal (1FN) estabelece que uma entidade está em conformidade com ela somente se:
 - não possuir atributos multivalorados ou grupos repetitivos
 - todos os seus atributos são monovalorados e atômicos (i.e., não forem compostos por múltiplas partes)
 - existir uma chave primária que permita identificar cada ocorrência da relação
 - as ocorrências da relação forem distintas entre si
- Em suma, uma relação está na 1FN se todos os seus atributos são atômicos e monovalorados e se ela não contém relações aninhadas

- Para isso, dois procedimentos são necessários:
 - criar uma nova relação que contenha os atributos que formam a chave primária e os atributos multivalorados ou grupos repetitivos identificados
 - reorganizar a relação inicial sem os atributos multivalorados ou grupos repetitivos
- Neste sentido, com a aplicação da 1ª FN sobre a relação inicial **Venda**, teremos:

Produto (#num_produto, nom_produto, vlr_preco_unitario)

Venda (#num_vendedor, #num_produto, nom_vendedor, vlr_pct_comissao, dat_contratacao, num_depto, nom_gerente, qtd_produto)

- Note que ainda há uma mistura de diferentes domínios de dados numa única relação, especificamente de vendedor e venda

Venda (#num_vendedor, #num_produto, nom_vendedor, vlr_pct_comissao, dat_contratacao, num_depto, nom_gerente, qtd_produto)

- Tal situação será objeto de verificação pela 2ª FN

- A 2ª FN visa eliminar as dependências parciais em relação à chave primária, conforme observadas nas relações que já estão em 1FN
 - analisa-se se algum atributo possui dependência parcial em relação à chave primária “composta”
- Neste caso, uma relação está em conformidade com a 2ª FN somente se:
 - estiver na 1ª FN
 - não possuir atributos com dependência parcial em relação à chave primária da mesma relação
- Portanto, uma relação está em 2ª FN se ela estiver em 1ª FN e não contiver dependências funcionais parciais
 - uma dependência funcional parcial ocorre quando os atributos não chave não dependem funcionalmente de toda a chave primária, quando esta é **composta**

- A aplicação da 2ª FN dar-se-á por meio de dois procedimentos
 - criação de uma nova relação para manter os atributos que possuem dependência parcial em relação à chave primária – nesta relação também será colocado o atributo que é parte da chave primária, sobre o qual recai a dependência parcial
 - reorganização da relação normalizada sem os atributos que continham dependência parcial em relação à chave primária

- Assim, a partir da aplicação da 2ª FN sobre as relações em 1ª FN, teremos:

Venda (#num_vendedor, #num_produto, qtd_produto)

Vendedor (#num_vendedor, nom_vendedor,
vlr_pct_comissao, dat_contratacao, num_depto,
nom_gerente)

Produto (#num_produto, nom_produto, vlr_preco_unitario)

- Note que, mesmo após a passagem das relações pela 2ª FN nem toda redundância foi eliminada, pois alguns valores de atributos ainda se repetem, conforme se pode observar no que segue:

<u>Número do vendedor</u>	Nome do vendedor	Porcentagem de comissão	Ano contratação	Número departamento	Nome do gerente
137	João	3	1995	20	Carlos
221	Marco	5	2000	10	Pedro
456	Maria	3	1995	20	Carlos

- Tal situação decorre do fato de o atributo **Nome do Gerente** (ou simplesmente **nom_gerente**) ser funcionalmente dependente do atributo **Numero Departamento** (ou **num_depto**)
- Esta situação caracteriza uma dependência transitiva
- Tal situação será objeto de verificação pela 3ª FN

- A 3ª FN visa eliminar as dependências transitivas em relação à chave primária, conforme observado nas relações que já se encontram na 2ªFN
- Deste modo, uma relação está na 3ª FN somente se:
 - estiver na 2ª FN
 - não houver dependência transitiva entre os atributos que não fazem parte da chave primária
 - os atributos que não fazem parte da chave primária dependerem exclusivamente da chave primária
- Isto é, uma relação se encontra na 3ª FN se além de estar na 2ª FN, ela não conter dependências transitivas

- Uma dependência transitiva é caracterizada quando um atributo é indiretamente determinado pela chave primária da relação, porque tal atributo é funcionalmente dependente de outro atributo, o qual é dependente da chave primária
- Da mesma forma como ocorre com as demais FNs, a aplicação da 3ª FN dar-se-á por meio de dois procedimentos
 - criação uma nova entidade para conter os atributos que possuam dependência transitiva em relação à chave primária, juntamente com o atributo que estabelece o elo de ligação deste com a chave primária
 - reorganização da relação normalizada sem os atributos que continham dependência transitiva em relação à chave primária

- No caso, pela aplicação da 3ª FN sobre as relações em 2ª FN, teremos:

Vendedor (#num_vendedor, nom_vendedor,
vlr_pct_comissao, dat_contratacao, #num_depto)

Departamento (#num_depto, nom_gerente)



Normalização de Dados – Aspectos Importantes

- De um modo geral, as 3 formas normais aqui tratadas são suficientes para especificar um projeto lógico de banco de dados consistente e ausente de anomalias
- Por outro lado, a despeito destas 3 formas normais, há ainda três outras formas normais, pouco exploradas, mas que podem ser aplicadas em casos mais raros e específicos, são elas:
 - Forma Normal de Boyce-Codd (FNBC)
 - 4ª Forma Normal
 - 5ª Forma Normal
- Essas formas normais geralmente são aplicáveis em situações onde as relações são derivadas de relacionamentos ternários
- A ordem de aplicação é FNBC, 4ª NF e 5ª FN



Exercícios

Ficha Médica				
Número Paciente:	Nome:			
Data Nascimento:	Sexo:	Convênio:		
Estado Civil:	RG:	Telefone:		
Endereço:				
Consultas				
Número Consulta	Data	Médico	CRM	Diagnóstico
Exames				
Código Exame	Descrição Exame		Data	