



WEBAULA 15

- Normalização de Dados
- Referências

QUESTION BANKS

Normalização de Dados

Para Alves (2014) a normalização é um processo de refinamento do esquema de banco de dados que visa eliminar possíveis redundâncias (dados repetidos em entidades), sanar problemas de dependências parciais entre atributos e reduzir ao mínimo as anomalias de inclusão, alteração e exclusão de dados. O processo é dividido em várias etapas, que são denominadas tecnicamente de formas normais (FNs). Na aplicação das mesmas são efetuados diversos testes com o objetivo de se certificar de que o esquema satisfaz determinadas condições presentes na especificação cada FN. A partir desses testes, as relações são decompostas em relações menores conforme a necessidade. Por exemplo, uma relação pode ser dividida em duas ou mais, dependendo da situação.

A forma normal de uma relação indica o grau de normalização em que ela se encontra. Academicamente falando, existem cinco formas normais, embora apenas as três primeiras já sejam suficientes para se ter uma boa definição da estrutura do banco de dados. No fim da normalização tem-se a resposta à principal pergunta que surge logo no início de um projeto: quantas tabelas são necessárias em nosso banco de dados? As regras de normalização permitem que bancos de dados robustos e eficientes possam ser criados e facilmente alterados. Se essas regras forem seguidas com cuidado, o sistema todo (banco de dados e sistema) será bastante flexível e confiável.



(i) Dica

É possível que você estranhado o fato de ter usado o termo tabela ao questionar: "quantas tabelas são necessárias em nosso banco de dados"? Adiante falaremos sobre a derivação do MER conceitual para o projeto físico do banco de dados. Em bancos

relacionais, uma entidade é representada por uma tabela. Aos poucos vamos nos apropriando e se acostumando com estes termos.

Conforme antecipamos acima, três são as FNs mais usadas e aplicadas na modelagem de dados para sistemas computacionais. A partir de agora faremos uso de um exemplo para aplicar a definição de cada FN. Mas, antes, você vai conhecer a definição, o objetivo e a aplicação de cada uma delas.

É importante que você saiba de antemão que o fato de você conhecer conceitos de orientação a objetos favorecerá o seu entendimento. Mas isso não o(a) dispensa da atenção e de cuidados, pois são aos pequenos detalhes que devemos nos ater quando aplicamos as FNs. A partir deste momento, nosso foco abrange também os atributos, e não mais somente as entidades e os relacionamentos. O exemplo que iremos utilizar é considerado um minimundo completo por Alves (2014) e Heuser (2011) por apresentar os elementos necessários para identificação das fragilidades e consequência demonstração da aplicação das FNs. Procure observar em detalhes o conteúdo da nota fiscal abaixo. No cabeçalho temos as informações da empresa e o número do cupom fiscal. Em seguida temos as informações do cliente, do vendedor e dos produtos contidos na nota, conforme ilustrado na figura 20.

	ı	NOTA FISC	AL - VENDA A CONSUM	DOR	
	Maria S. da Silva o: Av. Martin Luther, s/n		Vendedor: Zé do Balcão		
,	· · · · · · · · · · · · · · · · · · ·		Data: 12/12/12		
Cód.	Descrição	UN.	QUANT.	Preço unitário	Preço total
123	Régua acrílico 30 cm	un.	1	0,80	0,8
234	Penal escolar ref. 1	un.	1	2,60	2,6
345	Grampo 0.08'	cx.	2	1,50	3,0
Total da n	ota				6,4

Dificilmente um analista/projetista iria propor a criação de uma tabela a partir do cenário apresentado como exemplo. Mesmo assim, vamos considerar que isso viesse a acontecer. Neste caso, uma descrição em linguagem natural iria possuir a seguinte estrutura visualizada na figura 21.

Tabela "nota fiscal":

Número, NomeCliente, EndereçoCliente, NomeVendedor, DataEmissão, CodProd1, DescrProd1, UnProd1, QtdeProd1, PrecoProd1, PrecoTotal1, CodProd2, DescrProd2, UnProd2, QtdeProd2, PrecoProd2, PrecoTotal2, CodProd3, DescrProd3, UnProd3, QtdeProd3, PrecoProd3, PrecoTotal3, CodProd4, DescrProd4, UnProd4, QtdeProd4, PrecoProd4, PrecoTotal4, CodProd5, DescrProd5, UnProd5, QtdeProd5, PrecoProd5, PrecoTotal5, CodProd6, DescrProd6, UnProd6, QtdeProd6, PrecoProd6, PrecoTotal6, TotalNota

Figura 21



(i) Reflita

Considerando a estrutura apresentada... será que o analista/projetista encontrará problemas na implementação e operacionalidade do sistema computacional?

Acredito que há dúvidas de quão a estrutura proposta é ineficaz. Nela temos a possibilidade de apresentar redundância nos dados, ausência ou excesso de atributos para manter os dados entre outras. Vamos iniciar a aplicação das FNs. Observe com atenção o que estabelece cada uma das FNs, sua aplicação e o resultado obtido. Uma tabela está na primeira forma normal sem nenhum dos seus atributos tem domínio multivalorado. Entende-se como multivalorado os atributos que apresentam a possibilidade de receber um ou mais valores. No exemplo temos o conjunto de atributos que

representam os itens (produtos) que compõe a "nota fiscal". Para evitar a reserva de espaços que não serão utilizados, bem como a falta de espaços que serão necessários, pois vemos no exemplo que há a indicação da existência de seis itens (produtos) na "nota fiscal", devemos projetar estes atributos multivalorados para outra tabela, levando um atributo como ligação.



(i) Dica

Temos dados multivalorados (ou domínio multivalorado) quando um atributo apresenta valores repetitivos ou tem mais de um valor, ou seja, não é atômico.

Vamos identificar na estrutura proposta inicialmente pelo nosso analista/projetista onde estão os dados multivalorados. A figura 22 ilustra o detalhe na tabela nota fiscal onde temos os atributos multivalorados.

Número, NomeCliente, EndereçoCliente, NomeVendedor, DataEmissao,

CodProd1, DescrProd1, UnProd1, QtdeProd1, PrecoProd1, PrecoTotal1, CodProd2, DescrProd2, UnProd2, QtdeProd2, PrecoProd2, PrecoTotal2, CodProd3, DescrProd3, UnProd3, QtdeProd3, PrecoProd3, PrecoTotal3, CodProd4, DescrProd4, UnProd4, QtdeProd4, PrecoProd4, PrecoTotal4, CodProd5, DescrProd5, UnProd5, QtdeProd5, PrecoProd5, PrecoTotal5, CodProd6, DescrProd6, UnProd6, QtdeProd6, PrecoProd6, PrecoTotal6,

TotalNota

Figura 22

Ao aplicarmos a 1ª FN teremos a seguinte estrutura:

Tabela "nota fiscal":

Número, NomeCliente, EndereçoCliente, NomeVendedor, DataEmissão, TotalNota

Tabela "item nota fiscal":

Número, CodProd, DescrProd, UnProd, QtdeProd, PrecoProd, PrecoTotal

Observe que houve um desmembramento da tabela "nota fiscal" para uma derivada que denominamos de "item nota fiscal". Nessa nova estrutura é possível observar o atributo "número" como ligação entre as mesmas. Logo, é possível deduzir que houve um relacionamento entre estas tabelas. Adiante facilmente será possível identificar a conectividade entre as tabelas. Agora, observe como ficam os dados armazenados após o redesenho da estrutura (figura 23).

Numero	Nome	Cliente Endered	coCliente	NomeVendedo	DataEmi	issao TotalNota
001234	Maria de S. Si	lva Av. Martin Lu	ther, s/n	Zé do Balcão	12/12/12	
Tabela "Item Nota	Fiscal"					
Tabela "Item Nota Numero	Fiscal" CodProd	DescrProd	UnidProd	QtdePro	d PrecoPr	od PrecoTotal
		DescrProd	UnidProd	QtdePro	d PrecoPr	od PrecoTotal
		DescrProd Régua acrílico 30 cm	UnidProd	QtdePro	i PrecoPr	od PrecoTotal 0,80
Numero	CodProd			QtdePro		
Numero 001234	CodProd	Régua acrílico 30 cm	Un	1	0,80	0,80

Figura 23

(i) Reflita			

Observe que ao aplicarmos a 1ª forma normal (FN) eliminamos a possibilidade de espaços que poderiam ficar ociosos e até mesmo a ausência de espaços para a inclusão de mais produtos à nota fiscal. Esse é o principal reflexo da aplicação da 1ª FN.

Passamos agora para a aplicação da 2ª FN. Esta prevê que projetemos os dados que dependem unicamente de um identificador (chave) para outra estrutura, mantendo esta ligação por meio do identificador. O objetivo desta FN é evitar que sejam mantidas informações sobre um conjunto que tem interseção com o conjunto representado na tabela, mas tem existência independente. No exemplo utilizado podemos observar que a tabela "itens nota fiscal" apresenta conjuntos de dados relacionados diretamente ao "produto". Assim, a aplicação da 2ª FN estabelece que projetemos os atributos que dependem funcionalmente do "número" (chave) para fora da tabela, levando a mesma (chave) para servir como ligação para refazer a ligação e recuperar o conteúdo da tabela original. Vamos identificar, na estrutura ajustada após a aplicação da 1ª FN onde ocorre a interseção de dados com existência própria, o seguinte:

Tabela "item nota fiscal":

Número,

CodProd, DescrProd, UnProd, QtdeProd, PrecoProd, PrecoTotal

Ao aplicar a 2ª forma normal, teremos uma estrutura desmembrada em duas, ou seja, mantemos a "item nota fiscal" e criamos uma derivada que denominamos de "produto", pois armazenará dados relacionados ao mesmo, seja:

Tabela "item nota fiscal":

Número,

CodProd, QtdeProd, PrecoProd (estático), PrecoTotal

Tabela "produto":

CodProd,

DescrProd, UnProd, PrecoProd (dinâmico)



(i) Reflita

Observe que o atributo "PrecoProd" está presente nas duas estruturas. Isso ocorre porque, ao inserirmos o preço do produto como item faturado (estrutura "item nota fiscal"), este não poderá sofrer alterações, portanto é estático. Já o preço do produto pode sofrer alterações ao longo do tempo (estrutura "produto"), portanto é dinâmico.

Agora, observe como ficam os dados armazenados após o redesenho da estrutura (figura 24).

Tabela "item nota fiscal"

l'abela "item nota fiscal				
Numero	CodProd	QtdeProd	PrecoProd	PrecoTotal
001234	123	1	0,80	0,80
001234	234	1	2,60	2,60
001234	345	2	1,50	3,00

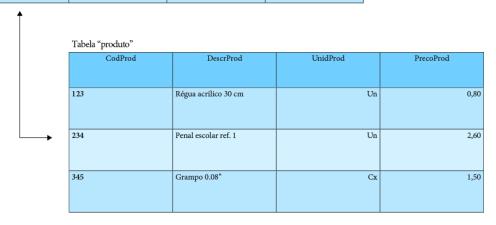


Figura 24



(i) Reflita

A aplicação da 2ª forma normal possibilitará a economia de espaço (estrutura "item nota fiscal") e a reutilização de dados já armazenados (estrutura "produto"). Esse é o principal reflexo da aplicação da 2ª FN.

A aplicação da 3ª FN é consolidada a partir do momento em que esteja na 2ª FN e que não apresente dependência funcional transitiva entre seus atributos. No exemplo temos duas situações que ilustram esta dependência funcional. O primeiro envolve os atributos que representam o "cliente". Observe que há atributos (NomeCliente, EndCliente) que dependem se complementam para dar sentido ao "cliente". Há também, um atributo que representa o "vededor". Mesmo este estando isolado no exemplo, deve ser considerado para a aplicação da 3º FN. Esta afirmação ocorre pelo fato de haver, naturalmente, outros atributos associados ao "vendedor", porém não presentes no cenário.



(i) Dica

Dependência funcional transitiva é a situação em que um atributo depende de outro, e este segundo depende de um terceiro.

Assim, a aplicação da 3ª FN prevê que separaremos subconjuntos insertos em um superconjunto e dessa forma podemos evitar redundância nas informações. No exemplo, a redundância estaria na necessidade de cadastrarmos, repetidas vezes, os dados do "cliente" e do "vendedor", no caso de os mesmos estarem presentes e mais de uma "nota fiscal"

Vamos identificar, na estrutura ajustada após a aplicação da 2ª FN onde ocorre a dependência funcional transitiva e/ou o atributo que possibilite a redundância nas informações, a partir das seguintes estruturas:

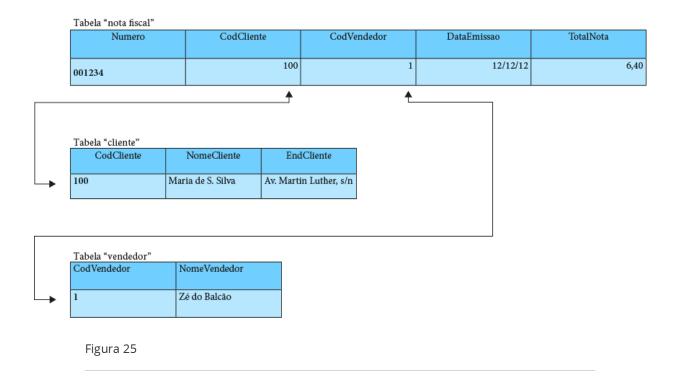
Tabela "nota fiscal":

Numero,

NomeCliente, EndClient	e, NomeVendedor,	DataEmissao,	, TotalNota
------------------------	------------------	--------------	-------------

Ao aplicar a 3ª FN, teremos a criação de novas estruturas com vemos:	o "cliente" e "vendedor", conforme
Tabela "nota fiscal": CodCliente, CodVendedor, DataEmissao, TotalNota	Numero,
Tabela "cliente": NomeCliente, EndCliente	CodCliente,
Tabela "vendedor": NomeVendedor	CodVendedor,

É importante observar que no exemplo não há um identificador (chave) para servir de ligação entre a estrutura "nota fiscal" e as novas criadas (cliente e vendedor). Por esta razão é que foram criados atributos e esses serviram de ligação (chave). Esta é diferença entre a aplicação da 2ª FN e a 3ª FN. Vamos observar como ficam os dados armazenados após o redesenho da estrutura (figura 25):



Após a aplicação das três primeiras e principais FN, observe como ficou o redesenho da estrutura considerando o conjunto de dados inicial (figura 26).

Tabela "nota fiscal"

Numero	CodCliente	CodVendedor	DataEmissao	TotalNota
001234	100	1	12/12/12	6,40

Tabela "cliente"

CodCliente	NomeCliente	EndCliente
100	Maria de S. Silva	Av. Martin Luther, s/n

Tabela "vendedor"

CodVendedor	NomeVendedor
1	Zé do Balcão

Tabela "item nota fiscal"

Tabela I	Tabela Relii liota liseai						
N	Jumero	CodProd	QtdeProd	PrecoProd	Preco Total		
0	001234	123	1	0,80	0,80		
0	001234	234	1	2,60	2,60		
O	001234	345	2	1,50	3,00		

Tabela "produto"

	a produto	D D 1	II D 1	D D I
CodP	rod	DescrProd	UnProd	PrecoProd
123		Régua acrílico 30 cm	Un	0,80
234		Penal escolar ref. 1	Un	2,60
345		Grampo 0.08"	Cx	1,50

				\sim	-
-	ıσ	ΙII	- 2	2	h
	جرا	uı	а	_	v

Encerramento

Você teve a oportunidade de conhecer as três primeiras FNs. Como já dito, essas são as mais importantes e aplicáveis. Porém, é necessário registrar que a literatura apresenta a 4ª e a 5ª FN, além da forma normal de Boyce Codd (FNBC). A aplicação destas é muito restrita e, portanto, desconsiderada no campo profissional. Mesmo assim, caso deseje saber mais sobre elas, recomendo a leitura da obra

de Chen (1990).

Como você pôde observar, a normalização é fundamental para a modelagem de dados relacional. Contudo, é importante que você faça uso do senso crítico para avaliar as circunstâncias pelas quais um atributo ou conjunto de atributos (ou até mesmo uma tabela) deve ou não ser segmentado através da normalização.

Lembre-se de que a normalização tem inúmeras vantagens, porém a longo prazo poderá requerer mais recursos computacionais para atender as requisições do sistema, pois o crescimento dos dados em um banco de dados é exponencial. Mas isso não deve ser, para você, ao menos neste momento, objeto de ponderação e preocupação, visto que tiraria o foco do tema proposto. Oportunamente você terá a chance de conhecer uma técnica já discutida e defendida por alguns profissionais, denominada "desnormalização". Convido você a avançarmos agora para o projeto físico de banco de dados. Trata-se da etapa em que iremos escolher qual o SGBD Relacional que iremos utilizar.

CONTINUE

Referências

Referências

ALVES, William Pereira. Banco de dados. São Paulo: Erica, 2014

CARDOSO, Vírginia M. Sistemas de banco de dados. São Paulo: Saraiva, 2008.

CHEN, Peter. Gerenciando banco de dados: a abordagem entidade-relacionamento para projeto lógico. São Paulo, McGraw-Hill, 1990.

HEUSER, Carlos Alberto. Projeto de banco de dados. Porto Alegre: Bookman, 2011.

MACHADO, Felipe Nery Rodrigues. Banco de dados: projeto e implementação.3. São Paulo: Erica, 2014.

Referências de Imagens

Divisão de Modalidades de ensino (DME), Fundação Universidade Regional de Blumenau (FURB), 2019a - o.