

# Modelagem de Dados em Microsoft Access 95

---

IDEOLÓGICA PRODUTIVIDADE EMPRESARIAL E INFORMÁTICA

# **Modelagem de Dados em Microsoft Access 95**

---

© Ideológica Produtividade Empresarial Informática  
R. Iguatemi, 252, 9º Andar  
Telefone (011) 883-3103 • Fax (011) 853-3259

---

# Índice

## Analítico

|   |          |
|---|----------|
| <b>Introdução à modelagem de dados</b>        | <b>1</b> |
| Por que Projetar?                             | 2        |
| Problemas Resultantes de um Projeto Pobre     | 2        |
| Introdução ao Projeto de Banco de Dados       | 3        |
| Modelo Domínio/Chave                          | 5        |
| O Processo de Normalização                    | 7        |
| <b>Um Método de Projeto de Banco de Dados</b> | <b>8</b> |
| <b>Projeto do Modelo do Banco de Dados</b>    | <b>9</b> |

# Introdução à modelagem de dados

## Capítulo

# 1

### *Aprenda os conceitos básicos de normalização*

Um projeto de banco de dados é uma matéria complexa, não importando como algumas pessoas achem que seja fácil. Esta sessão apenas arranha a superfície, mas com um belo arranhão.

Um banco de dados projetado de forma conveniente é um modelo de uma empresa, ou de alguma outra “coisa” no mundo real. Como seu modelo físico, em contrapartida, um modelo de dados permite a você fazer perguntas sobre os fatos que compõem os objetivos a serem alcançados. Estas são as perguntas que precisam de respostas e que determinarão quais fatores precisarão serem armazenados no modelo de dados.

No modelo relacional, os dados são organizados em tabelas que possuem as seguintes características:

- Todo registro tem o mesmo número de fatos;
- Todo campo contém o mesmo tipo de fato em cada um dos registros;
- Há apenas um ingresso para cada fato;
- Dois registros nunca são exatamente os mesmos;
- A ordem dos registros e campos não é importante.

Ao final desta leitura, você terá a compreensão básica dos problemas resultantes de um projeto pobre de banco de dados, estará familiarizado com o Modelo Domínio/Chave, compreenderá o processo para se projetar um banco de dados relacional, e saberá sobre as ferramentas usadas no Microsoft Access para suportar integridade coagindo num banco de dados.

---

## Por que Projetar?

Um projeto preciso é crucial para a operação de um sistema de informações seguro e eficiente. A tecnologia dos microcomputadores é atualmente tão avançada que o impacto de um projeto pobre pode não se mostrar tão cedo quanto no passado; todavia, quando os problemas aparecerem, eles serão severos.

Um projeto de um banco de dados tem que fazer com que o caminho dos dados seja armazenado e mostrar como os dados serão relatados. Os processos do projeto são desenvolvidos depois de você determinar exatamente quais informações precisam ser armazenadas e como elas serão recuperadas.

Quanto mais cuidadoso seu projeto, tanto melhor o banco de dados físico se identificará com as necessidades do usuário. No processo de desenvolvimento de um sistema completo, você precisa considerar as necessidades do usuário de vários pontos de vista.

## Problemas Resultantes de um Projeto Pobre

Diversos problemas podem se manifestar como resultado de um banco de dados mal projetado:

- O banco de dados e/ou aplicação não podem funcionar adequadamente.
- Os dados podem não ser confiáveis ou serão inexatos.
- A performance pode ser degradada.
- A flexibilidade poderá ser perdida.

A seção seguinte explica sobre alguns dos problemas mais comuns resultantes de um projeto de banco de dados pobre. Os problemas podem ser agrupados sob duas categorias: dados redundantes e modificações anômalas.

---

## Introdução ao Projeto de Banco de Dados

Considere a tabela seguinte que armazena dados sobre produtos e fornecedores. Esta aparentemente inofensiva tabela contém muitos problemas potenciais.

| IDProduto | Descrição              | Fornecedor               | Endereço                     | Cidade        | Região | País      |
|-----------|------------------------|--------------------------|------------------------------|---------------|--------|-----------|
| 34        | Sasquatch Ale          | Bigfoot Breweries        | 3400 - 8th Avenue, Suite 210 | Bend          | OR     | USA       |
| 27        | Schoggi Schokolade     | Heli Süßwaren GmbH       | Tiergartenstraße 5           | Berlin        |        | Germany   |
| 68        | Scottish Longbreads    | Specialty Biscuits, Ltd. | 29 King's Way                | Manchester    |        | UK        |
| 42        | Singaporean Fried Mee  | Leka Trading             | 471 Serangoon Loop,          | Singapore     |        | Singapore |
| 20        | Sir Rodney's Marmalade | Specialty Biscuits, Ltd. | 29 King's Way                | Manchester    |        | UK        |
| 21        | Sir Rodney's Scones    | Specialty Biscuits, Ltd. | 29 King's Way                | Manchester    |        | UK        |
| 61        | Sirop d'érable         | Forêts d'érables         | 148 rue Chasseur             | Ste-Hyacinthe | Québec | Canada    |
| 46        | Spegesild              | Lyngbysild               | Lyngbysild Fiskebakken 10    | Lyngby        |        | Denmark   |
| 35        | Steeleye Stout         | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210  | Bend          | OR     | USA       |

Suponhamos que você deseje adicionar outro registro

|    |                   |                   |                             |      |    |     |
|----|-------------------|-------------------|-----------------------------|------|----|-----|
| 37 | Lumberman's Lager | Bigfoot Breweries | 3400 - 8th Avenue Suite 210 | Bend | OR | USA |
|----|-------------------|-------------------|-----------------------------|------|----|-----|

O espaço em disco é perdido por duplicação dos dados sobre o fornecedor. Toda vez que um novo produto é registrado para um fornecedor em particular, todos os dados sobre este fornecedor tem de ser repetidos. Imagine o problema se diversos fornecedores fornecem centenas de produtos cada um.

### **Modificações Anômalas**

O que acontece se o fornecedor Bigfoot Breweries se muda para Portland? Quantos campos terão de ser modificados para se assegurar que o novo endereço foi registrado?

| IDProduto | Descrição           | Fornecedor               | Endereço                    | Cidade     | Região | País    |
|-----------|---------------------|--------------------------|-----------------------------|------------|--------|---------|
| 34        | Sasquatch Ale       | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend       | OR     | USA     |
| 27        | Schoggi Schokolade  | Heli Süßwaren GmbH       | Tiergartenstraße 5          | Berlin     |        | Germany |
| 68        | Scottish Longbreads | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester |        | UK      |

|    |                        |                          |                             |               |        |           |
|----|------------------------|--------------------------|-----------------------------|---------------|--------|-----------|
| 42 | Singaporean Fried Mee  | Leka Trading             | 471 Serangoon Loop,         | Singapore     |        | Singapore |
| 37 | Lumberman's Lager      | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |
| 20 | Sir Rodney's Marmalade | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 21 | Sir Rodney's Scones    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 61 | Sirop d'érable         | Forêts d'érables         | 148 rue Chasseur            | Ste-Hyacinthe | Québec | Canada    |
| 46 | Spegesild              | Lyngbysild               | Lyngbysild Fiskebakken 10   | Lyngby        |        | Denmark   |
| 35 | Steeleye Stout         | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |

Novamente, imagine o que pode resultar da modificação de centenas de campos de dados para apenas um fornecedor. Quando modificações são feitas, elas deverão abranger todas as cópias dos dados. Pense a respeito da confusão que resulta de modificar apenas um subgrupo de um dado duplicado.

### **Exclusões Anômalas**

Suponhamos que você não trabalhou por muito tempo com o produto 42 e decidiu eliminar esta linha de dados de sua tabela?

| IDProduto | Descrição              | Fornecedor               | Endereço                    | Cidade        | Região | País      |
|-----------|------------------------|--------------------------|-----------------------------|---------------|--------|-----------|
| 34        | Sasquatch Ale          | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |
| 27        | Schoggi Schokolade     | Heli Süßwaren GmbH       | Tiergartenstraße 5          | Berlin        |        | Germany   |
| 68        | Scottish Longbreads    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 42        | Singaporean Fried Mee  | Leka Trading             | 471 Serangoon Loop,         | Singapore     |        | Singapore |
| 20        | Sir Rodney's Marmalade | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 21        | Sir Rodney's Scones    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 61        | Sirop d'érable         | Forêts d'érables         | 148 rue Chasseur            | Ste-Hyacinthe | Québec | Canada    |
| 46        | Spegesild              | Lyngbysild               | Lyngbysild Fiskebakken 10   | Lyngby        |        | Denmark   |
| 35        | Steeleye Stout         | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |

Agora, observando os dados restantes, onde está o endereço do fornecedor Leka Trading?



| IDProduto | Descrição              | Fornecedor               | Endereço                    | Cidade        | Região | País    |
|-----------|------------------------|--------------------------|-----------------------------|---------------|--------|---------|
| 34        | Sasquatch Ale          | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA     |
| 27        | Schoggi Schokolade     | Heli Süßwaren GmbH       | Tiergartenstraße 5          | Berlin        |        | Germany |
| 68        | Scottish Longbreads    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK      |
| 20        | Sir Rodney's Marmalade | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK      |
| 21        | Sir Rodney's Scones    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK      |
| 61        | Sirop d'érable         | Forêts d'érables         | 148 rue Chasseur            | Ste-Hyacinthe | Québec | Canada  |
| 46        | Spegesild              | Lyngbysild               | Lyngbysild Fiskebakken 10   | Lyngby        |        | Denmark |
| 35        | Steeleye Stout         | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA     |

Uma exclusão anômala faz com que percamos mais informações do que o necessário. Nós perdemos dados sobre mais de um assunto a cada exclusão.

### *Inserção Anômala*

Agora você precisa adicionar um novo fornecedor, StarStruck, mas você ainda não tem encomendado nenhum produto deste fornecedor. O que você adicionará?

| IDProduto | Descrição              | Fornecedor               | Endereço                    | Cidade        | Região | País      |
|-----------|------------------------|--------------------------|-----------------------------|---------------|--------|-----------|
| 34        | Sasquatch Ale          | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |
| 27        | Schoggi Schokolade     | Heli Süßwaren GmbH       | Tiergartenstraße 5          | Berlin        |        | Germany   |
| 68        | Scottish Longbreads    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 42        | Singaporean Fried Mee  | Leka Trading             | 471 Serangoon Loop,         | Singapore     |        | Singapore |
| 20        | Sir Rodney's Marmalade | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 21        | Sir Rodney's Scones    | Specialty Biscuits, Ltd. | 29 King's Way               | Manchester    |        | UK        |
| 61        | Sirop d'érable         | Forêts d'érables         | 148 rue Chasseur            | Ste-Hyacinthe | Québec | Canada    |
| 46        | Spegesild              | Lyngbysild               | Lyngbysild Fiskebakken 10   | Lyngby        |        | Denmark   |
| 35        | Steeleye Stout         | Bigfoot Breweries        | 3400 - 8th Avenue Suite 210 | Bend          | OR     | USA       |
| ???       | ??????                 | StarStruck, Inc.         | 101 Mariposa                | Seattle       | WA     | USA       |

Esta situação é chamada de inserção anômala. Não podemos adicionar um dado sobre um assunto até que nós tenhamos dados adicionais sobre outro assunto.

## Modelo Domínio/Chave

Teorias relacionais têm classificado os esquemas de banco de dados que têm inconsistências, baseando-se nas anomalias a qual eles são suscetíveis . Você pode ter encontrado discussões sobre diferentes formas. Uma das únicas normalizações foi propostas por R. Fagin, em 1981, e é usada como a base desta apresentação. Fagin apurou que se as tabelas em seu banco de dados estão no Modelo Domínio/Chave, então eles estão livres de modificações anormais .

Para compreender este modelo aí estão quatro termos que devem ser compreendidos: dependência, chave, domínio e restrição .

### ***Dependência***

Uma dependência é uma relação que pode existir entre duas colunas . Dados os valores da primeira coluna , você estará apto para determinar o valor de uma outra. Vamos usar a tabela dos exemplos anteriores. Dados o número do produto, nós estaremos capazes de determinar as descrições dos produtos. Esta é uma dependência: descrições são dependências nos números dos produtos. Dados um nome de fornecedor, seremos nós capazes de determinar a descrição do produto? Não necessariamente . No caso do BigFoot Breweries, este fornecedor tem um número de produtos associados a ele. Portanto , nas tabelas acima, descrição não é uma dependência de fornecedores.

Para detectar uma dependência, pergunte a si mesmo o seguinte:

Nesta tabela, pode o valor de uma coluna determinar TODOS OS VALORES POSSÍVEIS de outra coluna?

- |              |                      |     |
|--------------|----------------------|-----|
| • IDProduto  | determina Descrição? | SIM |
| • Fornecedor | determina Endereço?  | SIM |
| • Fornecedor | determina IDProduto? | NÃO |

### ***Chave***

A maior parte das tabelas deve ter uma coluna ou uma combinação de colunas que unicamente identifique uma linha de dados. Uma coluna é chave se todas as outras colunas numa linha de dados são dependentes dela.

A primeira olhada, pode parecer que a coluna IDProduto em nosso exemplo unicamente identifica uma linha de dados. Mas IDProduto 34 identifica o fornecedor como BigFoot Breweries, que também faz parte

---

dos números 35 e 37. Portanto, a coluna *IDProduto* não é uma chave. Nesta tabela, nós temos uma chave complexa, derivada de IDProduto, Descrição, e Fornecedor.

---

## **Domínio**

Um domínio é o conjunto de valores que uma coluna pode ter. Toda coluna tem um domínio que tem, por sua vez, as propriedades lógicas e físicas.

- **Descrição Física.** A parte física do domínio é o tipo de informação sobre cada coluna. Em nosso exemplo, Fornecedor é definido como TEXTO 40. A partir desta definição, a descrição física do domínio é o conjunto de valores de dados de TEXTO com 40 caracteres ou menos. Similarmente, a descrição física para o domínio de IDProduto é expressa como INTEIRO. Isto resulta em dados com nove números ou menos.
- **Descrição Lógica.** A parte lógica do domínio é o conjunto de informações associadas com os fatos. Os endereços dos fornecedores não estão no mesmo domínio dos endereços dos clientes, apesar deles terem a mesma propriedade física de TEXTO 60.

Considere o valor 7124 E. 41st Place. Este é um valor no domínio de endereços de fornecedores? Para estar neste domínio ele precisa ter menos de 60 caracteres e ser um endereço de fornecedor.

## **Restrição**

Uma restrição é uma limitação de algum tipo nos valores da tabela. Uma dependência é um tipo de restrição. Especificar que a Descrição é dependente de IDProduto é uma restrição. Chaves são um tipo de restrição. Quando uma coluna é uma chave, significa que todas as outras colunas na tabela são dependentes dela. Lembre-se que uma chave pode ser uma combinação de colunas.

Um domínio é outro tipo de restrição. Quando definimos as propriedades lógica e física de uma coluna, nós restringimos os dados que ela poderá armazenar.

Restrição é um termo geral. Existem muitas maneiras de restringir os dados numa tabela. Abaixo estão alguns exemplos:

- Especificar que as datas deverão ser formatadas como DD/MM/AA.
  - IDProduto precisa iniciar a partir do número 100.
  - Fornecedores precisam ser TEXTO com 40 caracteres ou menos.
-

- Taxa Total precisa ser MOEDA com valores entre \$1,00 e \$9.999.999,99.

---

## O Processo de Normalização

Normalizar o banco de dados garante que a estrutura permitirá as mudanças a serem feitas sem que ocorram conseqüências inesperadas. O papel da normalização é manter estável, e com dados confiáveis, o banco de dados, através de um bom projeto.

O objetivo de um bom projeto de banco de dados é assegurar que todas as restrições sejam conseqüências lógicas das restrições de domínio e chave.

Tabelas, como parágrafos, devem ter um só tema. A tabela nos exemplos de anomalias teve dois temas:

- Informações sobre os produtos.
- Informações sobre os fornecedores dos produtos.

A maneira de administrar esta informação mais eficientemente é dividir a tabela em duas: uma tabela de produtos e uma tabela de fornecedores.

### *Produtos*

| ID Produto | Descrição                     | Fornecedor                  |
|------------|-------------------------------|-----------------------------|
| 34         | Sasquatch Ale                 | Bigfoot Breweries           |
| 27         | Schoggi Schokolade            | Heli Süßwaren GmbH & Co. KG |
| 68         | Scottish Longbreads           | Specialty Biscuits, Ltd.    |
| 42         | Singaporean Hokkien Fried Mee | Leka Trading                |
| 20         | Sir Rodney's Marmalade        | Specialty Biscuits, Ltd.    |
| 21         | Sir Rodney's Scones           | Specialty Biscuits, Ltd.    |
| 61         | Siroop d'érable               | Forêts d'érables            |
| 46         | Spegesild                     | Lyngbysild                  |
| 35         | Steeleye Stout                | Bigfoot Breweries           |

### *Fornecedores*

| Fornecedor                  | Endereço                       | Cidade        | Região | País      |
|-----------------------------|--------------------------------|---------------|--------|-----------|
| Bigfoot Breweries           | 3400 - 8th Avenue Suite 210    | Bend          | OR     | USA       |
| Heli Süßwaren GmbH & Co. KG | Tiergartenstraße 5             | Berlin        |        | Germany   |
| Specialty Biscuits, Ltd.    | 29 King's Way                  | Manchester    |        | UK        |
| Leka Trading                | 471 Serangoon Loop, Suite #402 | Singapore     |        | Singapore |
| Forêts d'érables            | 148 rue Chasseur               | Ste-Hyacinthe | Québec | Canada    |

Agora você pode adicionar produtos sem duplicações, modificar a localização de fornecedores sem alterar várias linhas de dados, e não perderá informações se você excluir uma parte realmente não necessária.

Se você desejar, poderá sempre trazer de volta a tabela original usando uma pergunta em associação a Fornecedor.

## Um Método de Projeto de Banco de Dados

Assim como você viu, o projeto de banco de dados desempenha o maior papel na estabilidade e confiança de seus dados. Nesta seção, nós mostraremos a você o processo de projetar um banco de dados. Para ajudar a ilustrar este processo, um banco de dados chamado Zipper será criado para um fictício atacadista e fabricante de roupas chamado Zipper.

Apesar de existirem várias regras que deverão ser seguidas no projeto da estrutura do banco de dados, o processo de projeto é muito mais uma arte que uma ciência. Siga estas regras sempre que possível, mas não até o ponto de perder a funcionalidade que é tão importante ao usuário.

Fazer o projeto do banco de dados primeiramente no papel traz diversas vantagens:

- Economiza dinheiro, tempo e sanidade;
- Torna o sistema mais seguro; evita potenciais problemas com modificações de dados;
- Serve de planta para discussões;
- Ajuda a estimar custos e tamanho.

Um bom projeto deverá ter os seguintes objetivos:

---

- Ir ao encontro a necessidade dos usuários
- Solucionar o problema
- Ser livre de modificações anômalas
- Ter um confiável e estável banco de dados, onde as tabelas sejam tão independentes quanto possível
- Ser fácil de usar



# Projeto do Modelo do Banco de Dados

O projeto da estrutura do banco de dados necessita dos seguintes passos:

1. Relacione os objetos.
  2. Liste os fatores relacionados aos objetos.
  3. Transforme os objetos e fatores em tabelas e colunas.
  4. Determine as relações entre os objetos.
  5. Determine as chaves das colunas.
  6. Determine as colunas relacionadas.
  7. Determine as relações de reserva.
  8. Avalie o modelo projetado.
  9. Desenvolva o banco de dados.
-

### ***Passo 1: Relacione os objetos***

Faça uma lista de todos os objetos. Um objeto é um tema único, similar a um parágrafo. Na Zipper os objetos são:

- Cliente
- Produto
- Funcionário
- Taxa de Embarque
- Fatura
- Dependente

### ***Passo 2: Liste os fatores relacionados aos objetos***

Aqui está o grande negócio da informação associada a cada objeto. Neste passo, você deve listar os fatos sobre cada objeto e então eliminar os fatos que não são importantes para a solução do problema. O objeto cliente, por exemplo, pode ter muitos fatos associados a ele: nome da companhia, endereço, cidade, fundadores, número de funcionários, preço das ações. Neste caso, não é importante manter informações sobre o número de funcionários, preço das ações, ou fundadores. A Zipper precisa somente das informações que irá usar agora ou que possivelmente irá usar no futuro.

| <b>Objeto</b>    | <b>Fatores Importantes Sobre o Objeto</b>                                 |
|------------------|---|
| Funcionário      | funcionário, nome, data de nascimento, sexo, estado civil                 |
| Cliente          | nome da companhia, endereço, cidade, estado, CEP, contato                 |
| Fatura           | data, vendedor, cliente, quantidade, despesa de transporte, taxas, fretes |
| Produto          | nome do produto, descrição, custo, remarcação de preços                   |
| Dependente       | nome, data de nascimento  |
| Taxa de Embarque | estado , taxas  |

### ***Passo 3: Transforme os objetos e fatores em tabelas e colunas***

Objetos automaticamente irão tornar-se tabelas, e fatos irão tornar-se colunas assim que seus domínios sejam determinados. Recorde que um domínio é um conjunto de valores que a coluna pode ter. Toda coluna tem um domínio, que tem propriedades lógica e física. Por exemplo, a

coluna para “último nome” dos funcionários é definida como TEXTO 15. TEXTO 15 é a propriedade física da coluna. A partir desta definição, o domínio é o conjunto de todos os “últimos nomes” de todos os empregados com 15 caracteres ou menos.

Se uma coluna é usada para conectar duas ou mais tabelas, o domínio precisa ser o mesmo e as colunas devem ter o mesmo nome. Se a descrição lógica diferir (por exemplo, “último nome” do funcionário e “último nome” do cliente), as colunas não são as mesmas e não devem partilhar o mesmo nome. A seguir se encontra uma lista das tabelas preliminares, colunas e domínios para a Zipper:

Tabela:  
CLIENTE

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| COMPANY     | TEXTO       | 45           |
| CADD1       | TEXTO       | 30           |
| CADD2       | TEXTO       | 30           |
| CCITY       | TEXTO       | 25           |
| CSTATE      | TEXTO       | 2            |
| CZIP        | TEXTO       | 10           |
| CAC         | TEXTO       | 3            |
| CTELPH      | TEXTO       | 7            |
| CONTACT     | TEXTO       | 30           |
| TITLE       | TEXTO       | 30           |

Tabela:  
PRODUTO

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| PRODNAME    | TEXTO       | 30           |
| PRODDISC    | TEXTO       | 50           |
| PRODCOST    | MOEDA       |              |
| PMARKUP     | NÚM         |              |

*Tabela: DEPENDENTE*

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| DLAST       | TEXTO       | 15           |
| DFIRST      | TEXTO       | 10           |
| DDOB        | D/H         |              |

Tabela: FATURA

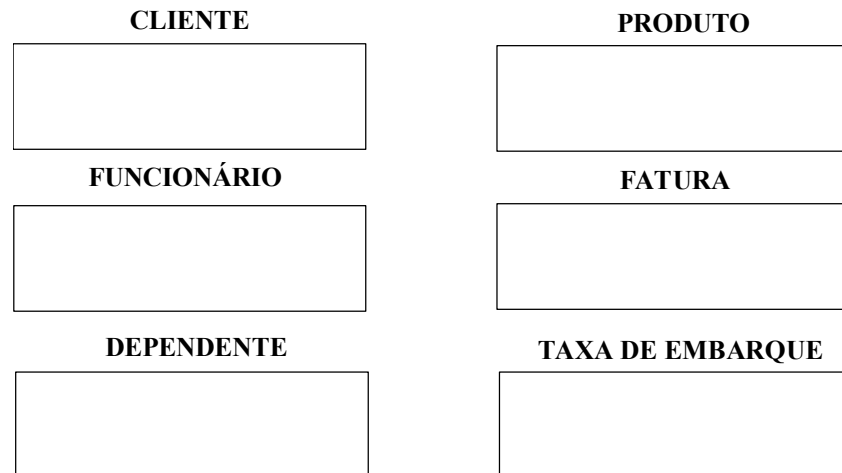
| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| INVDATE     | D/H         |              |
| REQDATE     | D/H         |              |
| SHIPNAME    | TEXTO       | 45           |
| SHIPADDR    | TEXTO       | 30           |
| SHIPCITY    | TEXTO       | 25           |
| SHIPZIP     | TEXTO       | 10           |
| INVTOTAL    | MOEDA       |              |

Tabela: FUNCIONÁRIO

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| ESSN        | TEXTO       | 11           |
| ELASTN      | TEXTO       | 15           |
| EFIRSTN     | TEXTO       | 10           |
| EDOB        | D/H         |              |
| EGENDER     | TEXTO       | 1            |
| EMARITAL    | TEXTO       | 1            |
| EADDR1      | TEXTO       | 30           |

| Tabela: TAXA DE EMBARQUE |       |       | EADDR2  | TEXTO | 20 |
|--------------------------|-------|-------|---------|-------|----|
| Nome                     | Tipo  | Compr | ECITY   | TEXTO | 25 |
| SHIPST                   | TEXTO | 2     | ESTATE  | TEXTO | 2  |
| SHIPRATE                 | NÚM   |       | EZIP    | TEXTO | 10 |
|                          |       |       | EAC     | TEXTO | 3  |
|                          |       |       | EHOMEPH | TEXTO | 7  |

Freqüentemente uma ajuda no estágio de projeto é desenhar caixas representando as tabelas. Em passos posteriores você poderá então preencher as colunas chave e desenhar as relações entre as tabelas.



#### ***Passo 4: Determine as relações entre objetos***

Para determinar as relações entre os objetos, pegue um objeto e olhe como este objeto pode se relacionar com outro. Mantenha em mente que nem toda relação existente entre objetos é importante. As relações que são importantes são aquelas que permitem a você modelar o banco de dados de maneira a corresponder às situações do mundo real que ele representa.

**Relações um-para-um.** Para qualquer linha de dados na tabela A, existe uma única linha na tabela B. Para qualquer linha de dados na tabela B, existe uma única linha na tabela A. Não existe nenhum relacionamento um-para-um no banco de dados da Zipper. Um exemplo

de uma relação um-para-um é aquela que abrange dados sobre o funcionário e dados pessoais sobre o funcionário. Informações gerais, como nome do funcionário, endereço, data de contratação, são mantidas em uma tabela, e para manter privacidade, informações pessoais, como salário, são mantidas em outra tabela.

**Relações um-para-vários.** Para qualquer linha de dados na tabela A, existem várias linhas na tabela B. Para qualquer linha de dados na tabela B, existe uma única linha na tabela A. As relações entre funcionários e seus dependentes é um-para-vários, porque um funcionário pode ter muitos dependentes, mas um dependente é relacionado a apenas um funcionário. A relação entre clientes e faturas é também um-para-vários. Uma fatura é relacionada a um cliente, mas um cliente pode ter várias faturas.

---

**Relações vários-para-vários.** Para qualquer linha de dados na tabela A, existem várias linhas na tabela B. Para qualquer linha de dados na tabela B, existem várias linhas na tabela A. Existe uma relação vários-para-vários entre a tabela de produtos e a tabela de faturas. Um produto pode ser associado a diferentes faturas e uma fatura pode conter vários produtos diferentes.

No caso do banco de dados Zipper, nós estamos tentando modelar um ambiente que é baseado em transações de vendas. Pegue o exemplo de produtos e clientes: Apesar de em algumas circunstâncias nós podermos nos interessar nas relações entre clientes e produtos, nas transações de vendas, o cliente é relacionado a um produto somente quando uma venda ocorre. Portanto, um cliente é relacionado a uma fatura, e uma fatura carrega a relação a um produto.

O primeiro passo na determinação do tipo de relacionamento entre tabelas é listar todas as tabelas e verificar como cada uma se relaciona com todas as outras:

- *Cliente* é relacionado à *fatura*.
- *Cliente* não é relacionado a nenhuma outra tabela na lista.
- *Funcionário* é relacionado a *dependente*.
- *Funcionário* (vendas) é relacionado a *fatura*.
- *Produto* é relacionado a *fatura*.

Um método eficaz de encontrar o tipo de relacionamento é perguntar se um determinado registro na tabela A pode apontar para (é relacionado a) uma ou mais linhas de dados na tabela B, e em seguida inverter as tabelas e fazer a pergunta novamente.

- Um registro de cliente aponta para uma ou várias faturas?  
Várias
- Uma linha de dados de fatura se liga a um ou vários clientes? Um

A relação entre as tabelas é um-para-vários.

- Um funcionário de vendas registra uma ou várias faturas?  
Várias
- Uma fatura é registrada por um ou vários funcionários? Um

A relação entre funcionário e fatura também é um-para-vários.

---

- Um produto pode ser um item listado em uma ou várias faturas?  
Várias
- Uma fatura pode se relacionar a um ou vários produtos? Vários

A relação entre produto e fatura é vários-para-vários.

A tabela Taxa de Embarque ilustra a idéia de que uma tabela pode ser incluída num banco de dados não precisando de relacionamentos com qualquer outra tabela.

### ***Passo 5: Determine as chaves das colunas***

Uma chave pode ser um número de conta, número de seguridade social, número de licença, ou qualquer outro valor numérico ou combinação de caracteres que seja único. Uma chave complexa é aquela que se deriva de mais de uma coluna. O Microsoft Access suporta chaves complexas diretamente.

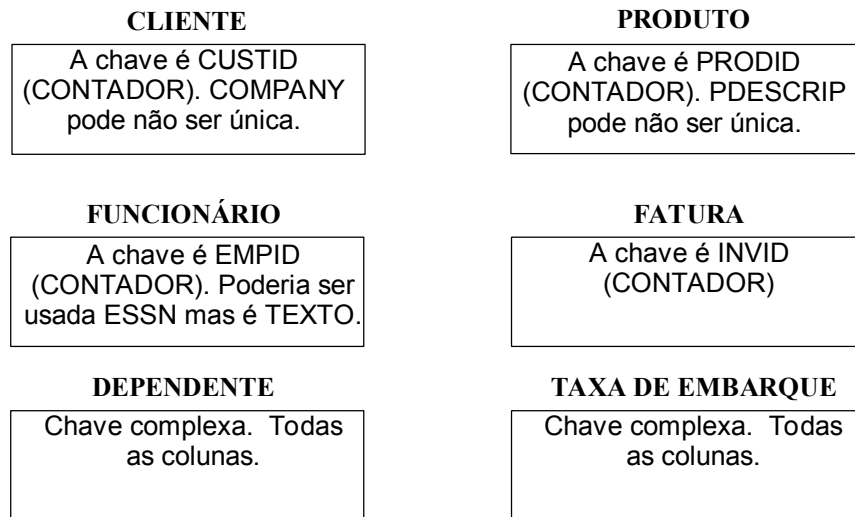
Nenhuma outra linha na tabela pode ter o valor da(s) coluna(s) chave. As outras tabelas podem compartilhar do mesmo conjunto de informações chave. Se o nome de uma companhia é universalmente único, ele é usado como linha única e identificadora. Porém, se existe a possibilidade de outra companhia ter o mesmo nome, então ele não é único e não deverá ser empregada como coluna chave. Não use qualquer coluna como chave se existe a possibilidade de duplicidade. Uma coluna chave não pode conter valores nulos.

Por definição, todas as colunas chave deverão ser indexadas.

Como normalmente os textos de nomes não são únicos e não podem ser usados em operações matemáticas, é costumeiro fazer com que as colunas chave assumam um valor numérico seqüencial. Em muitos casos, é mais fácil você desenvolver sua própria linha de dados, única e identificadora. Se você necessita de uma numeração automática para números de faturas ou números de funcionários, o tipo de dado CONTADOR no Microsoft Access é uma boa escolha para a descrição física do domínio da coluna chave.

A maior parte das tabelas ao final do projeto do banco de dados Zipper contém colunas com o tipo de dados CONTADOR ou NÚMERO para exercerem a função de linhas identificadoras e únicas. Cada chave é também indexada, e duplicatas não são permitidas. A performance do banco de dados é acentuada com uma simples coluna numérica como chave.

---



### ***Passo 6: Determine as colunas relacionadas***

Se você foi cuidadoso na determinação das colunas chave, também deve ter determinado as colunas relacionadas. Relações fornecem uma maneira de vincular informações (linhas de dados) entre tabelas. Se uma tabela tem uma coluna chave, esta coluna pode geralmente servir como elo num vínculo. Tabelas são vinculadas através de suas colunas chave. Porém, a colocação da chave é importante, e onde o elo será colocado dependerá do tipo de relação entre as tabelas.

Para determinar a colocação dos elos, você deverá inicialmente conhecer o tipo de relacionamento entre os objetos ou tabelas. Uma vez que você conheça o tipo de relacionamento entre tabelas, é muito mais fácil determinar onde colocar os elos para vinculá-las.

Lembre-se que nem todas as tabelas precisam ser relacionadas. Funcionários precisam ser vinculados a dependentes, mas você não gostaria de relacionar funcionários com taxas de embarque ou produtos.

**Vinculando numa relação um-para-um.** Numa relação um-para-um o elo deve ser a coluna mais estável ou deve vir da tabela onde a coluna chave foi criada. A coluna mais estável é aquela que possui a menor parcela de chances de que venha a sofrer modificações. Se algum sistema de numeração automática está sendo usado, então use esta coluna como elo.

**Vinculando numa relação um-para-vários.** Na relação um-para-vários a coluna elo deve vir da tabela “um”. A coluna chave da tabela funcionário (lado “um”) deve ser colocada na tabela dependente (lado



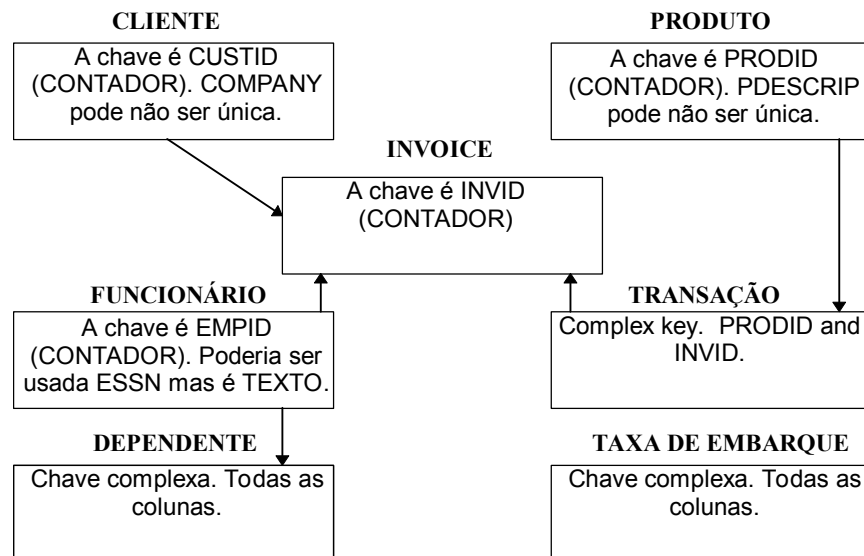
“vários”). Quando a chave empid é colocada na tabela dependente, ela é referida como uma chave estrangeira na tabela.

**Vinculando numa relação vários-para-vários.** A relação vários-para-vários causa problemas quando tenta recuperar dados e quando relata um valor em uma tabela para seu valor correspondente em outra. É importante compreender esta relação para ser capaz de reconhecer e controlar esta situação quando ela surgir.

Uma clássica relação vários-para-vários é a que existe entre produto e fatura. Um determinado produto pode ter um item citado em várias faturas diferentes e uma fatura pode ter muitos produtos associados a ela.

Mas qual chave iremos usar como elo? Se invid for colocado na tabela produto, então todos os dados sobre os produtos terão de ser repetidos para cada fatura que contém aquele produto. Se prodid for colocado na tabela fatura, as informações sobre a fatura deverão ser repetidas para cada produto contido na fatura. Isto leva a dados redundantes, e o potencial para dados inválidos aumenta. A performance pode ser avariada.

A solução para relações vários-para-vários é criar tabelas intermediárias. Esta tabela deve conter as colunas chave de ambas as tabelas principais. Isto é ilustrado pelo diagrama seguinte.



### ***Passo 7: Determine as relações de reserva***

Muitas vezes as informações que nós recuperamos de um banco de dados vem de mais de uma tabela. Por exemplo, se nós quisermos saber qual o pai ou mãe de um dependente em particular, o nome é determinado usando o valor de empid para procurar a linha com o valor correto na tabela funcionário. A questão de quem é o pai ou mãe poderá ser respondida somente se existir uma linha na tabela funcionários com um valor correspondente em empid da tabela dependente.

Para assegurar a integridade dos dados em nosso banco de dados, nosso modelo deve requerer, por exemplo, que nenhuma linha possa ser adicionada à tabela dependente, sem que exista linha correspondente na tabela funcionário. Este requisito é conhecido como relação de reserva. Neste caso, deve existir uma reserva na tabela dependente que assegure que o funcionário (pai ou mãe) exista.

Se você está criando uma fatura, você deve ter um cliente para mandar a conta. Um registro na tabela cliente deve existir antes que a fatura possa ser redigida. Neste caso, uma reserva deve existir na tabela fatura para assegurar que o cliente existe.

Aí estão, por menores que sejam, quatro métodos para implementar relações de reserva:

- Construção de controles em DBMS
- Entrada de dados e procedimentos de acesso
- Programação
- Implementação de regras

O Microsoft Access tem certos mecanismos de integridade referencial construídos em seus motores. Com o Microsoft FoxPro, a relação de reserva deverá ser criada a partir de programação.

No Microsoft Access, regras podem ser empregadas para impor o domínio de colunas (por exemplo, aceitar valores menores que 200, ou valores de texto deverão ser F ou M) ou em qualquer outra operação onde você necessite de um teste nos dados a serem registrados.

### ***Passo 8: Avalie o modelo projetado***

O próximo passo no processo de projeto é a avaliação do projeto. Neste passo, você deverá procurar por todas as falhas que podem vir a causar dados irre recuperáveis, instáveis ou redundantes.

---

Cada tabela deverá ser avaliada respondendo-se as seguintes questões:

1. Cada tabela possui um tema único? Assim deve ser. Cada coluna deverá ser um fato sobre a chave.
  2. Cada tabela tem sua(s) coluna(s) chave? Assim deve ser.
  3. Existem dependências? Somente conseqüências lógicas da chave devem existir.
  4. São os domínios únicos entre as tabelas? Não misture domínios a menos que a coluna seja comum entre tabelas.
  5. Existe domínio de restrição ou chave?
  6. A tabela é fácil de usar?
-

### *Avaliação da Tabela Cliente*

|        |         |
|--------|---------|
| CUSTID | COMPANY |
|        | CADD1   |
|        | CADD2   |
|        | CCITY   |
|        | CSTATE  |
|        | CZIP    |
|        | CAC     |
|        | CTELPH  |
|        | CONTACT |
|        | TITLE   |

- A tabela tem um único tema: clientes.
- A tabela tem uma chave: custid.
- A tabela não tem nenhuma dependência que não seja consequência lógica da chave. Dado custid, uma companhia e endereço da companhia podem ser determinados unicamente. Dada uma companhia, nós não podemos determinar nenhum custid em particular. Dada um estado, nós não podemos determinar nenhum custid em particular. Portanto, a tabela cliente não tem nenhuma dependência.
- O nome das colunas não é usado em nenhuma outra tabela, exceto custid que é uma chave estrangeira na tabela fatura.
- As restrições são domínio ou chave.

### ***Passo 9: Desenvolva o banco de dados***

Uma vez que o banco de dados tenha sido projetado no papel, o próximo passo é desenvolver o banco de dados no Microsoft Access. Quando definir tabelas no Microsoft Access, é extremamente importante manter seu projeto de papel em mente. Desenvolver um banco de dados sem parâmetros pode causar problemas que podem ser difíceis de desfazer. (Lembre-se das anomalias anteriormente descritas.)

No Microsoft Access 2.0, existem duas ferramentas que o ajudarão a completar a implementação de seu projeto. O Assistente de The Table Tabelas pode ser usado para gerar uma variedade de tabelas comuns. A janela de sistema gráfico de relações pode ser usada para criar

---

relacionamentos e dependências chave. Perceba que, enquanto usar o Assistente de Tabelas assegura um desenvolvimento relacional eficaz, os oito passos anteriores para implementação permanecem importantes e devem ser completados antes da construção das tabelas.

A seguir está uma lista das tabelas finais, colunas e domínios para a Zipper, incluindo colunas elo:

Tabela:  
CLIENTE

| Nome    | Tipo     | Compr |
|---------|----------|-------|
| CUSTID  | CONTADOR |       |
| COMPANY | TEXTO    | 45    |
| CADD1   | TEXTO    | 30    |
| CADD2   | TEXTO    | 30    |
| CCITY   | TEXTO    | 25    |
| CSTATE  | TEXTO    | 2     |
| CZIP    | TEXTO    | 10    |
| CAC     | TEXTO    | 3     |
| CTELPH  | TEXTO    | 7     |
| CONTACT | TEXTO    | 30    |
| TITLE   | TEXTO    | 30    |

Tabela:  
PRODUTO

| Nome     | Tipo     | Compr |
|----------|----------|-------|
| PRODID   | CONTADOR |       |
| PNAME    | TEXTO    | 30    |
| PDESCRIP | TEXTO    | 50    |
| PCOST    | MOEDA    |       |
| PMARKUP  | NÚM      |       |

Tabela: TAXA DE EMBARQUE

| Nome     | Tipo  | Compr |
|----------|-------|-------|
| SHIPST   | TEXTO | 2     |
| SHIPRATE | NÚM   |       |

Tabela:  
FATURA

| Nome     | Tipo     | Compr |
|----------|----------|-------|
| INVID    | CONTADOR |       |
| CUSTID   | NÚM      |       |
| INVDATA  | D/H      |       |
| REQDATE  | D/H      |       |
| SHIPNOME | TEXTO    | 45    |
| SHIPADDR | TEXTO    | 30    |
| SHIPCITY | TEXTO    | 25    |
| SHIPST   | TEXTO    | 2     |
| SHIPZIP  | TEXTO    | 10    |
| INVTOTAL | MOEDA    |       |

Tabela:  
TRANSAÇÃO

| Nome   | Tipo  | Compr |
|--------|-------|-------|
| INVID  | NÚM   |       |
| PRODID | NÚM   |       |
| TQTY   | NÚM   |       |
| TDISC  | NÚM   |       |
| TPRICE | MOEDA |       |

Tabela:  
FUNCIONÁRIO

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| EMPID       | CONTADOR    |              |
| ESSN        | TEXTO       | 11           |
| ELASTN      | TEXTO       | 15           |
| EFIRSTN     | TEXTO       | 10           |
| EDOB        | D/H         |              |
| EGENDER     | TEXTO       | 1            |
| 1EMARITAL   | TEXTO       | 1            |
| EADDR1      | TEXTO       | 30           |
| EADDR2      | TEXTO       | 20           |
| ECITY       | TEXTO       | 25           |
| ESTATE      | TEXTO       | 2            |
| EZIP        | TEXTO       | 10           |
| EAC         | TEXTO       | 3            |
| EHOMEPH     | TEXTO       | 7            |

Tabela:  
DEPENDENTE

| <b>Nome</b> | <b>Tipo</b> | <b>Compr</b> |
|-------------|-------------|--------------|
| EMPID       | NÚM         |              |
| DLAST       | TEXTO       | 15           |
| DFIRST      | TEXTO       | 10           |
| DDOB        | D/H         |              |