

Introdução

O objetivo do projeto final é chegar o mais próximo possível da realidade de um projeto de banco de dados. Um projeto real envolve detalhes próprios e específicos de cada empresa. Por isso, cada banco de dados é único. É impossível abranger tudo em um único curso, mas certamente o que você aprendeu aqui lhe dará elementos para modelar com segurança.

A maioria dos projetos de banco de dados segue um modelo básico de estrutura. Ela não é uma coisa rígida, que deve ser seguida ao pé da letra, e você pode usá-la como parâmetro para o seu projeto. A seguir, os tópicos do projeto:

O problema.

Modelo entidade relacionamento.

Modelo lógico textual.

Modelo lógico gráfico.

O problema

Com o aumento de sistemas informatizados hoje, qualquer empresa – seja de pequeno ou médio porte – é capaz de manter seu próprio sistema informatizado para tornar seu processo de trabalho mais rápido e automático. Com o objetivo de melhorar os processos de trabalho da loja DeTudo LTDA, o projeto que você modelará deverá conter as seguintes funcionalidades:

- Manter o histórico de todas as vendas efetuadas pela loja e seus itens.
- Manter o histórico de todas as compras realizadas e seus itens.
- Gravar a comissão dos vendedores sobre cada venda.
- Gerenciar a quantidade de produtos em estoque.
- Manter o cadastro dos seguintes elementos: Clientes, Fornecedores, Funcionários, Vendedores, Produtos e Categoria de produtos.
- No banco, o modelador deve garantir que sejam armazenados os seguintes dados:

1. Sobre os clientes:

– Pessoa física: nome, RG, CPF, endereço, e-mail e telefone;

– Em caso de pessoa jurídica: razão social, nome fantasia, CNPJ, endereço, e-mail e telefone.

2. Sobre o produto: código, nome, descrição, unidade de medida, valor de compra, valor de venda, quantidade mínima e máxima em estoque e sua categoria. Da categoria: código, nome e a descrição. Da unidade de medida: código, sigla, descrição.

3. Dos fornecedores: as mesmas informações que os clientes pessoa jurídica.

4. Sobre os funcionários: nome, CPF, RG, endereço, e-mail, telefone e número carteira de trabalho.

5. Os vendedores devem ter todas as informações dos funcionários, mais o valor da meta de venda mensal.

6. Toda venda deve conter os seguintes dados: produtos vendidos, valor unitário de cada item, valor total de venda, quantidade de cada item, a data, o cliente que realizou a compra, o vendedor e a comissão daquela venda.

7. Das compras não é muito diferente: produtos, a quantidade unitária de cada item, o valor de cada item, o valor total da compra e a data.

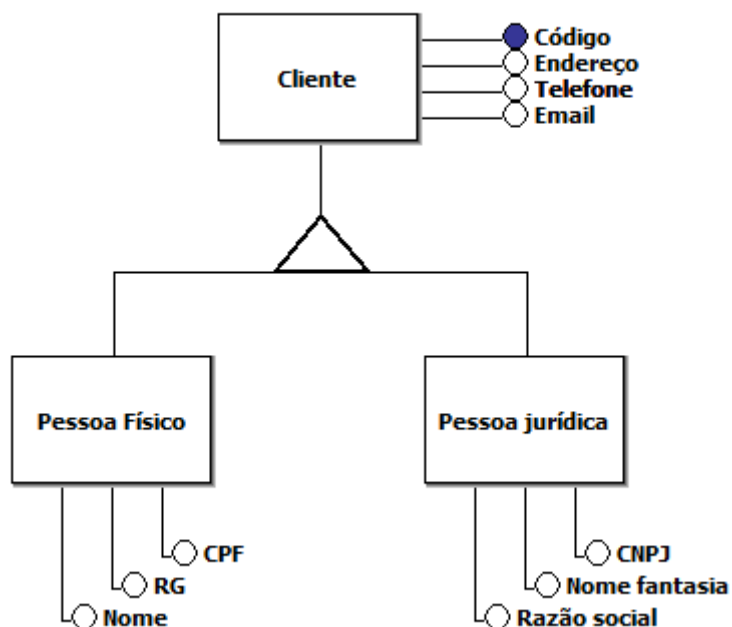
8. O estoque deve ter apenas a quantidade de produto e o produto.

Construindo o MER

O modelo entidade relacionamento deve ser construído no mesmo formato usado para o desenvolvimento dos exercícios no final dos tópicos sobre modelo conceitual. A esta altura do curso, não será mais demonstrado como inserir os elementos no BrModelo, pois isso já deve ser de domínio de todos os alunos.

Primeira parte

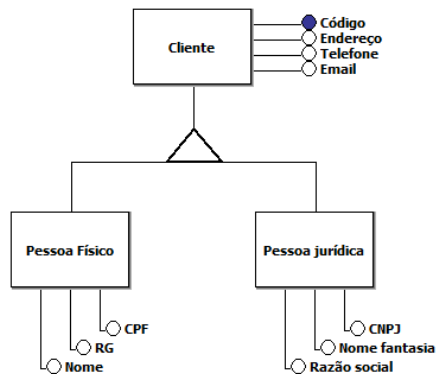
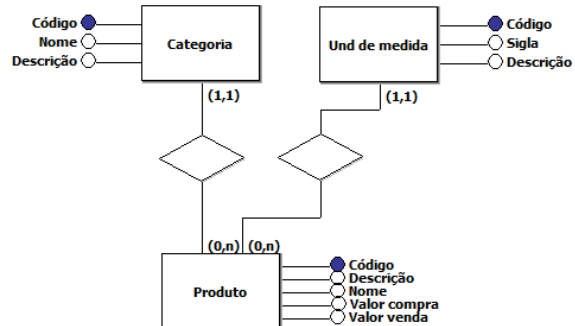
No banco o modelador deve garantir que sejam armazenados os seguintes dados: Sobre os clientes pessoa física – nome, RG, CPF, endereço, e-mail e telefone; em caso de pessoa jurídica – razão social, nome fantasia, CNPJ, endereço, e-mail e telefone.



Você deve criar uma estrutura de generalização/especialização do tipo exclusiva/total. Na entidade genérica coloque os atributos comuns, tanto para clientes pessoa física quanto para clientes pessoa jurídica.

Segunda parte

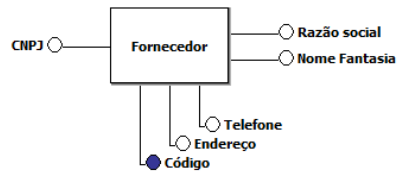
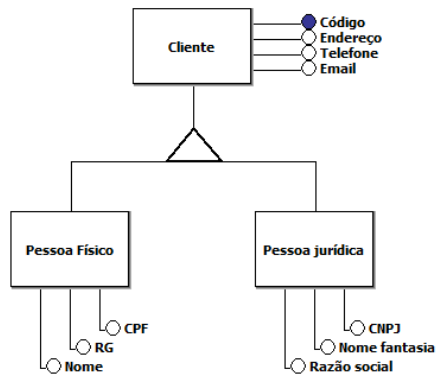
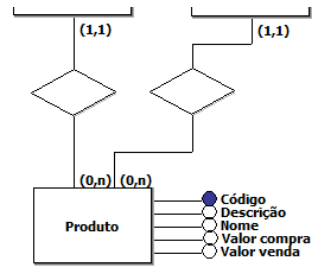
Sobre o produto: código, nome, descrição, unidade de medida, valor de compra, valor de venda, quantidade mínima e máxima em estoque e sua categoria. Da categoria: código, nome e a descrição; e da unidade de medida: código, sigla e descrição.



Crie a entidade Produto e seus respectivos atributos. Crie também a entidade Categoria. Apesar de a descrição do problema não trazer a cardinalidade do relacionamento, você deve analisar pelo modo lógico: um produto pode ter, no mínimo, uma e, no máximo, uma categoria; uma categoria pode não ter nenhum produto como pode ter vários produtos. Crie também a entidade unidade de medida e seus atributos. Quanto ao relacionamento, um produto pode ter, no mínimo, uma unidade de medida e, no máximo, 1; uma unidade de medida pode ter nenhum ou N produtos. Nos relacionamentos acima, a experiência com bancos de dados ajudou a definir a cardinalidade dos relacionamentos, mas depois, em casos reais, você deve consultar a(s) pessoa(s) que detém o conhecimento sobre a empresa/ negócio para definir com segurança.

Terceira parte

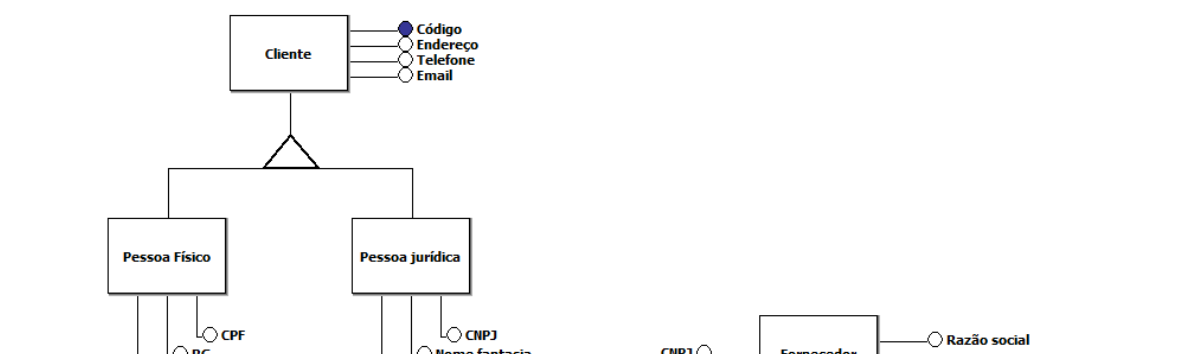
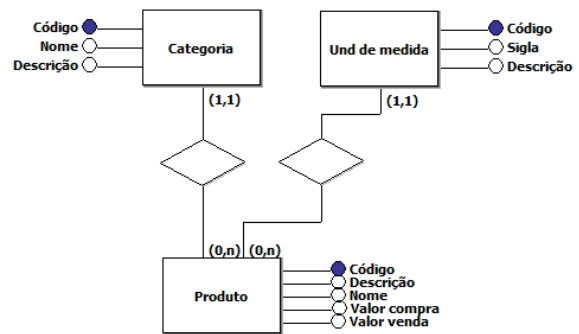
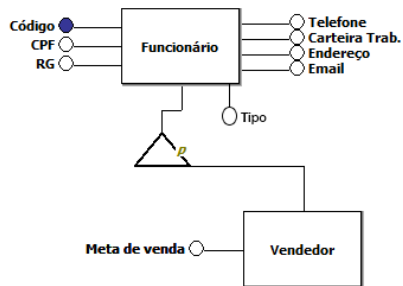
Dos fornecedores: as mesmas informações dos clientes pessoa jurídica:



Crie apenas a entidade Fornecedor com os mesmos atributos da entidade Pessoa jurídica.

Quarta parte

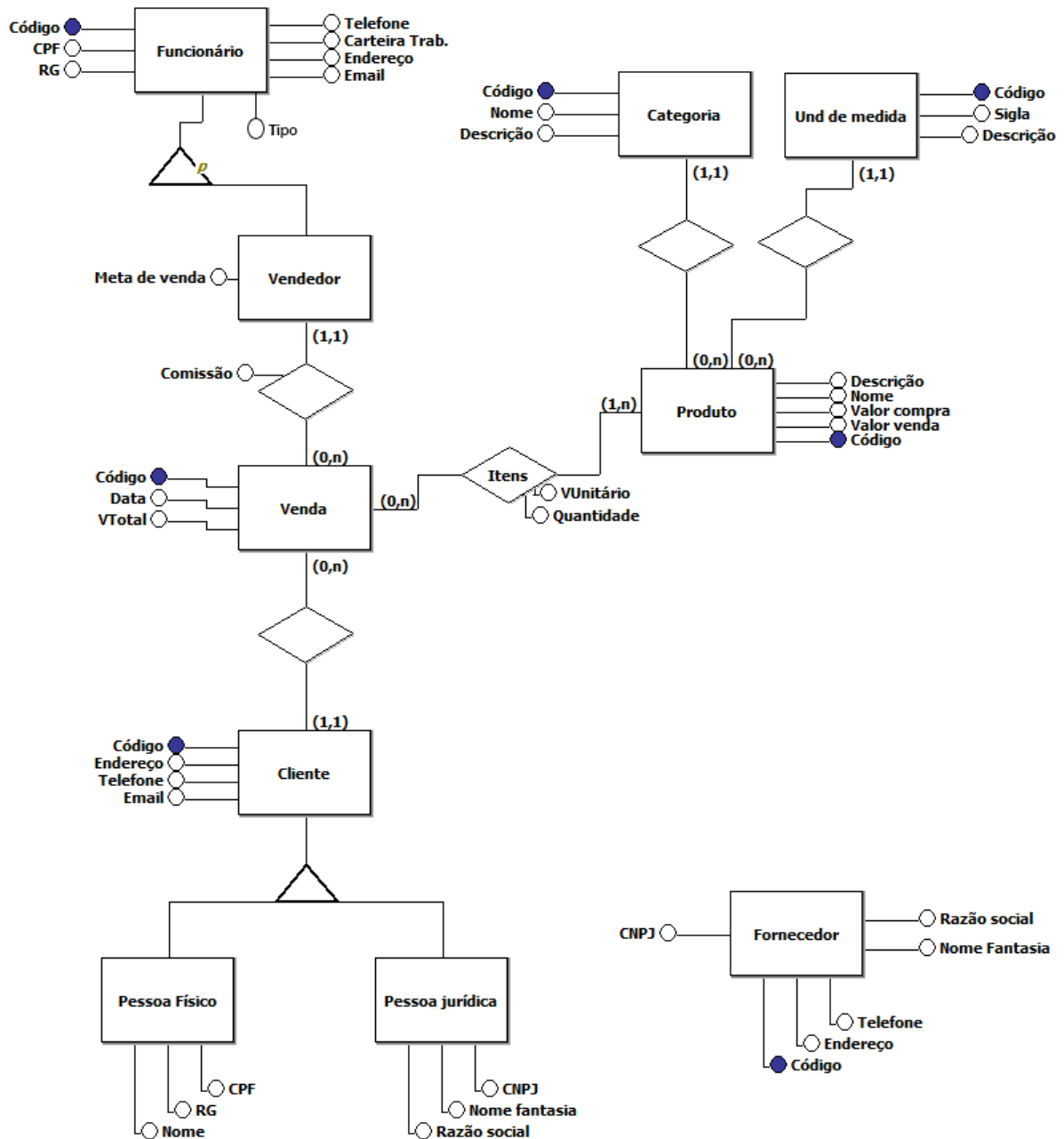
Sobre os funcionários: nome, CPF, RG, endereço, email, telefone e número carteira de trabalho. Os vendedores devem ter todas as informações dos funcionários mais o valor da meta de venda mensal. Faça assim:



Crie a entidade Funcionário e seus atributos. Como vendedor também é um funcionário, e tem os mesmos atributos, crie uma estrutura de generalização/ especialização exclusiva/parcial e adicione apenas o atributo meta de venda.

Quinta parte

Toda venda deve conter os seguintes dados: produtos vendidos, valor unitário de cada item, valor total de venda, quantidade de cada item, a data, o cliente que realizou a compra, o vendedor e a comissão daquela venda. Faça assim:



Vamos à lista de modificações:

Adicione a entidade vendas e seus atributos.

Adicione o relacionamento entre a venda e o produto, que será responsável pelos itens da venda. Por isso há os atributos sobre o valor de cada unidade e a quantidade de produto nesse relacionamento. Um produto pode ser vendido nenhuma vez como pode ser vendido N vezes. Uma venda deve ter, no mínimo, um produto e, no máximo, N produtos.

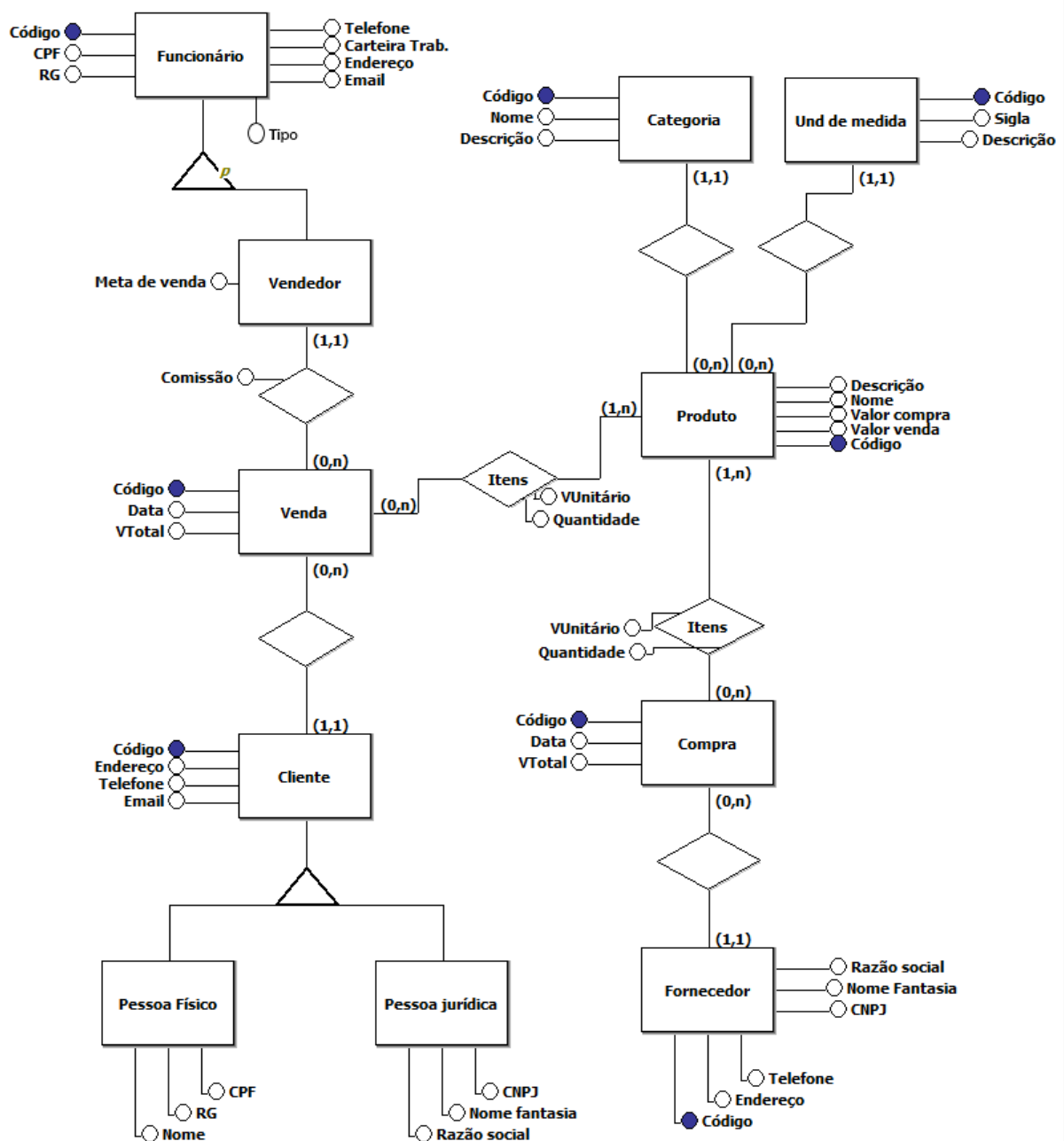
Também adicione o relacionamento que será responsável por indicar o vendedor que realizou determinada venda. A cardinalidade dele fica assim: um vendedor pode ter, no mínimo, uma

venda e, no máximo, N vendas. O atributo comissão é usado para saber a comissão do vendedor em determinada venda.

O último relacionamento é responsável por indicar qual cliente realizou determinada compra. Cada cliente pode realizar nenhuma compra como várias compras. Cada compra pode ser realizada apenas por um cliente.

Sexta parte

Das compras não é muito diferente: produtos, quantidade unitária de cada item, valor de cada item, valor total da compra e a data. Veja:

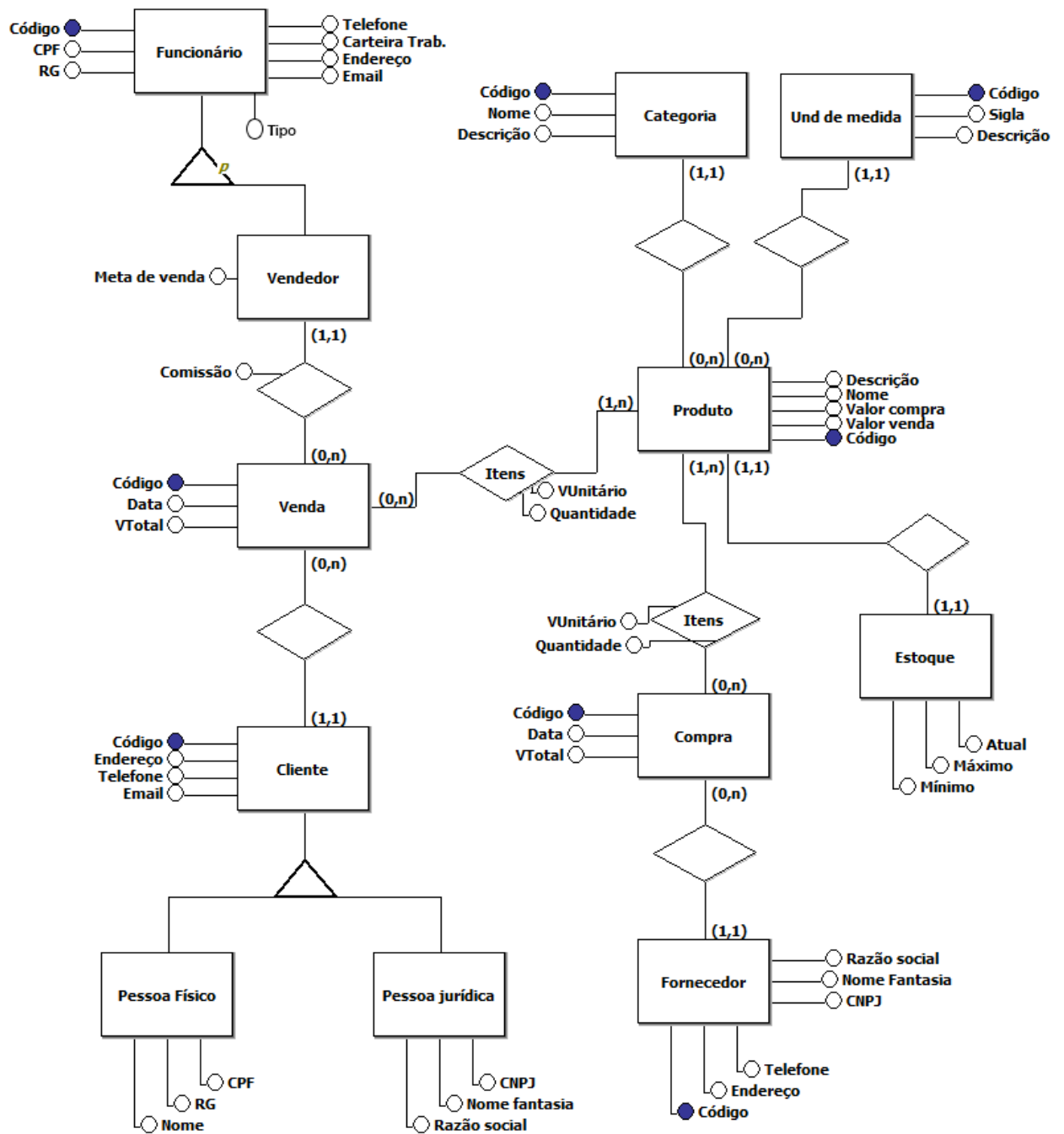


Adicione a entidade compra e seus atributos. Em seguida, crie o relacionamento responsável por indicar o valor e a quantidade de produtos que compõem a compra. Acompanhe a descrição:

Um produto pode ser comprado 0 ou N vezes. Uma compra deve ter, no mínimo, um e, no máximo, N produtos. O relacionamento entre compra e Fornecedor mostra o fornecedor responsável pela compra. Um fornecedor pode vender 0 ou N vezes para a loja. Cada compra que a loja faz deve ter, no máximo, um fornecedor.

Sétima parte

O estoque deve ter apenas a quantidade de produto e o produto. Confira:



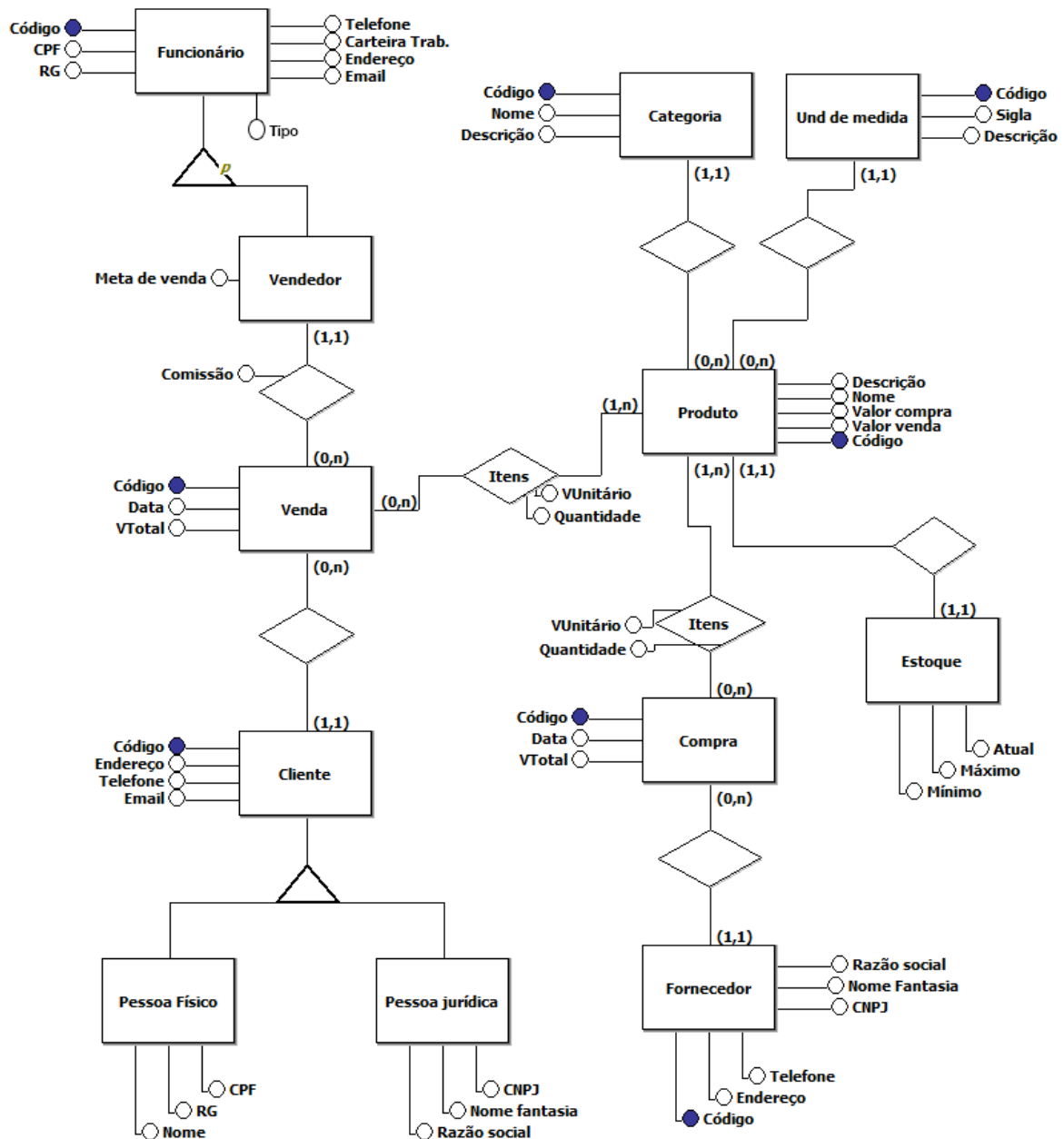
Adicione a entidade estoque, seus atributos e o relacionamento com produto. Observe: os atributos máximo e mínimo são descritos no produto, mas devem ser adicionados no estoque por questão de organização.

Essa é a versão final do modelo entidade relacionamento deste projeto. Agora você deve transformá-lo para o modelo relacional.

Tradução para o Modelo lógico

A tradução para o modelo relacional se dará do mesmo modo que no tópico exercício de transformação.

Veja a versão final novamente para facilitar a explicação:



Primeira parte

Comece sempre a tradução pelas tabelas que não receberam nenhuma coluna de relacionamento. Faça as transformações:

Categoria (catCod, Nome, Desc)

UMedida (undCod, Sigla, Desc)

Fornecedor (forCod, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, CNPJ, NomeFan, Razao)

Você deve estar se perguntando de onde surgiram as colunas logradouro, bairro, cidade, estado e CEP. Elas são referentes ao atributo endereço. Como no MER, você está tratando com coisas conceituais e, na hora da transformação, pode usar o bom senso e trazer para a realidade.

Segunda parte

Passe à estrutura generalização/especialização, cuja entidade genérica é Cliente. Nela, use a alternativa de transformação “uma tabela para cada entidade” por se tratar de estrutura mais complexa e com um grande número de atributos. Acompanhe:

Cliente (cliCod, Tipo, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, Email)

PF (cliCod, Nome, RG, CPF)

cliCod referencia cliente

PJ (cliCod, omeFan, Razão, CNPJ)

cliCod referencia cliente

T

Na segunda generalização/especialização, por ter poucos atributos, use a alternativa de transformação em que toda a estrutura vai para uma tabela única:

Funcionario (funCod, Tipo, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, Email, RG, CPF, CTrabalho, Meta)

Terceira parte

Você já tem as colunas de que as entidades Produto, Compra e Venda precisam para serem implementadas no modelo lógico. Faça as transformações:

Produto (proCod, Desc, Nome, vVenda, vCompra, EstMin, EstMax, qtdEst, catCod, undCod)

catCod referencia categoria

undCod referencia UMedida

Compra (comCod, vTotal, Data, forCod)

forCod referencia fornecedor

Venda (venCod, vTotal, Data, cliCod, funCod, Comissao)

cliCod referencia Cliente

funCod referencia Funcionario

Descrição

A tabela Produto recebe cinco colunas que não são originais de sua entidade. As colunas estmin, estmax e qtdest são referentes ao relacionamento com a entidade estoque. Como a cardinalidade é 1 para 1, use a alternativa de transformação fusão de tabela. Já as outras duas colunas catcod e Undcod são referentes aos relacionamentos com categoria e unidade de medida, respectivamente. Como ambos são do tipo 1 para N use a alternativa de adição de colunas: produto tem a menor cardinalidade e recebe as colunas.

A tabela Compra recebe a coluna forcod, referente ao relacionamento 1 para N com a tabela Fornecedor; compra tem a menor cardinalidade, então recebe a colunas com a referência de Fornecedor.

A tabela Venda recebe três colunas: clicod, funcod e comissao. A primeira é referente ao relacionamento com Cliente; as duas últimas referentes ao relacionamento com Vendedor. Todas elas são fruto de relacionamentos 1 para N, como nos outros casos.

Quarta parte

Agora só restaram os relacionamentos N para N: os itens de venda e de compra. Confira como eles ficam:

ItensVenda(venCod, codPro, Qtd, VUnitario)

codVen referencia Venda

codPro referencia Produto

ItensCompra (codCom, codPro, Qtd, VUnitario)

codCom referencia Compra

codpro referencia Produto

Para todo relacionamento N para N deve ser usada a alternativa de tradução tabela própria. Por isso, gere uma tabela para cada relacionamento acima. Lembre-se: para essa alternativa de transformação você deve criar uma nova tabela, receber os campos das entidades envolvidas e transformar em chave primária. As tabelas geradas acima são de grande importância porque, apenas com a tabela compra, não se consegue colocar vários produtos em uma mesma compra. O que permite isso é a tabela que relaciona o produto com a compra. A mesma coisa vale para a entidade venda.

Versão final do modelo lógico textual

Categoria (catCod, Nome, Desc)

UMedida (undCod, Sigla, Desc)

Fornecedor (forCod, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, CNPJ, NomeFan, Razao)

Cliente (cliCod, Tipo, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, Email)

PF (cliCod, Nome, RG, CPF)

cliCod referencia cliente

PJ (cliCod, NomeFan, Razão, CNPJ)

cliCod referencia cliente

Funcionario (funCod, Tipo, Logradouro, Bairro, Cidade, Estado, CEP, telFixo, Cel, Email, RG, CPF, CTrabalho, Meta)

Produto (proCod, Desc, Nome, vVenda, vCompra, EstMin, EstMax, qtdEst, catCod, undCod)

catCod referencia categoria

undCod referencia UMedida

Compra (codCom, vTotal, data, forCod)

forCod referencia fornecedor

Venda (venCod, vTotal, Data, cliCod, funCod, Comissao)

cliCod referencia Cliente

funCod referencia Funcionario

ItensVenda(venCod, codPro, Qtd, VUnitario)

codVen referencia Venda

codPro referencia Produto

ItensCompra (codCom, codPro, Qtd, VUnitario)

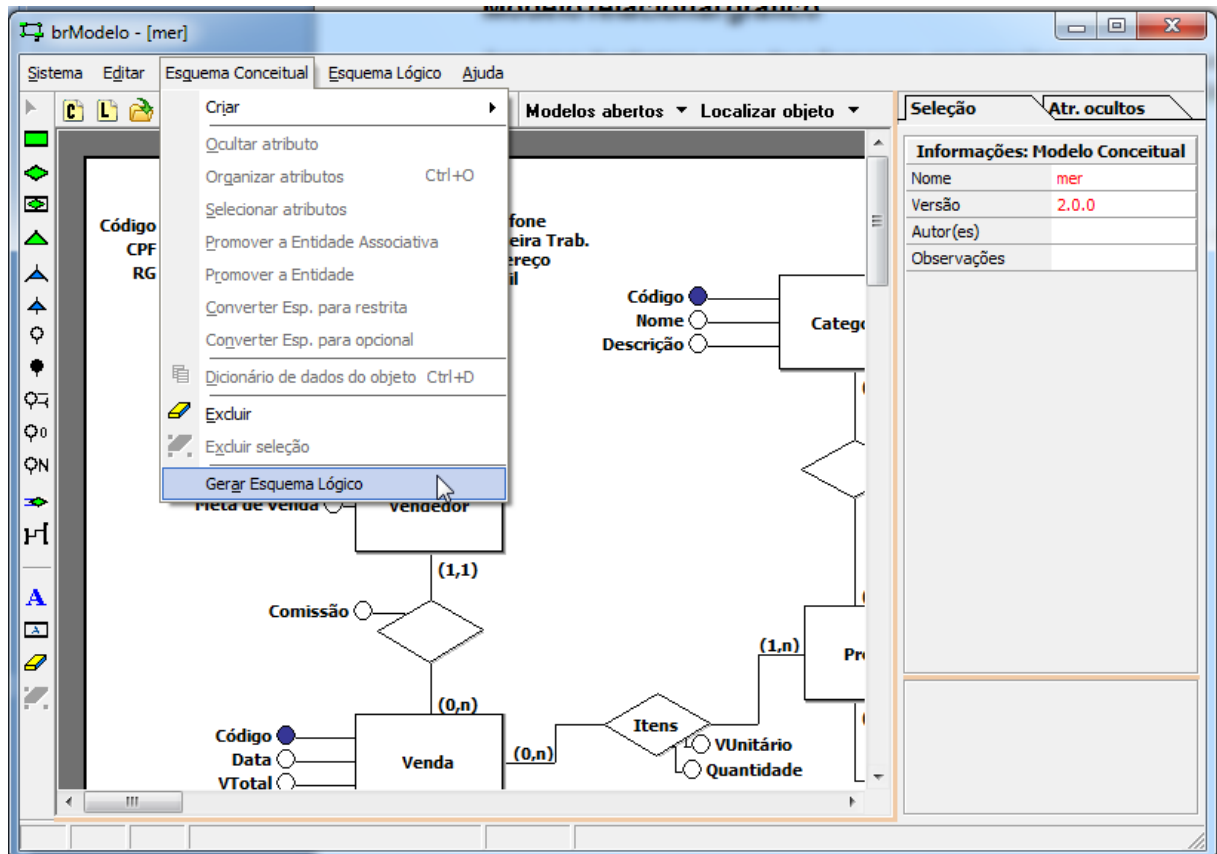
codCom referencia Compra

codpro referencia Produto

Modelo relacional gráfico

Agora que você já sabe como deve ficar o esquema lógico, pode transformar o modelo entidade relacionamento em modelo lógico gráfico, usando o BrModelo. Abra a versão final do MER e siga as instruções.

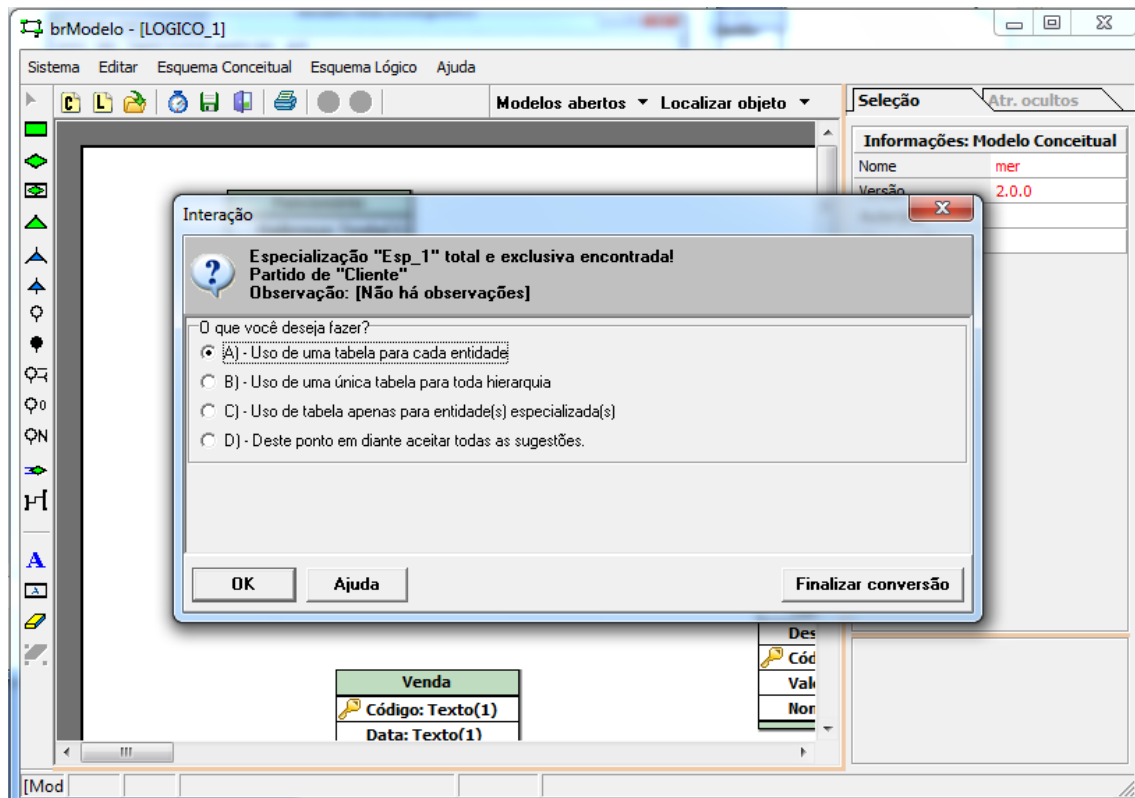
Para transformar no BrModelo, primeiro vá em Esquema Conceitual, depois em Gerar Esquema Lógico. Veja:



O BrModelo fará duas perguntas:

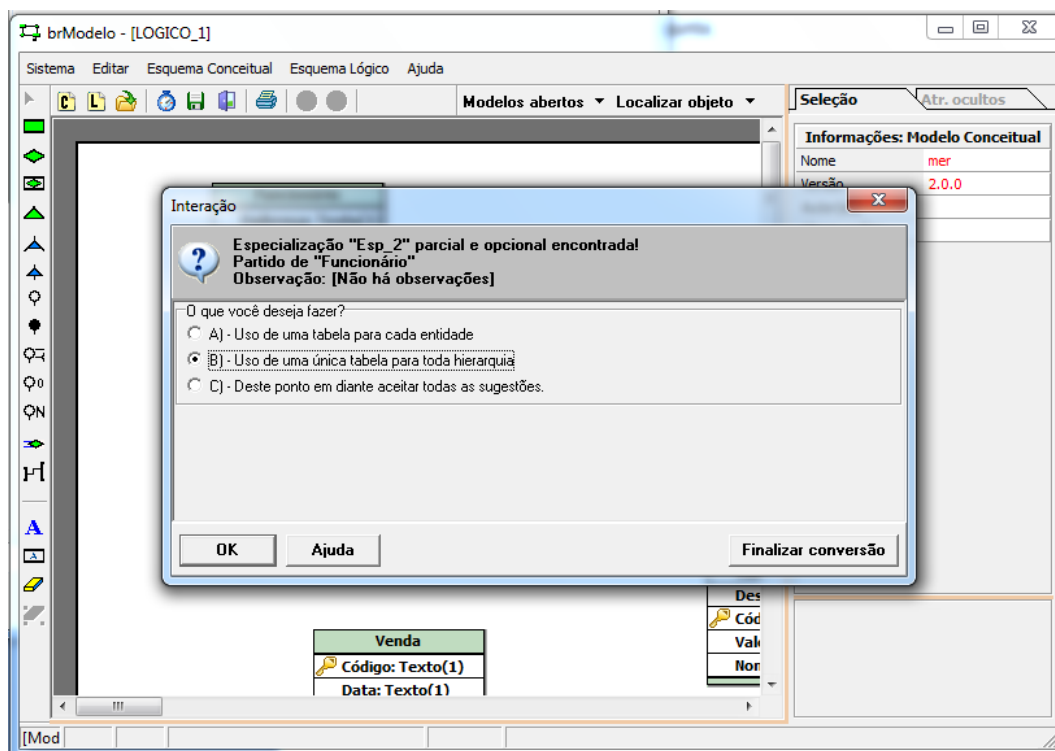
Primeira pergunta

Ele perguntará o que deve fazer com a estrutura generalização/especialização cuja entidade genérica é Cliente. Assim como feito na transformação manual, indique a resposta: A) - uso de uma tabela para cada entidade e clique em OK.



Segunda pergunta

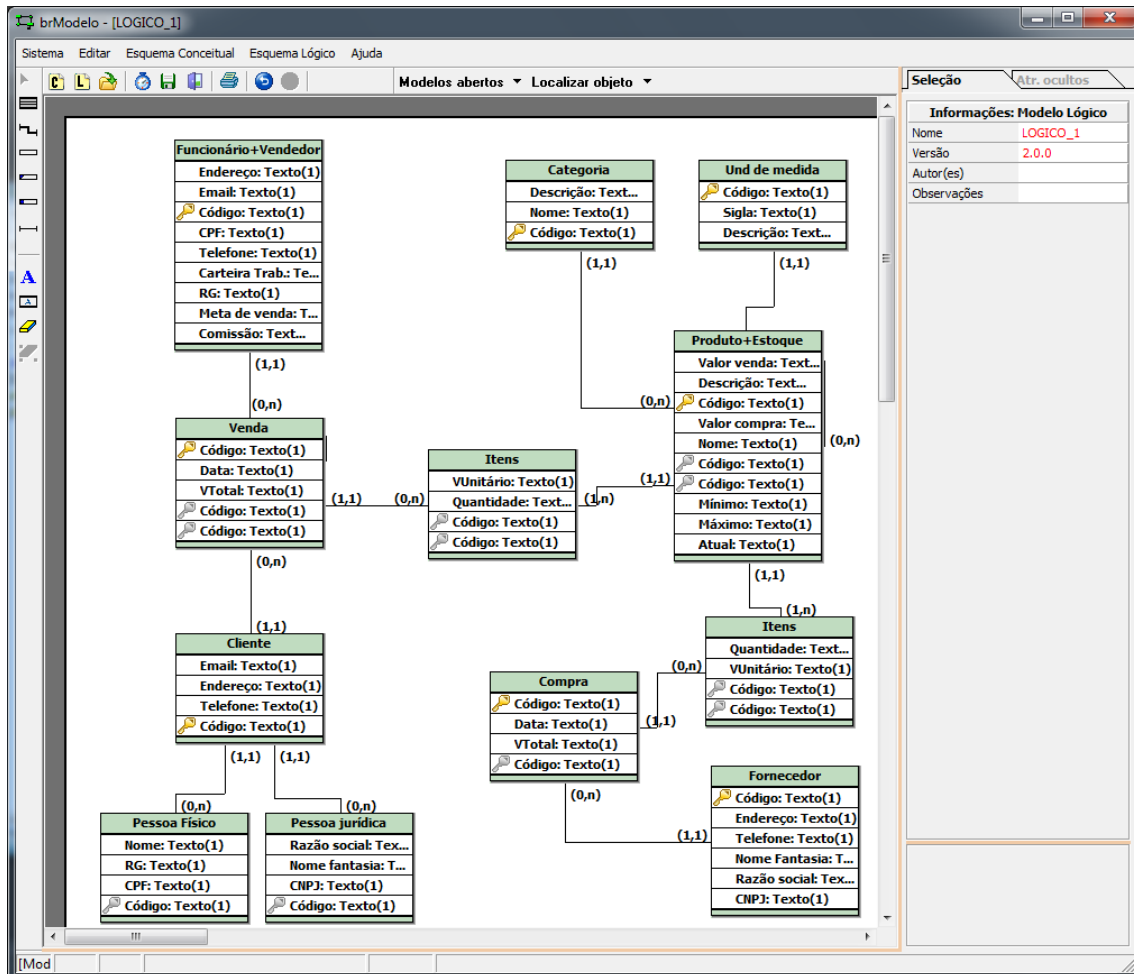
A pergunta refere-se à segunda estrutura de generalização/especialização. Ele diz que a entidade genérica da estrutura é Funcionário, então você deve tomar a mesma decisão da transformação manual: B) – Uso de uma única tabela para toda a hierarquia e clique em OK.



Pronto, este é o modelo relacional gráfico.

Modelo relacional gráfico

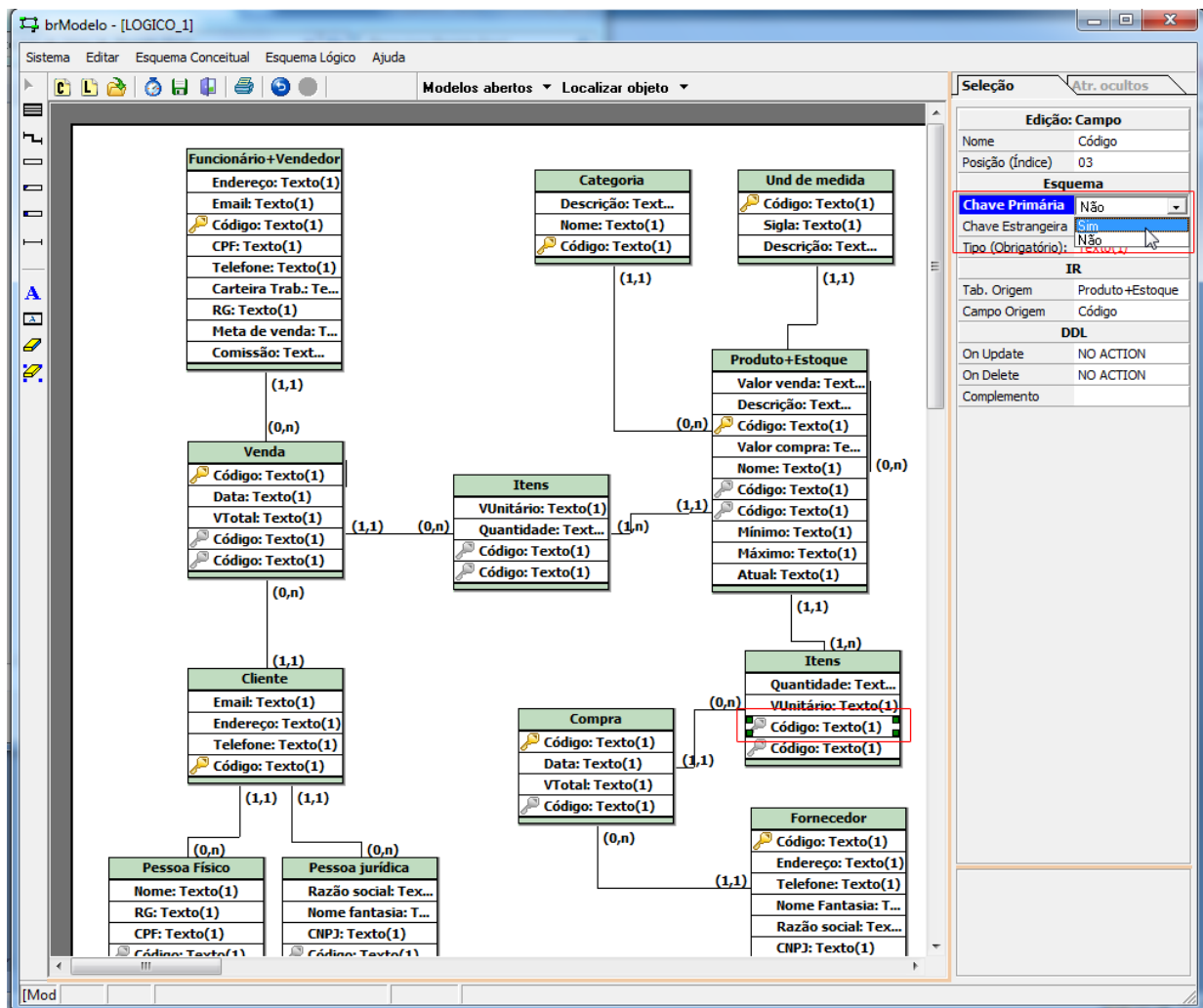
O BrModelo mostrará a versão inicial do modelo gráfico. Você deve, sempre, verificar a posição das tabelas para deixar o mais organizado possível. Confira a imagem abaixo:



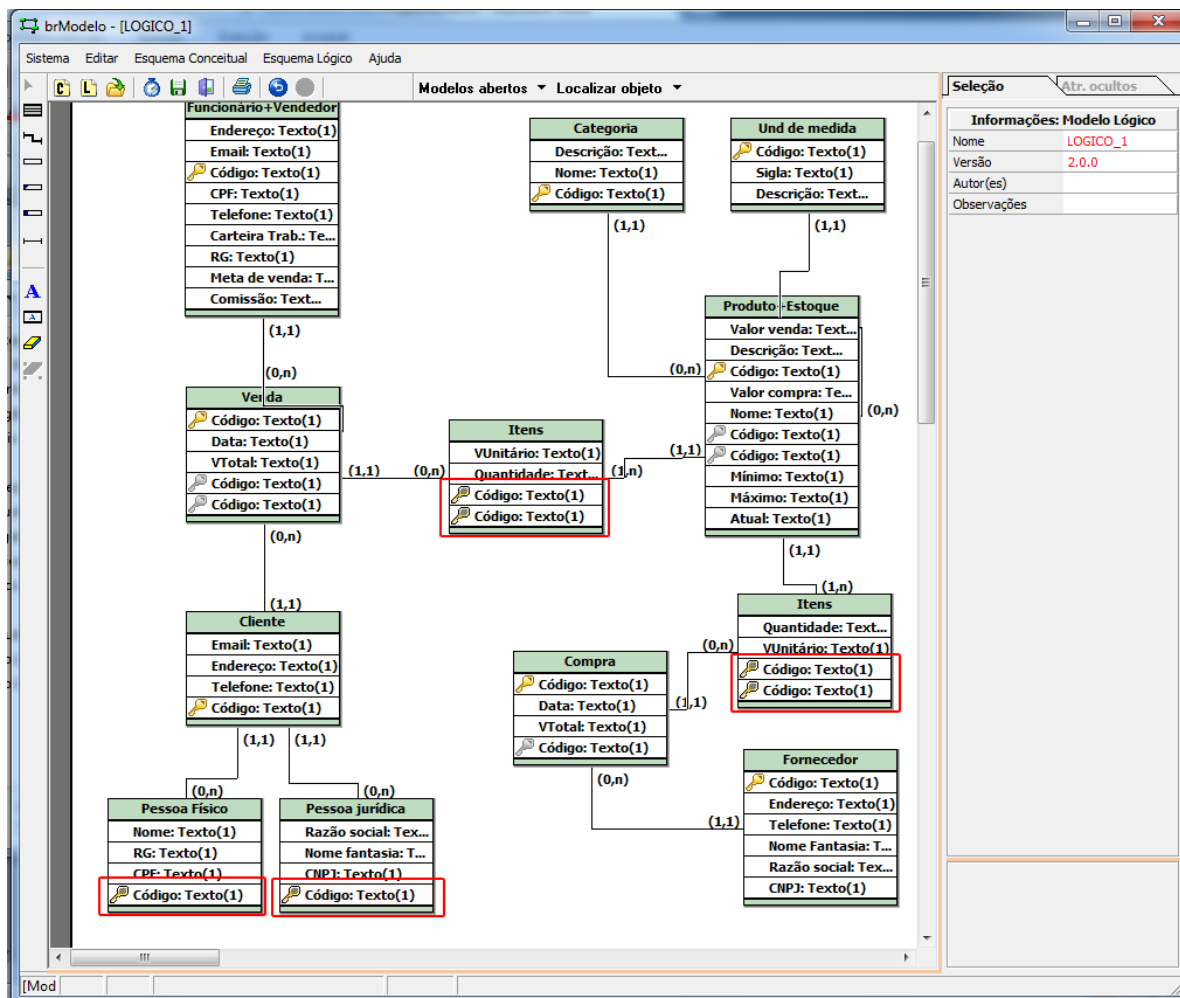
Você precisará fazer dois ajustes: o primeiro é transformar a chave estrangeira dos relacionamentos N para N em chave primária, e também as chaves das tabelas Pessoa física e Pessoa jurídica; o segundo é renomear todas as chaves estrangeiras para facilitar a identificação.

Ajustando relacionamentos N para N

Você precisa transformar as chaves estrangeiras em chave primária também. Selecione a chave, vá em Seleção > Chave Primária e marque Sim.



Deve fazer o mesmo nas seis chaves estrangeiras: duas de cada tabela Itens, uma da tabela Pessoa física e uma da Pessoa jurídica. Elas devem ficar como mostra a imagem a seguir:

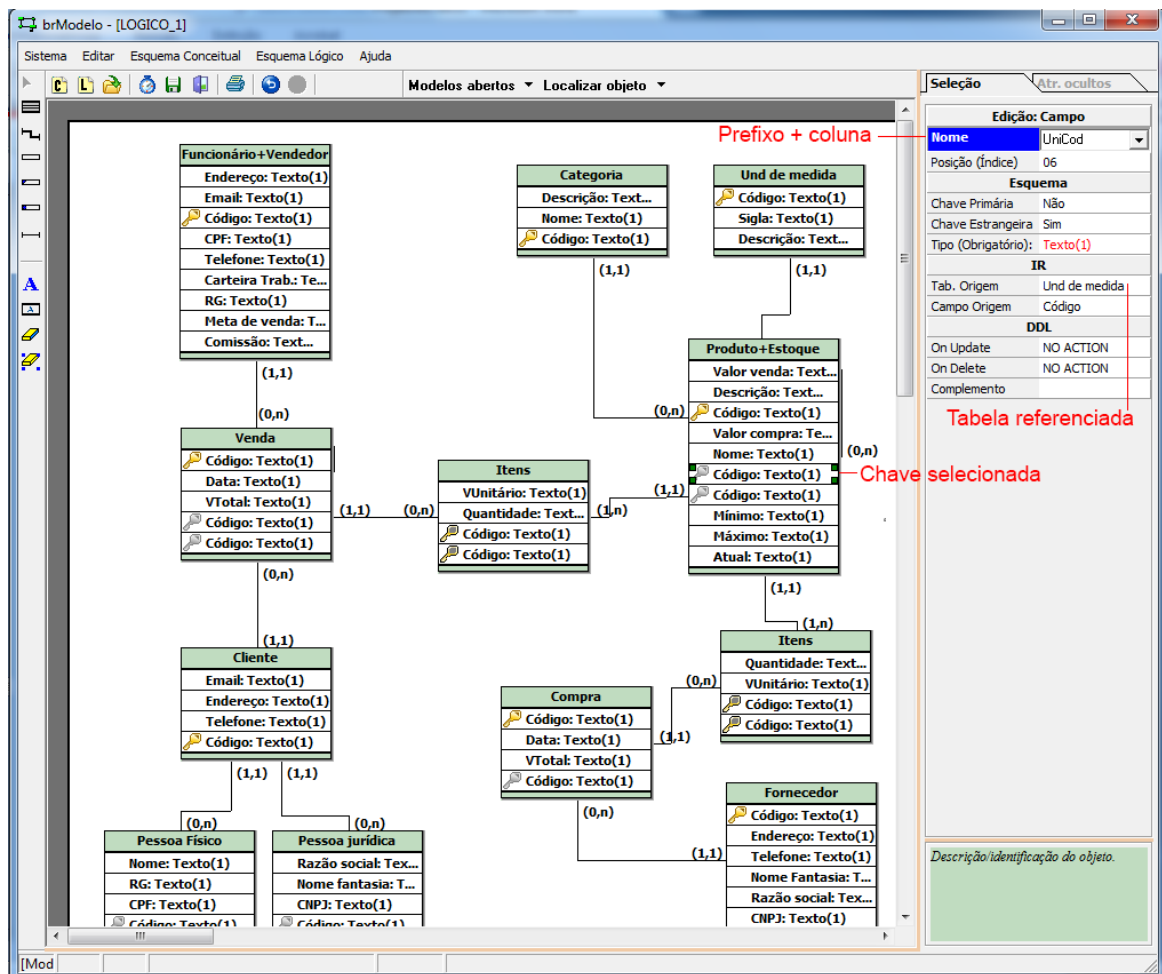


Lembre-se: o ícone que informa a chave estrangeira e primária ao mesmo tempo é a chave amarela com o meio cinza.

Alterando o nome das chaves estrangeiras

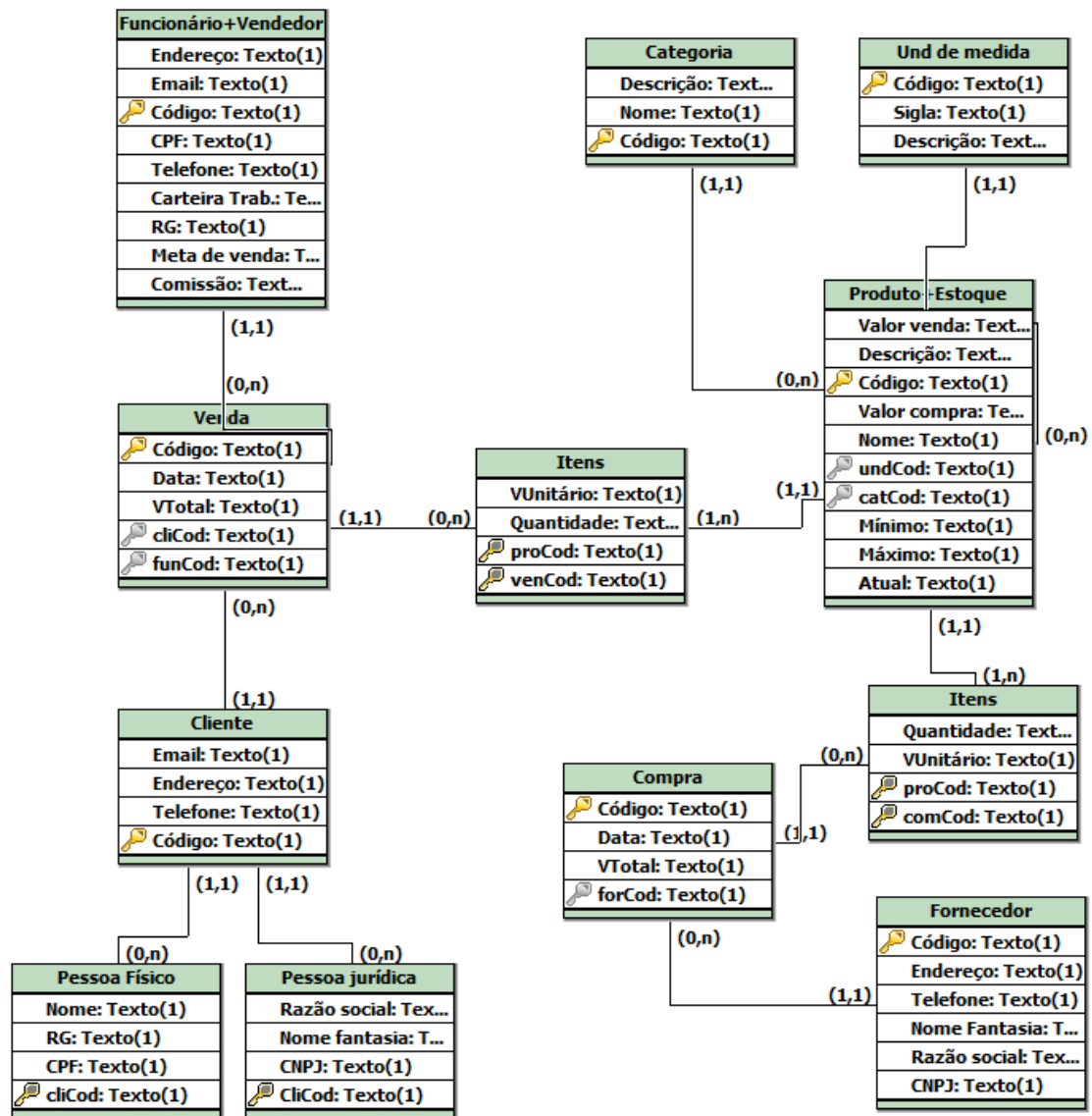
Todos os atributos identificadores do modelo entidade relacionamento chamam código, por isso o BrModelo, na hora da transformação, deixou todos com o mesmo nome, dificultando a identificação da entidade que o atributo referencia. Para solucionar esse detalhe, você deve renomear todas as chaves estrangeiras com o prefixo da tabela de origem + (mais) o nome da coluna. Selecione a chave estrangeira, vá na aba Seleção > Alterar a propriedade nome. Para saber qual tabela essa chave referencia basta olhar a propriedade Tab. Origem.

Veja na imagem abaixo:



Faça isso para todas as chaves estrangeiras.

Versão final



Note que todas as chaves estrangeiras do modelo foram renomeadas.