

1.1.1 Notação IDEF1X

A notação IDEF1X (*Integrated DEFinition for Information Modelling*) foi desenvolvida no final da década de 70 para a força aérea norte-americana. Esta metodologia faz parte de um conjunto de técnicas (IDEF) para a modelagem completa de um sistema, incluindo modelagem de processos, dados, simulação e descrição de procedimentos. A metodologia IDEF1X é um padrão do birô federal de processamento de informações norte-americano (padrão FIPS 184) e de uso público e livre.

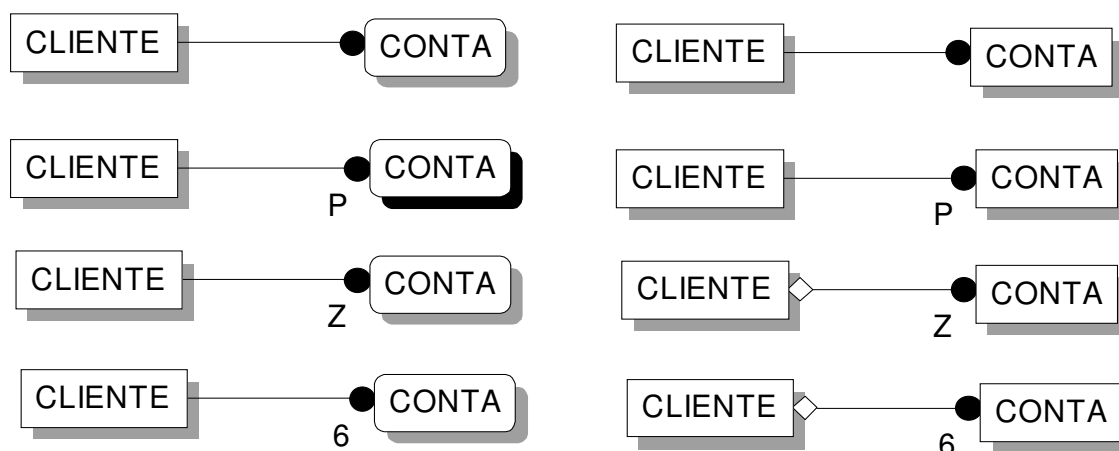


Figura 1. DER na notação IDEF1X (visão de entidades).

Da mesma forma que no modelo IE, a notação IDEF1X fornece indicações de cardinalidade tanto nos relacionamentos identifikantes quanto não-identifikantes. Observe (Figura 2) que os relacionamentos são nomeados.

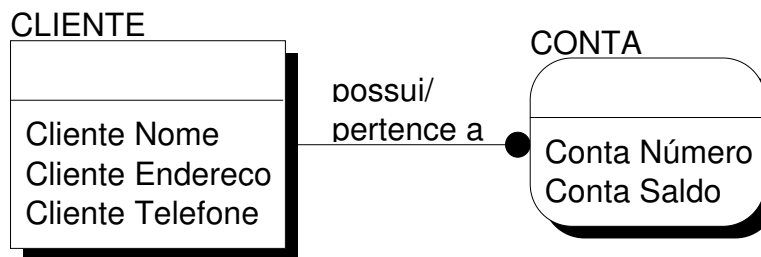


Figura 2. DER em notação IDEF1X (visão de entidades/atributos)

Na Figura 3 apresentamos um trecho do documento oficial de definição do padrão do FIPS para a metodologia IDEF1X.

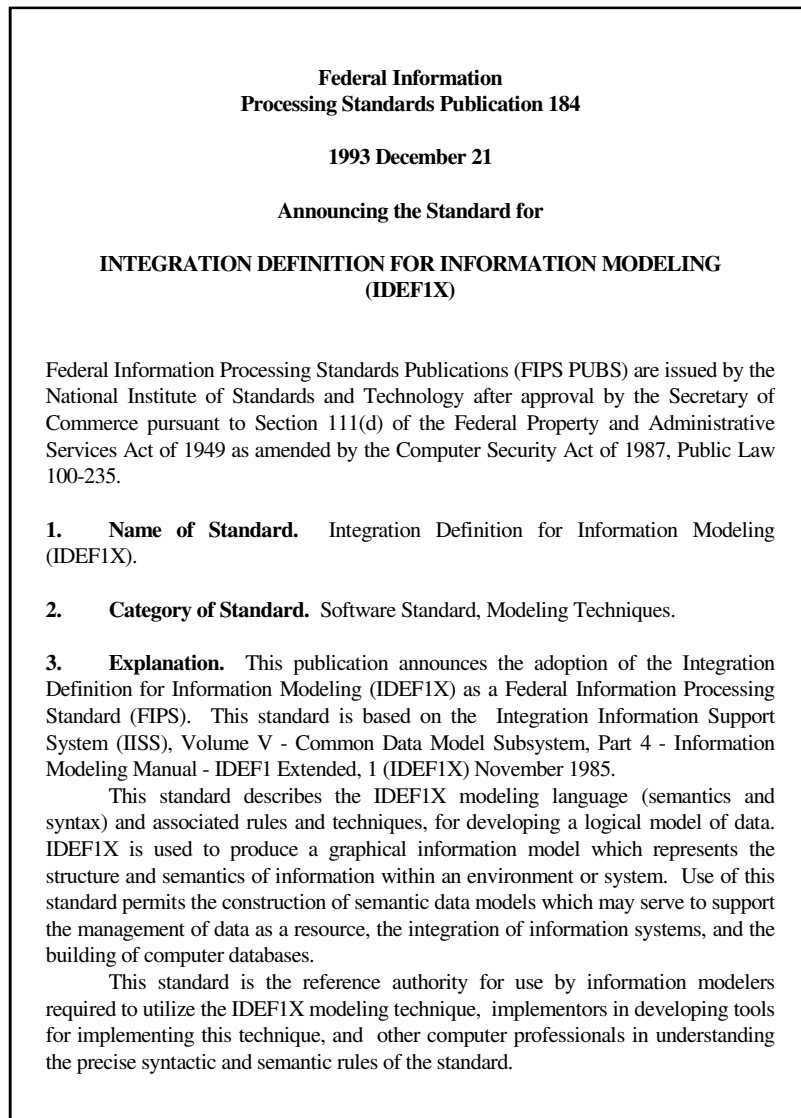


Figura 3. Trecho do documento oficial FIPS 184 - Metodologia IDEF1X.

1.2 MODELO RELACIONAL

O modelo relacional consiste numa coleção de tabelas, cada qual designada por um nome único. Cada tabela possui diversas colunas que representam os campos ou atributos da tabela. Uma linha em uma tabela representa um relacionamento entre um conjunto de valores.

O modelo possui este nome pois o conceito de tabela está intimamente ligado ao conceito de *relação* definido na matemática: “*relação é um subconjunto de um produto cartesiano de uma lista de domínios*”.

O modelo relacional possui uma relação praticamente direta com o modelo E-R, principalmente nas notações *IDEF1X* e *IE*.

As linguagens de consulta no modelo relacional são a álgebra relacional, o cálculo relacional de tupla (“*sinônimo formal*” de linha de uma tabela) e o cálculo relacional de domínio. É

importante ter alguma noção da álgebra relacional pois muitas considerações sobre otimização de bancos de dados remontam às expressões em álgebra relacional.

Exemplo de representação de relações e relacionamentos.

CLIENTE

Nome	Endereço	Cidade	Conta
JEAN	RUA 5	PARIS	428
SMITH	RUA 3	LONDRES	428
SMITH	RUA 3	LONDRES	629
PHIL	RUA 3	BERLIN	423

CONTA

Número	Saldo
423	\$12
428	\$600
629	\$150

Terminamos aqui nossos comentários sobre os modelos de dados. Em seguida nos concentraremos na modelagem de dados, através dos modelos E-R (DER) e definiremos formalmente entidades, relacionamentos, atributos e outros conceitos importantes.

Para maiores detalhes sobre os diversos modelos de dados o leitor deve consultar [KORTH, 1994].

2. MODELAGEM DE DADOS

2.1 INTRODUÇÃO

O objetivo da modelagem de dados é construir um modelo estável do conjunto de informações necessário para o funcionamento de um negócio ou empreendimento qualquer.

As informações para a construção de um modelo de dados vêm das seguintes formas:

1. Dos dados obtidos na etapa de levantamento de informações, feita através de entrevistas, observação e análise dos manuais de organização e procedimentos;
2. Da análise funcional das áreas de negócio no escopo do sistema;
3. Da engenharia reversa dos sistemas de informação existentes.

A modelagem de dados não se refere à crítica da natureza das informações tratadas no ambiente de negócio, ou em que intensidade estas informações contribuem para o sucesso do empreendimento. Refere-se apenas à construção ótima de estruturas de dados que dêem suporte ao empreendimento.

DEFINIÇÃO 1 – MODELAGEM DE DADOS

“Uma abordagem estruturada utilizada para identificar os principais componentes de um sistema de informação”.

DEFINIÇÃO 2 – MODELO DE DADOS

“O modelo de dados é a especificação de estrutura de dados e regras de negócio para a representação das necessidades da empresa”.

Com a modelagem de dados conseguimos:

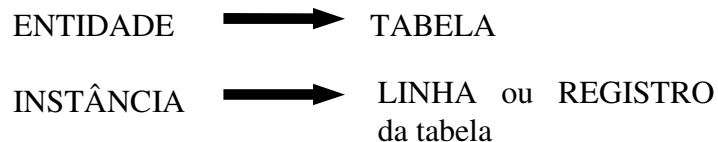
- reduzir ou eliminar a redundância de dados na empresa;
- fornecer aos profissionais da área de negócios (à administração) um modelo gráfico das necessidades e regras de negócio da empresa, reduzindo assim a distância entre a administração e os técnicos;
- estabelecer um consenso quanto aos dados necessários ao funcionamento da empresa;
- construir uma base de dados estável para a empresa.

DEFINIÇÃO 3 – ENTIDADE

“ENTIDADE é uma pessoa, coisa, lugar, evento ou conceito sobre o qual a empresa necessita manter dados.”

Cada entidade é um conjunto ou coleção de objetos *de mesmas características* denominados INSTÂNCIAS. Cada instância deve ser unicamente identificada e distinta de todas as outras instâncias.

Pensando no modelo relacional, podemos fazer a seguinte analogia a estes termos:



Segundo o padrão IDEF1X, toda entidade deve ser unicamente nomeada em letras maiúsculas, de acordo com o termo utilizado no ambiente de negócios. A representação da entidade pode ser vista na figura 6.

DEFINIÇÃO 4 – ATRIBUTO

“É uma característica específica da entidade sobre a qual dados são mantidos.”.

Nos exemplos vistos anteriormente, CLIENTE era uma entidade. Cliente-Nome é um atributo da entidade cliente (precisamos guardar informação sobre o nome do cliente).

Em outras palavras, entidade é algo sobre que desejamos guardar informações. Os atributos são as informações a serem guardadas.

Segundo o padrão IDEF1X os atributos são nomeados com a primeira letra em maiúsculo, com nomes relacionados ao ambiente de negócios.

TANTO ENTIDADES QUANTO ATRIBUTOS DEVEM SER DEFINIDOS. Assim, deve-se criar um dicionário de dados que armazene as informações sobre estas definições. Um exemplo de definição para *cliente* poderia ser:

- CLIENTE = Pessoa física ou jurídica vinculada ao banco através de uma conta bancária.

DEFINIÇÃO 5 – RELACIONAMENTO

“É uma ligação lógica entre duas entidades que representa uma regra ou característica do ambiente de negócio”.

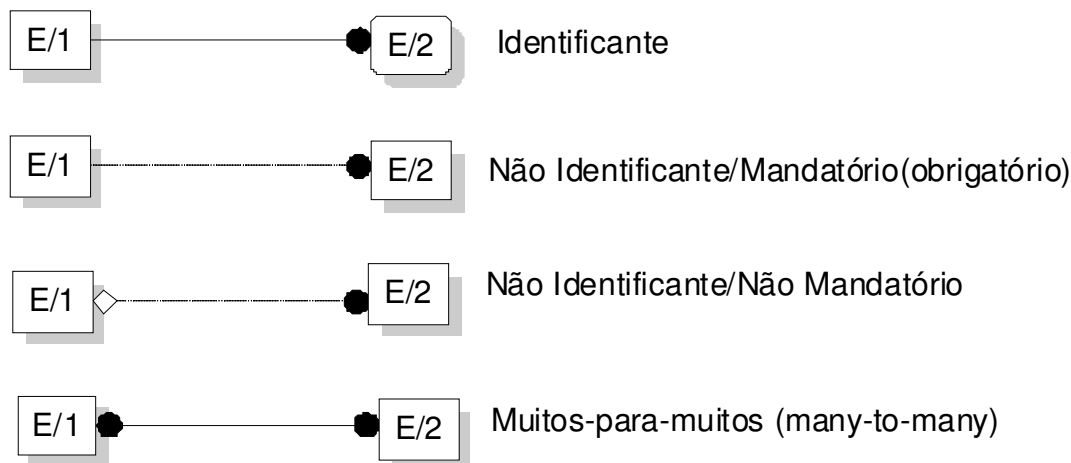


Figura 4. Tipos de relacionamento em notação IDEF1X.

Os relacionamentos devem ser nomeados com letras minúsculas. A cardinalidade representa uma relação quantitativa entre as entidades. A cardinalidade é indicada como mostrado na Figura 1, através das letras P, Z e N.

Cardinalidade	Significado	Exemplo
(sem indicação)	0, 1 ou mais	Um FUNCIONÁRIO possui 0, 1 ou mais FILHOS.
P	Vem de Positive. 1 ou mais.	Um GRUPO é composto de 1 ou mais PESSOAS.
Z	Vem de Zero. 0 ou 1	Um ESTUDANTE possui 0 ou 1 BOLSA DE ESTUDO
N	exatamente “N”	Um NAVIO possui exatamente 6 movimentos. (Neste caso, N=6).

Tabela 1. Indicação da cardinalidade em notação IDF1X.

O relacionamento muitos-para-muitos representa conceitos do seguinte tipo: uma clínica possui diversos médicos e cada médico pode atender em diversas clínicas.

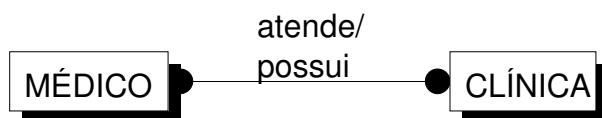


Figura 5. Um relacionamento muitos-para-muitos (many-to-many).

2.2 ÍNDICES e ATRIBUTOS CHAVE

Uma outra propriedade importante relacionada ao modelo de dados relacional é que cada instância deve ser unicamente identificada e diferente das demais. Por isso, é necessário determinar quais são os atributos que caracterizam e diferenciam uma instância das demais. Estes atributos são denominados *atributos-chave* da entidade.

DEFINIÇÃO 6 – ATRIBUTOS CHAVE–CANDIDATOS

São atributos que são únicos para cada instância da entidade.

Em princípio, qualquer atributo único poderia ser utilizado para identificar uma entidade. No caso de diversos atributos únicos para uma mesma entidade, a escolha deverá ser feita em função de características do atributo que impliquem em melhor desempenho do SGBD ou que tenham mais afinidade com as operações realizadas no ambiente de negócios.

Por exemplo, suponha que um ambiente de negócios precisa guardar as seguintes informações do fornecedor:

FORNECEDOR	
Fornecedor Nome Fantasia	
Fornecedor Razão Social	
Fornecedor CGC	
Fornecedor Endereço	
Fornecedor Telefone	
Fornecedor Fax	
Fornecedor e-mail	

Figura 6. Entidade FORNECEDOR, em visão de entidade/atributos.

Supondo-se que esta empresa vai lidar somente com fornecedores nacionais, sabemos que em nosso país não é possível registrar duas empresas com mesma razão social, nem duas empresas com mesmo CGC. Portanto, estes dois atributos são dois candidatos para a identificação da entidade.

DEFINIÇÃO 7 – CHAVE–PRIMÁRIA (PRIMARY KEY – PK)

É o campo ou atributo escolhido que identifica unicamente cada instância da entidade.

A chave–primária é o campo ou atributo chave–candidato escolhido para identificar uma entidade. É este atributo que vai garantir que todas as instâncias da entidade são diferentes umas das outras. Ainda, é utilizando este atributo que o SGBD fará a grande maioria das operações sobre os registros de uma tabela.

O primeiro passo da modelagem de dados é identificar a chave primária de uma entidade.

No exemplo do fornecedor, citado há pouco, se escolhermos o CGC como chave primária, a representação da entidade tomaria a seguinte forma:



Figura 7. Entidade FORNECEDOR com a chave primária definida.

Observe que na notação IDEF1X o atributo chave-primária fica na divisão superior da entidade.

Mas, não poderíamos escolher a razão social como chave-primária? Para respondermos a esta pergunta precisaremos explicar qual a implicação física, no SGBD, da definição de uma chave primária.

2.2.1 Índice de Chave-Primária

Ao definirmos uma chave-primária para uma entidade o SGBD cria fisicamente na base de dados um *índice* único, que nada mais é que um arquivo físico que tem o mesmo conceito do índice de um livro: ele utiliza uma palavra de referência, de conhecimento do usuário (o atributo chave) para apontar para o endereço físico no BD onde se encontra aquela instância da entidade. O índice é dito *único* pois não admite que se registre duas instâncias que contenham o mesmo valor de chave-primária.

Assim sendo, para procurar por uma instância não precisaríamos especificar a instância inteira, mas somente o atributo chave seria suficiente para que o SGBD encontrasse rapidamente aquele registro procurado.

O índice de chave-primária é o que promove mais rápido acesso a um registro. *Quanto menor for, em bytes, o atributo escolhido como chave, mais rapidamente o SGBD conseguirá encontrar o registro procurado. Daí não optarmos pela razão social como chave primária.*

Por outro lado, a definição de uma chave-primária não significa que este será o único meio de acessar as linhas da tabela.

2.2.2 Índices e Chaves Alternativos

É razoável supor que, na maioria das atividades em um ambiente de negócios, seria muito exigir que os funcionários memorizem o CGC de todos os fornecedores. Neste caso, pode ser uma boa opção possuir um meio de acesso rápido aos dados do fornecedor através da razão social e até através do nome-fantasia. Vamos estudar estes casos.

Como sabemos, a razão social é um atributo único para cada instância da entidade fornecedor (era um atributo chave-candidato). Se desejamos, eventualmente, fazer uma busca rápida de registros através da razão social, pode ser interessante criar um índice utilizando este atributo.

DEFINIÇÃO 8 – CHAVE ALTERNATIVA SEM REPETIÇÃO (*Alternate Key*)

É um atributo chave-candidato que não foi escolhido, por algum motivo, para ser chave-primária de uma entidade, mas que foi escolhido como um meio de acesso alternativo para os registros. O índice do tipo alternate key (AK), assim como a chave-primária, não admite valores repetidos entre instâncias.

Ainda, pode não ser viável do ponto de vista do ambiente de negócios que o funcionário utilize a razão social como meio de acesso, uma vez que é muito comum que o funcionário utilize cotidianamente o nome fantasia.

Entretanto, como não há exigências legais de que duas empresas não tenham o mesmo nome fantasia, então pode ocorrer que dois fornecedores tenham o mesmo nome fantasia. Neste caso, para acesso aos registros através deste campo deve-se utilizar um índice que admita repetições.

DEFINIÇÃO 9 – CHAVE ALTERNATIVA COM REPETIÇÃO (*Inversion Entry*)

É um atributo que não possui características suficientes para ser uma chave primária, mas que é utilizado com frequência no ambiente de negócios como meio de acesso rápido às informações de uma instância da entidade, mesmo que este atributo possua o mesmo valor para instâncias diferentes. Assim, o índice do tipo inversion entry (IE), ao contrário dos PK e AK, admite valores repetidos entre instâncias.

Todos os demais atributos de uma entidade são denominados *atributos não-chave*.

2.2.3 Chave Artificial

É comum encontrarmos entidades que não possuem nenhum atributo único, que possa ser candidato a chave primária. Por exemplo, imagine um estabelecimento de vendas de vestuário onde haja venda através de crediário. Tradicionalmente, principalmente em cidades do interior, estas lojas não exigem nenhum documento para venda através de crediário. Exigem apenas um comprovante de residência. Neste caso, o cadastro de clientes seria representado pela seguinte entidade

CLIENTE

Cliente Nome Cliente Endereço Cliente Filiação Cliente Data de Nascimento

Figura 8. Entidade sem atributo candidato a chave-primária.

Nenhum dos atributos é adequado para ser uma chave primária (pense o por quê). Neste caso, uma boa solução é criarmos uma chave artificial (*surrogate key*) para identificar individualmente cada um dos registros e utilizarmos, por exemplo, o nome como um índice alternativo do tipo AK. Esta chave artificial poderia ser o código do cliente. Assim, a entidade seria representada da seguinte forma:

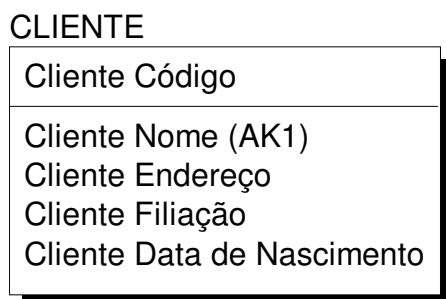


Figura 9. Entidade CLIENTE com uma chave-primária artificial.

Outras considerações sobre a definição de chaves e índices serão feitas durante o estudo dos relacionamentos, normalização e otimização do modelo de banco de dados.

2.3 TIPOS DE RELACIONAMENTO

Apresentamos rapidamente junto à definição 5 os tipos de relacionamentos representados segundo a notação IDEF1X (Figura 4).

Conforme dissemos, os relacionamentos representam regras ou restrições do ambiente de negócios. A determinação do relacionamento entre duas entidades está pois completamente ligado à compreensão que se tem do sistema sob estudo. Pelo que observamos, possivelmente a determinação do tipo de relacionamento é a tarefa mais difícil para os iniciantes na modelagem de dados. Alguns exemplos baseados em cenários ilustrarão o processo, com o objetivo de tornar mais clara esta compreensão. Vale lembrar aos leitores “mais avançados” na teoria de que os exemplos visam esclarecer o tópico onde estão inseridos, não estando sempre 100% corretos do ponto de vista de modelos de dados práticos.

2.3.1 RELACIONAMENTO IDENTIFICANTE

O relacionamento identificante entre duas entidades E1 e E2 (neste sentido) indica que uma instância da entidade E2 não pode ser completamente identificada sem que se conheça a instância da entidade E1 a que ela está associada.

Por exemplo, considere que em uma determinada empresa de vendas no atacado os pedidos sejam identificados da seguinte forma: para cada novo cliente os pedidos são numerados a partir do número 1. Assim existirão tantos pedidos com o número "1" quantos forem os cliente da empresa.

Dados a serem guardados:

Pedido: Número, Data, Valor, Data de Entrega, Observação.

Cliente: Nome, telefone, Endereço, Cidade, UF.

Num ambiente como o comportamento ao acessar os pedidos deverá ser o seguinte: "Onde está o pedido número X de tal cliente ? " Ou seja, o ambiente de negócios aponta que, para conhecermos bem um pedido, precisamos saber qual o cliente a quem ele está associado. Portanto o modelo de dados para este caso toma a forma apresentada na Figura 10.

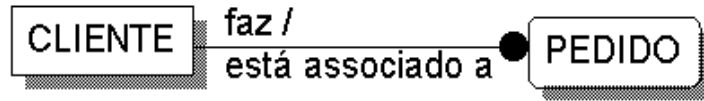


Figura 10. Relacionamento identificante entre duas entidades (visão de entidades).

Lê-se da seguinte forma: um cliente faz 0, 1 ou mais pedidos ou, no sentido inverso, que cada pedido está associado a um e a só um cliente.

Este modelo diz que para a identificação de um pedido implica na identificação prévia do cliente a quem ele está associado. Assim, *a entidade cliente implica na identificação da entidade pedido*. Isto tem uma repercussão importante nos atributos da entidade PEDIDO.

Primeiro, vamos determinar as chaves-primárias para estas entidades. Um modelo praticável seria o expresso na Figura 11.

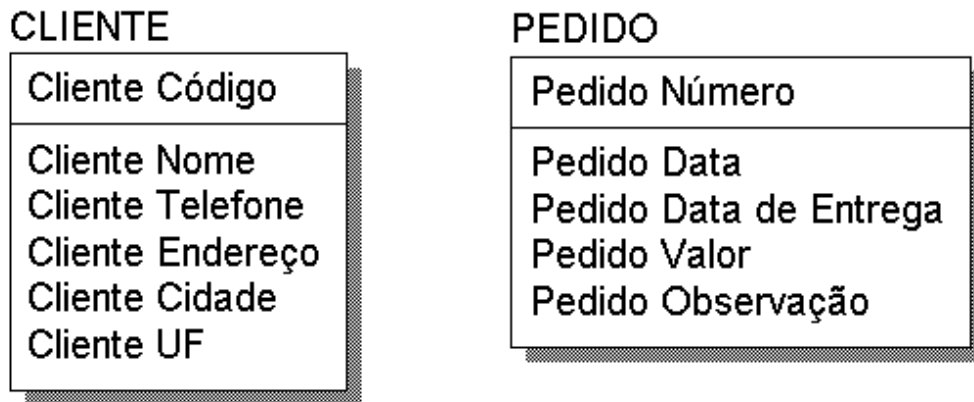


Figura 11. Entidades CLIENTE e PEDIDO com as chaves-primárias determinadas.

Conforme pode-se observar, estamos utilizando uma chave artificial para CLIENTE e um atributo natural para PEDIDO. Ao colocarmos o relacionamento identificante entre estas duas entidades, o modelo em visão de entidades/atributos toma a forma da Figura 12.

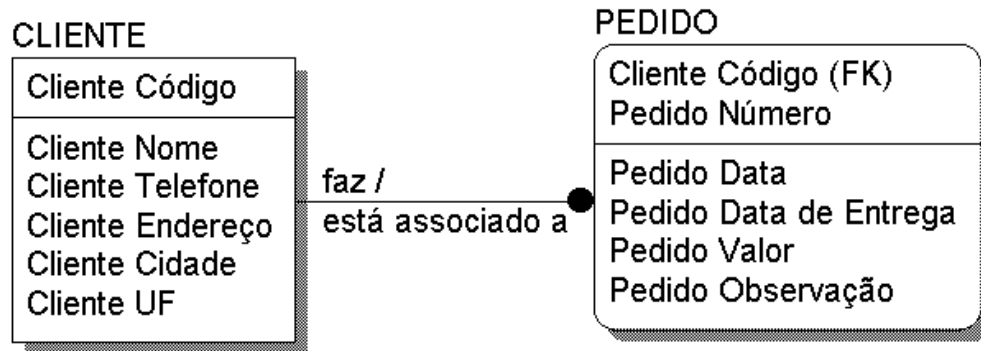


Figura 12. Entidades CLIENTE e PEDIDO com relacionamento identificante.

O que ocorreu foi que o relacionamento identificante obrigou que a chave-primária na entidade CLIENTE fizesse parte da chave-primária da entidade PEDIDO. Assim fica explícito que uma instância na entidade dependente (PEDIDO) não pode existir sem uma instância associada na entidade CLIENTE.

A indicação FK no atributo "Cliente Código" da entidade PEDIDO indica que aquele atributo é uma chave-primária em outra entidade relacionada (no caso, FUNCIONÁRIO). Então, o atributo é denominado uma chave-estrangeira (*Foreign Key*), ou seja, uma chave que “migrou” para a outra entidade devido ao relacionamento.

É importante que se perceba que os relacionamentos são próprios do ambiente de negócios. Não é o analista que "escolhe" arbitrariamente qual relacionamento irá utilizar, mas as informações colhidas no ambiente, principalmente as regras sobre as quais o empreendimento funciona, apontarão a natureza do relacionamento.

2.3.2 RELACIONAMENTO NÃO IDENTIFICANTE/OBRIGATÓRIO

O relacionamento entre duas entidades E1 e E2 (neste sentido) é dito *não identificante/obrigatório* quando uma instância da entidade E2 não precisa de uma instância na entidade E1 para ser identificado, porém o ambiente de negócios obriga que exista uma instância de E1 associada à instância de E2.

Para ficar claro, imagine o seguinte cenário: “Estamos modelando as informações de uma clínica veterinária. Esta clínica precisa, obviamente, saber os dados do animal (cliente) sob seus cuidados. Imagine ainda que esta clínica em particular atende somente a animais que ficam em canis pela cidade. Ou seja, *todo animal atendido pela clínica está, necessariamente, em algum canil da cidade*”. Neste caso, o modelo de dados tomaria a forma da Figura 13.

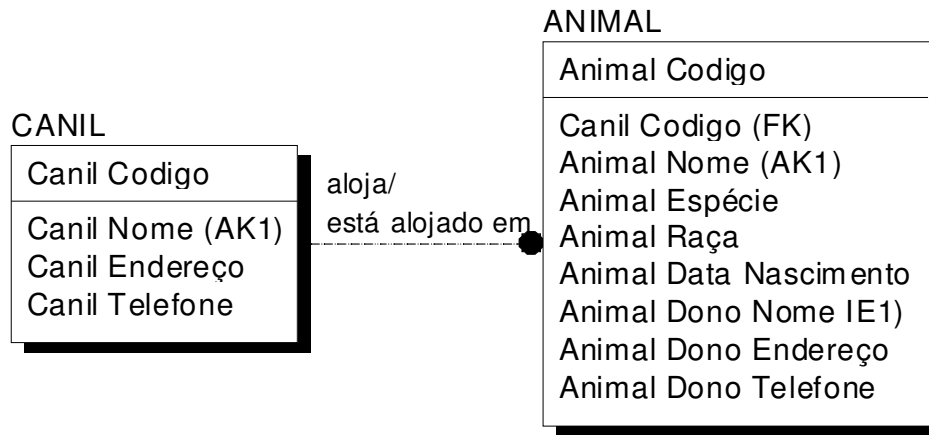


Figura 13. Relacionamento Não Identificante/Obrigatório.

Observe que a chave-primária da entidade CANIL migrou da entidade ANIMAL. O atributo torna-se então uma chave-estrangeira (FK). Como o relacionamento é “não identificante”, a chave migrou para a área de atributos não-chave.

A opção pelo relacionamento não identificante vem do fato de que não seria muito prático ter que saber qual o canil em que um determinado animal se encontra para que a “ficha cadastral” do animal fosse encontrada. Assim, a informação sobre o canil em que se encontra o animal apenas agrega valor ao cadastro do animal, mas não é necessária para se identificar o animal.

2.3.3 RELACIONAMENTO NÃO IDENTIFICANTE, NÃO OBRIGATÓRIO

O relacionamento entre duas entidades E1 e E2 (neste sentido) é dito *não identificante/não obrigatório* quando uma instância da entidade E2 não precisa de uma instância na entidade E1 para ser identificado e ainda o ambiente de negócios não impõe nenhuma obrigação que exista uma instância de E1 associada à instância de E2.

Utilizando o mesmo cenário descrito na seção 2.3.2, considere-se agora que a clínica estendeu seu atendimento para além dos canis, atendendo animais de particulares. Assim sendo, nem todos os animais atendidos pela clínica estão em canis. O modelo toma então a seguinte forma:

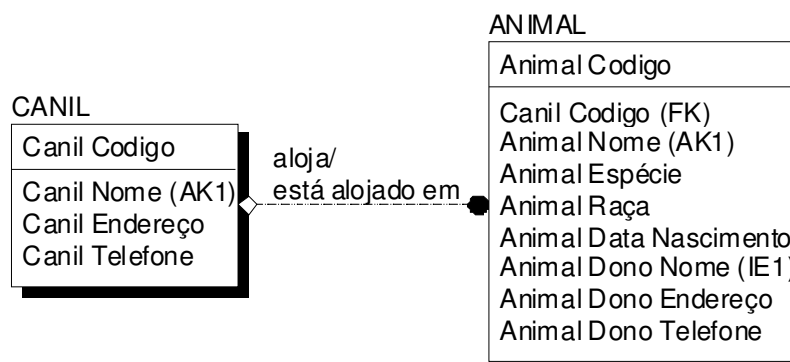


Figura 14. Relacionamento não identificante/não Obrigatório.

O que este modelo diz é que uma instância da entidade ANIMAL não precisa necessariamente estar associada a uma instância da entidade CANIL. Se um animal está alojado em algum canil,

esta informação estará registrada. Se um animal fica normalmente com seu dono, então o animal não possui as informações sobre CANIL em sua ficha cadastral.

2.3.4 RELACIONAMENTO MUITOS-PARA-MUITOS (*Many-to-Many*)

O relacionamento muitos-para-muitos surge em relações como a da Figura 5, onde um médico pode prestar atendimento em várias clínicas e cada clínica pode possuir vários médicos.

Entretanto este tipo de relacionamento, apesar de representável num DER, não é possível de ser implementado na prática. Para que possa ser implementado, este tipo de relacionamento tem que ser “resolvido”.

Suponha que temos as seguintes relações entre os conjuntos médico e clínica:

Médico		Clínica
Dr. Mauro	→	Cirúrgica I
Dr. César	→	Cirúrgica II
Dr. Luís	→	Pediátrica
Dr. Antônio	→	Queimados

Tabela 2. Exemplo de relacionamento muitos-para-muitos para um caso específico de relacionamento entre Médico e Clínica.

Se tentarmos relacionar estes dados através das tabelas MÉDICO e CLÍNICA, como na Figura 5, veríamos que tais relações não seriam representáveis. Na melhor das hipóteses, seríamos obrigados a inserir alguma redundância nas informações.

A resolução deste problema é feita da seguinte forma:

- Cria-se uma nova entidade que relacione as informações entre as entidades da relação muitos-para-muitos através de relações binárias 1-para-muitos;
- Cria-se o relacionamento apropriado entre as tabelas pré-existentes e a tabela nova;
- Promove-se a “migração” das chaves;

É importante certificar-se que as informações do ambiente de negócios foram representadas. No caso apresentado, o DER toma a forma da Figura 16.



Figura 15. Resolução de um relacionamento muitos-para-muitos (visão de entidades).

Na entidade MÉDICO POR CLÍNICA são inseridas as informações que dependem tanto do médico como da clínica, como por exemplo o horário de atendimento daquele médico naquela clínica.

Experimente por exemplo preencher as tabelas com informações para verificar como as relações da Tabela 2 são representadas.

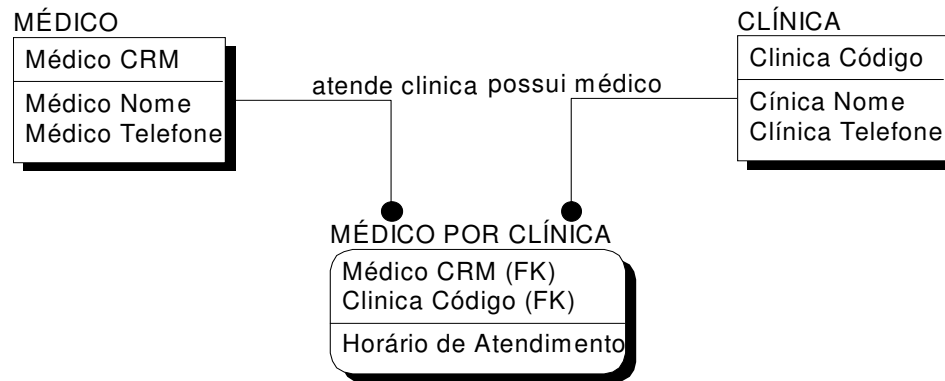


Figura 16. Resolução de um relacionamento muitos-para-muitos (visão de entidade/atributos).

2.3.4.1 EXERCÍCIO 01

Como primeira atividade em modelagem de dados, vamos estudar o documento abaixo, que supõe-se obtido durante a etapa de levantamento de informações em um ambiente de negócios ligado a vendas no varejo.

NOTA FISCAL DE VENDA AO CONSUMIDOR		Nº.	DATA - _/_/	
CLIENTE				
Nome _____				
Endereço: _____				
Cidade _____				UF: _____
CEP _____				
PRODUTOS				
Código	Descrição	Quantidade	Valor Unitário	Total
			TOTAL DA NOTA	

Com base nas informações do documento acima, identifique as entidades e seus atributos, representando-os em notação IDEF1X (ambos em visão de entidade/atributos).