

# Modelagem de Dados com MER

[Entidades atributos e conjunto de entidades](#)

[Relacionamentos e Conjunto de Relacionamentos](#)

[Recursos Adicionais do Modelo ER](#)

[Restrições de Chave](#)

[Restrições de Chave para Relacionamentos Ternários](#)

[Restrições de Participação](#)

[Entidades Fracas](#)

[Hierarquias de Classe](#)

[Agregação](#)

[Projeto Conceitual com o Modelo ER](#)

[Entidade versus Atributo](#)

[Entidade versus Relacionamento](#)

[??Relacionamentos Binários versus Relacionamentos Ternários](#)

[Agregação versus Relacionamentos Ternários](#)

## Entidades atributos e conjunto de entidades

- **Entidades** são objetos do mundo real que podem ser identificados, como brinquedos ou departamentos.
- Entidades são descritas por **atributos**, que refletem o nível de detalhe desejado, como nome e CPF.
- Cada atributo tem um **domínio (tipo)** de valores possíveis, como uma cadeia de caracteres ou um intervalo numérico.
- Para identificar única e distintamente uma entidade, é escolhida uma **chave primária**, como o CPF de um funcionário.
  - Em uma entidade pode haver mais de uma **chave candidata**, mas apenas uma **chave primária**.
- Uma coleção de entidades semelhantes é chamada de conjunto **de entidades** e agrupam objetos semelhantes.
- Uma representação é ilustrada na Figura 2.1 e mostra um conjunto de **entidades** (como Funcionários - representado por um **retângulo**) com seus **atributos** (cpf, nome e vaga - representado por um símbolo **oval**).
  - Cada atributo da **chave primária** é **sublinhado**.
  - A informação do **domínio** poderia ser listada juntamente com o nome do atributo.

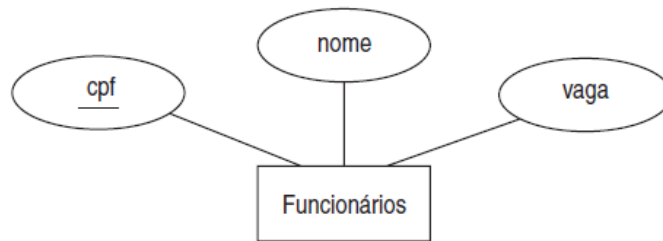
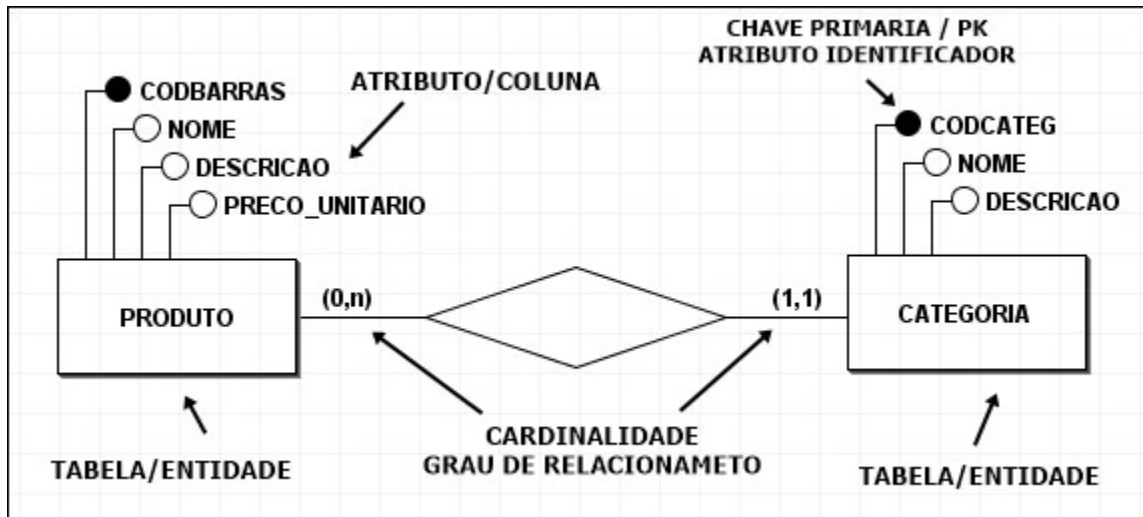


Figura 2.1 O conjunto de entidades Funcionários.



## Relacionamentos e Conjunto de Relacionamentos

- Um **relacionamento** é uma associação entre duas ou mais entidades.
- Como com as entidades, podemos desejar reunir um conjunto de relacionamentos semelhantes em um **conjunto de relacionamentos**.
- Um relacionamento também pode ter **atributos descritivos**. Estes são usados para registrar informações sobre o relacionamento; por exemplo, podemos desejar registrar que Attishoo trabalha no departamento de farmácia a partir de janeiro de 1991. Essa informação é registrada na Figura 2.2, com o atributo, desde, a Trabalha\_em.

Na Figura 2.2 ilustramos o conjunto de relacionamentos Trabalha\_em, no qual cada relacionamento indica um departamento em que um funcionário trabalha. Observe que diversos conjuntos de relacionamentos podem envolver os mesmos conjuntos de entidades. Por exemplo, poderíamos também ter um relacionamento Gerência envolvendo Funcionários e Departamentos. Cada relacionamento Trabalha\_em deve ser identificado univocamente pela combinação de cpf de funcionário e id-depto de departamento. Assim, para um determinado par funcionário-departamento, não podemos ter mais de um valor desde associado.

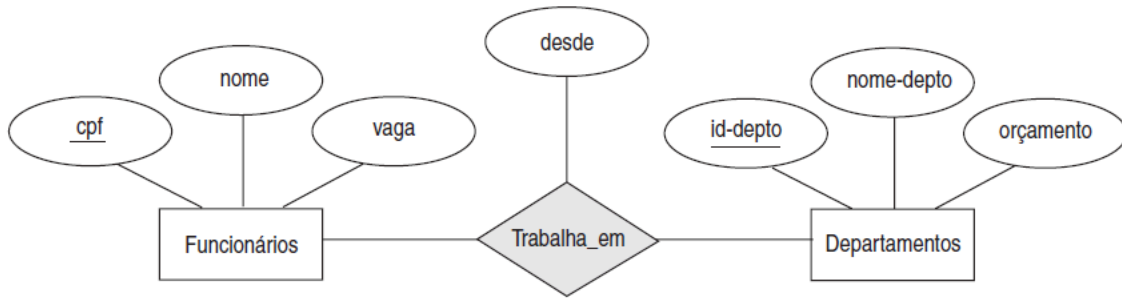


Figura 2.2 O conjunto de relacionamentos *Trabalha\_em*.

Uma **instância** de um conjunto de relacionamentos é um conjunto de seus relacionamentos. Intuitivamente, uma instância pode ser considerada o ‘retrato’ do conjunto de relacionamentos em determinado momento. Uma instância do conjunto de relacionamentos *Trabalha\_em* é ilustrada na Figura 2.3. Cada entidade *Funcionários* é denotada pelo seu *cpf*, e cada entidade *Departamentos* é denotada pela sua *id-depto*, por simplicidade. O valor *desde* é ilustrado ao lado de cada relacionamento.

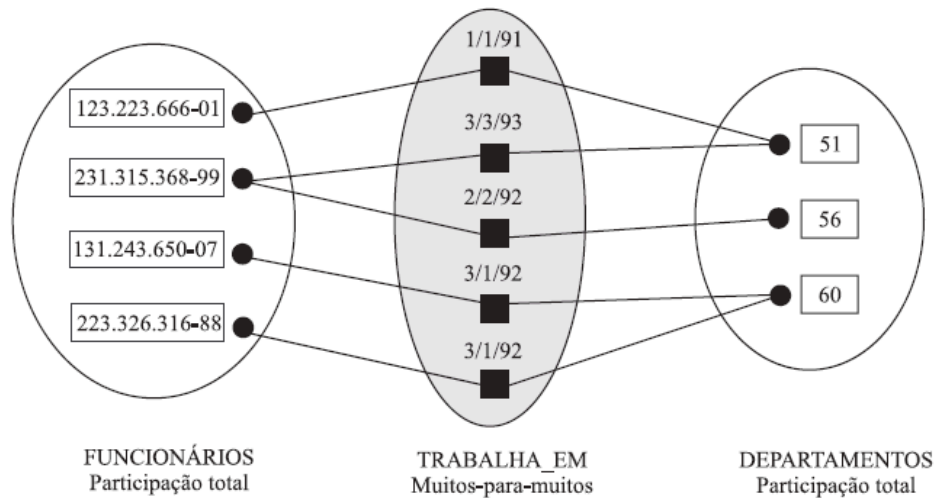


Figura 2.3 Uma instância do conjunto de relacionamentos *Trabalha\_em*.

Suponha que cada departamento tenha escritórios em diversas localidades e desejamos registrar as localidades nas quais cada funcionário trabalha. Esse relacionamento é **ternário** porque devemos registrar uma associação entre um funcionário, um departamento e uma localidade. O diagrama ER é ilustrado na Figura 2.4.

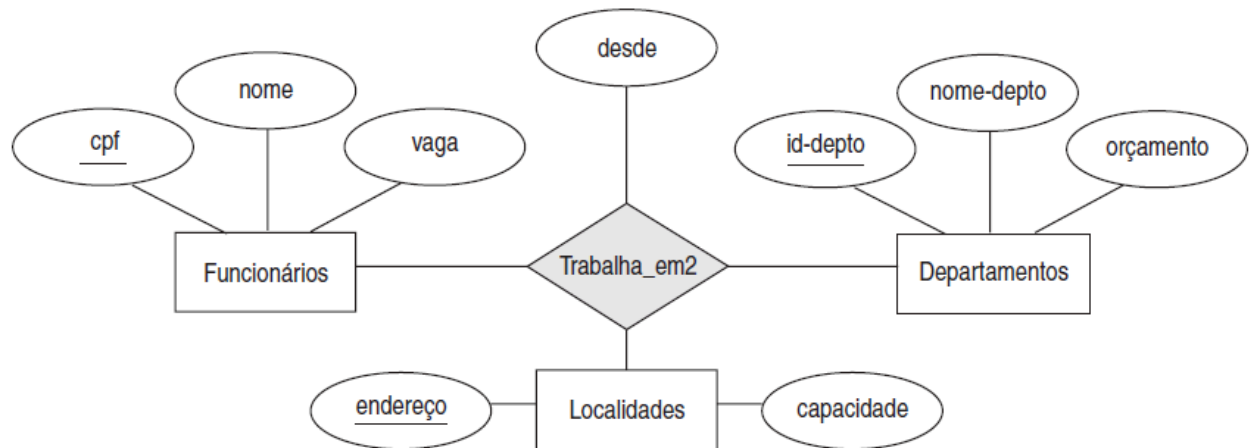


Figura 2.4 Um conjunto de relacionamentos Ternário.

Um relacionamento pode envolver duas entidades no mesmo conjunto de entidades. Por exemplo, considere o conjunto de relacionamentos *Reporta\_a* ilustrado na Figura 2.5. Como os funcionários reportam a outros funcionários, todo relacionamento em *Reporta\_a* é da forma (func1,func2), sendo que func1 e func2 são entidades de *Funcionários*. Entretanto, eles desempenham **papéis** diferentes: func1 reporta ao funcionário supervisor func2, o que é refletido nos **indicadores de papel** *supervisor* e *subordinado* na Figura 2.5.



Figura 2.5 O conjunto de relacionamento *Reporta\_a*.

## Recursos Adicionais do Modelo ER

Examinaremos agora alguns dos construtores do modelo ER que nos permitem descrever algumas propriedades sutis dos dados. A expressividade do modelo ER é uma boa explicação para seu uso tão difundido.

### Restrições de Chave

O funcionário 231.315.368-99 trabalha no Departamento 51 desde 3/3/93 e no Departamento 56 desde 2/2/92. O Departamento 51 tem dois funcionários, Figura 2.3. Considere agora um outro conjunto de

relacionamentos chamado Gerência entre os conjuntos de entidades Funcionários e Departamentos, tal que cada departamento tenha no máximo um gerente, embora um mesmo funcionário possa gerenciar mais do que um departamento. A restrição de que cada departamento tem no máximo um gerente é um exemplo de uma **restrição de chave**, e isso implica que cada entidade Departamentos apareça em no máximo um relacionamento Gerencia em qualquer instância permitida de Gerência. Essa restrição é indicada no diagrama ER da Figura 2.6 utilizando-se uma **seta** de Departamentos a Gerência.

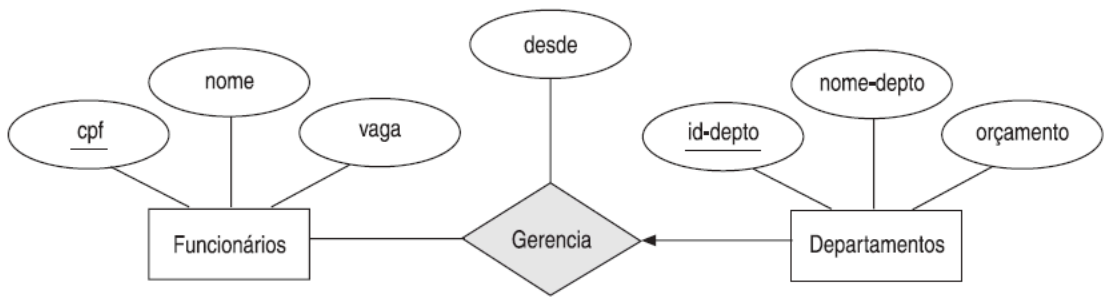


Figura 2.6 Restrição de chave em Gerencia.

Uma instância do conjunto de relacionamentos Gerencia é ilustrada na Figura 2.7.

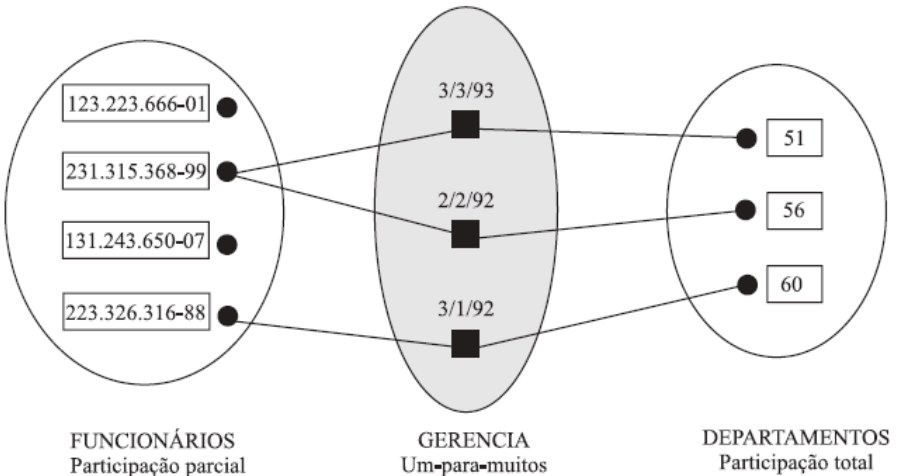


Figura 2.7 Uma instância do conjunto de relacionamentos Gerencia.

- Um conjunto de relacionamentos como Gerência é considerado **um-para-muitos (1:N)**, indicando que **um funcionário** pode estar associado a **muitos departamentos** (na qualificação de gerente), enquanto cada departamento pode estar associado a, no máximo, um funcionário como seu gerente.
- Em contraste, o conjunto de relacionamentos Trabalha\_em, no qual é permitido que **um funcionário** trabalhe em **diversos departamentos** e que **um departamento** tenha **diversos funcionários**, é considerado **muitos-para-muitos (N:N)**.
- Se adicionarmos ao conjunto de relacionamentos Gerencia a restrição de que cada funcionário pode gerenciar no máximo um departamento, sendo que esta restrição seria indicada acrescentando-se uma seta de Funcionários a Gerência na Figura 2.6, teremos um conjunto de relacionamentos **um-para-um (1:1)**.

## Restrições de Chave para Relacionamentos Ternários

Podemos estender esta convenção — e o conceito subjacente de restrição de chave — aos conjuntos de relacionamentos envolvendo três ou mais conjuntos de entidades. Uma instância do conjunto de relacionamentos Trabalha\_em3 é ilustrada na Figura 2.9. Observe que cada departamento pode estar associado a diversos funcionários e localidades e que cada localidade pode estar associada a diversos departamentos e funcionários; entretanto, cada funcionário está associado a um único departamento e localidade.

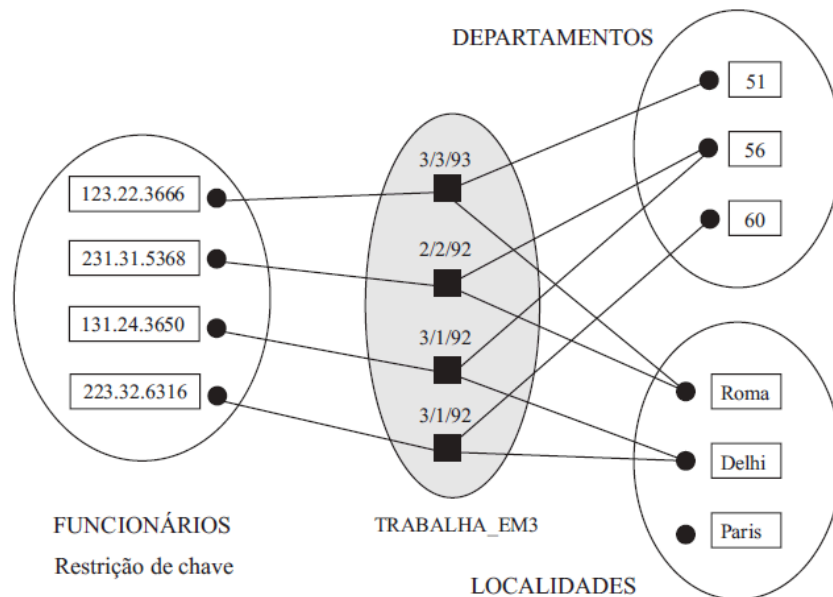


Figura 2.9 Uma instância de Trabalha\_em3.

## Restrições de Participação

A restrição de chave em Gerência nos informa que um departamento tem no máximo um gerente. Uma dúvida natural é questionar se todo departamento tem um gerente. Consideremos que é exigido que todo departamento tenha um gerente. Esse requisito é um exemplo de **restrição de participação**;

- A participação do conjunto de entidades Departamentos no conjunto de relacionamentos Gerência é considerada **total**.
- Uma participação é dita **parcial** em Funcionários e Gerência é parcial, uma vez que nem todo funcionário gerencia um departamento.

Retomando o conjunto de relacionamentos Trabalha\_em, é natural esperarmos que cada funcionário trabalhe em, no mínimo, um departamento e que cada departamento tenha, no mínimo, um funcionário. Isso significa que a participação tanto de Funcionários quanto de Departamentos em Trabalha\_em é **total**. O diagrama ER da Figura 2.10 ilustra ambos os conjuntos de relacionamentos Gerência e Trabalha\_em e todas as restrições determinadas.

- Se a participação de um conjunto de entidades em um conjunto de relacionamentos for **total**, os dois são conectados por uma linha grossa;

- independentemente, a presença de uma **seta** indica uma **restrição de chave**.

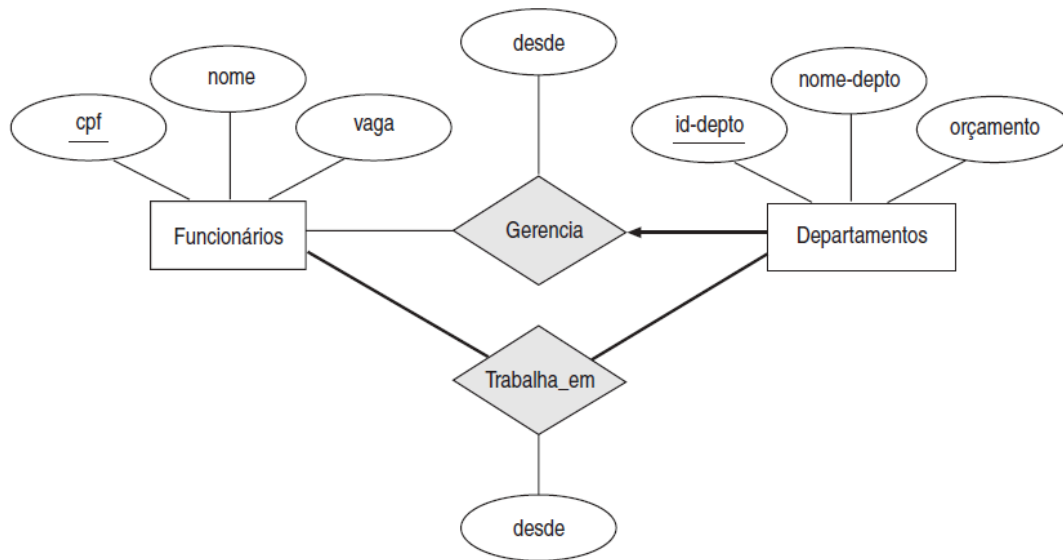


Figura 2.10 Gerencia e Trabalha\_em.

## Entidades Fracas

Uma entidade fraca pode ser univocamente identificada **apenas** se considerarmos alguns dos seus atributos em conjunto com a **chave primária** de uma outra entidade, que é chamada **proprietária identificadora**.

- O conjunto de entidades proprietárias e o conjunto de entidades fracas devem participar em um conjunto de relacionamentos **um-para-muitos** (uma entidade proprietária está associada a uma ou mais entidades fracas, mas cada entidade fraca tem uma única proprietária). Esse relacionamento é chamado **conjunto de relacionamentos identificadores** do conjunto de entidades fracas.
- O conjunto de **entidades fracas** deve ter **participação total** no conjunto de **relacionamentos identificadores**.

O conjunto de entidades fracas Dependentes e seu relacionamento com Funcionários são ilustrados na Figura 2.11. A **participação total** de Dependentes em Apólice é indicada pela ligação entre eles com uma **linha grossa**. A **seta** de Dependentes a Apólice indica que cada entidade de Dependentes aparece em, no máximo, um relacionamento Apólice. Para ressaltar o fato de que **Dependentes** é uma **entidade fraca** e **Apólice** é seu **relacionamento identificador**, **desenhamos ambos com linhas grossas**. Para indicar que **nomed** é uma **chave parcial** de Dependentes, **sublinhamos o atributo usando uma linha tracejada**. Isso significa que pode muito bem haver dois dependentes com o mesmo valor de **nomed**.

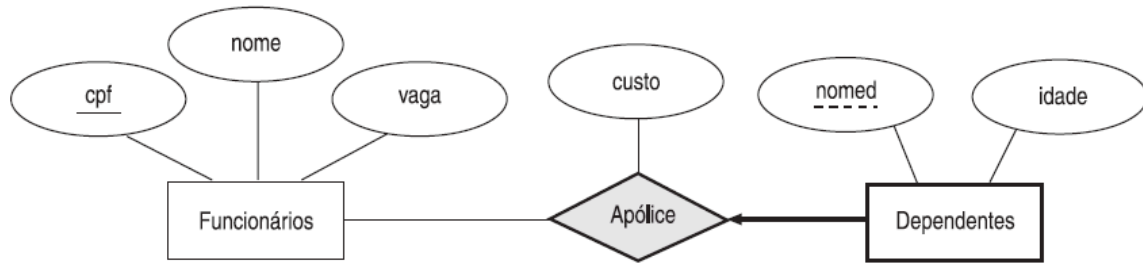


Figura 2.11 Um conjunto de entidades fracas.

## Hierarquias de Classe

Os atributos definidos para uma entidade *Funcion\_Horistas* são os atributos de *Funcionários* mais os de *Funcion\_Horistas*. Dizemos que os atributos do conjunto de entidades *Funcionários* são **herdados** pelo conjunto de entidades *Funcion\_Horistas* e que *Funcion\_Horistas* **É-UM (IS-A)** *Funcionários*. Há uma restrição nas consultas sobre instâncias desses conjuntos de entidades: uma consulta que solicita todas as entidades *Funcionários* deve considerar também todas as entidades *Funcion\_Horistas* e *Funcion\_Contratados*. A Figura 2.12 ilustra a hierarquia de classe.

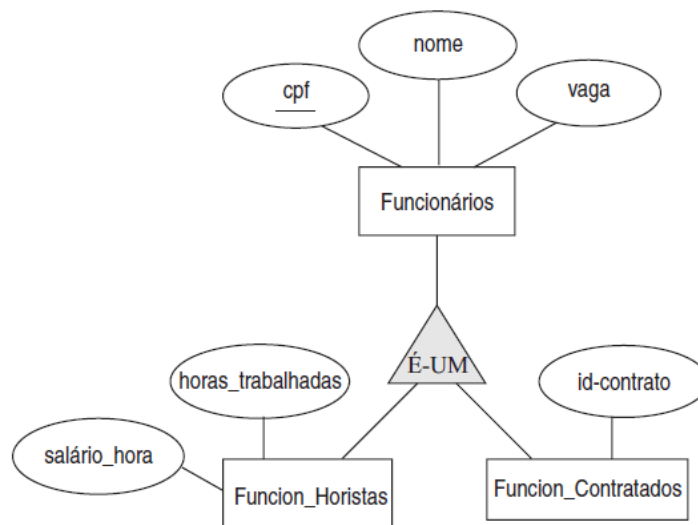


Figura 2.12 Hierarquia de classe.

Uma hierarquia de classe pode ser vista de duas formas:

- *Funcionários* é especializado em subclasses. A **especialização** é o processo de identificar subconjuntos de um conjunto de entidades (a **superclasse**) que compartilham algumas características distinguíveis. Tipicamente, a **superclasse** é definida primeiro (*Funcionários*), as **subclasses** (*Funcion\_Horistas* e *Funcion\_Contratados*) são definidas em seguida, e os atributos específicos da subclasse e os conjuntos de relacionamentos são adicionados depois.
- *Funcion\_Horistas* e *Funcion\_Contratados* são **generalizados** em *Funcionários*. Como um outro exemplo, dois conjuntos de entidades *Barcos\_Motorizados* e *Carros* podem ser generalizados em um



conjunto de entidades Veículos\_Motorizados. A **generalização** consiste em identificar algumas características comuns de uma coleção de conjuntos de entidades e criar um novo conjunto de entidades que contenha as entidades possuindo essas características comuns. Tipicamente, as **subclasses** são definidas primeiro, a **superclasse** é definida em seguida, e quaisquer conjuntos de relacionamentos que envolvam a superclasse são definidos depois.

Podemos especificar dois tipos de restrições com relação a hierarquias **É-UM**, a saber, restrições de sobreposição e de cobertura.

- As **restrições de sobreposição** determinam se duas subclasses podem conter a mesma entidade. Por exemplo, Attishoo pode ele ser tanto uma entidade Funcion\_Contratados como também Funcion\_Seniores? Intuitivamente, sim. Denotamos isso escrevendo '**Funcion\_Contratados SOBREPÕE Funcion\_Seniores**'. Na falta de tal afirmação, consideramos por padrão que os conjuntos de entidades têm restrições de não se sobreporem.
- As **restrições de cobertura** determinam se as entidades das subclasses incluem coletivamente todas as entidades da superclasse. Por exemplo, toda entidade Funcionários deve pertencer a uma de suas subclasses? Intuitivamente, não. Toda entidade Veículos\_Motorizado deve ser ou uma entidade Barcos\_Motorizados ou uma entidade Carros? Intuitivamente, sim; Denotamos isso escrevendo '**Barcos\_Motorizados E Carros COBREM Veículos\_Motorizados**'. Na falta de tal afirmação, consideramos por padrão que não há restrição de cobertura; podemos ter veículos motorizados que não são barcos motorizados nem carros.

Há duas razões básicas para identificar subclasses (por especialização ou generalização):

- Poderíamos acrescentar atributos descritivos que fazem sentido apenas às entidades em uma subclasse. Por exemplo, salário\_hora não faz sentido para uma entidade Funcion\_Contratados, cujo pagamento é determinado por um contrato individual.
- Poderíamos identificar o conjunto de entidades que participam em algum relacionamento. Por exemplo, poderíamos definir o relacionamento Gerencia de forma que os conjuntos de entidades participantes sejam Funcion\_Seniores e Departamentos, para assegurar que apenas os funcionários seniores possam ser gerentes.

## Agregação

Suponha que tenhamos um conjunto de entidades chamado Projetos seja que cada entidade Projetos seja financiada por um ou mais departamentos. O conjunto de relacionamentos Financia captura essa informação. Um departamento que financia um projeto pode designar funcionários para monitorar o financiamento. Intuitivamente, Monitora deve ser um conjunto de relacionamentos que associa um relacionamento Financia (em vez de uma entidade Projetos ou Departamentos) a uma entidade Funcionários. Entretanto, definimos relacionamentos para associar duas ou mais entidades.

Para definir um conjunto de relacionamentos como Monitora, introduzimos um novo recurso do modelo ER, chamado **agregação**, que nos permite indicar que um conjunto de relacionamentos (identificado através de um quadro tracejado) participa de outro conjunto de relacionamentos. Isso é ilustrado na Figura 2.13, com um quadro tracejado ao redor de Financia (e seus conjuntos de entidades participantes) usado para denotar agregação. Algo assim efetivamente nos possibilita tratar Financia como um conjunto de entidades para

propósitos de definição do conjunto de relacionamento Monitora. Quando devemos usar agregação? Intuitivamente, nós a usamos quando é preciso expressar um relacionamento entre relacionamentos.

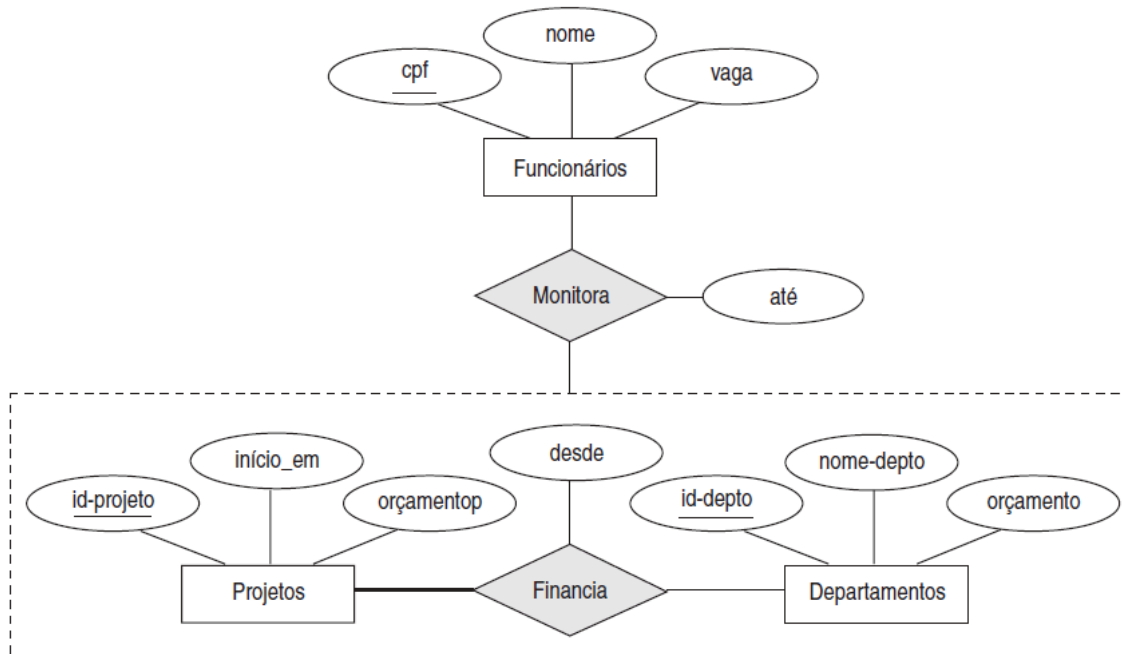


Figura 2.13 Agregação.

## Projeto Conceitual com o Modelo ER

O desenvolvimento de um diagrama ER envolve diversas escolhas, incluindo as seguintes:

- Um conceito deve ser modelado como uma entidade ou um atributo ou um relacionamento?
- Quais são os conjuntos de relacionamentos e seus conjuntos de entidades participantes?
- Devemos usar relacionamentos binários ou ternários?
- Devemos usar agregação?

### Entidade versus Atributo

Para um outro exemplo de quando modelar um conceito como um conjunto de entidades e não como um atributo, considere o conjunto de relacionamentos (chamado Trabalha\_em4) ilustrado na Figura 2.14.

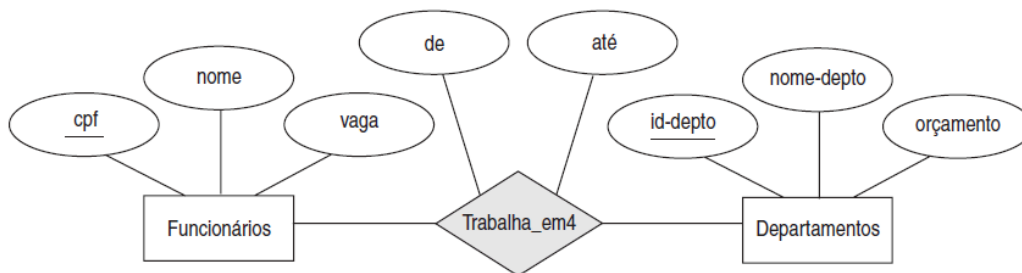


Figura 2.14 O conjunto de relacionamentos Trabalha\_em4.

Intuitivamente, ele registra o intervalo durante o qual um funcionário trabalha para um departamento. Agora suponha que seja possível que um empregado trabalhe em determinado departamento por mais do que um período. Essa possibilidade não é permitida pela semântica do diagrama ER, porque um relacionamento é identificado unicamente pelas entidades participantes. O problema é que desejamos registrar diversos valores para os atributos descritivos de cada instância do relacionamento *Trabalha\_em4*. Podemos tratar esse problema introduzindo um conjunto de entidades chamado, digamos, *Duração*, com atributos *de* e *até*, como ilustrado na Figura 2.15.

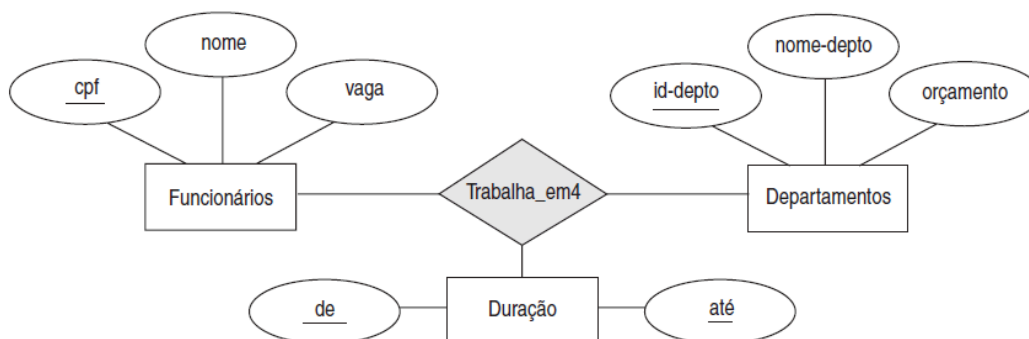


Figura 2.15 O conjunto de relacionamentos Trabalha\_em4.

## Entidade versus Relacionamento

Dado um departamento, Figura 2.16, conhecemos o gerente, assim como a data inicial da gerência e o orçamento para esse departamento. Essa abordagem é natural se assumirmos que um gerente recebe um orçamento arbitrário separado para cada departamento que ele ou ela gerencia. Mas e se o orçamento arbitrário for uma soma que abranja todos os departamentos gerenciados por esse funcionário? Nesse caso, cada relacionamento *Gerencia2* que envolve determinado funcionário terá o mesmo valor no campo *orçamento-a*, gerando um armazenamento redundante da mesma informação. Outro problema com este projeto é ser enganoso; ele sugere que o orçamento está associado ao relacionamento, quando ele está, na verdade, associado ao gerente.

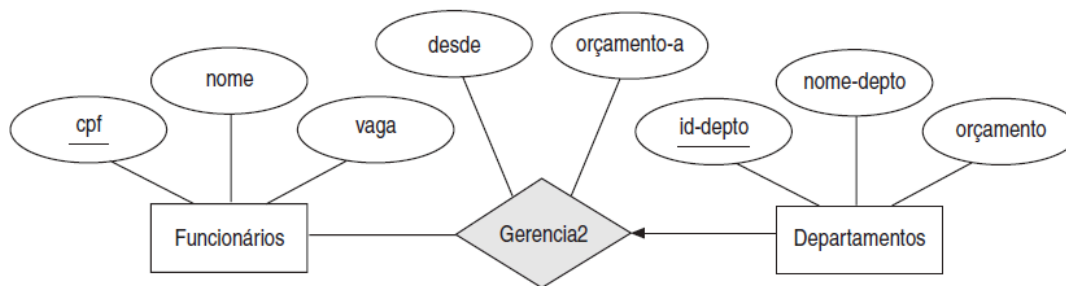


Figura 2.16 Entidade *versus* Relacionamento.

Podemos tratar tais problemas introduzindo um novo conjunto de entidades chamado Gerentes (que pode ser posicionado abaixo de Funcionários na hierarquia É-UM, para ilustrar que todo gerente - **subclasse** - também é um funcionário - **superclasse**). Os atributos desde e orçamento-a agora descrevem uma entidade gerente, como planejado. Como uma variante, enquanto cada gerente tem um orçamento, cada gerente pode ter uma data inicial diferente (como gerente) para cada departamento. Nesse caso, orçamento-a é um atributo de Gerentes, mas desde é um atributo do conjunto de relacionamentos entre gerentes e departamentos.

A natureza imprecisa da modelagem ER pode, assim, dificultar o reconhecimento de entidades subjacentes, e poderíamos associar atributos a relacionamento em vez das entidades apropriadas. Em geral, tais erros conduzem a **armazenamento redundante** da mesma informação e podem causar vários problemas. Discutiremos a **redundância** e seus conseqüentes problemas nos próximos resumos (**Documento 04**), e apresentaremos uma técnica chamada **normalização** para **eliminar a redundância das tabelas**.

## ??Relacionamentos Binários versus Relacionamentos Ternários

Considere o diagrama ER ilustrado na Figura 2.17. Ele modela uma situação na qual um funcionário pode possuir várias apólices de seguro, cada apólice pode ser pertencente a diversos funcionários, e cada dependente pode ser coberto por diversas apólices. Suponha que tenhamos os seguintes requisitos adicionais:

- Uma apólice não pode pertencer em conjunto a dois ou mais funcionários.
- Cada apólice deve pertencer a algum funcionário.
- Dependentes é um conjunto de entidades fracas, e cada entidade dependente é univocamente identificada pelo nome em conjunto com o id-apólice de uma entidade Apólices (que, intuitivamente, cobre o dependente em questão).

O primeiro requisito sugere impormos uma restrição de chave em Apólices com respeito a Cobre, mas esta restrição tem o efeito colateral não intencionado de que uma apólice pode cobrir apenas um dependente. O segundo requisito sugere impormos uma restrição de participação total em Apólices. Essa solução é aceitável se cada apólice cobrir no mínimo um dependente. O terceiro requisito nos força a introduzir um relacionamento identificador binário (em nossa versão de diagramas ER, embora haja versões nas quais esse não seja o caso).

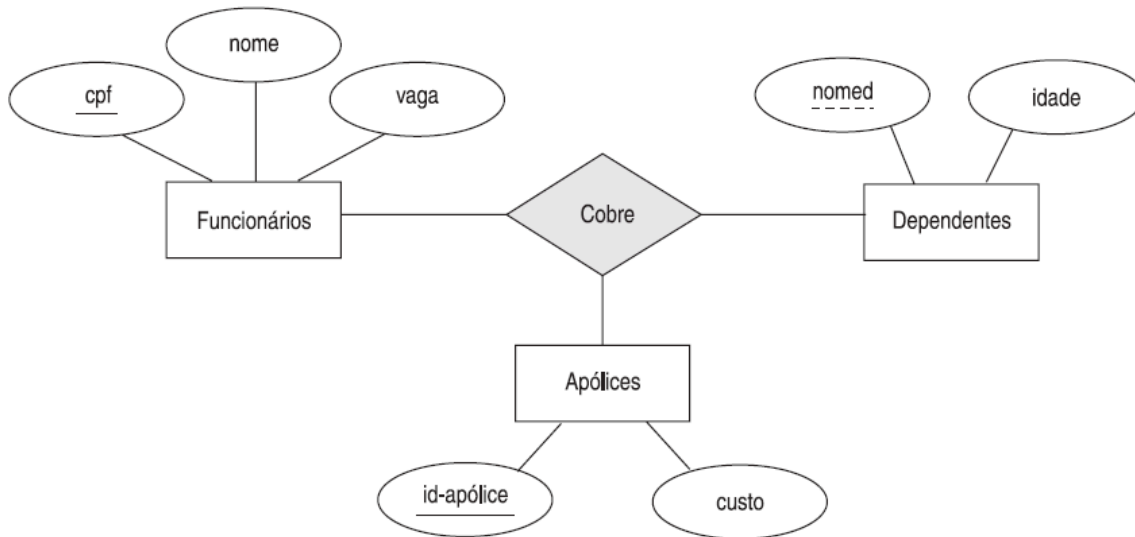


Figura 2.17 Apólices como um Conjunto de Entidades.

Mesmo ignorando o terceiro requisito, a melhor maneira de modelar essa situação é usar dois relacionamentos binários, como ilustrado na Figura 2.18. Esse exemplo realmente tem dois relacionamentos envolvendo Apólices, e nossa tentativa de usar um único relacionamento ternário (Figura 2.17) é inadequada.

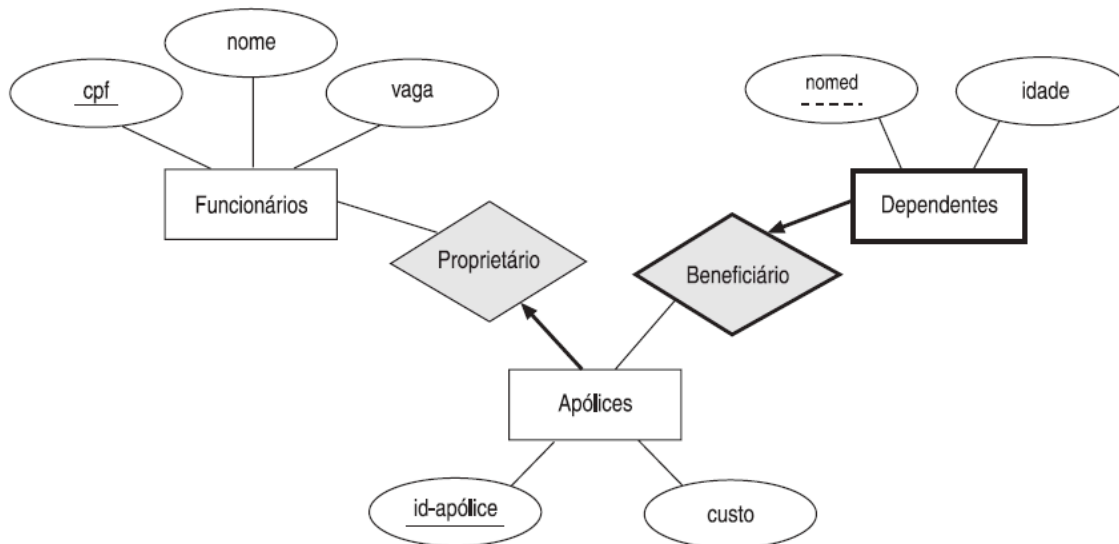


Figura 2.18 Apólice reformulada.

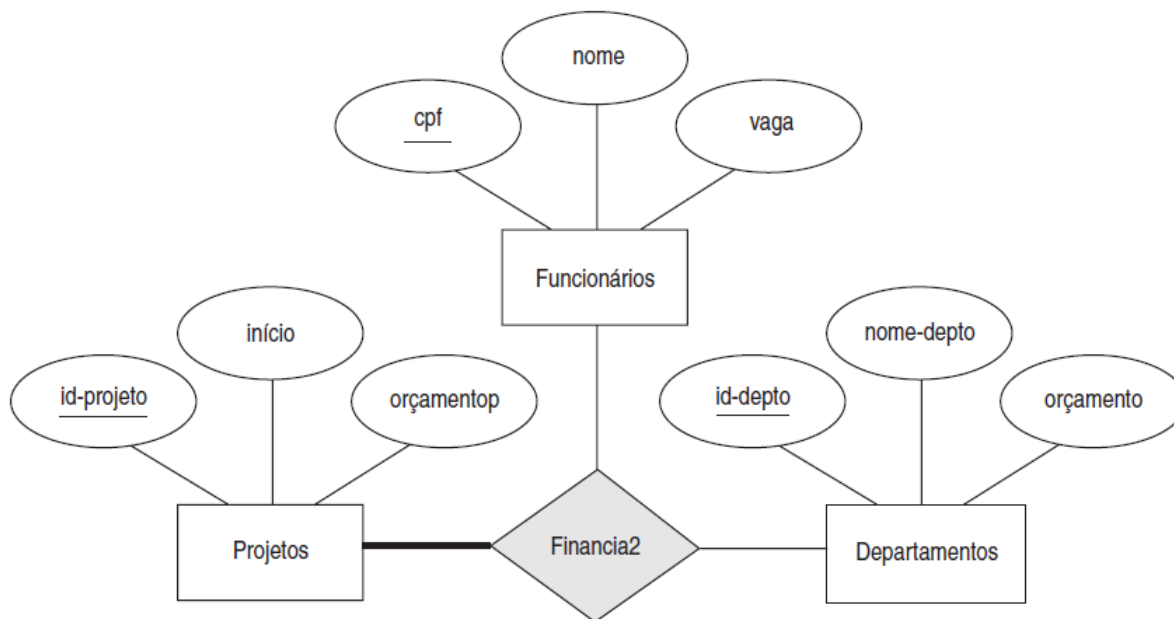
Como um exemplo típico de relacionamento ternário, considere os conjuntos de entidades Suprimentos, Fornecedores e Departamentos, e um conjunto de relacionamentos Contratos (com o atributo descritivo qtidade) que envolve todos eles. Um contrato especifica que um fornecedor fornecerá (alguma quantidade de) um suprimento a um departamento. Esse relacionamento não pode ser adequadamente capturado por uma coleção de relacionamentos binários (sem o uso de agregação). Com os relacionamentos binários, podemos denotar que um fornecedor 'pode fornecer' determinados suprimentos, que um departamento 'necessita' de alguns suprimentos, ou que um departamento 'lida com' determinado fornecedor. Nenhuma combinação desses relacionamentos expressa o significado de um contrato adequadamente por, no mínimo, duas razões:

- Os fatos de que o fornecedor F pode fornecer o suprimento S, que o departamento D precisa do suprimento S, e que D comprar de F não necessariamente implicam que o departamento D compra, de fato, o suprimento S do fornecedor F!
- Não podemos representar o atributo qtidade de um contrato de forma limpa.

## Agregação versus Relacionamentos Ternários

Conforme observamos na Seção 2.4.5, a escolha entre agregação ou um relacionamento ternário é determinada principalmente pela existência de um relacionamento que associa um conjunto de relacionamentos a um conjunto de entidades (ou segundo conjunto de relacionamentos). A escolha também pode ser guiada por determinadas restrições de integridade que desejamos expressar. Por exemplo, considere o diagrama ER ilustrado na Figura 2.13. De acordo com esse diagrama, um projeto pode ser financiado por qualquer número de departamentos, um departamento pode financiar um ou mais projetos, e cada financiamento é monitorado por um ou mais funcionários. Se não necessitamos registrar o atributo até de Monitora, então poderíamos usar razoavelmente um relacionamento ternário, digamos, Financia2, como ilustrado na Figura 2.19.

Considere a restrição de que cada financiamento (de um projeto por um departamento) seja monitorado por no máximo um funcionário. Não podemos expressar essa restrição em termos do conjunto de relacionamentos Financia2. Por outro lado, podemos expressar facilmente a restrição desenhando uma **seta** do relacionamento agregado Financia ao relacionamento Monitora na Figura 2.13. Assim, a presença de tal restrição serve como outra razão para usar agregação em vez de um conjunto de relacionamento ternário.



**Figura 2.19** Usando um Relacionamento Ternário em vez de uma Agregação.

Referencias:

- Modelagem de Dados com o Modelo Entidade-Relacionamento - Peter Chen (2ª edição)
- Fundamentos de Sistemas de Gerenciamento de Banco de Dados - Elmasri & Navathe (7ª edição)
- Sistemas de Gerenciamento de Banco de Dados - Ramakrishnan Gehrke (3ª edição)

#### Isenção de Responsabilidade:

Os autores deste documento não reivindicam a autoria do conteúdo original compilado das fontes mencionadas. Este documento foi elaborado para fins educativos e de referência, e todos os créditos foram devidamente atribuídos aos respectivos autores e fontes originais.

Qualquer utilização comercial ou distribuição do conteúdo aqui compilado deve ser feita com a devida autorização dos detentores dos direitos autorais originais. Os compiladores deste documento não assumem qualquer responsabilidade por eventuais violações de direitos autorais ou por quaisquer danos decorrentes do uso indevido das informações contidas neste documento.

Ao utilizar este documento, o usuário concorda em respeitar os direitos autorais dos autores originais e isenta os compiladores de qualquer responsabilidade relacionada ao conteúdo aqui apresentado.