

Modelagem de dados usando o ModeloEntidade-Relacionamento (ER)

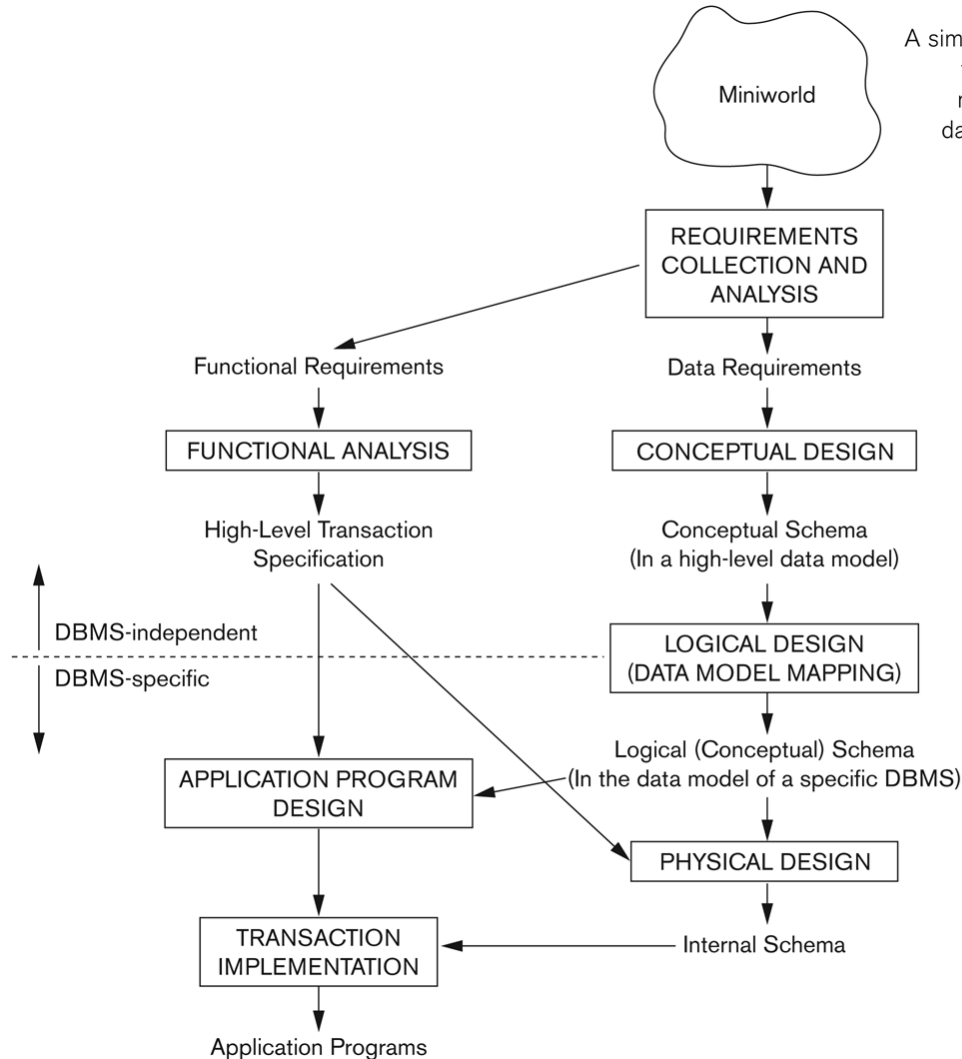
Chapter Outline

- Visão geral do processo de design do banco de dados
- Exemplo de aplicativo de banco de dados (EMPRESA)
- Conceitos do Modelo ER
 - Entidades e Atributos
 - Tipos de entidades, conjuntos de valores e atributos-chave
 - Relacionamentos e tipos de relacionamento
 - Tipos de entidades fracas
- Funções e Atributos em Tipos de Relacionamento
- Diagramas ER - Notação
- Diagrama ER para o Esquema da EMPRESA
- Notações Alternativas - diagramas de classes UML, outros
- Relacionamentos de Grau Superior

Overview of Database Design Process

Figure 3.1

A simplified diagram to illustrate the main phases of database design.



Methodologies for Conceptual Design

- Diagramas de relacionamento com entidades (ER) (este capítulo)
- Diagramas aprimorados de relacionamento de entidade (EER) (Próximo Capítulo)
- Uso de ferramentas de design na indústria para projetar e documentar projetos de grande escala
- Os diagramas de classe UML (Unified Modeling Language) são populares na indústria para documentar projetos conceituais de bancos de dados

Example COMPANY Database

- Precisamos criar um projeto de esquema de banco de dados com base nos seguintes requisitos (simplificados) do banco de dados da EMPRESA:
 - A empresa é organizada em DEPARTAMENTOOS. Cada DEPARTAMENTOOo tem um nome, número e um funcionário que gerencia o DEPARTAMENTOOo. Acompanhamos a data de início do gerente do DEPARTAMENTOOo. Um DEPARTAMENTOOo pode ter vários locais.
 - Cada DEPARTAMENTOOo controla vários PROJETOS. Cada projeto tem um nome único, um número exclusivo e está localizado em um único local.

Example COMPANY Database (Continued)

- O banco de dados armazenará o número de seguridade social, endereço, salário, sexo e data de nascimento de cada funcionário do EMPLOYEE.
- Cada funcionário trabalha para um DEPARTAMENTO, mas pode trabalhar em vários projetos.
- O DB acompanhará o número de horas por semana que um funcionário trabalha atualmente em cada projeto.
- É necessário acompanhar o supervisor direto de cada funcionário.
- Cada empregado pode ter um número de DEPENDENTES.
- Para cada dependente, o DB mantém um registro de nome, sexo, data de nascimento e relacionamento com o funcionário.

Types of Attributes (1)

- **Simples**
 - Cada entidade possui um único valor atômico para o atributo. Por exemplo, SSN ou Sex.
- **Composto**
 - O atributo pode ser composto por vários componentes. Por exemplo:
 - Endereço (Apt #, House #, Street, Cidade, Estado, ZipCode, País) ou
 - Nome (nome, nome do meio, sobrenome)
 - Composição: pode formar uma hierarquia onde alguns componentes são compostos.
- **Multivalorado**
 - Uma entidade pode ter vários valores para esse atributo. Por exemplo, cor de um CAR ou PreviousDegrees de um aluno.
 - Denotado como {Cor} ou {PreviousDegrees}.

ER Model Concepts

■ Entidades e Atributos

- Entidade é um conceito básico para o modelo ER. Entidades são coisas específicas ou objetos no mini-mundo que são representados no banco de dados.
 - Por exemplo, o EMPREGADO John Smith, o DEPARTAMENTOO de Pesquisa, o PROJETO ProductX
- Atributos são propriedades usadas para descrever uma entidade.
 - Por exemplo, uma entidade EMPLOYEE pode ter os atributos Name, SSN, Address, Sex, BirthDate
- Uma entidade específica terá um valor para cada um dos seus atributos.
 - Por exemplo, uma entidade de funcionário específica pode ter Nome = 'John Smith', SSN = '123456789', Endereço = '731, Fondren, Houston, TX', Sexo = 'M', Data de nascimento = '09 -JAN-55 '
- Cada atributo tem um conjunto de valores (ou tipo de dados) associado a ele. número inteiro, string, data, tipo enumerado,...

Types of Attributes (1)

- **Simples**
 - Cada entidade possui um único valor atômico para o atributo. Por exemplo, SSN ou Sex.
- **Composto**
 - O atributo pode ser composto por vários componentes. Por exemplo:
 - Endereço (Apt #, House #, Street, Cidade, Estado, ZipCode, País) ou
 - Nome (nome, nome do meio, sobrenome)
 - Composição: pode formar uma hierarquia onde alguns componentes são compostos.
- **Multivalorado**
 - Uma entidade pode ter vários valores para esse atributo. Por exemplo, cor de um CAR ou PreviousDegrees de um aluno.
 - Denotado como {Cor} ou {PreviousDegrees}.

Types of Attributes (2)

- Em geral, os atributos compostos e multivalorados podem ser aninhados arbitrariamente em qualquer número de níveis, embora isso seja raro.
 - Por exemplo, PreviousDegrees de um ALUNO é um atributo composto de vários valores denotado por {PreviousDegrees (College, Year, Degree, Field)}
 - Vários valores de PreviousDegrees podem existir
 - Cada um tem quatro atributos de subcomponente:
 - Faculdade, ano, grau, campo

Example of a composite attribute

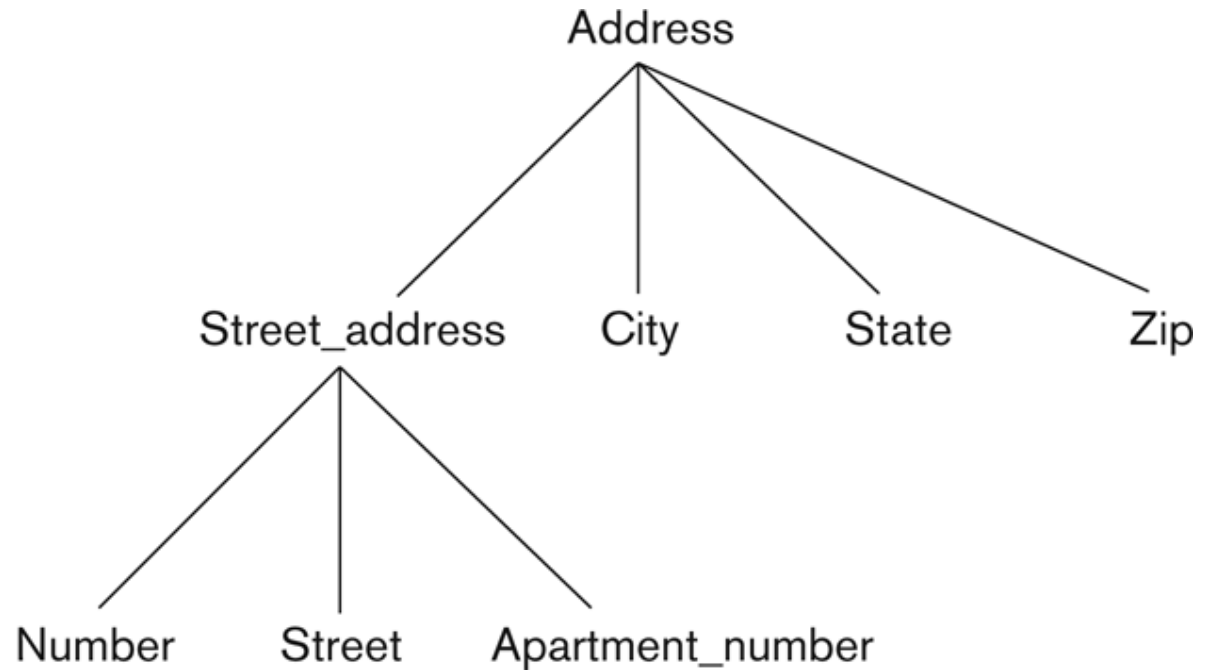


Figure 3.4

A hierarchy of composite attributes.

Entity Types and Key Attributes (1)

- Entidades com os mesmos atributos básicos são agrupadas ou digitadas em um tipo de entidade.
 - Por exemplo, o tipo de entidade EMPREGADO e PROJETO.
- Um atributo de um tipo de entidade para o qual cada entidade deve ter um valor exclusivo é chamado de atributo chave do tipo de entidade.
 - Por exemplo, SSN do EMPLOYEE.

Entity Types and Key Attributes (2)

- Um atributo-chave pode ser composto.
 - VehicleTagNumber é uma chave do tipo de entidade CAR com componentes (Number, State).
- Um tipo de entidade pode ter mais de uma chave.
 - O tipo de entidade CAR pode ter duas chaves:
 - VehicleIdentificationNumber (popularmente chamado de VIN)
 - VehicleTagNumber (Number, State), também conhecido como número da placa.
- Em geral, escolhe-se uma das chaves para sublinhar (Nota: isso é diferente do esquema relacional onde apenas uma “chave primária é sublinhada).

Entity Types and Key Attributes (2)

- Um atributo-chave pode ser composto.
 - VehicleTagNumber é uma chave do tipo de entidade CAR com componentes (Number, State).
- Um tipo de entidade pode ter mais de uma chave.
 - O tipo de entidade CAR pode ter duas chaves:
 - VehicleIdentificationNumber (popularmente chamado de VIN)
 - VehicleTagNumber (Number, State), também conhecido como número da placa.
- Cada chave está sublinhada (Nota: isso é diferente do esquema relacional onde apenas uma “chave primária é sublinhada).

Entity Set

- Cada tipo de entidade terá uma coleção de entidades armazenadas no banco de dados
 - Chamado de conjunto de entidades ou, por vezes, de coleção de entidades

Mesmo nome (CAR) usado para se referir ao tipo de entidade e ao conjunto de entidades

Entity Set

- Cada tipo de entidade terá uma coleção de entidades armazenadas no banco de dados
 - Chamado de conjunto de entidades ou, por vezes, de coleção de entidades

Mesmo nome (CAR) usado para se referir ao tipo de entidade e ao conjunto de entidades

Value Sets (Domains) of Attributes

- Cada atributo simples está associado a um conjunto de valores
 - Por exemplo, Lastname tem um valor que é uma cadeia de caracteres de até 15 caracteres, digamos
 - A data tem um valor que consiste em MM-DD-AAAA, em que cada letra é um número inteiro
- O domínio de um atributo especifica o conjunto de valores associados a um atributo

Attributes and Value Sets

- Domínios de atributo são semelhantes aos tipos de dados na maioria das linguagens de programação - por exemplo, inteiro, caractere (n), real, bit
- Matematicamente, um atributo A para um tipo de entidade E cujo conjunto de valores é V é definido como uma função

$$A: E \rightarrow P(V)$$








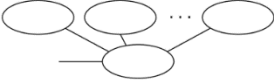

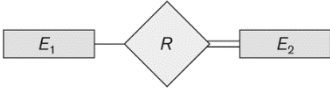


- Onde $P(V)$ indica um conjunto de energia (o que significa todos os subconjuntos possíveis) de V. A definição acima abrange atributos simples e de valor múltiplo.
- Referimo-nos ao valor do atributo A para entidade e como $A(e)$.

Displaying an Entity type

- Nos diagramas ER, um tipo de entidade é exibido em uma caixa retangular
- Atributos são exibidos no formato oval
- Cada atributo está conectado ao seu tipo de entidade
- Componentes de um atributo composto são conectados ao oval representando o atributo composto
- Cada atributo chave está sublinhado
- Atributos multivalorados exibidos em ovais duplos

NOTATION for ER diagrams

Figure 3.14
Summary of the
notation for ER
diagrams.

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Identifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute
	Total Participation of E_2 in R
	Cardinality Ratio 1: N for $E_1:E_2$ in R
	Structural Constraint (min, max) on Participation of E in R

Entity Type CAR with two keys and a corresponding Entity Set

(a)

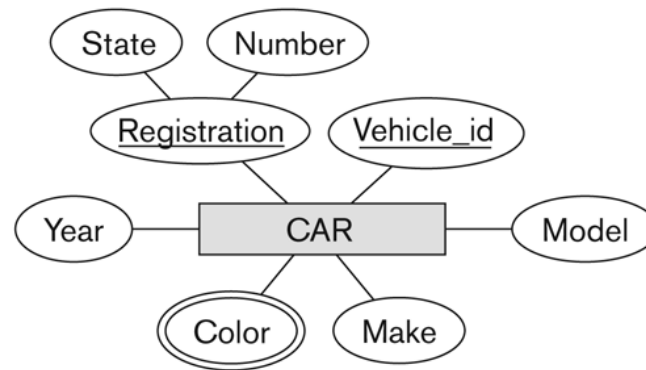


Figure 3.7

The CAR entity type with two key attributes, Registration and Vehicle_id. (a) ER diagram notation. (b) Entity set with three entities.

(b)

CAR
Registration (Number, State), Vehicle_id, Make, Model, Year, {Color}

CAR₁
((ABC 123, TEXAS), TK629, Ford Mustang, convertible, 2004 {red, black})

CAR₂
((ABC 123, NEW YORK), WP9872, Nissan Maxima, 4-door, 2005, {blue})

CAR₃
((VSY 720, TEXAS), TD729, Chrysler LeBaron, 4-door, 2002, {white, blue})

⋮

Initial Conceptual Design of Entity Types for the COMPANY Database Schema

- Com base nos requisitos, podemos identificar quatro tipos de entidades iniciais no banco de dados da EMPRESA:
 - DEPARTAMENTOO
 - PROJETO
 - EMPREGADO
 - DEPENDENTE
- Seu projeto conceitual inicial é mostrado no próximo slide
- Os atributos iniciais mostrados são derivados da descrição dos requisitos

Initial Design of Entity Types:

EMPLOYEE, DEPARTMENT, PROJETO, DEPENDENT

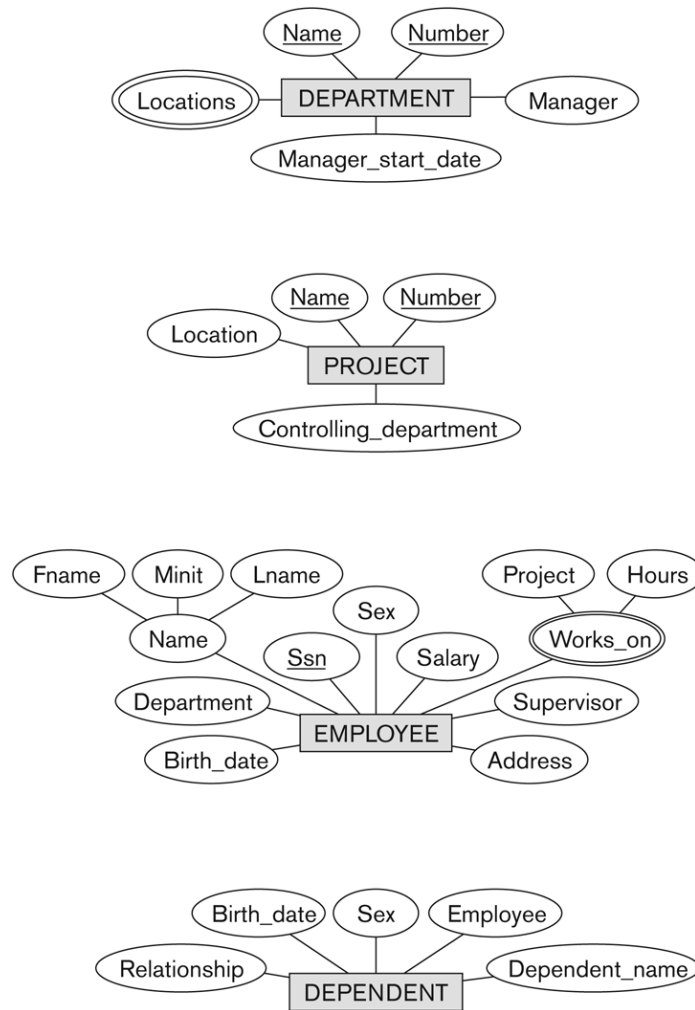


Figure 3.8

Preliminary design of entity types for the COMPANY database. Some of the shown attributes will be refined into relationships.

Refining the initial design by introducing relationships

- O design inicial normalmente não é completo
- Alguns aspectos nos requisitos serão representados como relacionamentos
- O modelo ER tem três conceitos principais:
 - Entidades (e seus tipos de entidades e conjuntos de entidades)
 - Atributos (simples, compostos, de vários valores)
 - Relacionamentos (e seus tipos de relacionamento e conjuntos de relacionamento)

Relationships and Relationship Types (1)

- Um relacionamento relaciona duas ou mais entidades distintas com um significado específico.
 - Por exemplo, o EMPREGADO John Smith trabalha no PROJETO ProductX, ou EMPREGADO Franklin Wong gerencia o DEPARTAMENTO de Pesquisa.
- Relacionamentos do mesmo tipo são agrupados ou digitados em um tipo de relacionamento.
 - Por exemplo, o tipo de relacionamento TRABALHA no qual os EMPLOYEES e PROJETOs participam, ou o tipo de relacionamento MANAGES no qual os EMPLOYEES e DEPARTMENTS participam.
- O grau de um tipo de relacionamento é o número de tipos de entidades participantes.
 - Tanto MANAGENS quanto TRABALHA são relacionamentos binários.

Relationship instances of the WORKS_FOR N:1 relationship between EMPLOYEE and DEPARTMENT

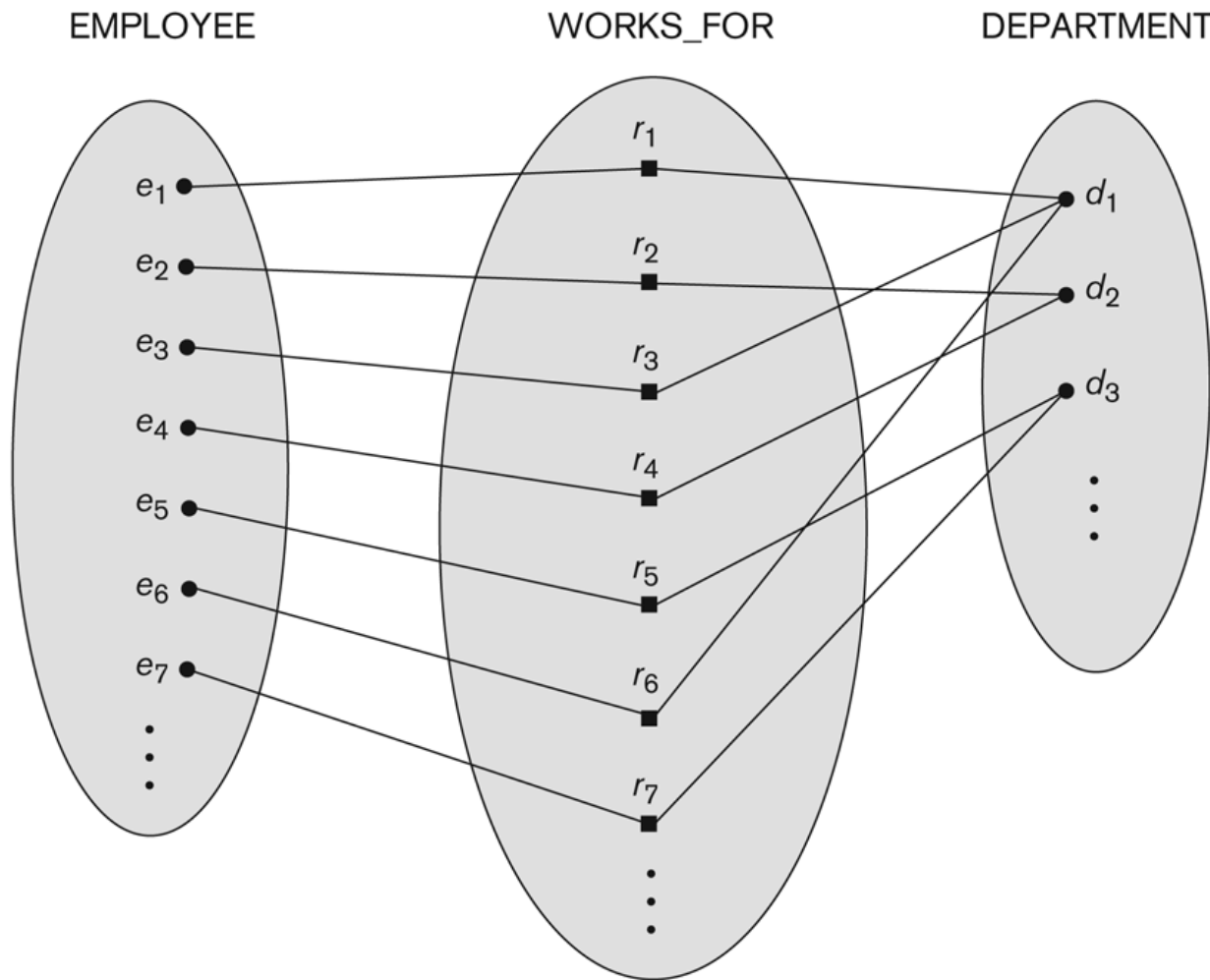


Figure 3.9

Some instances in the WORKS_FOR relationship set, which represents a relationship type WORKS_FOR between EMPLOYEE and DEPARTMENT.

Relationship instances of the M:N TRABALHA relationship between EMPLOYEE and PROJETO

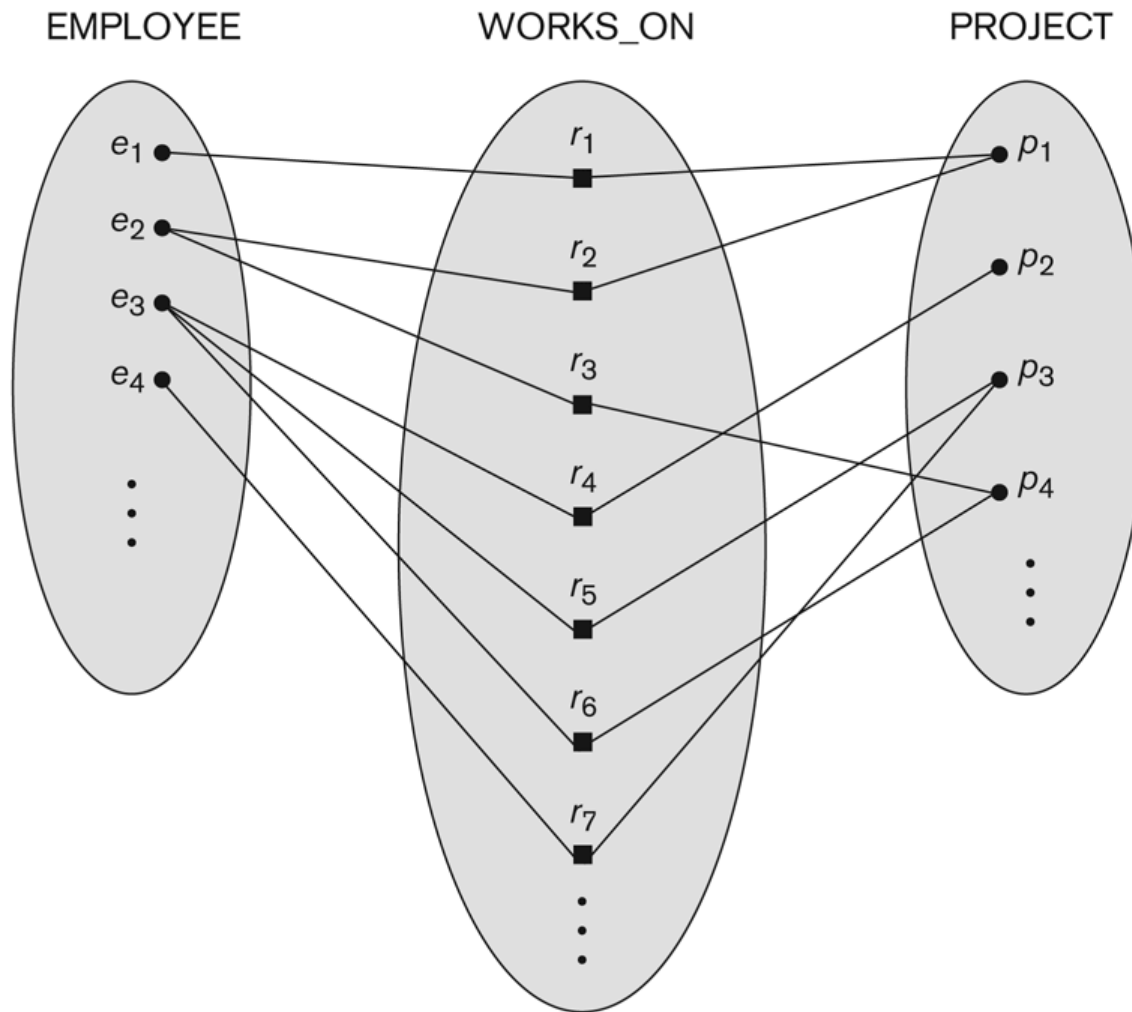


Figure 3.13
An M:N relationship,
WORKS_ON.

Relationship type vs. relationship set (1)

- Tipo de Relacionamento:
 - É a descrição do esquema de um relacionamento
 - Identifica o nome do relacionamento e os tipos de entidade participantes
 - Também identifica certas restrições de relacionamento
- Conjunto de Relacionamento:
 - O conjunto atual de instâncias de relacionamento representadas no banco de dados
 - O estado atual de um tipo de relacionamento

Relationship type vs. relationship set (2)

- Figuras anteriores exibiram os conjuntos de relacionamento
 - Cada instância no conjunto relaciona entidades participantes individuais - uma de cada tipo de entidade participante
- Nos diagramas ER, representamos o tipo de relacionamento da seguinte forma:
 - Caixa em forma de diamante é usada para exibir um tipo de relacionamento
 - Conectado aos tipos de entidades participantes por meio de linhas retas
 - Observe que o tipo de relacionamento não é mostrado com uma seta.
 - Normalmente, o nome deve ser legível da esquerda para a direita e de cima para baixo.

Refining the COMPANY database schema by introducing relationships

- Examinando os requisitos, seis tipos de relacionamento são identificados
- Todos são relacionamentos binários (grau 2)
- Listados abaixo com seus tipos de entidades participantes:
 - WORKS_FOR (between EMPLOYEE, DEPARTMENT)
 - MANAGES (also between EMPLOYEE, DEPARTMENT)
 - CONTROLS (between DEPARTMENT, PROJETO)
 - TRABALHA (between EMPLOYEE, PROJETO)
 - SUPERVISION (between EMPLOYEE (as subordinate), EMPLOYEE (as supervisor))
 - DEPENDENTS_OF (between EMPLOYEE, DEPENDENT)

ER DIAGRAM – Relationship Types are:

WORKS_FOR, MANAGES, TRABALHA, CONTROLS, SUPERVISION, DEPENDENTS_OF

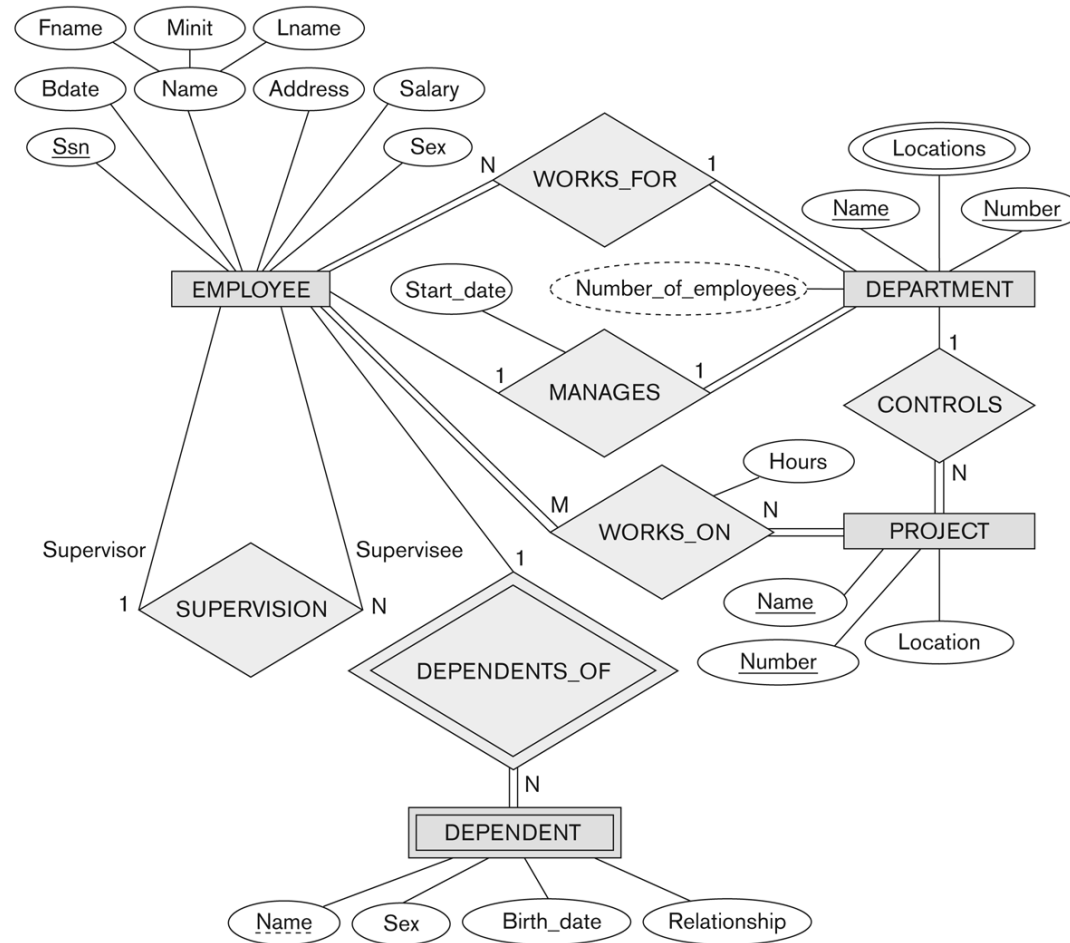


Figure 3.2

An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter.

Discussion on Relationship Types

- Em geral, mais de um tipo de relacionamento pode existir entre os mesmos tipos de entidades participantes
 - MANAGES e WORKS_FOR são tipos de relacionamento distintos entre EMPREGADO e DEPARTAMENTOO
 - Diferentes significados e diferentes instâncias de relacionamento.

Constraints on Relationships

- Restrições em tipos de relacionament
 - Cardinalidade (especifica a participação máxima)
 - Um-para-um (1: 1)
 - Um-para-muitos (1: N) ou Muitos-para-um (N: 1)
 - Muitos-para-muitos (M: N)
- Restrição de Dependência de Existência (especifica participação mínima) (também chamada de restrição de participação)
 - zero (participação opcional, não dependente da existência)
 - um ou mais (participação obrigatória, dependente da existência)

Many-to-one (N:1) Relationship

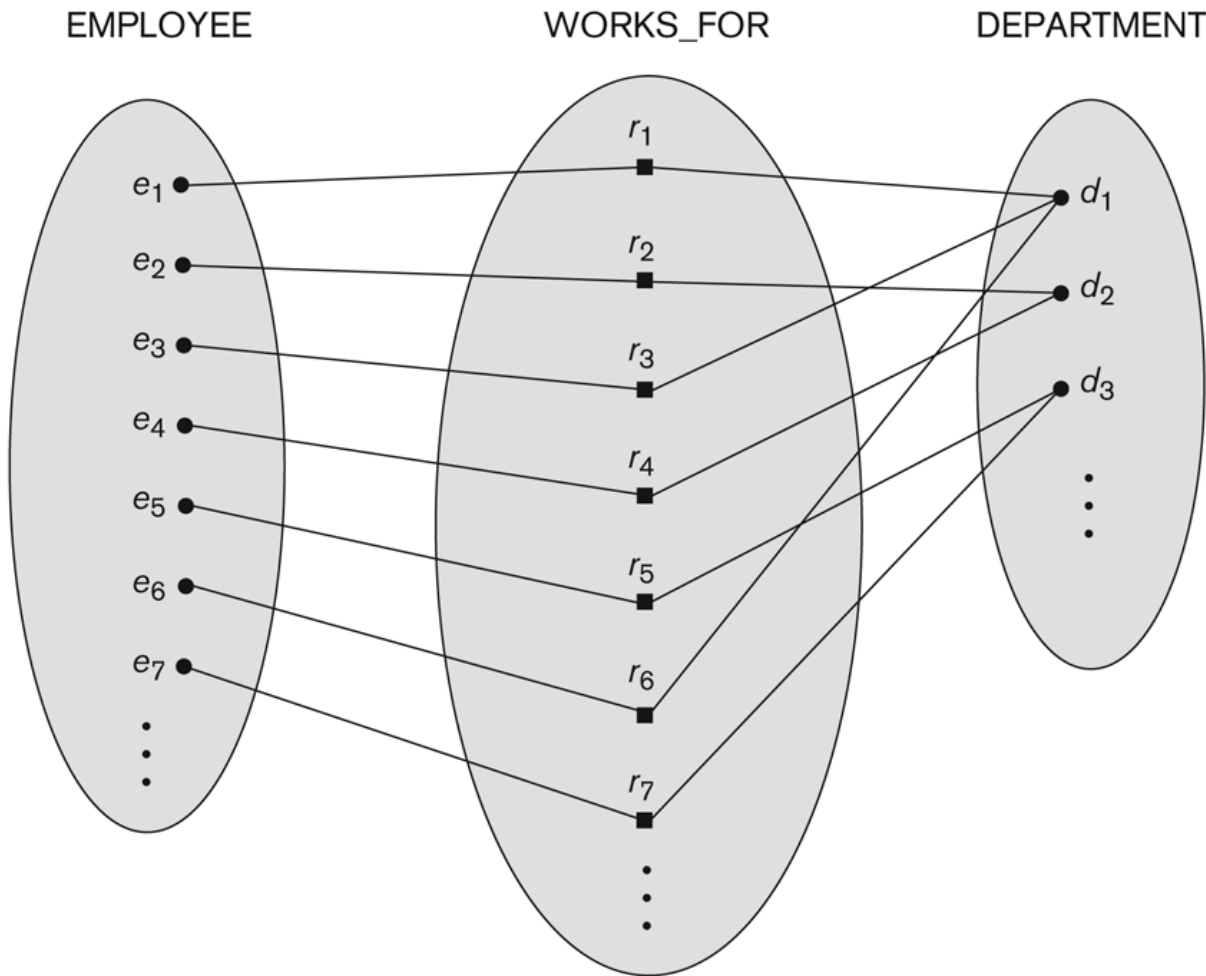


Figure 3.9

Some instances in the WORKS_FOR relationship set, which represents a relationship type WORKS_FOR between EMPLOYEE and DEPARTMENT.

Many-to-many (M:N) Relationship

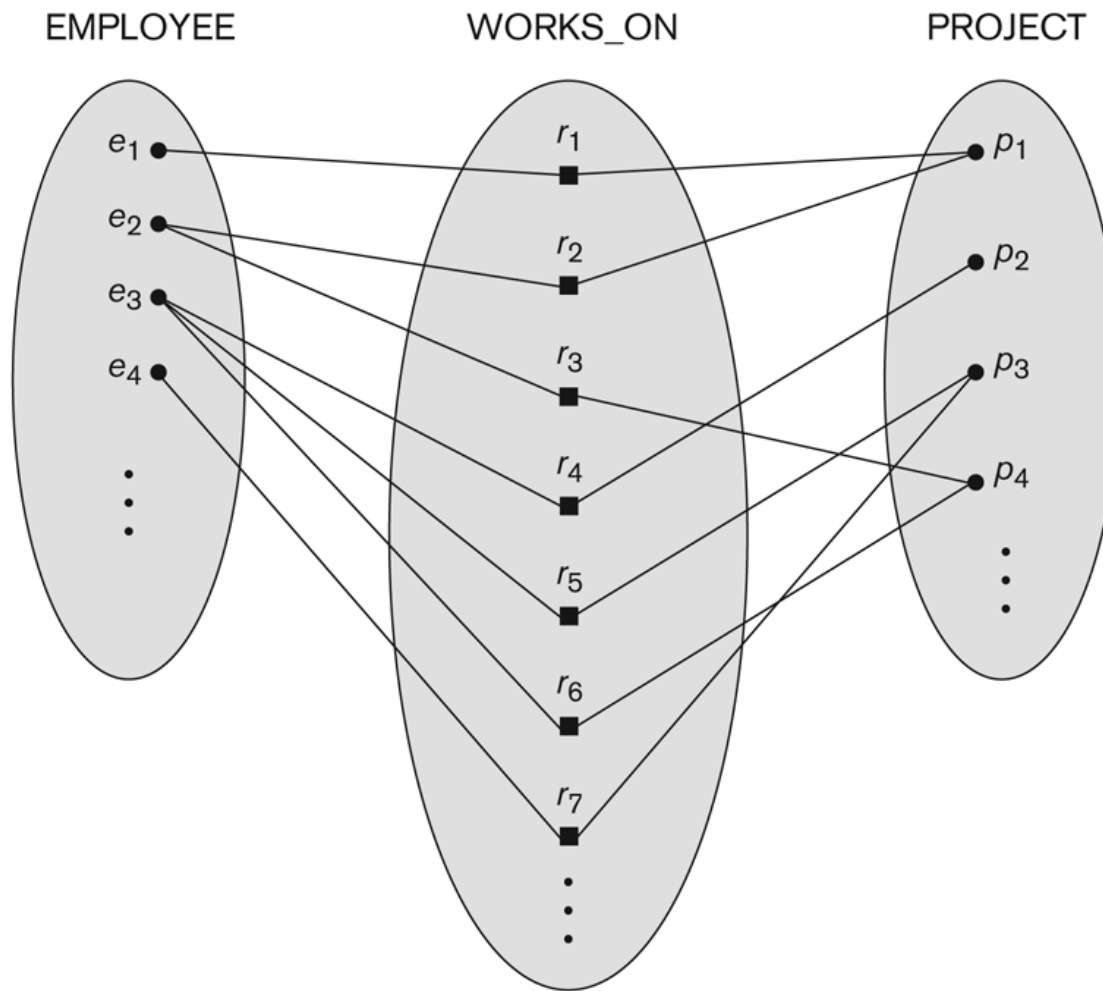


Figure 3.13
An M:N relationship,
WORKS_ON.

Recursive Relationship Type

- Um tipo de relacionamento entre o mesmo tipo de entidade participante em funções distintas
- Também chamado de tipo de relacionamento de auto-referência.
 - Exemplo: o relacionamento SUPERVISION
 - EMPLOYEE participa duas vezes em dois papéis distintos:
 - papel de supervisor (ou chefe)
 - função supervisionada (ou subordinada)
- Cada instância de relacionamento relaciona duas entidades EMPLOYEE distintas:
 - Um funcionário no papel de supervisor
 - Um empregado no papel de supervisão

Displaying a recursive relationship

- Em um tipo de relacionamento recursivo.
 - No diagrama de ER, precisa exibir nomes dos **papeis** para distinguir as participações.

A Recursive Relationship Supervision`

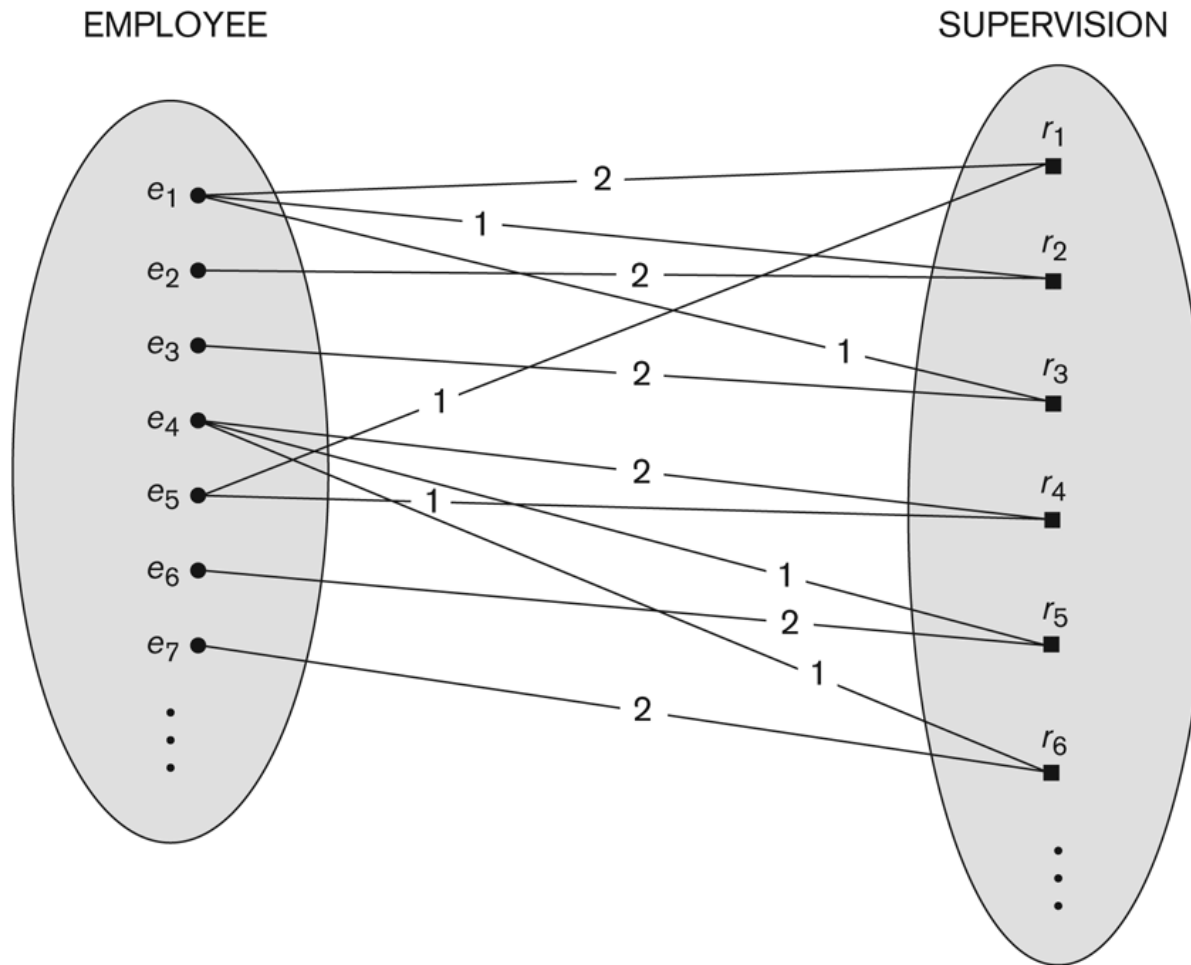


Figure 3.11

A recursive relationship SUPERVISION between EMPLOYEE in the *supervisor* role (1) and EMPLOYEE in the *subordinate* role (2).

Recursive Relationship Type is: SUPERVISION (participation role names are shown)

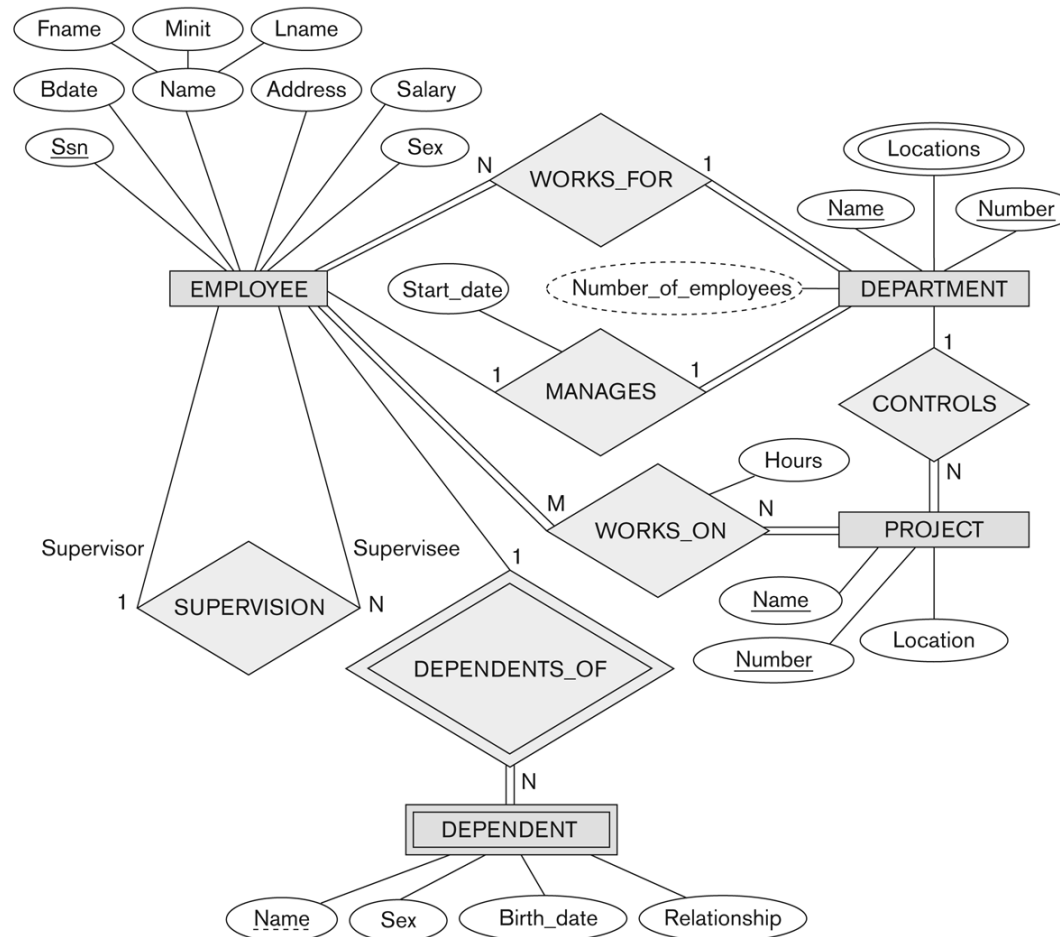


Figure 3.2

An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter.

Weak Entity Types

- Uma entidade que não possui um atributo chave e que é dependente de identificação em outro tipo de entidade.
- Uma **entidade fraca** deve participar de um tipo de relacionamento de identificação com um proprietário ou tipo de entidade de identificação
- As entidades são identificadas pela combinação de:
 - Uma chave parcial do tipo de entidade fraca
 - A entidade específica à qual eles estão relacionados no tipo de relacionamento de identificação
- Exemplo:
 - Uma entidade dependente é identificada pelo primeiro nome do dependente e pelo EMPREGADO específico com o qual o dependente está relacionado
 - Nome do DEPENDENTE é a chave parcial
 - DEPENDENTE é um tipo de entidade fraco
 - EMPREGADO é o seu tipo de entidade de identificação através do tipo de relacionamento de identificação DEPENDENT_OF

Attributes of Relationship types

- Um tipo de relacionamento pode ter atributos:
 - Por exemplo, HoursPerWeek de TRABALHA
 - Seu valor para cada instância de relacionamento descreve o número de horas por semana que um EMPREGADO trabalha em um PROJETO.
- Um valor de HoursPerWeek depende de uma combinação particular (EMPREGADO, PROJETO)
 - A maioria dos atributos de relacionamento é usada com relacionamentos M: N
 - Em relacionamentos 1: N, eles podem ser transferidos para o tipo de entidade no lado N do relacionamento

Example Attribute of a Relationship Type: Hours of TRABALHA

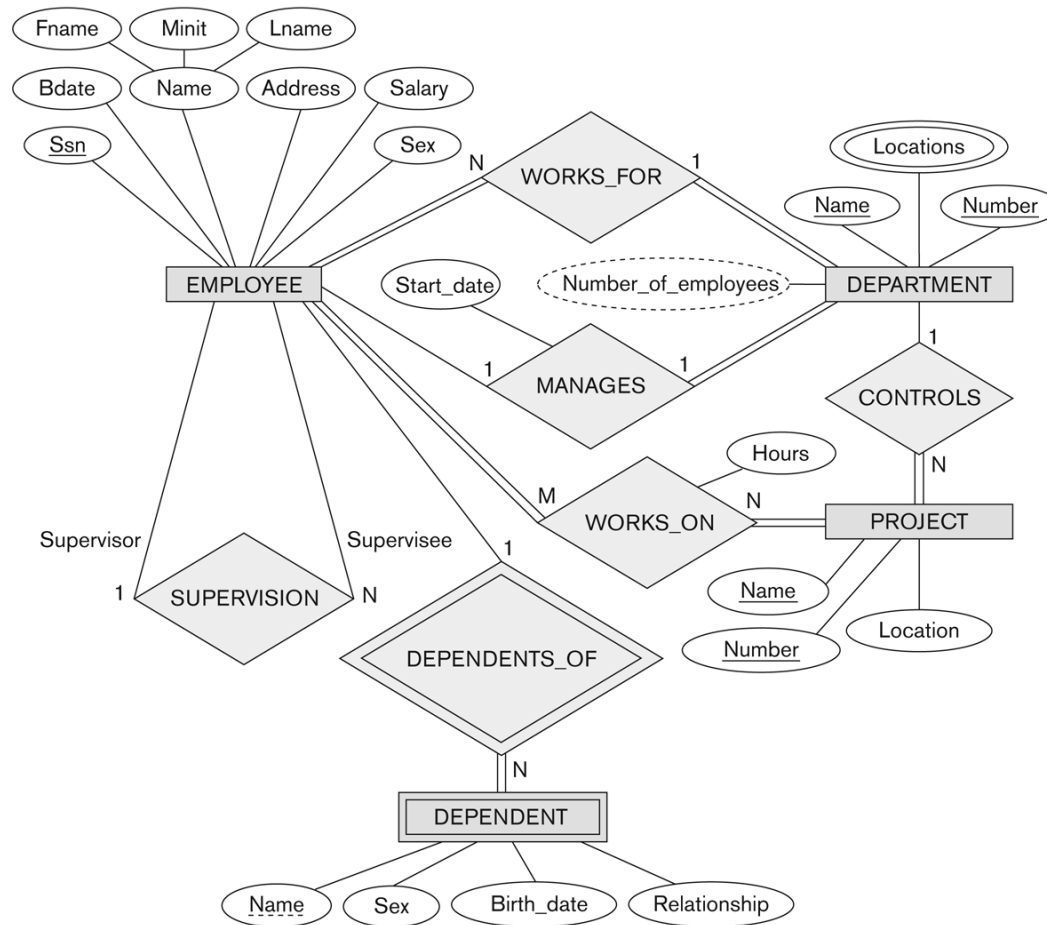


Figure 3.2

An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter.

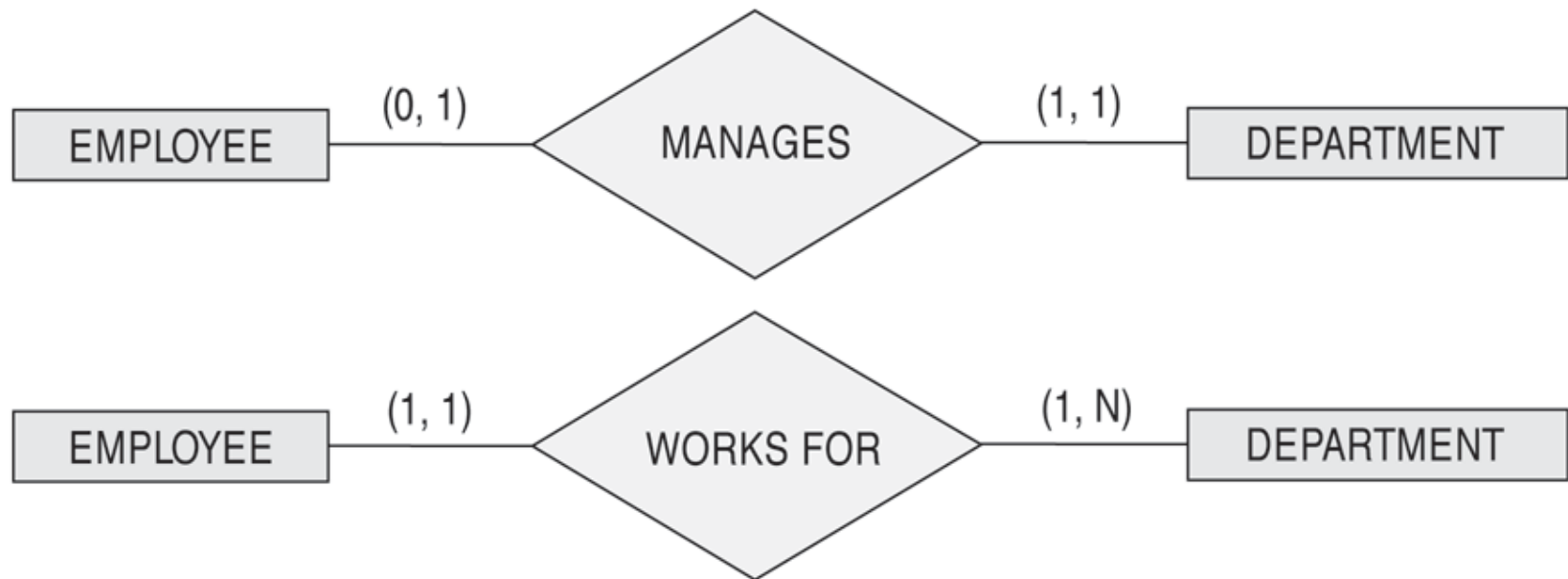
Notation for Constraints on Relationships

- Relação de cardinalidade (de uma relação binária):
1: 1, 1: N, N: 1 ou M: N
 - Mostrado colocando números apropriados nas bordas do relacionamento.
- Restrição de participação (em cada tipo de entidade participante): total (chamado dependência de existência) ou parcial.
 - Total mostrado por linha dupla, parcial por linha única.
 - NOTA: Estes são fáceis de especificar para Tipos de Relacionamento Binário.

Alternative (min, max) notation for relationship structural constraints:

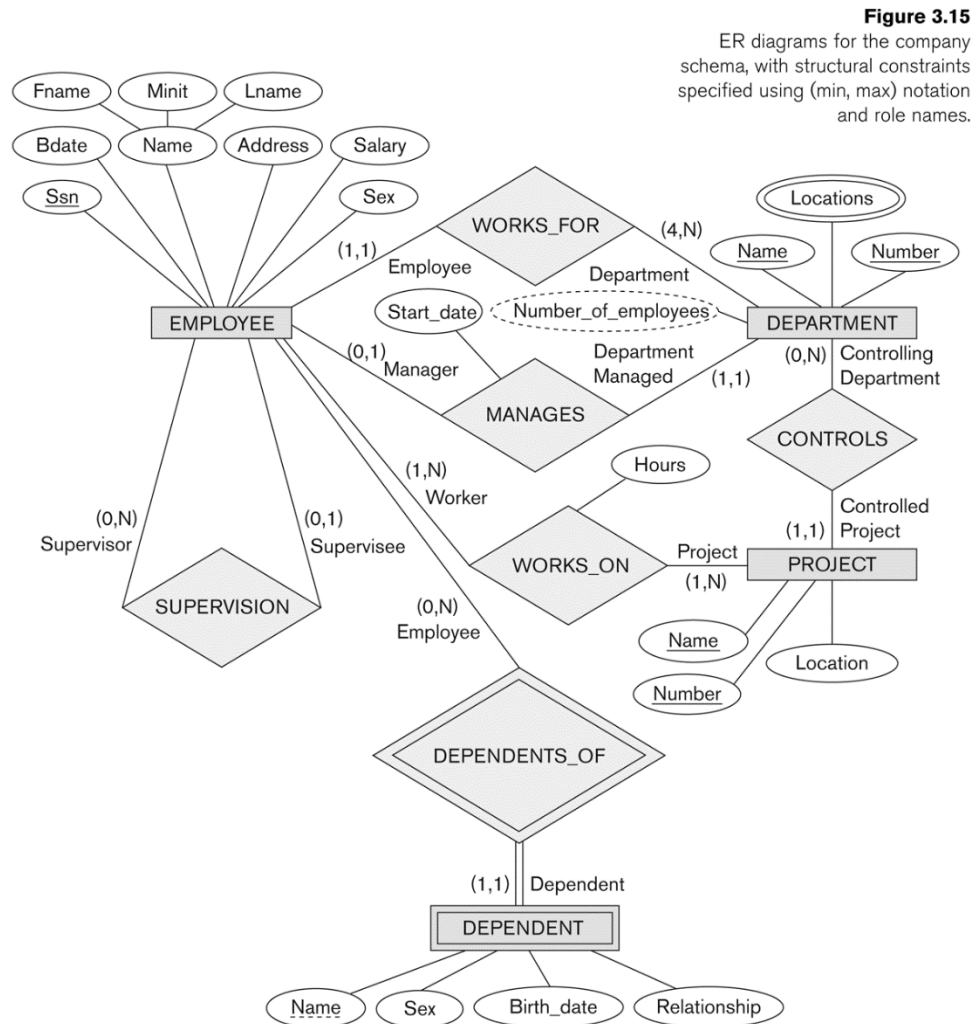
- Especificado em cada participação de um tipo de entidade E em um tipo de relacionamento R
- Especifica que cada entidade e em E participa em, pelo menos, min e, no máximo, instâncias de relacionamento máximas em R
- Padrão (sem restrição): min = 0, max = n (significando sem limite)
- Deve ter $\text{min} \leq \text{max}$, $\text{min} \geq 0$, $\text{max} \geq 1$
- Derivado do conhecimento de restrições de mini-mundo
 - Exemplos:
 - Um departamento tem exatamente um gerente e um EMPREGADO pode gerenciar no máximo um departamento.
 - Especifique (0,1) para participação do EMPREGADO em MANAGENS
 - Especifique (1,1) para participação do DEPARTAMENTO em GERENCIAMENTO
 - Um EMPREGADO pode trabalhar para exatamente um departamento, mas um departamento pode ter qualquer número de EMPREGADOS.
 - Especifique (1,1) para participação do EMPREGADO no TRABALHA_PARA
 - Especifique (0, n) para participação do DEPARTAMENTO no TRABALHA_PARA

The (min,max) notation for relationship constraints



Read the min,max numbers next to the entity type and looking **away from** the entity type

COMPANY ER Schema Diagram using (min, max) notation








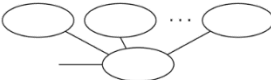

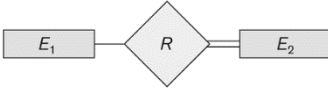
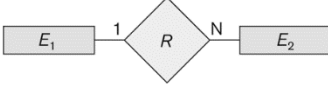
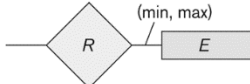


Alternative diagrammatic notation

- Diagramas ER é um exemplo popular para exibir esquemas de banco de dados
- Existem muitas outras notações na literatura e em várias ferramentas de design e modelagem de banco de dados
- Os diagramas de classe UML são representativos de outra maneira de exibir conceitos de ER que são usados em várias ferramentas de design comercial

Summary of notation for ER diagrams

Figure 3.14
Summary of the
notation for ER
diagrams.

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Identifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute
	Total Participation of E_2 in R
	Cardinality Ratio 1: N for $E_1:E_2$ in R
	Structural Constraint (min, max) on Participation of E in R

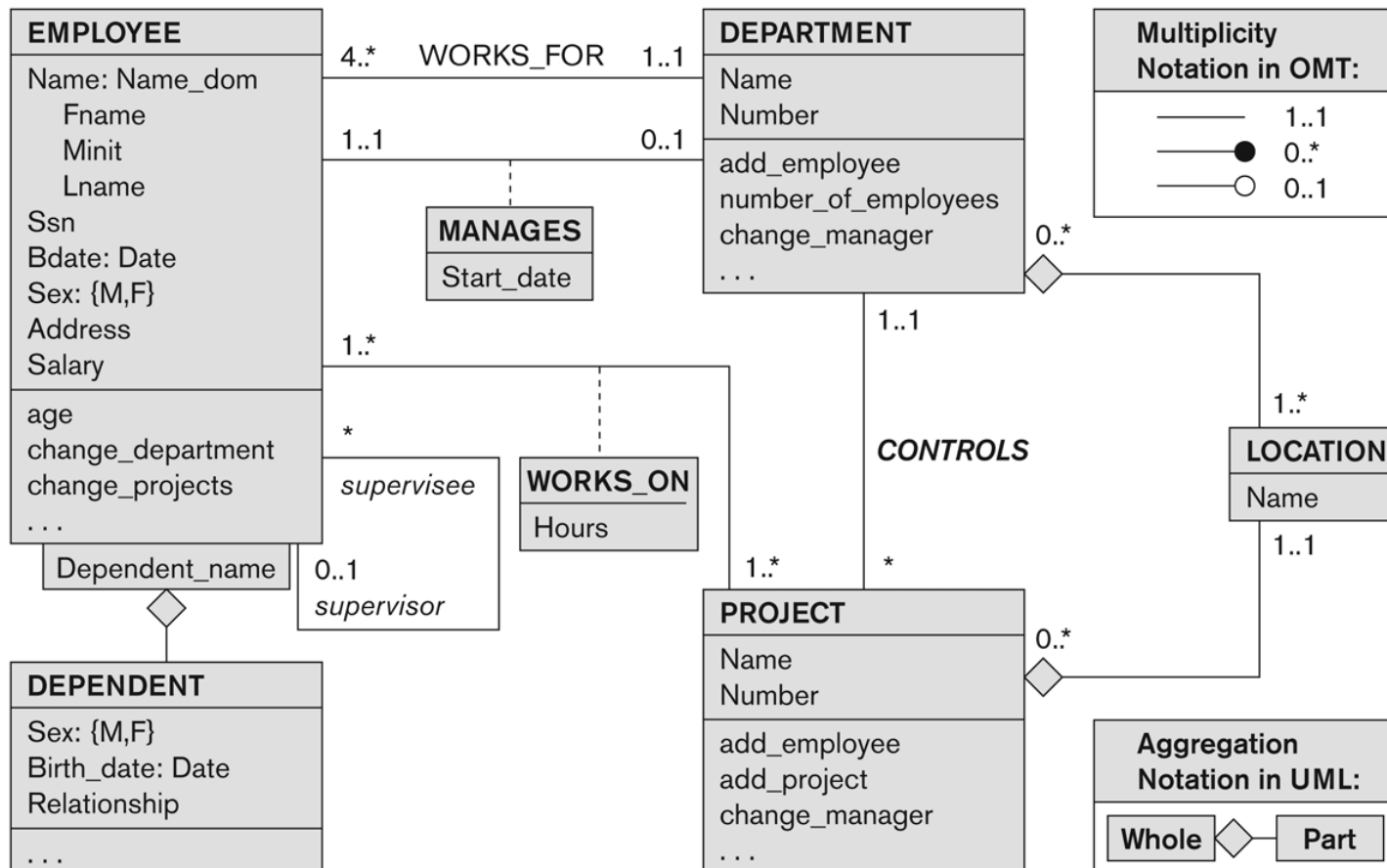
UML class diagrams

- Represente classes (semelhantes a tipos de entidade) como grandes caixas arredondadas com três seções:
 - A seção superior inclui o nome do tipo de entidade (classe)
 - Segunda seção inclui atributos
 - Terceira seção inclui operações de classe (as operações não estão no modelo ER básico)
- Relacionamentos (chamados associações) representados como linhas conectando as classes

UML class diagram for COMPANY database schema

Figure 3.16

The COMPANY conceptual schema in UML class diagram notation.



Other alternative diagrammatic notations

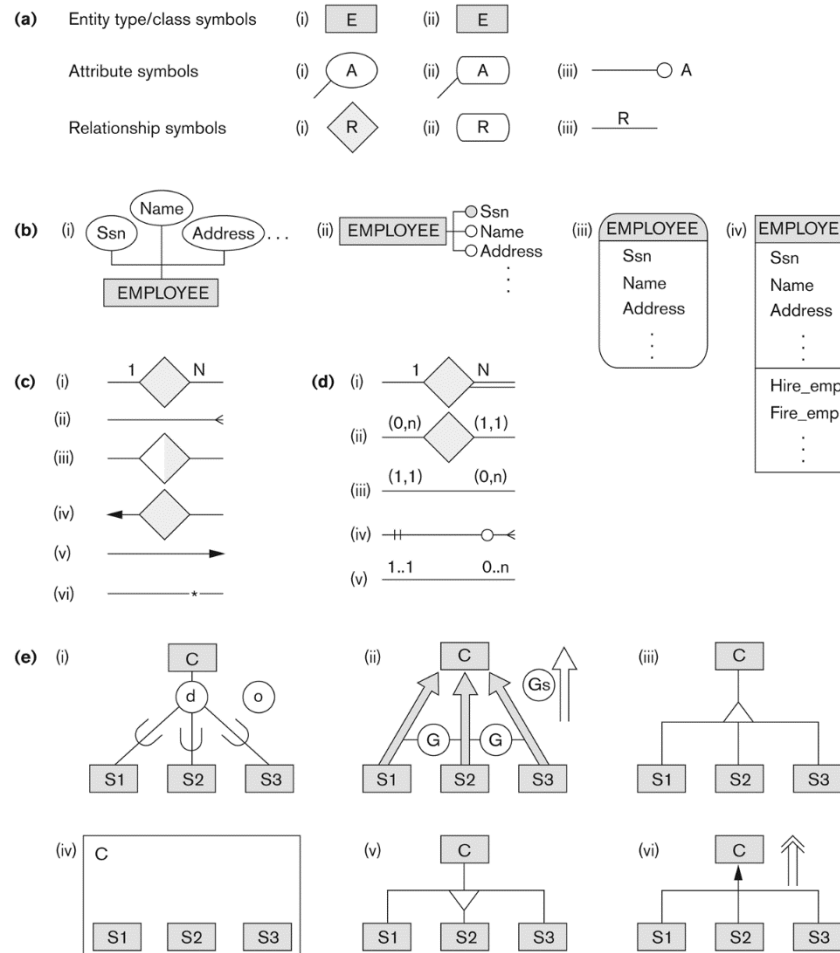


Figure A.1

Alternative notations. (a) Symbols for entity type/class, attribute, and relationship. (b) Displaying attributes. (c) Displaying cardinality ratios. (d) Various (min, max) notations. (e) Notations for displaying specialization/generalization.

Relationships of Higher Degree

- Tipos de relacionamento de grau 2 são chamados de binário
- Tipos de relacionamento de grau 3 são chamados ternários e de grau n são chamados n -ários
- Em geral, um relacionamento n -ário não é equivalente a n relacionamentos binários
- Restrições são mais difíceis de especificar para relacionamentos de nível superior ($n > 2$) do que para relacionamentos binários

Discussion of n-ary relationships ($n > 2$)

- Em geral, 3 relacionamentos binários podem representar informações diferentes de um único relacionamento ternário
- Se necessário, as relações binárias e n-árias podem ser incluídas no design do esquema, onde todas as relações transmitem diferentes significados
- Em alguns casos, um relacionamento ternário pode ser representado como uma entidade fraca .

Example of a ternary relationship

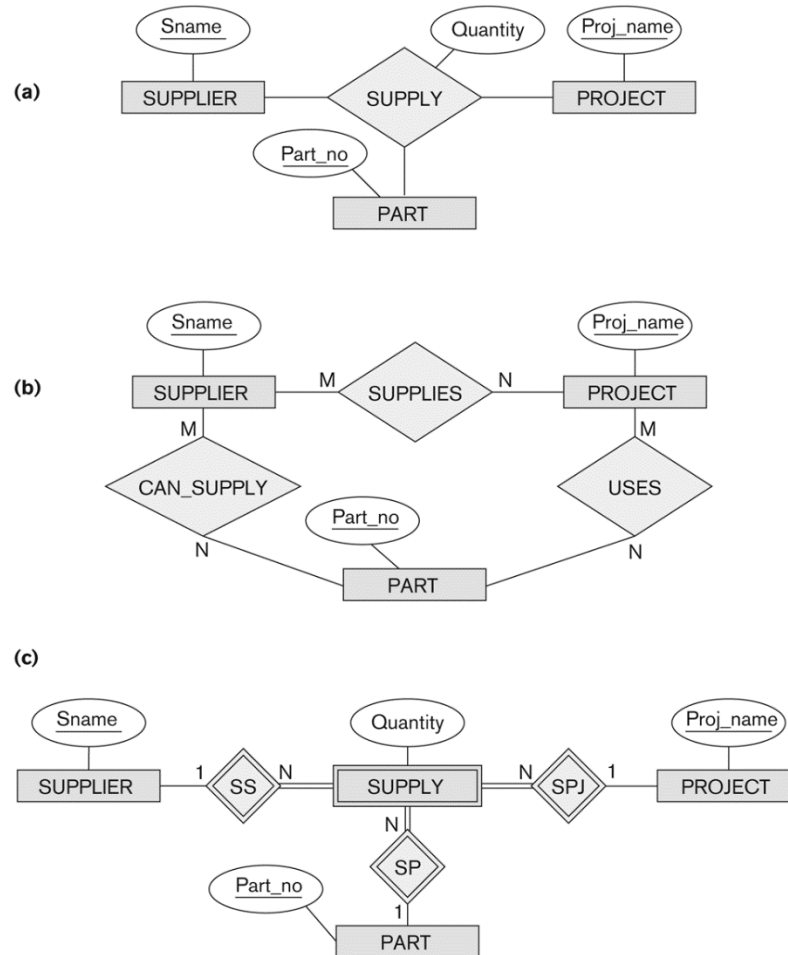


Figure 3.17

Ternary relationship types. (a) The SUPPLY relationship. (b) Three binary relationships not equivalent to SUPPLY. (c) SUPPLY represented as a weak entity type.

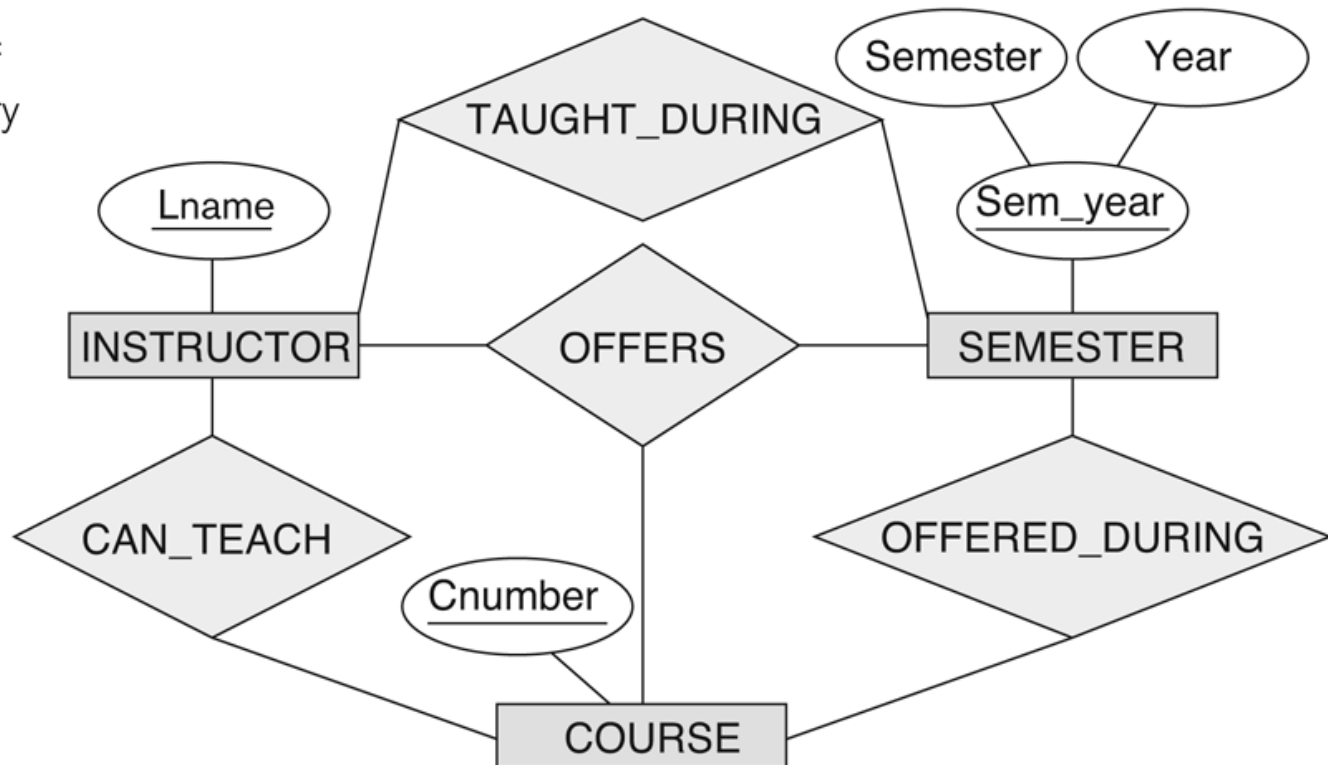
Discussion of n-ary relationships ($n > 2$)

- Se um relacionamento binário em particular puder ser derivado de um relacionamento de nível superior em todos os momentos, ele será redundante
- Por exemplo, o relacionamento binário TAUGHT_DURING (veja o próximo slide) pode ser derivado do relacionamento ternário OFFERS (baseado no significado dos relacionamentos)

Another example of a ternary relationship

Figure 3.18

Another example of ternary versus binary relationship types.



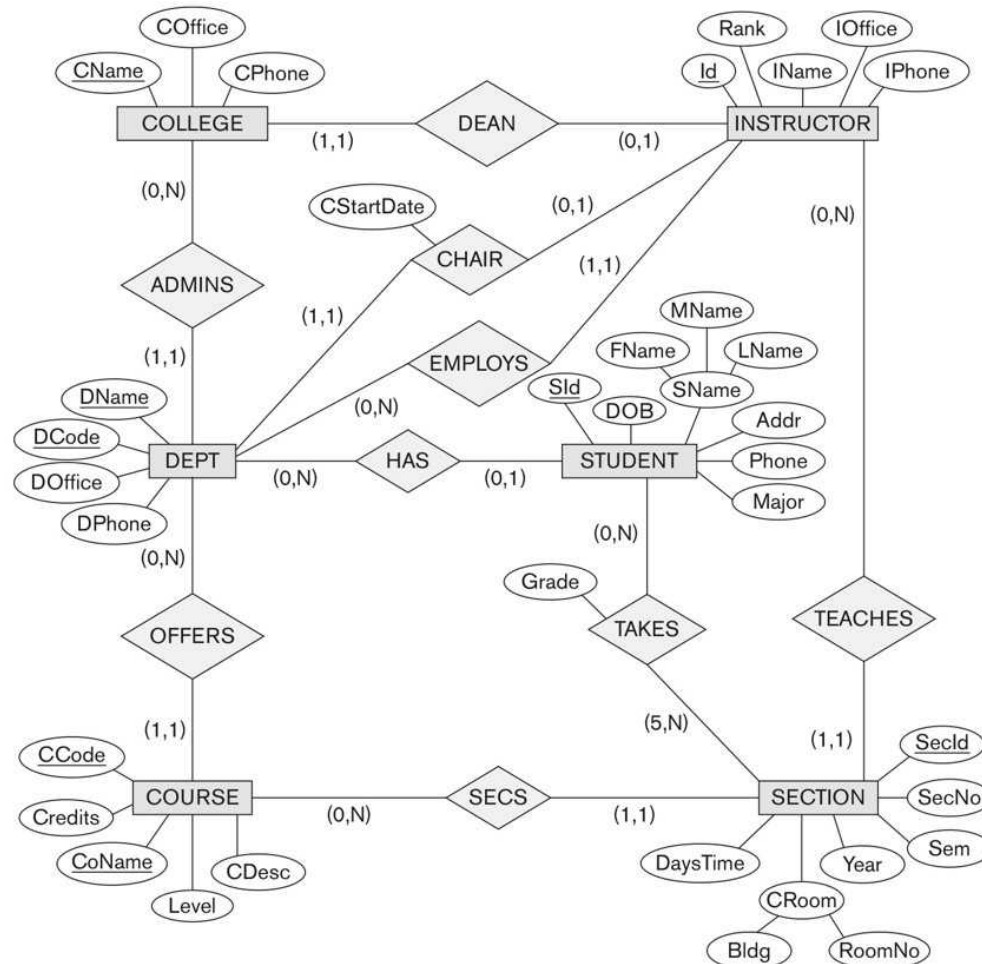
Displaying constraints on higher-degree relationships

- Em geral, ambos (min, max) e 1, M ou N são necessários para descrever totalmente as restrições
- No geral, a especificação de restrição é difícil e possivelmente ambígua quando consideramos relações de grau superior a dois.

Another Example: A UNIVERSITY Database

- To keep track of the enrollments in classes and student grades, another database is to be designed.
- It keeps track of the COLLEGES, DEPARTMENTS within each college, the COURSEs offered by departments, and SECTIONS of courses, INSTRUCTORS who teach the sections etc.
- These entity types and the relationships among these entity types are shown on the next slide in Figure 3.20.

UNIVERSITY database conceptual schema



Extended Entity-Relationship (EER) Model (in the next chapter)

- O modelo ER em sua forma original não suporta as abstrações de especialização e generalização
- O próximo capítulo ilustra como o modelo ER pode ser estendido com
 - Hierarquias de Especialização / Generalização
 - Notação para exibí-los em diagramas EER

Extended Entity-Relationship (EER) Model (in the next chapter)

- O modelo ER em sua forma original não suporta as abstrações de especialização e generalização
- O próximo capítulo ilustra como o modelo ER pode ser estendido com
 - Hierarquias de Especialização / Generalização
 - Notação para exibi-los em diagramas EER