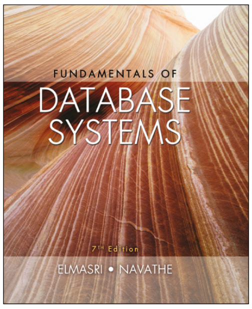
Fundamentals of Database Systems

Seventh Edition



Chapter 4

Enhanced Entity Relationship (EER) Modeling

Pearson

1

Enhanced/Extended ER (EER) model

- Created to design more accurate database schemas
  - Reflect the data properties and constraints more precisely
- Result of adding more semantic constructs to original entity relationship (ER) model
- Diagram using this model is called an EER diagram (EERD)

Advanced Data Modeling

- The Extended Entity Relationship (EER) model
- How to use flexible solutions for special data modeling cases

Pearson 3

3

Subclasses, Superclasses, and Inheritance

- EER model includes all modeling concepts of the ER model
- In addition, EER includes:
  - **Subclasses** and **superclasses**
  - **Specialization** and **generalization**
  - **Attribute** and **relationship inheritance**
- **Enhanced ER** or **EER diagrams**
  - Diagrammatic technique for displaying these concepts in an EER schema
  - Represented explicitly because of their significance to the database application

Pearson

5

Subclasses and Superclasses

- An entity type may have additional meaningful subgroupings of its entities
  - Example: EMPLOYEE may be further grouped into:
    - SECRETARY, ENGINEER, TECHNICIAN, …
      - Based on the EMPLOYEE's Job
    - MANAGER
      - EMPLOYEEs who are managers (the role they play)
    - SALARIED_EMPLOYEE, HOURLY_EMPLOYEE
      - Based on the EMPLOYEE's method of pay
- E E R diagrams extend E R diagrams to represent these additional subgroupings, called **subclasses** or **subtypes**

Pearson

6

Subclasses and Superclasses

- Terms for relationship between a superclass and any one of its subclasses
  - **Superclass/subclass**
  - **Supertype/subtype**
  - **Class/subclass** relationship
  - "**IS-A**" or **"IS-AN"** relationship
- **Type inheritance**
  - Subclass entity inherits all attributes and relationships of superclass
- **Entity superclass**
  - Generic entity type related to one or more entity subtypes
  - Contains common characteristics
- **Entity subclass**
  - Contains unique characteristics of each entity subtype

Pearson 7

7

8

FIGURE 6.1 Nulls created by unique attributes

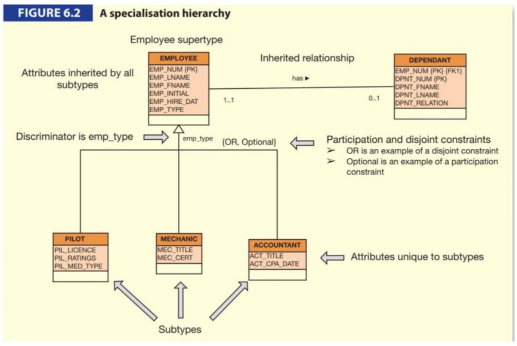
EMP_ NUM	EMP_ LNAME	EMP_ FNAME	EMP_ INITIAL	EMP_ LICENCE	EMP_RATINGS	EMP_ MED_TYPE	EMP_HIRE_ DATE
100	Nkosi	Cela	T				15-Mar-98
101	Lewis	Marcos		ATP	SEL/MEL/Instr/CFII	1	25-Apr-99
102	Vandam	Jean					20-Dec-03
103	Jones	Victoria	R				28-Aug-13
104	Lange	Edith		ATP	SEL/MEL/Instr	1	20-Oct-07
105	Williams	Gabriel	U	COM	SEL/MEL/Instr/CFI	2	08-Nov-07
106	Naidu	Theeban		COM	SEL/MEL/Instr	2	05-Jan-14
107	Diante	Venite	L				02-Jul-07
108	Shenge	Mhambi					18-Nov-05
109	Travis	Brett	T	COM	SEL/MEL/SES/Instr/CFII	1	14-Apr-11
110	Genkazi	Stan					01-Dec-13

8

Subclasses and Superclasses

Figure 4.1 EER diagram notation to represent subclasses and specialization.

FIGURE 6.2 A specialisation hierarchy



- A member entity of the subclass represents the *same real-world entity* as some member of the superclass; MECHANIC entity is also an EMPLOYEE.

Pearson

9

Subclasses and Superclasses

FIGURE 6.3 The EMPLOYEE – PILOT supertype – subtype relationship

Table name: EMPLOYEE

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIRE_DATE	EMP_TYPE
100	Nkosi	Cela	T	15-Mar-98	
101	Lewis	Marcos		25-Apr-99	P
102	Vandam	Jean		20-Dec-03	A
103	Jones	Victoria	R	28-Aug-13	
104	Lange	Edith		20-Oct-07	P
105	Williams	Gabriel	U	08-Nov-07	P
106	Naidu	Theeban		05-Jan-14	P
107	Diante	Venite	L	02-Jul-07	M
108	Shengi	Mhambi		18-Nov-05	M
109	Travis	Brett	T	14-Apr-11	P
110	Genkazi	Stan		01-Dec-13	A

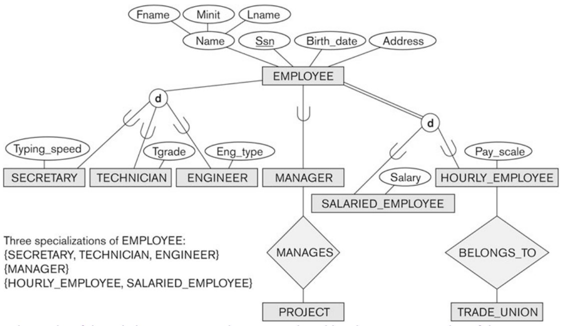
Table name: PILOT

EMP_NUM	PIL_LICENCE	PIL_RATINGS	PIL_MED_TYPE
101	ATP	SEL/MEL/Instr/CFII	1
104	ATP	SEL/MEL/Instr	1
105	COM	SEL/MEL/Instr/CFI	2
106	COM	SEL/MEL/Instr	2
109	COM	SEL/MEL/SES/Instr/CFII	1

10

Subclasses and Superclasses

Figure 4.1 EER diagram notation to represent subclasses and specialization.



- A member entity of the subclass represents the *same real-world entity* as some member of the superclass; SECRETARY entity is also an EMPLOYEE.
- A salaried employee who is also an engineer belongs to the two subclasses ENGINEER and SALARIED employee. A member of the superclass can be optionally included as a member of any number of its subclasses

11

Subclasses and Superclasses

- Each of these subgroupings is a subset of EMPLOYEE entities
- Each is called a subclass of EMPLOYEE
- EMPLOYEE is the superclass for each of these subclasses
- These are called superclass/subclass relationships:
  - EMPLOYEE/SECRETARY
  - EMPLOYEE/TECHNICIAN
  - EMPLOYEE/MANAGER
  - …

Pearson

12

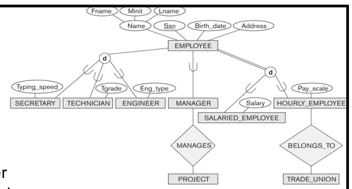
Subclasses and Superclasses

- These are also called IS-A relationships
  - SECRETARY IS-AN EMPLOYEE, TECHNICIAN IS-AN EMPLOYEE, ….
- Note: An entity that is member of a subclass represents the same real-world entity as some member of the superclass:
  - The subclass member is the same entity in a **distinct specific role**
  - An entity cannot exist in the database merely by being a member of a subclass; it must also be a member of the superclass
  - A member of the superclass can be optionally included as a member of any number of its subclasses

Pearson

13

Subclasses and Superclasses



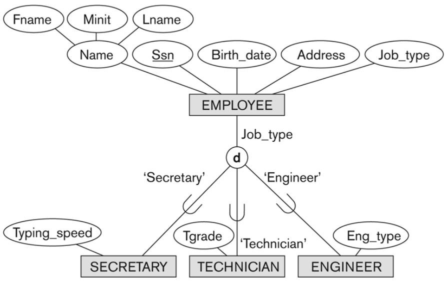
- Examples:
  - A salaried employee who is also an engineer belongs to the two subclasses:
    - ENGINEER, and
    - SALARIED_EMPLOYEE
  - A salaried employee who is also an engineering manager belongs to the three subclasses:
    - MANAGER,
    - ENGINEER, and
    - SALARIED_EMPLOYEE
- It is not necessary that every entity in a superclass be a member of some subclass

Pearson

14

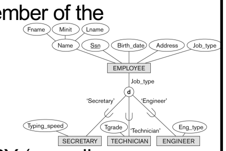
Representing Specialization in EER Diagrams

Figure 4.4 EER diagram notation for an attribute-defined specialization on Job_type.



15

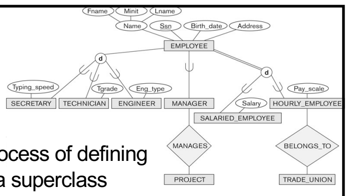
Attribute Inheritance in Superclass / Subclass Relationships



- An entity that is member of a subclass **inherits**
  - All attributes of the entity as a member of the superclass
  - All relationships of the entity as a member of the superclass
- Example:
  - In the previous slide, SECRETARY (as well as TECHNICIAN and ENGINEER) inherit the attributes Name, SSN, …, from EMPLOYEE
  - Every SECRETARY entity will have values for the inherited attributes

16

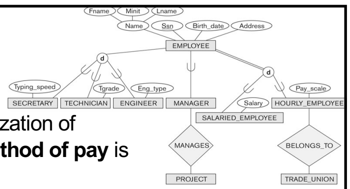
Specialization



- Specialization is the process of defining a set of subclasses of a superclass
- The set of subclasses is based upon some distinguishing characteristics of the entities in the superclass
  - Example: {SECRETARY, ENGINEER, TECHNICIAN} is a specialization of EMPLOYEE based upon **job type**.
  - Example: MANAGER **is a specialization of EMPLOYEE** based on **the role the employee plays**
    - May have several specializations of the same superclass

17

Specialization

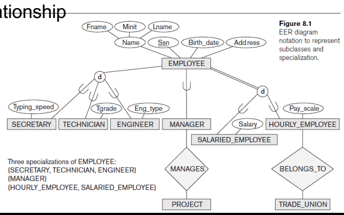


- Example: Another specialization of EMPLOYEE based on **method of pay** is {SALARIED_EMPLOYEE, HOURLY_EMPLOYEE}.
  - Superclass/subclass relationships and specialization can be diagrammatically represented in EER diagrams
  - Attributes of a subclass are called **specific** or **local** attributes.
    - For example, the attribute TypingSpeed of SECRETARY
  - The subclass can also participate in specific relationship types.
    - For example, a relationship BELONGS_TO of HOURLY_EMPLOYEE

18

Specialization and Generalization

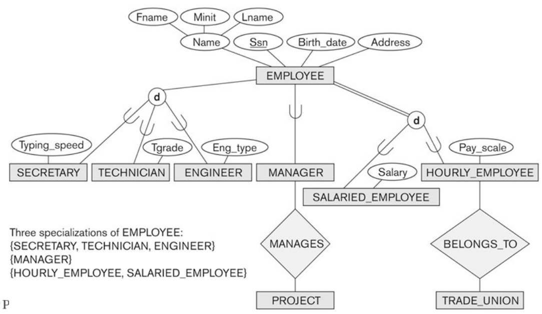
- Subclass can define:
  - **Specific attributes**
    - i.e. TypingSpeed of SECRETARY
  - **Specific relationship types**
    - i.e. HOURLY EMPLOYEE subclass participating in the BELONGS_TO relationship



19

Specialization

Figure 4.1 EER diagram notation to represent subclasses and specialization.



20

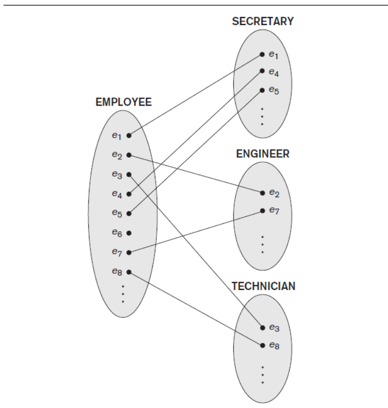


Figure 8.2
Instances of a specialization.

Pearson

21

Specialization and Generalization (cont'd.)

- Two main reasons for including class/subclass relationships and specialization in the data model;
  - Certain attributes may apply to some but not all entities of the superclass
  - Some relationship types may be participated in only by entities that are members of the subclass

Pearson

22

Specialization Hierarchy

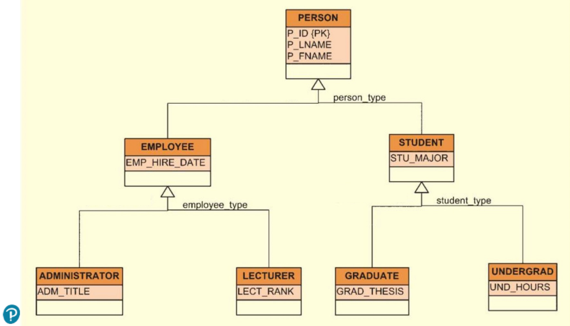
- Depicts arrangement of higher-level entity supertypes and lower-level entity subtypes
- Relationships described in terms of "IS-A" relationships
- Subtype exists only within context of supertype
- Every subtype has only one supertype to which it is directly related
- Can have many levels of supertype/subtype relationships

Pearson 23

23

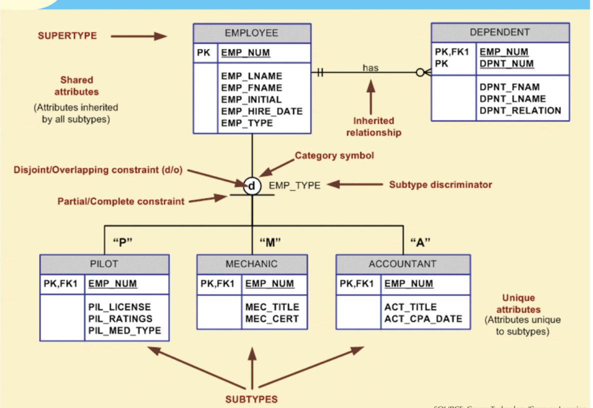
Specialization Hierarchy

FIGURE 6.4 Specialisation hierarchy



24

FIGURE 5.2 A specialization hierarchy 25



SOURCE: Course Technology/Cengage Learning

25

Inheritance

- Enables entity subtype to inherit attributes and relationships of supertype
- All entity subtypes inherit their primary key attribute from their supertype
- At implementation level, supertype and its subtype(s) maintain a 1:1 relationship
- Entity subtypes inherit all relationships in which supertype entity participates
- Lower-level subtypes inherit all attributes and relationships from all upper-level supertypes

Pearson 26

26

FIGURE 6.3 **The EMPLOYEE – PILOT supertype – subtype relationship**

Table name: EMPLOYEE

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIRE_DATE	EMP_TYPE
100	Nkosi	Cela	T	15-Mar-98	
101	Lewis	Marcos		25-Apr-99	P
102	Vandam	Jean		20-Dec-03	A
103	Jones	Victoria	R	28-Aug-13	
104	Lange	Edith		20-Oct-07	P
105	Williams	Gabriel	U	08-Nov-07	P
106	Naidu	Theeban		05-Jan-14	P
107	Diante	Venite	L	02-Jul-07	M
108	Shengi	Mhambi		18-Nov-05	M
109	Travis	Brett	T	14-Apr-11	P
110	Genkazi	Stan		01-Dec-13	A

Table name: PILOT

EMP_NUM	PIL_LICENCE	PIL_RATINGS	PIL_MED_TYPE
101	ATP	SEL/MEL/Instr/CFII	1
104	ATP	SEL/MEL/Instr	1
105	COM	SEL/MEL/Instr/CFI	2
106	COM	SEL/MEL/Instr	2
109	COM	SEL/MEL/SES/Instr/CFII	1

27

Specialization and Generalization

- Specialization
  - Identifies more specific entity subtypes from higher-level entity supertype
  - Top-down process
  - Based on grouping unique characteristics and relationships of the subtypes
- Generalization
  - Identifies more generic entity supertype from lower-level entity subtypes
  - Bottom-up process
  - Based on grouping common characteristics and relationships of the subtypes
  - Process of defining a generalized entity type from the given entity types

Pearson 28

28

Generalization

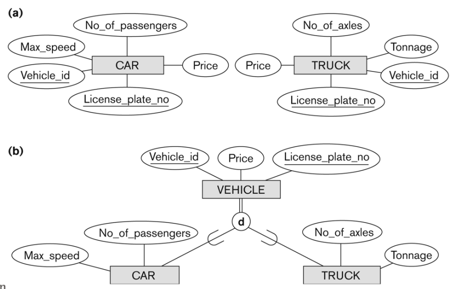
- Generalization is the reverse of the specialization process
- Several classes with common features are generalized into a superclass;
  - original classes become its subclasses
- Example: CAR, TRUCK generalized into VEHICLE;
  - both CAR, TRUCK become subclasses of the superclass VEHICLE.
  - We can view {CAR, TRUCK} as a specialization of VEHICLE
  - Alternatively, we can view VEHICLE as a generalization of CAR and TRUCK
- **Generalize** into a single **superclass**
  - Original entity types are special subclasses

Pearson

30

Generalization

Figure 4.3 Generalization. (a) Two entity types, CAR and TRUCK. (b) Generalizing CAR and TRUCK into the superclass VEHICLE.



Pearson

31

Generalization and Specialization

- Data Modeling with Specialization and Generalization
  - A superclass or subclass represents a collection (or set or grouping) of entities
  - It also represents a particular **type of entity**
  - Shown in rectangles in EER diagrams (as are entity types)
  - We can call all entity types (and their corresponding collections) **classes**, whether they are entity types, superclasses, or subclasses

Pearson

33

Specialization/Generalization Hierarchies,

- In **specialization**, start with an entity type and then define subclasses of the entity type by successive specialization
  - called a **top down** conceptual refinement process
- In **generalization**, start with many entity types and generalize those that have common properties
  - Called a **bottom up** conceptual synthesis process
- In practice, a **combination of both processes** is usually employed

Pearson

48

Design Cases: Learning Flexible Database Design

- Data modeling and design requires skills acquired through experience
- Experience acquired through practice
- Four special design cases that highlight:
  - Importance of flexible design
  - Proper identification of primary keys
  - Placement of foreign keys

Pearson 59

59

Design Case 1: Implementing 1:1 Relationships

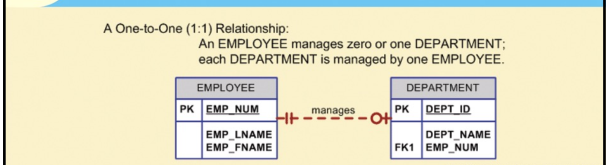
- Foreign keys work with primary keys to properly implement relationships in relational model
- Put primary key of the "one" side on the "many" side as foreign key
  - Primary key: parent entity
  - Foreign key: dependent entity
- In 1:1 relationship, there are two options:
  - Place a foreign key in both entities (not recommended)
  - Place a foreign key in one of the entities
    - Primary key of one of the two entities appears as foreign key of other
    - As FK in entity that causes least number of nulls

Pearson 60

60

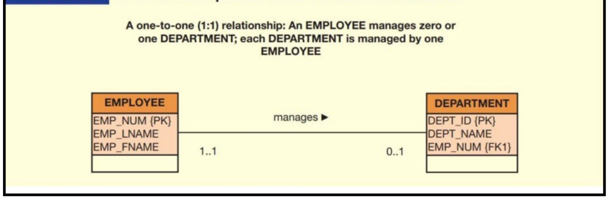
63

FIGURE 5.7 The 1:1 relationship between DEPARTMENT and EMPLOYEE



SOURCE: Course Technology/Cengage Learning

FIGURE 6.9 A 1:1 relationship between DEPARTMENT and EMPLOYEE



63

Design Case 2: Maintaining History of Time-Variant Data

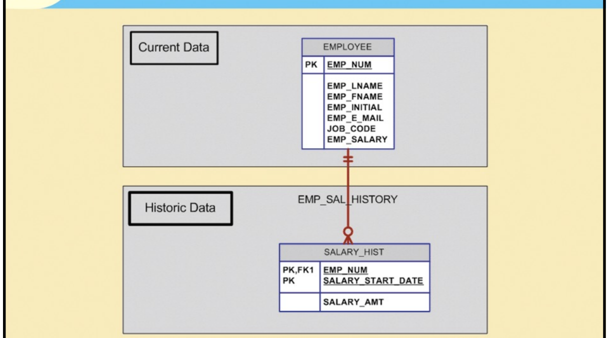
- Normally, existing attribute values are replaced with new value without regard to previous value
- Time-variant data:
  - Values change over time
  - Must keep a history of data changes
- Keeping history of time-variant data equivalent to having a multivalued attribute in your entity
- Must create new entity in 1:M relationships with original entity
- New entity contains new value, date of change

Pearson 64

64

65

FIGURE 5.8 Maintaining salary history

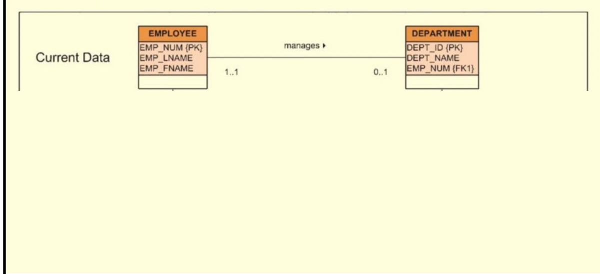


SOURCE: Course Technology/Cengage Learning

65

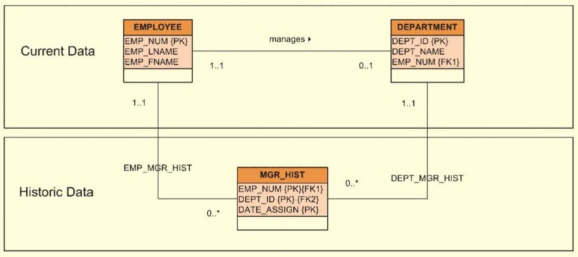
66

FIGURE 6.10 Maintaining manager history



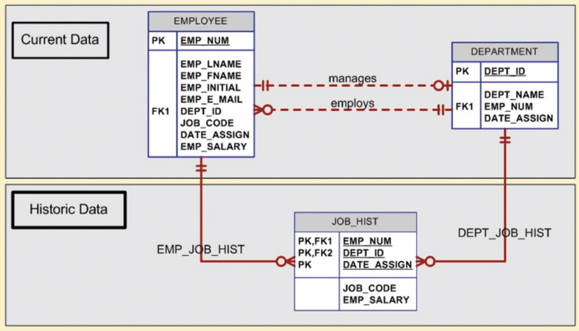
66

FIGURE 6.10 Maintaining manager history



67

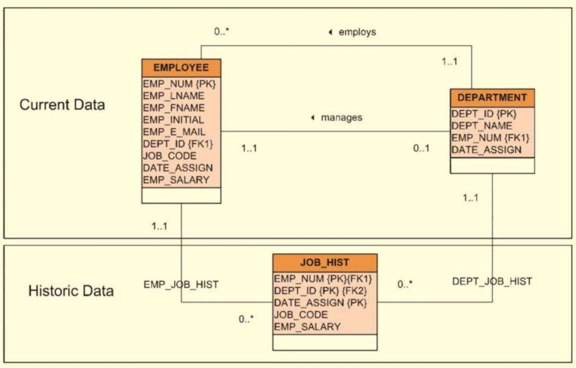
FIGURE 5.10 Maintaining job history



SOURCE: Course Technology/Cengage Learning

68

FIGURE 6.11 Maintaining job history



69

Design Case 3: Fan Traps

- Design trap occurs when relationship is improperly or incompletely identified
  - Represented in a way not consistent with the real world
- Most common design trap is known as fan trap
- Fan trap occurs when one entity is in two 1:M relationships to other entities
  - Produces an association among other entities not expressed in the model

Pearson

70

FIGURE 5.11 Incorrect ERD with fan trap problem

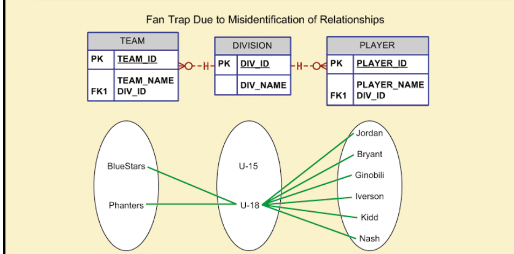
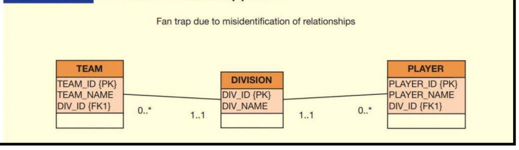


FIGURE 6.12 Incorrect ERD with fan trap problem



71

FIGURE 5.12 Corrected ERD after removal of the fan trap

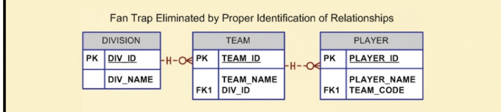
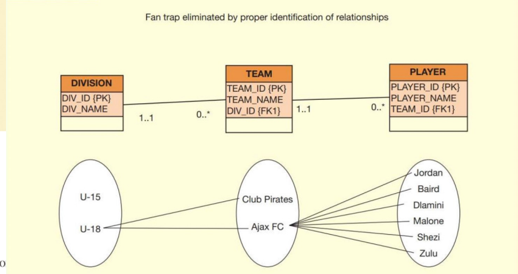


FIGURE 6.13 Corrected ERD after removal of the fan trap



Pearso

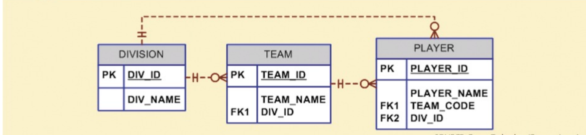
72

Design Case 4: Redundant Relationships

- Redundancy is seldom a good thing in database environment
- Occurs when there are multiple relationship paths between related entities
- Main concern is that redundant relationships remain consistent across model
- Some designs use redundant relationships to simplify the design

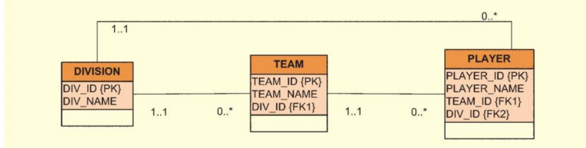
73

FIGURE 5.13 A redundant relationship



SOURCE: Course Technology/Cengage Learning

FIGURE 6.14 A redundant relationship



74

Summary

- Extended entity relationship (EER) model adds semantics to ER model
  - Adds semantics via entity supertypes and subtypes
  - Entity supertype is a generic entity type related to one or more entity subtypes
- Specialization hierarchy
  - Depicts arrangement and relationships between entity supertypes and entity subtypes
- Inheritance means an entity subtype inherits attributes and relationships of supertype

75

Summary (cont'd.)

- Composite keys are useful to represent
  - M:N relationships
- In a 1:1 relationship, place the PK of mandatory entity:
  - As FK in optional entity
  - As FK in entity that causes least number of nulls
  - As FK where the role is played

76

Summary (cont'd.)

- Time-variant data
  - Data whose values change over time
  - Requires keeping a history of changes
- To maintain history of time-variant data:
  - Create entity containing the new value, date of change, other time-relevant data
  - Entity maintains 1:M relationship with entity for which history maintained

77

Summary (cont'd.)

- Fan trap:
  - One entity in two 1:M relationships to other entities
  - Association among the other entities not expressed in model
- Redundant relationships occur when multiple relationship paths between related entities
  - Main concern is that they remain consistent across the model

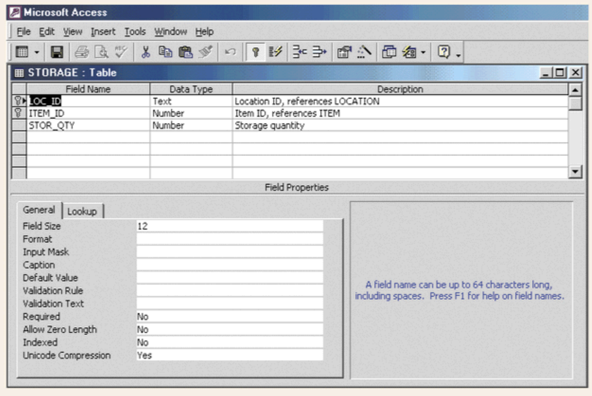
78

Logical Design

- When the conceptual design is completed, the ERD reflects the business rules that define the entities, relationships, optionalities, connectivities, cardinalities and constraints.
- Conceptual model's entities must be normalized before they are implemented.
- Logical design translates conceptual model to format for selected DBMS
- Table structures are created, relationships are established

Pearson

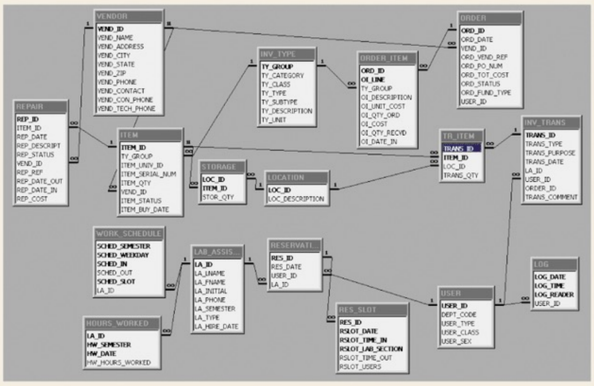
79



The STORAGE Table Structure Defined in Microsoft Access

Pearson

80



The COMPUTER LAB Database

Pearson

81