**Database Chapter Six Outline**

**Design Process**

**Conceptual-design --- Chooses a data model, translates the requirements into a conceptual schema of database**

**Redundancy**

**Incompleteness**

**Modeling**

**A *database* can be modeled as:**

**a collection of entities,**

**relationship among entities.**

**An entity is an object that exists and is distinguishable from other objects.**

**Entities have *attributes***

**An entity set is a set of entities of the same type that share the same properties.**

**Relationship Sets**

**A relationship is an association among several entities**

**A relationship set is a mathematical relation among *n* ≥ 2 entities, each taken from entity sets**

**{(*e*1, *e*2, … *en*) | *e*1 ∈ *E*1, *e*2 ∈ *E*2, …, *en* ∈ *En*}**

**where (*e*1, *e*2, …, *en*) is a relationship**

**The association between entity sets is referred to as participation**

**The entity sets *E1,E2,…,En* participate in relationship set *R***

***Relationship instance* represents an association between the named entities in the real-world enterprise that is being modeled**

**Role---The function that an entity plays in a relationship**

**Recursive relationship set---the same entity set participate in a relationship set more than once**

**Descriptive attribute --An attribute can also be property of a relationship set.**

**Degree-Refers to number of entity sets that participate in a relationship set.**

**Attributes**

**An entity is represented by a set of attributes, that is descriptive properties possessed by all members of an entity set.**

**Domain – the set of permitted values for each attribute**

**Attribute types:**

***Simple* and *composite* attributes.**

***Single-valued* and *multi-valued* attributes**

* + - **Example: multivalued attribute: *phone\_numbers***

***Derived* attributes**

* + - **Can be computed from other attributes**

**Mapping Cardinality Constraints**

**Express the number of entities to which another entity can be associated via a relationship set.**

**For a binary relationship set the mapping cardinality must be one of the following types:**

**One to one**

**One to many**

**Many to one**

**Many to many**

**E-R Diagrams**

**Rectangles represent entity sets.**

**Diamonds represent relationship sets.**

**Lines link attributes to entity sets and entity sets to relationship sets.**

**Ellipses represent attributes**

**Double ellipses represent multivalued attributes.**

**Dashed ellipses denote derived attributes.**

**Underline indicates primary key attributes**

**Roles**

**Role labels are optional, and are used to clarify semantics of the relationship**

**We express cardinality constraints by drawing either a directed line (→), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set.**

**One-To-Many Relationship**

**In the one-to-many relationship a loan is associated with at most one customer via *borrower*, a customer is associated with several (including 0) loans via *borrower***

**Many-To-One Relationships**

**In a many-to-one relationship a loan is associated with several (including 0) customers via *borrower*, a customer is associated with at most one loan via *borrower***

**Many-To-Many Relationship**

**Participation of an Entity Set in a Relationship Set**

**Total participation (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set**

**Partial participation: some entities may not participate in any relationship in the relationship set**

**Design Issues**

**Use of entity sets vs. relationship sets  
Possible guideline is to designate a relationship set to describe an action that occurs between entities**

**May cause replication**

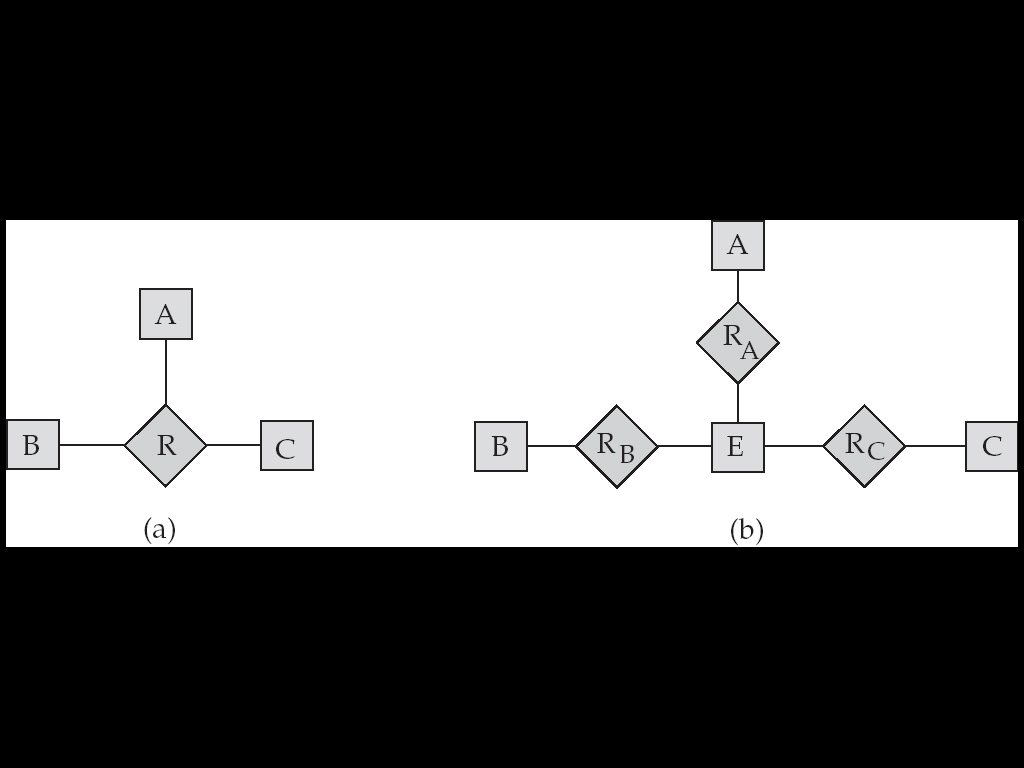
**Binary versus n-ary relationship sets  
Although it is possible to replace any nonbinary (*n*-ary, for *n* > 2) relationship set by a number of distinct binary relationship sets, a *n*-ary relationship set shows more clearly that several entities participate in a single relationship.**

**Placement of relationship attributes**

**For 1-1, reposition to either**

**For 1-m,or m-1, reposition to m side**

**Converting Non-Binary Relationships to Binary Form**

****

**Weak Entity Sets**

**An entity set that does not have a primary key is referred to as a weak entity set.**

**An entity set that does not have a primary key is referred to as a weak entity set. e.g. Employee and Family member, Contract and Attachment**

**The existence of a weak entity set depends on the existence of a identifying entity set**

**it must relate to the identifying entity set via a total, one-to-many relationship set from the identifying to the weak entity set**

**Identifying relationship depicted using a double diamond**

**The discriminator(*or partial key)* of a weak entity set is the set of attributes that distinguishes among all the entities of a weak entity set.**

**标示性联系是从弱实体机到标识实体集的多对一关系，并且弱实体集全部参与。**

**弱实体集主码由标志实体集的主码并上弱实体集的分辨符。**

**Design Constraints on a Specialization/Generalization**

**Constraint on which entities can be members of a given lower-level entity set.**

**condition-defined**

* + - **Example: all customers over 65 years are members of *senior-citizen* entity set; *senior-citizen* ISA *person*.---attribute-defined**

**user-defined**

**Constraint on whether or not entities may belong to more than one lower-level entity set within a single generalization.**

**Disjoint**

* + - **an entity can belong to only one lower-level entity set**
    - **Noted in E-R diagram by writing *disjoint* next to the ISA triangle**

**Overlapping**

* + - **an entity can belong to more than one lower-level entity set**

**Completeness constraint -- specifies whether or not an entity in the higher-level entity set must belong to at least one of the lower-level entity sets within a generalization.**

**total : an entity must belong to one of the lower-level entity sets. We use a double line to connect the box representing the higher-level entity set to the triangle symbol**

**partial: an entity need not belong to one of the lower-level entity sets**

**Aggregation (Cont.)**

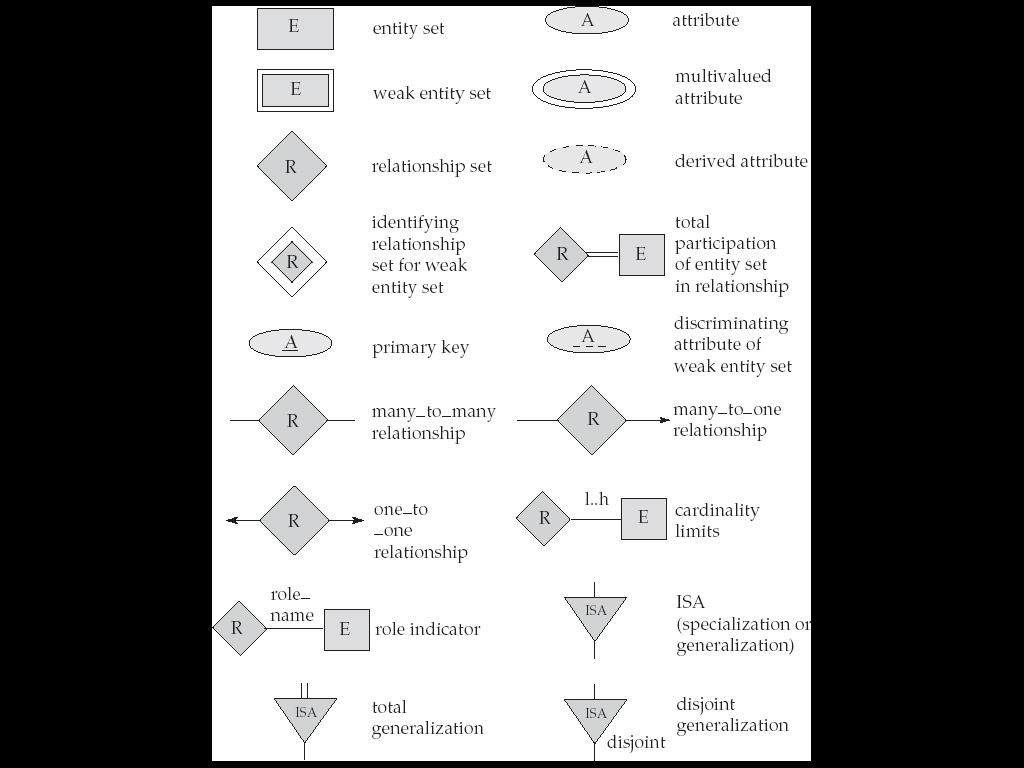
**Eliminate this redundancy via *aggregation***

**Treat relationship as an abstract entity**

**Allows relationships between relationships**

**Abstraction of relationship into new entity**

**Summary of Symbols Used in E-R Notation**

****

**Representing Entity Sets as Schemas**

**A strong entity set reduces to a schema with the same attributes.**

**A weak entity set becomes a table that includes a column for the primary key of the identifying strong entity set**

**Representing Relationship Sets as Schemas**

**A binary many-to-many relationship set is represented as a schema with attributes for the primary keys of the two participating entity sets, and any descriptive attributes of the relationship set.**

**{a1,a2,…,an} ∪{b1,b2,…,bn}**

**参与联系的实体集的主码跟联系自身的属性。**

**Redundancy of Schemas**

**多对一时，可以将多的实体类跟联系集合并在成一个包含了两个模式所有属性的并集的模式。**

**一对一时，联系集的关系模式可以跟参与联系的任何一个实体集的模式进行合并。**

**一般来讲，连接弱实体集与强实体集的联系是冗余的，可以忽略。**

**Representing Specialization via Schemas**

* **Method 1:**

**Form a schema for the higher-level entity**

**Form a schema for each lower-level entity set, include primary key of higher-level entity set and local attributes  
schema attributes  
 *person name, street, city   
 customer name, credit\_rating  
 employee name, salary***

**Drawback: getting information about, an *employee* requires accessing two relations, the one corresponding to the low-level schema and the one corresponding to the high-level schema**

**Schemas Corresponding to Aggregation**

**To represent aggregation, create a schema containing**

**primary key of the aggregated relationship,**

**the primary key of the associated entity set**

**any descriptive attributes**