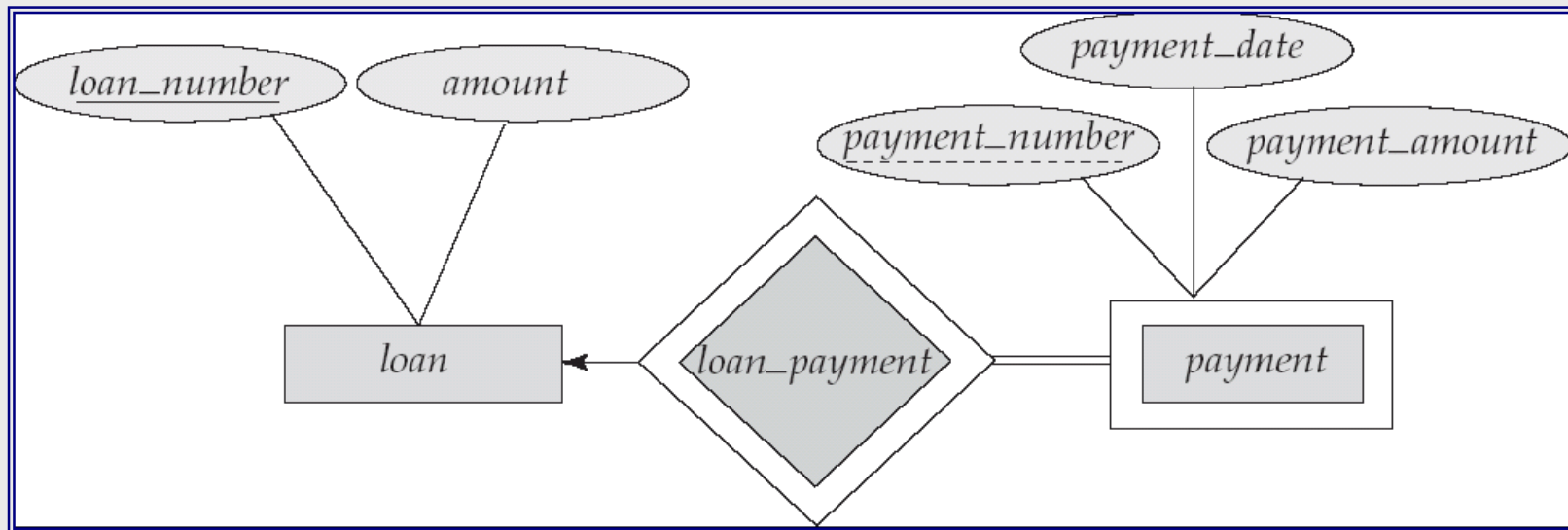


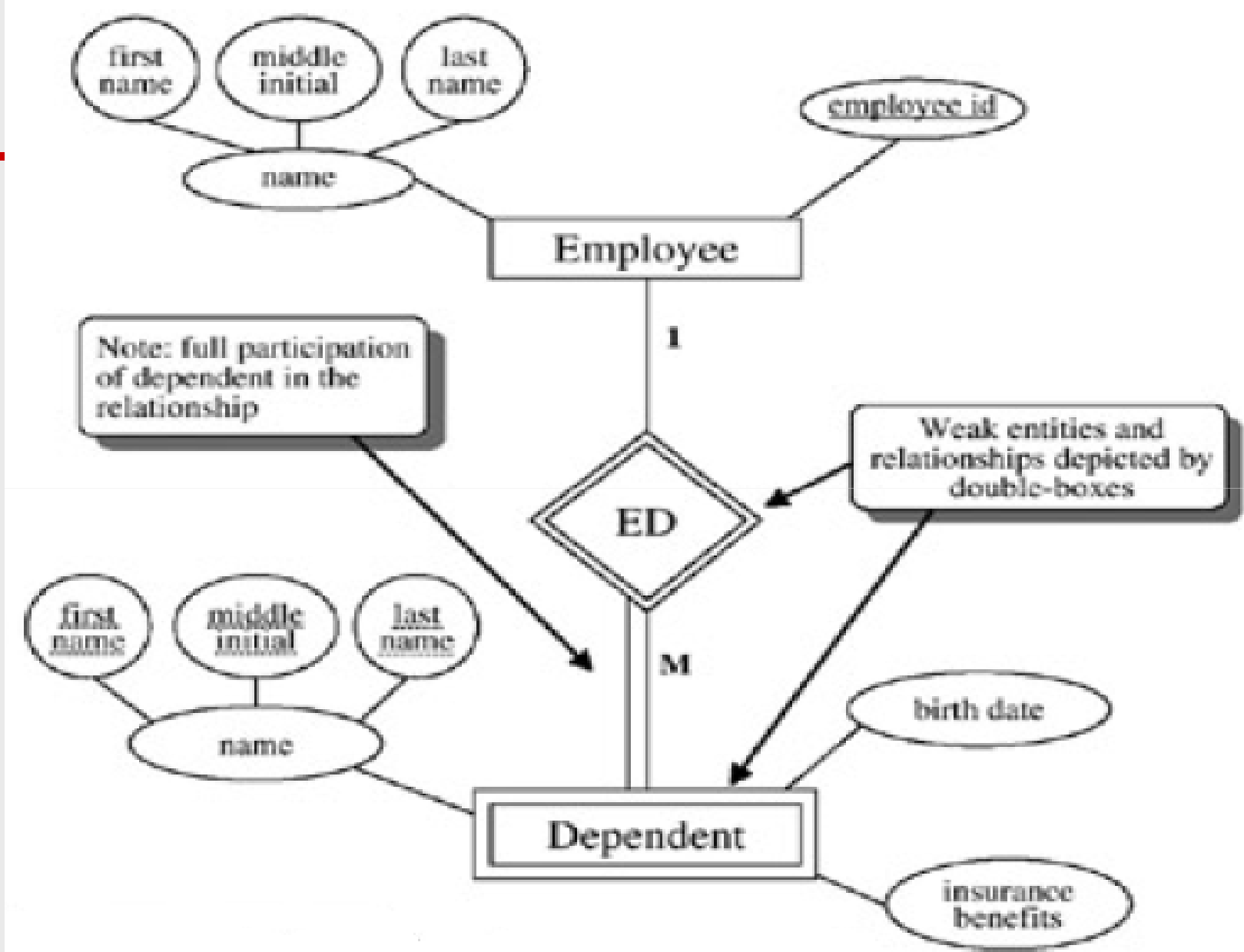
Weak Entity Sets

- u An entity set that does not have a primary key is referred to as a **weak entity set**.
- u 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
- u 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.
- u The primary key of a weak entity set is formed by the primary key of the strong entity set on which the weak entity set is existence dependent, plus the weak entity set's discriminator.

Weak Entity Sets

- u We depict a weak entity set by double rectangles.
- u We underline the discriminator of a weak entity set with a dashed line.
- u *payment_number* – discriminator of the *payment* entity set
- u Primary key for *payment* – (*loan_number*, *payment_number*)



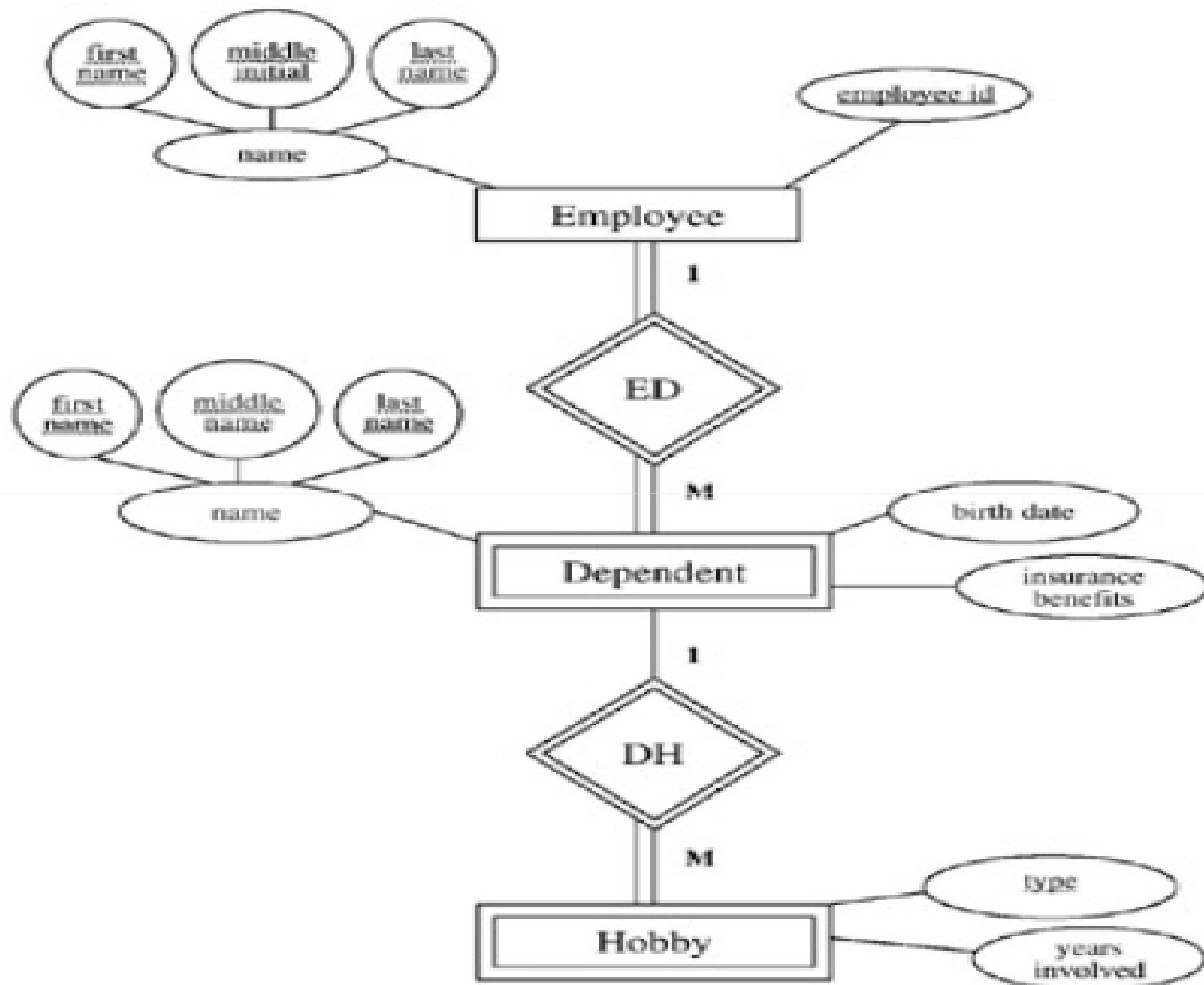


Weak Entity Sets

- u In dealing with weak entities, it is appropriate to consider how each instance of the entity would be identified. Because the owner of the weak entity, DEPENDENT, is the strong entity EMPLOYEE, the identification process would involve the EMPLOYEE key plus some information from the weak entity, DEPENDENT. Name is a likely candidate as an identifier for DEPENDENT, and will be called a *partial key*.
- u In Figure , we have dash-underlined the atomic parts of the composite attribute, name. Name is a *partial key as it identifies dependents, but not uniquely*.

More Weak Entity Set Examples

- u In a university, a *course* is a strong entity and a *course_offering* can be modeled as a weak entity
- u The discriminator of *course_offering* would be *semester* (including year) and *section_number* (if there is more than one section)
- u If we model *course_offering* as a strong entity we would model *course_number* as an attribute.
Then the relationship with *course* would be implicit in the *course_number* attribute



Chapter 13 - Objectives

- u Limitations of basic concepts of the ER model and requirements to represent more complex applications using additional data modeling concepts.**
- u Most useful additional data modeling concept of Enhanced ER (EER) model is called specialization/generalization.**
- u A diagrammatic technique for displaying specialization/generalization in an EER diagram using UML.**

Enhanced Entity-Relationship Model

- u Since 1980s there has been an increase in emergence of new database applications with more demanding requirements.**
- u Basic concepts of ER modeling are not sufficient to represent requirements of newer, more complex applications.**
- u Response is development of additional ‘semantic’ modeling concepts.**

The Enhanced Entity-Relationship Model

- u Semantic concepts are incorporated into the original ER model and called the Enhanced Entity-Relationship (EER) model.**
- u Examples of additional concept of EER model is called specialization / generalization.**

Specialization / Generalization

u Superclass

- An entity type that includes one or more distinct subgroupings of its occurrences.

u Subclass

- A distinct subgrouping of occurrences of an entity type.

Specialization / Generalization

- u Superclass/subclass relationship is one-to-one (1:1).**
- u Superclass may contain overlapping or distinct subclasses.**
- u Not all members of a superclass need be a member of a subclass.**

Specialization / Generalization

u Attribute Inheritance

- **An entity in a subclass represents same ‘real world’ object as in superclass, and may possess subclass-specific attributes, as well as those associated with the superclass.**

Specialization / Generalization

- u Specialization**

- Process of maximizing differences between members of an entity by identifying their distinguishing characteristics.**

- u Generalization**

- Process of minimizing differences between entities by identifying their common characteristics.**

AllStaff relation holding details of all staff

Attributes appropriate for all staff

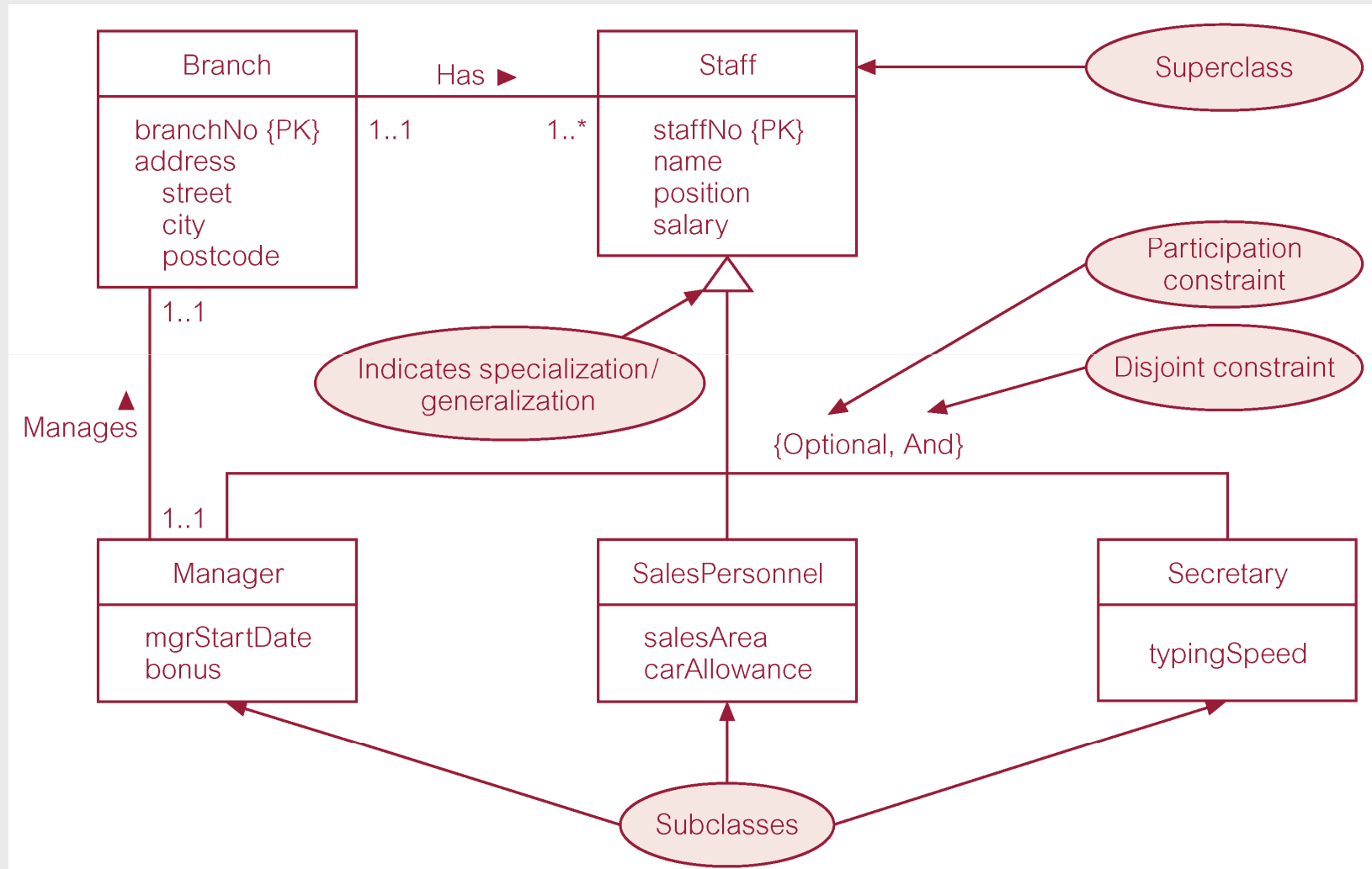
Attributes appropriate for branch Managers

Attributes appropriate for Sales Personnel

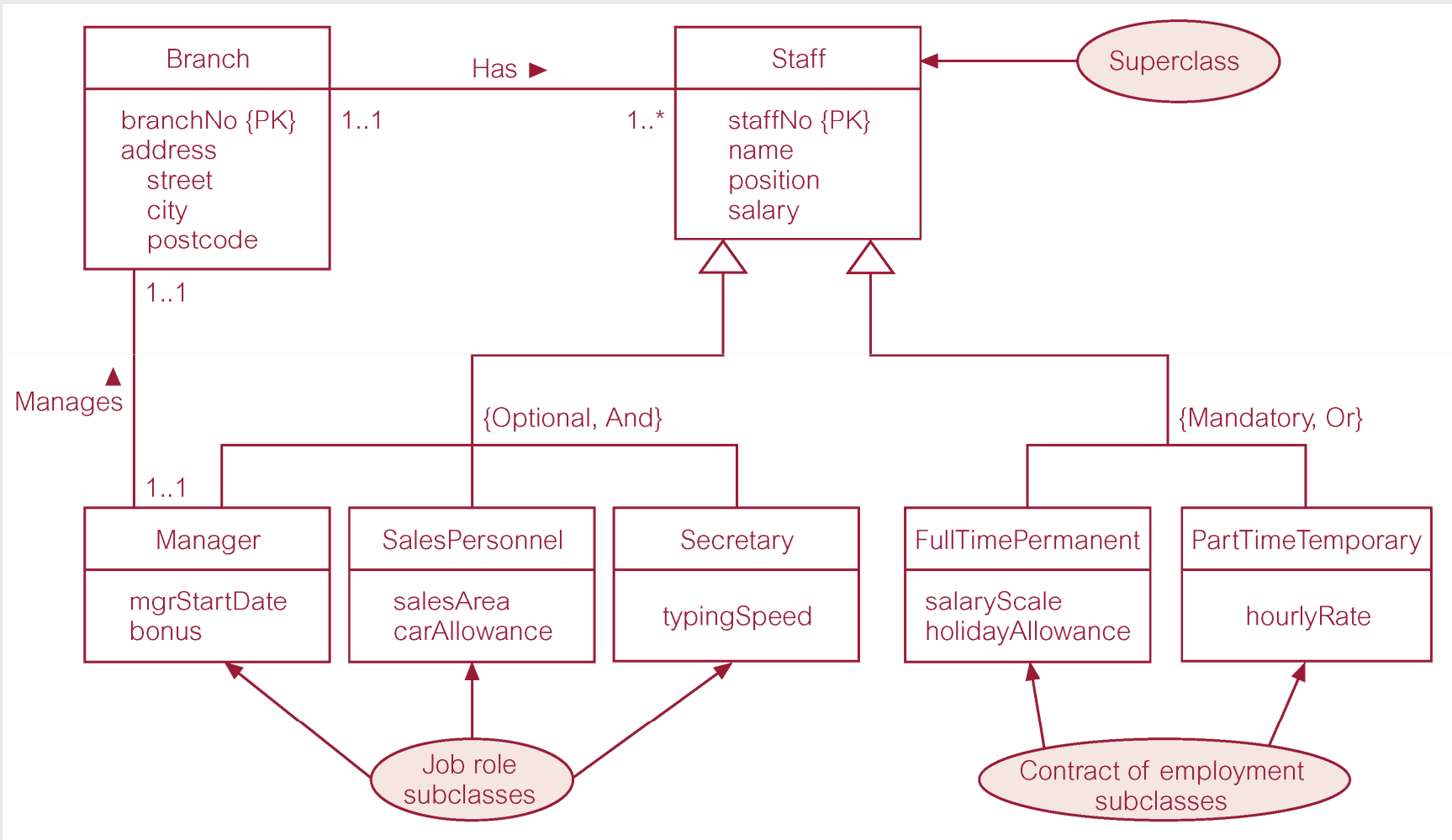
Attribute appropriate for Secretarial staff

staffNo	name	position	salary	mgrStartDate	bonus	sales Area	car Allowance	typing Speed
SL21	John White	Manager	30000	01/02/95	2000			
SG37	Ann Beech	Assistant	12000					
SG66	Mary Martinez	Sales Manager	27000			SA1A	5000	
SA9	Mary Howe	Assistant	9000					
SL89	Stuart Stern	Secretary	8500					100
SL31	Robert Chin	Snr Sales Asst	17000			SA2B	3700	
SG5	Susan Brand	Manager	24000	01/06/91	2350			

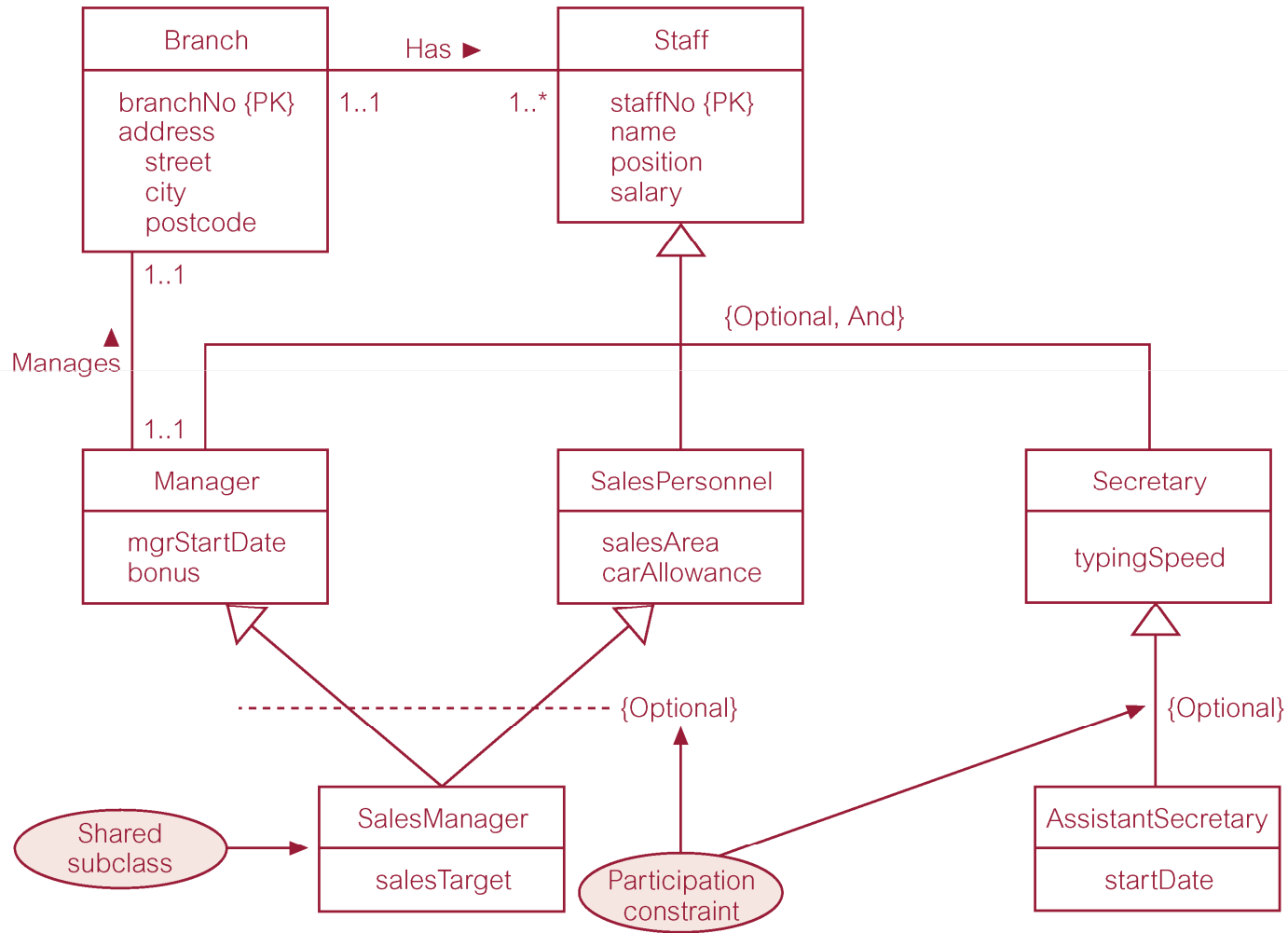
Specialization/generalization of Staff entity into subclasses representing job roles



Specialization/generalization of Staff entity into job roles and contracts of employment



EER diagram with shared subclass and subclass with its own subclass



Extended E-R Features: Specialization

- u Top-down design process; we designate subgroupings within an entity set that are distinctive from other entities in the set.
- u These subgroupings become lower-level entity sets that have attributes or participate in relationships that do not apply to the higher-level entity set.
- u Depicted by a *triangle* component labeled ISA (E.g. *customer “is a” person*).
- u Attribute inheritance – a lower-level entity set inherits all the attributes and relationship participation of the higher-level entity set to which it is linked.

Extended ER Features: Generalization

- u **A bottom-up design process** – combine a number of entity sets that share the same features into a higher-level entity set.
- u Specialization and generalization are simple inversions of each other; they are represented in an E-R diagram in the same way.
- u The terms specialization and generalization are used interchangeably.

Constraints on Specialization / Generalization

- u **Two constraints that may apply to a specialization/generalization:**
 - participation constraints
 - disjoint constraints.

- u **Participation constraint**
 - **Determines whether every member in superclass must participate as a member of a subclass.**
 - **May be *mandatory* or *optional*.**

Constraints on Specialization / Generalization

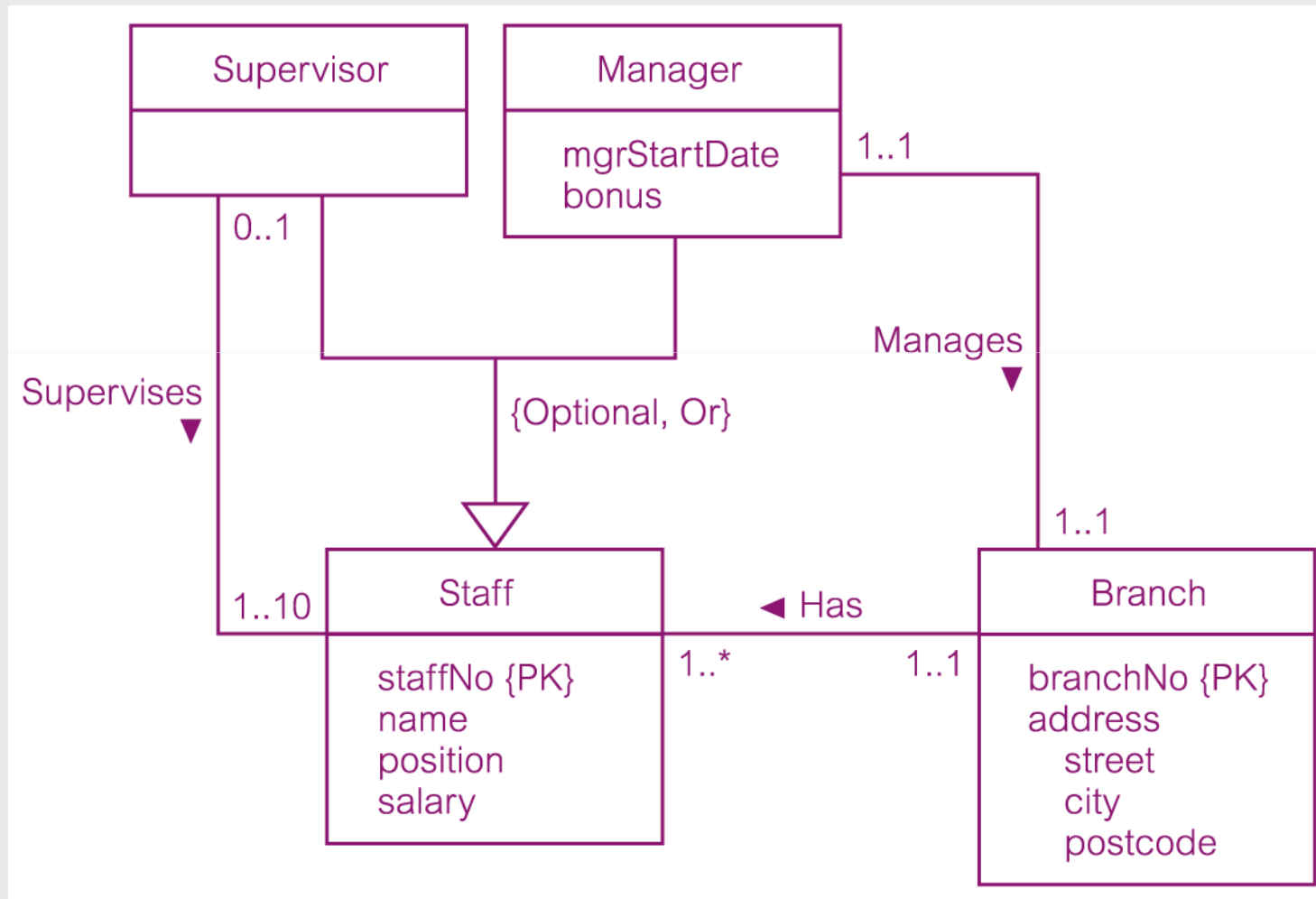
u Disjoint constraint

- Describes relationship between members of the subclasses and indicates whether member of a superclass can be a member of one, or more than one, subclass.
- May be *disjoint* or *nondisjoint*.

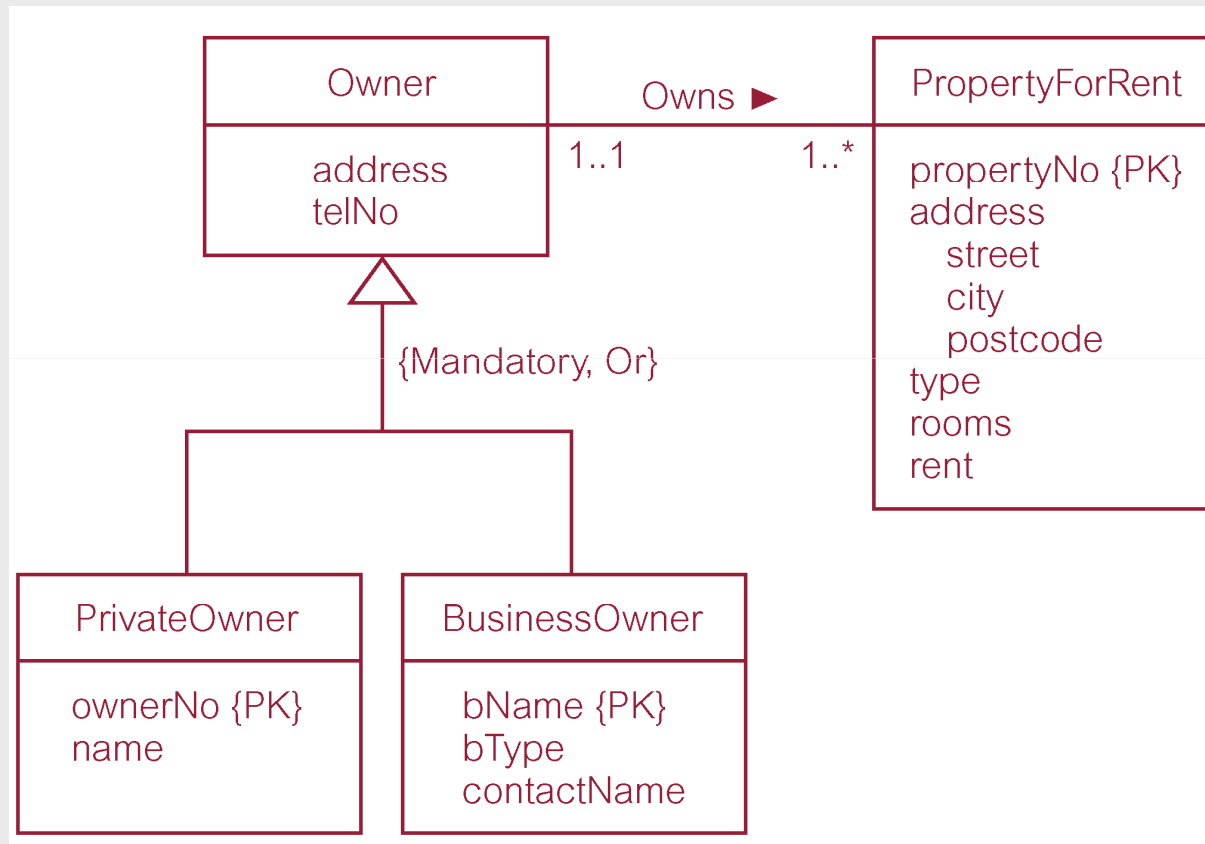
Constraints on Specialization / Generalization

- u There are four categories of constraints of specialization and generalization:**
 - mandatory and disjoint**
 - optional and disjoint**
 - mandatory and nondisjoint**
 - optional and nondisjoint.**

DreamHome worked example - Staff Superclass with Supervisor and Manager subclasses



DreamHome worked example - Owner Superclass with PrivateOwner and BusinessOwner subclasses



DreamHome worked example - Person superclass with Staff, PrivateOwner, and Client subclasses

