



西安电子科技大学
XIDIAN UNIVERSITY

07

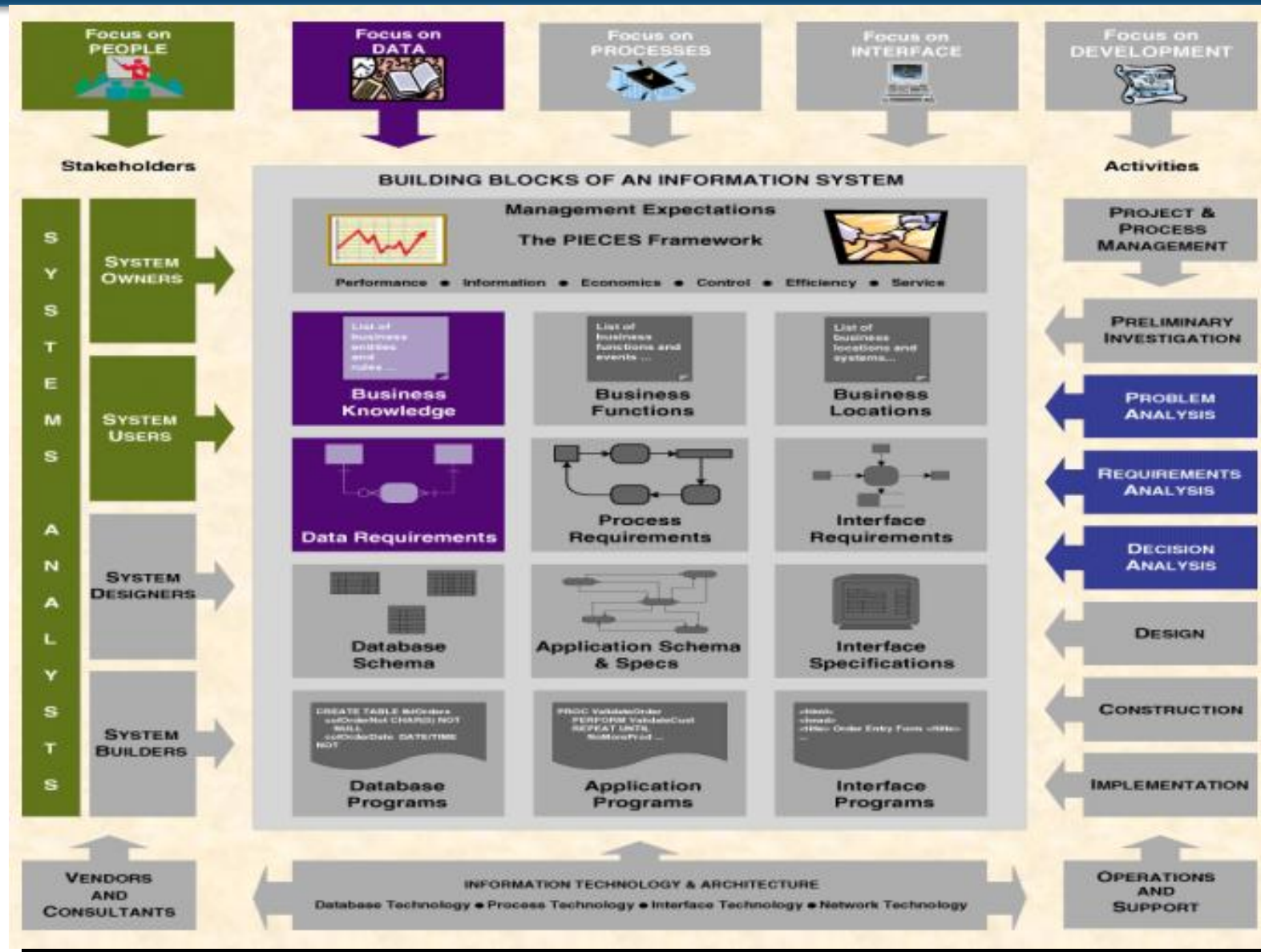
系统分析与设计 (SYSTEM ANALYSIS AND DESIGN)

Data Modeling and Analysis

Content Structure

- ✿ An Introduction to Systems Modeling
- ✿ System Concepts for Data Modeling
 - 实体联系图的各种要素。
- ✿ The Process of Logical Data Modeling
- ✿ How to Construct Data Models?
- ✿ Analyzing the Data Model
 - 将涉及到一些数据库设计的概念。
- ✿ Mapping Data Requirements to Locations
 - 确定数据与其所在地域位置的关系。

Chapter Map





An Introduction to System Models

System Models

- ❁ A **model** (模型) is a representation of reality.
- ❁ **Logical models** show what a system is or does. They are implementation independent; that is, they depict the system independent of any technical implementation. As such, logical model illustrate the *essence* of the system.
- ❁ **Physical models** show not only what a system is or does, but also how the system is physically and technically implemented. They are implementation dependent because they reflect technology choices and the limitations of those technology choices.

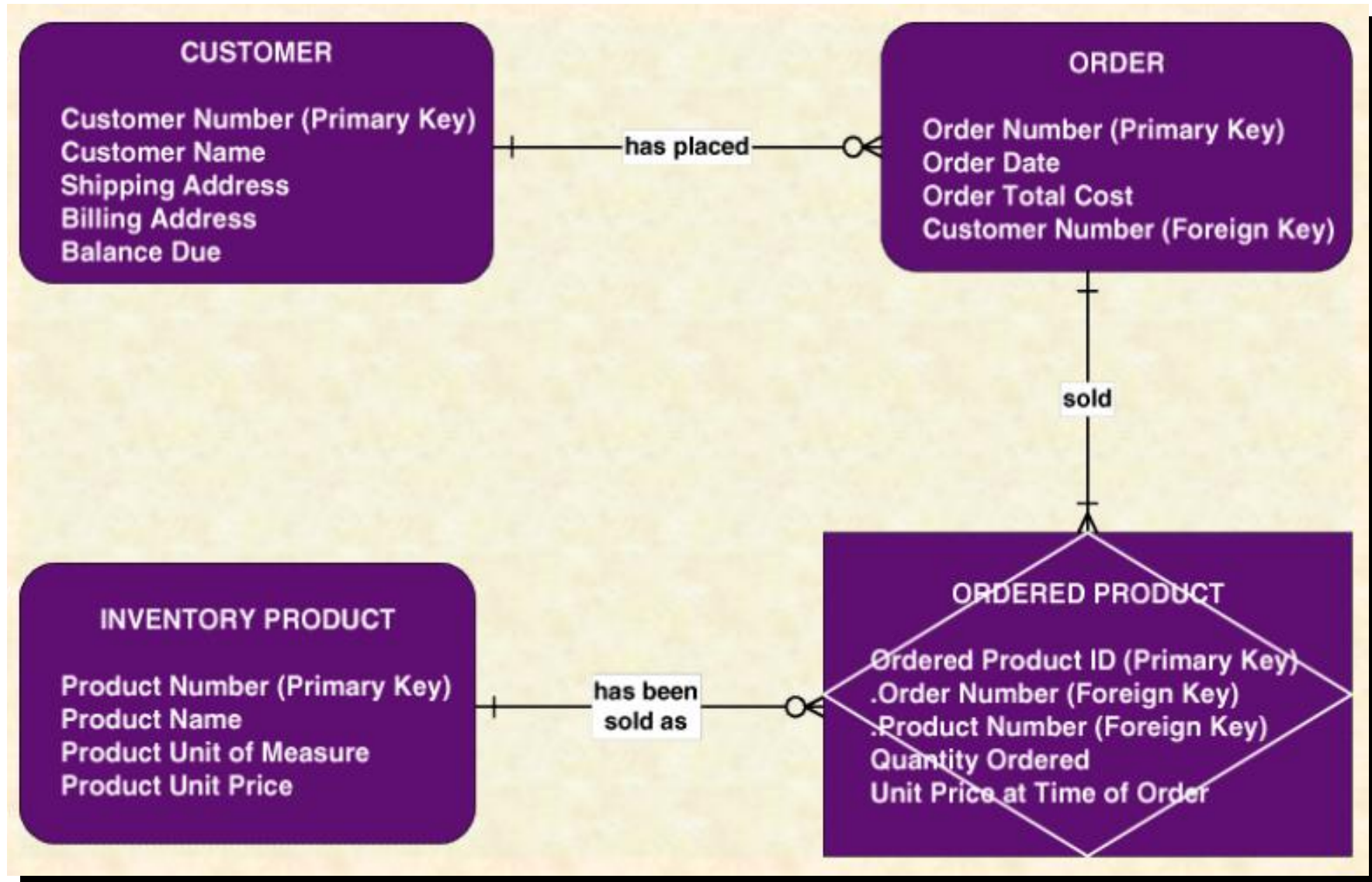
The Reasons for Introducing the Logical Models

- ❁ Logical models remove biases that the result of the way the current system is implemented or the way any one person thinks the system might be implemented.
- ❁ Logical models reduce the risk of missing business requirements because we are too preoccupied with technical details.
- ❁ Logical models allow us to communicate with end-users in nontechnical or less technical languages.

Data Modeling

- ❁ **Data Modeling** (数据建模) is a technique for organizing and documenting a system's data. Data modeling is sometimes called database modeling because a data model is eventually implemented as a database. It is sometimes called information modeling.
- ❁ The actual model is frequently called an **Entity Relationship Diagram** (ERD) because it depicts data in terms of the entities and relationships described by the data.

Sample Entity Relationship Diagram (ERD)





System Concepts For Data Modeling

Entity

- ✿ An **entity** (实体) is a class of persons, places, objects, events, or concepts about which we need to capture and store data.

Name of Entity

- **Persons:** agency, contractor, customer, department, division, employee, instructor, student, supplier.
- **Places:** sales region, building, room, branch office, campus.
- **Objects:** book, machine, part, product, raw material, software license, software package, tool, vehicle model, vehicle.
- **Events:** application, award, cancellation, class, flight, invoice, order, registration, renewal, requisition, reservation, sale, trip.
- **Concepts:** account, block of time, bond, course, fund, qualification, stock.

Entity

- ✿ An **entity instance** (实体的实例) is a single occurrence of an entity.

The diagram illustrates the concept of an entity instance. A blue box labeled 'entity' has a line pointing to the first row of the table (the header row). A blue box labeled 'instance' has a bracket pointing to the subsequent six rows of the table, which represent individual student records.

Student ID	Last Name	First Name
2144	Arnold	Betty
3122	Taylor	John
3843	Simmons	Lisa
9844	Macy	Bill
2837	Leath	Heather
2293	Wrench	Tim

Attributes

- ❁ An **attribute** (属性) is a descriptive property or characteristic of an entity. Synonyms include element, property, and field.
- ❁ A **compound attribute** (组合属性) is one that actually consists of other attributes.



Domains

- ❁ The **data type** (数据类型) for an attribute defines what type of data can be stored in that attribute. (See Table 7.1 - the logical data types)
- ❁ The **domain** (定义域) of an attribute defines what values an attribute can legitimately take on. (See Table 7.2)
- ❁ The **default value** (缺省值) for an attribute is the value that will be recorded if not specified by the user. (See Table 7.3)

Identification

- ❁ A **key** (关键字) is an attribute, or a group of attributes, that assumes a unique value for each entity instance.
- ❁ A group of attributes that uniquely identifies an instance of an entity is called a **concatenated key** (组合关键字) .
- ❁ A **candidate key** (候选关键字) is a “candidate to become the primary key” of instances of an entity.
- ❁ A **primary key** (主关键字) is that candidate key that will most commonly be used to uniquely identify a single entity instance
- ❁ Any candidate key that is not selected to become the primary key is called an **alternate key** (备用关键字) .
- ❁ A **subsetting criteria** (子集准则) is an attribute (or concatenated attribute) whose finite values divide all entity instances into useful subsets.

Identification Keys & Subsetting Criteria

STUDENT

Student Number (Primary Key)
Social Security Number (Alternate Key)

Name

.Last Name

.First Name

.Middle Initial

Address

.Street Address

.City

.State or Province

.Country

.Postal Code

Phone Number

.Area Code

.Exchange Number

.Number Within Exchange

Date of Birth

Gender (Subsetting Criteria 1)

Race (Subsetting Criteria 2)

Major (Subsetting Criteria 3)

Grade Point Average

按性别划分

Identification Keys & Subsetting Criteria

STUDENT

Student Number (Primary Key)
Social Security Number (Alternate Key)
Name
 .Last Name
 .First Name
 .Middle Initial
Address
 .Street Address
 .City
 .State or Province
 .Country
 .Postal Code
Phone Number
 .Area Code
 .Exchange Number
 .Number Within Exchange
Date of Birth
Gender (Subsetting Criteria 1)
Race (Subsetting Criteria 2)
Major (Subsetting Criteria 3)
Grade Point Average

按民族划分

Identification Keys & Subsetting Criteria

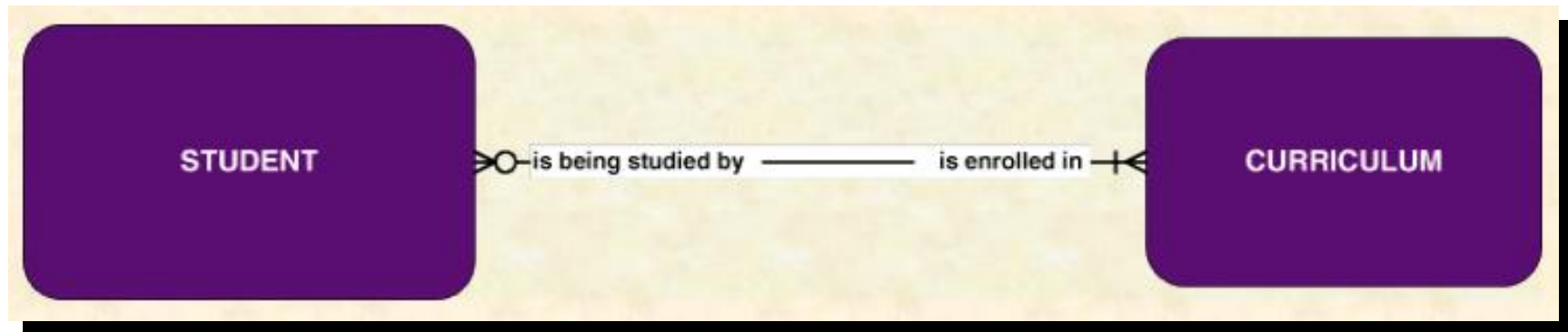
STUDENT

Student Number (Primary Key)
Social Security Number (Alternate Key)
Name
 .Last Name
 .First Name
 .Middle Initial
Address
 .Street Address
 .City
 .State or Province
 .Country
 .Postal Code
Phone Number
 .Area Code
 .Exchange Number
 .Number Within Exchange
Date of Birth
Gender (Subsetting Criteria 1)
Race (Subsetting Criteria 2)
Major (Subsetting Criteria 3)
Grade Point Average

按专业划分

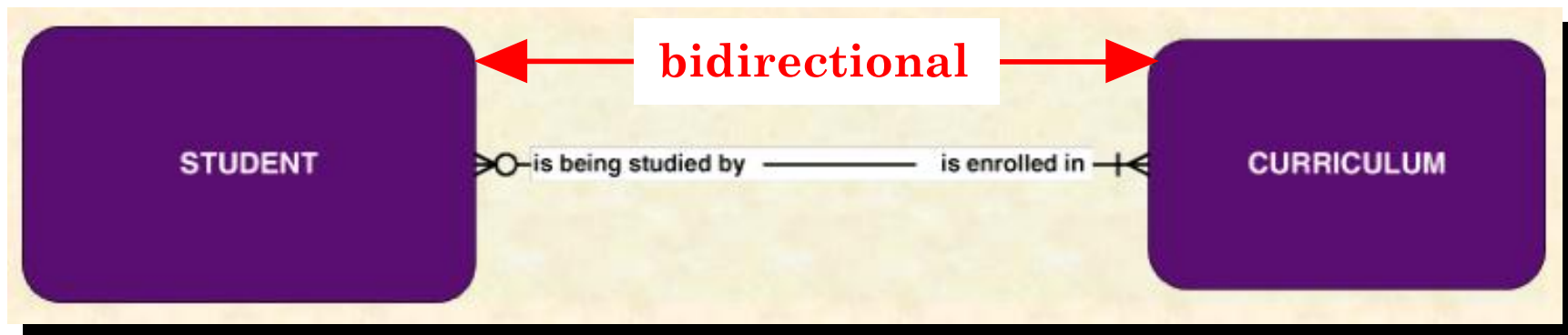
Relationships

- ❁ A **relationship** (联系) is a natural business association (关联) that exists between one or more entities. The relationship may represent an event that links the entities or merely a logical affinity (逻辑上的亲缘关系) that exists between the entities.



Cardinality

- ❁ **Cardinality** (基数) defines the minimum and maximum number of occurrences of one entity that may be related to a single occurrence of the other entity.
- ❁ Because all relationships are bidirectional, cardinality must be defined in both directions for every relationship.

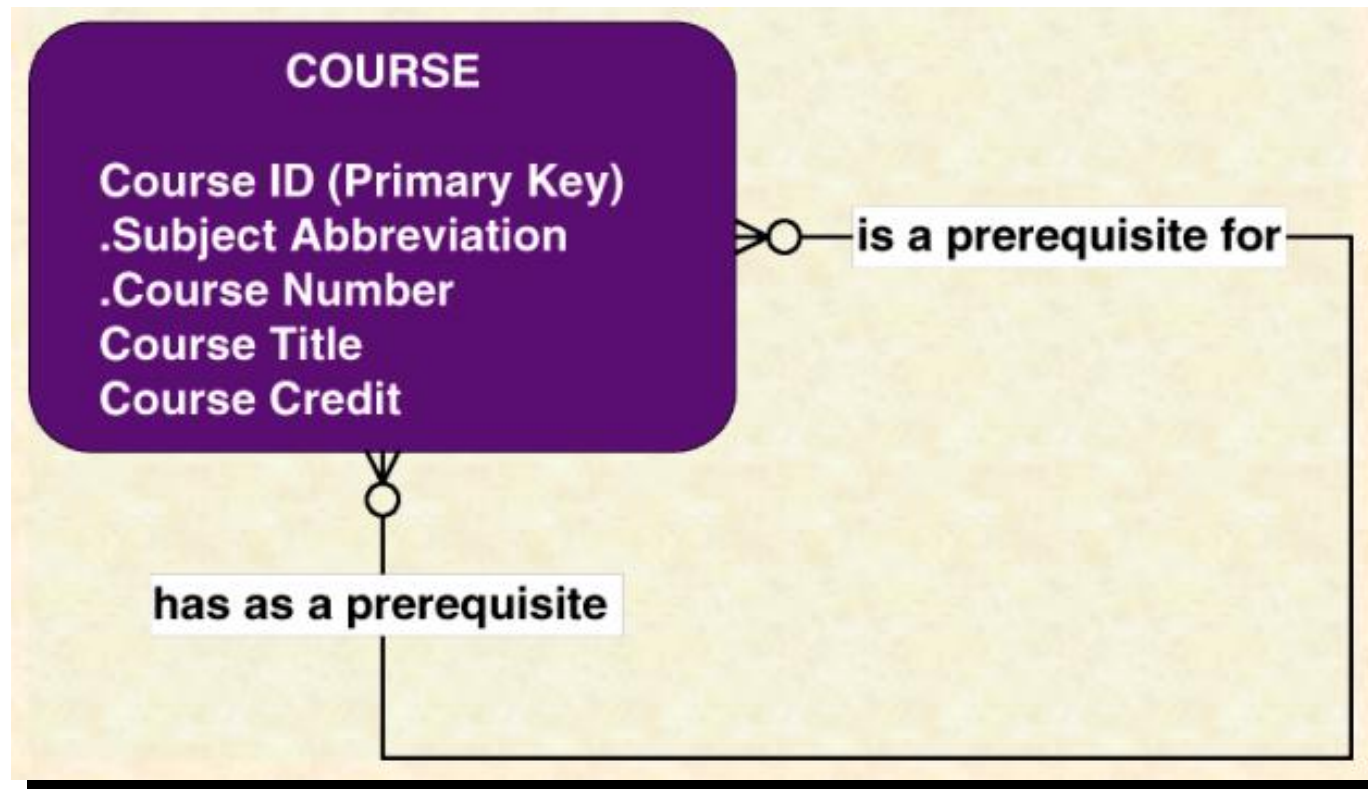


Degree

- ❁ The **degree** (度) of a relationship is the number of entities that participate in the relationship.
- ❁ All the relations we've explored so far are binary (degree = 2).

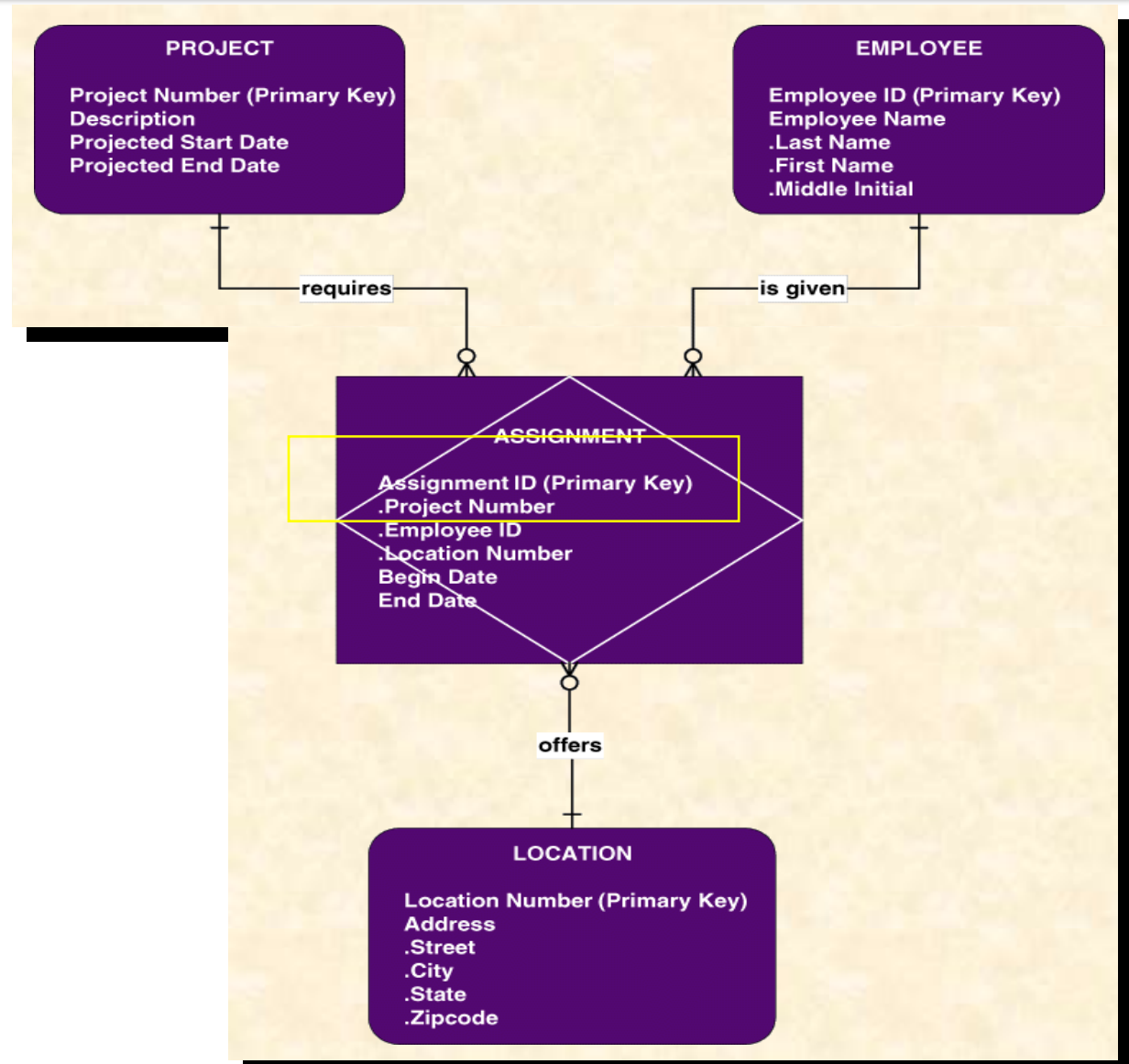
Degree

- ❁ A **recursive relationship** (递归联系) (degree = 1) is a relationship that exists between different instances of the same entity.



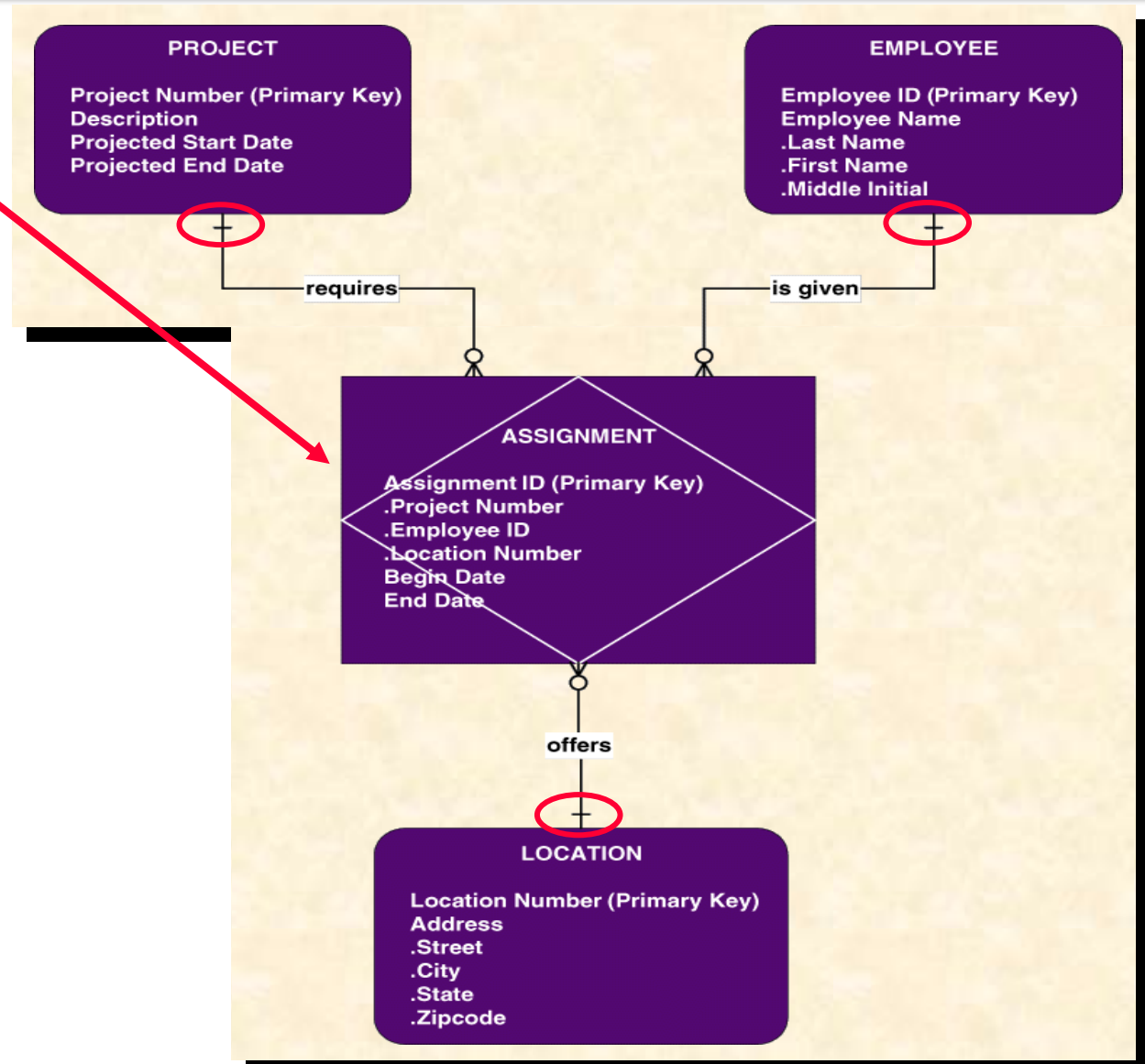
Degree

- ❁ Relationships may exist between more than two entities and are called N-ary (N 度) relationships.
- ❁ The example ERD depicts a ternary (degree = 3) relationship.



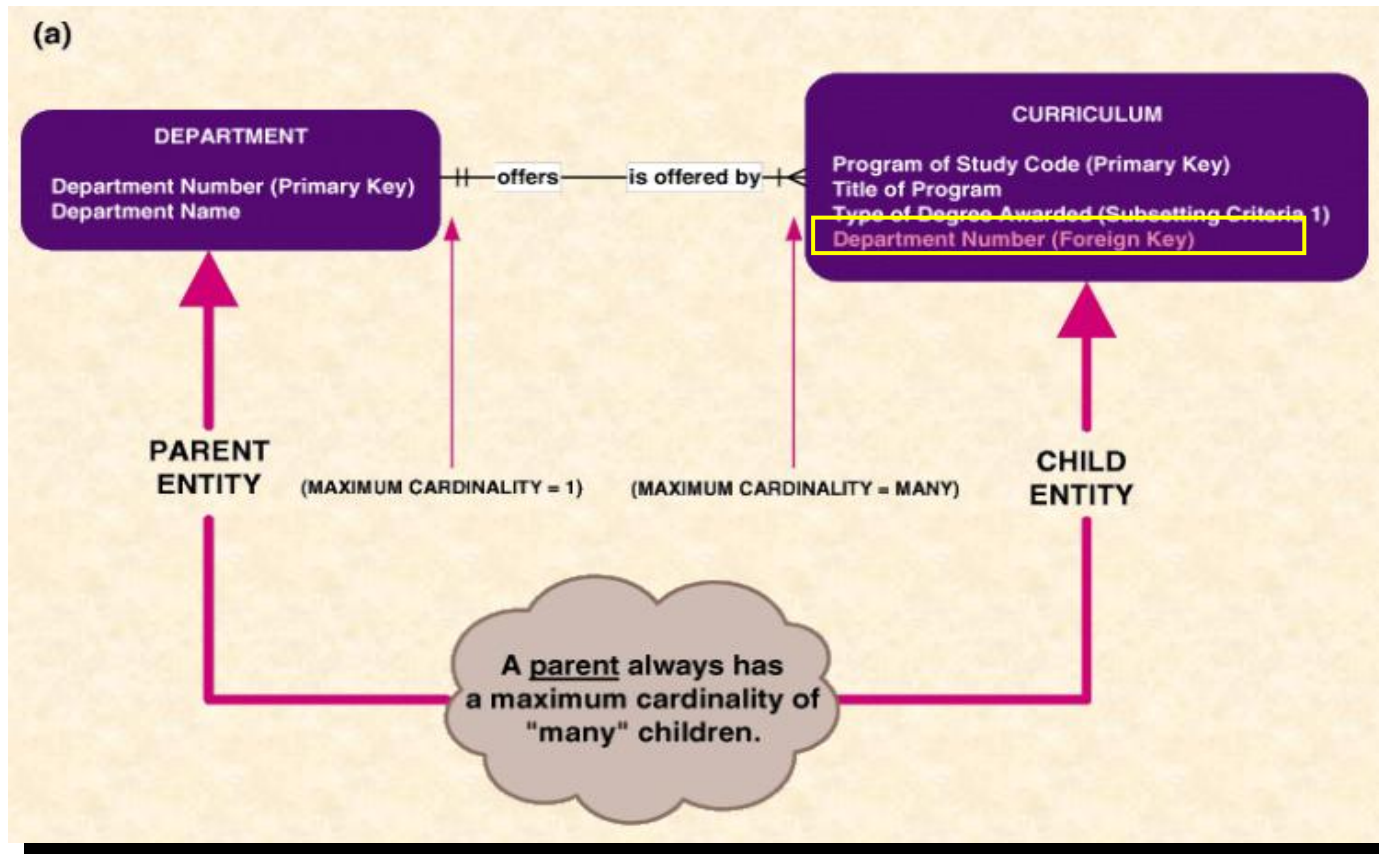
Degree

- ❁ An **associative entity** (关联实体) is an entity that *inherits* its primary key from more than one other entity (called *parents*).
- ❁ Each part of that concatenated key points to *one and only one instance* of each of the connecting entities.



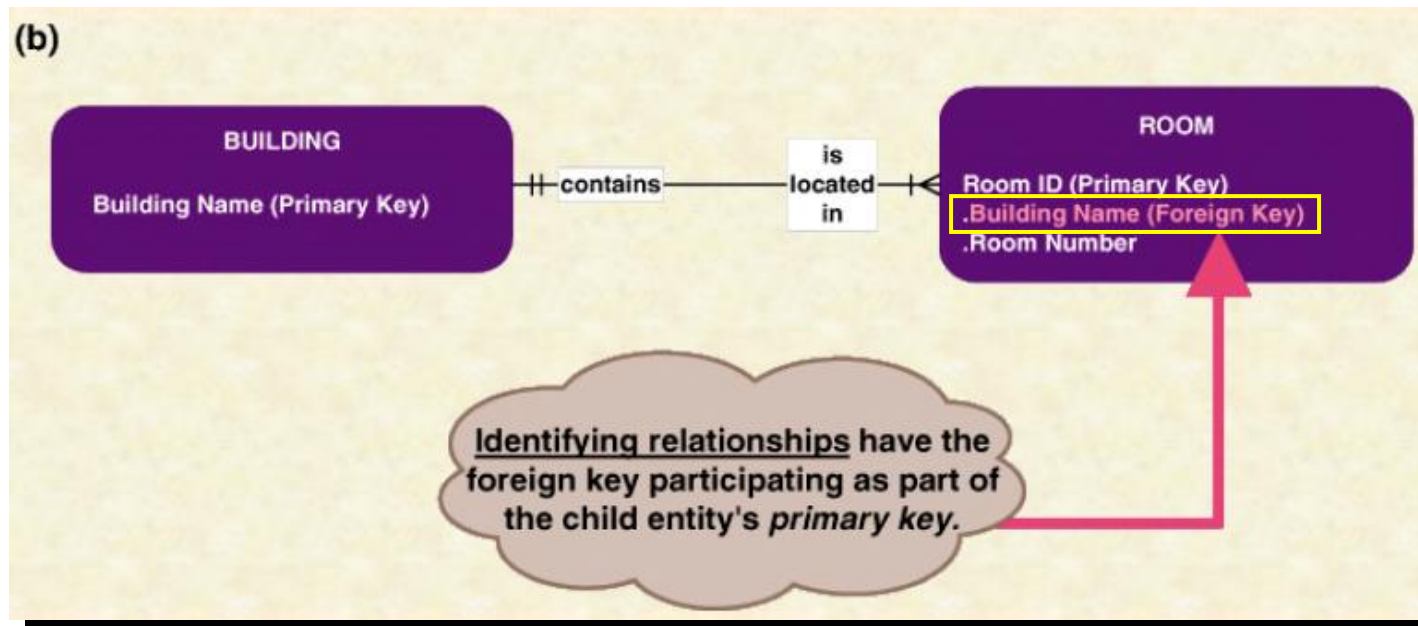
Foreign Keys

- ❁ A **foreign key** (外关键字) is a primary key of one entity that is contributed to (duplicated in) another entity to identify instances of a relationship.



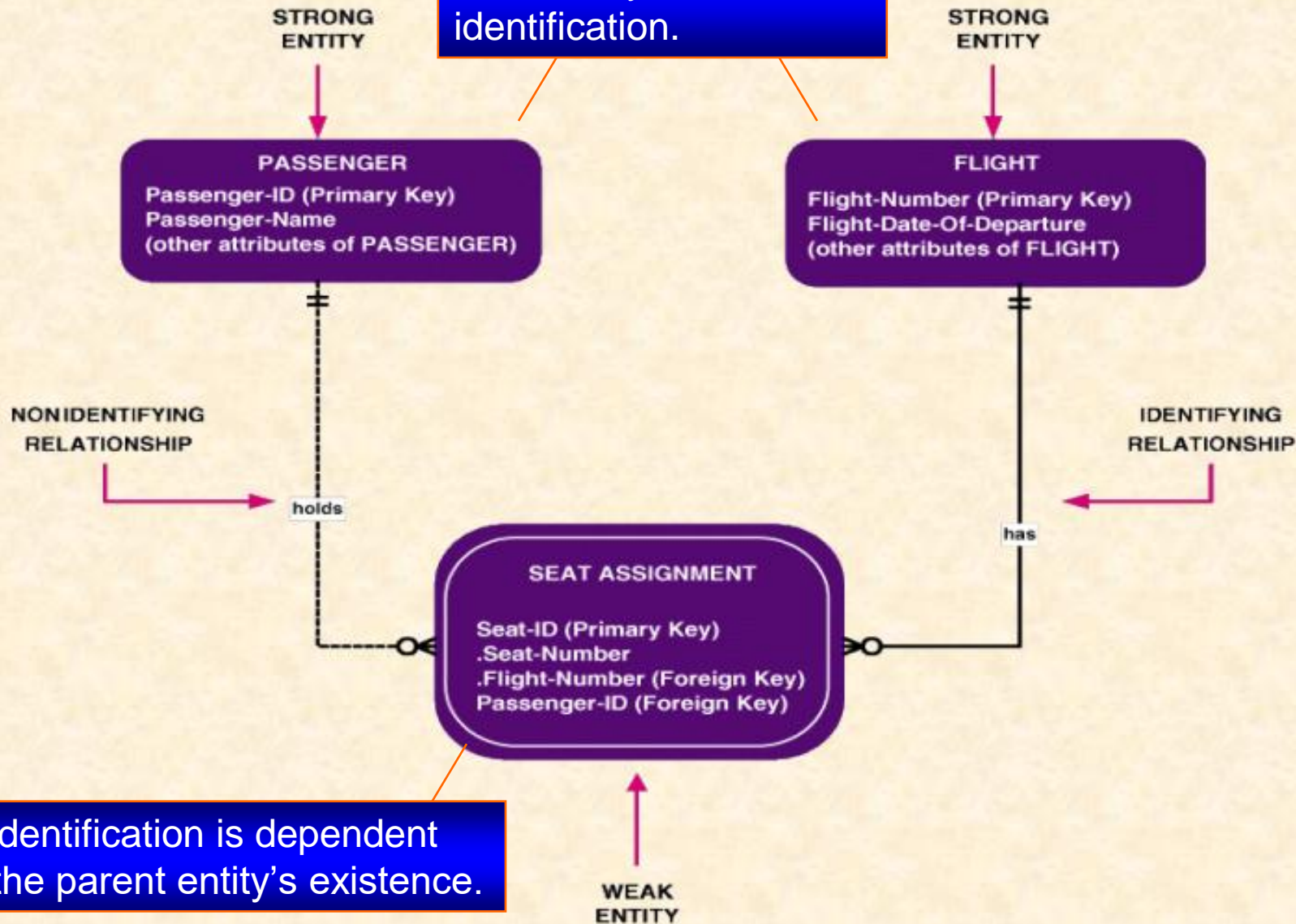
Foreign Keys

- ❁ **Nonidentifying relationships** (非标识性联系) are those in which each of the participating entities has its own independent primary key. In other words, none of the primary key attributes is shared.
- ❁ **Identifying relationships** (标识性联系) are those in which the parent entity contributes its primary key to become part of the primary key of the child entity.



Foreign Keys

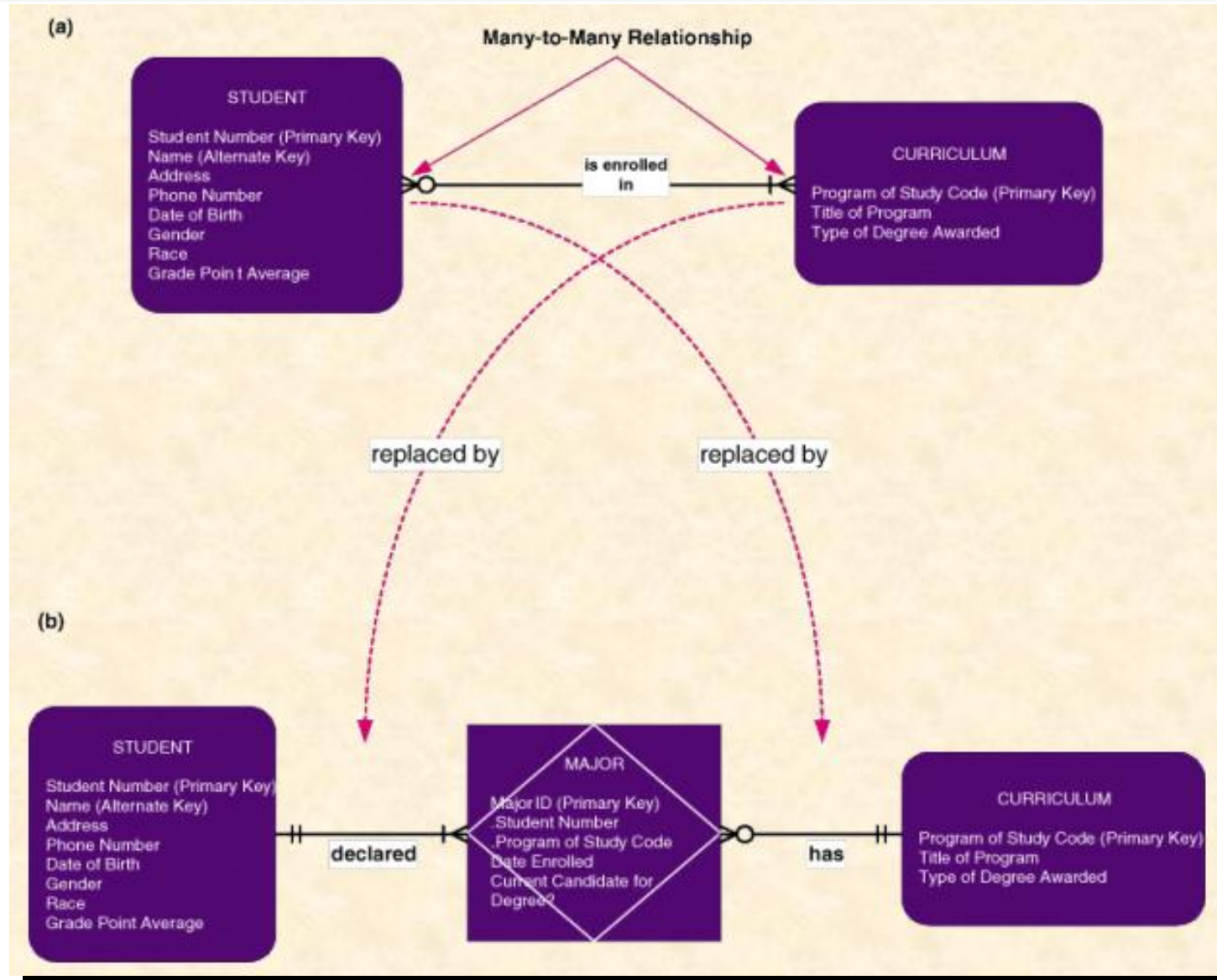
Neither depends on any other entity for its identification.



Resolving Nonspecific Relationships

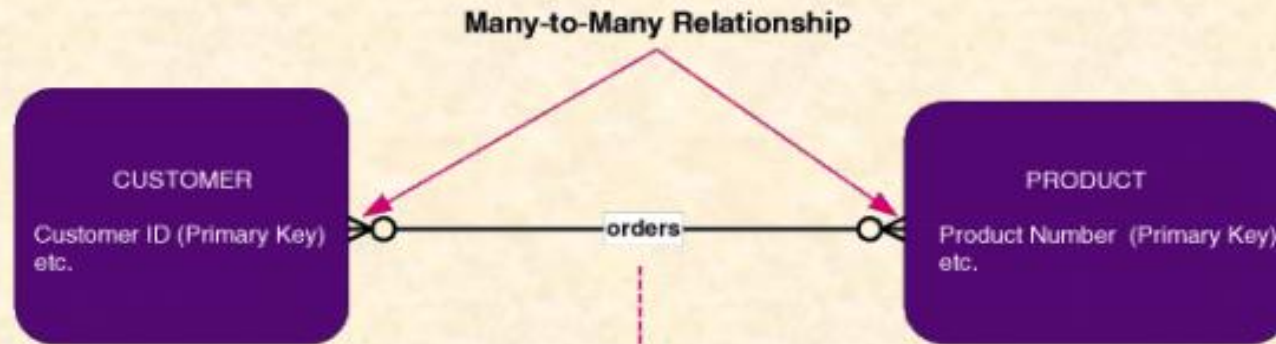
- ❁ A **nonspecific relationship** (or many-to-many relationship) (非特定性联系) is one in which many instances of one entity are associated with many instances of another entity.
- ❁ Nonspecific relationships must be resolved. Most nonspecific relationships can be resolved by introducing an associative entity.
- ❁ Many nonspecific relationships can be resolved into a pair of one-to-many relationships.

Resolving Nonspecific Relationships



Resolving Nonspecific Relationships (continued)

(a)



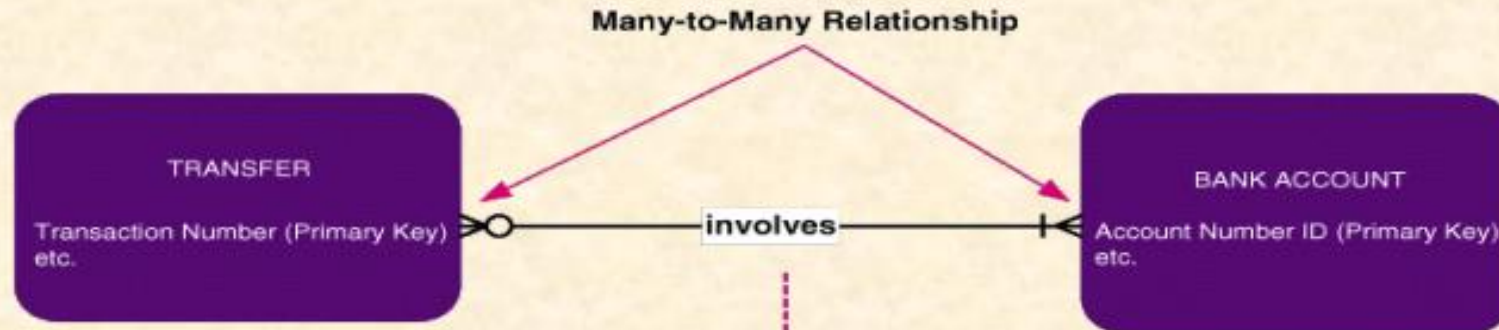
The verb or verb phrase of a many-to-many relationship sometimes suggests other entities. In this example the many-to-many is resolved by recognizing that the verb "orders" actually suggests an *event entity* called ORDER that relates CUSTOMERs to PRODUCTs. Notice that the new many-to-many relationship between ORDER and PRODUCT would need to be resolved.

(b)



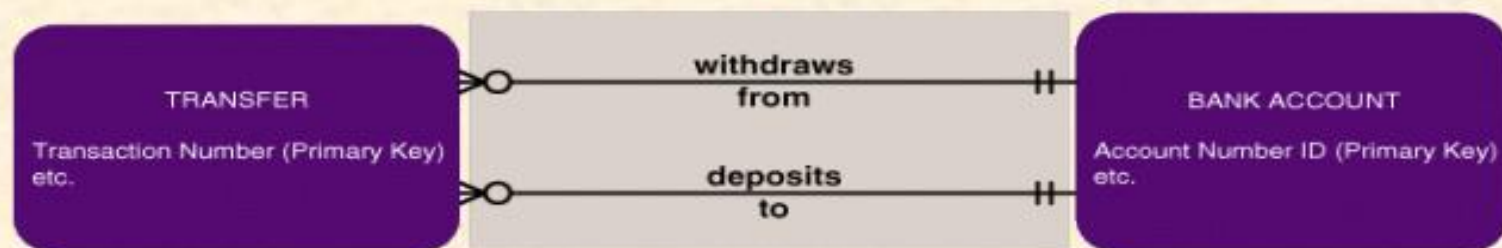
Resolving Nonspecific Relationships (continued)

(a)



While the above relationship is a many-to-many, the many on the **BANK ACCOUNT** side is a known maximum of "2". This suggests that the relationship may actually represent multiple relationships... in this case two *separate* relationships.

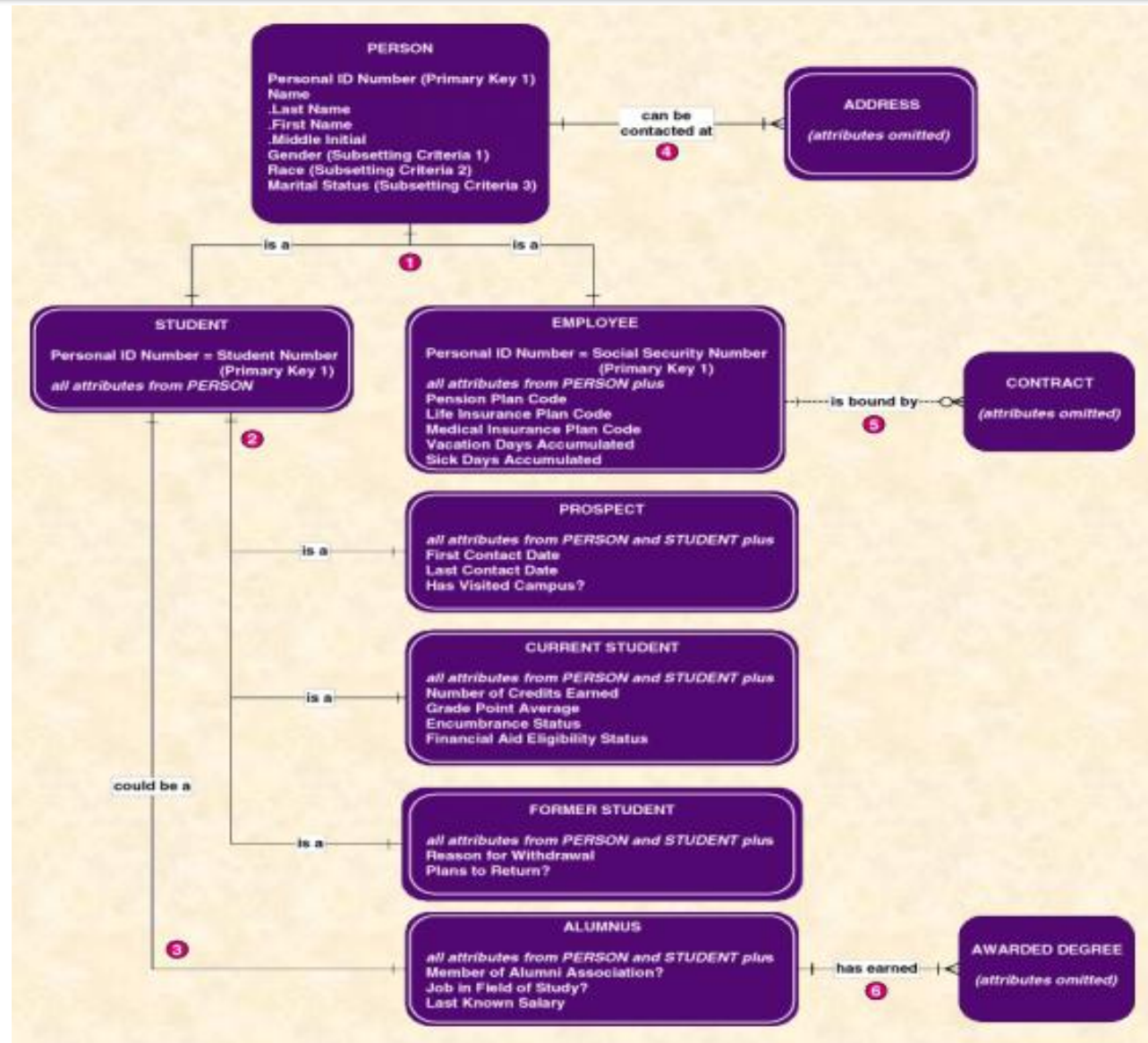
(b)



Generalization

- ❁ **Generalization** (泛化) is a technique wherein the attributes that are common to several types of an entity are grouped into their own entity, called a supertype.
- ❁ An **entity supertype** (超类型实体) is an entity whose instances store attributes that are common to one or more entity subtypes.
- ❁ An **entity subtype** (子类型实体) is an entity whose instances inherit some common attributes from an entity supertype and then add other attributes that are unique to an instance of the subtype.

Generalization Hierarchy





The Process Of Logical Data Modeling

Data Modeling during Systems Analysis

- ❁ The data model for a single information system is usually called an **application data model**.
- ❁ A problem analysis phase model includes only entities and relationships, but no attributes – called a **context data model** (上下文数据模型) .

Stages to Develop a Logical Data Model

- ❁ Construct the **context data model** (to establish project scope).
- ❁ Draw a **key-based data model** (this model will eliminate nonspecific relationships, add associative entities, and include primary and alternate keys).
- ❁ Construct a **fully attributed data model** (this model includes all remaining descriptive attributes and subsetting criteria).
- ❁ Normalization – to construct the **normalized data model** of the system.



How To Construct Data Models?

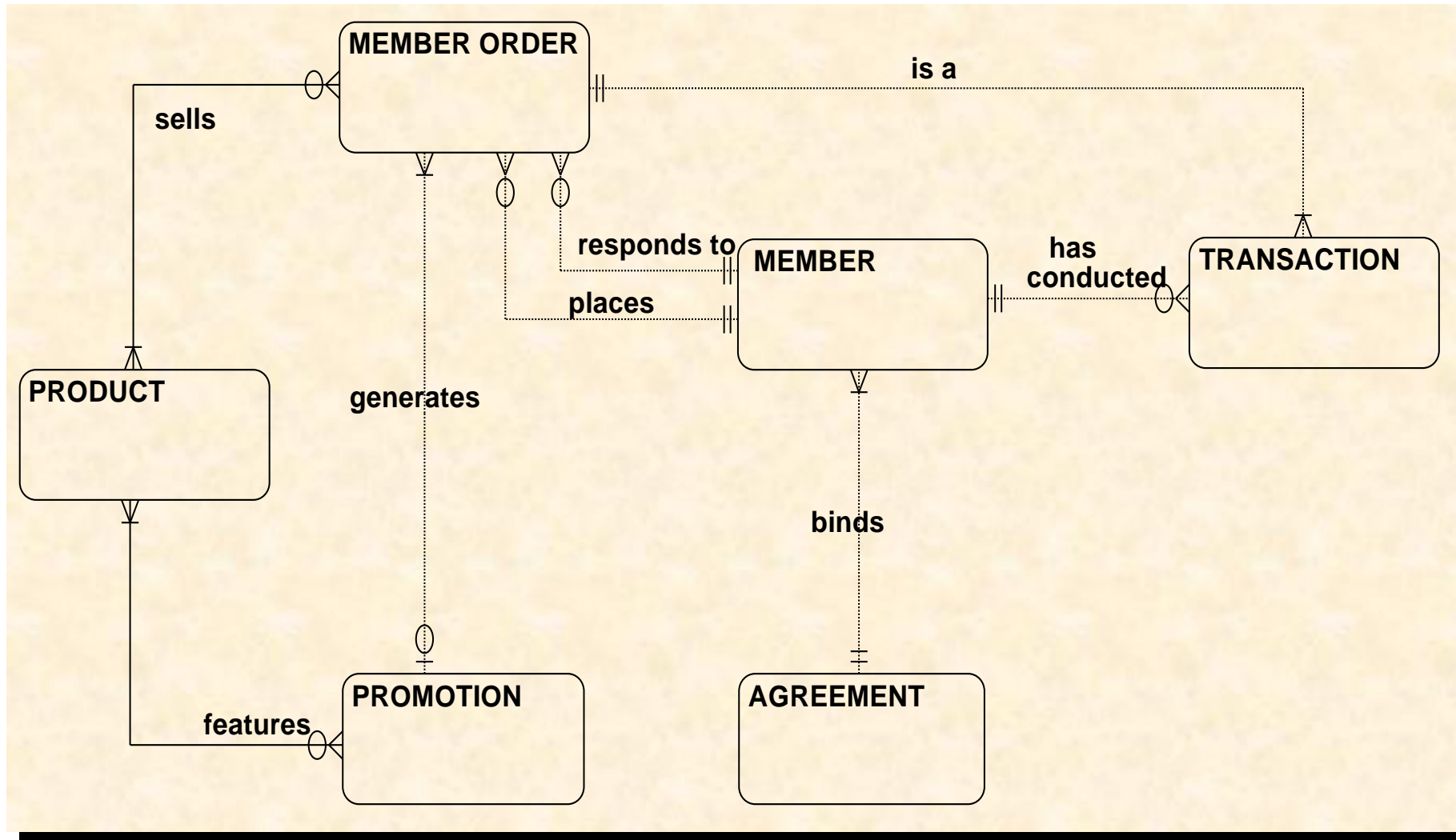
Entity Discovery

- ❁ During interviews or JRP sessions with system owners and users, pay attention to **key words** in their discussion
- ❁ During interviews or JRP sessions, specifically ask system owners and users to identify **things** about which they would like to capture.
- ❁ Study existing **forms, files, and reports**
- ❁ Reverse existing **files** and **databases** into physical data models using some CASE tools.

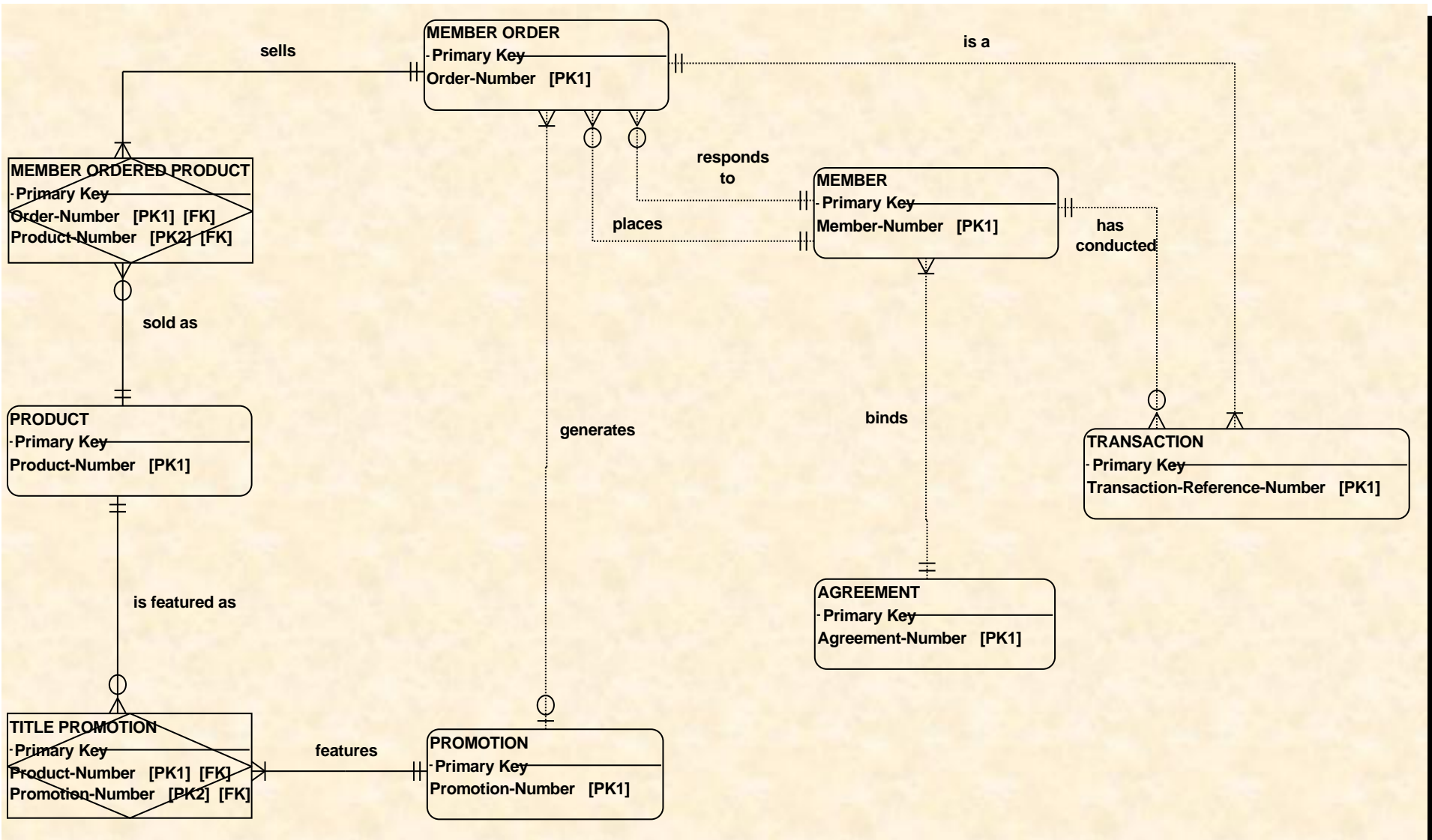
Entity Discovery for SoundStage

Entity Name	Business Definition
agreement	<p>A contract whereby a member agrees to purchase a certain number of products within a certain time. After fulfilling that agreement, the member becomes eligible for bonus credits that are redeemable for free or discounted products.</p>
member	<p>An active member of one or more clubs.</p> <p>Note: A target system objective is to re-enroll inactive members as opposed to deleting them.</p>
member order	<p>An order generated for a member as part of a monthly promotion, or an order initiated by a member.</p> <p>Note: The current system only supports orders generated from promotions; however, customer initiated orders have been given a high priority as an added option in the proposed system.</p>
transaction	<p>A business event to which the Member Services System must respond.</p>
product	<p>An inventoried product available for promotion and sale to members.</p> <p>Note: System improvement objectives include (1) compatibility with new bar code system being developed for the warehouse, and (2) adaptability to a rapidly changing mix of products.</p>
promotion	<p>A monthly or quarterly event whereby special product offerings are made available to members.</p>

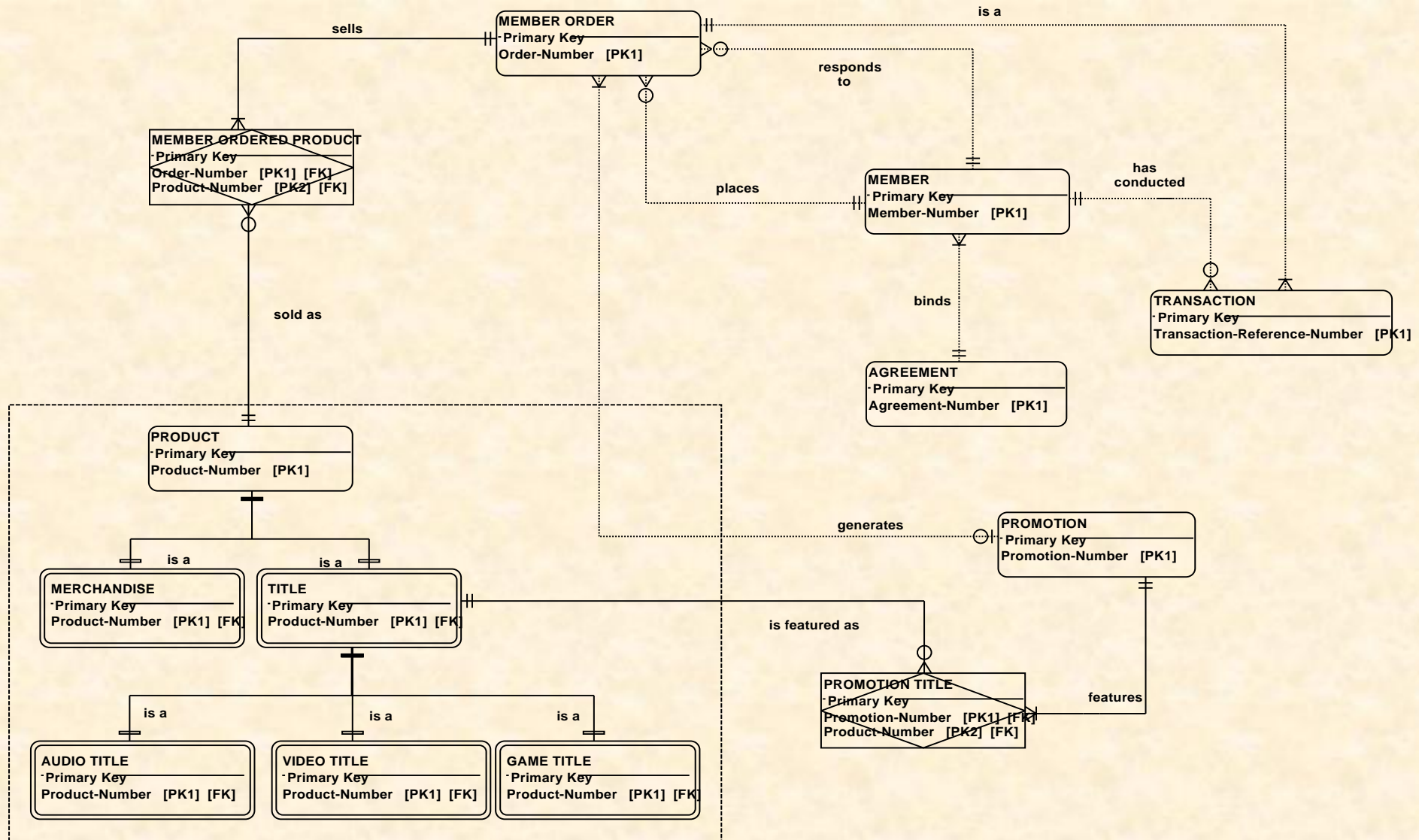
Construct the Context Data Model



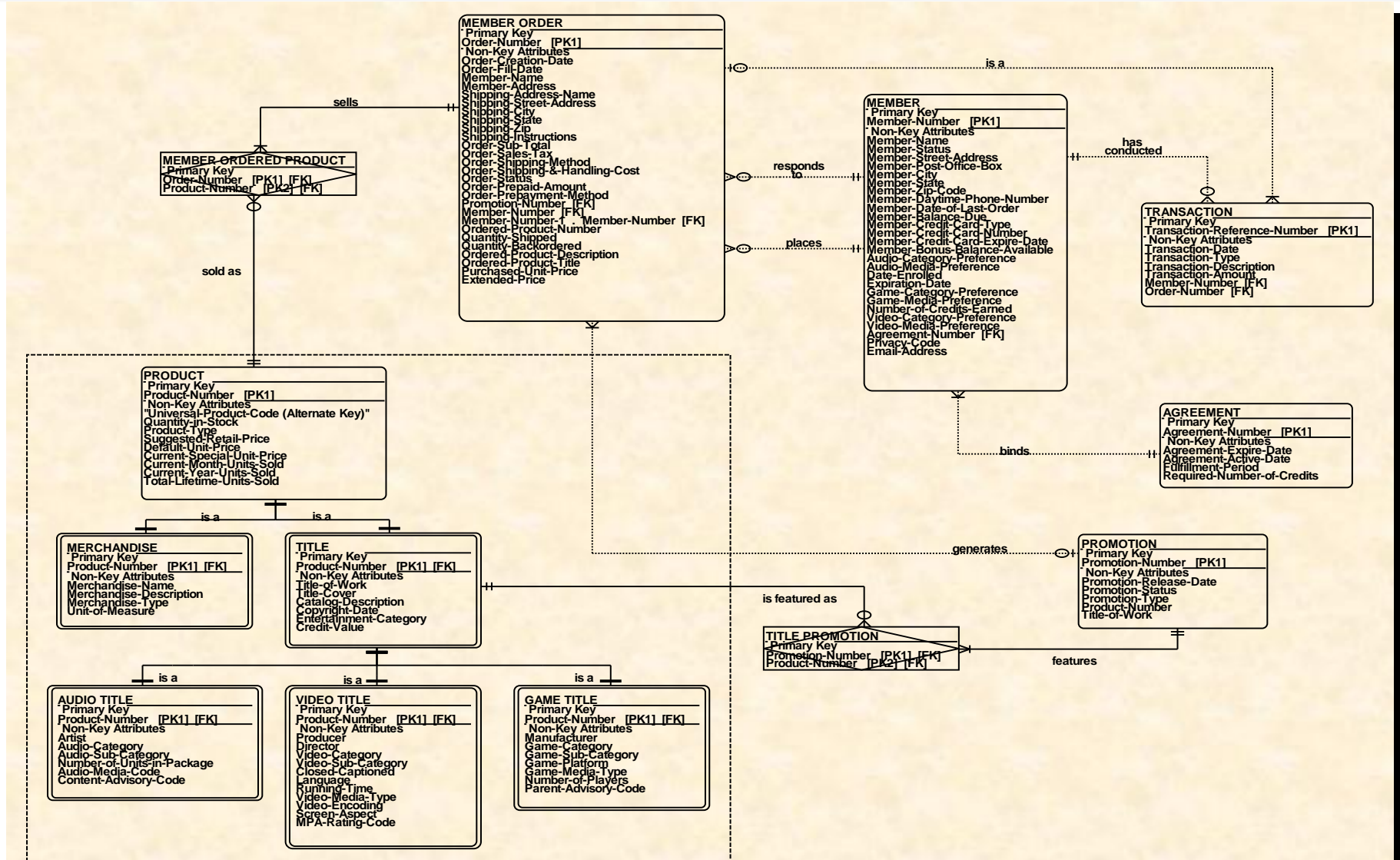
Draw the Key-based Data Model



The Key-based Data Model With Generalization



Construct the Fully-Attributed Data Model





Analyzing The Data Models

What is a Good Data Model?

- ❁ A good data model is simple.
- ❁ A good data model is essentially nonredundant.
- ❁ A good data model should be flexible and adaptable to future needs.

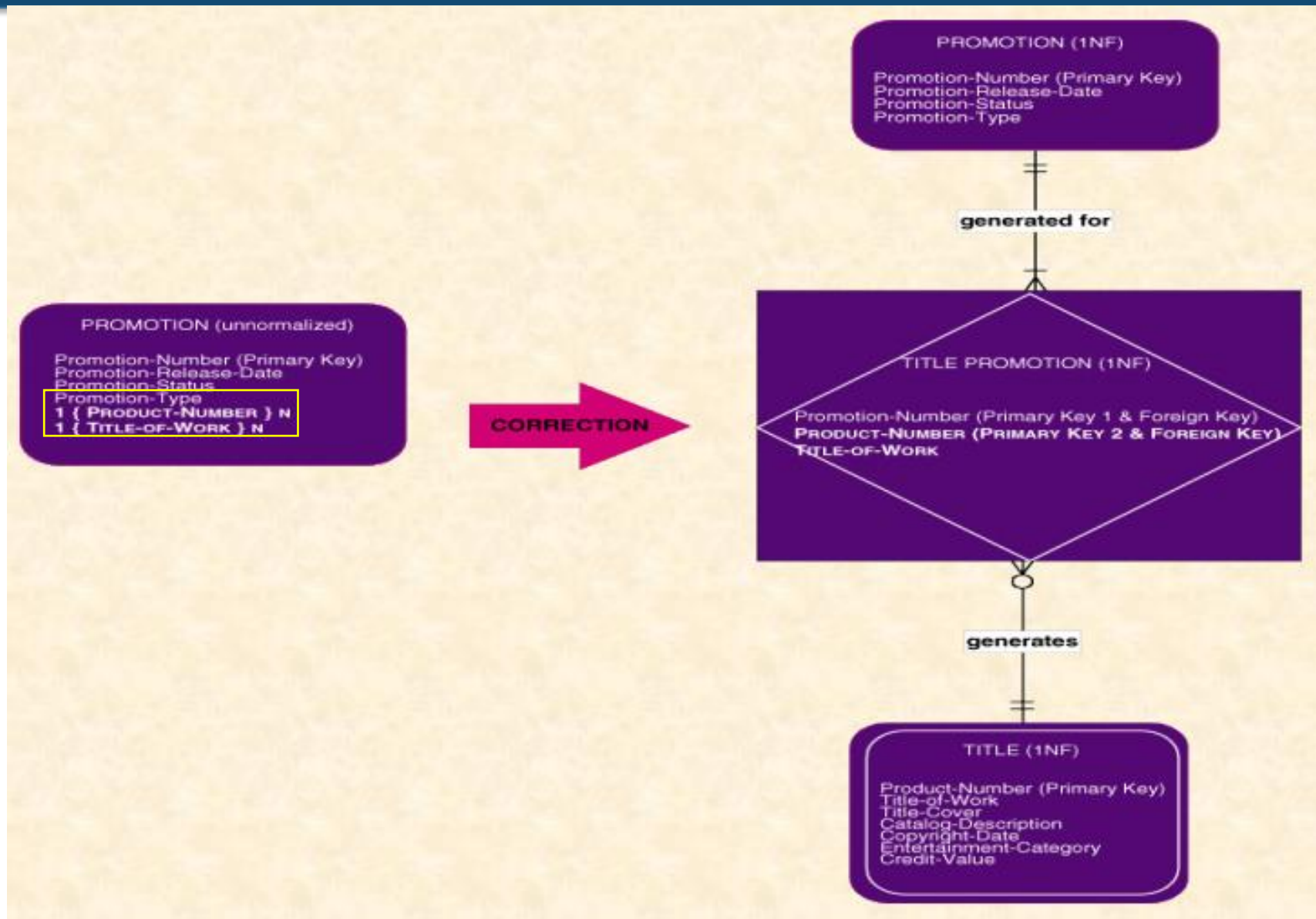
Data Analysis & Normalization

- ❁ **Data analysis** is a process that prepares a data model for implementation as a simple, nonredundant, flexible, and adaptable database. The specific technique is called normalization.
- ❁ **Normalization** is a data analysis technique that organizes data attributes such that they are grouped to form nonredundant, stable, flexible, and adaptive entities.

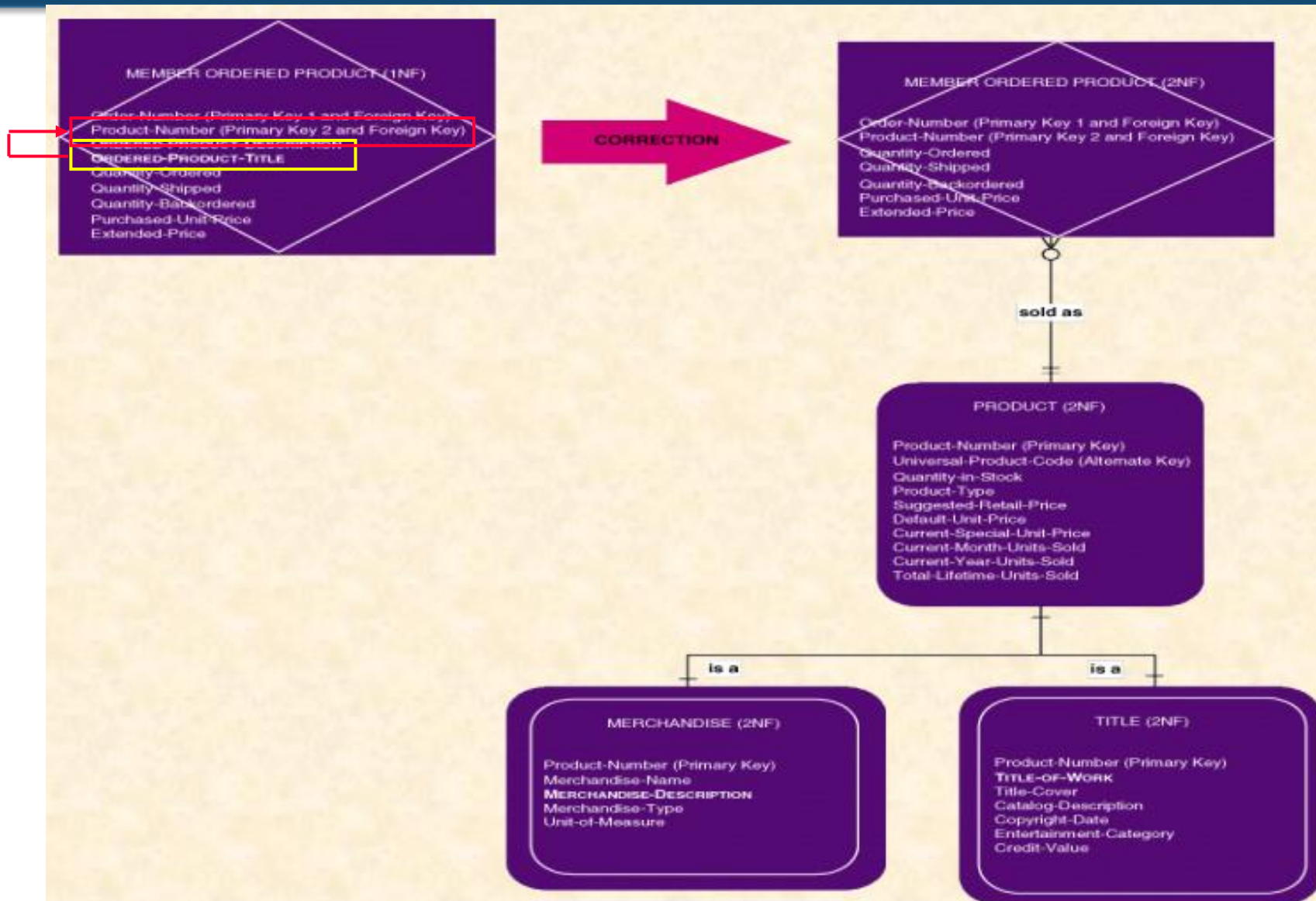
Normalization: 1NF, 2NF, 3NF

- ✿ An entity is in first normal form (1NF) if there are **no attributes that can have more than one value for a single instance of the entity**. Any attributes that can have multiple values actually describe a separate entity, possibly an entity and relationship.
- ✿ An entity is in second normal form (2NF) if it is already in 1NF and if **the values of all nonprimary key attributes are dependent on the full primary key—not just part of it**. Any nonkey attributes that are dependent on only part of the primary key should be moved to any entity where that partial key is actually the full key. This may require creating a new entity and relationship on the model.
- ✿ An entity is in third normal form (3NF) if it is already in 2NF and if **the values of its nonprimary key attributes are not dependent on any other non-primary key attributes**. Any nonkey attributes that are dependent on other nonkey attributes must be moved or deleted. Again, new entities and relationships may have to be added to the data model.

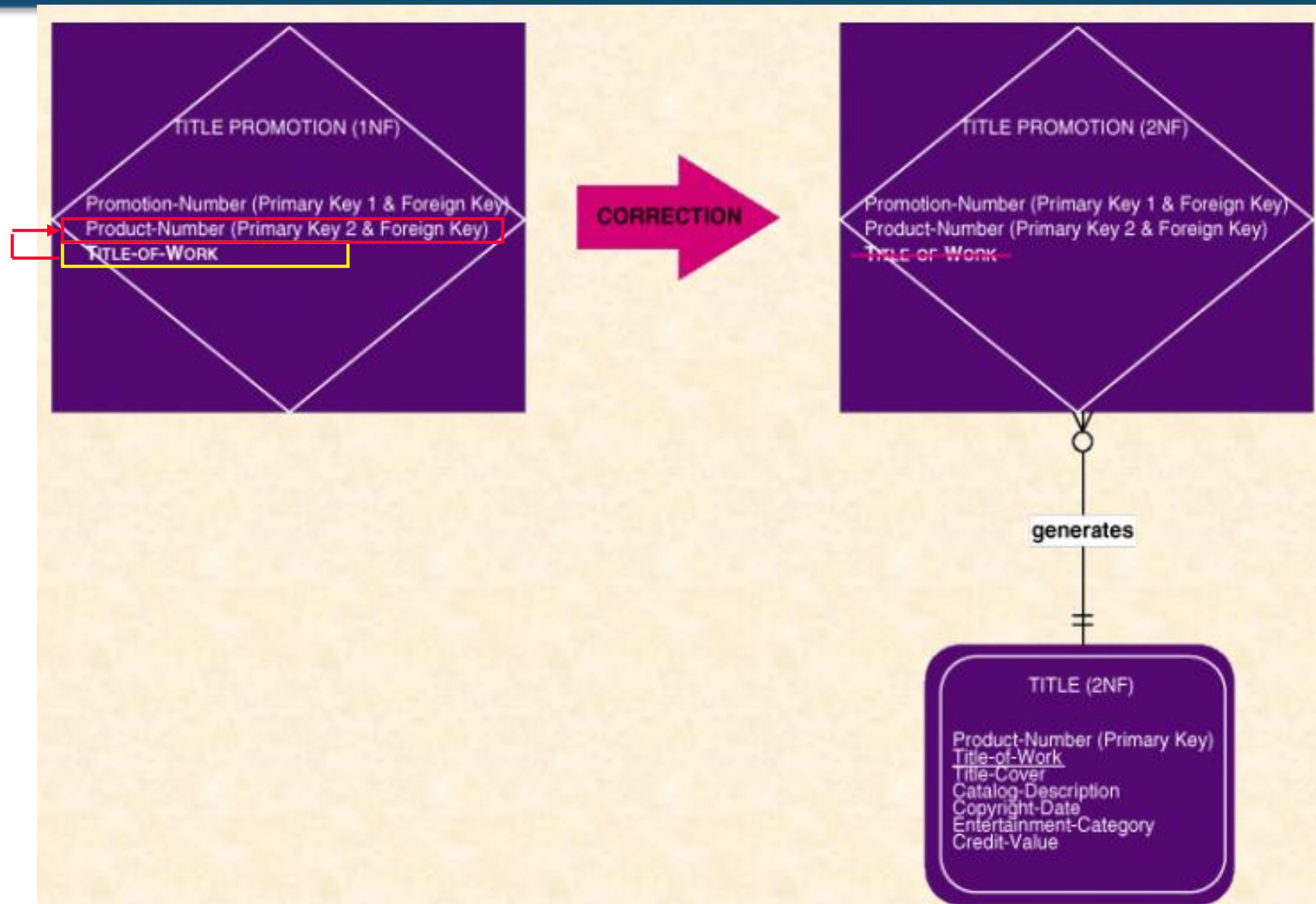
First Normal Form Example



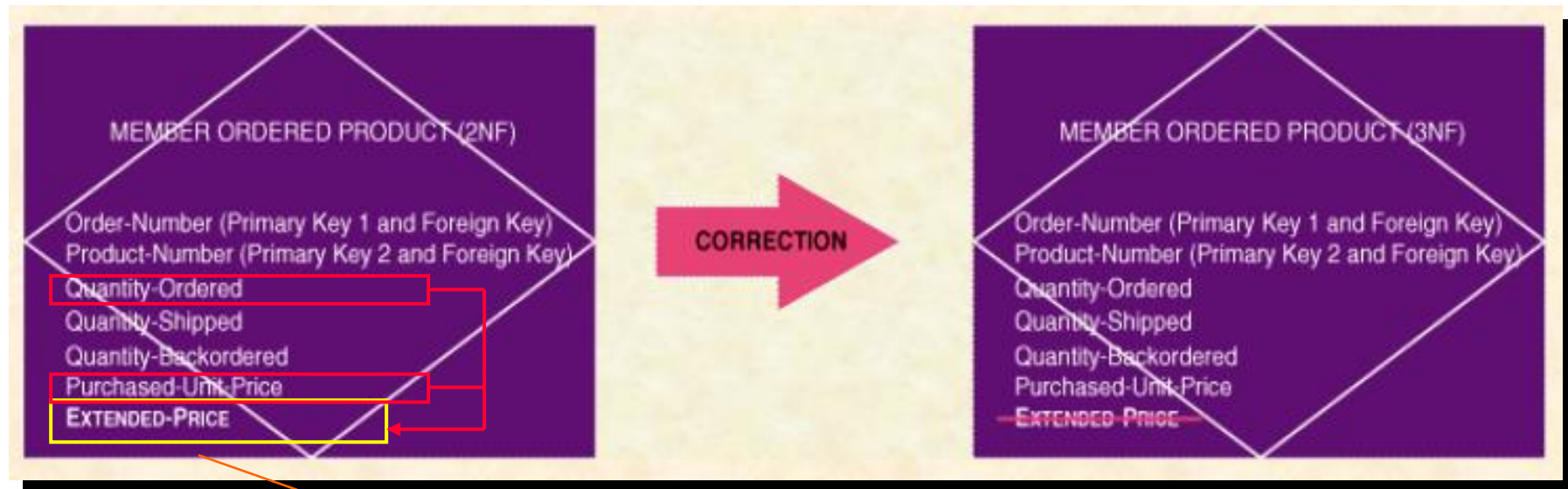
Second Normal Form Example



Second Normal Form Example



Third Normal Form Example

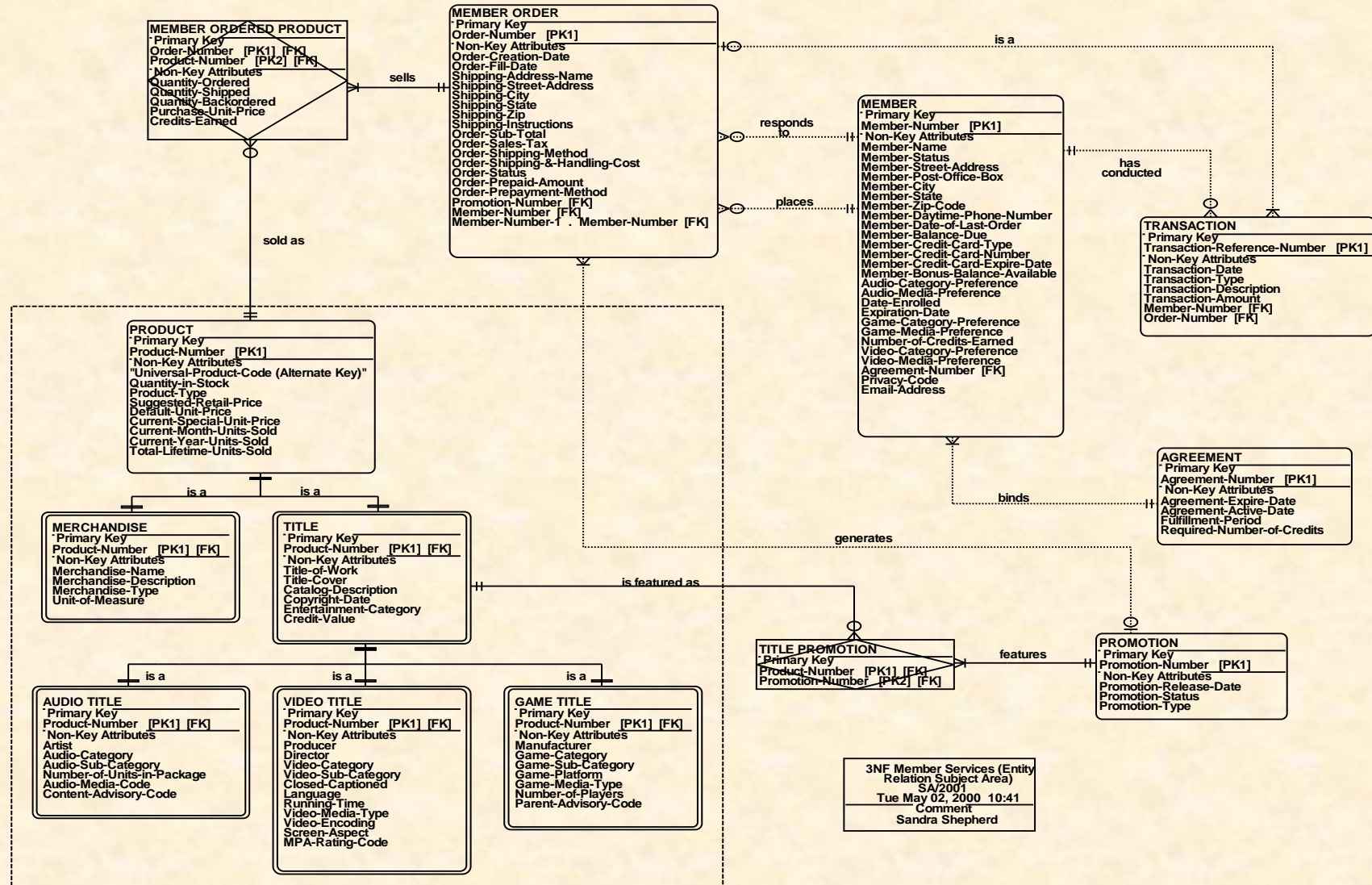


Derived attributes are those whose values can either be calculated from other attributes or derived through logic from the values of other attributes.

Third Normal Form Example



SoundStage 3NF Data Model



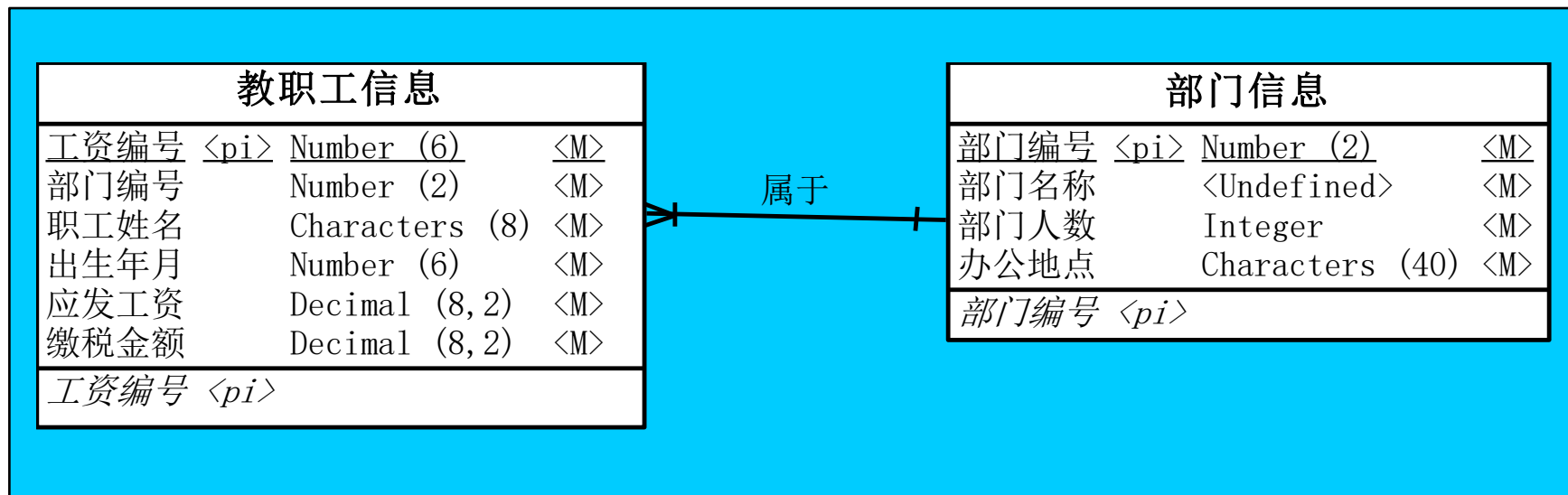
Denormalization

某软件公司成立项目组为某高校开发一套教职工信息管理系统。与教职工信息相关的数据需求和处理需求如下：

(1) 数据需求：在教职工信息中能够存储学校所有在
职的教工和职工信息，包括姓名、所属部门、出生年月、工
资编号、工资额和缴税信息；部门信息中包括部门编号、部
门名称、部门人数和办公地点信息。

(2) 处理需求：能够根据编制内或外聘教职工的工资
编号分别查询其相关信息；每个月的月底统一核发工资，要
求系统能够以最快速度查询出教工或者职工所在部门名称、
实发工资金额；由于学校人员相对稳定，所以数据变化及维
护工作量很少。

Denormalization



Denormalization

编制内教职工信息			
工资编号	<pi> Number (6)		<M>
部门编号	Number (2)		<M>
部门名称	Characters (20)		
职工姓名	Characters (8)		<M>
出生年月	Number (6)		<M>
应发工资	Decimal (8,2)		<M>
缴税金额	Decimal (8,2)		<M>
实发工资	Decimal (8,2)		
工资编号 <pi>			

外聘教职工信息			
工资编号	<pi> Number (6)		<M>
部门编号	Number (2)		<M>
部门名称	Characters (20)		
职工姓名	Characters (8)		<M>
出生年月	Number (6)		<M>
应发工资	Decimal (8,2)		<M>
缴税金额	Decimal (8,2)		<M>
实发工资	Decimal (8,2)		
工资编号 <pi>			

部门信息			
部门编号	<pi> Number (2)		<M>
部门名称	Characters (20)		<M>
部门人数	Integer		<M>
办公地点	Characters (40)		<M>
部门编号 <pi>			

属于

数据分区：将数据按照水平（子集准则）或按照垂直（字段）分到多个结构相同的数据表中

Data Modeling Exercise

快递详情单



寄件人姓名		始发地		收件人姓名		目的地	
单位名称				单位名称			
寄件人详细地址				收件人详细地址			
联系电话 (非常重要)				联系电话 (非常重要)			
手机				手机			
<input type="checkbox"/> 文件		<input type="checkbox"/> 物品		如系物品, 请据实填写内件名称及数量。 如需保价, 请据实申报保价金额并交纳保价费。			
重量	千克	体积	长	x 宽	x 高	=	厘米
内件品名及数量		特别注意: 请阅读背面快递服务协议, 贵重物品请保价, 未保价物品的理赔限额为资费的5倍。					
寄件人签名:		收寄人员签章:		付款方式 <input type="checkbox"/> 现金 <input type="checkbox"/> 到付 <input type="checkbox"/> 协议结算			
证件号		年 月 日 时		<input type="checkbox"/> 保价 保价金额 万 仟 佰 拾 元 (大写)			
				代收货款 ¥ 万 仟 佰 拾 元 (大写)			
				运费 ¥ 加急费 ¥ 包装费 ¥ 保价费 ¥ 总计 ¥			
				收件人签名:			
				证件号 年 月 日 时			
				备注:			

我们不是无所不能,
但一定竭尽所能!

寄件人	
姓名	Variable characters (20)
始发地	Variable characters (20)
单位名称	Text
详细地址	Text
联系电话	Characters (12)
证件号	Characters (20)
主属性: PK1 <pi>	

收件人	
姓名	Variable characters (20)
目的地	Variable characters (20)
单位名称	Text
详细地址	Text
联系电话	Characters (12)
证件号	Characters (20)
主属性: PK3 <pi>	

快递单	
编号	Characters (10)
类型	Short integer
重量	Decimal (4,2)
体积	Decimal (4,2)
名称	Variable characters (20)
数量	Integer
收寄员	Characters (20)
日期	Date & Time
付款方式	Short integer
保价金额	Money
代收货款	Money
运费	Money
加急费	Money
包装费	Money
保价费	Money
总计	Money
备注	Variable characters (40)
主属性: PK2 <pi>	

要点与引申

- ❁ 数据建模的过程，是识别问题空间中的对象、对象之间的关系、对象的标志性特征的过程。
- ❁ 在系统分析阶段，一般不可能识别出每一种实体的所有属性。但是，如果漏掉的是实体的标志性属性（它/它们所涉及的是如何确定该实体的实例，或者是如何体现这种实体与其他实体之间的联系），那么发现得越迟，付出的代价就越大。
- ❁ 注意：本章所述的数据建模方法，虽然一再声称与物理实现无关，但实际上已经约定的是，将来的物理实现是采用关系数据库。