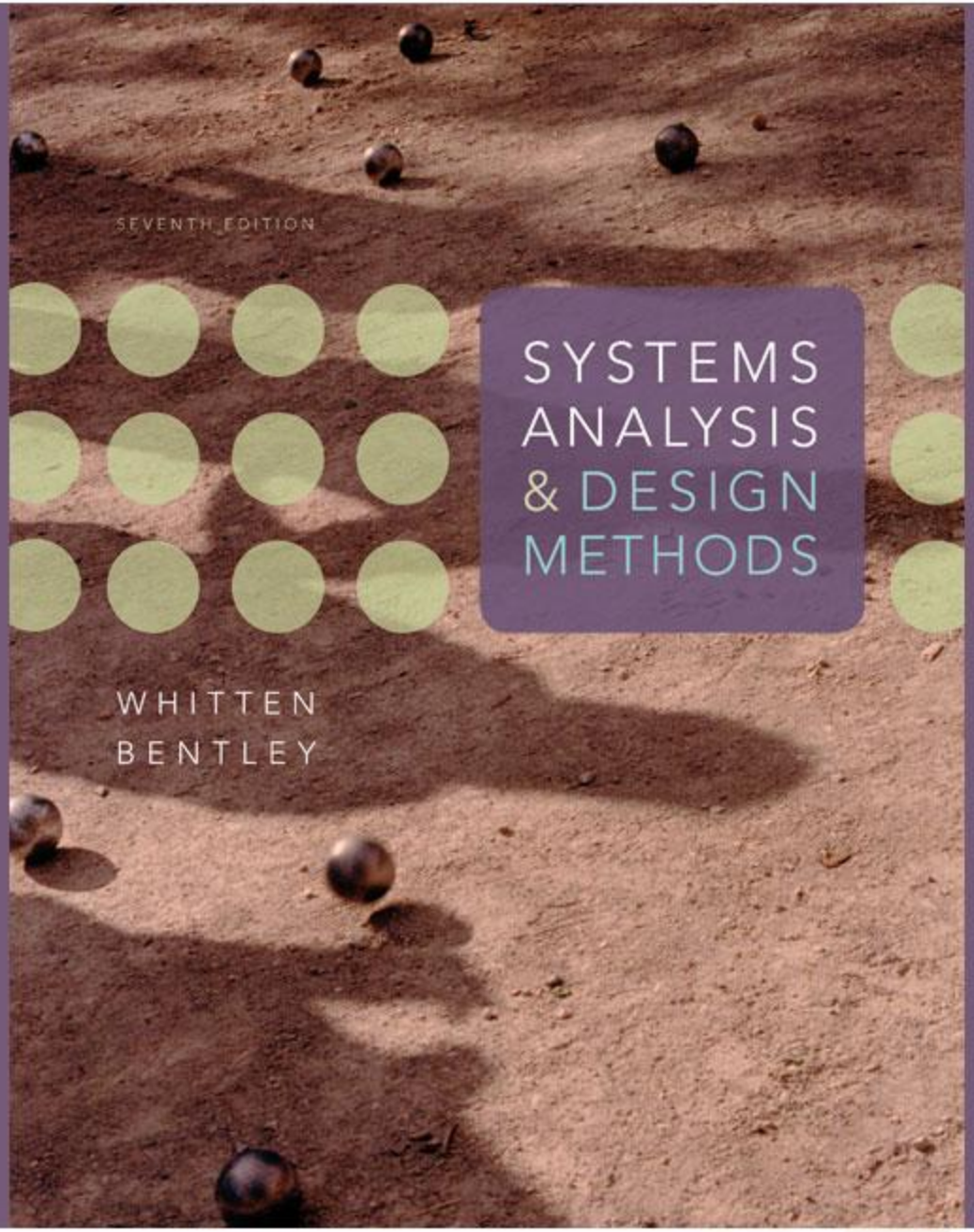
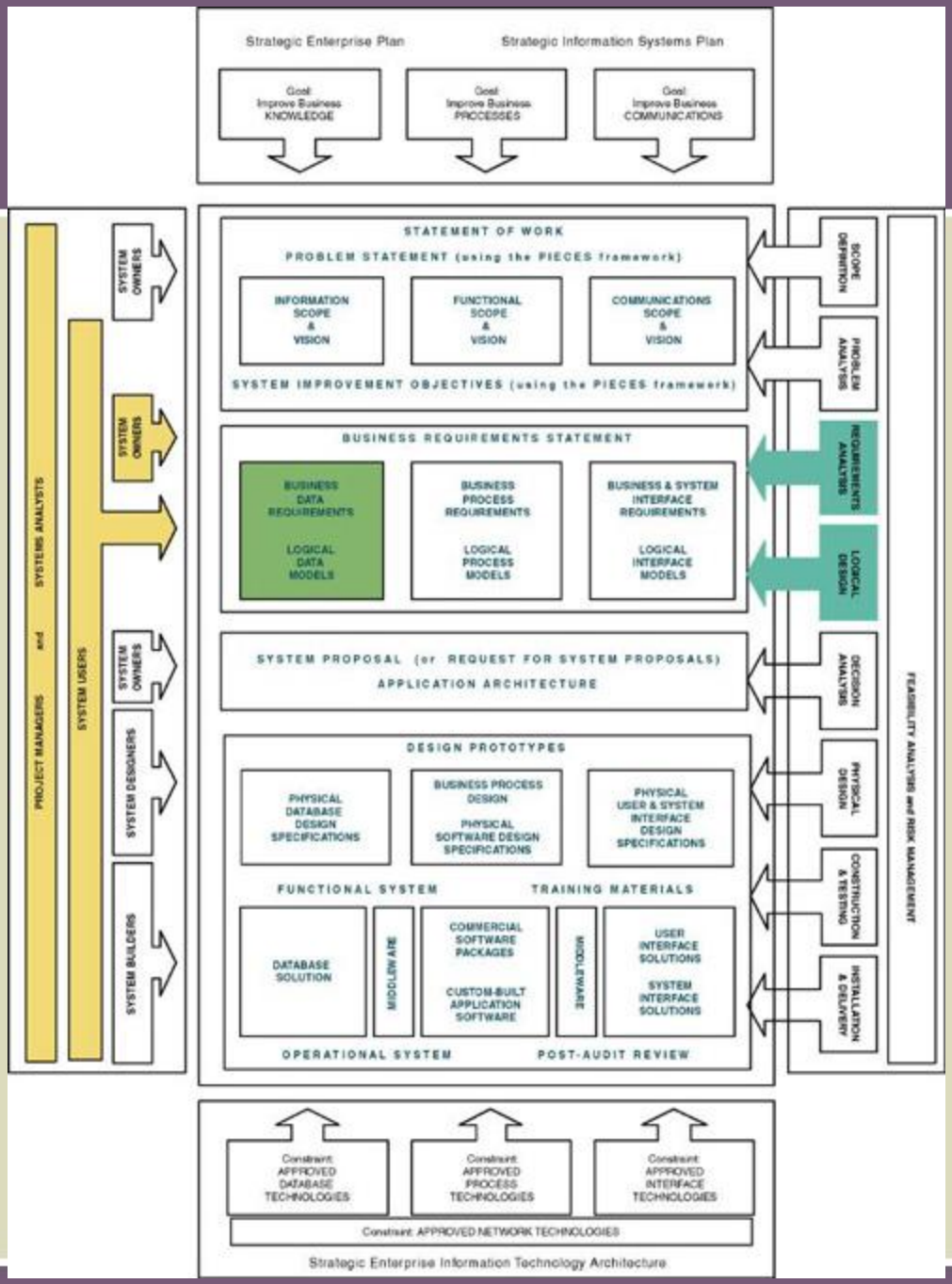

SEVENTH EDITION
SYSTEMS ANALYSIS & DESIGN METHODS
WHITTEN
BENTLEY

Objectives

- Define data modeling and explain its benefits.
- Recognize and understand the basic concepts and constructs of a data model.
- Read and interpret an entity relationship data model.
- Explain when data models are constructed during a project and where the models are stored.
- Discover entities and relationships.
- Construct an entity-relationship context diagram.
- Discover or invent keys for entities and construct a key-based diagram.
- Construct a fully attributed entity relationship diagram and describe data structures and attributes to the repository.
- Normalize a logical data model to remove impurities that can make a database unstable, inflexible, and nonscalable.
- Describe a useful tool for mapping data requirements to business operating locations.

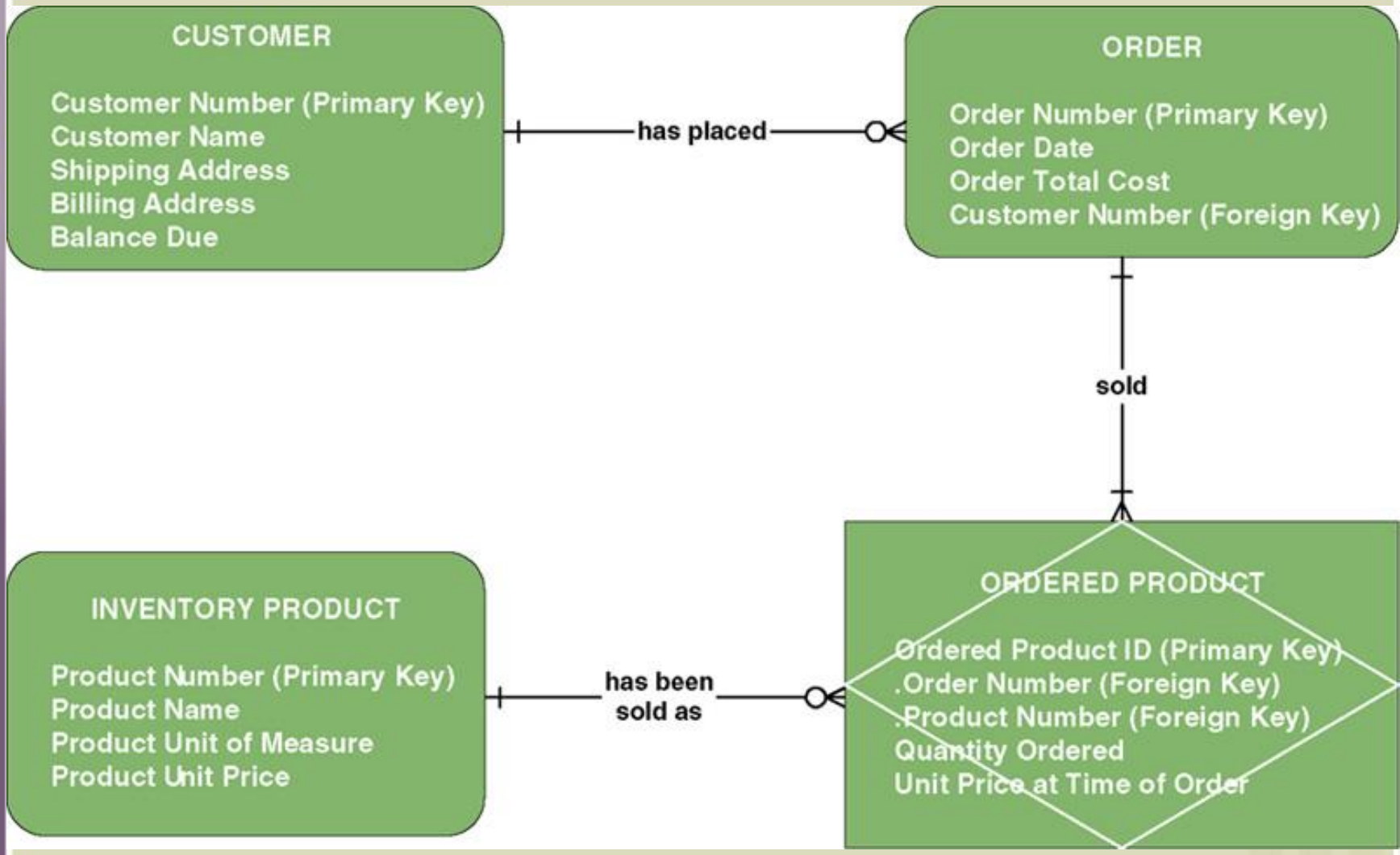

Strategic Enterprise Plan
Strategic Information Systems Plan
Goal: Improve Business KNOWLEDGE
Goal: Improve Business PROCESSES
Goal: Improve Business COMMUNICATIONS
PROJECT MANAGERS and SYSTEMS ANALYSTS
SYSTEM USERS
SYSTEM OWNERS
SYSTEM OWNERS
SYSTEM OWNERS
SYSTEM DESIGNERS
SYSTEM BUILDERS
STATEMENT OF WORK
PROBLEM STATEMENT (using the PIECES framework)
INFORMATION SCOPE & VISION
FUNCTIONAL SCOPE & VISION
COMMUNICATIONS SCOPE & VISION
SYSTEM IMPROVEMENT OBJECTIVES (using the PIECES framework)
BUSINESS REQUIREMENTS STATEMENT
BUSINESS DATA REQUIREMENTS
LOGICAL DATA MODELS
BUSINESS PROCESS REQUIREMENTS
LOGICAL PROCESS MODELS
BUSINESS & SYSTEM INTERFACE REQUIREMENTS
LOGICAL INTERFACE MODELS
SYSTEM PROPOSAL (or REQUEST FOR SYSTEM PROPOSALS)
APPLICATION ARCHITECTURE
DESIGN PROTOTYPES
PHYSICAL DATABASE DESIGN SPECIFICATIONS
BUSINESS PROCESS DESIGN
PHYSICAL SOFTWARE DESIGN SPECIFICATIONS
PHYSICAL USER & SYSTEM INTERFACE DESIGN SPECIFICATIONS
FUNCTIONAL SYSTEM
TRAINING MATERIALS
DATABASE SOLUTION
MIDDLEWARE
COMMERCIAL SOFTWARE PACKAGES
CUSTOM-BUILT APPLICATION SOFTWARE
MIDDLEWARE
USER INTERFACE SOLUTIONS
SYSTEM INTERFACE SOLUTIONS
OPERATIONAL SYSTEM
POST-AUDIT REVIEW
SCOPE DEFINITION
PROBLEM ANALYSIS
REQUIREMENTS ANALYSIS
LOGICAL DESIGN
DECISION ANALYSIS
PHYSICAL DESIGN
CONSTRUCTION & TESTING
INSTALLATION & DELIVERY
FEASIBILITY ANALYSIS and RISK MANAGEMENT
Constraint: APPROVED DATABASE TECHNOLOGIES
Constraint: APPROVED PROCESS TECHNOLOGIES
Constraint: APPROVED INTERFACE TECHNOLOGIES
Constraint: APPROVED NETWORK TECHNOLOGIES
Strategic Enterprise Information Technology Architecture

Data Modeling

Data modeling – a technique for organizing and documenting a system's data. Sometimes called *database modeling*.

Entity relationship diagram (ERD) – a data model utilizing several notations to depict data in terms of the entities and relationships described by that data.

Sample Entity Relationship Diagram (ERD)



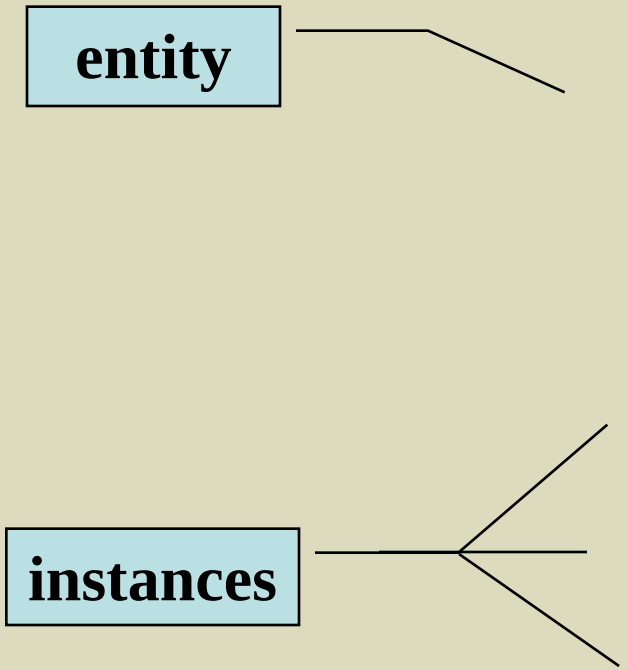
Data Modeling Concepts: Entity

Entity – a class of persons, places, objects, events, or concepts about which we need to capture and store data.

- Named by a singular noun
  - 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.

Data Modeling Concepts: Entity

Entity instance – a single occurrence of an entity.



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

Data Modeling Concepts: Attributes

Attribute – a descriptive property or characteristic of an entity. Synonyms include *element*, *property*, and *field*.

- Just as a physical student can have attributes, such as hair color, height, etc., data entity has data attributes

Compound attribute – an attribute that consists of other attributes. Synonyms in different data modeling languages are numerous: concatenated attribute, composite attribute, and data structure.

Data Modeling Concepts: Data Type

Data type – a property of an attribute that identifies what type of data can be stored in that attribute.

Representative Logical Data Types for Attributes	
Data Type	**Logical Business Meaning**
NUMBER	Any number, real or integer.
TEXT	A string of characters, inclusive of numbers. When numbers are included in a TEXT attribute, it means that we do not expect to perform arithmetic or comparisons with those numbers.
MEMO	Same as TEXT but of an indeterminate size. Some business systems require the ability to attach potentially lengthy notes to a give database record.
DATE	Any date in any format.
TIME	Any time in any format.
YES/NO	An attribute that can assume only one of these two values.
VALUE SET	A finite set of values. In most cases, a coding scheme would be established (e.g., FR=Freshman, SO=Sophomore, JR=Junior, SR=Senior).
IMAGE	Any picture or image.

Data Modeling Concepts: Domains

Domain – a property of an attribute that defines what values an attribute can legitimately take on.

Representative Logical Domains for Logical Data Types		
Data Type	**Domain**	**Examples**
NUMBER	For integers, specify the range. For real numbers, specify the range and precision.	{10-99} {1.000-799.999}
TEXT	Maximum size of attribute. Actual values usually infinite; however, users may specify certain narrative restrictions.	Text(30)
DATE	Variation on the MMDDYYYY format.	MMDDYYYY MMYYYY
TIME	For AM/PM times: HHMMT For military (24-hour times): HHMM	HHMMT HHMM
YES/NO	{YES, NO}	{YES, NO} {ON, OFF}
VALUE SET	{value#1, value#2,…value#n} {table of codes and meanings}	{M=Male F=Female}

Data Modeling Concepts: Default Value

Default value – the value that will be recorded if a value is not specified by the user.

Permissible Default Values for Attributes		
Default Value	**Interpretation**	**Examples**
A legal value from the domain	For an instance of the attribute, if the user does not specify a value, then use this value.	0 1.00
NONE or NULL	For an instance of the attribute, if the user does not specify a value, then leave it blank.	NONE NULL
Required or NOT NULL	For an instance of the attribute, require that the user enter a legal value from the domain. (This is used when no value in the domain is common enough to be a default but some value must be entered.)	REQUIRED NOT NULL

Data Modeling Concepts: Identification

Key – an attribute, or a group of attributes, that assumes a unique value for each entity instance. It is sometimes called an *identifier*.

- **Concatenated key** - group of attributes that uniquely identifies an instance. Synonyms: composite key, compound key.
- **Candidate key** – one of a number of keys that may serve as the primary key. Synonym: *candidate identifier*.
- **Primary key** – a candidate key used to uniquely identify a single entity instance.
- **Alternate key** – a candidate key not selected to become the primary key. Synonym: secondary key.

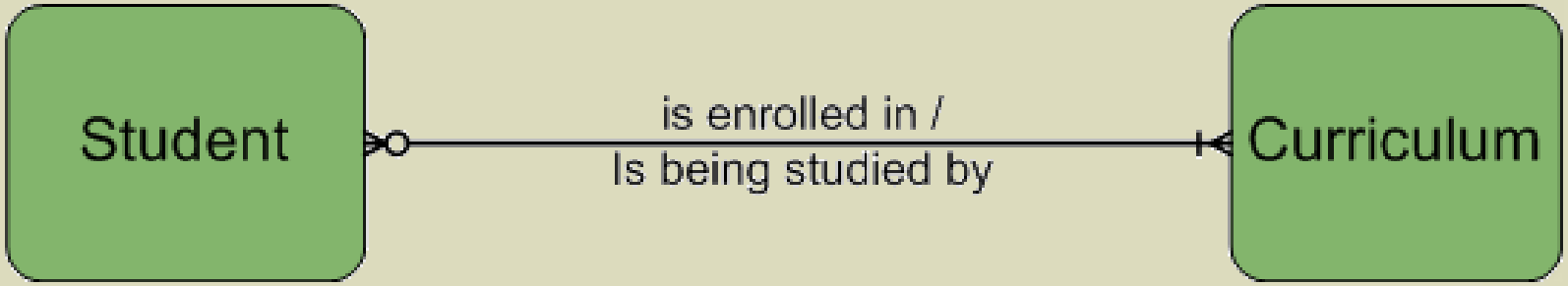
Data Modeling Concepts: Subsetting Criteria

Subsetting criteria – an attribute(s) whose finite values divide all entity instances into useful subsets. Sometimes called an inversion entry.

Data Modeling Concepts: Relationships

Relationship – a natural business association that exists between one or more entities.

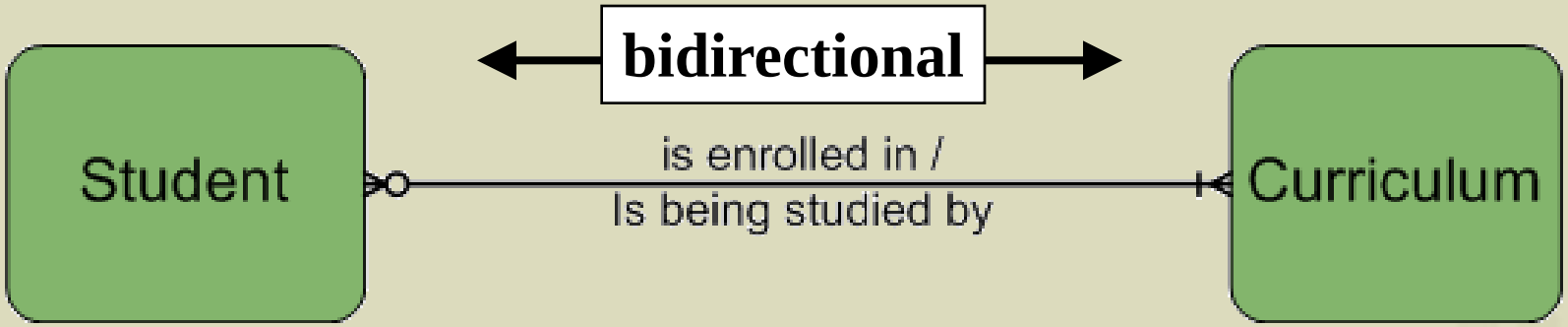
The relationship may represent an event that links the entities or merely a logical affinity that exists between the entities.



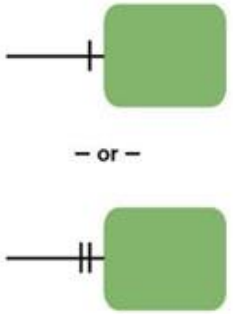
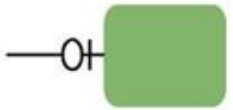
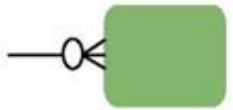
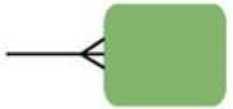
Data Modeling Concepts: Cardinality

Cardinality – 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.



Cardinality Notations

CARDINALITY INTERPRETATION	MINIMUM INSTANCES	MAXIMUM INSTANCES	GRAPHIC NOTATION
Exactly one (one and only one)	1	1	
Zero or one	0	1	
One or more	1	many (>1)	
Zero, one, or more	0	many (>1)	
More than one	>1	>1	

Data Modeling Concepts: Degree

Degree – the number of entities that participate in the relationship.

A relationship between two entities is called a *binary relationship*.

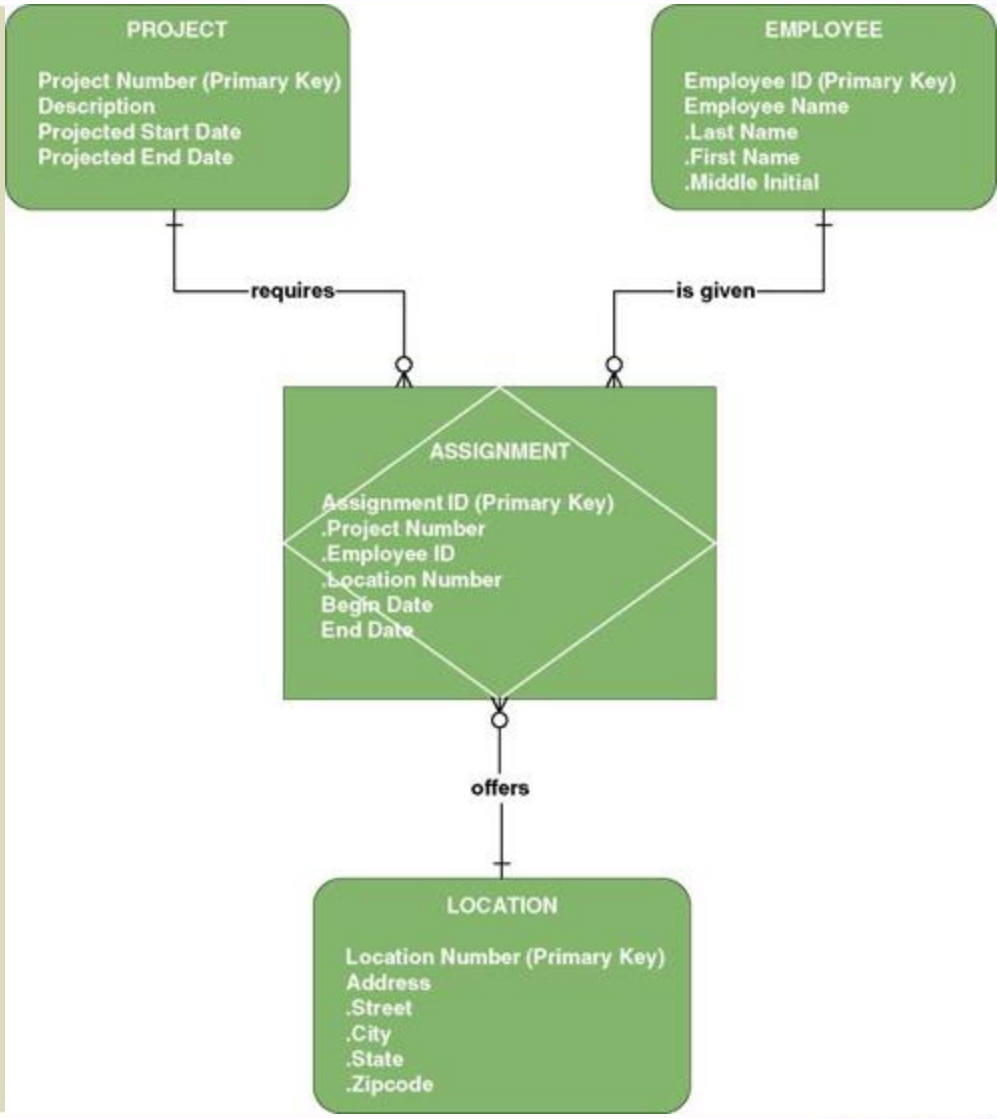
A relationship between three entities is called a *3-ary* or *ternary relationship*.

A relationship between different instances of the same entity is called a *recursive relationship*.

Data Modeling Concepts: Degree

Relationships may exist between more than two entities and are called *N-ary* relationships.

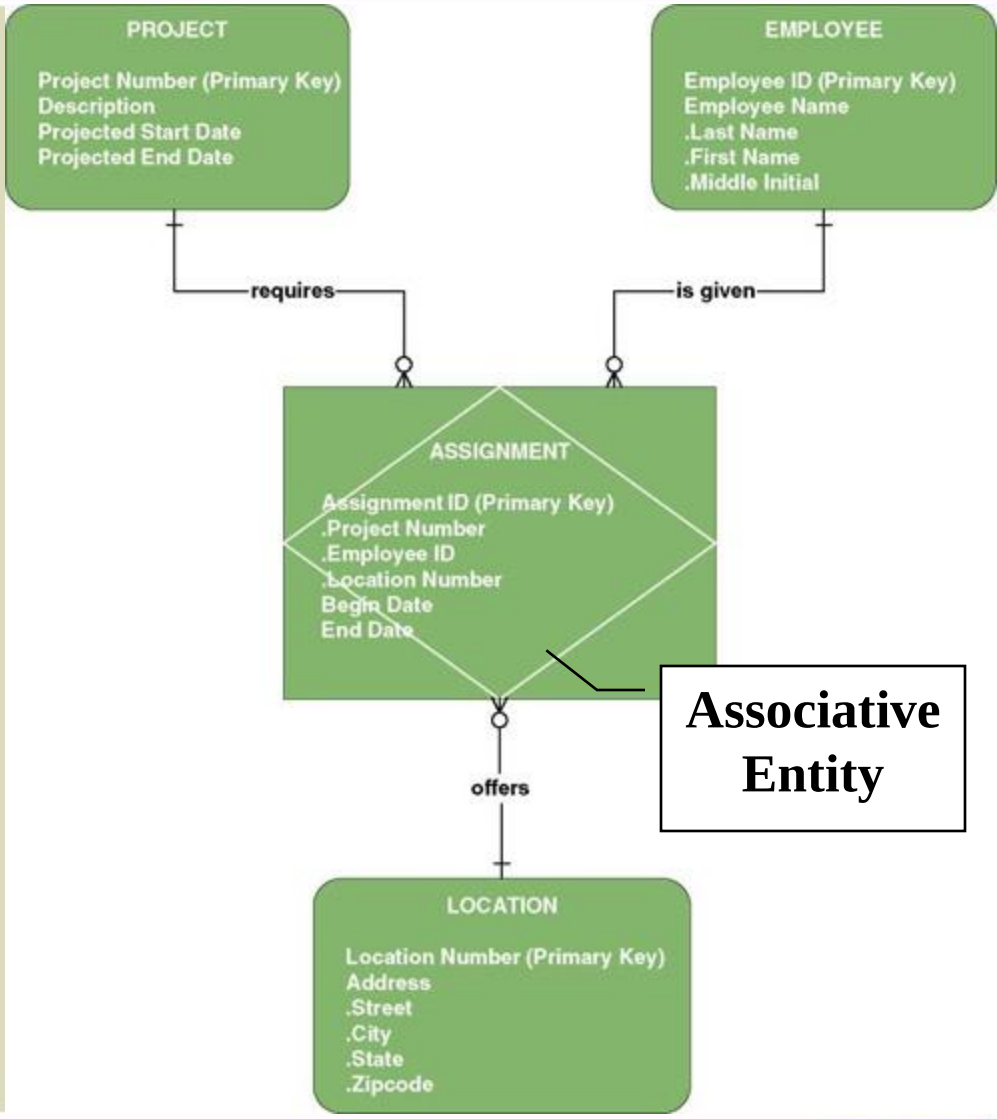
The example ERD depicts a *ternary relationship*.



Data Modeling Concepts: Degree

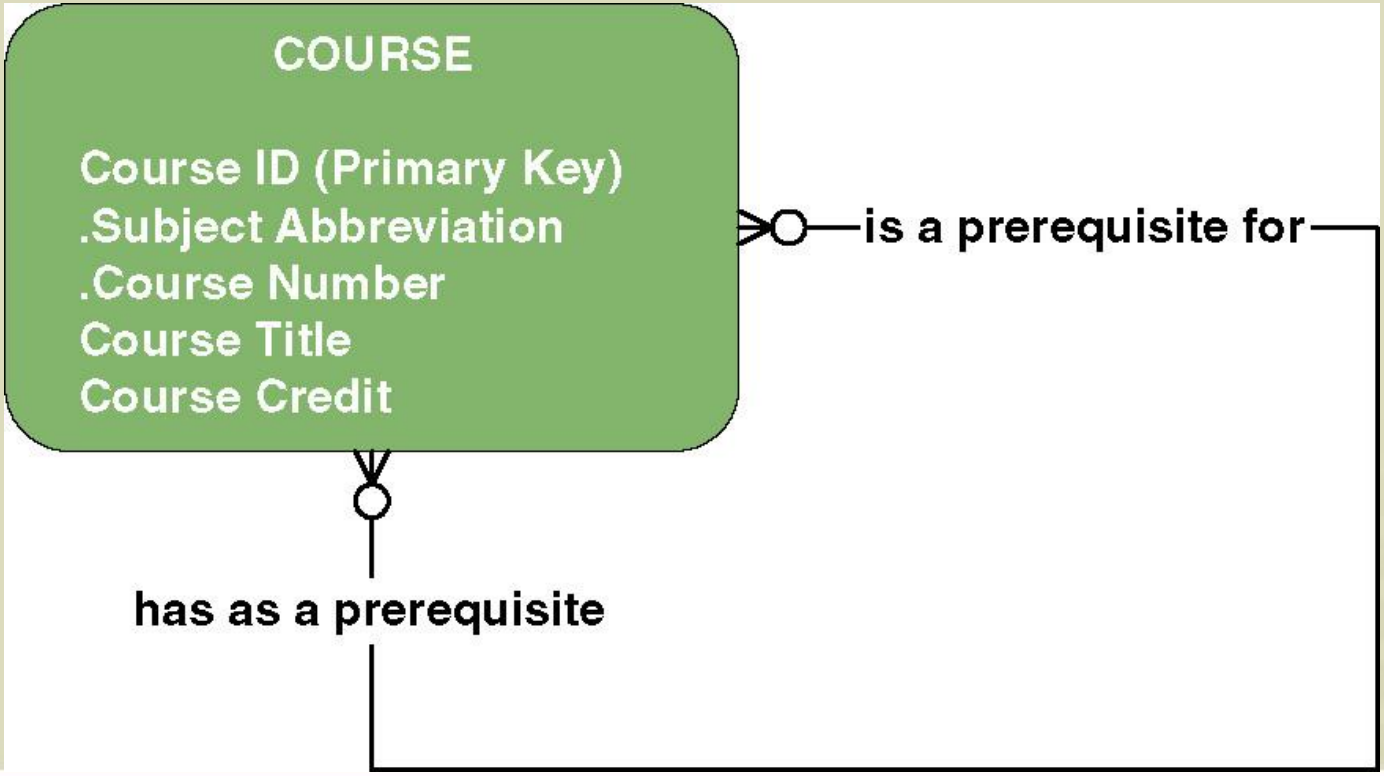
Associative entity – an entity that inherits its primary key from more than one other entity (called parents).

Each part of that concatenated key points to one and only one instance of each of the connecting entities.



Data Modeling Concepts: Recursive Relationship

Recursive relationship - a relationship that exists between instances of the same entity



Data Modeling Concepts: Foreign Keys

Foreign key – a primary key of an entity that is used in another entity to identify instances of a relationship.

- A foreign key is a primary key of one entity that is contributed to (duplicated in) another entity to identify instances of a relationship.
- A foreign key always matches the primary key in the another entity
- A foreign key may or may not be unique (generally not)
- The entity with the foreign key is called the child.
- The entity with the matching primary key is called the parent.

Data Modeling Concepts: Parent and Child Entities

Parent entity - a data entity that contributes one or more attributes to another entity, called the child. In a one-to-many relationship the parent is the entity on the "one" side.

Child entity - a data entity that derives one or more attributes from another entity, called the parent. In a one-to-many relationship the child is the entity on the "many" side.

Data Modeling Concepts: Foreign Keys

Primary Key

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

Foreign Key Duplicated from primary key of Dorm entity (not unique in Student entity)

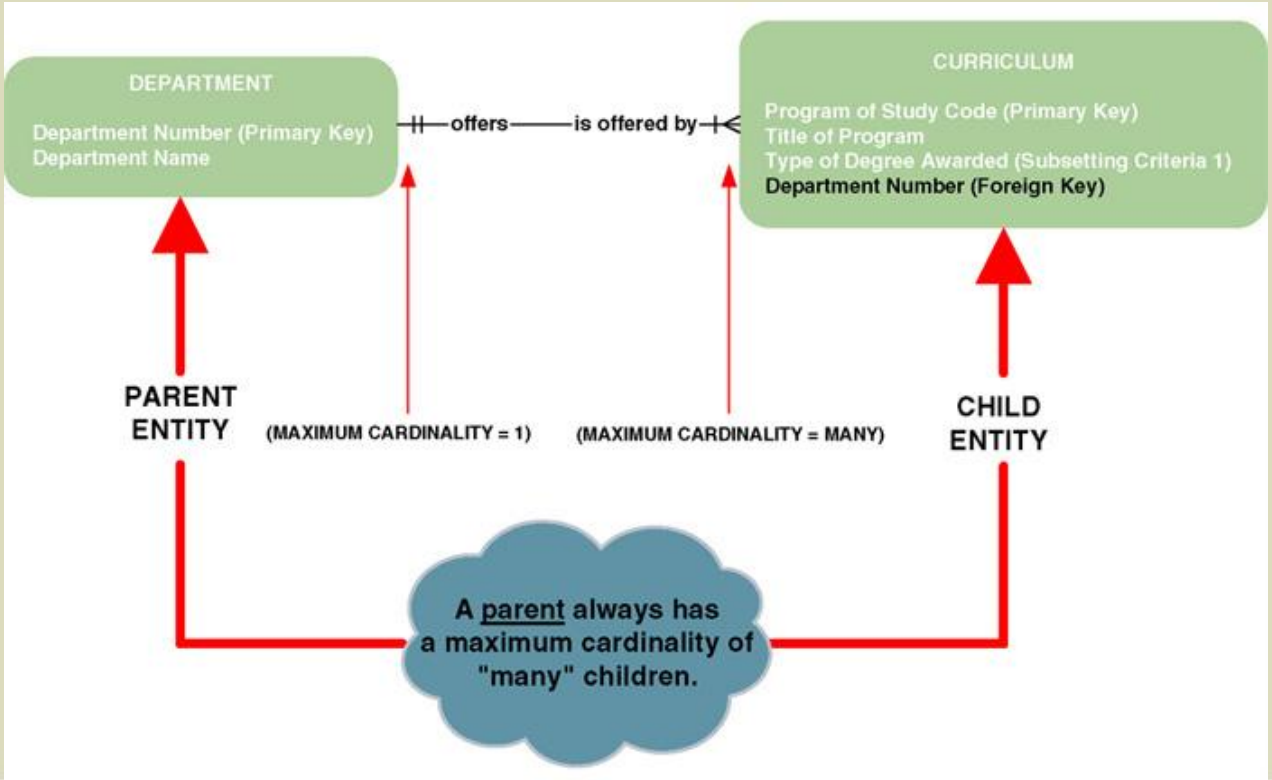
Primary Key

Dorm	Residence Director
Smith	Andrea Fernandez
Jones	Daniel Abidjan

Data Modeling Concepts: Nonidentifying Relationships

Nonidentifying relationship – relationship where each participating entity has its own independent primary key

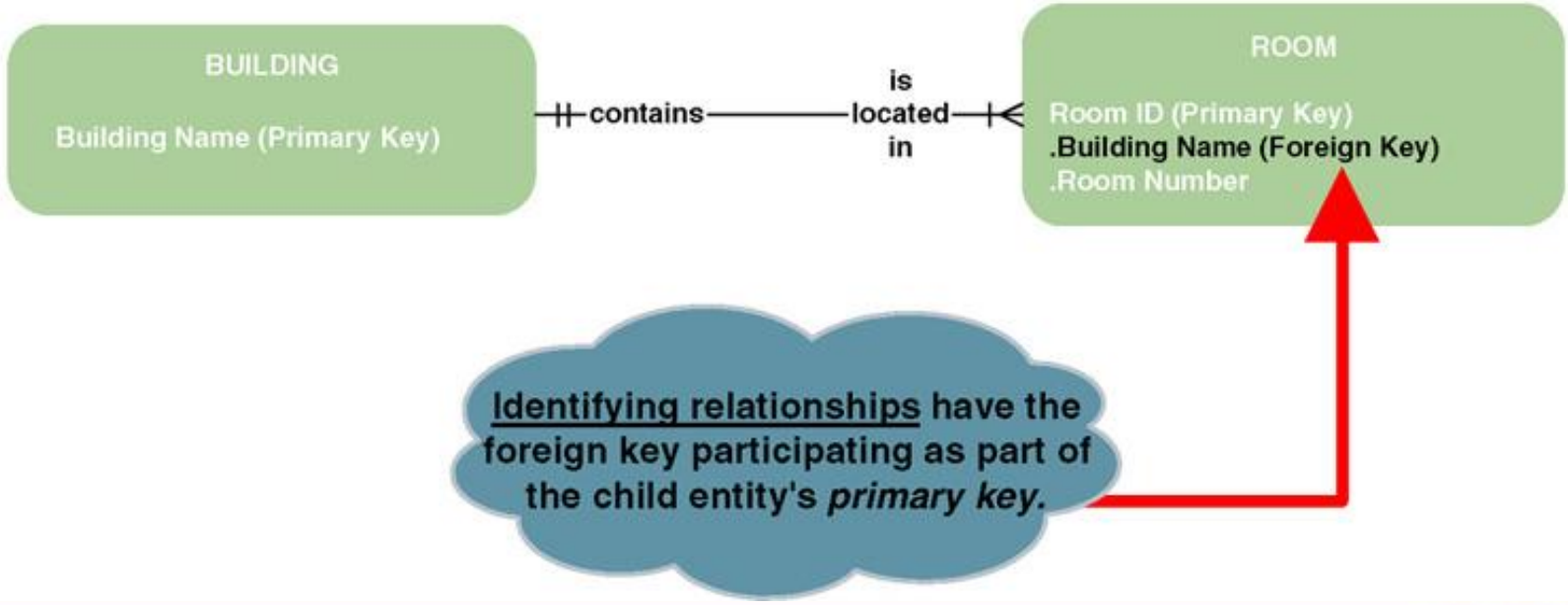
- Primary key attributes are not shared.
- The entities are called *strong* entities



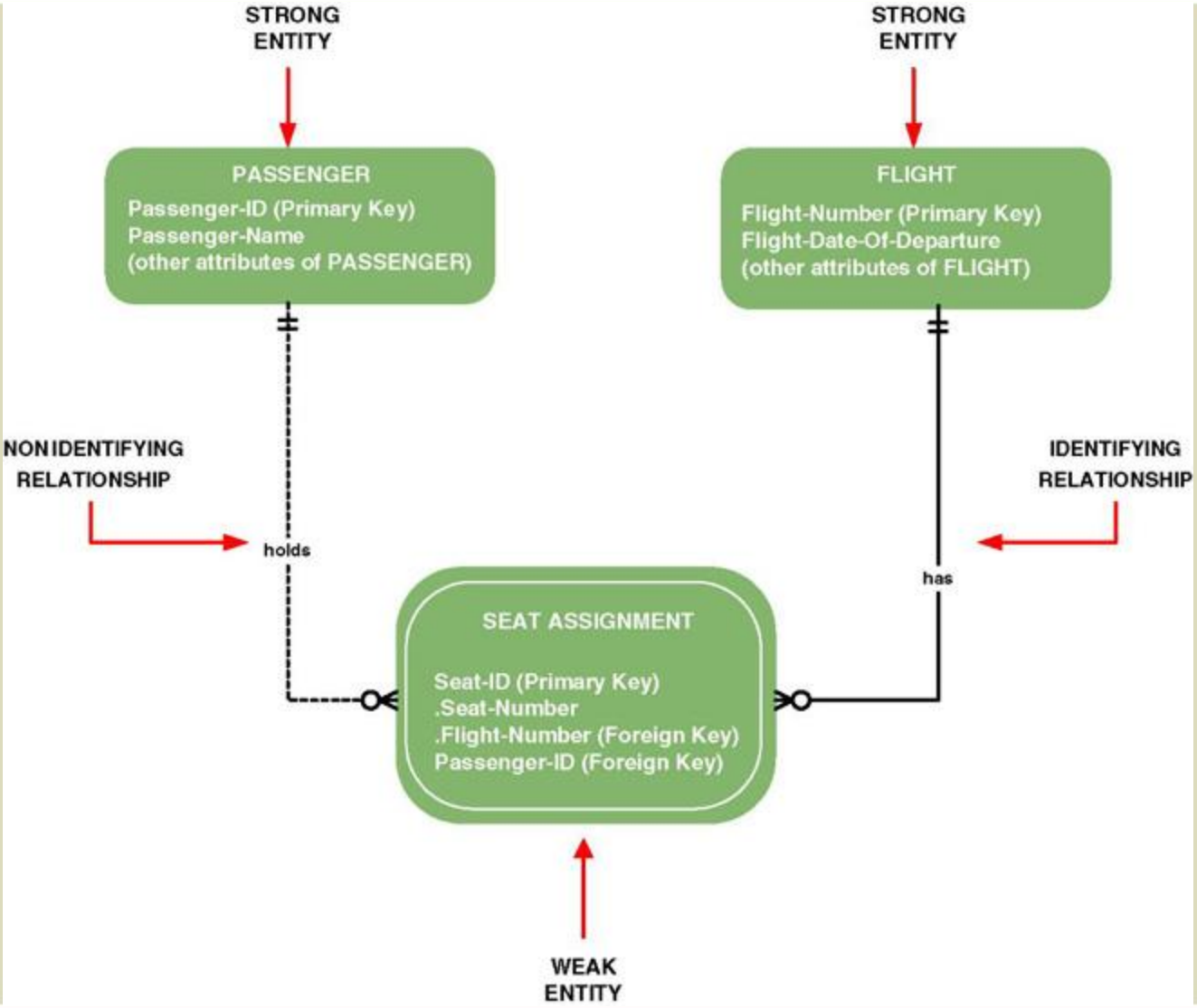
Data Modeling Concepts: Identifying Relationships

Identifying relationship – relationship in which the parent entity' key is also part of the primary key of the child entity.

- The child entity is called a *weak* entity.



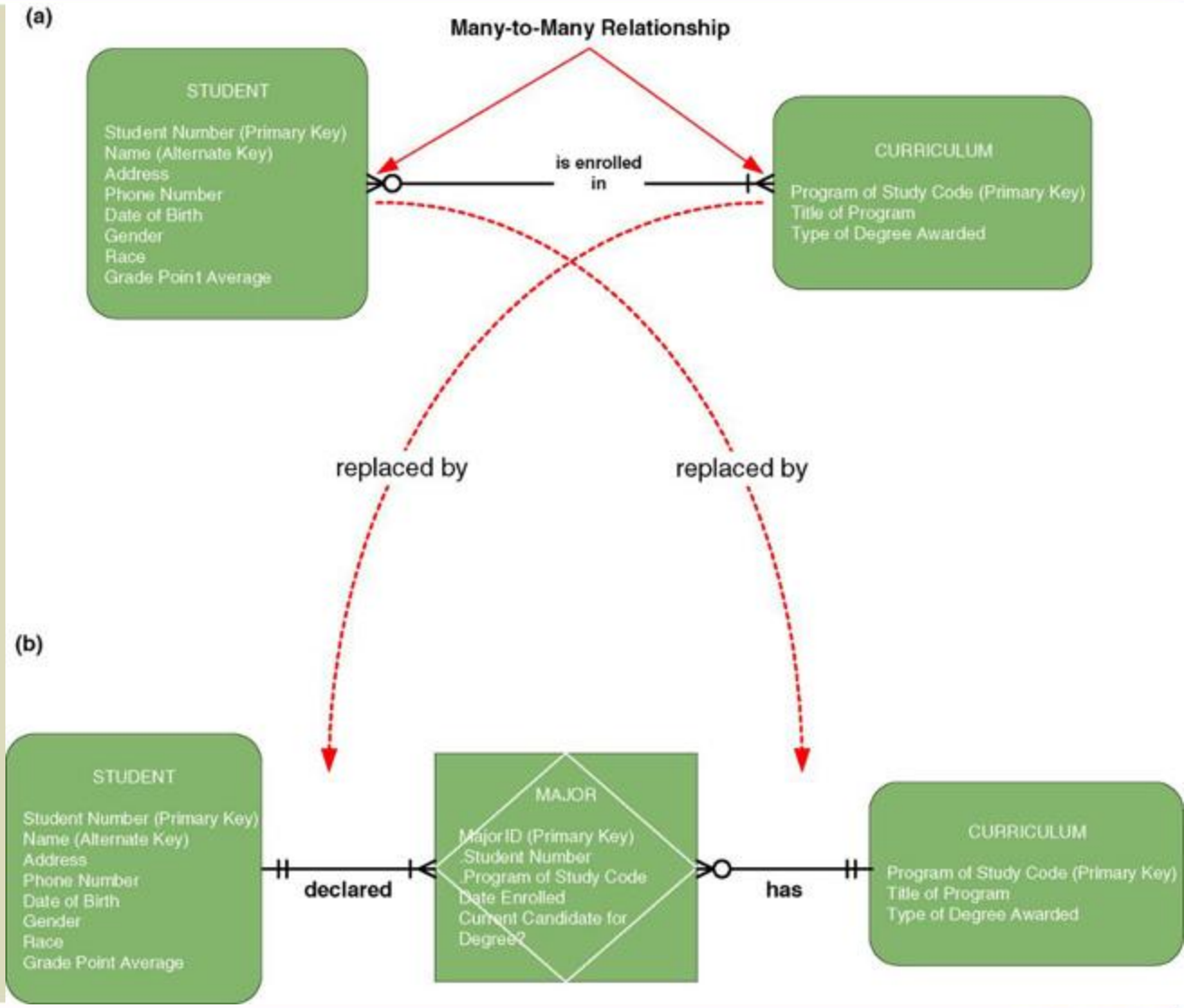
Data Modeling Concepts: Sample CASE Tool Notations



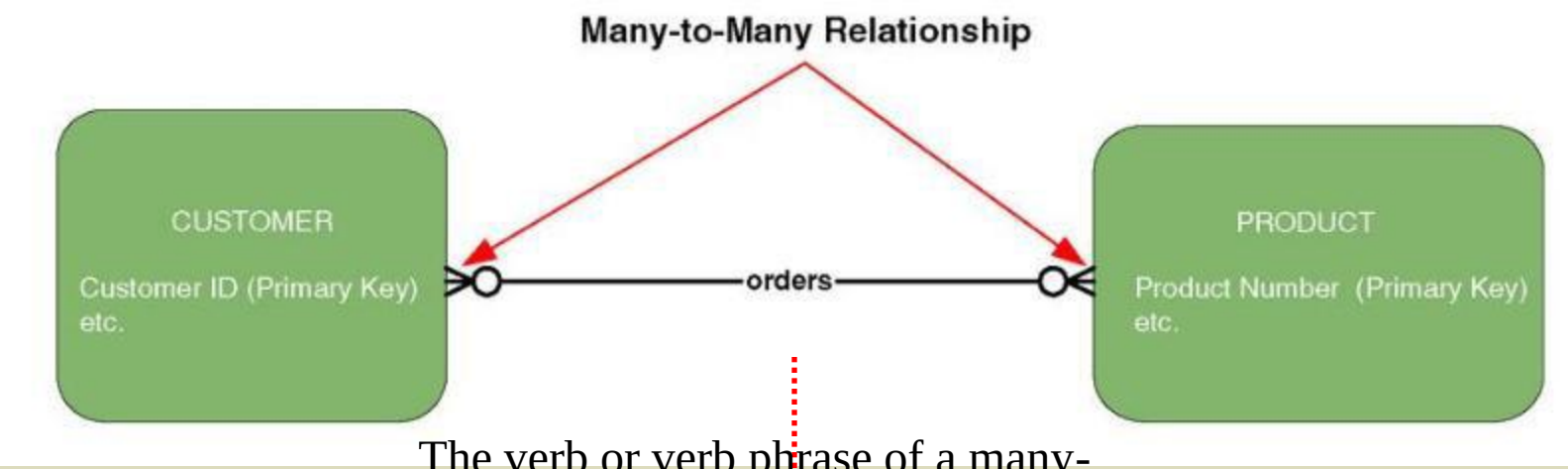
Data Modeling Concepts: Nonspecific Relationships

Nonspecific relationship – relationship where many instances of an entity are associated with many instances of another entity. Also called *many-to-many relationship*.

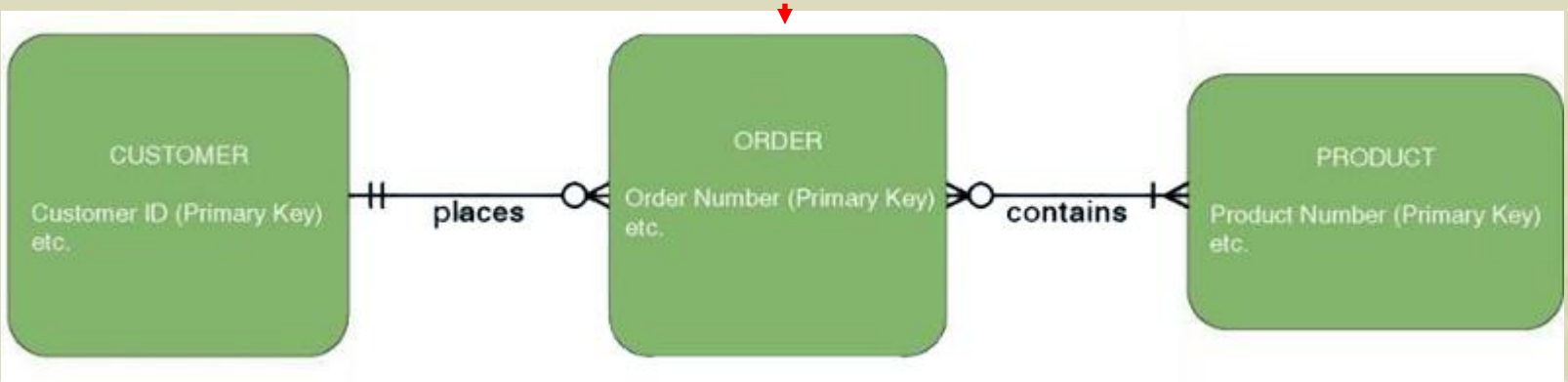
Nonspecific relationships must be resolved, generally by introducing an associative entity.



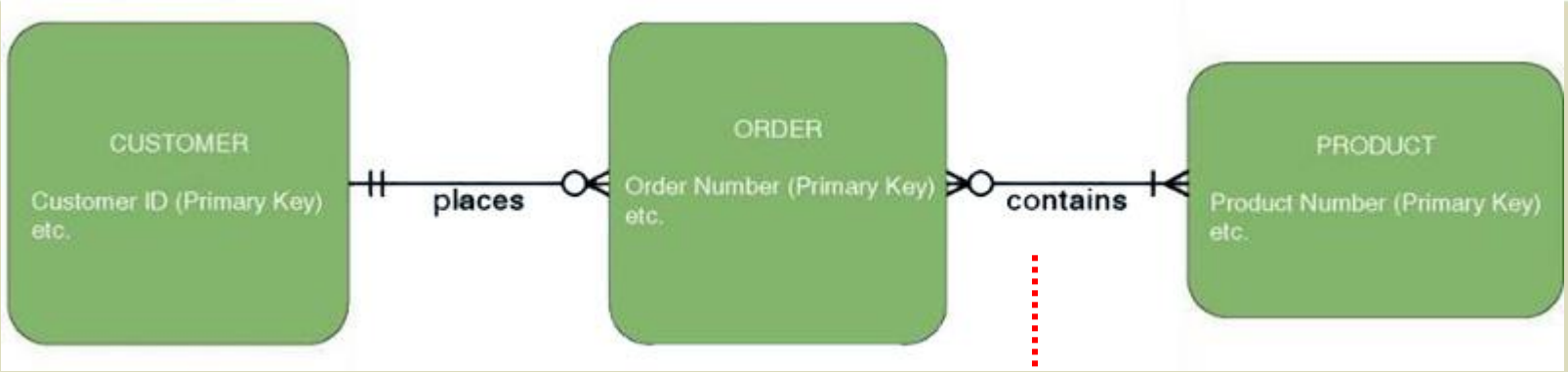
Resolving Nonspecific Relationships



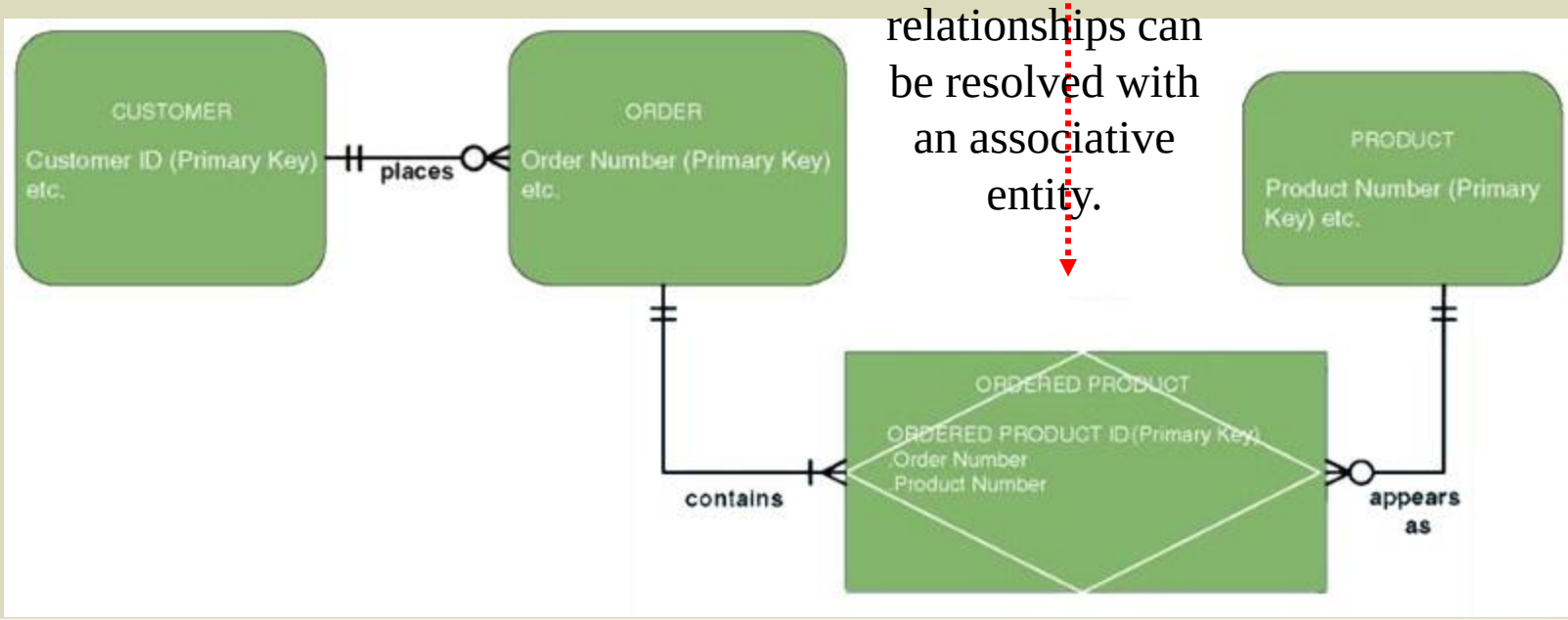
The verb or verb phrase of a many-to-many relationship sometimes suggests other entities.



Resolving Nonspecific Relationships (continued)

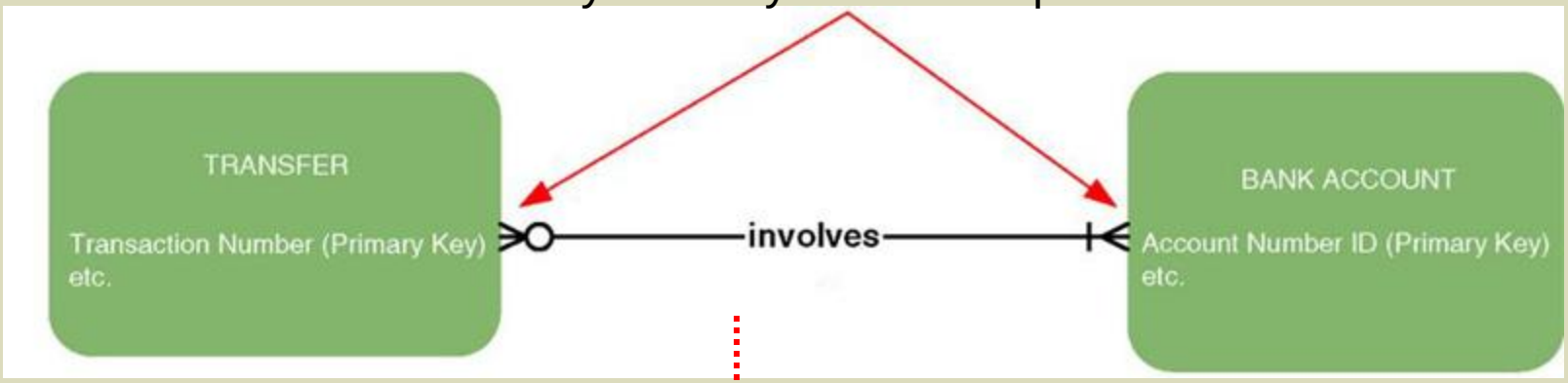


Many-to-many relationships can be resolved with an associative entity.

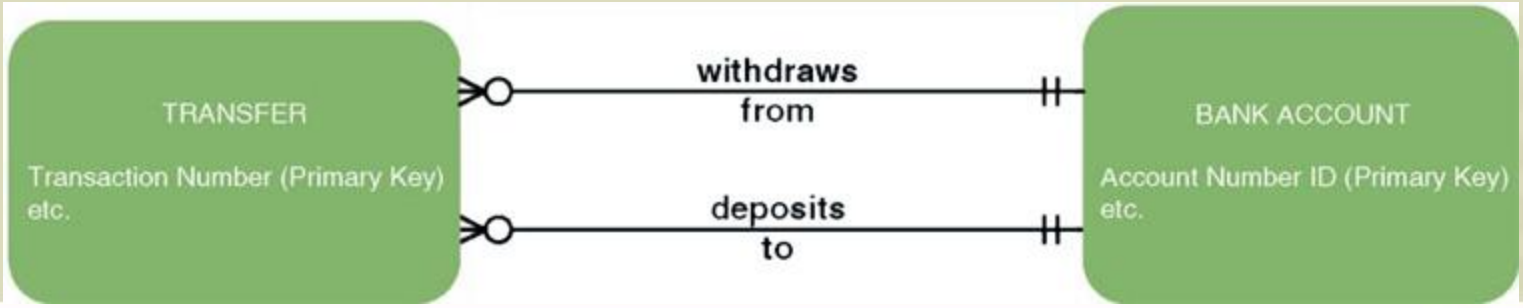


Resolving Nonspecific Relationships (continued)

Many-to-Many Relationship



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.



Data Modeling Concepts: Generalization

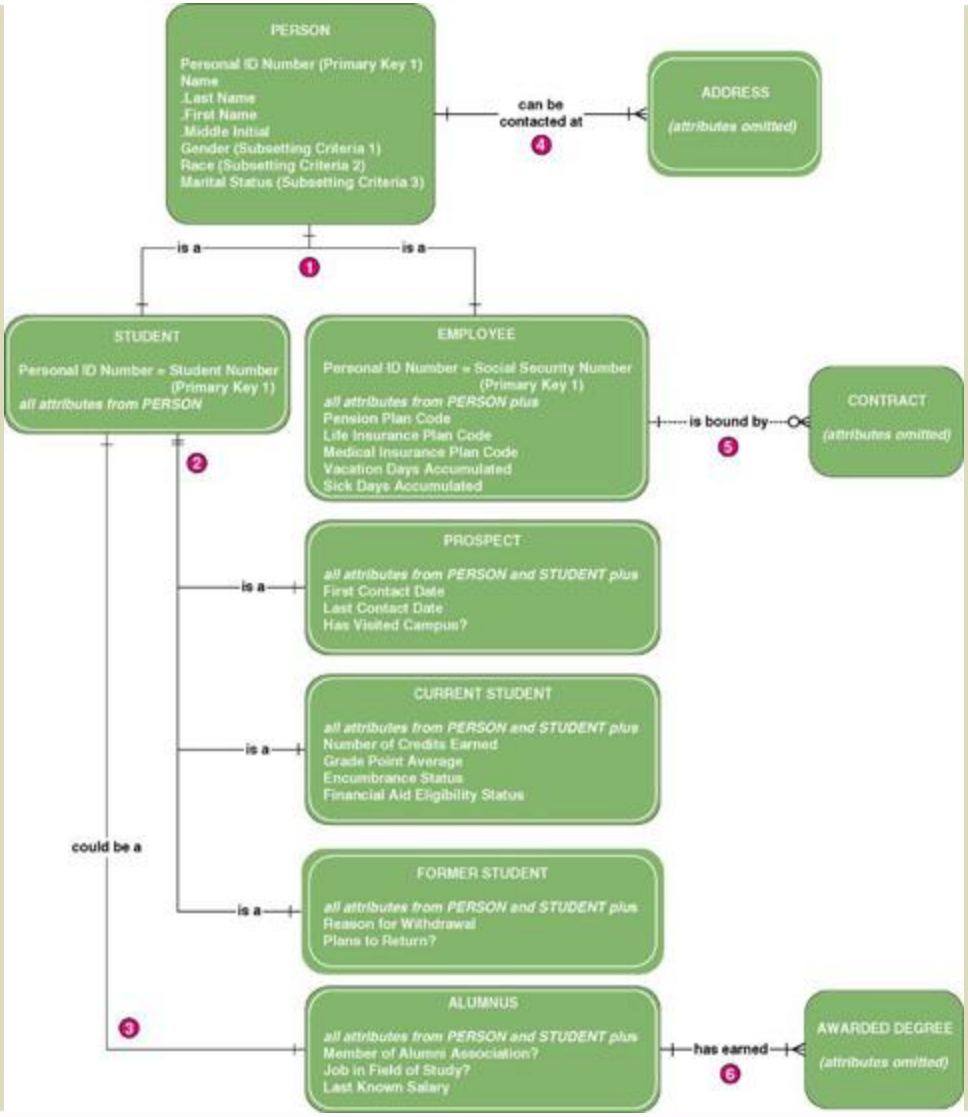
Generalization – a concept wherein the attributes that are common to several types of an entity are grouped into their own entity.

Supertype – an entity whose instances store attributes that are common to one or more entity subtypes.

Subtype – an entity whose instances may inherit common attributes from its entity supertype

And then add other attributes unique to the subtype.

Generalization Hierarchy



Process of Logical Data Modeling

- Strategic Data Modeling
  - Many organizations select IS development projects based on strategic plans.
    - Includes vision and architecture for information systems
    - Identifies and prioritizes develop projects
    - Includes enterprise data model as starting point for projects
- Data Modeling during Systems Analysis
  - Data model for a single information system is called an application data model.

Logical Model Development Stages

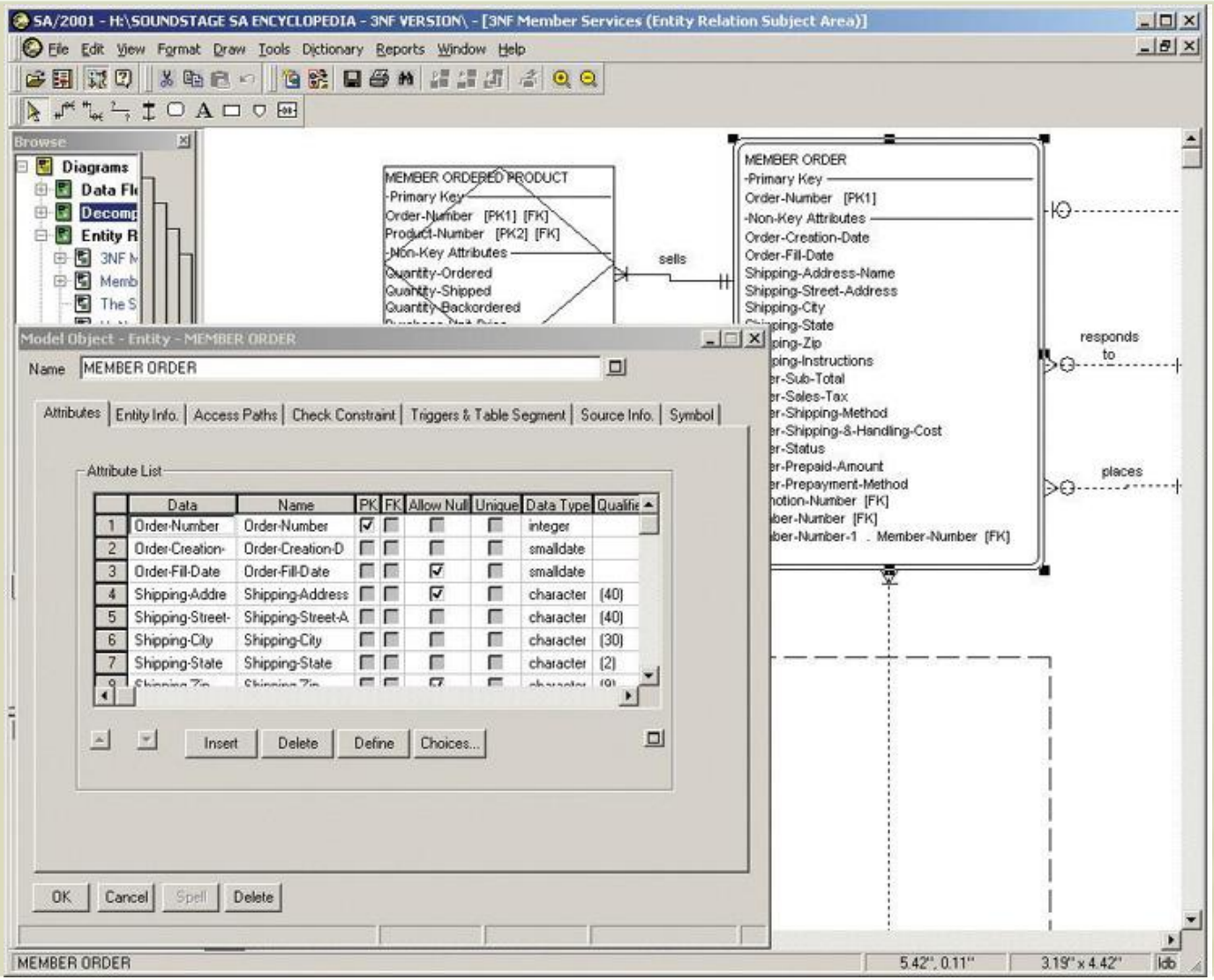
1. Context Data model
   - Includes only entities and relationships
   - To establish project scope
2. Key-based data model
   - Eliminate nonspecific relationships
   - Add associative entities
   - Include primary and alternate keys
   - Precise cardinalities
3. Fully attributed data model
   - All remaining attributes
   - Subsetting criteria
4. Normalized data model

Metadata - data about data.

JRP and Interview Questions for Data Modeling

Purpose	**Candidate Questions** (see textbook for a more complete list)
Discover system entities	What are the subjects of the business?
Discover entity keys	What unique characteristic (or characteristics) distinguishes an instance of each subject from other instances of the same subject?
Discover entity subsetting criteria	Are there any characteristics of a subject that divide all instances of the subject into useful subsets?
Discover attributes and domains	What characteristics describe each subject?
Discover security and control needs	Are there any restrictions on who can see or use the data?
Discover data timing needs	How often does the data change?
Discover generalization hierarchies	Are all instances of each subject the same?
Discover relationships?	What events occur that imply associations between subjects?
Discover cardinalities	Is each business activity or event handled the same way, or are there special circumstances?

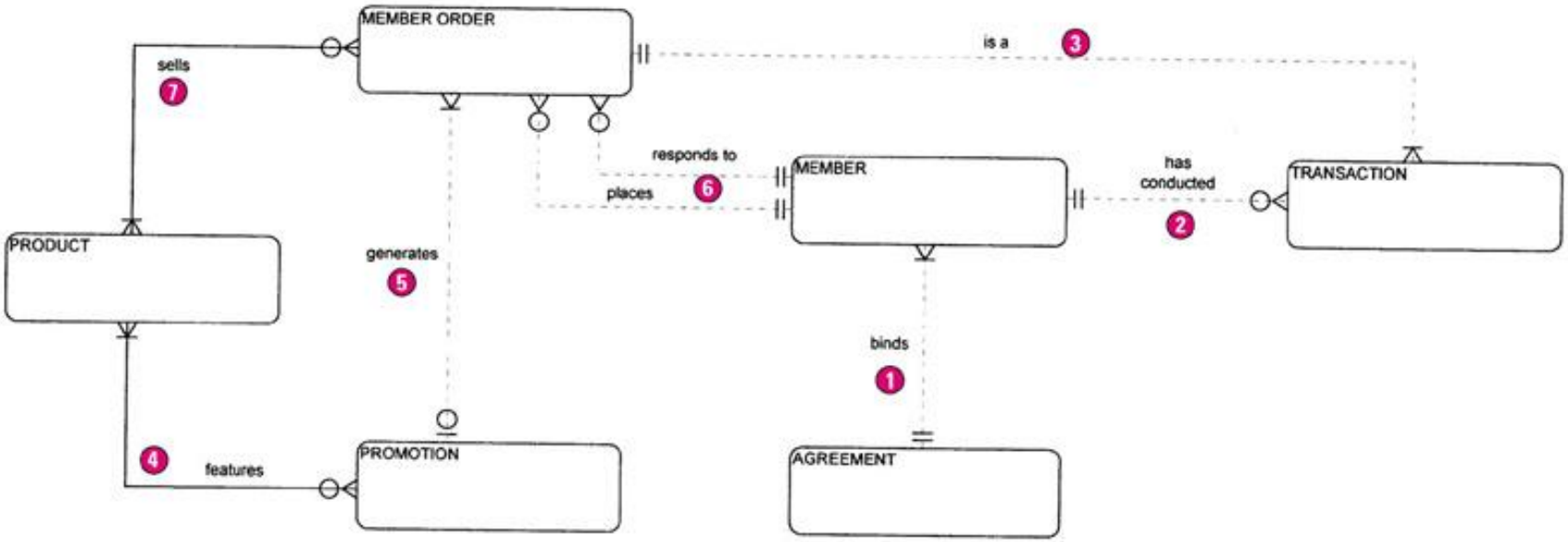
Automated Tools for Data Modeling



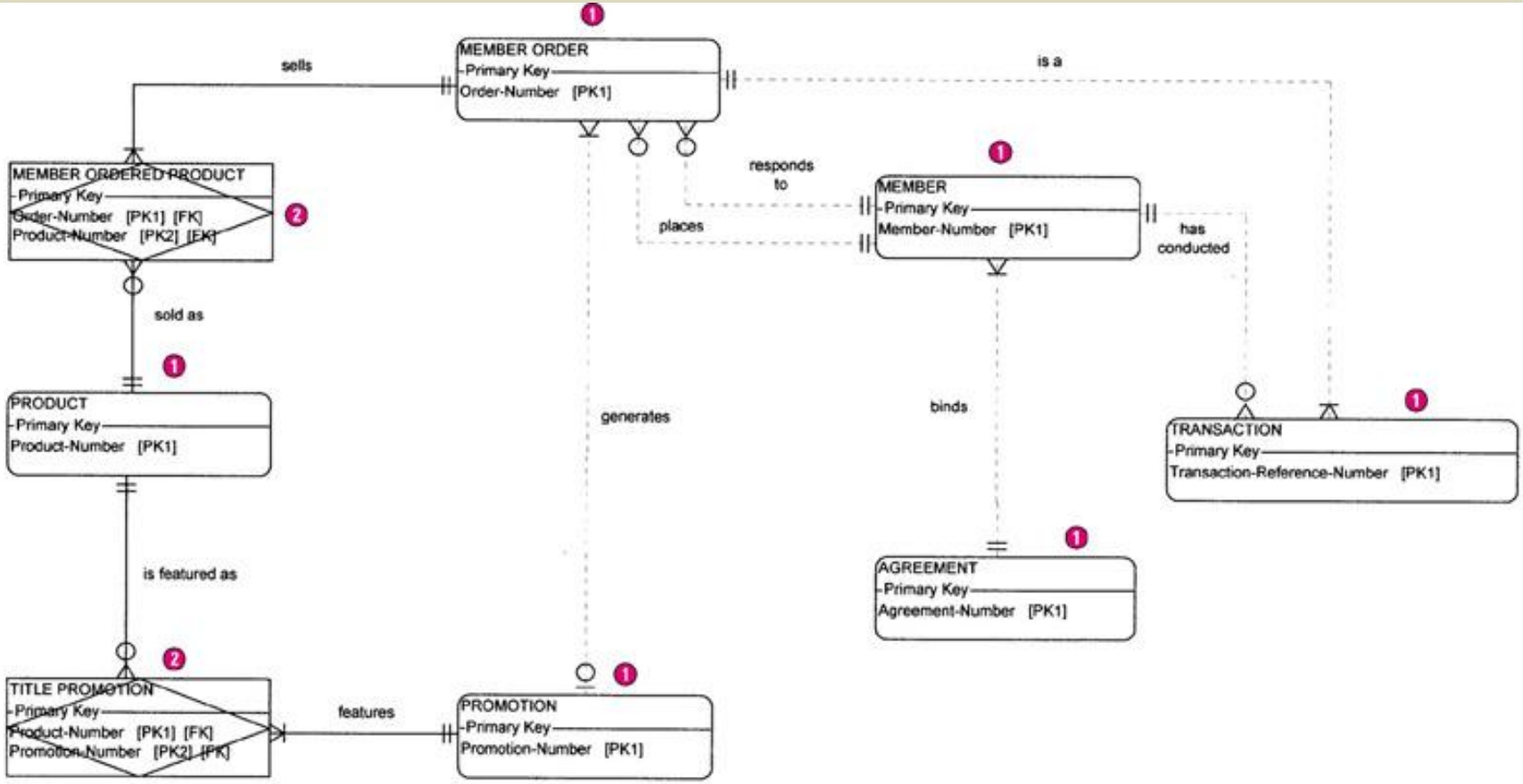
Entity Discovery

- In interviews or JRP sessions, pay attention to key words (i.e. "we need to keep track of ...").
- In interviews or JRP sessions, ask users to identify things about which they would like to capture, store, and produce information.
- Study existing forms, files, and reports.
- Scan use case narratives for nouns.
- Some CASE tools can reverse engineer existing files and databases.

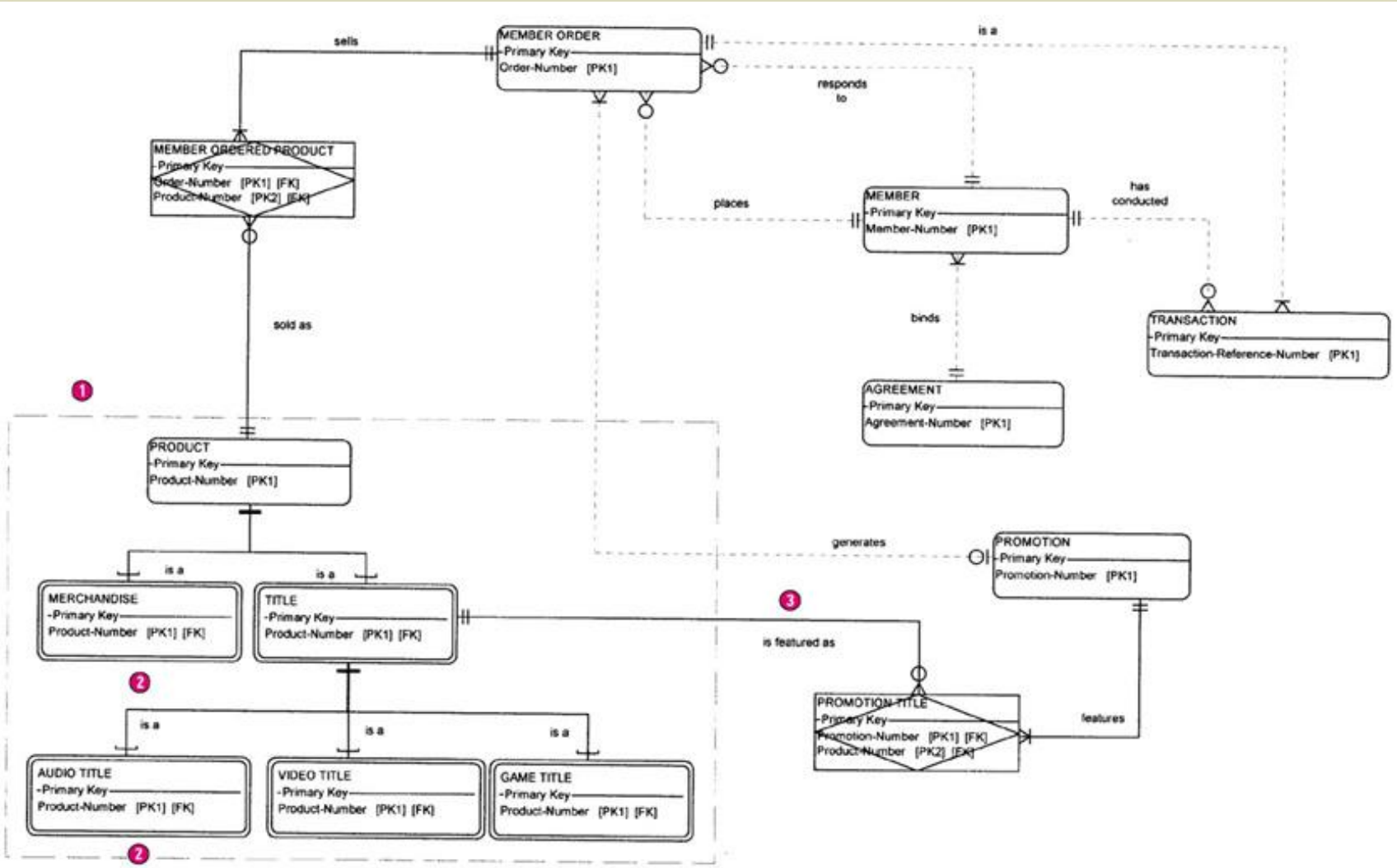
The Context Data Model



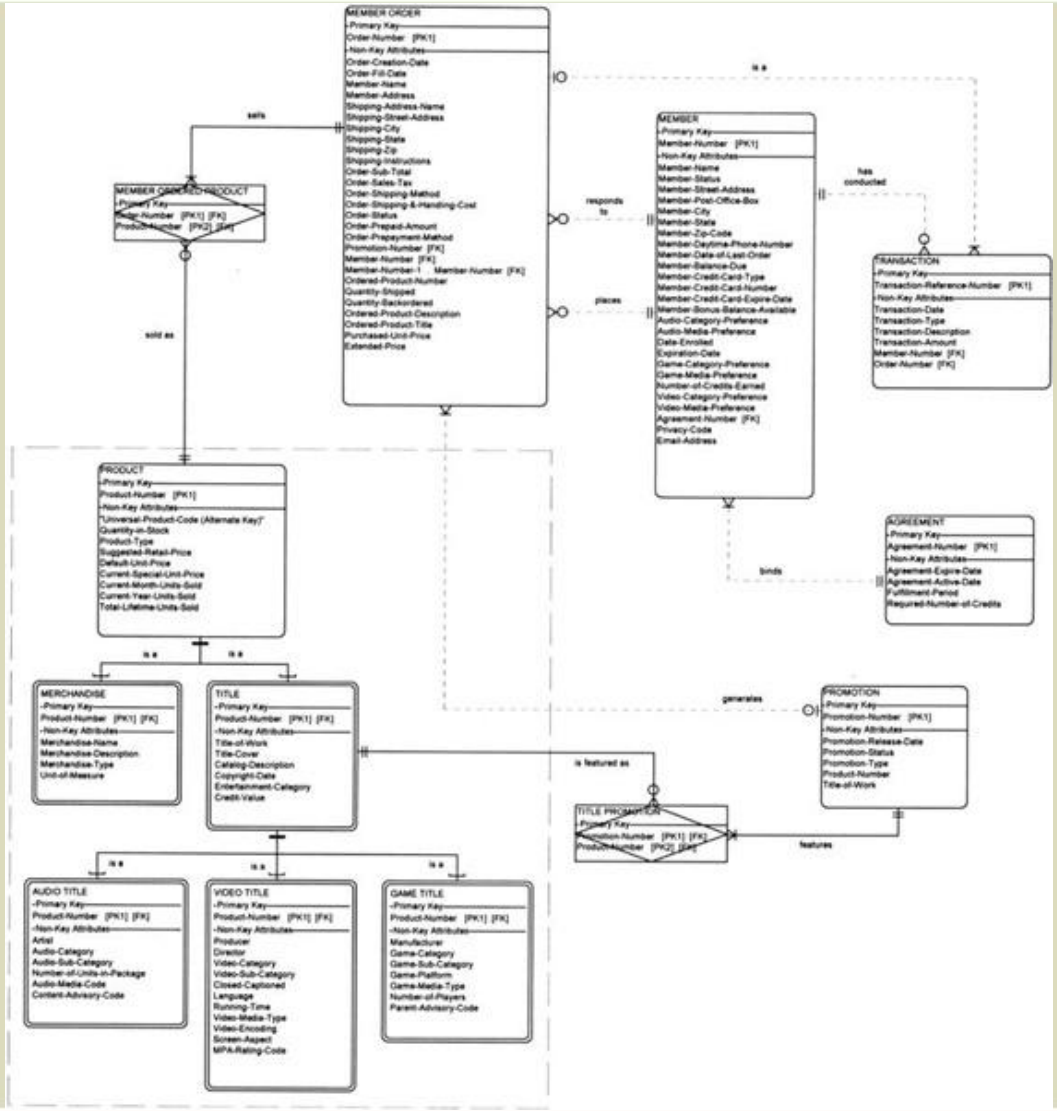
The Key-based Data Model



The Key-based Data Model with Generalization



The Fully-Attributed Data Model



What is a Good Data Model?

- A good data model is simple.
  - Data attributes that describe any given entity should describe only that entity.
  - Each attribute of an entity instance can have only one value.
- A good data model is essentially nonredundant.
  - Each data attribute, other than foreign keys, describes at most one entity.
  - Look for the same attribute recorded more than once under different names.
- A good data model should be flexible and adaptable to future needs.

Data Analysis & Normalization

Data analysis – a technique used to improve a data model for implementation as a database.

Goal is a simple, nonredundant, flexible, and adaptable database.

Normalization – a data analysis technique that organizes data into groups to form nonredundant, stable, flexible, and adaptive entities.

Normalization: 1NF, 2NF, 3NF

First normal form (1NF) – entity whose attributes have no more than one value for a single instance of that entity

- Any attributes that can have multiple values actually describe a separate entity, possibly an entity and relationship.

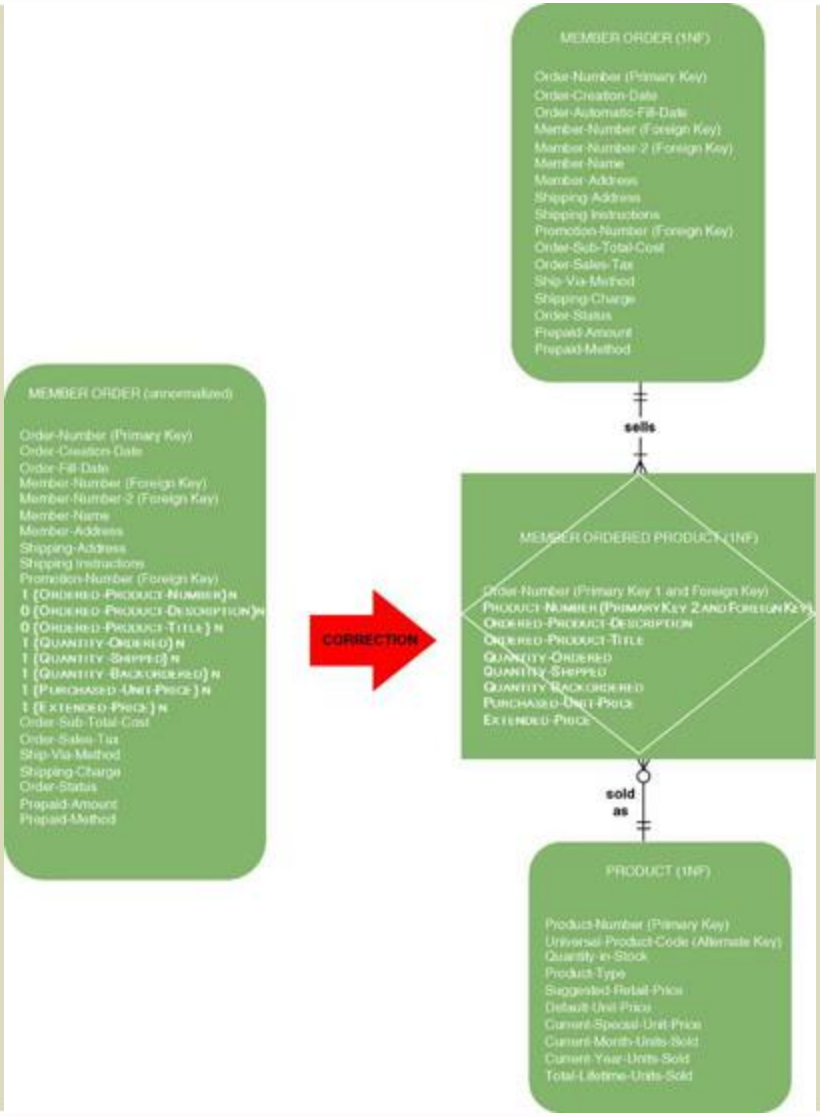
Second normal form (2NF) – entity whose nonprimary-key attributes are dependent on the full primary key.

- Any nonkey attributes dependent on only part of the primary key should be moved to entity where that partial key is the full key. May require creating a new entity and relationship on the model.

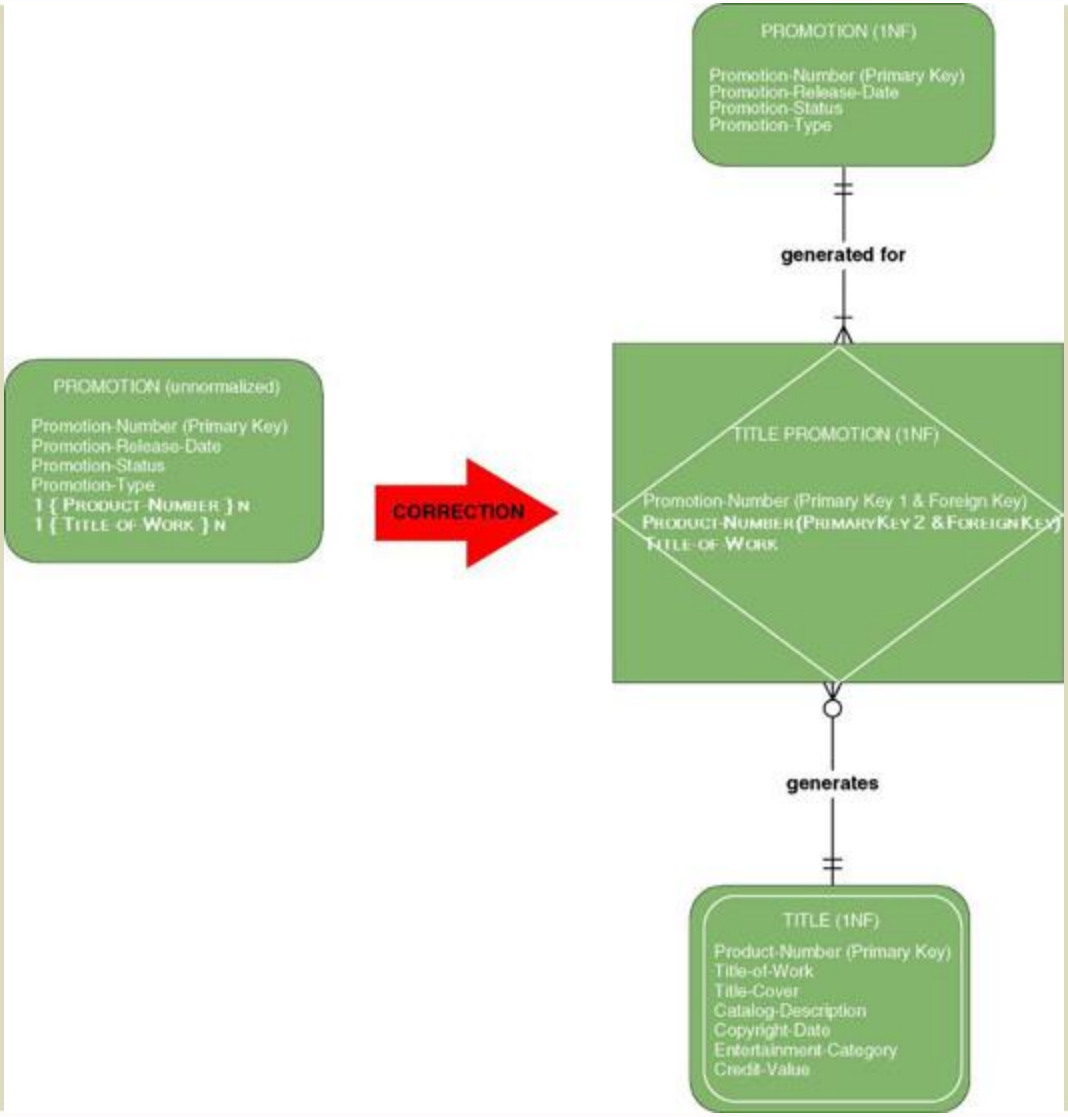
Third normal form (3NF) – entity whose nonprimary-key attributes are not dependent on any other non-primary key attributes.

- 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.

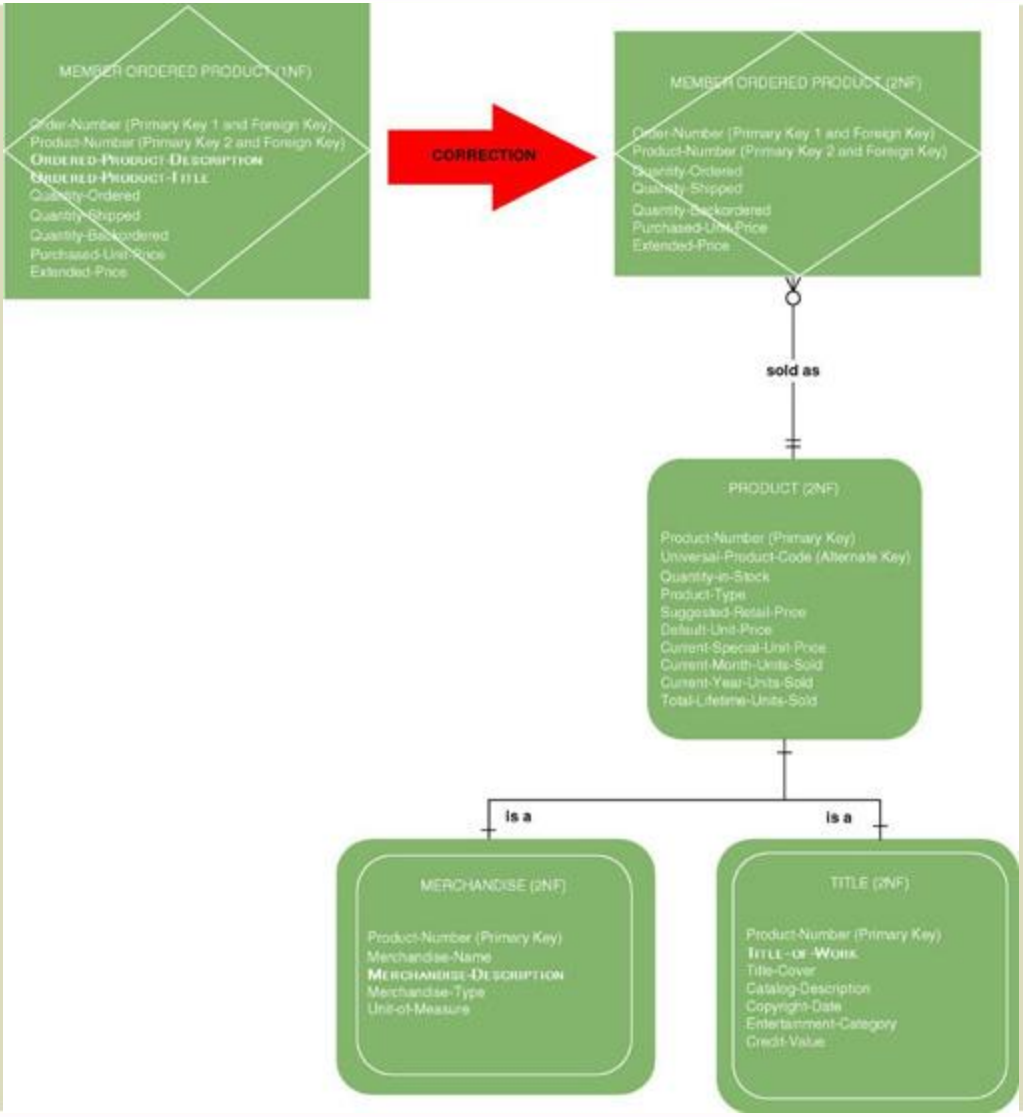
First Normal Form Example 1



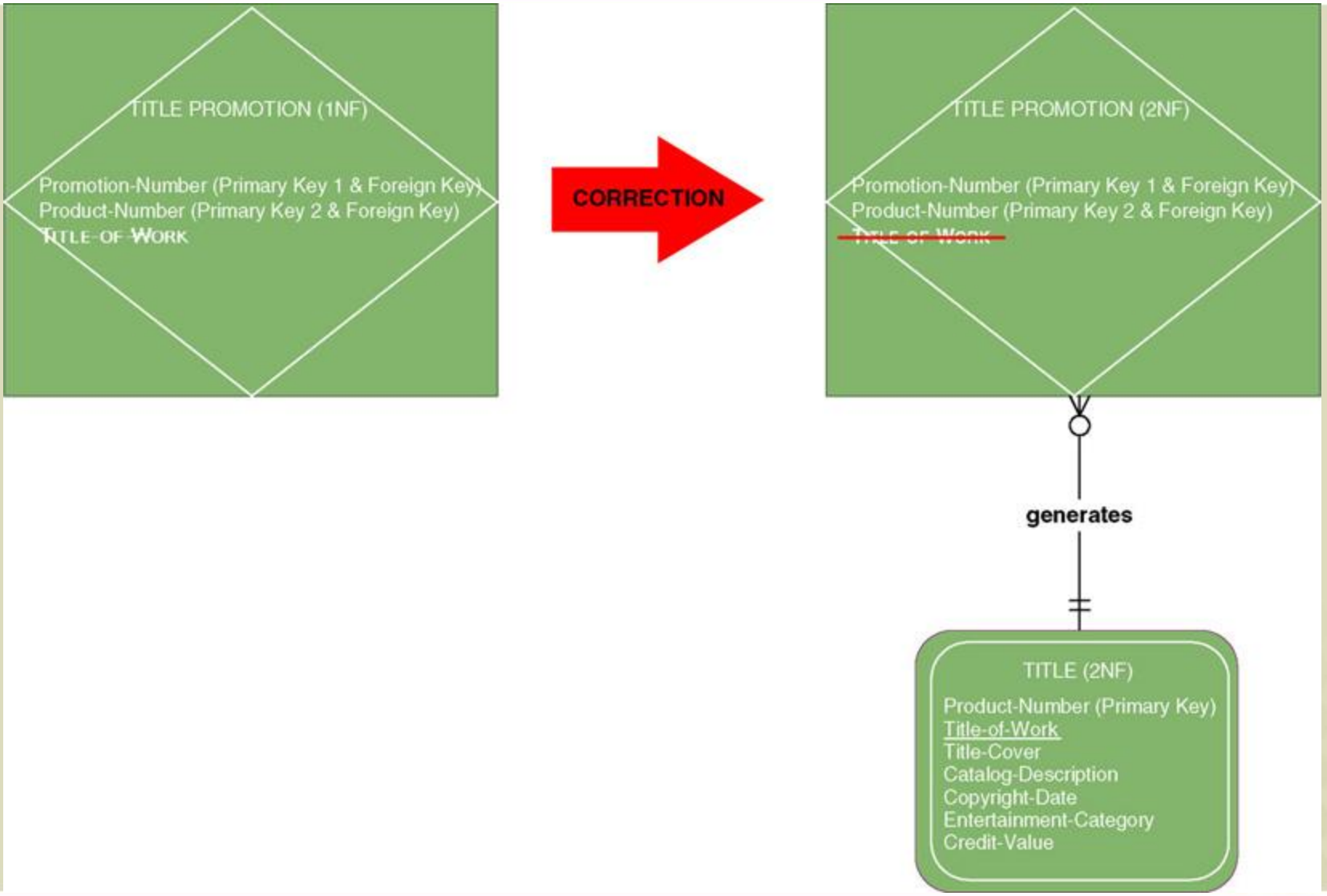
First Normal Form Example 2



Second Normal Form Example 1

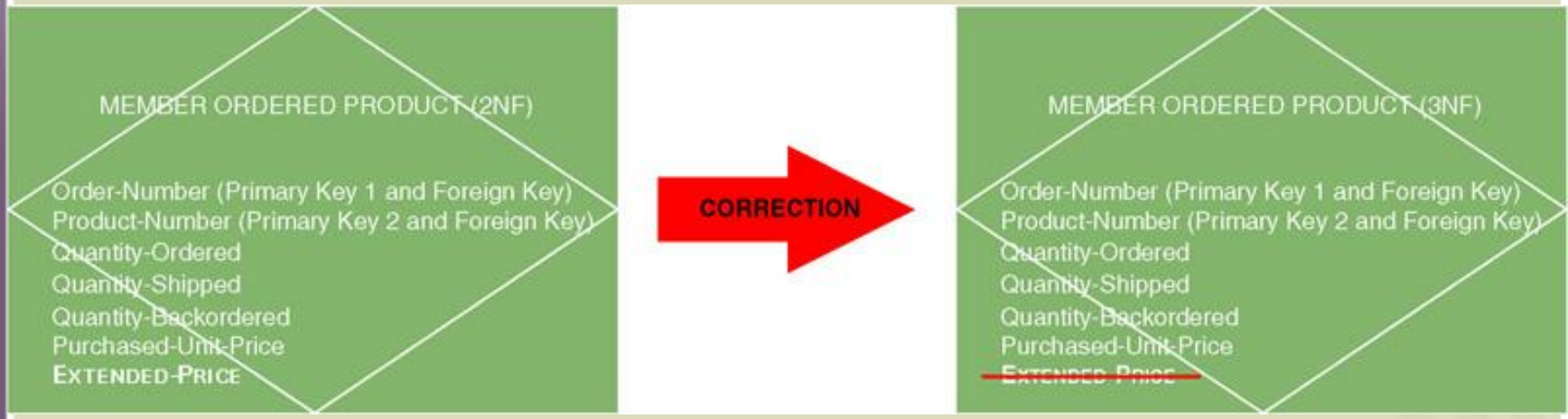


Second Normal Form Example 2



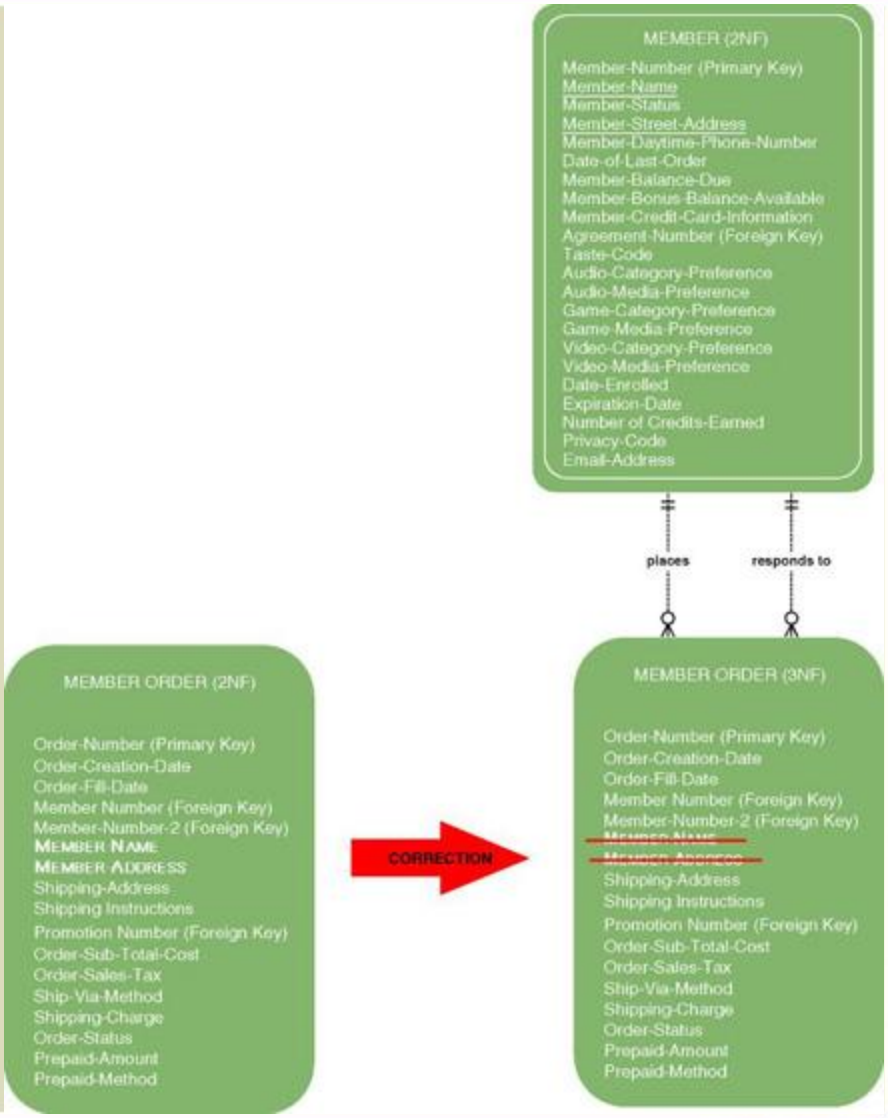
Third Normal Form Example 1

Derived attribute – an attribute whose value can be calculated from other attributes or derived from the values of other attributes.

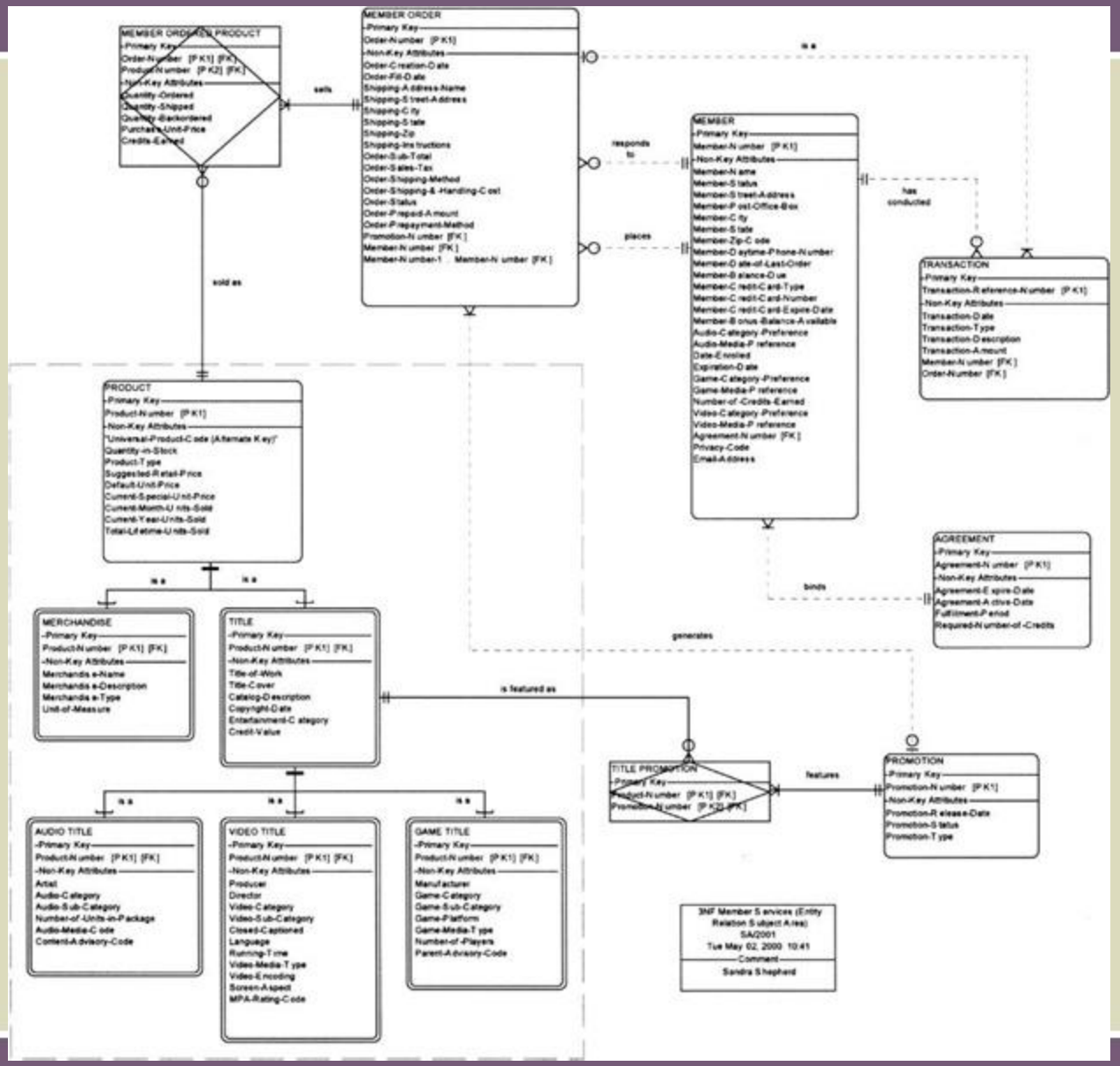


Third Normal Form Example 2

Transitive dependency – when the value of a nonkey attribute is dependent on the value of another nonkey attribute other than by derivation.



SoundStage 3NF Data Model



Data-to-Location-CRUD Matrix

Entity . Attribute / Location	Customers	Kansas City	. Marketing	. Advertising	. Warehouse	. Sales	. A/R	Boston	. Sales	. Warehouse	San Francisco	. Sales	San Diego	. Warehouse
Customer	**INDV**					**ALL**	**ALL**		**SS**	**SS**		**SS**		**SS**
.Customer Number	R				R	CRUD	R		CRUD	R		CRUD		R
.Customer Name	RU				R	CRUD	R		CRUD	R		CRUD		R
.Customer Address	RU				R	CRUD	R		CRUD	R		CRUD		R
.Customer Credit Rating	X					R	RU		R			R		
.Customer Balance Due	R					R	RU		R			R		
Order	**INDV**		**ALL**		**SS**	**ALL**			**SS**	**SS**		**SS**		**SS**
.Order Number	SRD		R	CRUD	R	CRUD	R		CRUD	R		CRUD		R
.Order Date	SRD		R	CRUD	R	CRUD	R		CRUD	R		CRUD		R
.Order Amount	SRD		R	CRUD		CRUD	R		CRUD	R		CRUD		R
Ordered Product	**INDV**		**ALL**		**SS**	**ALL**			**SS**	**SS**		**SS**		**SS**
.Quantity Ordered	SUD		R	CRUD	R	CRUD	R		CRUD			CRUD		
.Ordered Item Unit Price	SUD		R	CRUD		CRUD	R		CRUD			CRUD		
Product	**ALL**		**ALL**	**ALL**	**ALL**	**ALL**			**ALL**	**ALL**		**ALL**		**ALL**
.Product Number	R		CRUD	R	R	R			R	R		R		R
.Product Name	R		CRUD	R	R	R			R	R		R		R
.Product Description	R		CRUD	RU	R	R			R	R		R		R
.Product Unit of Measure	R		CRUD	R	R	R			R	R		R		R
.Product Current Unit Price	R		CRUD	R		R			R	R		R		R
.Product Quantity on Hand	X				RU	R			R	RU		R		RU

INDV = individual **ALL** = ALL **SS** = subset X = no access

S = submit C = create R = read U = update D = delete