Chapter 6. Data Modeling

UNIVERSITY of GREENWICH
Alliance with FPT Education

Pearson
BTEC

Chapter 6 Outline

- The Entity Relationship Diagram (ERD).
  - Elements of ERD
  - The Data Dictionary and Metadata
- Creating an Entity Relationship Diagram.
- Validating an ERD.

INTRODUCTION

- In this chapter, we discuss how the **data** are organized and presented.
- A ***data model*** is a formal way of representing the data that are used and created by a business system.
- During analysis (this Chapter), analysts draw a ***Logical data model*** which shows the logical organization of data without indicating how it is stored, created, or manipulated.
- During design (Chapter 11), analysts draw a ***physical data model*** to reflect how the data will physically be stored in databases or files.

(cont’d)

- Topics of this chapter:

- Creating an ***entity relationship diagram (ERD)***.

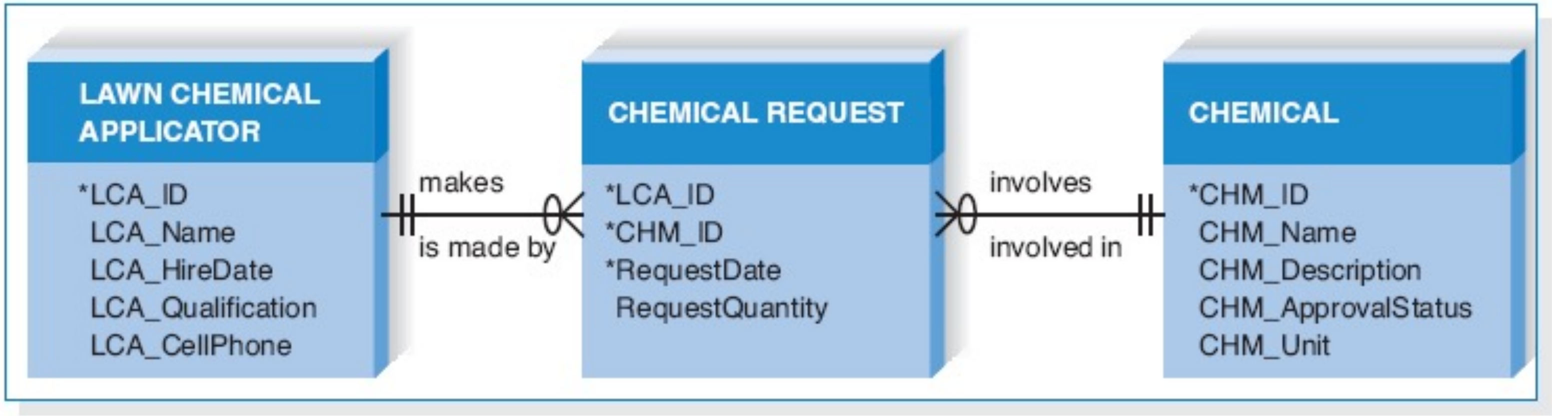
- ***Normalization***, a technique that helps analysts validate the data models.

- How data models balance, or interrelate, with the process models.

THE ENTITY-RELATIONSHIP DIAGRAM (ERD)

- An ***entity-relationship diagram (ERD)*** is a picture showing the information that is created, stored, and used by a business system.
- **Entities** lists similar kinds of information
- Lines drawn between entities represent **relationships** among the data.
- Special symbols communicate high-level business rules.

Reading an Entity Relationship Diagram



LAWN CHEMICAL APPLICATOR

*LCA_ID
LCA_Name
LCA_HireDate
LCA_Qualification
LCA_CellPhone

makes

is made by

CHEMICAL REQUEST

*LCA_ID
*CHM_ID
*RequestDate
RequestQuantity

involves

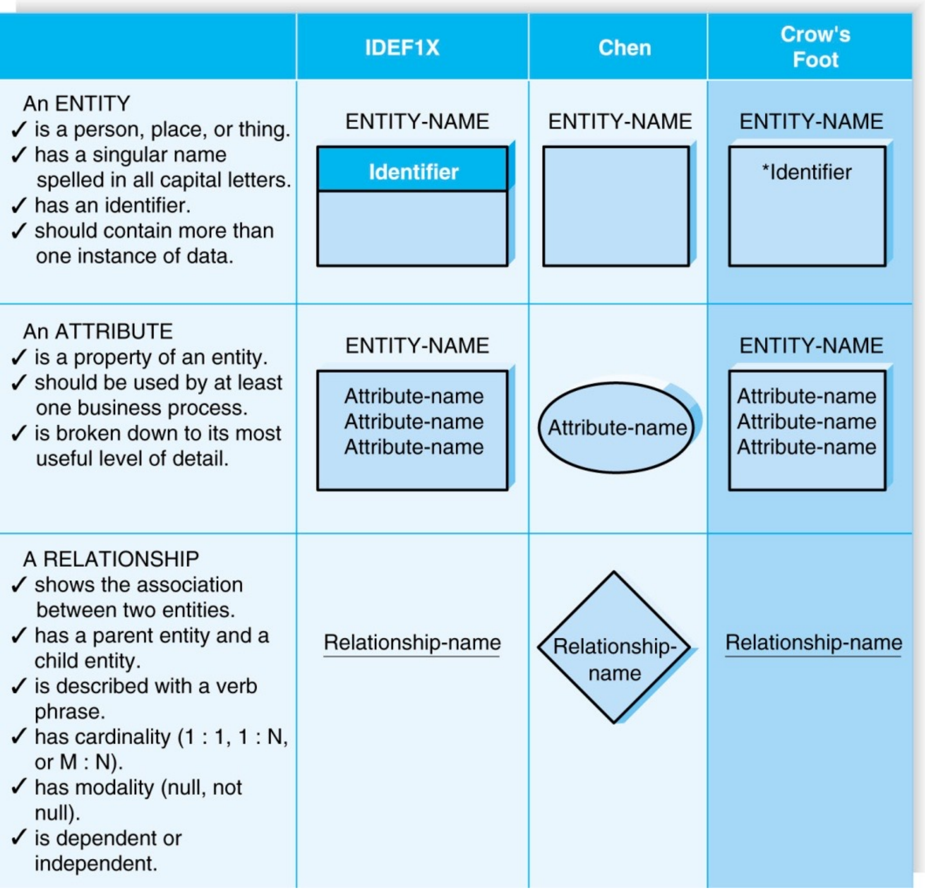
involved in

CHEMICAL

*CHM_ID
CHM_Name
CHM_Description
CHM_ApprovalStatus
CHM_Unit

FIGURE 6-1
Chemical Request ERD

Elements of an Entity Relationship Diagram



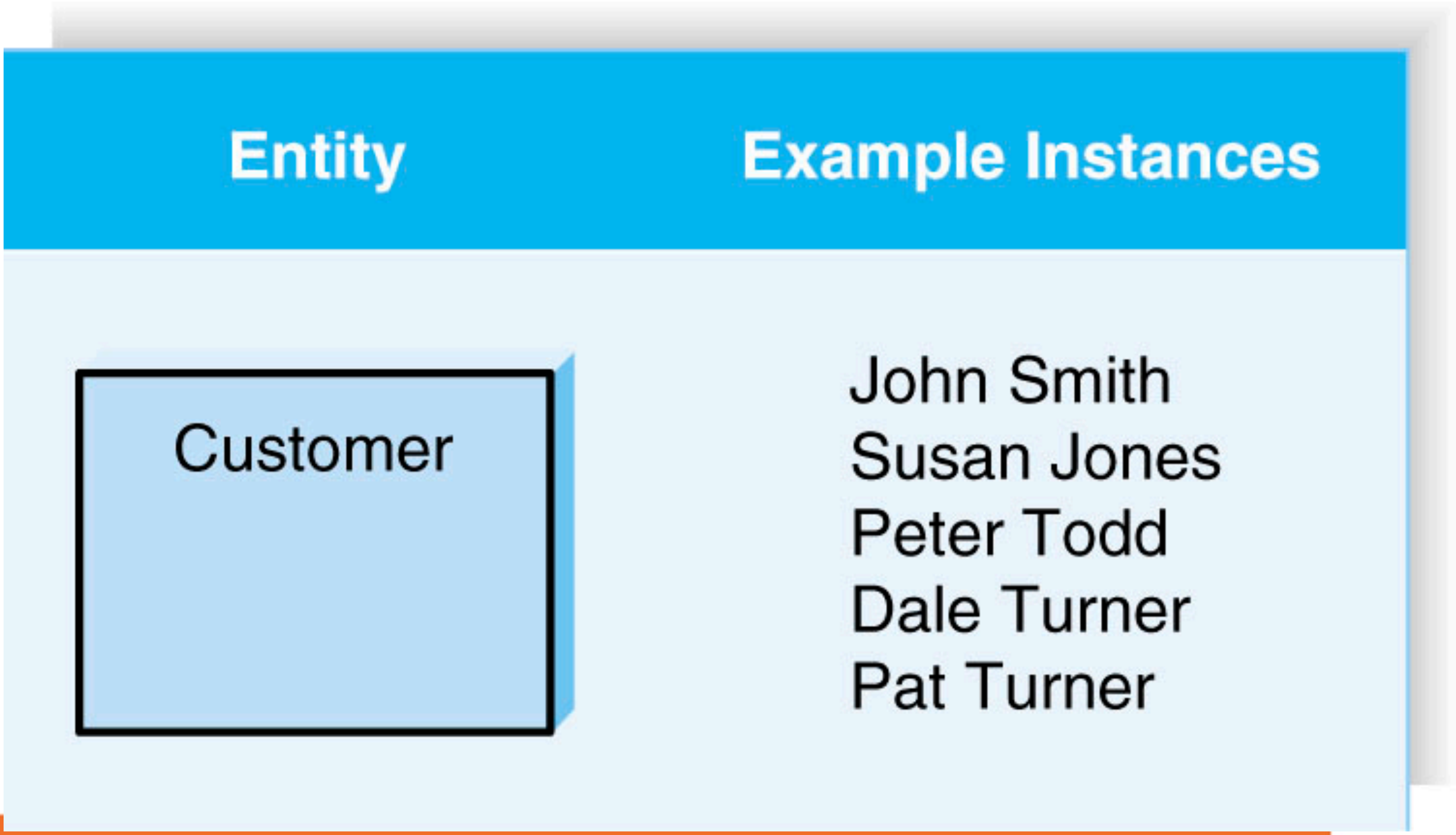
	IDEF1X	Chen	Crow's Foot
An ENTITY ✓ is a person, place, or thing. ✓ has a singular name spelled in all capital letters. ✓ has an identifier. ✓ should contain more than one instance of data.	ENTITY-NAME Identifier	ENTITY-NAME	ENTITY-NAME *Identifier
An ATTRIBUTE ✓ is a property of an entity. ✓ should be used by at least one business process. ✓ is broken down to its most useful level of detail.	ENTITY-NAME Attribute-name Attribute-name Attribute-name	Attribute-name	ENTITY-NAME Attribute-name Attribute-name Attribute-name
A RELATIONSHIP ✓ shows the association between two entities. ✓ has a parent entity and a child entity. ✓ is described with a verb phrase. ✓ has cardinality (1 : 1, 1 : N, or M : N). ✓ has modality (null, not null). ✓ is dependent or independent.	Relationship-name	Relationship-name	Relationship-name

Entity

- The ***entity*** is the basic building block for a data model. It is a person, place, event, or thing about which data is collected.
- **Entities** represent something for which there exist multiple ***instances***, or *occurrences*.

 - E.g., John Smith could be an instance of the customer entity.

(cont’d)



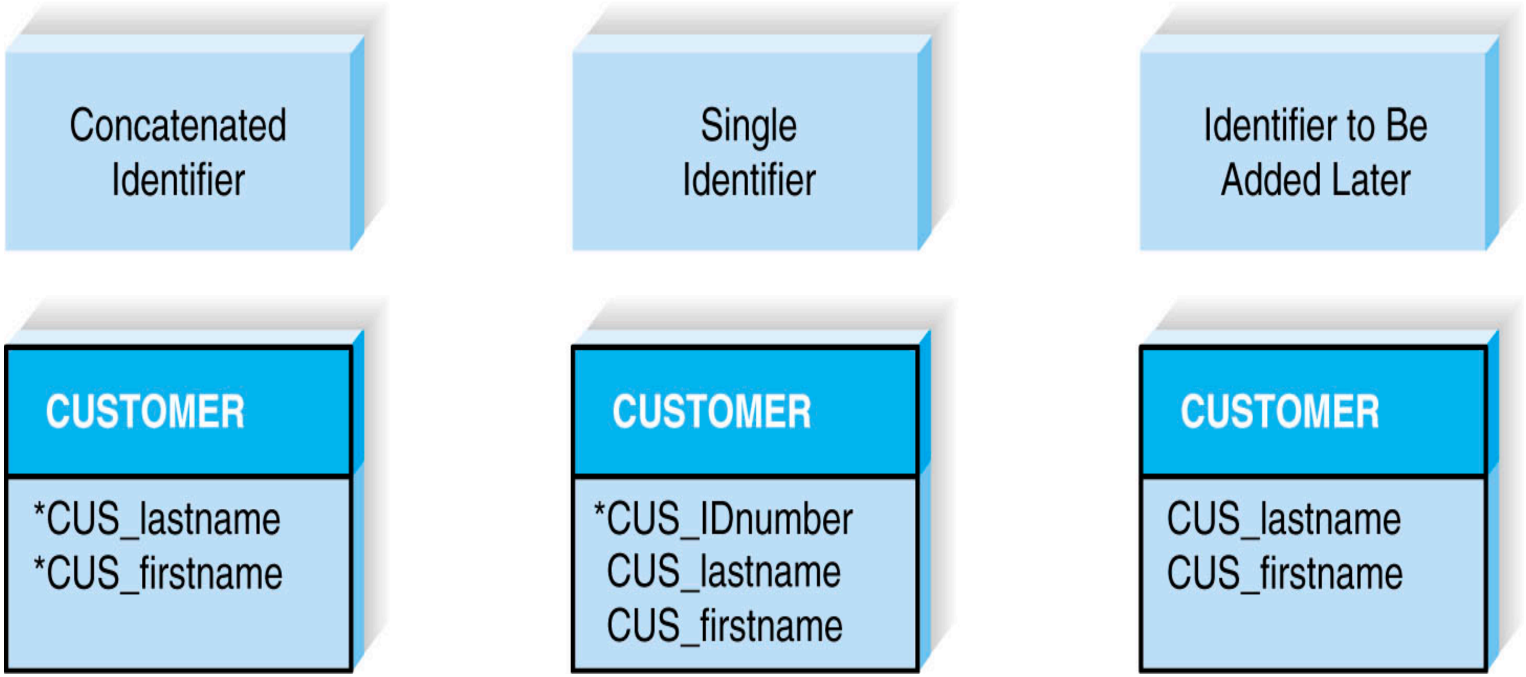
Entity	Example Instances
Customer	John Smith Susan Jones Peter Todd Dale Turner Pat Turner

Attributes

- An ***attribute*** is some type of information that is captured about an entity.
- Attributes are nouns that are listed with an entity.
- One or more attributes can serve as the entity **identifier** - the attribute(s) that can uniquely identify one instance of an entity.
- **Concatenated identifier** - several attributes are combined to uniquely identify an instance.

(cont’d)

Choices for Identifiers



Concatenated Identifier

CUSTOMER

*CUS_lastname
*CUS_firstname

Single Identifier

CUSTOMER

*CUS_IDnumber
CUS_lastname
CUS_firstname

Identifier to Be Added Later

CUSTOMER

CUS_lastname
CUS_firstname

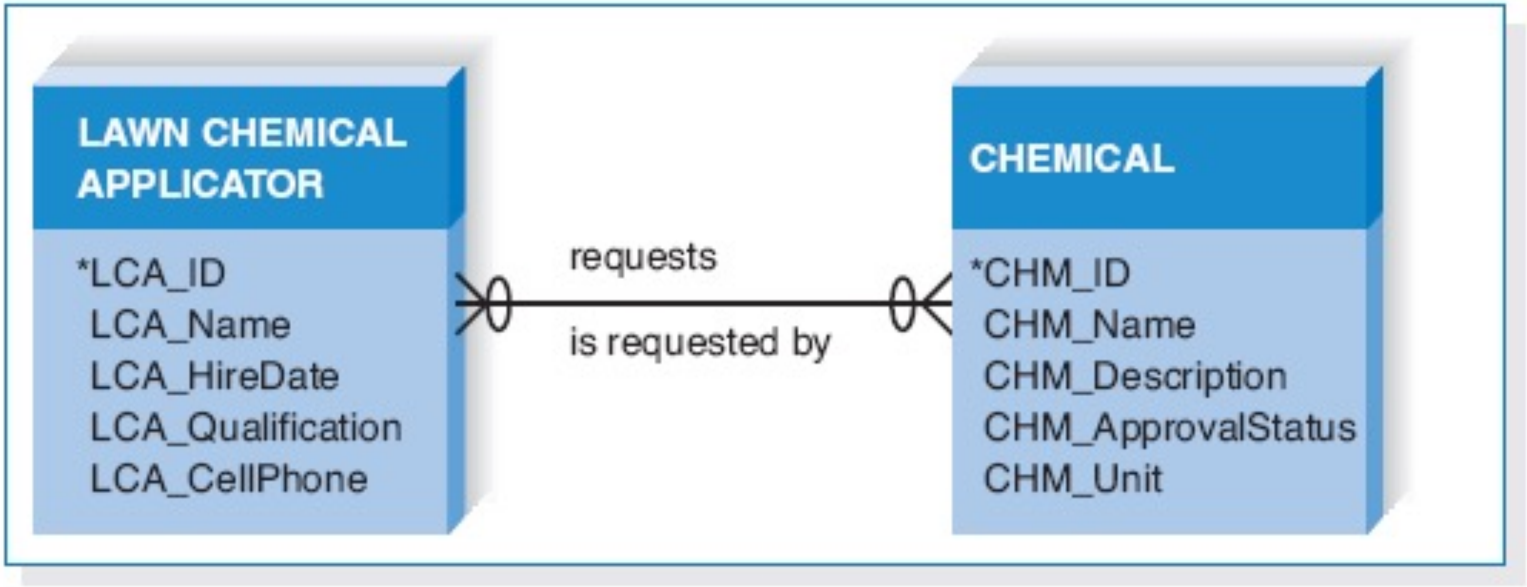
Relationships

- ***Relationships*** are associations between entities.
- Every relationship has a ***parent entity*** and a ***child entity.***
- Relationships should be labeled with active verbs.

Cardinality

- A relationship has ***cardinality*** which is the ratio of parent instances to child instances.
- The **1:1 relationship** means that one instance of the parent entity is associated with one instance of the child entity.
- The **1:N relationship** means that a single instance of a parent entity is associated with many instances of a child entity.
- The **M:N relationship** means that many instances of a parent entity can relate to many instances of a child entity.

Example of M:N Relationship



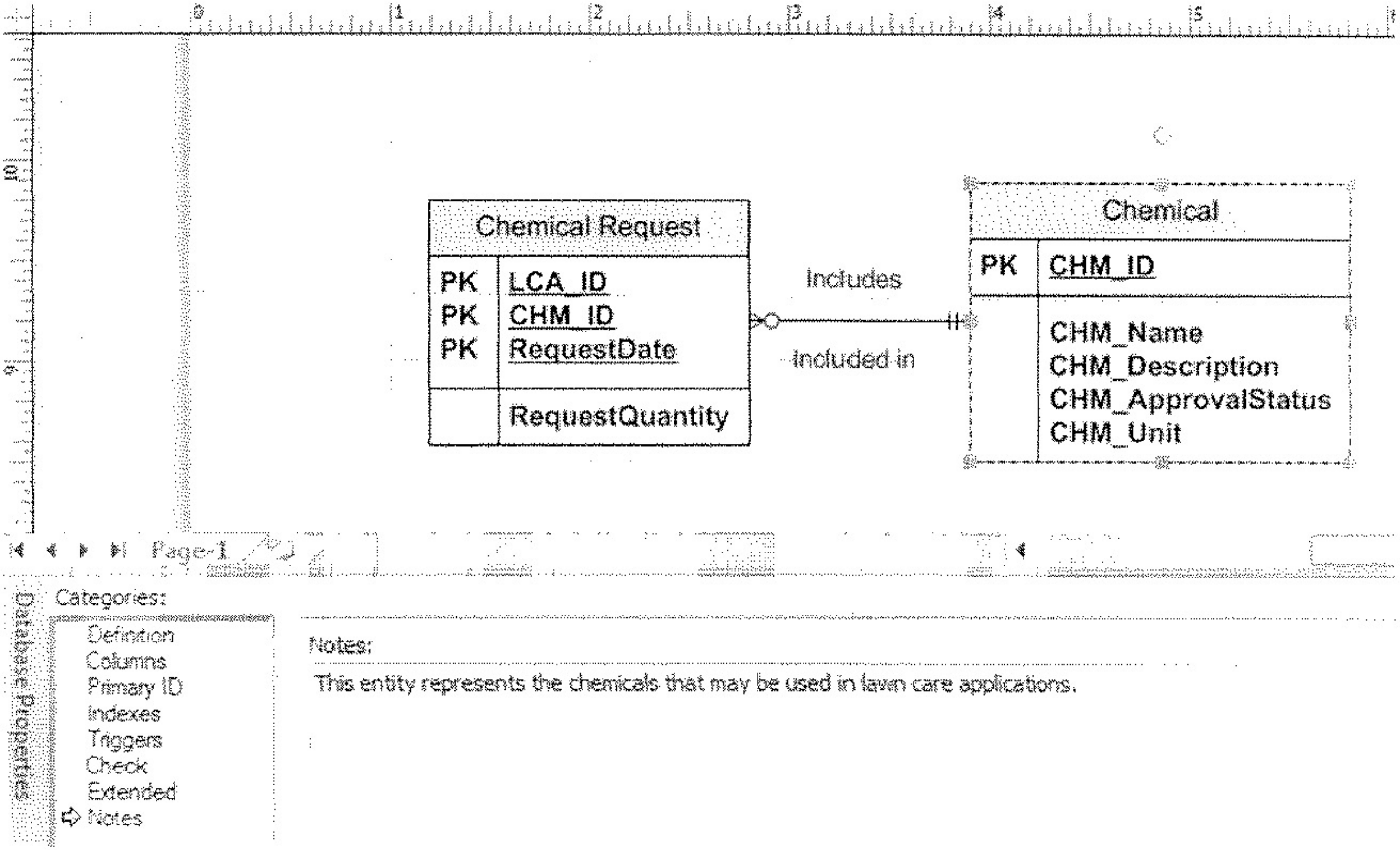
Modality

- A relationship has ***modality*** of null or not null, which refers to whether or not an instance of a child entity can exist without a related instance in the parent entity.
- Null means that an instance of a child entity can exist without a related instance in the parent entity.
- Not Null means that an instance of a child entity can’t exist without a related instance in the parent entity.

The Data Dictionary and Metadata

- A ***data dictionary*** contains the information about the entities, attributes, and relationships on the ERD, or ***metadata***.
- **Metadata** is data about data.
- Metadata is stored in the data dictionary so it can be shared by developers and users throughout the SDLC.

Ex (cont’d)



Chemical Request

PK	LCA_ID
PK	CHM_ID
PK	RequestDate
	RequestQuantity

Includes

Included in

Chemical

PK	CHM_ID
	CHM_Name CHM_Description CHM_ApprovalStatus CHM_Unit

Categories:

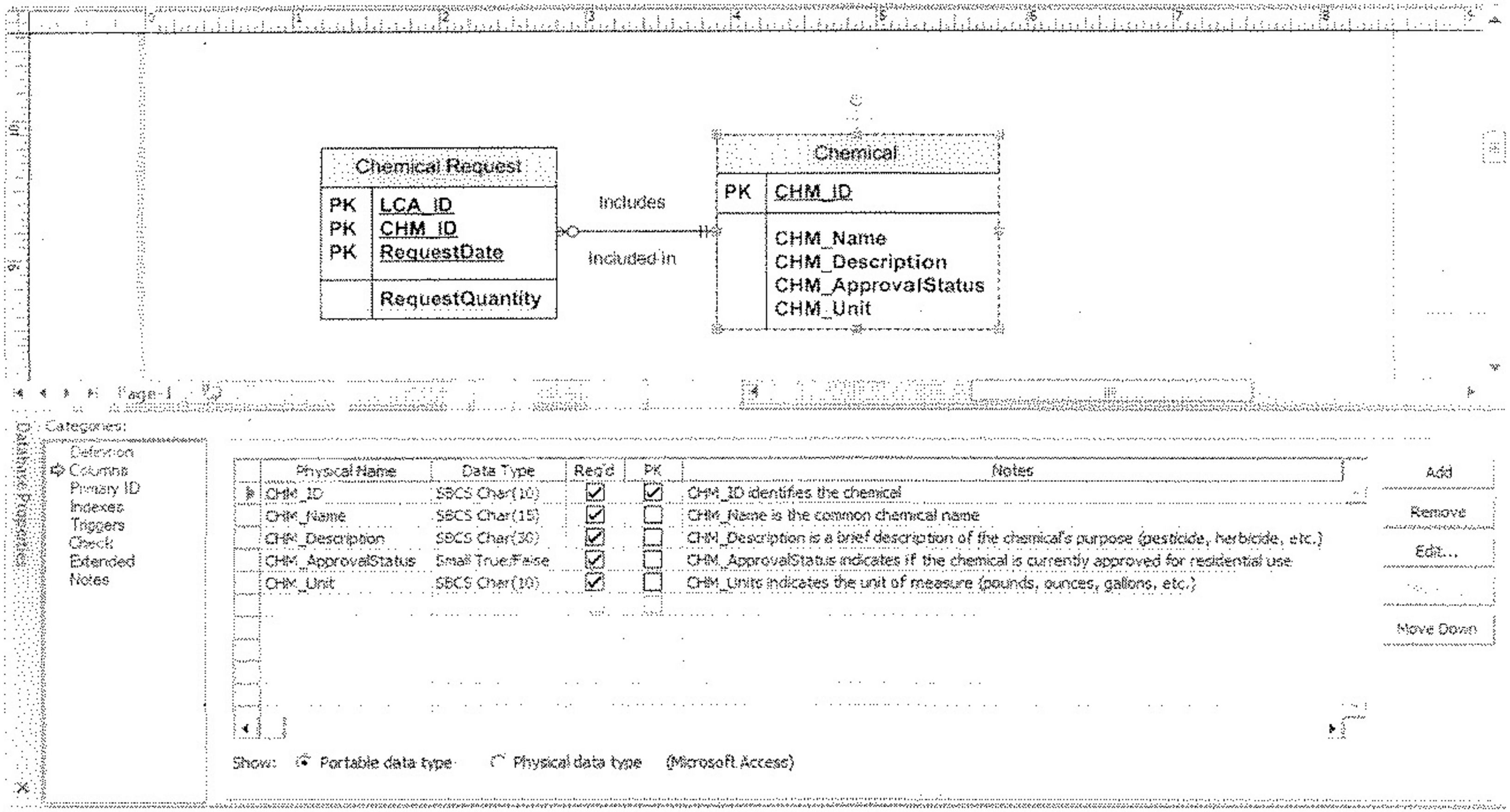
- Definition
- Columns
- Primary ID
- Indexes
- Triggers
- Check
- Extended
- Notes

Notes:

This entity represents the chemicals that may be used in lawn care applications.

Figure 6-6 Data Dictionary Entry for Chemical Entity (in Visio 2010)

(cont'd)

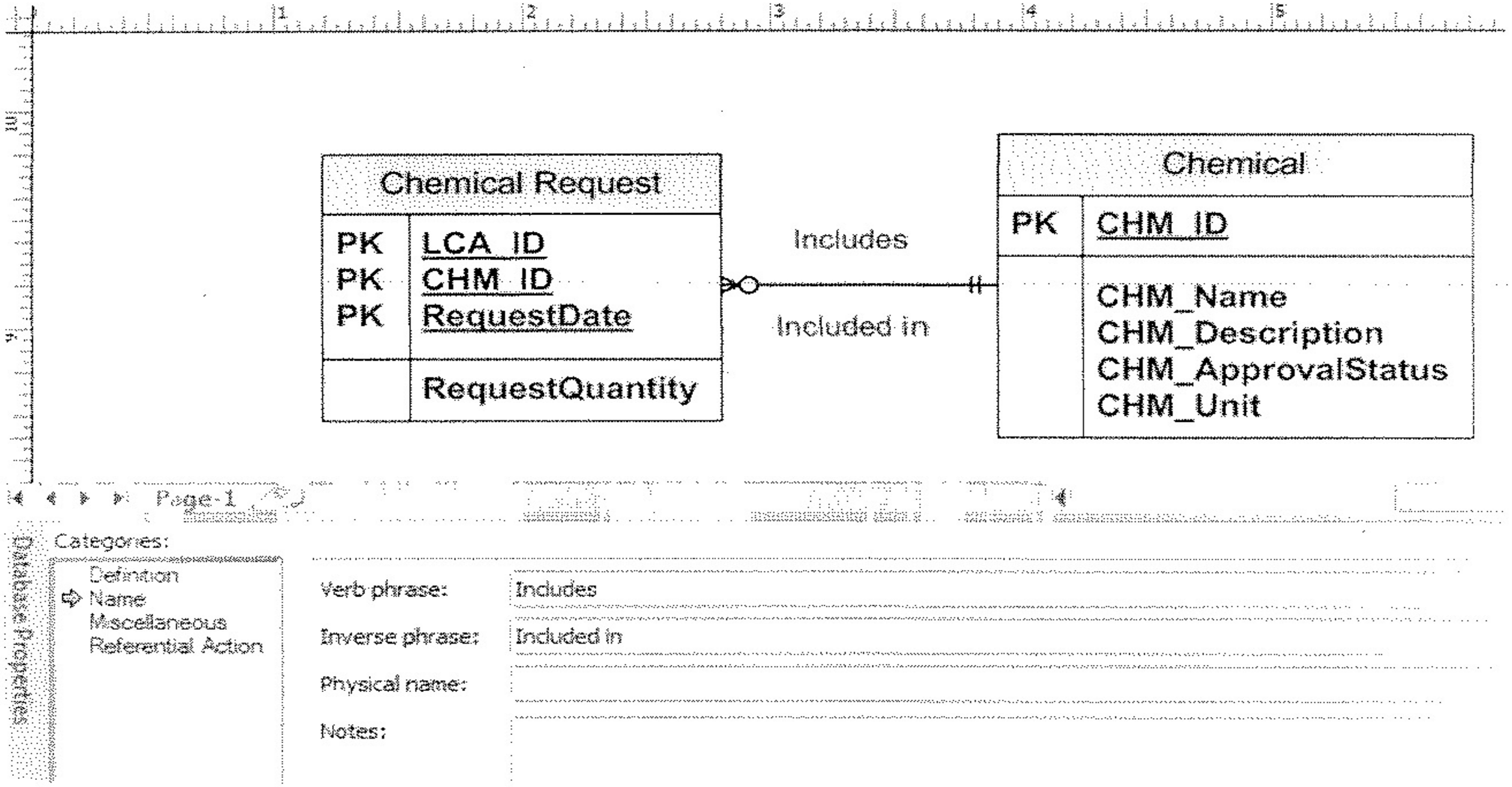


Physical Name	Data Type	Req'd	PK	Notes
CHM_ID	SBCS Char(10)	☑	☑	CHM_ID identifies the chemical
CHM_Name	SBCS Char(15)	☑	□	CHM_Name is the common chemical name
CHM_Description	SBCS Char(30)	☑	□	CHM_Description is a brief description of the chemical's purpose (pesticide, herbicide, etc.)
CHM_ApprovalStatus	Small True/False	☑	□	CHM_ApprovalStatus indicates if the chemical is currently approved for residential use
CHM_Unit	SBCS Char(10)	☑	□	CHM_Units indicates the unit of measure (pounds, ounces, gallons, etc.)

Show: ● Portable data type ○ Physical data type (Microsoft Access)

Figure 6-7 Data Dictionary Entry for Chemical Attributes (in Visio 2010)

(cont'd)



Chemical Request

PK	LCA_ID
PK	CHM_ID
PK	RequestDate
	RequestQuantity

Includes

Included in

Chemical

PK	CHM_ID
	CHM_Name CHM_Description CHM_ApprovalStatus CHM_Unit

Categories:

- Definition
- Name
- Miscellaneous
- Referential Action

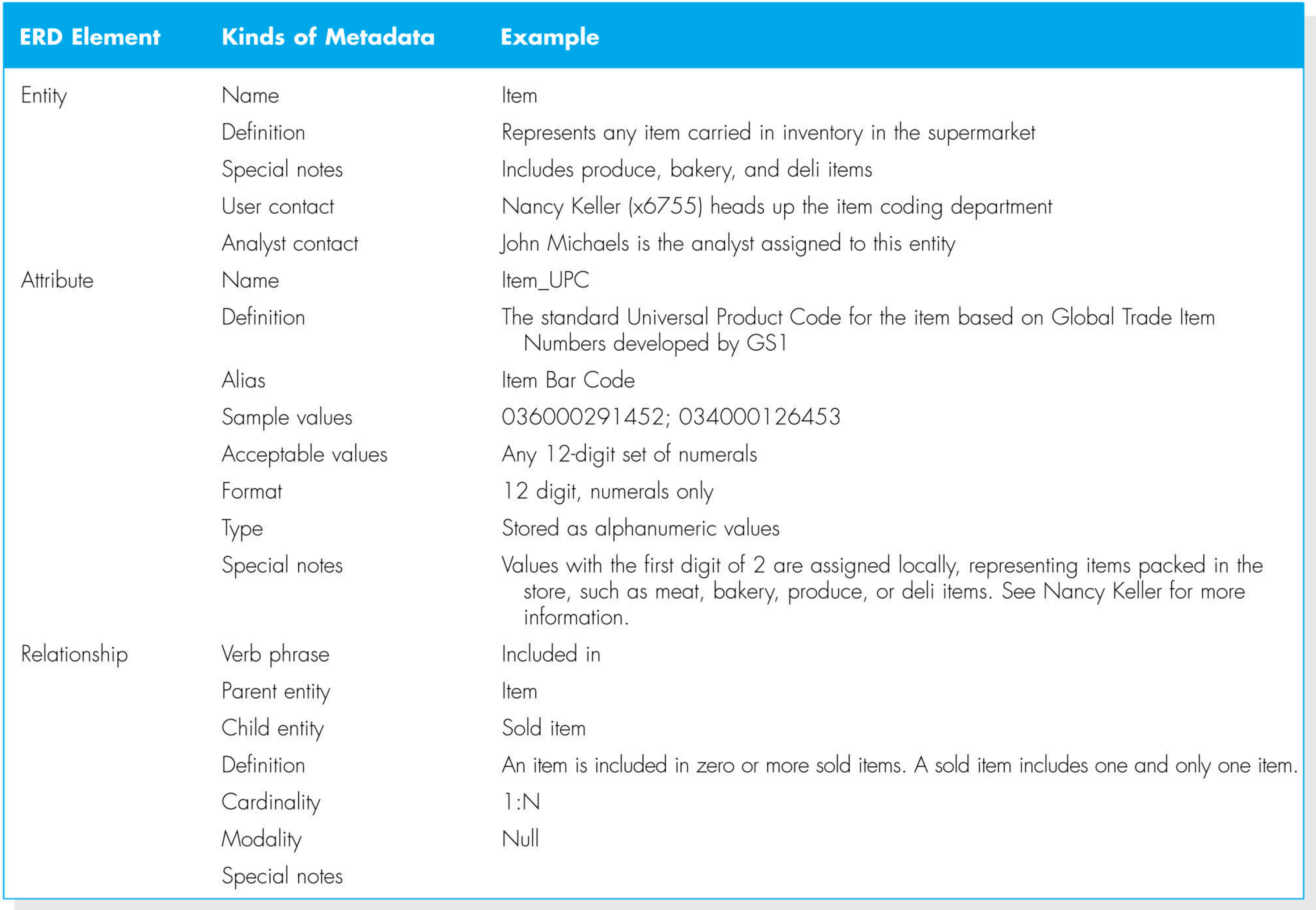
Verb phrase: Includes

Inverse phrase: Included in

Physical name:

Notes:

Figure 6-8 Data Dictionary Entry for a Relationship (in Visio 2010)



ERD Element	Kinds of Metadata	Example
Entity	Name	Item
	Definition	Represents any item carried in inventory in the supermarket
	Special notes	Includes produce, bakery, and deli items
	User contact	Nancy Keller (x6755) heads up the item coding department
	Analyst contact	John Michaels is the analyst assigned to this entity
Attribute	Name	Item_UPC
	Definition	The standard Universal Product Code for the item based on Global Trade Item Numbers developed by GS1
	Alias	Item Bar Code
	Sample values	036000291452; 034000126453
	Acceptable values	Any 12-digit set of numerals
	Format	12 digit, numerals only
	Type	Stored as alphanumeric values
	Special notes	Values with the first digit of 2 are assigned locally, representing items packed in the store, such as meat, bakery, produce, or deli items. See Nancy Keller for more information.
Relationship	Verb phrase	Included in
	Parent entity	Item
	Child entity	Sold item
	Definition	An item is included in zero or more sold items. A sold item includes one and only one item.
	Cardinality	1:N
	Modality	Null
	Special notes	

CREATING AN ENTITY RELATIONSHIP DIAGRAM (ERD)

- Drawing the ERD is an iterative process of trial and revision.
- The basic steps in building an ERD:

1. Identify the entities;
2. add the appropriate attributes to each entity; and
3. draw relationships among entities.

Step 1: Identify the Entities

- The entities should represent the major categories of information that you need to store in your system.
- If you begin the data model using a use case, look at the major inputs and outputs of the use case.
- If the process models are available, look at the data stores, external entities, and data flows.

Step 2: Add Attributes and Assign Identifiers

- The information that describes each entity becomes its attributes.
  - Check in the CASE repository of the process model for details on data flows and data stores.
  - Check the data requirements of the requirements definition.
  - Use requirements elicitation techniques (e.g., interview and document analysis).
- One or more of the attributes will become the entity's identifier.

Step3: Identify Relationships

- The last step in creating ERDs is to determine how the entities are related to each other.
- Lines are drawn between the entities that have **relationships**.
- Each relationship is labeled, and cardinality and modality are assigned.

Advanced Syntax

- Three special types of entities:
- ***Independent Entity***
  - Can exist without the help of another entity.
  - The identifiers is created from the entity’s own attributes.
  - Attributes from other entities are not needed to uniquely identify instances of these entities.

(cont'd)

- ***Dependent Entity***
  - There are situations when a child entity does require attributes from the parent entity to uniquely identify an instance. In these cases, the child entity is called a dependent entity, and its identifier consists of at least one attributes from the parent entity.

    (E.g., the Chemical Request entity in Figure 6.1).

(cont'd)

- ***Intersection Entity***
  - It exists in order to capture some information about the relationship between two other entities. Typically, intersection entities are added to a data model to store information about two entities sharing an M : N relationship.

(cont’d)

- There are three steps involved in adding an intersection entity:

Step 1. Remove the M:N relationship line and insert a new entity (intersection entity) in between the two existing ones.

Step 2. Add two 1:N relationships to the model.

Step 3. Name the intersection entity.

(cont'd)

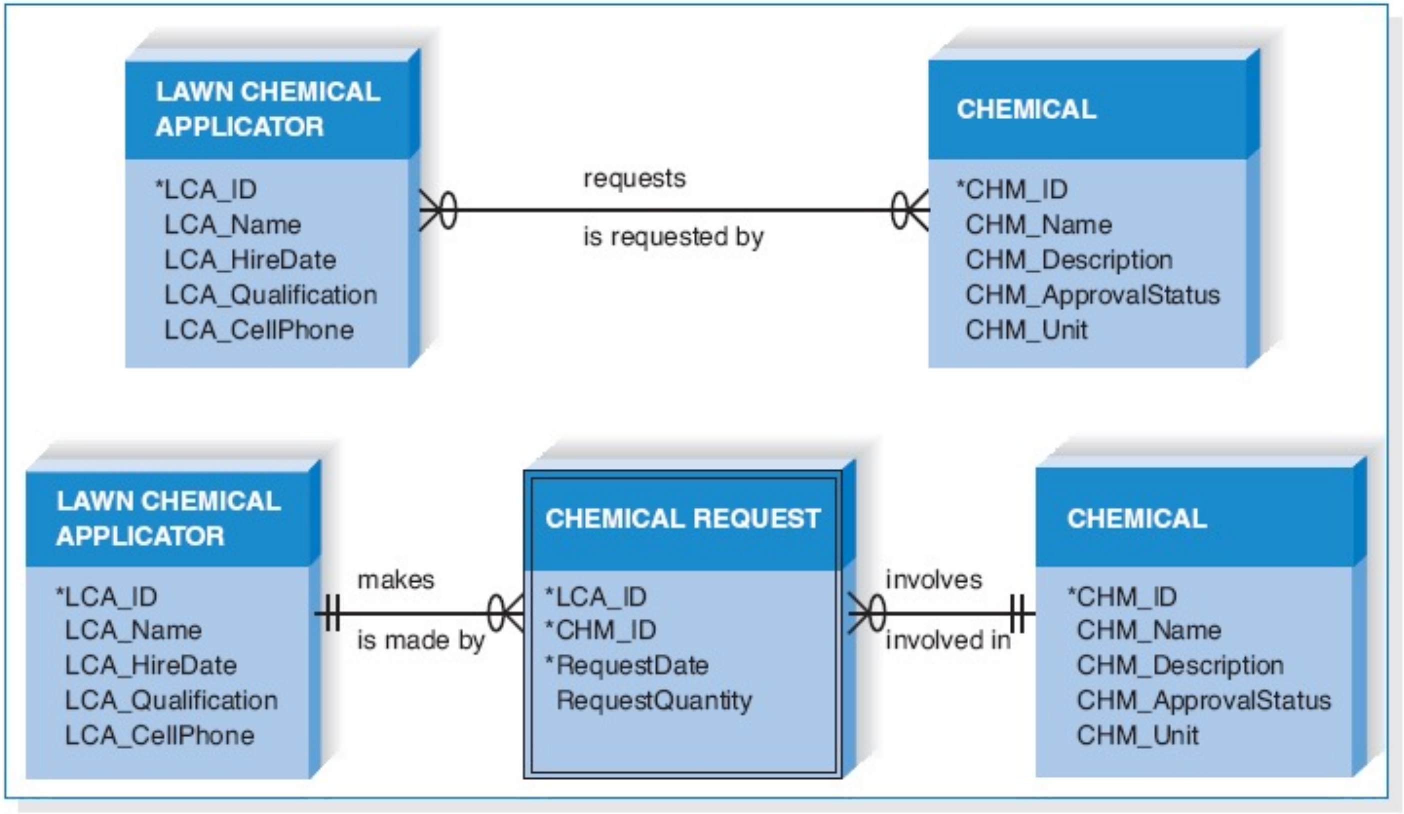
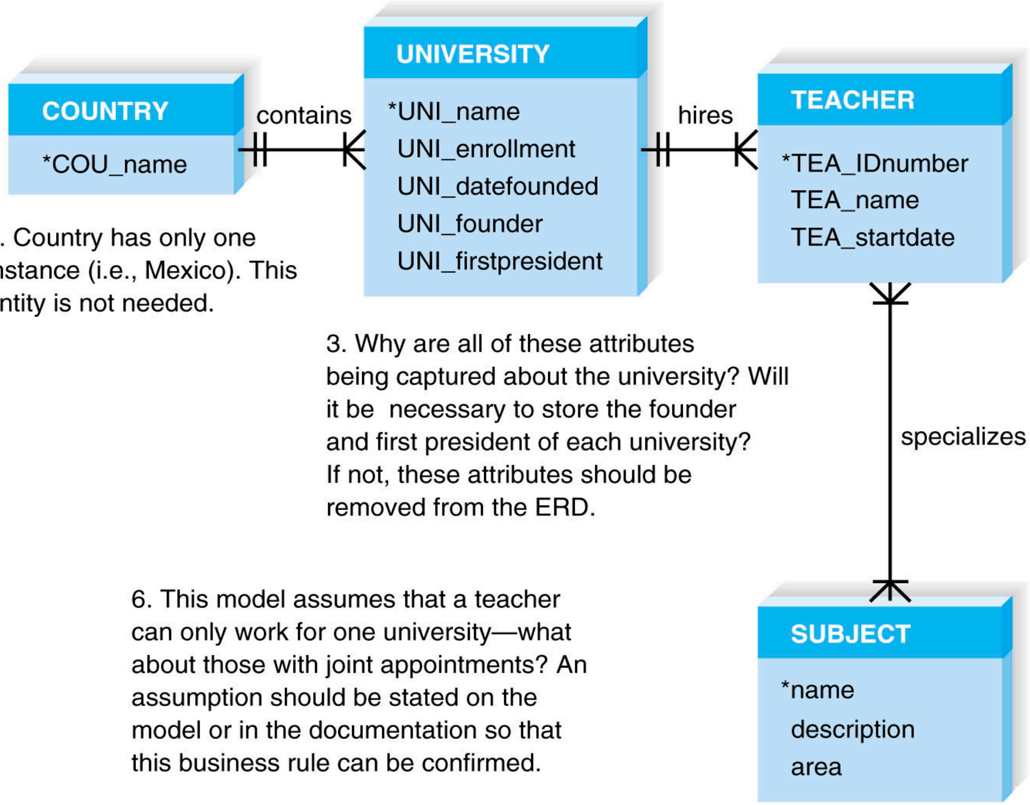


FIGURE 6-11
Resolving an M:N Relationship

VALIDATING AN ERD

- General design guidelines.
- **Normalization**.
- Check the ERD against the process models to make sure that both model balance each other.

Design Guidelines



1. Country has only one instance (i.e., Mexico). This entity is not needed.

. If teachers are called “ rofessors, ” then the ERD should contain an entity called “ rofessor,” to remain consistent.

3. Why are all of these attributes being captured about the university? Will it be necessary to store the founder and first president of each university? If not, these attributes should be removed from the ERD.

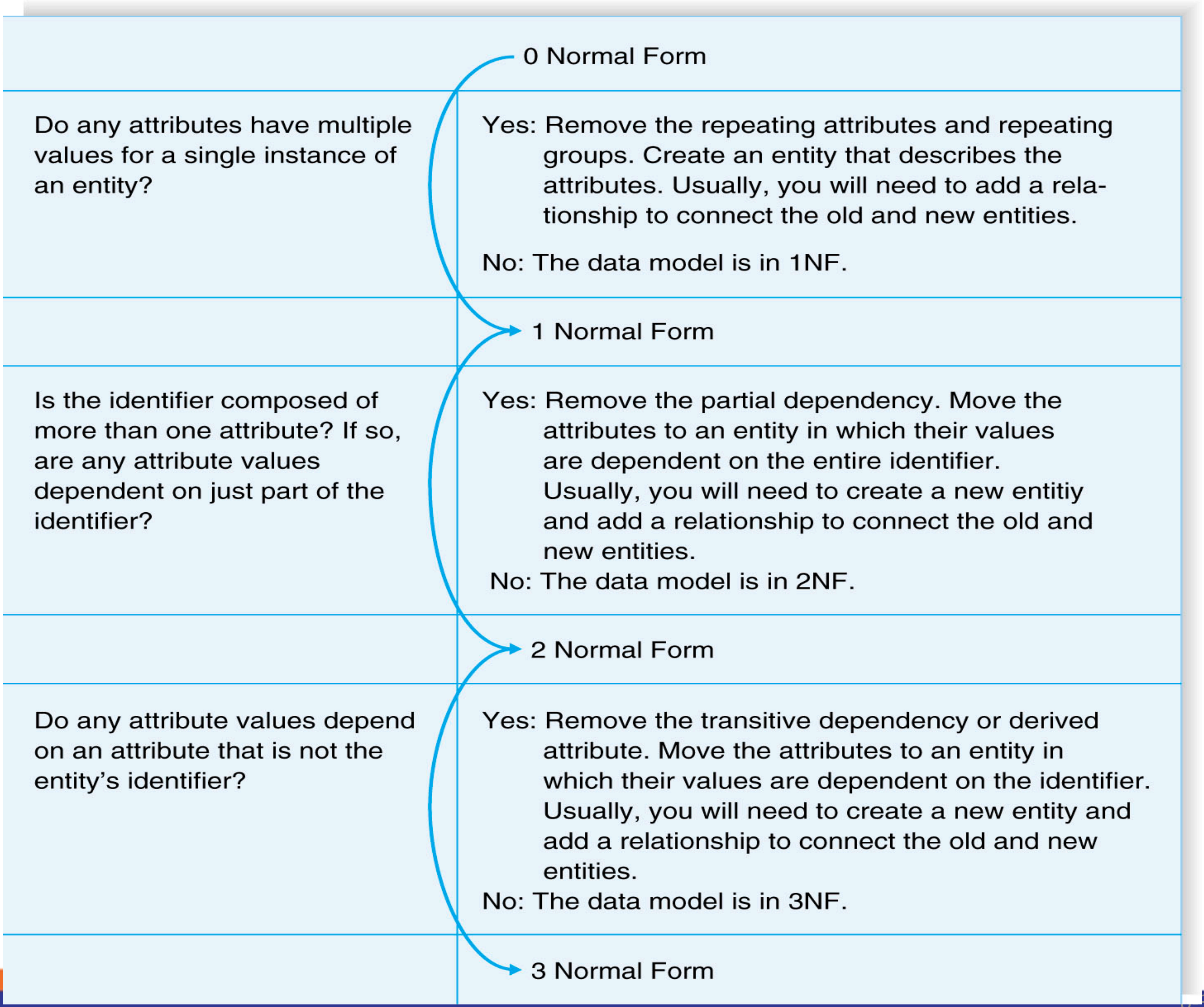
4. The attributes in the subject entity are poorly labeled. For one, we have no way of knowing to which entity they belong if they stood alone—it would be helpful to begin each attribute with SUB_. Also, what is “area”? A term like “department” or “field of research” would be more descriptive.

5. The name attribute really should be broken down into last name and first name—otherwise, there would be no way to manipulate names in the system. For example, there would be no way to sort by last name if it were combined with the first name.

6. This model assumes that a teacher can only work for one university—what about those with joint appointments? An assumption should be stated on the model or in the documentation so that this business rule can be confirmed.

Normalization

- ***Normalization*** is a technique that can help analysts validate the data models.
- It is a process whereby a series of rules are applied to a logical data model to determine how well formed it is.



	0 Normal Form
Do any attributes have multiple values for a single instance of an entity?	Yes: Remove the repeating attributes and repeating groups. Create an entity that describes the attributes. Usually, you will need to add a relationship to connect the old and new entities. No: The data model is in 1NF.
	1 Normal Form
Is the identifier composed of more than one attribute? If so, are any attribute values dependent on just part of the identifier?	Yes: Remove the partial dependency. Move the attributes to an entity in which their values are dependent on the entire identifier. Usually, you will need to create a new entitiy and add a relationship to connect the old and new entities. No: The data model is in 2NF.
	2 Normal Form
Do any attribute values depend on an attribute that is not the entity's identifier?	Yes: Remove the transitive dependency or derived attribute. Move the attributes to an entity in which their values are dependent on the identifier. Usually, you will need to create a new entity and add a relationship to connect the old and new entities. No: The data model is in 3NF.
	3 Normal Form

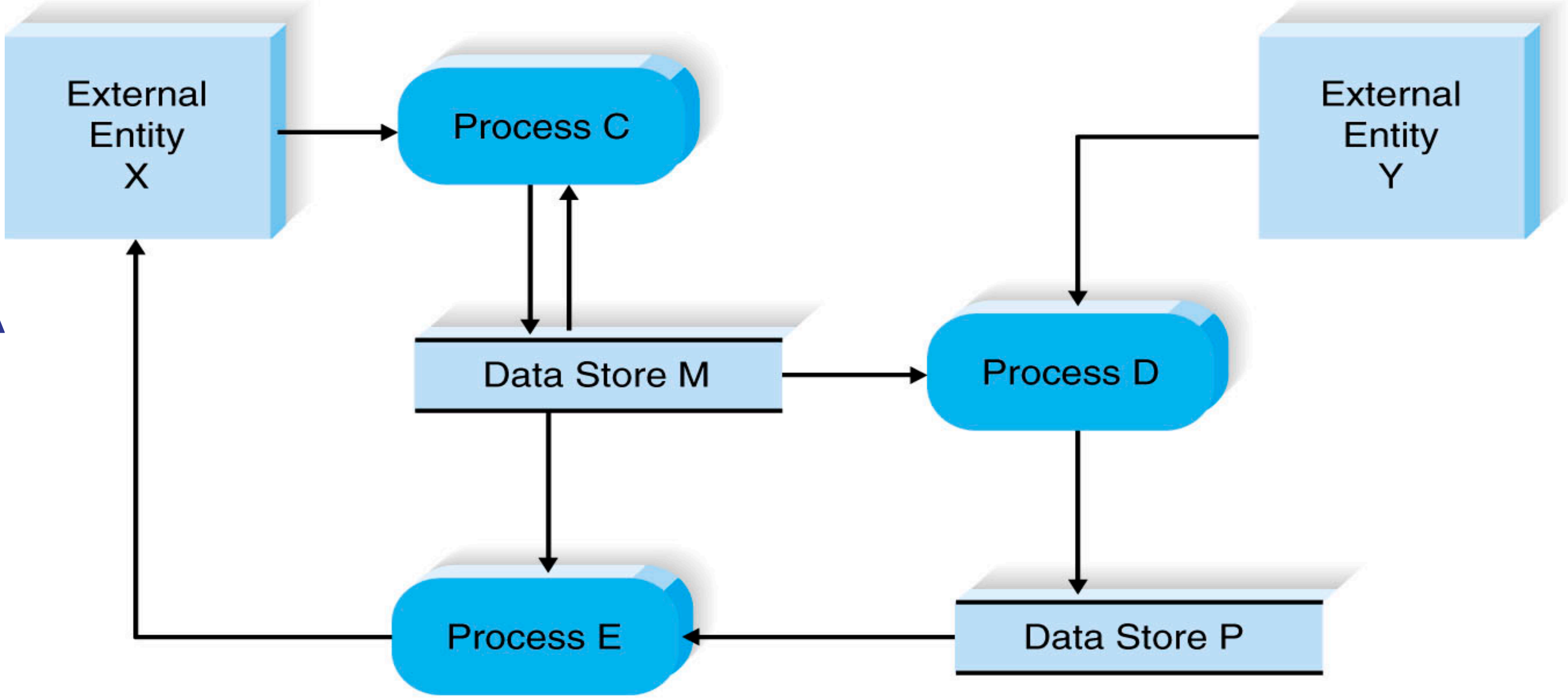
Balancing ERDs with DFDs

- The process models and data models are interrelated.
- Although the process model focuses on the business processes, it contains two data components – the data and the data store.
- These two data components of the DFDs need to ***balance*** with the ERDs.
- The DFD data components need to correspond with the ERD's data stores (i.e., entities) and the data elements that comprise the data flows (i.e., attributes) depicted on the data model.

(cont'd)

- Many CASE tools provide features of identifying problems with balance between DFDs and ERDs; however, it is important to understand how to identify problems on your own.
- Check your DFDs and ERDs to make sure all data components correspond between DFDs and ERDs.
- A useful tools to clearly depict the interrelationship between process and data models is the ***CRUD matrix*** (create, read, update, delete matrix).

A



	Process C	Process D	Process E
Data Entity M			
Attribute M-1	CRUD	R	R
Attribute M-2	CRUD		R
Attribute M-3	CRUD	R	
Attribute M-4	CRUD		R
Data Entity P			
Attribute P-1		C	R
Attribute P-2		C	
Attribute P-3		C	R

SUMMARY

- Basic Entity Relationship Diagram Syntax

  - Entity describes people, places, or things.

  - Attribute is some type of information about the entity.

  - Relationship conveys the associations between entities.

- Creating an ERD

  - Identify the entities.

  - Add the attributes to each entity.

  - Draw relationships among entities.

- Validating an ERD

  - Normalization

  - CRUD matrix