



**DAVID M. KROENKE and DAVID J. AUER**  
**DATABASE CONCEPTS, 4<sup>th</sup> Edition**



## Data Modeling and the Entity-Relationship Model

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems



## Chapter Objectives



- Learn the **basic stages** of database development, understand the purpose and role of a data model
- Know the **principal components** of the E-R data model and how to construct E-R diagrams
- Understand how to **interpret** traditional E-R diagrams
- Understand how to interpret **Information Engineering (IE) Crow's Foot** E-R diagrams
- Know how to **represent 1:1, 1:N, N:M**, and binary relationships with the E-R model

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-2



## Chapter Objectives (Continued)

- Understand two types of weak entities and know how to use them
- Understand non-identifying and identifying relationships and know how to use them
- Know how to represent subtype entities with the E-R model
- Know how to represent recursive relationships with the E-R model
- Learn how to create an E-R diagram from source documents



## Three Stages of Database Development

- Requirements Stage
- Design Stage
- Implementation Stage



## The Requirements Stage

- Sources of requirements:

1. User Interviews
2. Forms
3. Reports
4. Queries
5. Use Cases
6. Business Rules



## Requirements Become the E-R Data Model

- After the requirements have been gathered, they are transformed into an Entity Relationship (E-R) Data Model

- E-R Models consist of:

1. Entities
2. Attributes
3. Identifiers
4. Relationships



## Entity Class versus Entity Instance



- An **entity class** is a description of the structure and format of the occurrences of the entity
- An **entity instance** is a specific occurrence of an entity within an entity class



## Entity Class and Entity Instance



ITEM
ItemNumber
Description
Cost
ListPrice
QuantityOnHand

Entity Class

1100
100 amp panel
\$127.50
\$170.00
14

2000
Door handle set
\$52.50
\$39.38
0

Two Entity Instances



## Attributes

- Entities have **attributes** that describe the entity's characteristics:
  - **ProjectName**
  - **StartDate**
  - **ProjectType**
  - **ProjectDescription**
- Attributes have a data type and properties



## Identifiers

- Entity instances have **identifiers**
- An identifier will identify a particular instance in the entity class:
  - **SocialSecurityNumber**
  - **StudentID**
  - **EmployeeID**

2000
Door handle set
\$52.50
\$39.38
0



## Identifier Types

- **Uniqueness:**

- Identifiers may be unique or non-unique
- If the identifier is unique, the data value for the identifier must be unique for all instances

2000
Door handle set
\$52.50
\$39.38
0

- **Composite:**

- A composite identifier consists of 2 or more attributes
  - ✓ e.g., OrderNumber & LineItemNumber are both required



## Level of Entity Attribute Display

**ITEM**

ItemNumber
Description
Cost
ListPrice
QuantityOnHand

(a) Entity with All Attributes

**ITEM**

ItemNumber
------------

(b) Entity with Identifier Attributes

**ITEM**

ITEM
------

(c) Entity with No Attributes

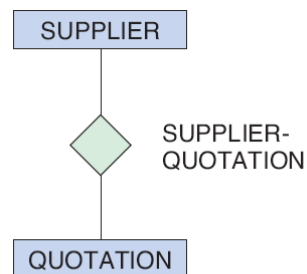


## Relationships

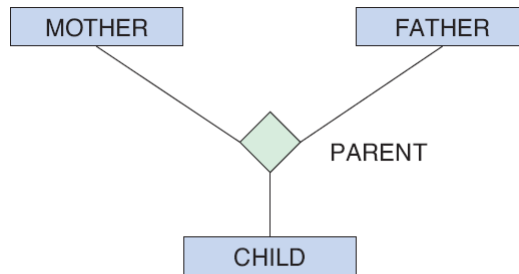
- Entities can be associated with one another in **relationships**
- Relationship *degree* defines the number of entity classes participating in the relationship:
  - Degree 2 is a binary relationship
  - Degree 3 is a ternary relationship



## Degree 2 Relationship: Binary



## Degree 3 Relationship: Ternary



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-15

## One-to-One Binary Relationship

- 1:1 (one-to-one)

➤ A single entity instance in one entity class is related to a single entity instance in another entity class:

- ✓ An employee may have no more than one locker; and
- ✓ A locker may only be accessible by one employee



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-16

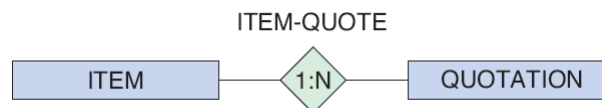


## One-to-Many Binary Relationship

- 1:N (one-to-many)

➤ A single entity instance in one entity class is related to many entity instances in another entity class:

- ✓ A quotation is associated with only one item; and
- ✓ An item may have several quotations

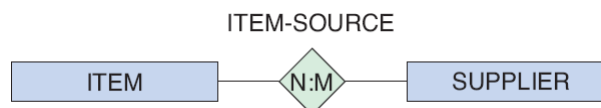


## Many-to-Many Binary Relationship

- N:M (many-to-many)

➤ Many entity instances in one entity class is related to many entity instances in another entity class:

- ✓ A supplier may supply several items; and
- ✓ A particular item may be supplied by several suppliers





## Maximum Cardinality

- Relationships are named and classified by their **cardinality**, which is a word that means *count*.
- Each of the three types of binary relationships shown above have different *maximum cardinalities*.
- **Maximum cardinality** is the maximum number of entity instances that may participate in a relationship instance—*one, many or some other fixed number*.



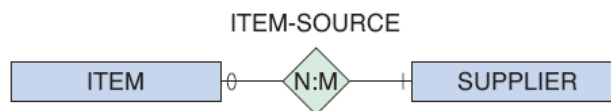
## Minimum Cardinality

- **Minimum cardinality** is the minimum number of entity instances that *must participate* in a relationship instance.
- These values typically assume a value of *zero (optional)* or *one (mandatory)*

## Cardinality Example



- Maximum cardinality is *many* for both ITEM and SUPPLIER
- Minimum cardinality is *zero* (optional) for ITEM and *one* (mandatory) SUPPLIER
  - A SUPPLIER does not have to supply an ITEM
  - An ITEM must have a SUPPLIER



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-21

## Entity-Relationship Diagrams



- The diagrams in previous slides are called entity-relationship diagrams:
  - Entity classes are shown by rectangles
  - Relationships are shown by diamonds
  - The maximum cardinality of the relationship is shown inside the diamond
  - The minimum cardinality is shown by the oval or hash mark next to the entity
  - The name of the entity is shown inside the rectangle
  - The name of the relationship is shown near the diamond

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-22



## HAS-A Relationships

- The relationships in the previous slides are called **HAS-A relationships**.
- The term is used because each entity instance *has a relationship to a second entity instance*:
  - An employee has a badge
  - A badge has an employee







## Types of Entity-Relationship Diagrams

- **Information Engineering (IE)** [James Martin 1990] - Uses "crow's feet" to show the many side of a relationship, and it is sometimes called the crow's foot model.
- **Integrated Definition 1, Extended 3 (IDEF1X)** is a version of the E-R model that is a national standard.
- **Unified Modeling Language (UML)** is a set of structures and techniques for modeling and designing object-oriented programs (OOP) and applications.



## Crow's Foot Symbols



Symbol	Meaning
	One—Mandatory
	Many—Mandatory
	One—Optional
	Many—Optional

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

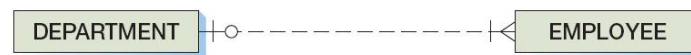
4-25



## Crow's Foot Example: One-to-Many Relationship



(a) Original E-R Model Version



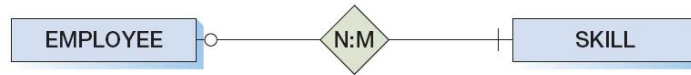
(b) Crow's Foot Version

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-26



## Crow's Foot Example: Many-to-Many Relationship



(a) Original E-R Model Version



(b) Crow's Foot Version



## Weak Entity

- A **weak entity** is an entity that **cannot exist** in the database without the existence of another entity.
- Not surprisingly any entity that is not a weak entity is called a **strong entity**.

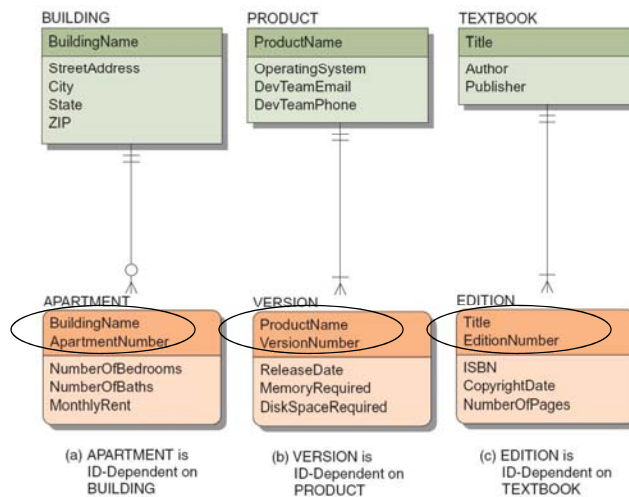


## ID-Dependent Weak Entities

- An ID-Dependent weak entity is a weak entity that cannot exist without its parent entity.
- An ID-dependent weak entity has a composite identifier:
  - The first part of the identifier is the identifier for the strong entity
  - The second part of the identifier is the identifier for the weak entity itself



## ID-Dependent Weak Entity Examples





## Weak Entity Relationships

- The relationship between a strong and weak entity is termed an **identifying relationship** if the weak entity is ID-dependent:
  - Represented by a **solid line**
- The relationship between a strong and weak entity is termed a **non-identifying relationship** if the weak entity is non-ID-dependent:
  - Represented by a **dashed line**
  - Also used between strong entities

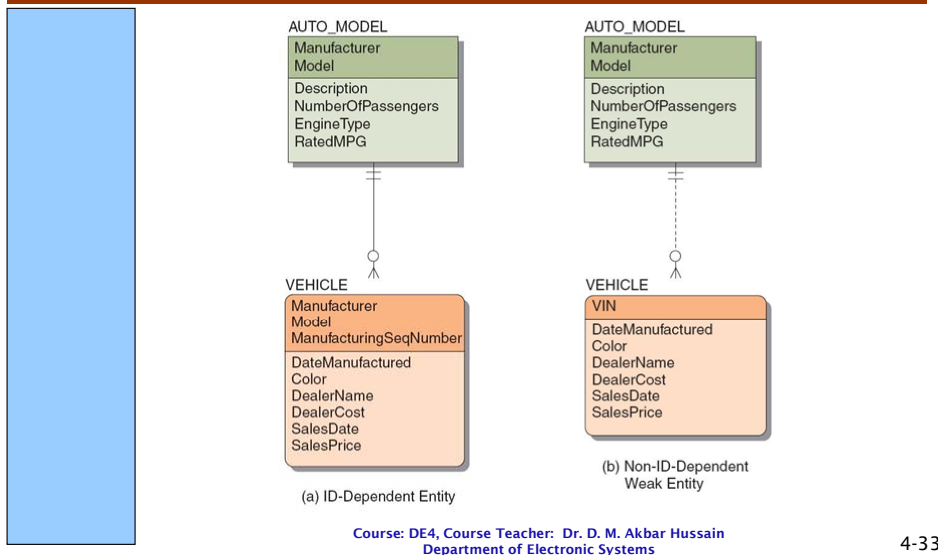


## Weak Entity Identifier: Non-ID-dependent

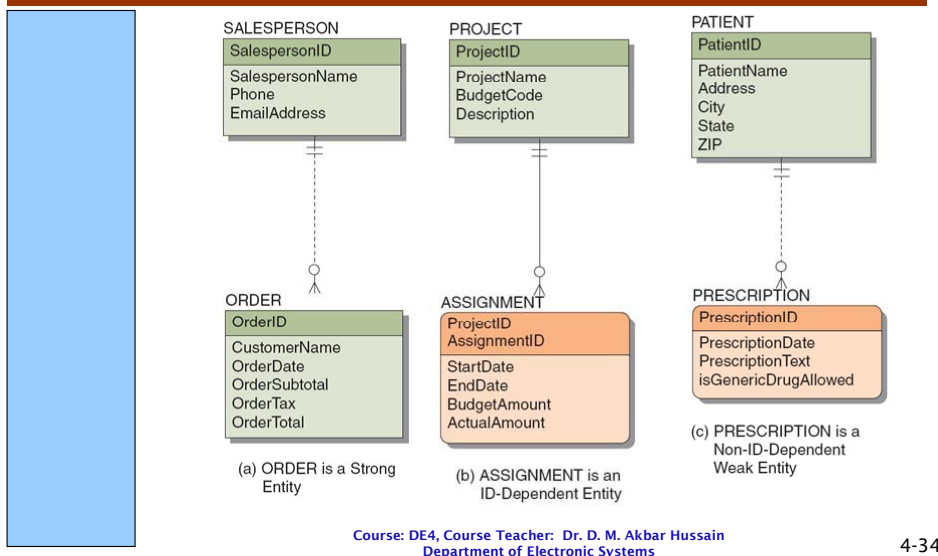
- All ID-dependent entities are weak entities, but there are other entities that are weak but **not ID-dependent**.
- A **non-ID-dependent weak entity** may have a single or composite identifier, but the identifier of the parent entity will be a **foreign key**.



## Non-ID-Dependent Weak Entity Examples



## Strong and Weak Entity Examples



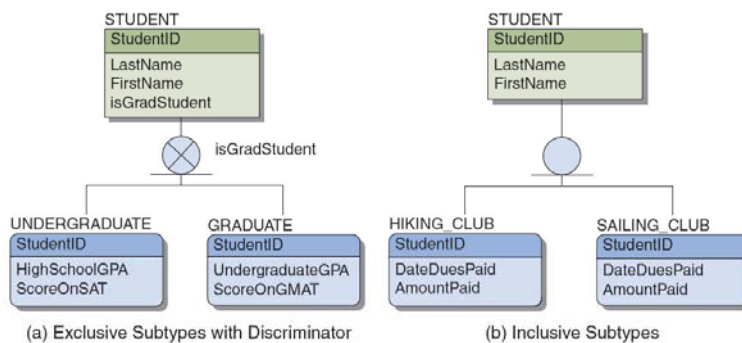


## Subtype Entities

- A subtype entity is a special case of another entity called its **supertype**.
- An attribute of the supertype may be included that indicates which of the subtypes is appropriate for a given instance — This attribute is called a **discriminator**.
- Subtypes can be **exclusive** or **inclusive**:
  - If **exclusive**, the supertype relates to at most one subtype
  - If **inclusive**, the supertype can relate to one or more subtypes



## Subtype Entity Examples





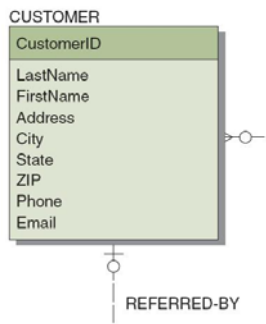
## Subtype Entity Identifiers

- The relationships that connect supertypes and subtypes are called **IS-A relationships** because a subtype is the same entity as the supertype.
- The identifier of a supertype and all of its subtypes is the same attribute.



## Recursive Relationships

- It is possible for an entity to have a relationship to itself—this is called a **recursive relationship**





## Developing an E-R Diagram

- Heather Sweeney Designs will be used as an ongoing example:
  - Heather Sweeney is an interior designer who specializes in home kitchen design.
  - She offers a variety of free seminars at home shows, kitchen and appliance stores, and other public locations.
  - She earns revenue by selling books and videos that instruct people on kitchen design.
  - She also offers custom-design consulting services



## Heather Sweeney Designs: The Seminar Customer List

*Heather Sweeney Designs  
Seminar Customer List*

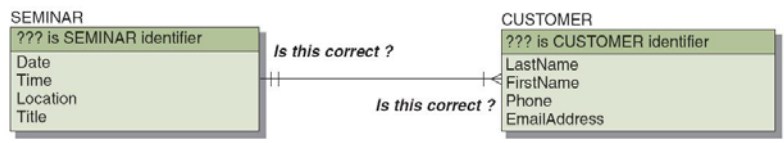
Date: October 11, 2008                      Location: San Antonio Convention Center

Time: 11 AM                                      Title: Kitchen on a Budget

Name	Phone	Email Address
Nancy Jacobs	817-871-8123	NJ@somewhere.com
Chantel Jacobs	817-871-8234	CJ@somewhere.com
Ralph Able	210-281-7687	RA@somewhere.com
Etc.		
27 names in all		



## Heather Sweeney Designs: Initial E-R Diagram I

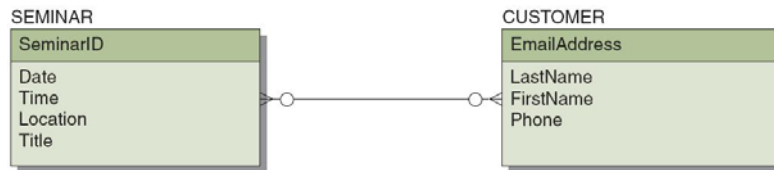


## Heather Sweeney Designs: Initial E-R Diagram II





## Heather Sweeney Designs: Initial E-R Diagram III



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-43



## Heather Sweeney Designs: The Customer Form Letter

Heather Sweeney Designs  
122450 Flockaway Road  
Dallas, Texas 75227  
972-233-6165

Ms. Nancy Jacobs  
1400 West Palm Drive  
Fort Worth, Texas 76110

Dear Ms. Jacobs:

Thank you for attending my seminar "Kitchen on a Budget" at the San Antonio Convention Center. I hope that you found the seminar topic interesting and helpful for your design projects.

As a seminar attendee, you are entitled to a 15 percent discount on all of my video and book products. I am enclosing a product catalog and I would also like to invite you to visit our Web site at [www.hussaindesigns.com](http://www.hussaindesigns.com).

Also, as I mentioned at the seminar, I do provide customized design services to help you create that just-perfect kitchen. In fact, I have a number of clients in the Fort Worth area. Just give me a call at my personal phone number of 555-122-4873 if you'd like to schedule an appointment.

Thanks again and I look forward to hearing from you!

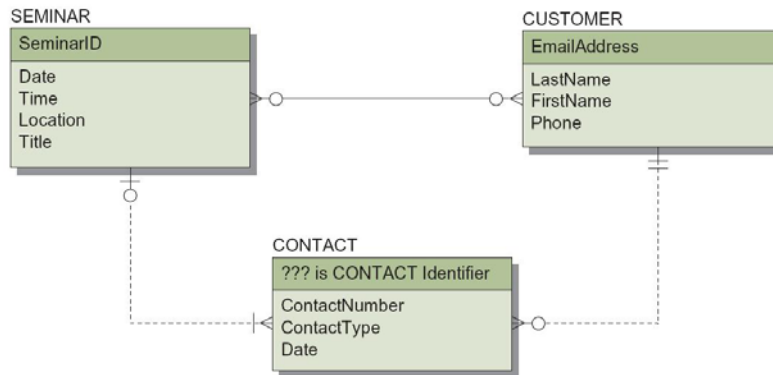
Best regards,

Heather Sweeney

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-44

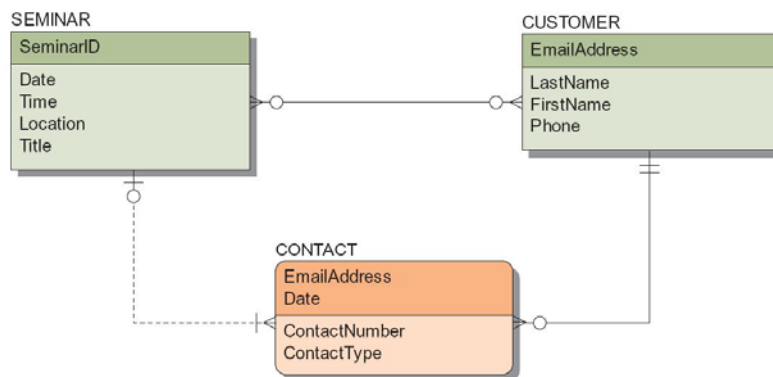
## Heather Sweeney Designs: Data Model with CONTACT



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-45

## Heather Sweeney Designs: Data Model with CONTACT as Weak Entity

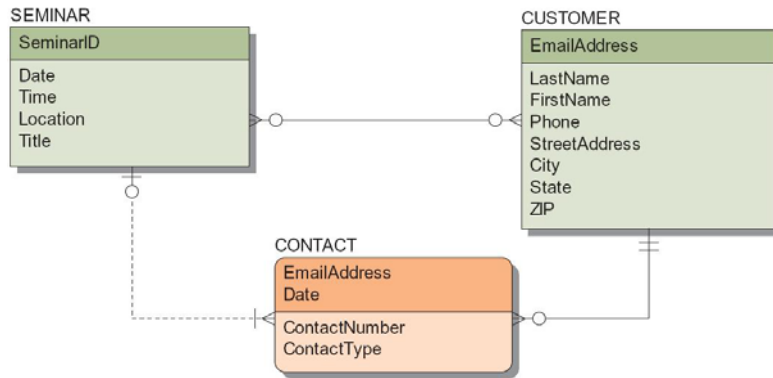


Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-46



## Heather Sweeney Designs: Data Model with Modified CUSTOMER



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-47



## Heather Sweeney Designs: Sales Invoice

Heather Sweeney Designs  
12240 Rockaway Road  
Dallas, Texas 75227

Invoice No. 35000

---

INVOICE

<b>Customer:</b>		<b>Misc:</b>	
Name: Ralph Aida	Address: 123 Elm Street	Date: 10/15/08	Order No.:
City: San Antonio	State: TX    ZIP: 78214	Rep:	FOB:
Phone: 210-281-7987			

Qty	Description	Unit Price	TOTAL
1	Kitchen Remodeling Basics - Video	\$ 14.95	\$ 14.95
1	Kitchen Remodeling Basics - Video Companion	\$ 7.99	\$ 7.99

Subtotal	\$ 22.94
Shipping	\$ 5.95
Tax Rate(s): 5.75%	\$ 1.31
<b>TOTAL</b>	<b>\$ 30.20</b>

**Payment:**  Credit

Comments: Visa  
Name: Ralph J. Aida  
CC #: xxxx xxx xxxxx  
Expire: May-11

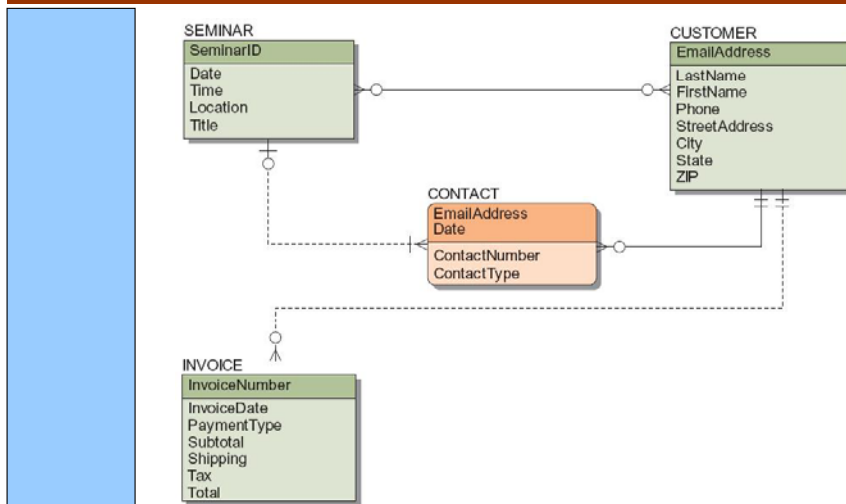
Office Use Only

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-48



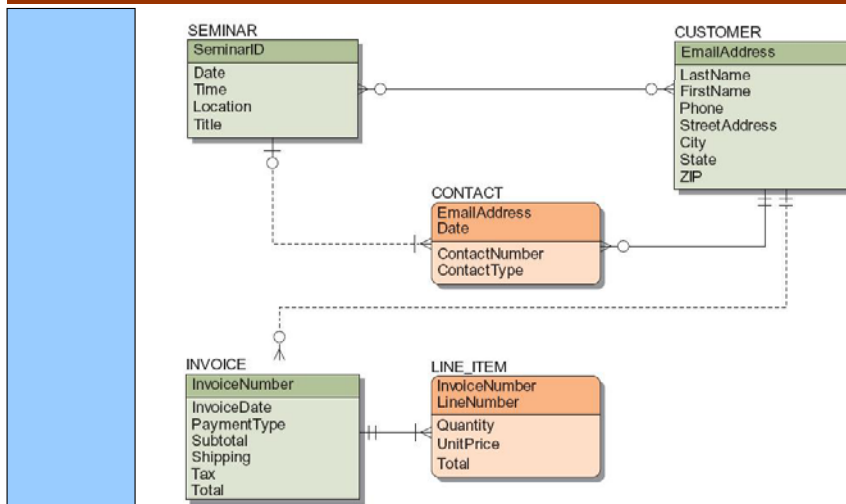
## Heather Sweeney Designs: Data Model with INVOICE



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-49

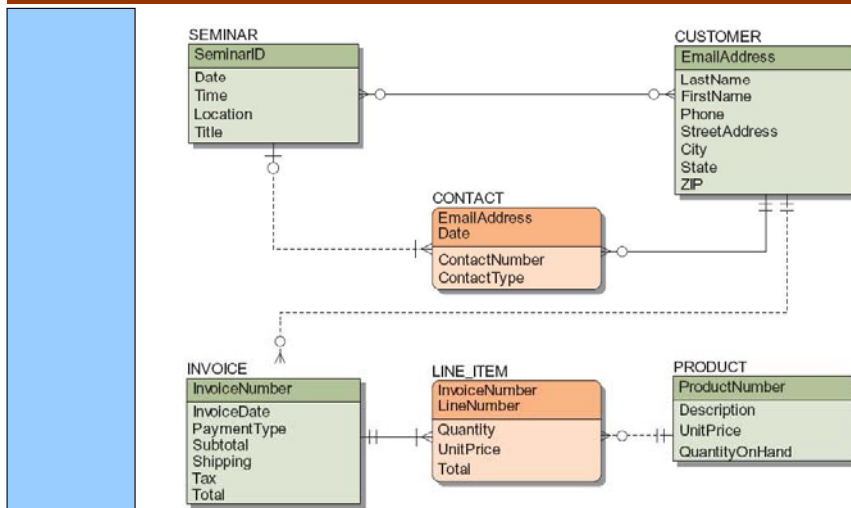
## Heather Sweeney Designs: Data Model with LINE\_ITEM



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-50

## Heather Sweeney Designs: Final Data Model



Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-51

## Heather Sweeney Designs: Attribute Specifications

- Attribute specifications must be created for each entity.
- The attribute specifications for **SEMINAR** are shown as an example.

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
SeminarID	AutoNumber	Primary Key	Yes	DBMS supplied	Surrogate Key; Initial value=1 Increment=1
Date	Date	No	Yes	None	As: mm/dd/yy
Time	Time	No	Yes	None	As: ##:##(AM/PM)
Location	Varchar (100)	No	Yes	None	
Title	Varchar (100)	No	Yes	None	

Course: DE4, Course Teacher: Dr. D. M. Akbar Hussain  
Department of Electronic Systems

4-52



## Heather Sweeney Designs: Business Rules and Model Validation

- Business rules may constrain the model and need to be recorded:
  - Heather Sweeney Designs has a business rule that no more than one form letter or email per day is to be sent to a customer
- After the data model has been completed, it needs to be validated:
  - Prototyping is commonly used to validate forms and reports