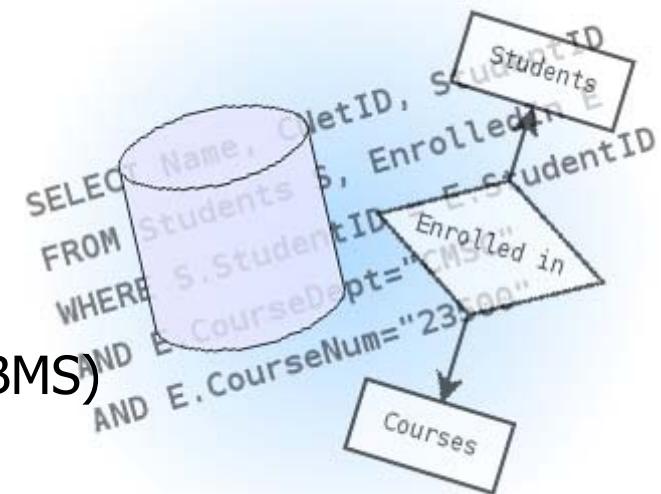


Database Management Systems

Sumayyea Salahuddin (Lecturer)
Dept. of Computer Systems Eng.
UET Peshawar

Overview

- Introduction to the Database & DBMS
 - Database Definitions
 - Advantages of Database Approach
 - Functions of Database Management Systems (DBMS)
 - Case Study: Pine Valley Furniture Company
 - The Database Lifecycle (DBLC)
 - Front-end & Back-end Databases



Definitions

- **Data:**

- Meaningful facts, text, graphics, images, sound, video segments
- Usually in certain context e.g. Class Roster

Class Roster			
Course:	MGT 500 Business Policy	Semester:	Spring 2010
Section:	2		
Name	ID	Major	GPA
Baker, Kenneth D.	324917628	MGT	2.9
Doyle, Joan E.	476193248	MKT	3.4
Finkle, Clive R.	548429344	PRM	2.8
Lewis, John C.	551742186	MGT	3.7
McFerran, Debra R.	409723145	IS	2.9
Sisneros, Michael	392416582	ACCT	3.3

Figure 1-1a: Data in Context

Definitions (Cont.)

- **Information:**

- Data processed to be useful in decision making and interpretation

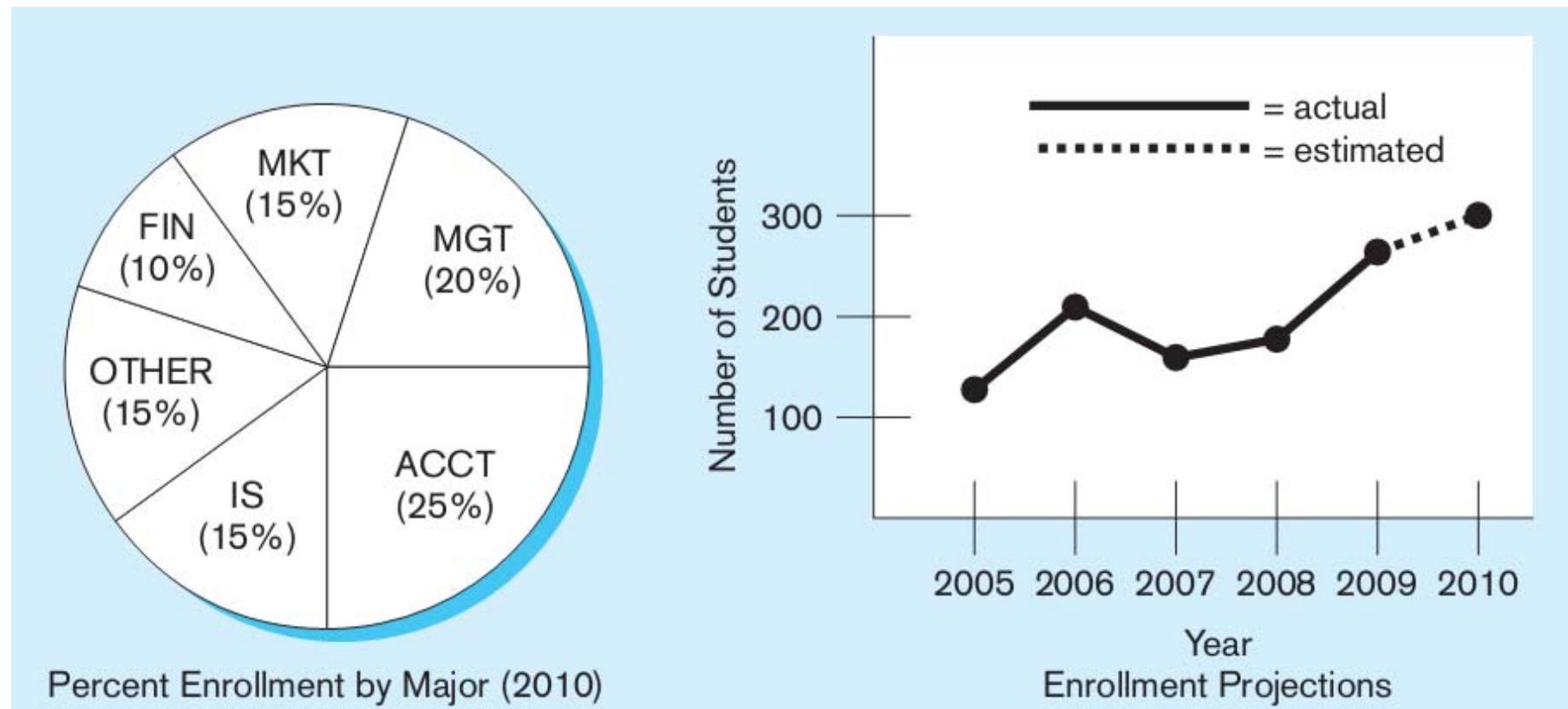
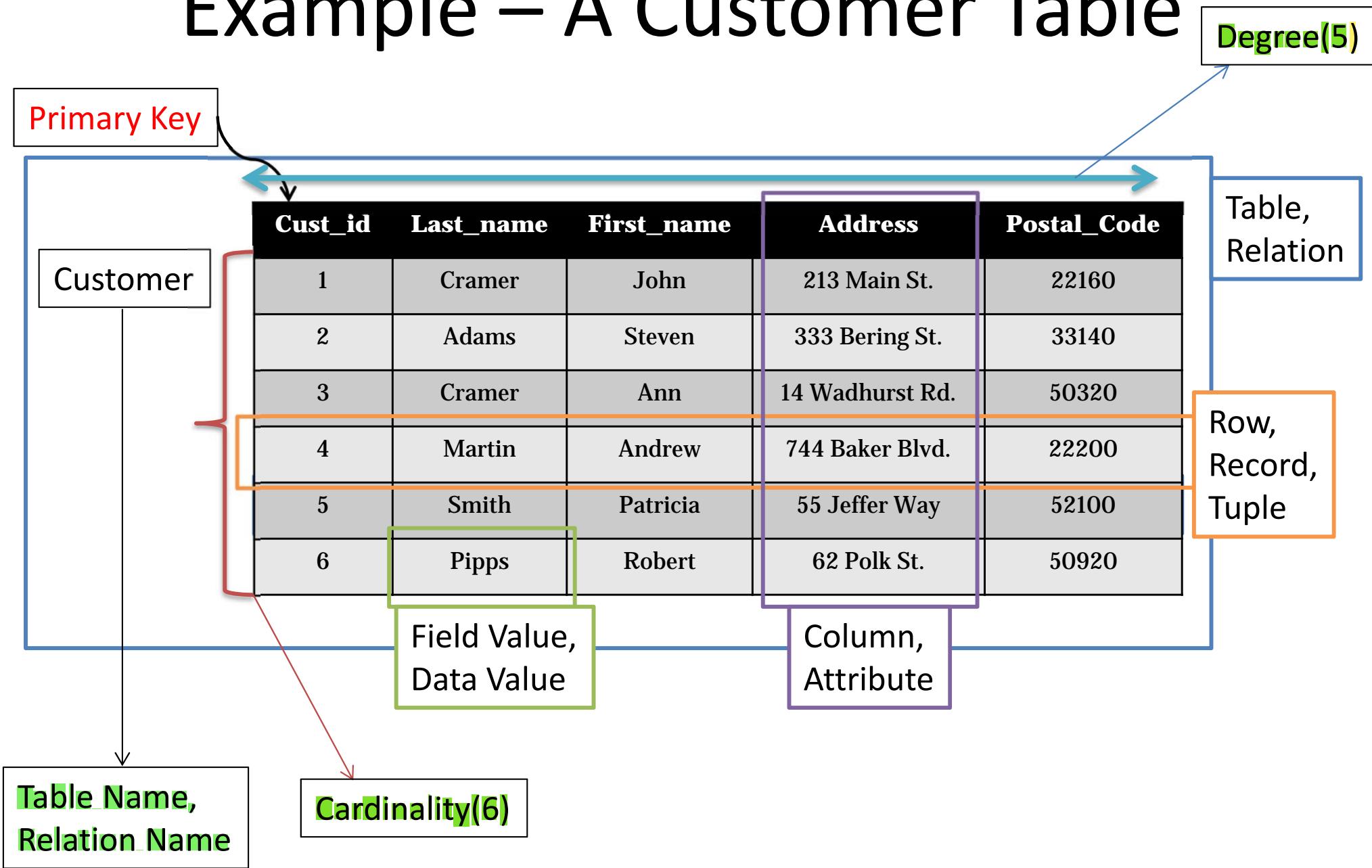


Figure 1-1b: Summarized Data/Information

Database & Its Terminologies

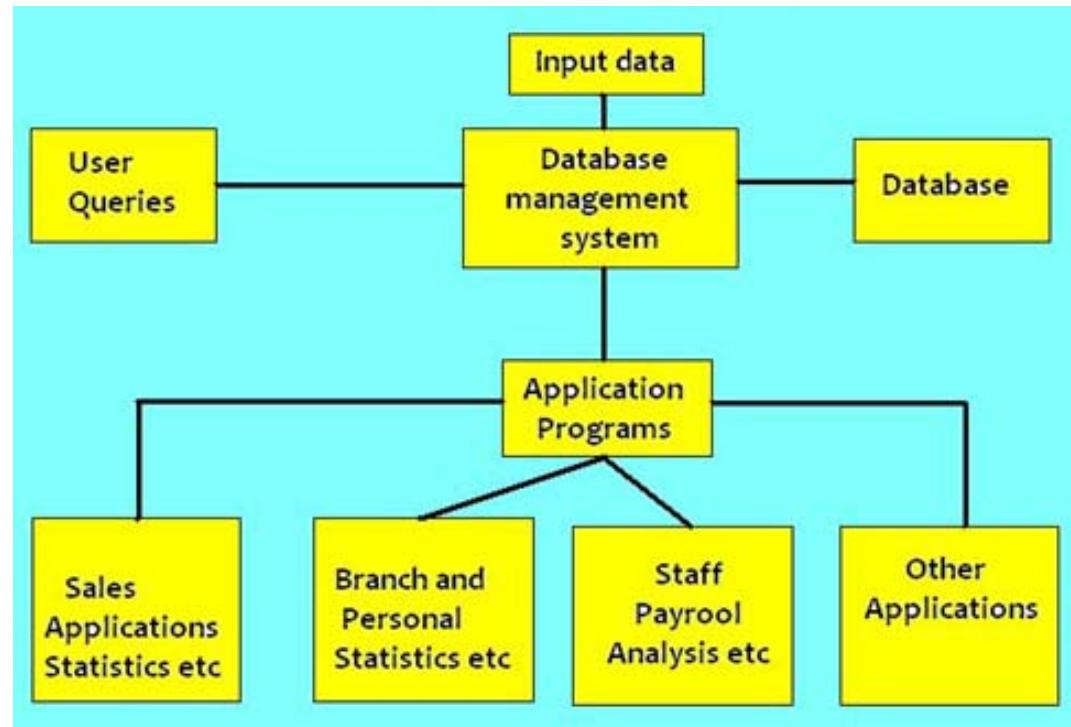
- **Database:** An organized collection of logically related data
- **Field (data item):**
 - Smallest unit of information in database
 - It is collection of related characters (numeric or alphabetic) that define a characteristic of an entity (person, place or thing)
- **Record:**
 - Fields are grouped together to form records.
 - It is collection of related (logically connected) fields
- **File:** collection of similar types of records

Example – A Customer Table



Definitions (Cont.)

- **Database Management System (DBMS):** a system software that facilitates the management of a database and controls the access to the data stored in the database



Definitions (Cont.)

- **Database System**: system consisting of a database, a DBMS, hardware and people
- **Database Design**: The design of the database structure that will be used to store and manage data (not the design of the DBMS)
- **Data Dictionary (DD) or Metadata**: describes the characteristics of data stored in a database and the inter-relationships among data

Table 1-1: Metadata

Description of the properties or characteristics of the data, including data types, field sizes, allowable values, and documentation

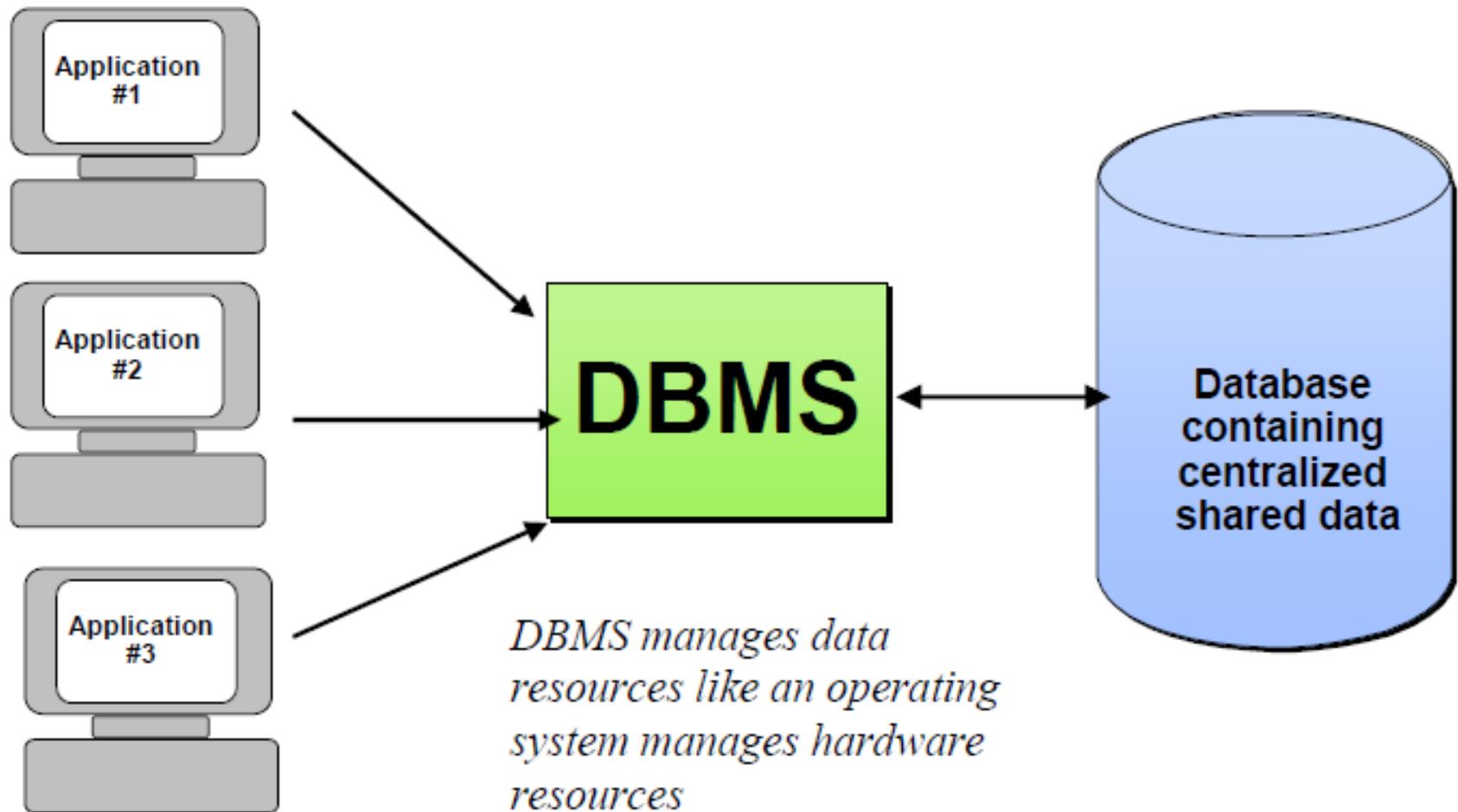
TABLE 1-1 Example Metadata for Class Roster

Data Item		Metadata				
Name	Type	Length	Min	Max	Description	Source
Course	Alphanumeric	30			Course ID and name	Academic Unit
Section	Integer	1	1	9	Section number	Registrar
Semester	Alphanumeric	10			Semester and year	Registrar
Name	Alphanumeric	30			Student name	Student IS
ID	Integer	9			Student ID (SSN)	Student IS
Major	Alphanumeric	4			Student major	Student IS
GPA	Decimal	3	0.0	4.0	Student grade point average	Academic Unit

Database Management System

- A DBMS is a data storage and retrieval system which permits data to be stored non-redundantly while making it appear to the user as if the data is well-integrated.

Database Management System



Advantages of Database Approach

- Program-Data Independence
 - Metadata stored in DBMS, so applications don't need to worry about data formats
 - Data queries/updates managed by DBMS so programs don't need to process data access routines
 - Results in: increased application development and maintenance productivity
- Minimal Data Redundancy
 - Leads to increased data integrity/consistency
- Improved Data Sharing
 - Different users get different views of the data
- Enforcement of Standards
 - All data access is done in same way
- Improved Data Quality
 - Constraints, data validation rules

Advantages of Database Approach (Cont.)

- Better Data Accessibility/Responsiveness
 - Use of standard data query language (SQL)
- Security, Backup/Recovery, Concurrency
 - Disaster recovery is easy

The Database Approach

- The enterprise data model is a graphical model that shows the high-level entities for the organization and the relationships among those entities.
- Enterprise data model is represented with the help of a E-R diagram that shows the entities, attributes, relations, connectivities and cardinalities.

Entities & Business Rules

CUSTOMER A person or an organization that buys or may potentially buy products from Pine Valley Furniture

ORDER The purchase of one or more products by a customer

PRODUCT The items Pine Valley Furniture makes and sells

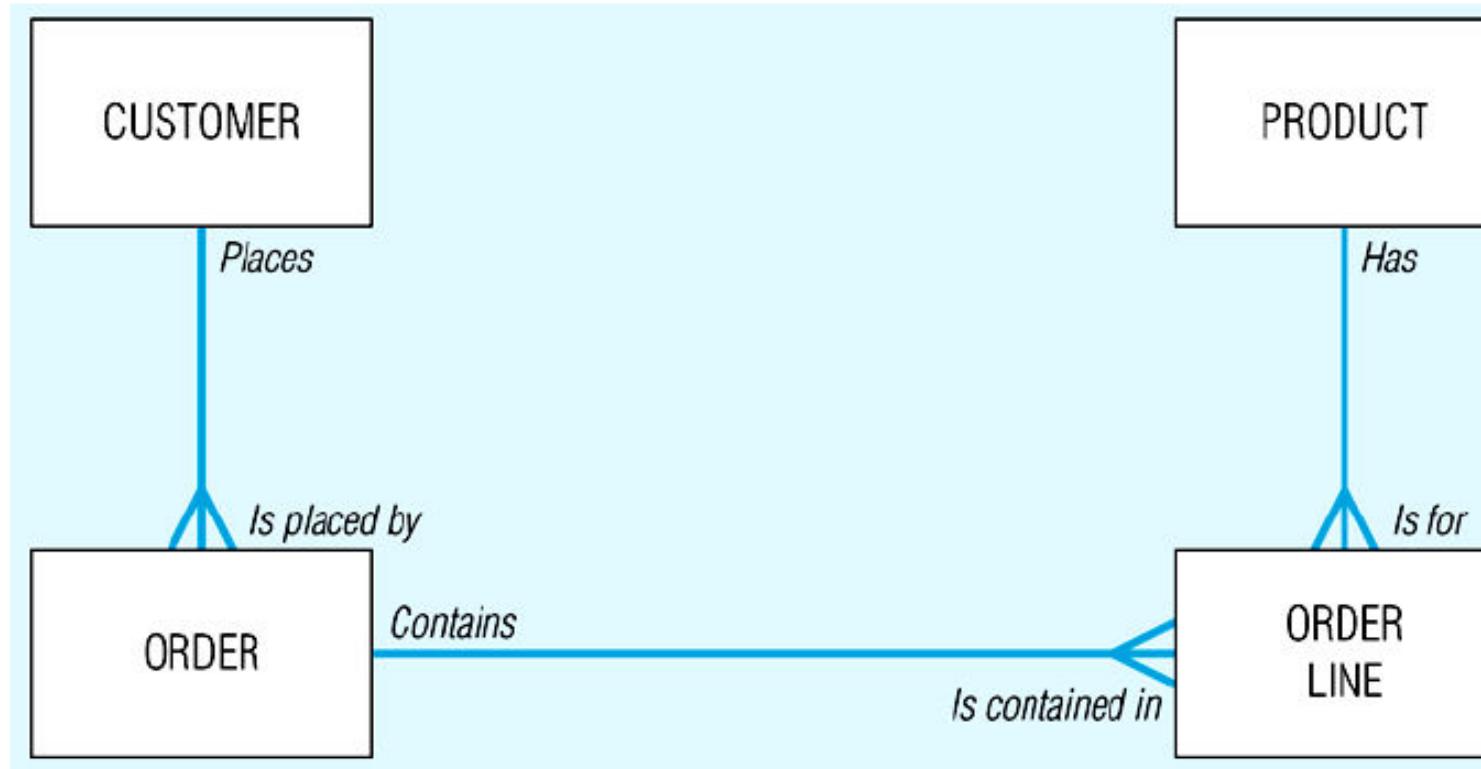
ORDER LINE Details about each product sold on a particular customer order (such as quantity and price)

Entities

1. Each CUSTOMER *Places* any number of ORDERS. Conversely, each ORDER *Is Placed By* exactly one CUSTOMER.
2. Each ORDER *Contains* any number of ORDER LINES. Conversely, each ORDER LINE *Is Contained In* exactly one ORDER.
3. Each PRODUCT *Has* any number of ORDER LINES. Conversely, each ORDER LINE *Is For* exactly one PRODUCT.

Business Rules

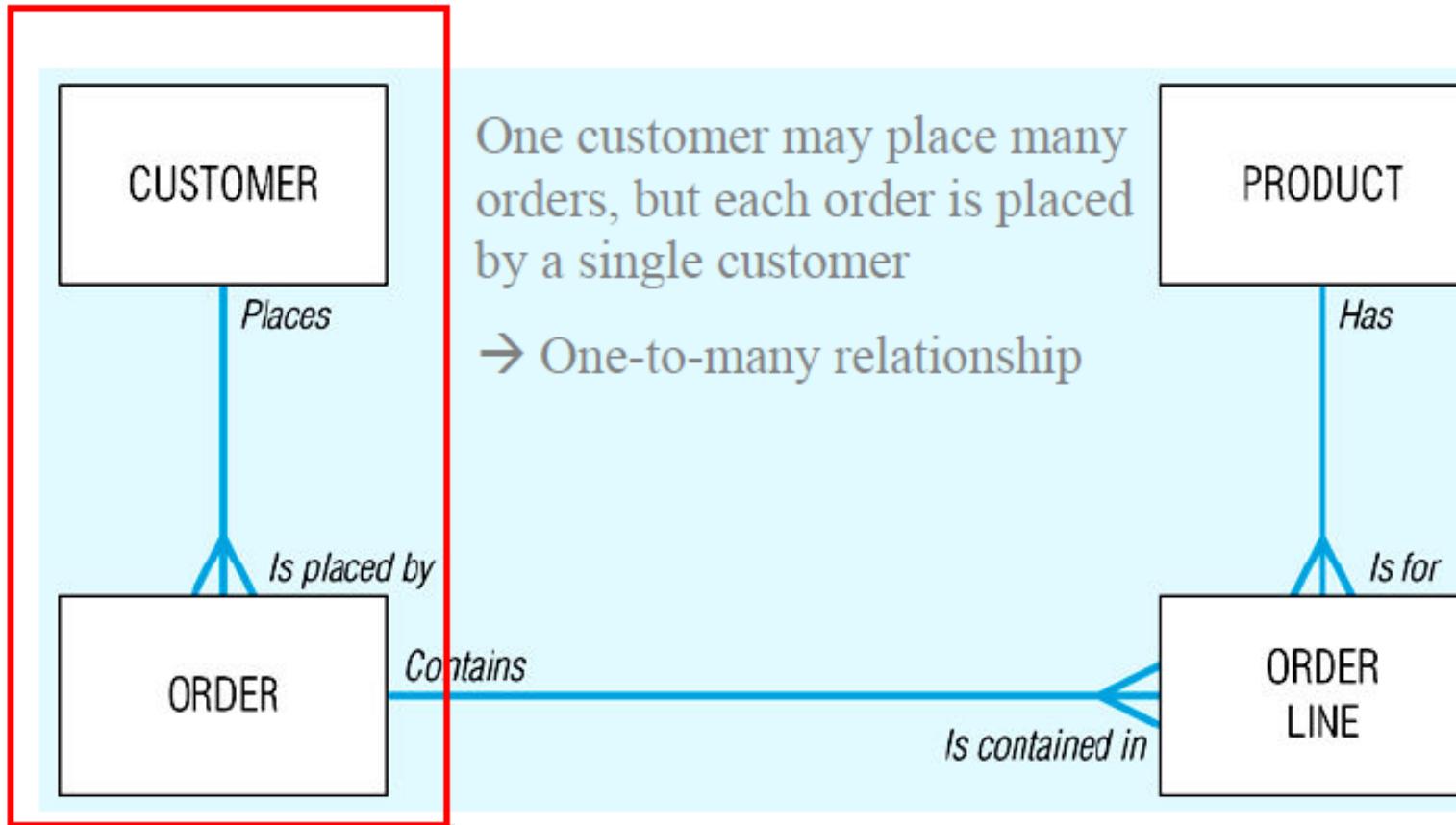
Initial Entity-Relationship Diagram



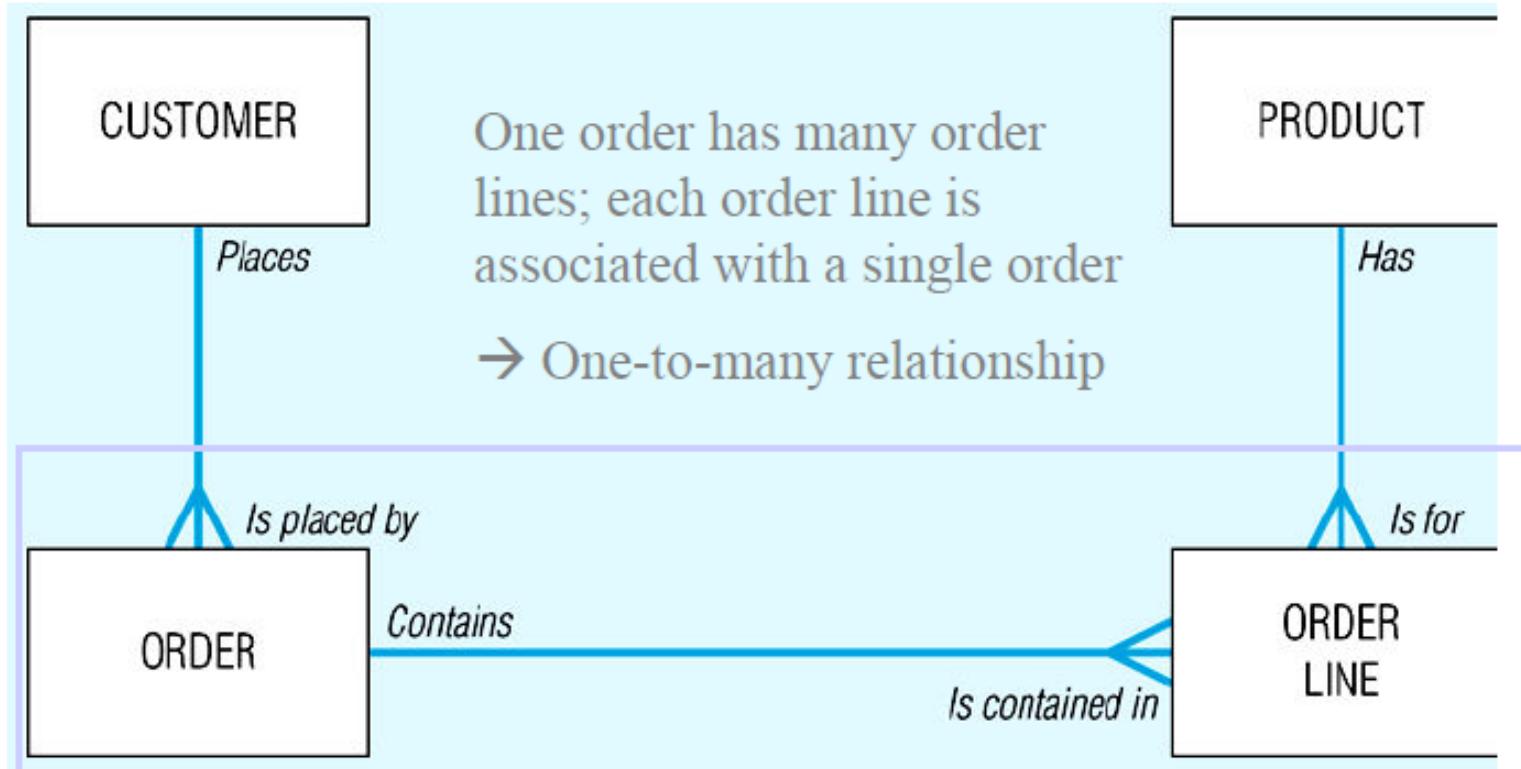
ERD

Version 1

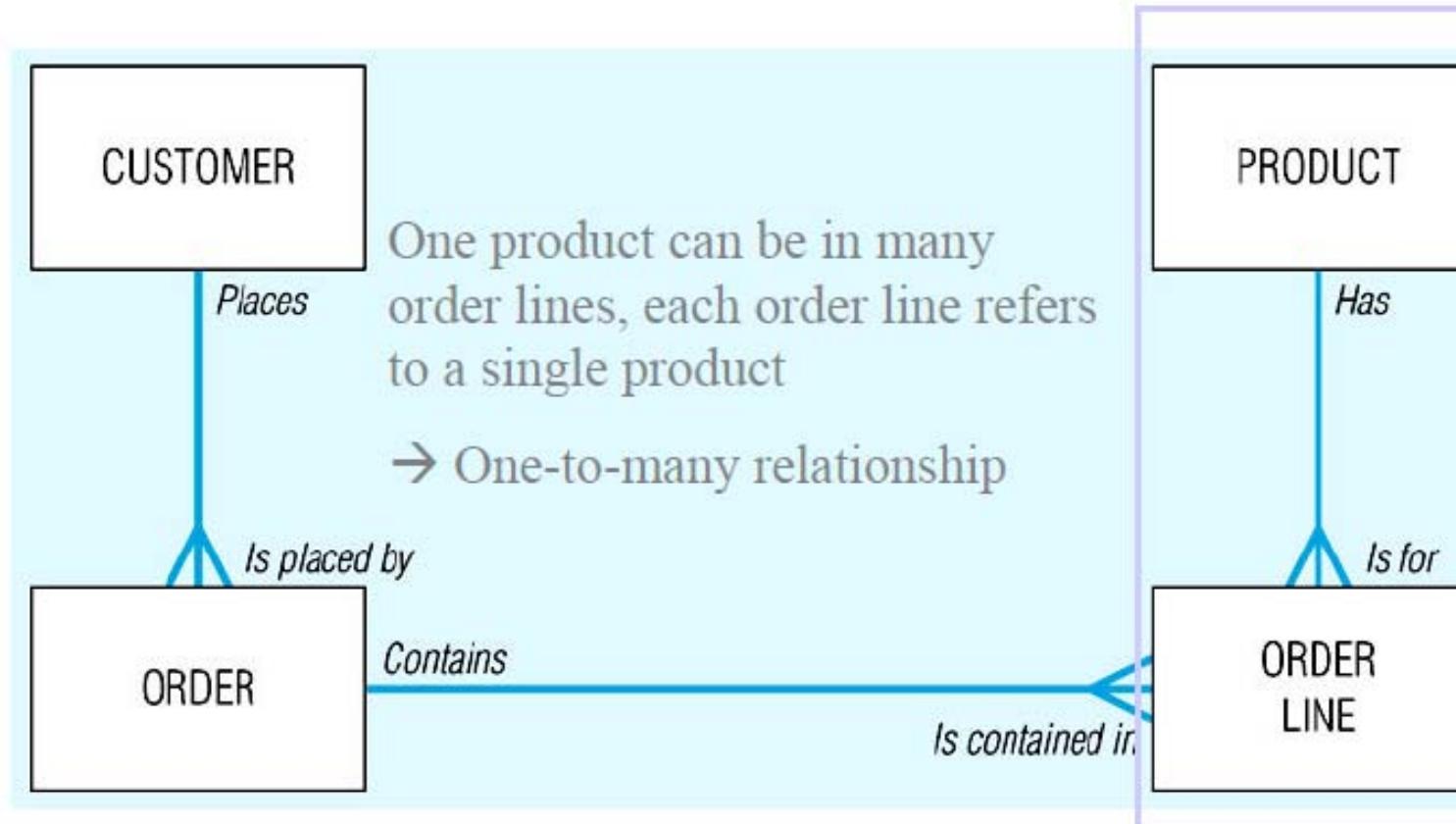
Initial Entity-Relationship Diagram (Cont.)



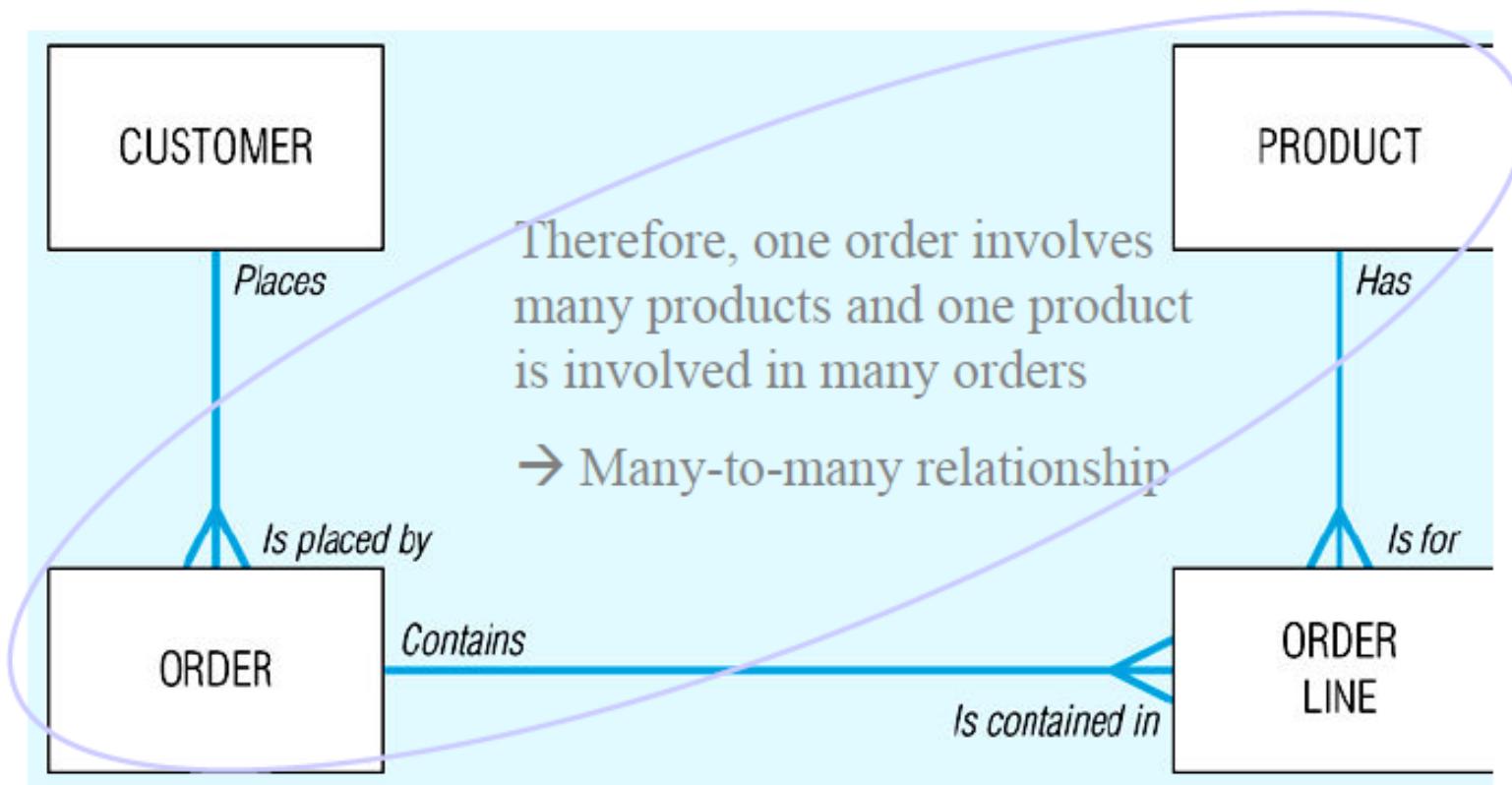
Initial Entity-Relationship Diagram (Cont.)



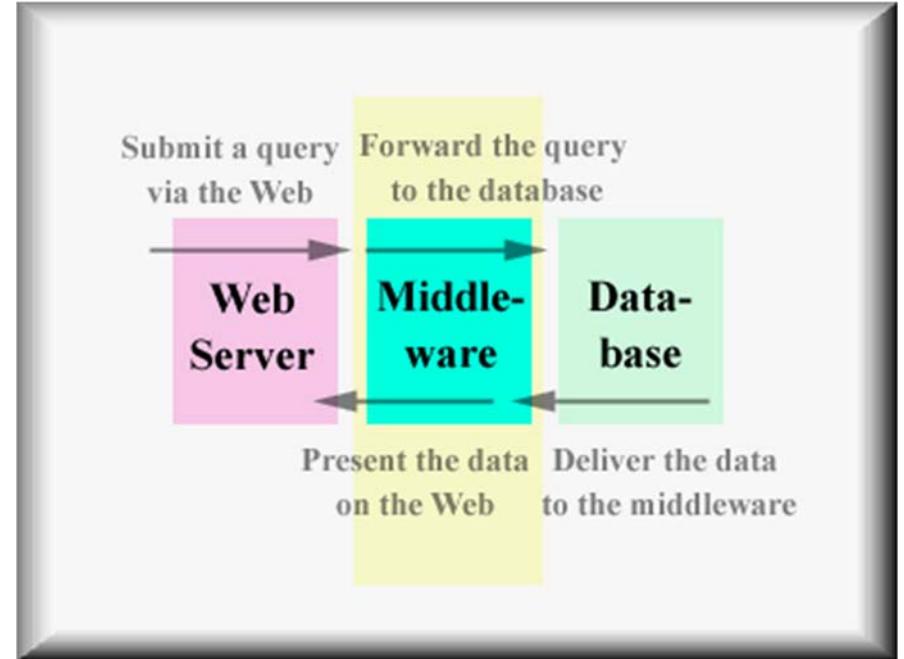
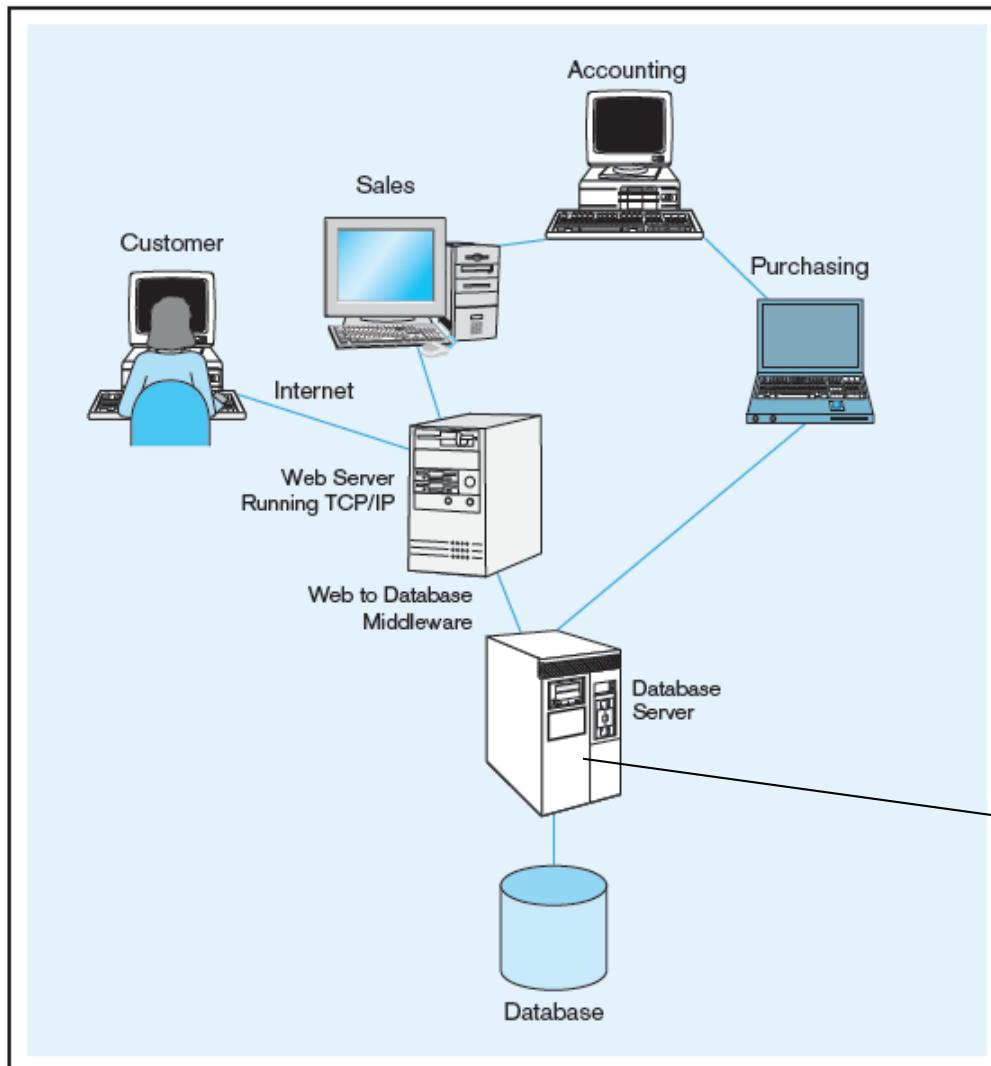
Initial Entity-Relationship Diagram (Cont.)



Initial Entity-Relationship Diagram (Cont.)

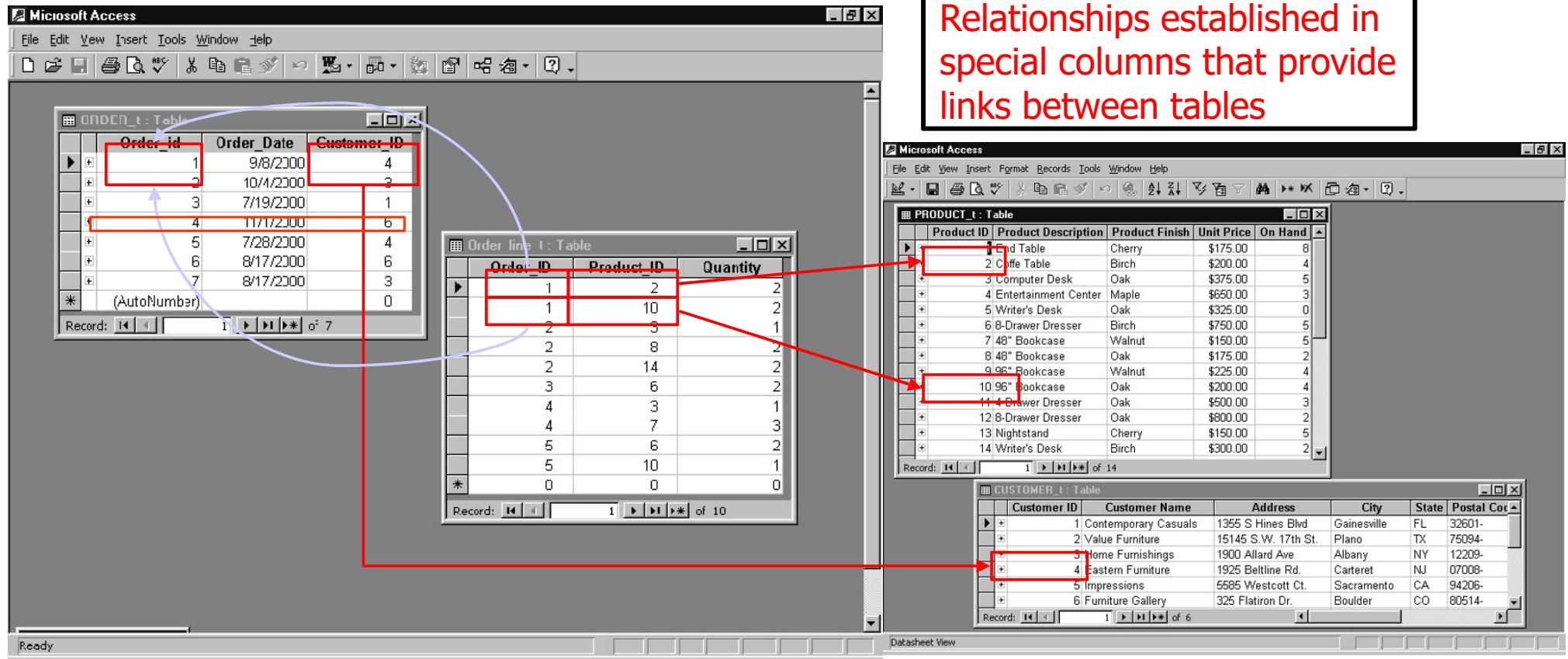


Client/Server System



A dedicated computer which provides database services to other computer programs or computers. Examples: Microsoft SQL Server, Oracle, DB2, and Informix.

Physical Tables in MS Access



- Each table row (tuple) represents a single entity occurrence within the entity set.
 - Each table column represents an attribute, and each column has a distinct name.
 - Each column/row intersection represents a single data value.
 - All values in a column must conform to the same data format.
 - The order of the row and columns is immaterial to the DBMS.
 - Each table must have an attribute or a combination of attributes that uniquely identifies each row.

Physical Tables in MySQL

Order

OrderID	OrderDate	CustomerID
1001	2010-10-21	4
1002	2010-10-21	3
1003	2010-10-22	1
1004	2010-10-22	6
1005	2010-10-24	4
1006	2010-10-24	6
1007	2010-10-27	3
1008	2010-10-30	5
1009	2010-11-05	1
1010	2010-11-05	7

10 rows in set (0.00 sec)

OrderID	ProductID	OrderedQuantity
1001	1	2
1001	2	2
1001	4	1
1002	3	5
1003	3	3
1004	6	2
1004	8	2
1005	4	4
1006	4	1
1006	5	2
1006	7	2
1007	1	3
1007	2	2
1008	3	3
1008	8	3
1009	4	2
1009	7	3
1010	8	10

18 rows in set (0.00 sec)

Customer

CustomerID	CustomerName
1	Contemporary Casuals
2	Value Furniture
3	Home Furnishings
4	Eastern Furniture
5	Impressions
6	Furniture Gallery
7	Period Furniture
8	California Classics
9	M and H Casual Furniture
10	Seminole Interiors
11	American Euro Lifestyles
12	Battle Creek Furniture
13	Heritage Furnishings
14	Kaneohe Homes
15	Mountain Scenes

15 rows in set (0.00 sec)

Product

ProductID	ProductStandardPrice
1	175.00
2	200.00
3	375.00
4	650.00
5	325.00
6	750.00
7	800.00
8	250.00

8 rows in set (0.00 sec)

Order Line

Metadata in MySQL

```
mysql> describe customer_t;
```

Field	Type	Null	Key	Default	Extra
CustomerID	int(11)	NO	PRI	NULL	auto_increment
CustomerName	varchar(30)	YES		NULL	

2 rows in set (0.10 sec)

Customer

Order

```
mysql> describe order_t;
```

Field	Type	Null	Key	Default	Extra
OrderID	int(11)	NO	PRI	NULL	auto_increment
OrderDate	date	NO		NULL	
CustomerID	int(11)	NO	MUL	NULL	

3 rows in set (0.07 sec)

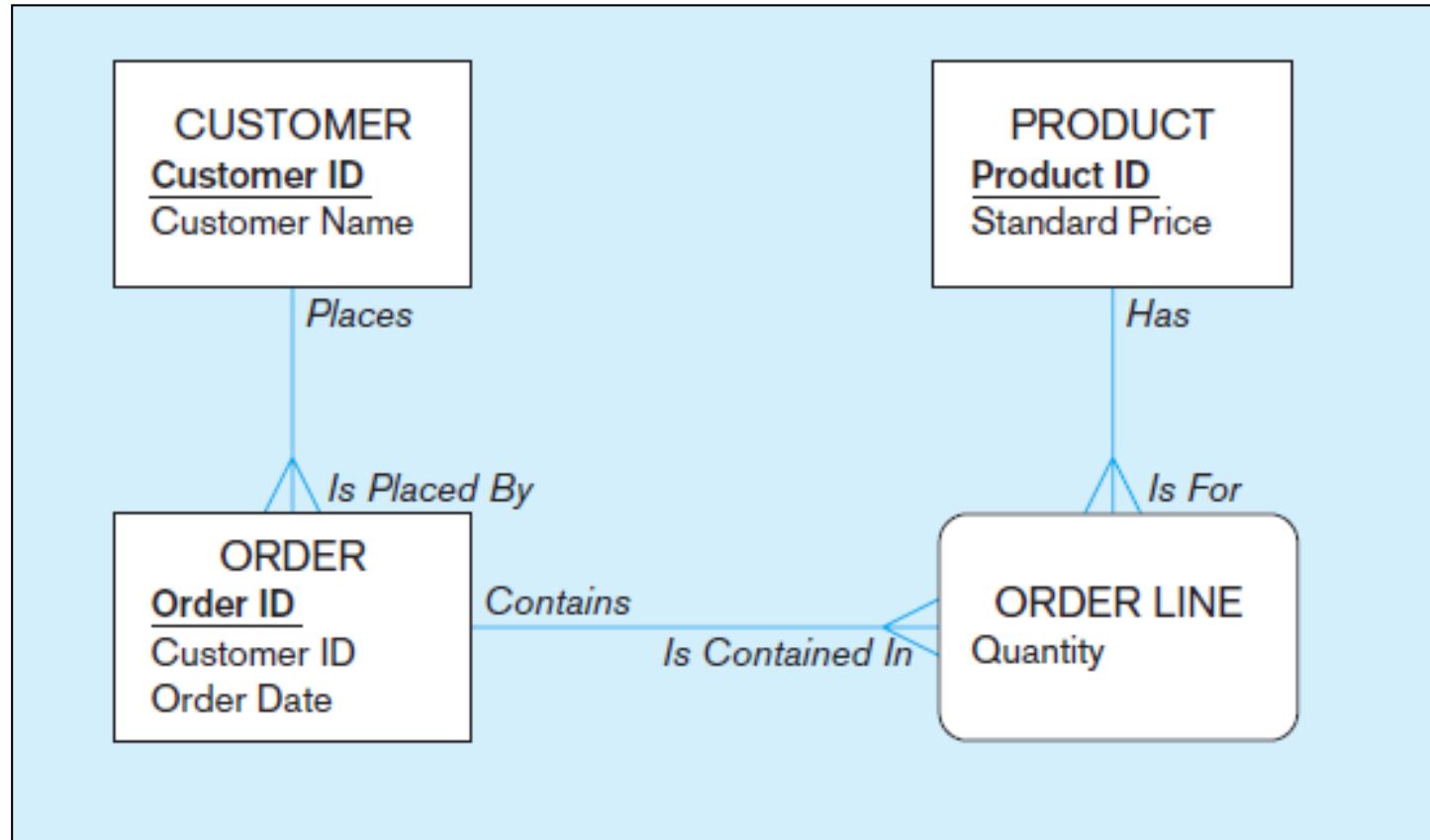
```
mysql> describe orderline_t;
```

Field	Type	Null	Key	Default	Extra
OrderID	int(11)	NO	PRI	NULL	
ProductID	int(11)	NO	PRI	NULL	
OrderedQuantity	int(11)	NO		NULL	

3 rows in set (0.04 sec)

Order
Line

Initial Entity-Relationship Diagram (Cont.)



Version 2

Figure 1-3b: Segment from Enterprise Data Model

Figure 1-14: Four Relations

(a) Order and Order Line Tables

The figure shows two Microsoft Access database tables side-by-side. The left table is titled 'Order_T' and contains columns for OrderID, OrderDate, and CustomerID. The right table is titled 'OrderLine_T' and contains columns for OrderID, ProductID, and OrderedQuantity. Both tables have their primary keys set to 'OrderID'. The 'OrderLine_T' table has an additional row at the bottom with values '0', '0', and '0' respectively.

	OrderID	OrderDate	CustomerID
[+]	1001	10/21/2010	
[+]	1002	10/21/2010	
[+]	1003	10/22/2010	
[+]	1004	10/22/2010	
[+]	1005	10/24/2010	
[+]	1006	10/24/2010	
[+]	1007	10/27/2010	
[+]	1008	10/30/2010	
[+]	1009	11/5/2010	
[+]	1010	11/5/2010	

Record: [Navigation Buttons] 1 of 10 [Search] [No Filter]

	OrderID	ProductID	OrderedQuantity
	1001	1	2
	1001	2	2
	1001	4	1
	1002	3	5
	1003	3	3
	1004	6	2
	1004	8	2
	1005	4	4
	1006	4	1
	1006	5	2
	1006	7	2
	1007	1	3
	1007	2	2
	1008	3	3
	1008	8	3
	1009	4	2
	1009	7	3
	1010	8	10
*	0	0	0

Record: [Navigation Buttons] 1 of 18 [Search] [No Filter]

Figure 1-14: Four Relations (Cont.)

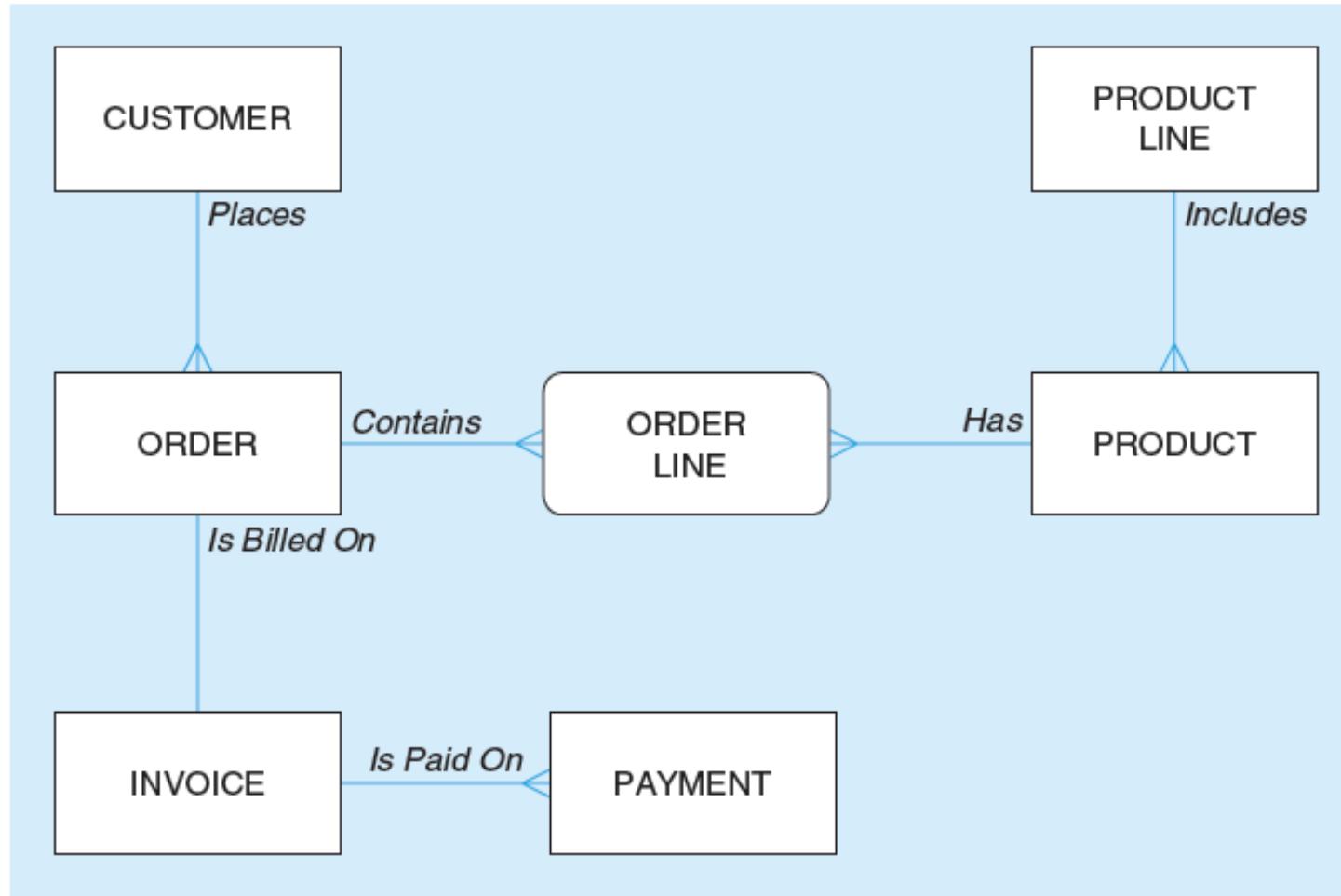
Customer_T		
	CustomerID	CustomerName
+	1	Contemporary Casuals
+	2	Value Furniture
+	3	Home Furnishings
+	4	Eastern Furniture
+	5	Impressions
+	6	Furniture Gallery
+	7	Period Furniture
+	8	California Classics
+	9	M and H Casual Furniture
+	10	Seminole Interiors
+	11	American Euro Lifestyles
+	12	Battle Creek Furniture
+	13	Heritage Furnishings
+	14	Kaneohe Homes
+	15	Mountain Scenes
*	(New)	

(b) Customer table

Product_T		
	ProductID	ProductStandardPrice
+	1	\$175.00
+	2	\$200.00
+	3	\$375.00
+	4	\$650.00
+	5	\$325.00
+	6	\$750.00
+	7	\$800.00
+	8	\$250.00
*	(New)	\$0.00

(c) Product table

Figure 1-15: Preliminary Data Model for Home Office Product Line Marketing



Version 3

Data Attributes for Entities in Preliminary Data Model

TABLE 1-6 Data Attributes for Entities in the Preliminary Data Model (Pine Valley Furniture Company)

Entity Type	Attribute
Customer	Customer Identifier
	Customer Name
	Customer Type
	Customer Zip Code
Product	Product Identifier
	Product Description
	Product Finish
	Product Price
	Product Cost
	Product Annual Sales Goal
	Product Line Name
Product Line	Product Line Name
	Product Line Annual Sales Goal

Data Attributes for Entities in Preliminary Data Model (Cont.)

Order	Order Number
	Order Placement Date
	Order Fulfillment Date
	Customer Identifier
Ordered Product	Order Number
	Product Identifier
	Order Quantity
Invoice	Invoice Number
	Order Number
	Invoice Date
Payment	Invoice Number
	Payment Date
	Payment Amount

Data Attributes for Entities in Final Data Model

TABLE 1-7 Data Attributes for Entities in Final Data Model (Pine Valley Furniture Company)

Entity Type	Attribute
Customer	Customer Identifier
	Customer Name
	Customer Type
	Customer Zip Code
	Customer Years
Product	Product Identifier
	Product Description
	Product Finish
	Product Price
	Product Cost
	<i>Product Prior Year Sales Goal</i>
	<i>Product Current Year Sales Goal</i>
Product Line	Product Line Name
	Product Line Name
	<i>Product Line Prior Year Sales Goal</i>
	<i>Product Line Current Year Sales Goal</i>

Data Attributes for Entities in Final Data Model (Cont.)

Order	Order Number Order Placement Date Order Fulfillment Date <i>Order Number of Shipments</i> Customer Identifier
Ordered Product	Order Number Product Identifier Order Quantity
Invoice	Invoice Number Order Number Invoice Date
Payment	Invoice Number Payment Date Payment Amount

*Changes from preliminary list of attributes appear in italics.

SQL Definitions

FIGURE 1-16 SQL definition of ProductLine table

```
CREATE TABLE ProductLine_T  
  
(ProductLineID      VARCHAR (40) NOT NULL PRIMARY KEY,  
  
PIPriorYearGoal    DECIMAL,  
  
PICurrentYearGoal  DECIMAL);
```

FIGURE 1-17 SQL definition of Product table

```
CREATE TABLE Product_T  
  
(ProductID          NUMBER(11,0) NOT NULL PRIMARY KEY  
  
ProductDescription  VARCHAR (50),  
  
ProductFinish       VARCHAR (20),  
  
ProductStandardPrice DECIMAL(6,2),  
  
ProductCost         DECIMAL,  
  
ProductPriorYearGoal DECIMAL,  
  
ProductCurrentYearGoal DECIMAL,  
  
ProductLineID       VARCHAR (40),  
  
FOREIGN KEY        (ProductLineID) REFERENCES ProductLine_T (ProductLineID);
```

Project Data Model

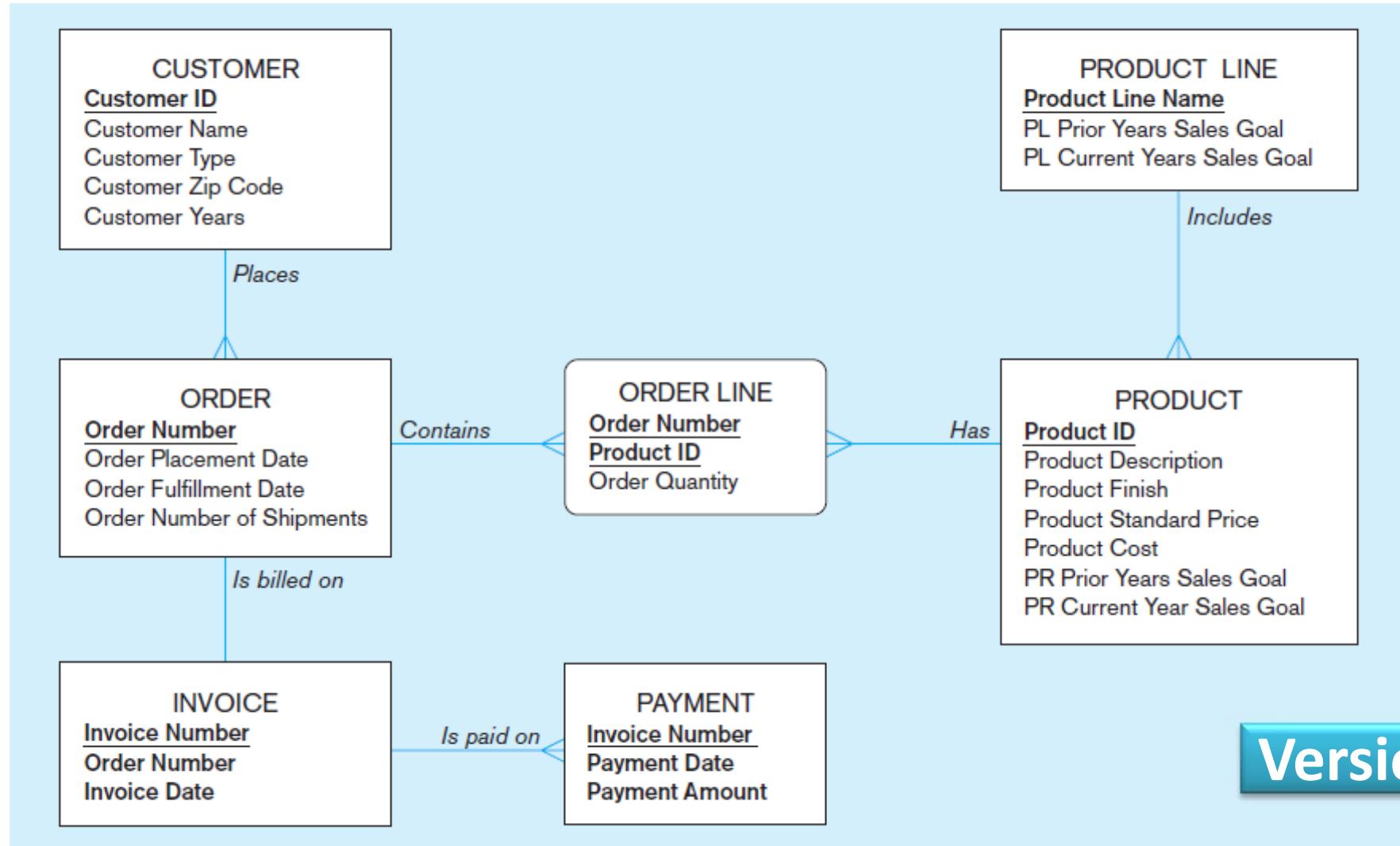
**Version 4**

Figure 1-18: Project Data Model for Home Office Product Line Marketing Support Sys.

SQL Query & its Result

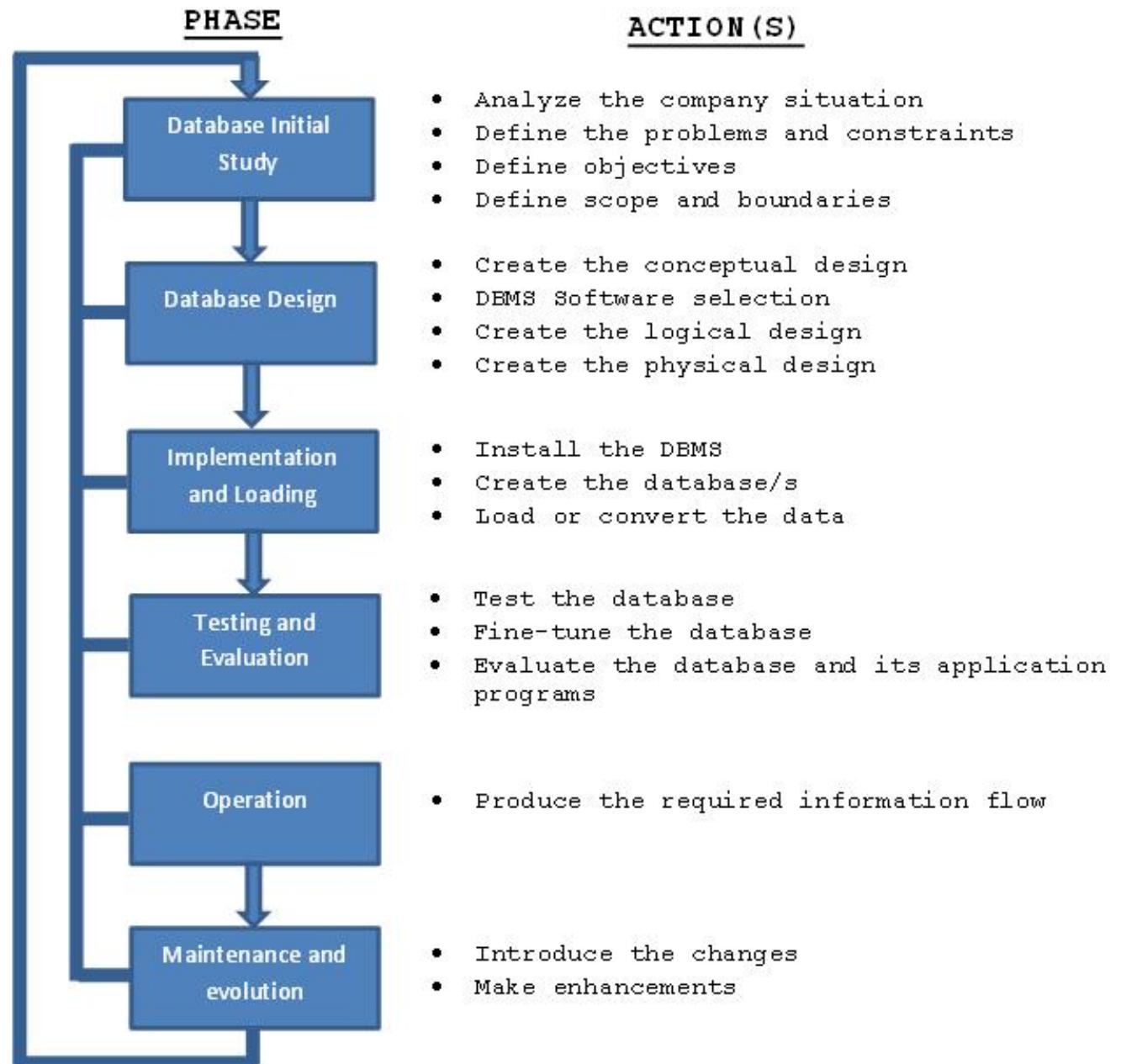
FIGURE 1-19 SQL query for Home Office sales-to-goal comparison

```
SELECT Product.ProductID, Product.ProductDescription, Product.PRCurrentYearSalesGoal,  
       (OrderQuantity * ProductPrice) AS SalesToDate  
  
FROM Order.OrderLine, Product.ProductLine  
  
WHERE Order.OrderNumber = OrderLine.OrderNumber  
  
AND Product.ProductID = OrderedProduct.ProductID  
  
AND Product.ProductID = ProductLine.ProductID  
  
AND Product.ProductLineName = "Home Office";
```

FIGURE 1-20 Home Office product line sales comparison

Home Office Sales to Date : Select Query				
	Product ID	Product Description	PR Current Year Sales Goal	Sales to Date
	3	Computer Desk	\$23,500.00	5625
	10	96" Bookcase	\$22,500.00	4400
	5	Writer's Desk	\$26,500.00	650
	3	Computer Desk	\$23,500.00	3750
	7	48" Bookcase	\$17,000.00	2250
	5	Writer's Desk	\$26,500.00	3900
▶				

The Database Life Cycle (DBLC)



Front-end & Back-end Databases

- The front end of a website is the part that users interact with. Also referred to as client-side, it includes everything that you see when you're navigating around the Internet, from fonts and colors to dropdown menus and sliders, is a combo of HTML, CSS, and JavaScript being controlled by your computer's browser.
- The back end of a website consists of a server, an application, and a database. Also referred as server-side. A back-end developer builds and maintains the technology that powers those components which, together, enable the client-side to even exist in the first place. Some common backend languages are PHP, Java, .Net, Python, and Rails.
- Full Stack Development

Summary

- Introduced various terms
- Covered database approach & its advantages
- Introduced Enterprise Data Model and its various constructs
- Discussed Database Lifecycle (DBLC)
- Covered a sample Case Study
- Discussed Front-end and Back-end Databases