The Database Approach to Data Management

- **Database:**
  - Collection of related files containing records on people, places, or things.
  - Prior to digitizing DBs, business used paper files.

- **Entity:**
  - Generalized category representing person, place, thing on which we store info.
  - E.g., SUPPLIER, PART

- **Attributes:**
  - Specific characteristics of each entity:
    - SUPPLIER name, address
    - PART description, unit price, supplier
• Relational database:
  • Organize data into tables
  • One table for each entity:
    • E.g., (CUSTOMER, SUPPLIER, PART, SALES)
  • Fields (columns) store data representing an attribute.
  • Rows store data for separate records.
  • Key field: uniquely identifies each record.
  • Primary key:
    • One field in each table
    • Cannot be duplicated
    • Provides unique identifier for all information in any row
A relational database organizes data in the form of two-dimensional tables. Illustrated here is a table for the entity SUPPLIER showing how it represents the entity and its attributes. Supplier_Number is the key field.

Figure 5-1
The PART Table

<table>
<thead>
<tr>
<th>Part_Number</th>
<th>Part_Name</th>
<th>Unit_Price</th>
<th>Supplier_Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>137</td>
<td>Door latch</td>
<td>22.00</td>
<td>8259</td>
</tr>
<tr>
<td>145</td>
<td>Side mirror</td>
<td>12.00</td>
<td>8444</td>
</tr>
<tr>
<td>150</td>
<td>Door molding</td>
<td>6.00</td>
<td>8263</td>
</tr>
<tr>
<td>152</td>
<td>Door lock</td>
<td>31.00</td>
<td>8259</td>
</tr>
<tr>
<td>155</td>
<td>Compressor</td>
<td>54.00</td>
<td>8261</td>
</tr>
<tr>
<td>178</td>
<td>Door handle</td>
<td>10.00</td>
<td>8259</td>
</tr>
</tbody>
</table>

Figure 5-2
• Establishing relationships
  • Entity-relationship diagram
    • Used to clarify table relationships in a relational database
  • Relational database tables may have:
    • One-to-one relationship
    • One-to-many relationship
    • Many-to-many relationship
      • Requires creating a table (join table, Intersection relation) that links the two tables to join information
A Simple Entity-Relationship Diagram

This diagram shows the relationship between the entities SUPPLIER and PART: One-to-many.

Figure 5-3
• **Normalization.** Process of streamlining complex groups of data (especially those with many-to-many relationship) to:
  - Minimize redundant data elements.
  - Minimize awkward many-to-many relationships.
  - Increase stability and flexibility.

• **Referential integrity rules**
  - Used by relational databases to ensure that relationships between coupled tables remain consistent.
  - Idea: not add a record to the table with the foreign key unless there is a corresponding record in the linked table
  - E.g, Not add supplier # 8266 to PART unless there is a 8266 in SUPPLIERS
This diagram shows the relationship between the entities SUPPLIER, ART, LINE_ITEM, and ORDER.

Figure 5-6
The Final Database Design with Sample Records

Figure 5-5

<table>
<thead>
<tr>
<th>PART</th>
<th>LINE_ITEM</th>
<th>ORDER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part_Number</td>
<td>Part_Name</td>
<td>Unit_Price</td>
</tr>
<tr>
<td>137</td>
<td>Door latch</td>
<td>22.00</td>
</tr>
<tr>
<td>145</td>
<td>Side mirror</td>
<td>12.00</td>
</tr>
<tr>
<td>150</td>
<td>Door molding</td>
<td>6.00</td>
</tr>
<tr>
<td>152</td>
<td>Door lock</td>
<td>31.00</td>
</tr>
<tr>
<td>155</td>
<td>Compressor</td>
<td>54.00</td>
</tr>
<tr>
<td>178</td>
<td>Door handle</td>
<td>10.00</td>
</tr>
</tbody>
</table>

| SUPPLIER | | | | |
|-----------|-----------|-----------|-----------|
| Supplier_Number | Supplier_Name | Supplier_Street | Supplier_City | Supplier_State | Supplier_Zip |
| 8259  | CBM Inc.  | 74 5th Avenue | Dayton | OH | 45220 |
| 8261  | B. R. Molds | 1277 Gandolly Street | Cleveland | OH | 49345 |
| 8263  | Jackson Components | 8233 Micklin Street | Lexington | KY | 56723 |
| 8444  | Bryant Corporation | 4315 Mill Drive | Rochester | NY | 11344 |
Sample Order Report

Order Number: 3502
Order Date: 1/15/2008

Supplier Number: 8259
Supplier Name: CBM Inc.
Supplier Address: 74 5th Avenue, Dayton, OH 45220

<table>
<thead>
<tr>
<th>Order_Number</th>
<th>Part_Number</th>
<th>Part_Quantity</th>
<th>Part_Name</th>
<th>Unit_Price</th>
<th>Extended_Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>3502</td>
<td>137</td>
<td>10</td>
<td>Door latch</td>
<td>22.00</td>
<td>$220.00</td>
</tr>
<tr>
<td>3502</td>
<td>152</td>
<td>20</td>
<td>Door lock</td>
<td>31.00</td>
<td>620.00</td>
</tr>
<tr>
<td>3502</td>
<td>178</td>
<td>5</td>
<td>Door handle</td>
<td>10.00</td>
<td>50.00</td>
</tr>
</tbody>
</table>

Order Total: $890.00

Figure 5-4
**DBMS**

- Specific type of software for creating, storing, organizing, and accessing data from a database
- Separates the logical and physical views of the data
  - **Logical view**: how end users view data
  - **Physical view**: how data are actually structured and organized
- **Examples of DBMS**: Microsoft Access, DB2, Oracle Database, Microsoft SQL Server, MySQL
Human Resources Database with Multiple Views

Figure 5-7
Operations of a Relational DBMS

• Select:
  • Creates a subset of all records meeting stated criteria

• Join:
  • Combines relational tables to present the server with more information than is available from individual tables

• Project:
  • Creates a subset consisting of columns in a table
  • Permits user to create new tables containing only desired information
The Three Basic Operations of a Relational DBMS

The select, project, and join operations enable data from two different tables to be combined and only selected attributes to be displayed.
Capabilities of Database Management Systems

• Data definition capabilities:
  • Specify structure of content of database.

• Data dictionary:
  • Automated or manual file storing definitions of data elements and their characteristics, e.g., names, descriptions, size, type, format, etc.

• Querying and reporting:
  • Data manipulation language (add, change, delete and retrieve data)
    • E.g., Structured query language (SQL)
    • E.g., Microsoft Access query-building tools
Example of an SQL Query

SELECT PART.Part_Number, PART.Part_Name, SUPPLIER.Supplier_Number, SUPPLIER.Supplier_Name
FROM PART, SUPPLIER
WHERE PART.Supplier_Number = SUPPLIER.Supplier_Number AND Part_Number = 137 OR Part_Number = 150;

Illustrated here are the SQL statements for a query to select suppliers for parts 137 or 150. They produce a list with the same results as Figure 5-8.

Figure 5-10
Online Analytical Processing (OLAP)

- Supports multidimensional data analysis
  - Enable users to view same data in different ways using multiple dimensions
  - Dimension can be — product, pricing, cost, region, or time period
  - E.g., comparing sales in East in June versus May and July
Figure 5-14
Multidimensional Data Model
Data Mining

- Finds **hidden** patterns and relationships in large databases
- **Types of information obtainable from data mining**
  - **Associations**: occurrences linked to single event, e.g., chip & coke
  - **Sequences**: events linked over time, e.g., purchase new appliance with the first two weeks of new house
  - **Classifications**: patterns describing a group an item belongs to, e.g., discover characteristics of customers who are likely to leave the services.
  - **Clusters**: discovering as yet unclassified or not defined groupings
  - **Forecasting**: uses series of values to forecast future values through the pattern of data.
Data Mining

- **One popular use of data mining**: identifying profitable customers

- **Predictive analysis**:
  - Uses historical data, and assumptions about future conditions to predict outcomes of events
  - E.g. such the probability a customer will respond to an offer or purchase a specific product
• **Text Mining**
  
  - Unstructured data (mostly text files) accounts for 80 percent of an organization’s useful information.
  
  - Text mining -- extract key elements from, discover patterns in, and summarize large unstructured data sets.

• **Web Mining**
  
  - Discovery and analysis of useful patterns and information from the Web