5

Changing Data into Information

- Data are the raw facts that are stored in databases.
- Raw facts are seldom immediately useful to a decision maker.
- What the decision maker really needs is information, which is defined as data processed and presented in a meaningful form.

TABLE 6.1 A SIMPLE TABULATION: TRANSFORMING DATA INTO INFORMATION

Male	119	1,892	2,641	876	5,528
Female	48	1,117	1,805	542	3,512
Total	167	3,009	4,446	1,418	9,040

5

The Information System

- A database is a carefully designed and constructed repository of facts and is part of larger whole known as an information system.
  - An IS provides for data collection, storage, and retrieval.
  - IS also facilitates the transformation of data into information and the management of both data and information.
  - Components of an information system:
    - People
    - Hardware
    - Software
    - Database(s)
    - Application programs
    - Procedures

5

The Information System

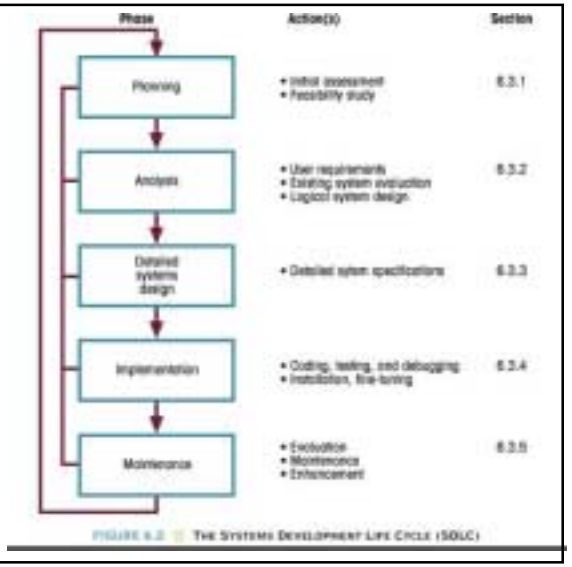
- System analysis is the process that establishes the need for and the extent of an IS.
- The process of creating an IS is known as systems development.
- Applications transform data into the information.
- An application is composed of two parts: the data and the code. (Figure 6.1)
- The performance of an IS depends on three factors:
  - Database design and implementation (DB development)
  - Applications design and implementation
  - Administrative procedures

5

Generating Information for Decision Making

Figure 6.1

Figure 6.2



Planning Phase

- An initial assessment of the information-flow-and-extent requirements must be made:
  - Should the existing system be continued?
  - Should the existing system be modified?
  - Should the existing system be replaced?
- A feasibility study must address the following issues if a new system is necessary:
  - Technical aspects of hardware and software requirements.
  - The system cost.

Robert Laurie - 6

Analysis Phase

- Audit of user requirements.
  - The existing hardware and software are studied.
  - End users and system designer(s) work together to identify processes and potential problem areas.
- Logical system design.
  - Specifies conceptual data model, inputs, processes, and expected output requirements.
  - System design tools:
    - Data flow diagram (DFD)
    - Entity Relationship (E-R) diagrams
- Defining the logical system also yields functional descriptions (FD) of the system's components (modules) for each process within the database environment.

Robert Laurie - 7

Design Phase

- Detailed Systems Design
  - The designer completes the design of the system's processes, including all technical specifications for:
    - Screen
    - Menus
    - Reports
    - Other devices
  - Conversion steps are laid out.
  - Training principles and methodologies are planned.

Robert Laurie - 8

Implementation Phase

- The hardware, the DBMS software, and application programs are installed; and the database design is implemented.
- The system enters into a cycle of coding, testing, and debugging.
- The database is created, and the system is customized.
- The database contents are loaded.
- The system is subjected to exhaustive testing.
- The final documentation is reviewed and printed.
- End users are trained.

Robert Laurie - 9

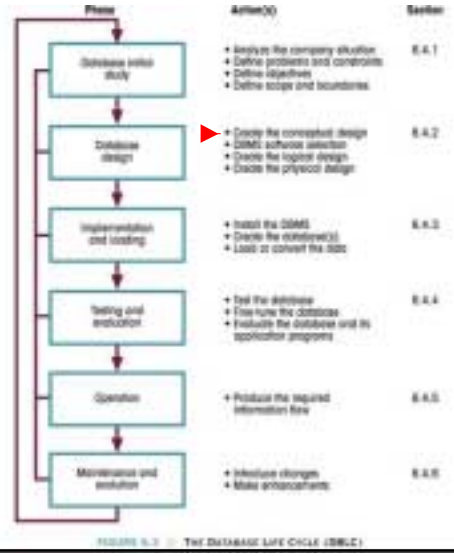
Maintenance Phase

- End users' requests for changes generate system maintenance activities.
- Three types of system maintenance:
  - Corrective maintenance in response to systems errors.
  - Adaptive maintenance due to changes in the business environment.
  - Perfective maintenance to enhance the system.
- Computer-assisted systems engineering (CASE) technology helps.

Robert Laurie - 10

Figure 6.3

The Database Life Cycle has Six Phases



Database Initial Study Phase

- Overall Purpose of the Initial Study:
  - Analyze the company situation.
  - Define problems and constraints.
  - Define objectives.
  - Define scope and boundaries.
    - Scope -- operational requirements?
    - Boundaries -- What are the limits?

Robert Laurie - 12

5

Database Design Phase

- Conceptual Design
  - Data modeling is used to create an abstract database structure that represents real-world objects.
  - The design must be software and hardware independent.
  - Minimal data rule: *All that is needed is there, and all that is there is needed.*
  - Four Steps of conceptual design:
    1. Data analysis and requirements
    2. Entity relationship modeling and normalization
    3. Data model verification
    4. Distributed database design

5

Conceptual Design

1. Data analysis and requirements
   - Designer's efforts are focused on needs, users, sources, and constitution of information.
   - Sources of information for the designer
     - Developing and gathering end user data views
     - Direct observation of the current system: existing and desired output
     - Interface with the systems design group
   - The designer must identify the company's business rules and analyze their impacts.
     - Business rule is a narrative description of a policy, procedure, or principle within a business environment.

Conceptual Design

2. Entity Relationship Modeling and Normalization
   a) Define entities, attributes, primary keys, and foreign keys.
   b) Make decisions about adding new primary key attributes in order to satisfy end user and/or processing requirements.
   c) Make decisions about the treatment of multivalued attributes.
   d) Make decisions about adding derived attributes to satisfy processing requirements.
   e) Make decisions about the placement of foreign keys in 1:1 relationships.
   f) Avoid unnecessary ternary relationships.
   g) Draw the corresponding E-R diagram.
   h) Normalize the data model.
   i) Make decisions about standard naming conventions.
   j) Include all the data element definitions in the data dictionary.

5

Standard Naming Conventions

- Use descriptive entity and attribute names wherever possible.
- Composite entities usually are assigned a name that is descriptive of the relationships they represent.
- An attribute name should be descriptive and it should contain a prefix that helps identify the table in which it is found.

Figure 6.9 Conceptual Design Tools And Information Sources

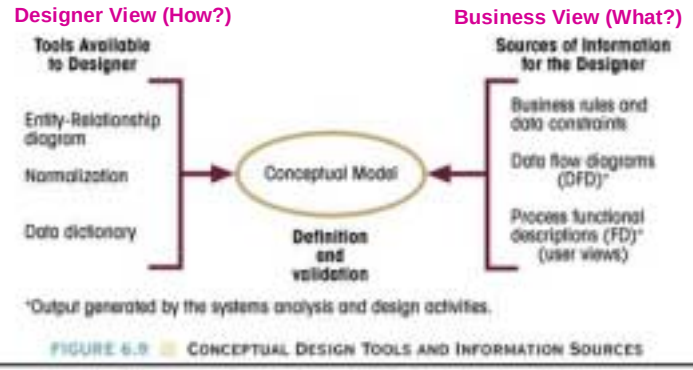


FIGURE 6.9 CONCEPTUAL DESIGN TOOLS AND INFORMATION SOURCES

Conceptual Design

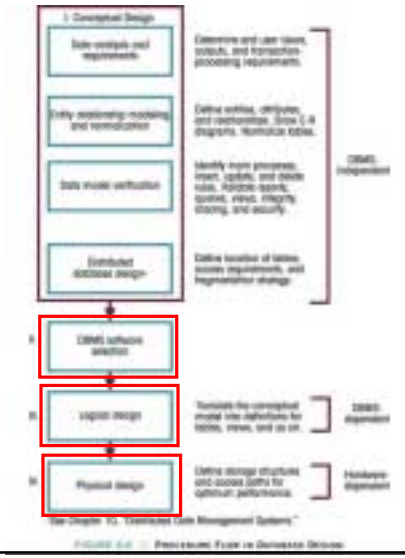
3. **Data Model Verification**
   - **Purposes of close review of entities and attributes**
     - The emergence of the attribute details may lead to a revision of the entities themselves.
     - The focus on attribute details can provide clues about the nature of the relationships as they are defined by the primary and foreign keys.
     - To satisfy processing and/or end user requirements, it might be useful to create a new primary key to replace an existing primary key.
     - Unless the entity details are precisely defined, it is difficult to evaluate the extent of the design's normalization.
   - **Advantages of the Modular Approach**
     - The modules can be delegated to design groups, greatly speeding up the development work.
     - The modules simplify the design work.
     - The modules can be prototyped quickly.

5

Conceptual Design

4. **Distributed Database Design**
   - **Design portion of a database may reside in different physical locations.**
   - **If the database process is to be distributed across the system, the designer must also develop the data distribution and allocation strategies for the database.**

Figure 6.6



5

Database Design Phase

- DBMS Software Selection
  - Common factors affecting the decision:
    - Cost -- Purchase, maintenance, operational, license, installation, training, and conversion costs.
    - DBMS features and tools.
    - Underlying model.
    - Portability -- Platforms, systems, and languages.
    - DBMS hardware requirements.
    - Vendors: IBM DB2, Oracle, MS Access, MS SQL Server, Informix, Ingress, SAP

5

Database Design Phase

- Logical Design
  - Logical design translates the conceptual design into the internal model for a selected DBMS.
  - It includes mapping of all objects in the model to the specific constructs used by the selected database software.
  - For a relational DBMS, the logical design includes the design of tables, indexes, views, transactions, access authorities, and so on.

5

Figure 6.11 Translate this Conceptual Model…

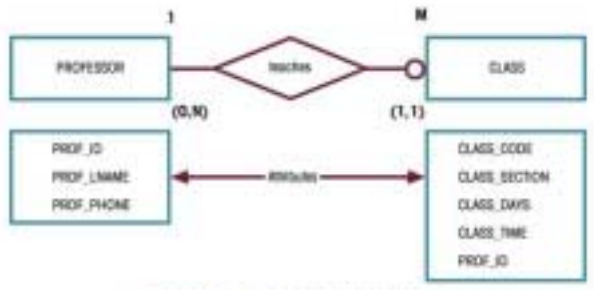


FIGURE 6.11 A SIMPLE CONCEPTUAL MODEL

- To Logical Design for specific DBMS
  - Attribute Domains

PROF_ID	Is a valid professor identification number. Type: numeric Range: low value = 1,000	high value =2,000
	Display format: 9999 Length: 4	
PROF_LNAME	Is a valid professor last name. Type: character Display format: XXXXXXXXXXXXXX Length: 15	
PROF_PHONE	Is a valid phone number. Type: character Display format: 999-999-9999 Length: 12	
CLASS_CODE	Is a valid class code. Type: numeric Range: low value = 1,000	high value =1,999
	Display format: 9999 Length: 4	

- To Logical Design for specific DBMS
  - Attribute Domains
  - Table Layout

TABLE 6.4 A SAMPLE TABLE LAYOUT (TABLE NAME: PROFESSOR)

PROF_ID	PROF_LNAME	PROF_PHONE
2134	Darnell	615-892-9118
2139	Smithson	615-892-1123
2210	Rodriguez	615-892-2368
2214	Vann	615-890-3246

5

Database Design Phase

- Physical Design
  - Physical design is the process of selecting the data storage and data access for the database. Affects both location of the data in the storage device and the performance.
  - The storage characteristics are a function of:
    - The types of devices supported by the computer hardware.
    - The type of data access methods supported by the DBMS Software.
  - Physical design is particularly important in the older hierarchical and network models.
  - Relational databases are more insulated from physical layer details than hierarchical and network models.

Robert Laurie - 26

Figure 6.3

The Database Life Cycle has Six Phases

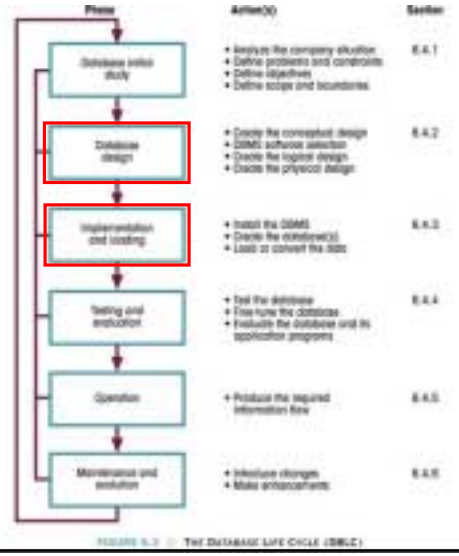
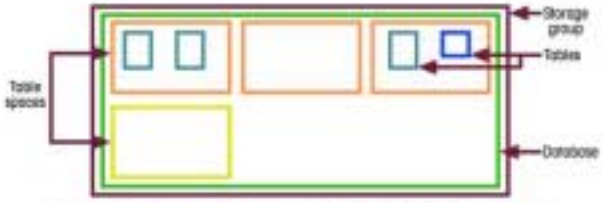


Figure 6.12: Physical Organization of a DB2 Database Environment



- Implementation and Loading Phase
  1. Create the database within the storage group.
  2. Assign the rights to use the database to administrator.
  3. Create the table space(s) within the database.
  4. Create the table(s) within the table space(s).
  5. Assign access rights to the table spaces and tables.
  6. Load the data.

5

Implementation and Loading Phase

- Physical Design Issues
  - Performance
  - Security
    - Physical security
    - Password security
    - Access rights
    - Audit trails
    - Data encryption
    - Diskless workstations
  - Backup and Recovery
  - Integrity
  - Company standards
  - Concurrency controls

5

The Database Life Cycle

- Testing and Evaluation Phase
  - The testing and evaluation phase occurs in parallel with application programming.
  - Programmers use database tools (e.g., report generators, screen painters, and menu generators) to prototype the applications during the coding of programs.
  - Options to enhance the system if the implementation fails performance criteria.
    - Fine-tuning the specific system and DBMS configuration parameters.
    - Modify physical design.
    - Upgrade or change the DBMS and hardware platform.

5

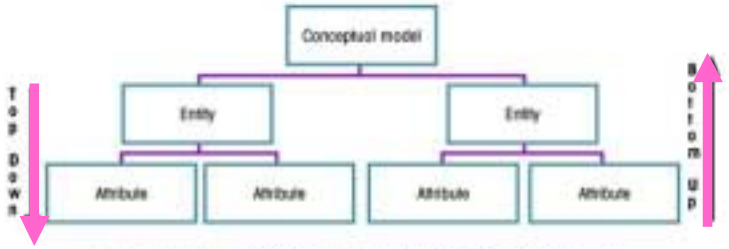
The Database Life Cycle

- Operation Phase
  - Once the database has passed the evaluation stage, it is considered to be operational.
  - The beginning of the operational phase invariably starts the process of system evolution.
    - Preventive maintenance (Backup)
    - Corrective maintenance (Recovery)
    - Adaptive maintenance (Enhancements)
    - Periodic security audits
    - Periodic system-usage summaries

A Special Note about Database Design Strategies

Figure 6.14: Top-Down Versus Bottom-Up Design Sequencing

Top-down design starts by identifying the data sets, and then defines the data elements for each of these sets.



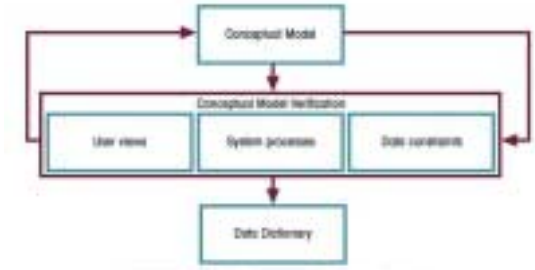
Bottom-up design first identifies the data elements (items), and then groups them together in data sets.

5

Two Different Database Design Philosophies

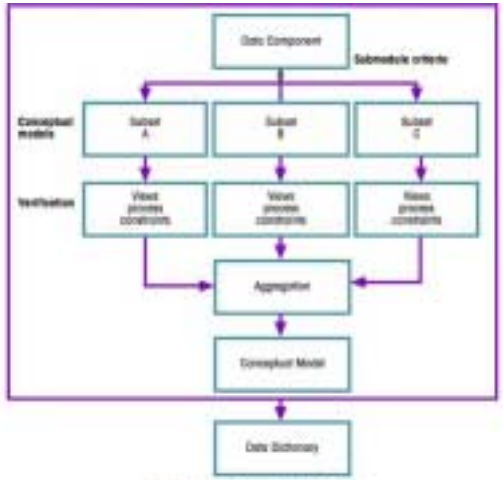
- Centralized design:

  It is productive when the data component is composed of a relatively small number of objects and procedures.



- 33

Figure 6.16



Decentralized design

Used for large databases that require group efforts.

Aggregation problems:

- **Synonyms and homonyms.**
- **Entity and entity subtypes.**
- **Conflicting object definitions.**