STSCI 5060

Database Management and SAS High Performance Computing with DBMS

Xiaolong Yang

Dept. of Statistical Science

Some Questions:

- What is a database?
- Any examples?
- Have you used a database?
- Why are you interested in this class?

There are three parts in the course

- Part 1: Basics of Database Management
- Part 2: PROC SQL
- Part 3: SAS High Performance Computing with DBMS

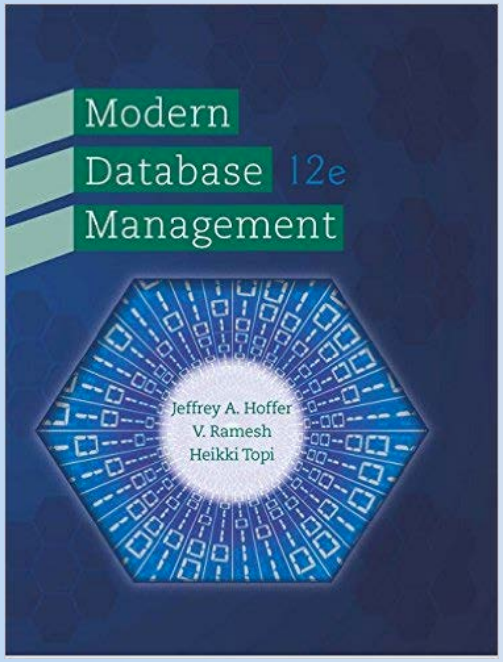
Part 1
Modern Database Management (MDBM)

Introduce the basics of modern relational database management systems, including database analysis, design and implementation

Reference book 1:

Modern Database Management

(12th Edition) by Jeffrey A. Hoffer, V. Ramesh, and Heikki Topi



MDBM-Chapter 1: The Database Environment and Development Process

Objectives

- Define terms
- Discuss limitations of conventional file processing
- Explain advantages of databases
- Identify costs and risks of databases
- List components of database environment
- Identify categories of database applications
- Describe database system development life cycle
- Explain prototyping development approach
- Explain the three-schema architecture for databases

Some Important Definitions

- Data
- Information
- Metadata
- Repository
- Database
- Database management system (DBMS)

Data

Stored representations of meaningful objects and events

- **Structured data**: numbers, characters, dates organized in a tabular form
- **Unstructured data**: images, video, audio, documents

Some examples of data

Baker, Kenneth D.	324917628
Doyle, Joan E.	476193248
Finkle, Clive R.	548429344
Lewis, John C.	551742186
McFerran, Debra R.	409723145
Sisneros, Michael	392416582

MGT 500	Business Policy
STSCI 5060	Data Base
STSCI 5010	SAS Programming
CS 4780	Machine Learning
…	…

Context helps users understand data

Class Roster

Course: MGT 500 Business Policy Semester: Spring 2010

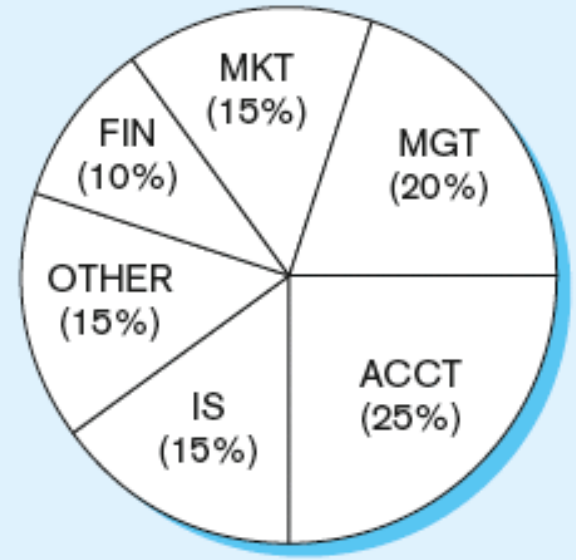
Section: 2

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

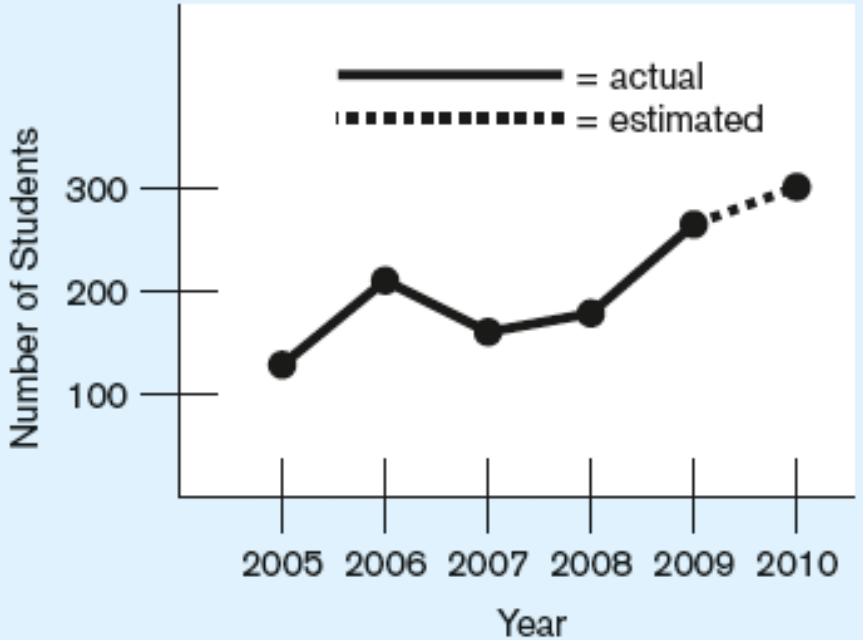
Information

Processed data that are of a meaning to the users.

Graphical displays turn data into useful information for decision making and interpretation



Percent Enrollment by Major (2010)



Enrollment Projections

Metadata

Data that describes the properties (name, type, length, range, etc.) or context (data source, storage location, etc.) of end-user data; it is data about data.

Examples of metadata

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

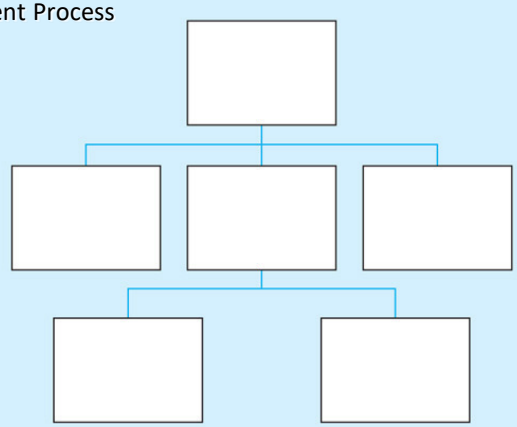
Repository: a centralized knowledge base of all data definitions, data relationships, screen and report formats, and other system components.

Database

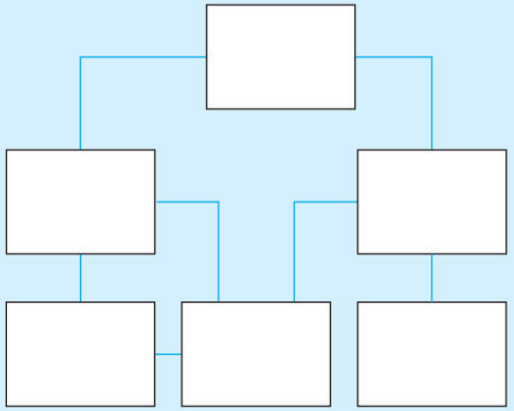
Organized collection of logically related data

- Computer based (for what we deal with)
- Controlled by a software system

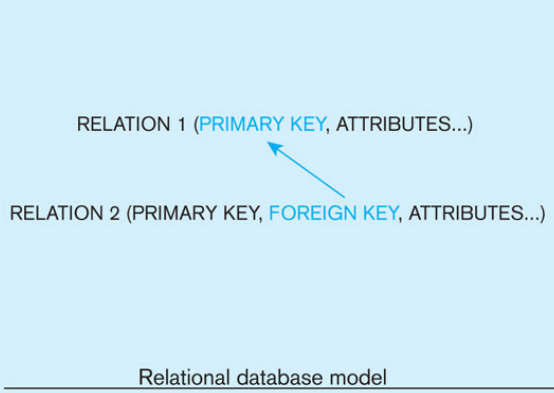
Database Architectures: major database technologies



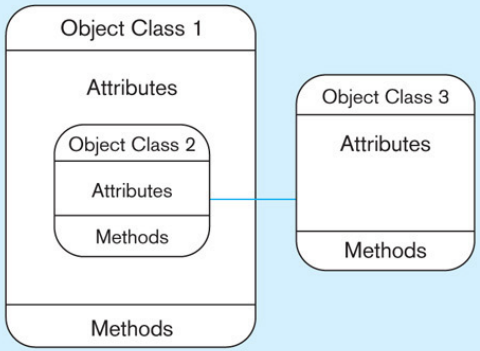
Hierarchical database model



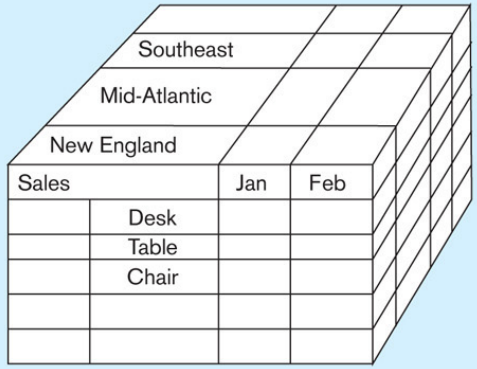
Network database model



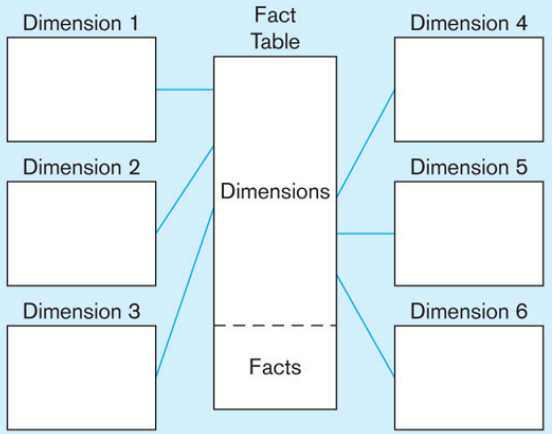
Relational database model



Object-oriented database model

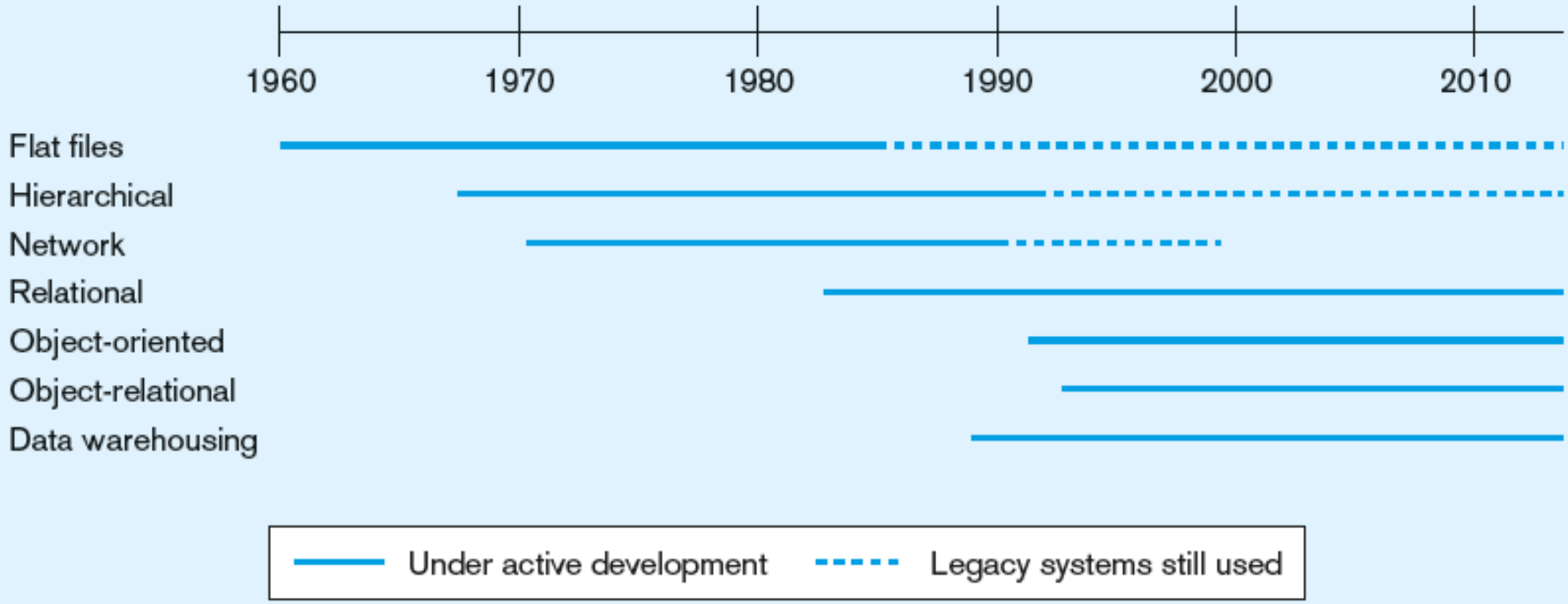


Multidimensional database model — multidimensional cube view



Multidimensional database model — star-schema view

History of database technologies



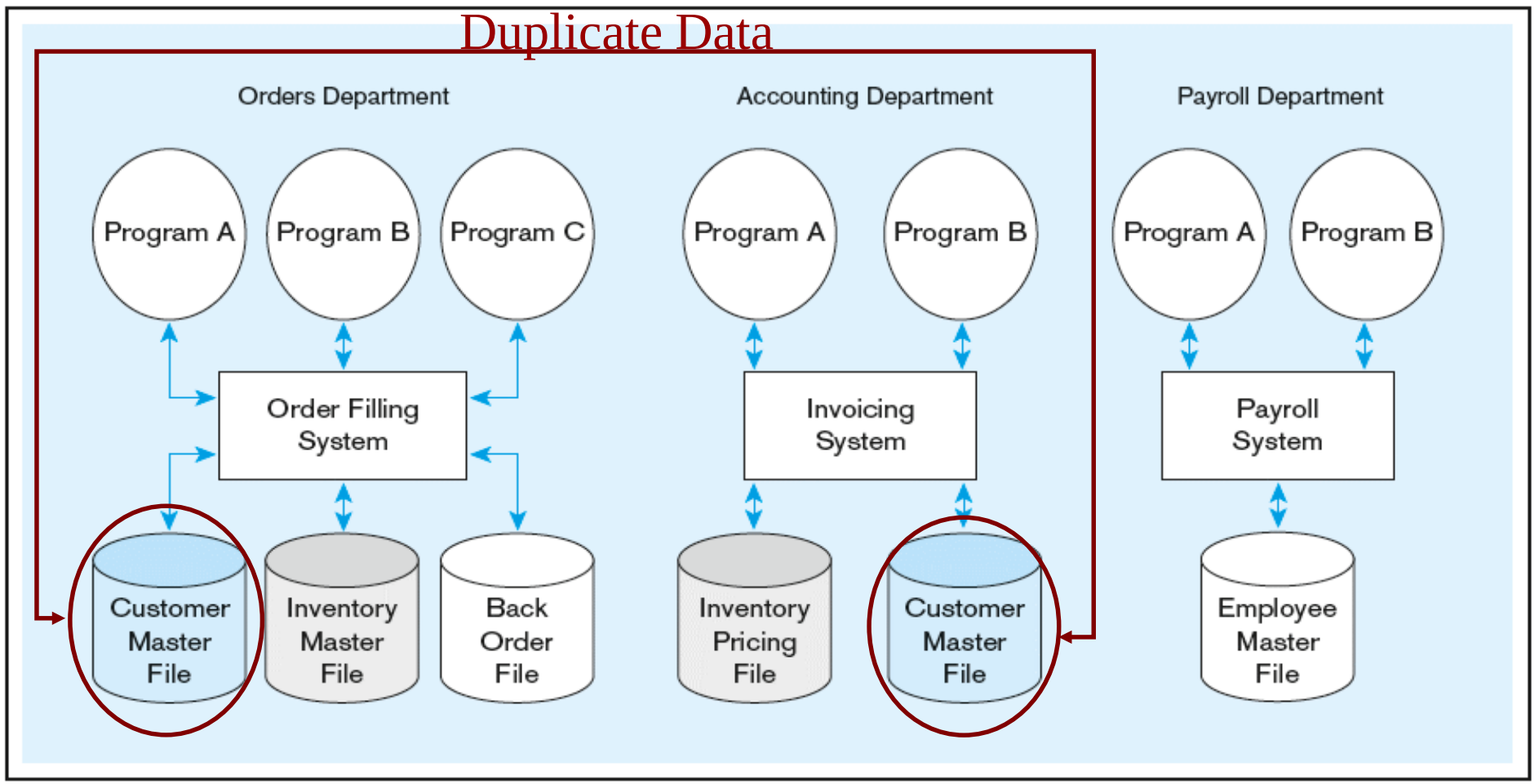
Disadvantages of File Processing

- **Program-Data Dependence**
  - All programs maintain metadata for each file they use; data are coupled with the programs that use the data
- **Duplication of Data**
  - Different systems/programs have separate copies of the same data
- **Limited Data Sharing**
  - No centralized control of data
- **Lengthy Development Time**
  - Programmers must design their own file formats
- **Excessive Program Maintenance**
  - 80% of information systems budget

Problems with Data Dependency

- Each application programmer must maintain his/her own data
- Each application program needs to include code for the metadata of each file
- Each application program must have its own processing routines for reading, inserting, updating, and deleting data
- Lack of coordination and central control
- Non-standard file formats

Old file processing systems at Pine Valley Furniture Company



Problems with Data Redundancy

- Waste of space to have duplicate data
- Causes more maintenance headaches
- The biggest problem:
  - **Data changes in one file could cause inconsistencies**
  - Compromises in ***data integrity***

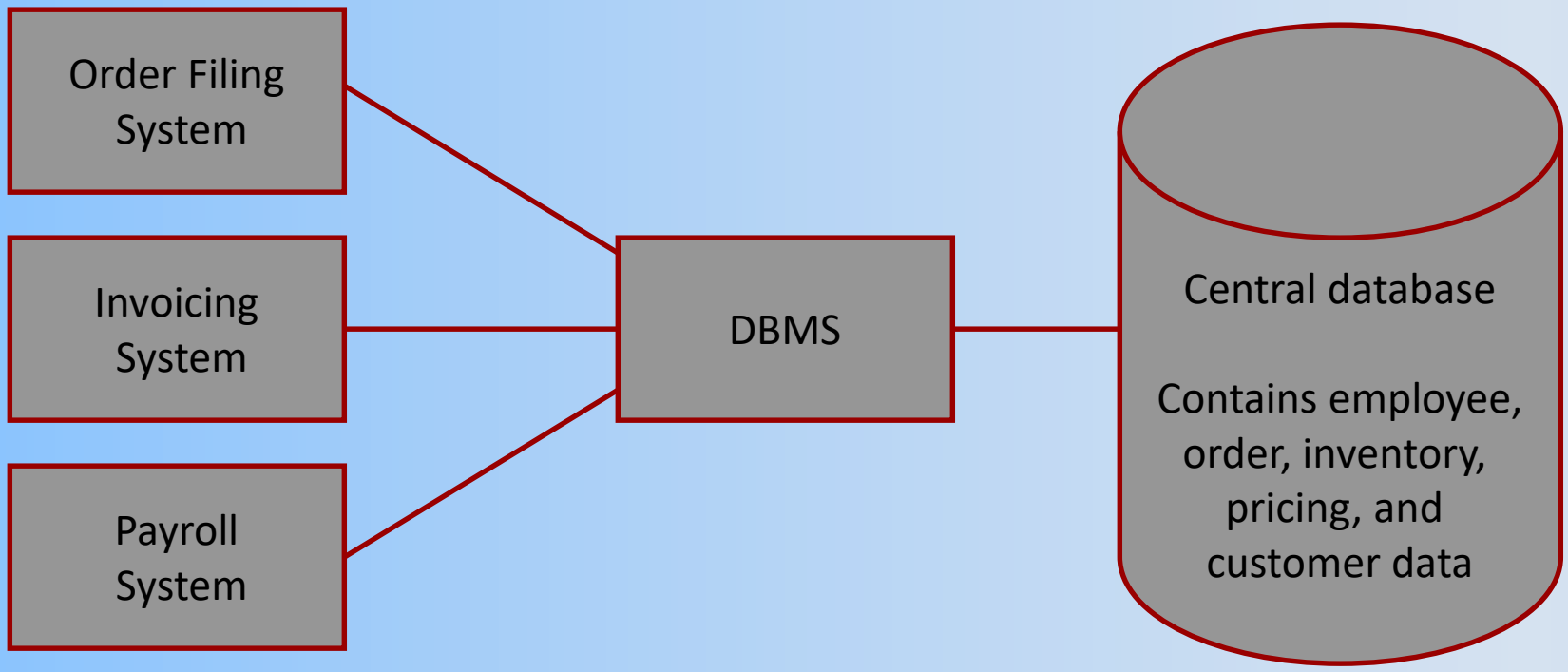
SOLUTION:
The DATABASE Approach

- Central repository of shared data
- Data is managed by a controlling agent
- Stored in a standardized, convenient form

Requires a Database Management System (DBMS)

Database Management System

A software system that is used to create, maintain, and provide controlled access to user databases



DBMS manages data resources like an operating system manages hardware resources

Advantages of the Database Approach

- Program-data independence
- Planned data redundancy
- Improved data consistency
- Improved data sharing
- Increased application development productivity
- Enforcement of standards
- Improved data quality
- Improved data accessibility and responsiveness
- Reduced program maintenance
- Improved decision support

Costs and Risks of the Database Approach

- New, specialized personnel
- Installation and management cost and complexity
- Conversion costs
- Need for explicit backup and recovery
- Organizational conflict

Elements of the Database Approach

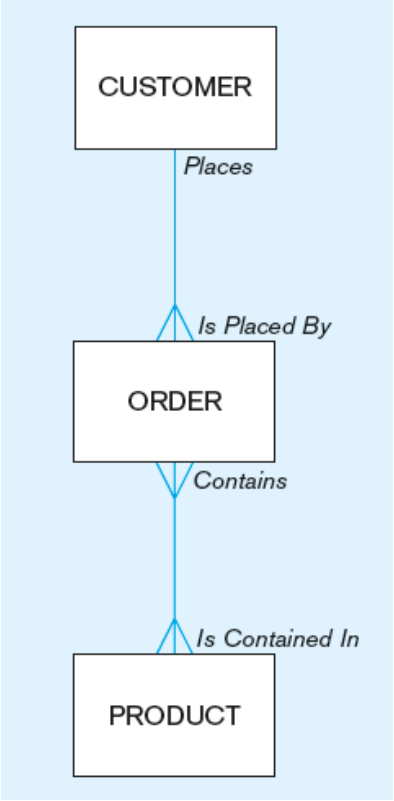
- Data model
- Entity
- Relationship
- Relational database

Data Model

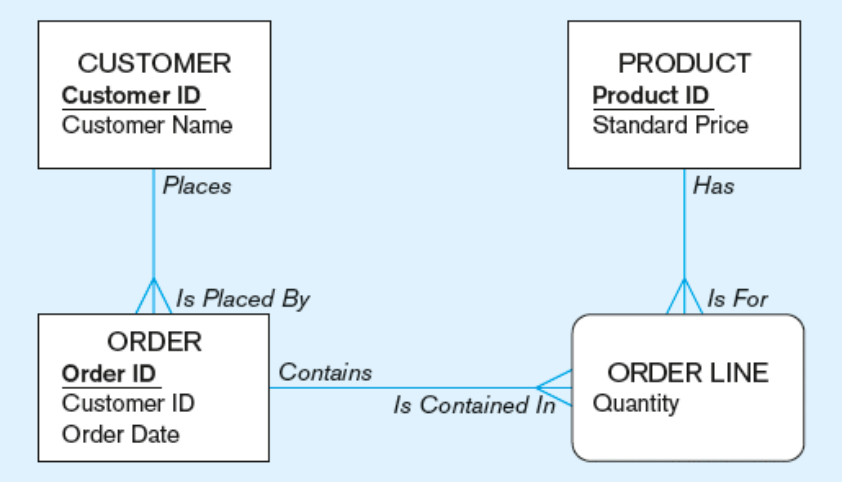
A graphical system capturing the nature and relationship of data. It has different levels.

- Enterprise Data Model: high-level entities and relationships for the organization
- Project Data Model: more detailed view, matching data structure in database or data warehouse

Segment of an enterprise data model



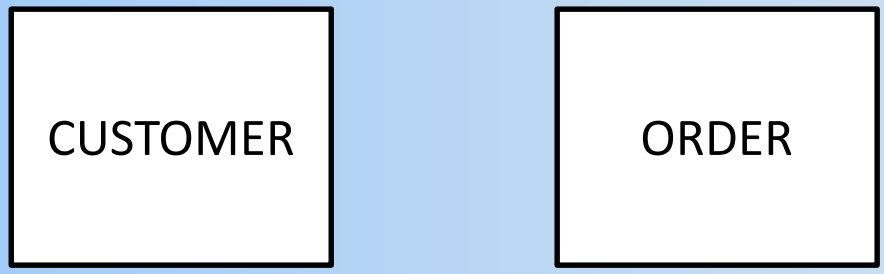
Segment of a project-level data model



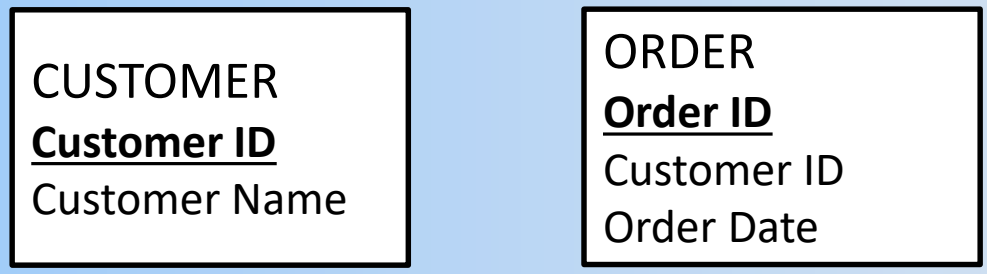
Entity

A person, a place, an object, an event or a concept about which people want to maintain data. It is

- expressed as noun



- composed of attributes



Relationship

The association between the entities

- Is described with a verb or verb phrase
- Associates the entities
- Is represented by a line or a crow's foot

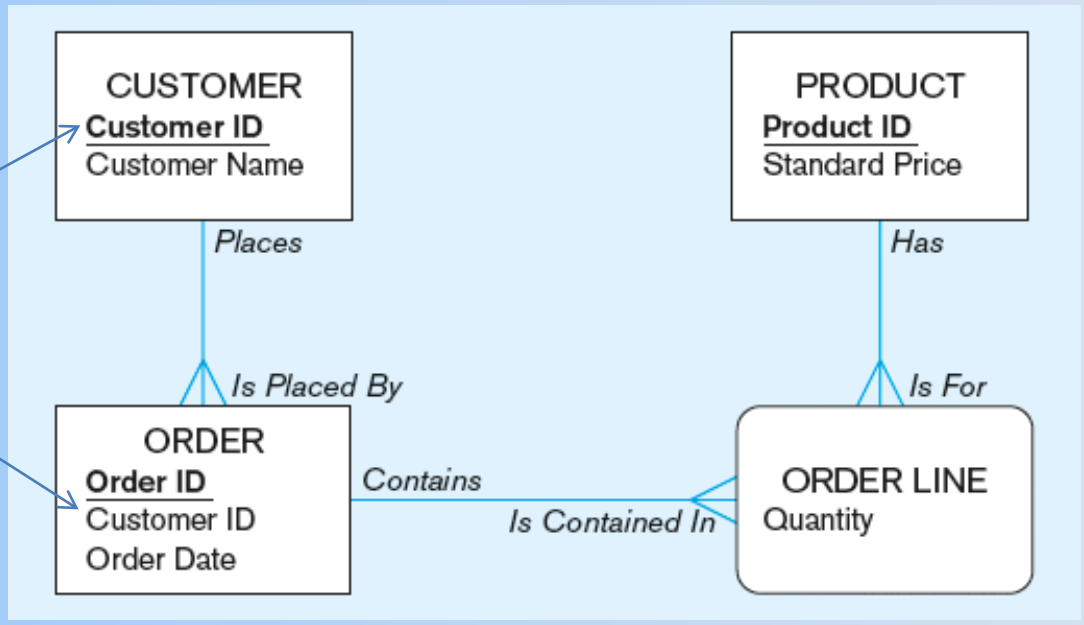
- Is usually one-to-many (1:M) or many-to-many (M:N)

Relational Database

A database that represents data as a collection of tables (relations) in which all data relationships are represented by common values in the related tables, i.e., the primary/foreign keys in the tables

Primary/foreign key pair

Customer ID is used to establish CUSTOMER and ORDER relationship



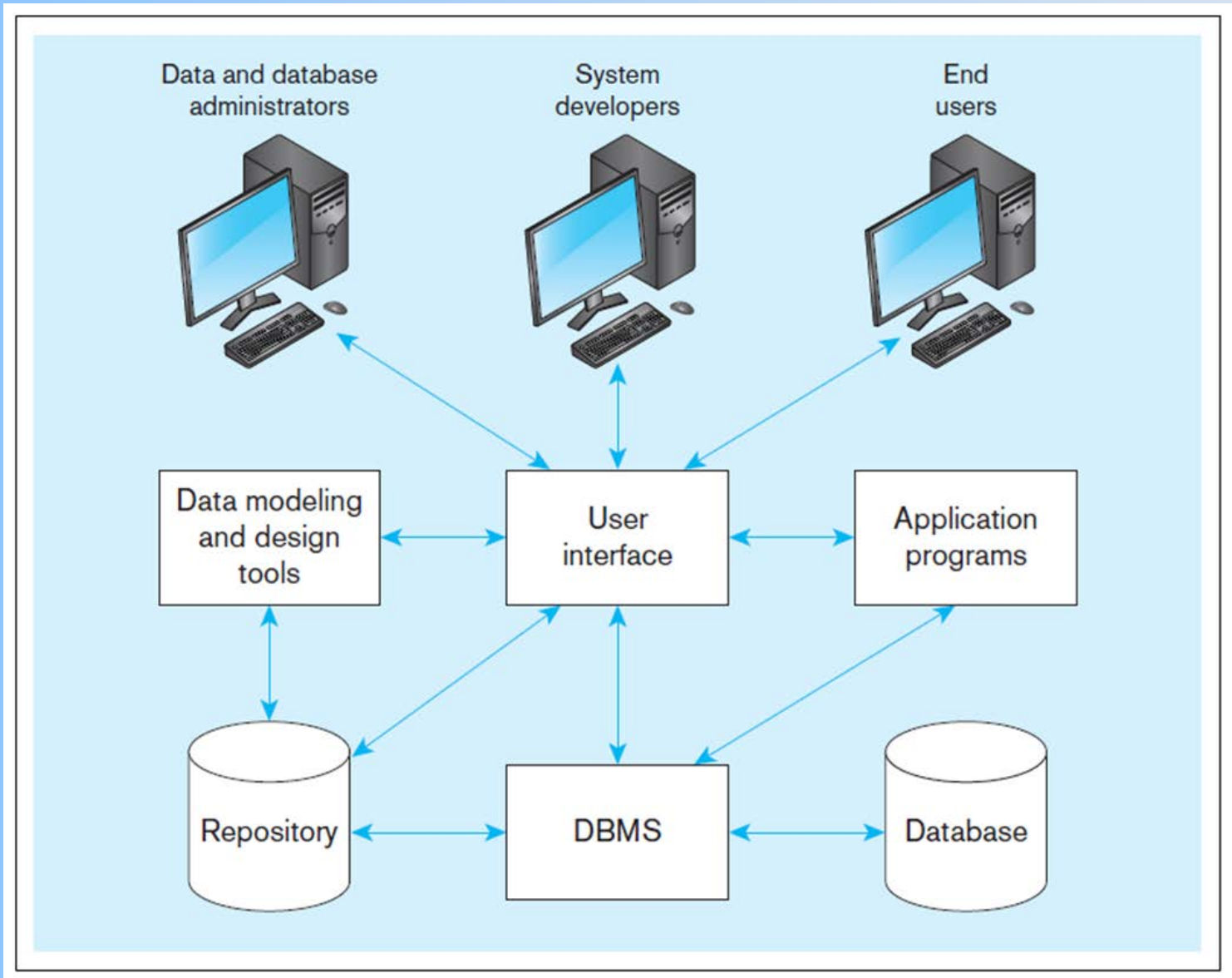
Examples of Relational Database Management Systems (RDBMS)

- Oracle
- MS SQL Server
- MySQL
- DB2
- Sybase
- Teradata
- MS Access
- (PROC SQL)

Components of the Database Environment

- **CASE Tools**–computer-aided software engineering
- **Repository**–centralized storehouse of metadata
- **Database Management System (DBMS)** –software for managing the database
- **Database**–storehouse of the data, which is organized and logically related
- **Application Programs**–software using the data (e.g., Peoplesoft, Student Center, Faculty Center, …)
- **User Interface**–text and graphical displays to users
- **Data/Database Administrators**–personnel responsible for controlling/maintaining the database
- **System Developers**–personnel responsible for designing databases and software
- **End Users**–people who use the databases and database applications

Components of the Database Environment



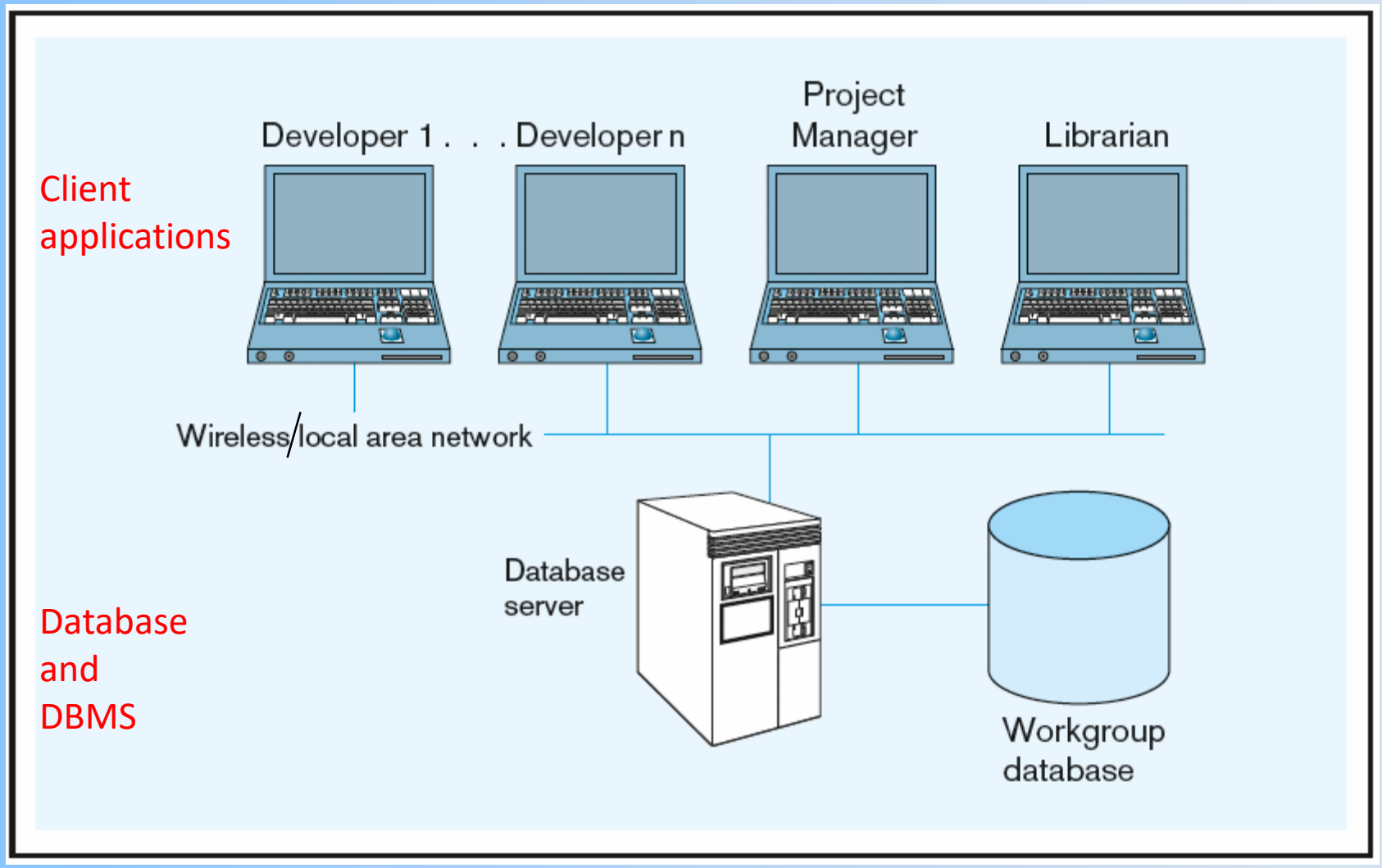
The Range of Database Applications

- Personal databases
- Two-tier Client/Server databases
- Multitier Client/Server databases
- Enterprise applications
  - Enterprise resource planning (ERP) systems
  - Data warehousing implementations

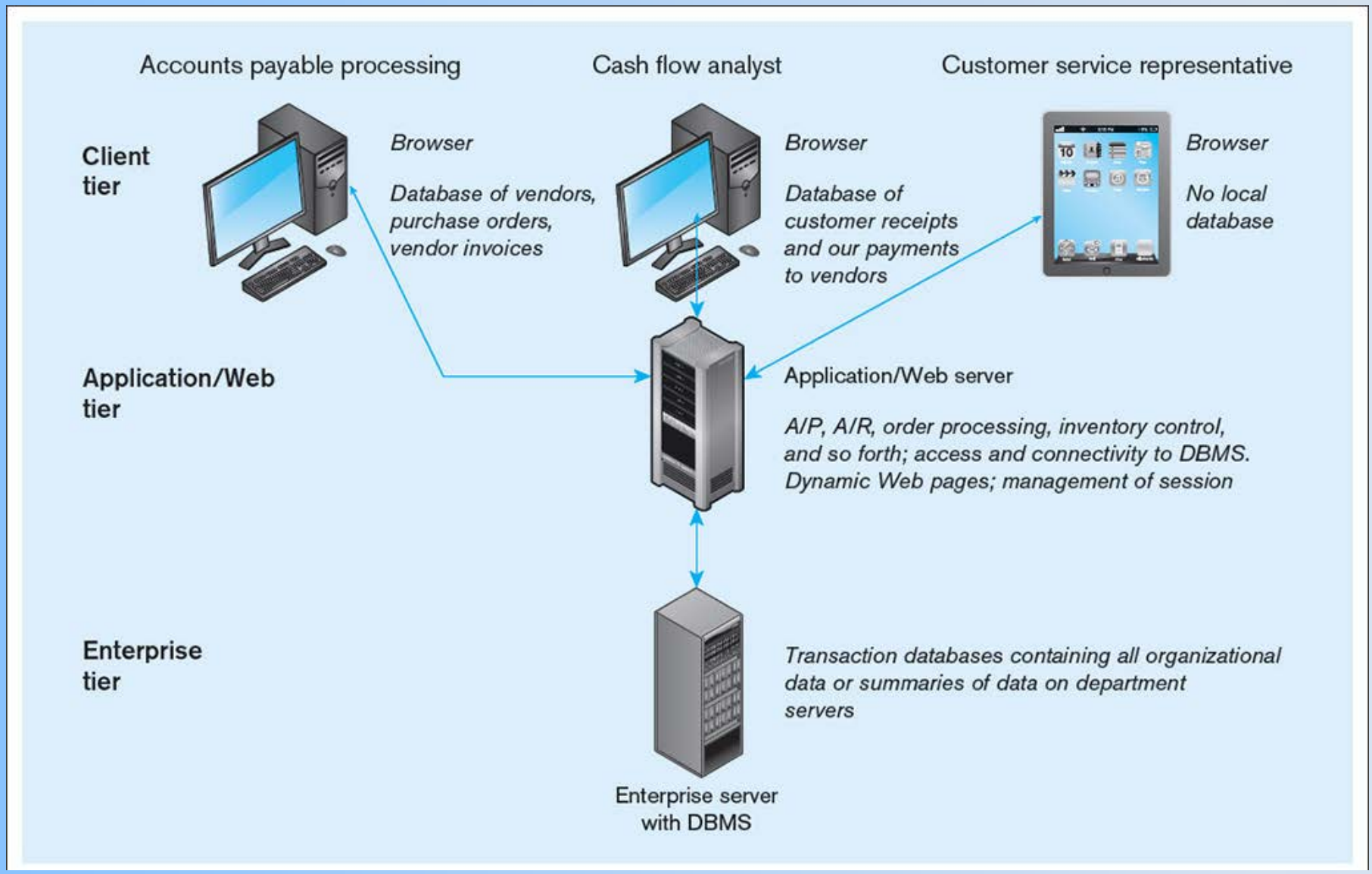
TABLE 1-5 Summary of Database Applications

Type of Database / Application	Typical Number of Users	Typical Size of Database
Personal	1	Megabytes
Two-tier	5–100	Megabytes–gigabytes
Three-tier	100–1000	Gigabytes
Enterprise resource planning	>100	Gigabytes–terabytes
Data warehousing	>100	Terabytes–petabytes

Two-tier database with local area network



Three-tiered client/server database architecture



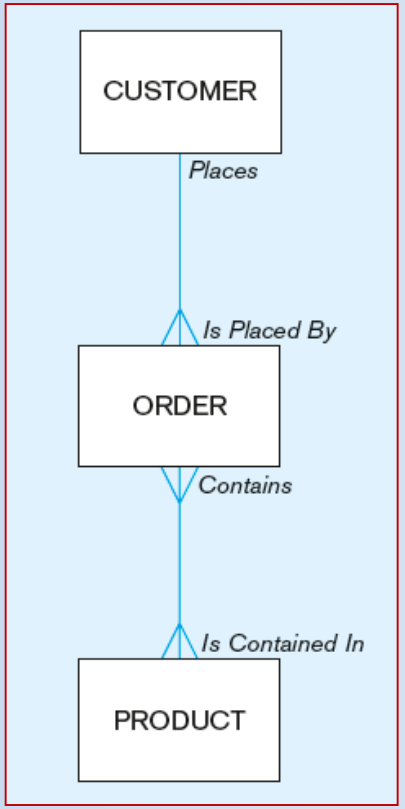
Enterprise Database Applications

- Enterprise Resource Planning (ERP)
  - An integration of all enterprise functions (e.g., manufacturing, finance, sales, marketing, inventory, accounting, human resources). It works with the current operational data of the enterprise. The performance of a database is highly important.
- Data Warehouse
  - An integrated decision support database, derived from various operational databases. It also provides historical data to identify patterns and trends and answer strategic business questions.

Database Development Process

Enterprise Data Model

- First step in database development
- Specifies scope and general content
- Overall picture of organizational data at high level of abstraction
- Entity-relationship diagram
- Descriptions of entity types
- Relationships between entities
- Business rules



Example: business function-to-data entity matrix

Business Functions \ Data Entity Types	Customer	Product	Raw Material	Order	Work Center	Work Order	Invoice	Equipment	Employee
Business Planning	X	X						X	X
Product Development		X	X		X			X	
Materials Management		X	X	X	X	X		X	
Order Fulfillment	X	X	X	X	X	X	X	X	X
Order Shipment	X	X		X	X		X		X
Sales Summarization	X	X		X			X		X
Production Operations		X	X	X	X	X		X	X
Finance and Accounting	X	X	X	X	X		X	X	X

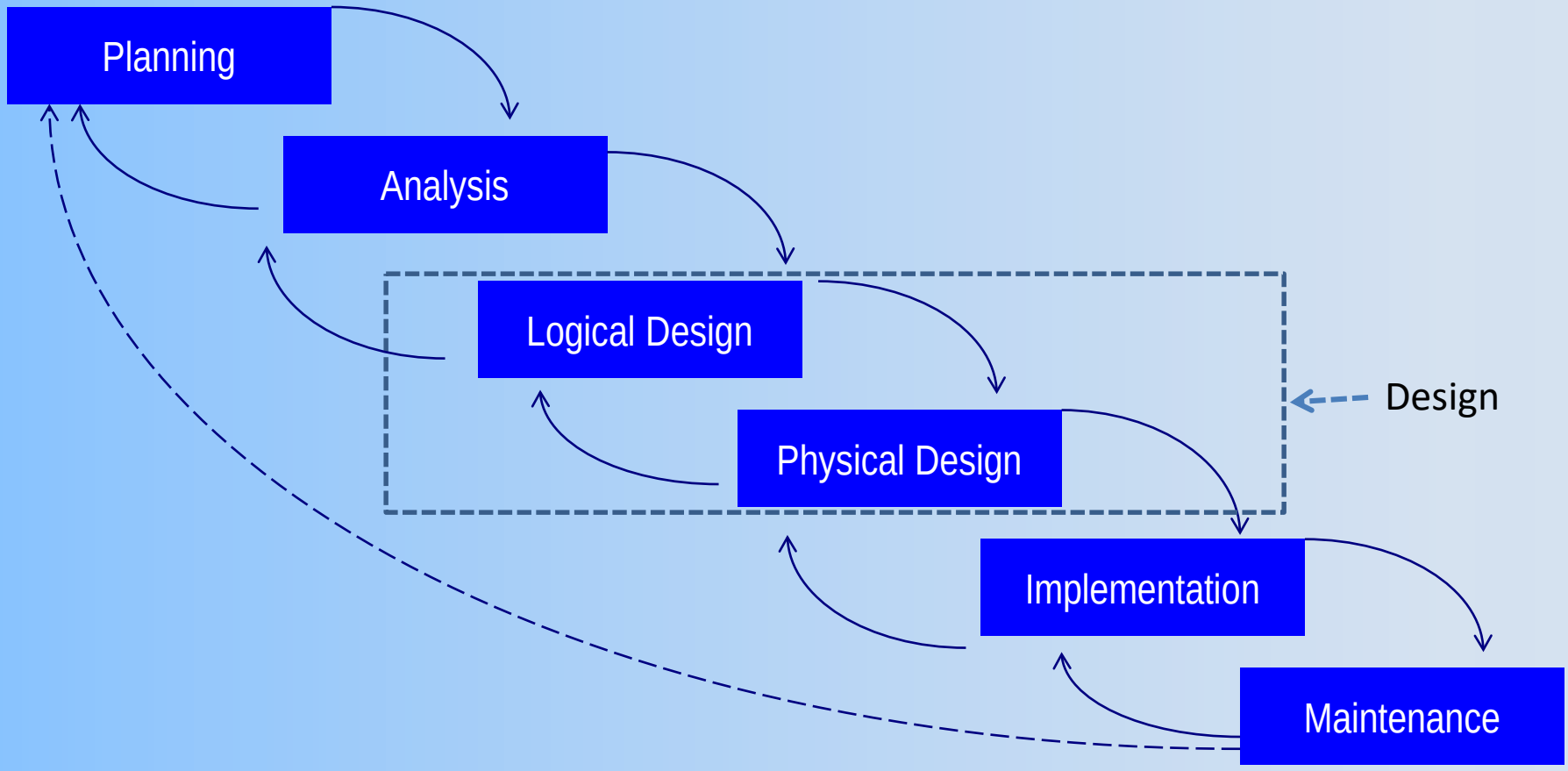
X = data entity is used within business function

Two Approaches to Database and IS Development

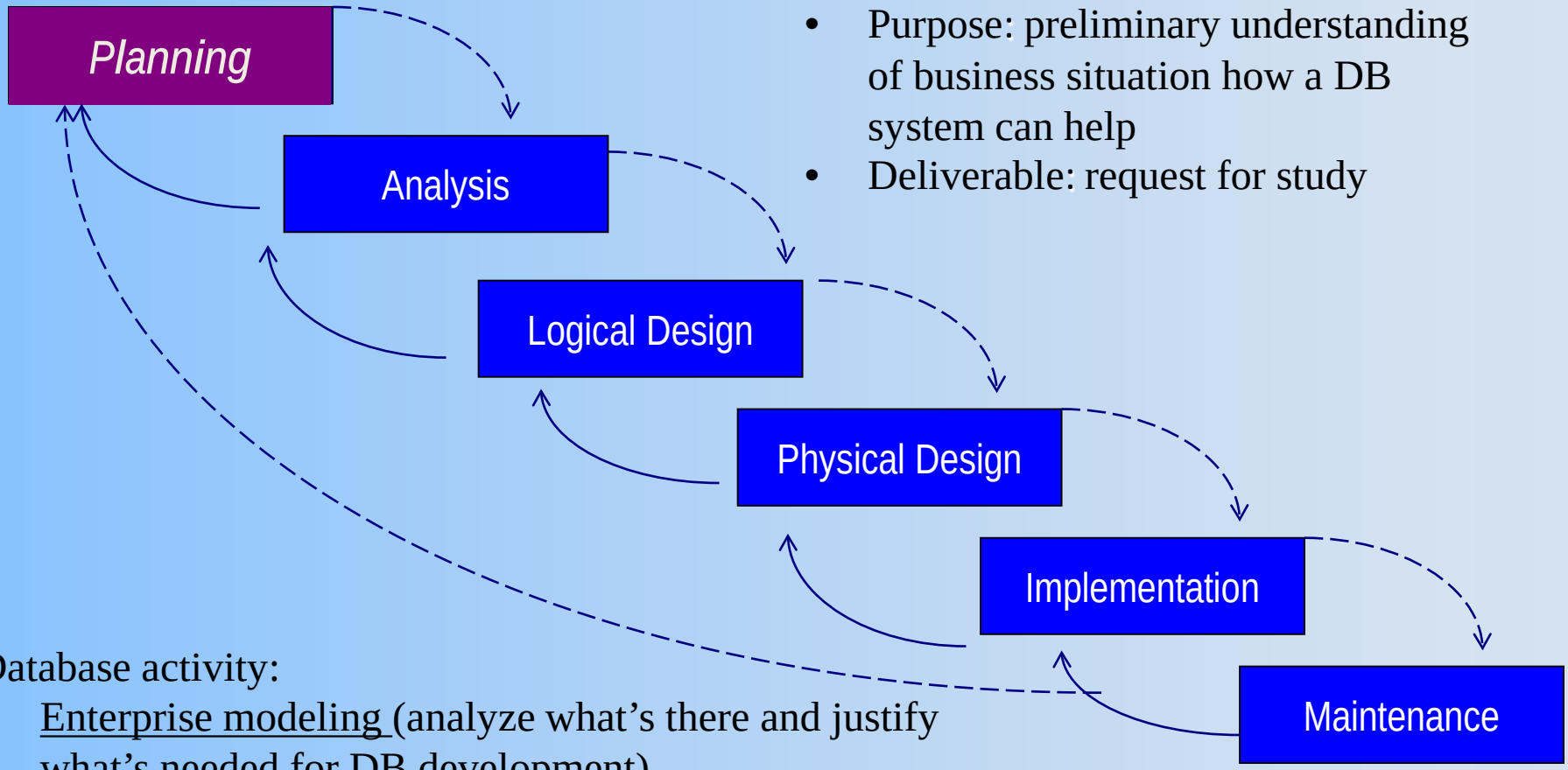
- Systems Development Life Cycle (SDLC): the traditional methodology for system development, which is a complete set of steps that developers follow to develop an information system
  - Detailed, well-planned development process
  - Time-consuming but comprehensive
  - Long development cycle

- Prototyping: an interactive process of systems development in which requirements are converted to a working system that is continually revised through close work between the developer and users
  - Rapid application development (RAD)
  - Cursory attempt at conceptual data modeling
  - Define database during development of initial prototype
  - Repeat implementation and maintenance activities with new prototype versions

Systems Development Life Cycle



Systems Development Life Cycle (cont'd)

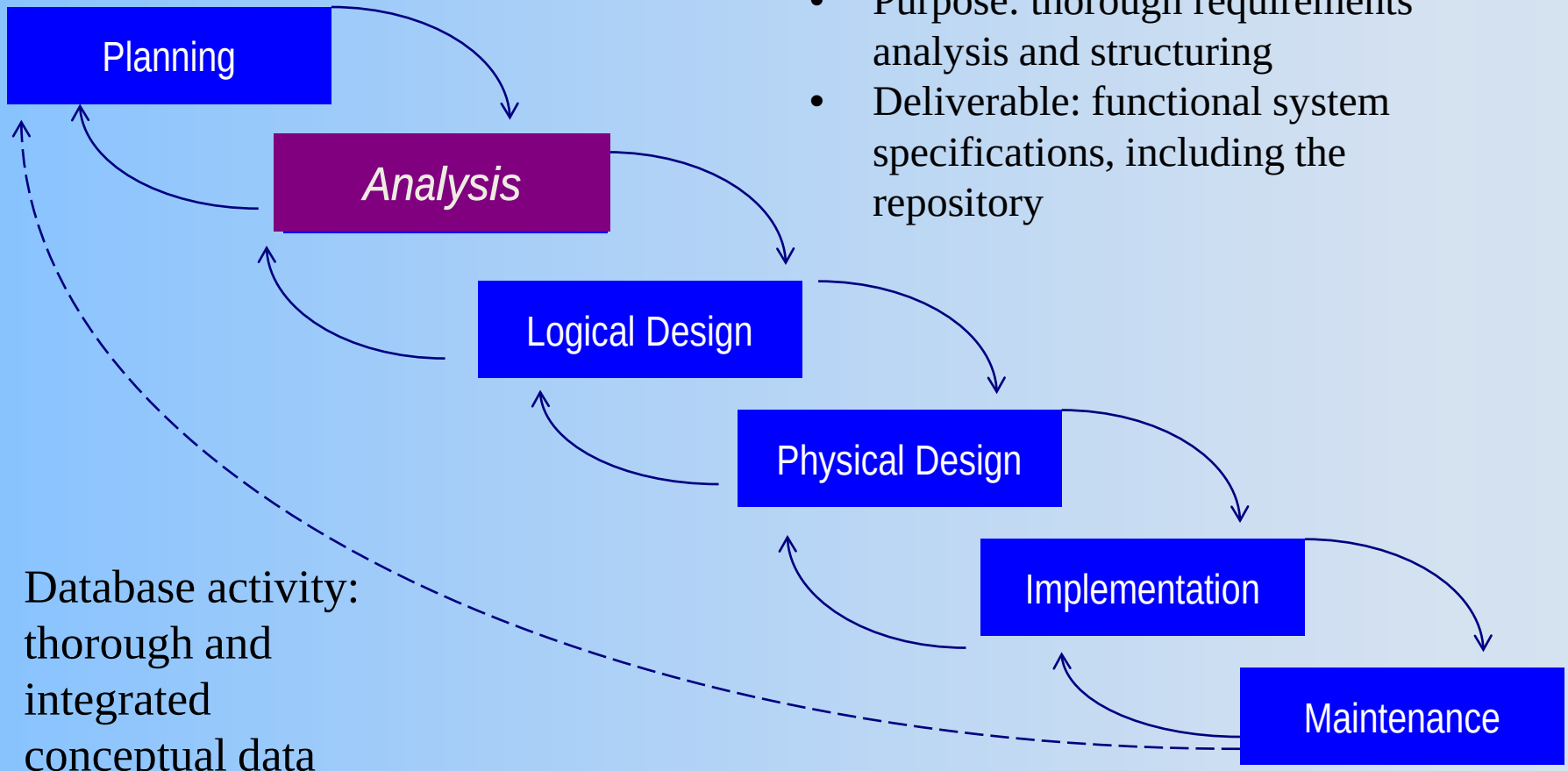


- Purpose: preliminary understanding of business situation how a DB system can help
- Deliverable: request for study

Database activity:

- Enterprise modeling (analyze what's there and justify what's needed for DB development)
- Early conceptual data modeling (identify the scope of DB and overall data requirements)

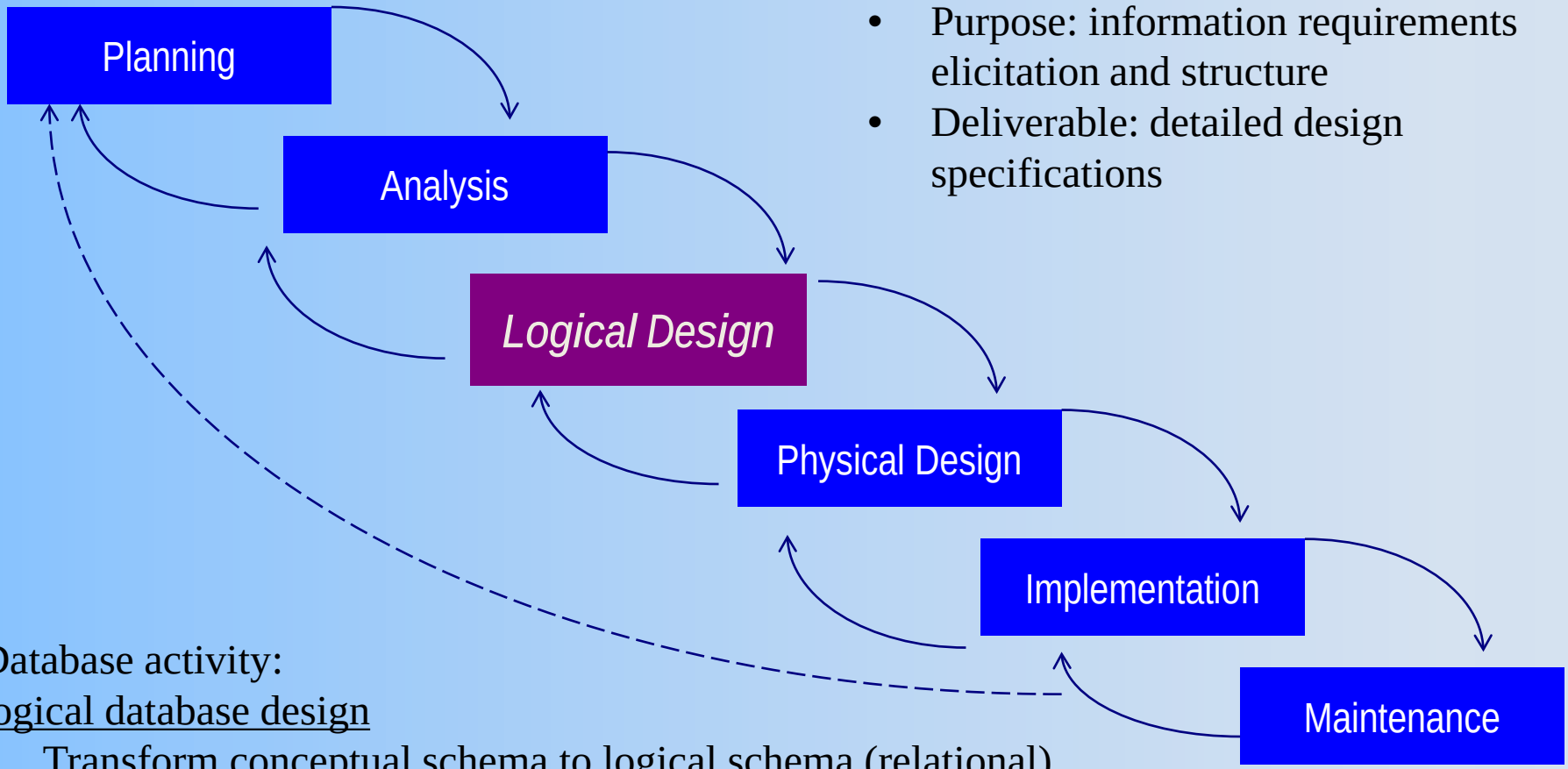
Systems Development Life Cycle (cont'd)



- Purpose: thorough requirements analysis and structuring
- Deliverable: functional system specifications, including the repository

Database activity: thorough and integrated conceptual data modeling (which produces a conceptual schema: a detailed, technology-independent specification of the overall structure of organizational data, including all entities, relationships, attributes, and business rules)

Systems Development Life Cycle (cont'd)

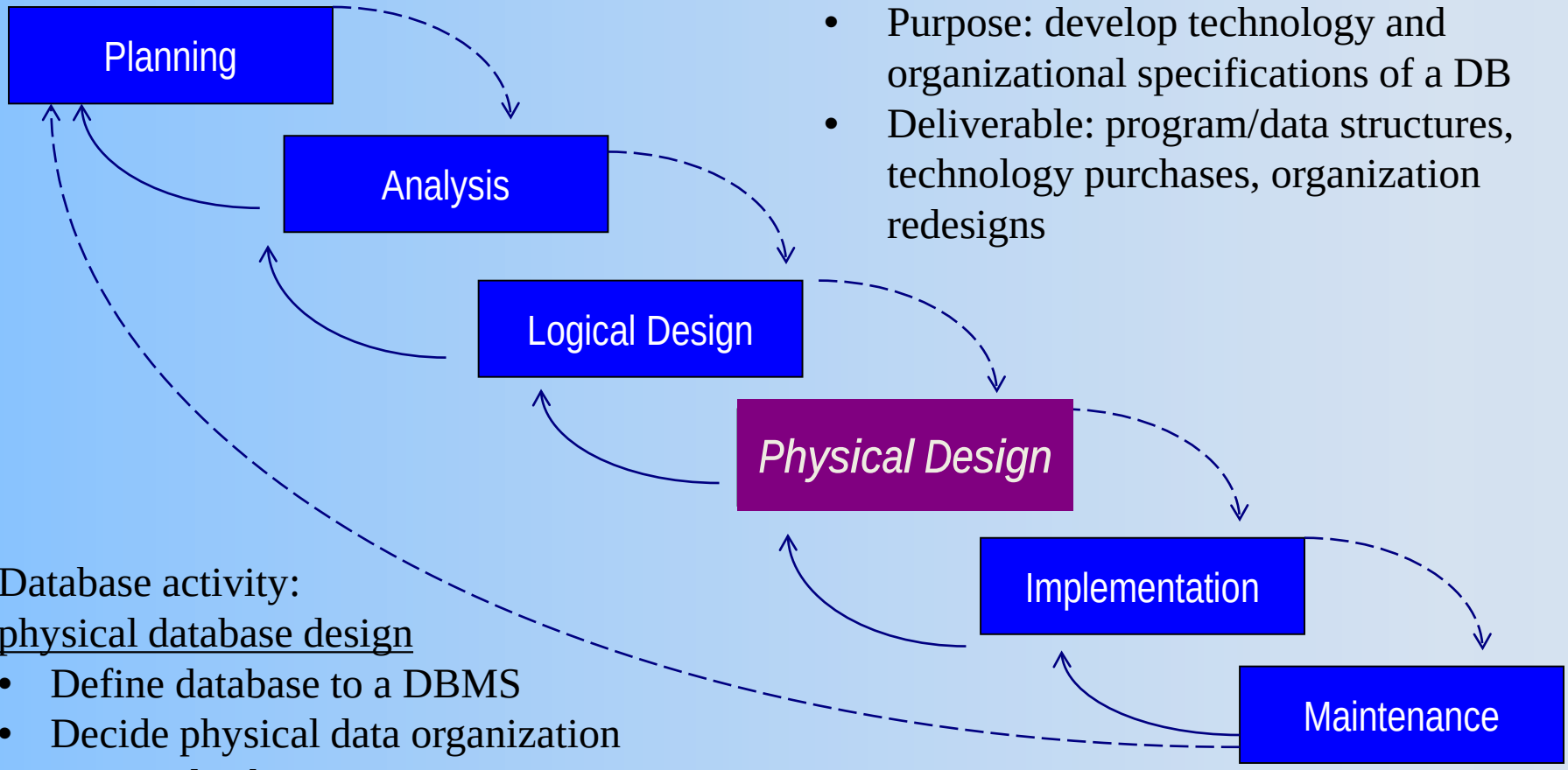


- Purpose: information requirements elicitation and structure
- Deliverable: detailed design specifications

Database activity:
<u>logical database design</u>

- Transform conceptual schema to logical schema (relational)
- Analyze transactions, forms, displays, views, data integrity and security
- Use normalization rules

Systems Development Life Cycle (cont'd)



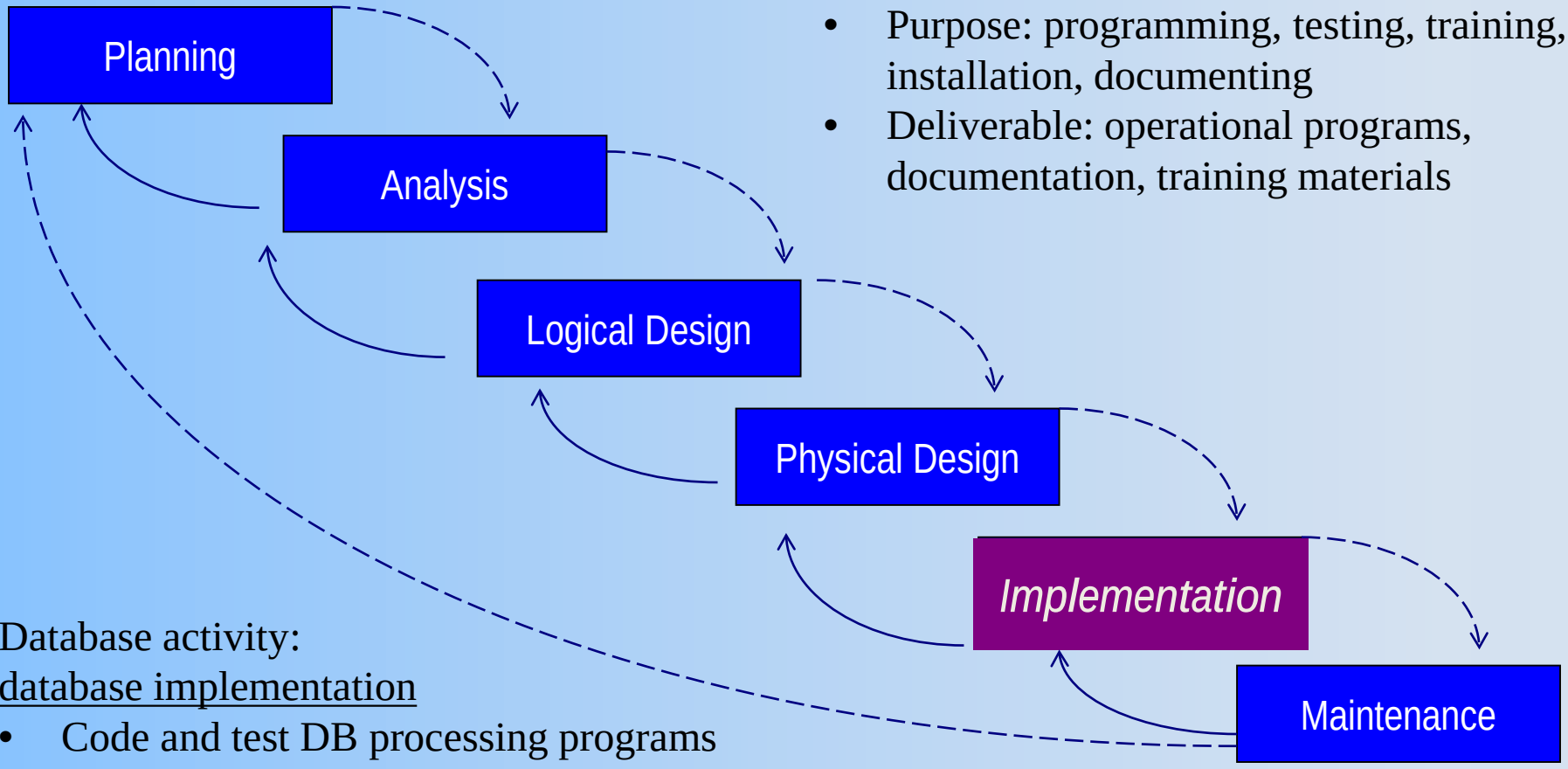
- Purpose: develop technology and organizational specifications of a DB
- Deliverable: program/data structures, technology purchases, organization redesigns

Database activity:
<u>physical database design</u>

- Define database to a DBMS
- Decide physical data organization
- Design database processing programs

(<u>Physical schema:</u> specifications for how data from a logical schema are stored in a computer's secondary memory by a DBMS)

Systems Development Life Cycle (cont'd)

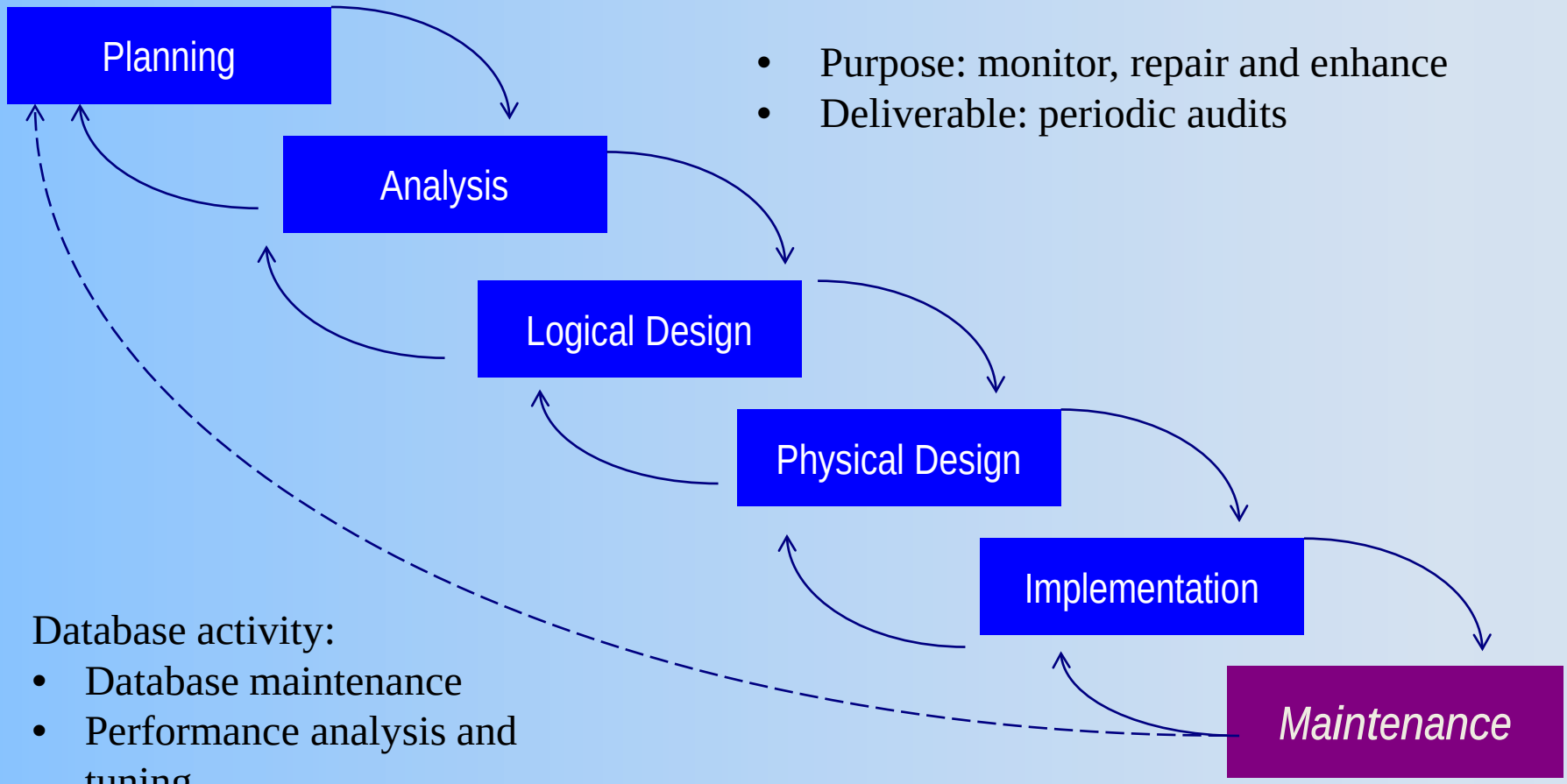


- Purpose: programming, testing, training, installation, documenting
- Deliverable: operational programs, documentation, training materials

Database activity:
<u>database implementation</u>

- Code and test DB processing programs
- Write DB documentation and training materials
- Install DB and load data (old data conversion and/or new data entering)

Systems Development Life Cycle (cont'd)

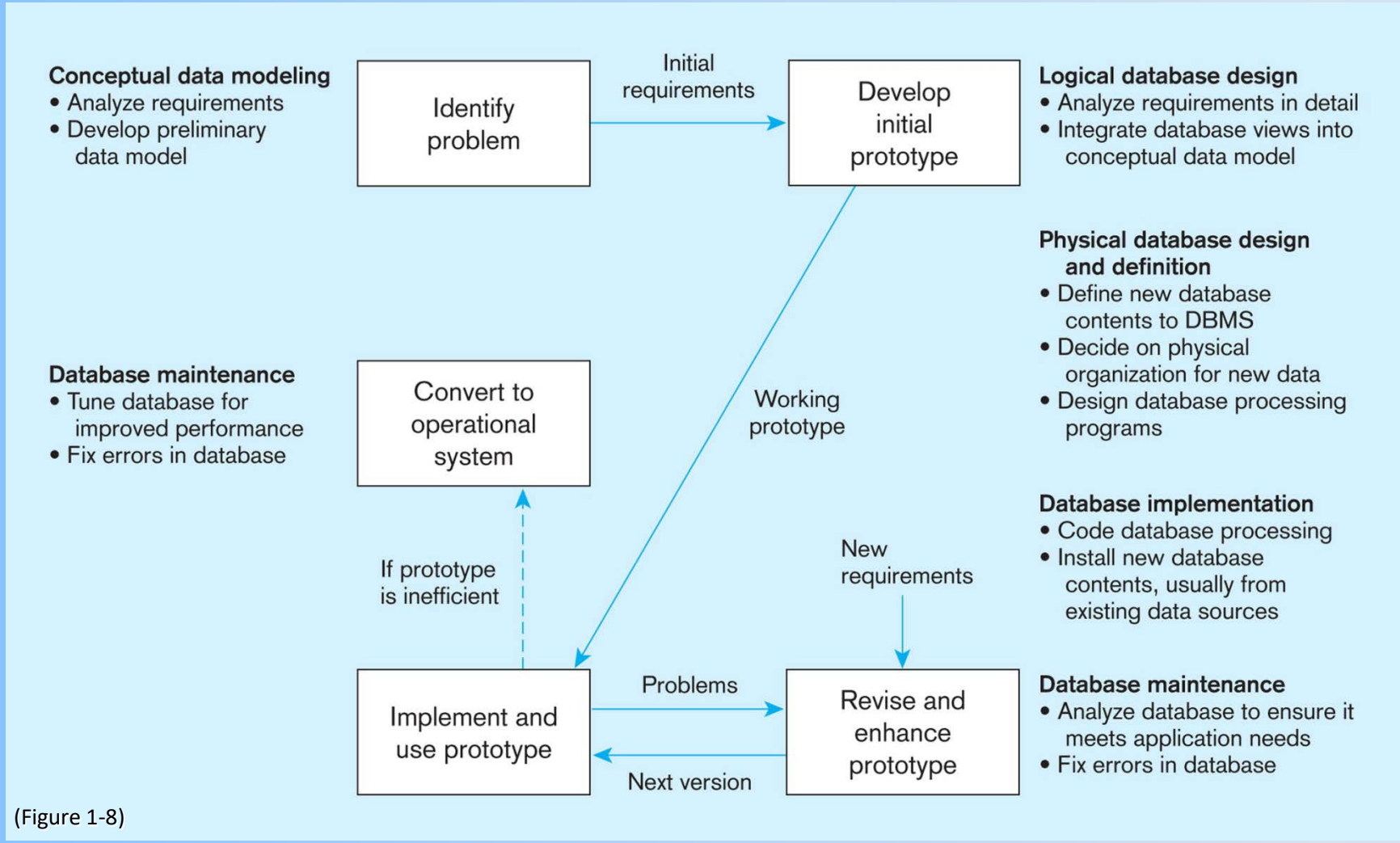


- Purpose: monitor, repair and enhance
- Deliverable: periodic audits

Database activity:

- Database maintenance
- Performance analysis and tuning
- Error corrections

Prototyping Database Methodology



(Figure 1-8)

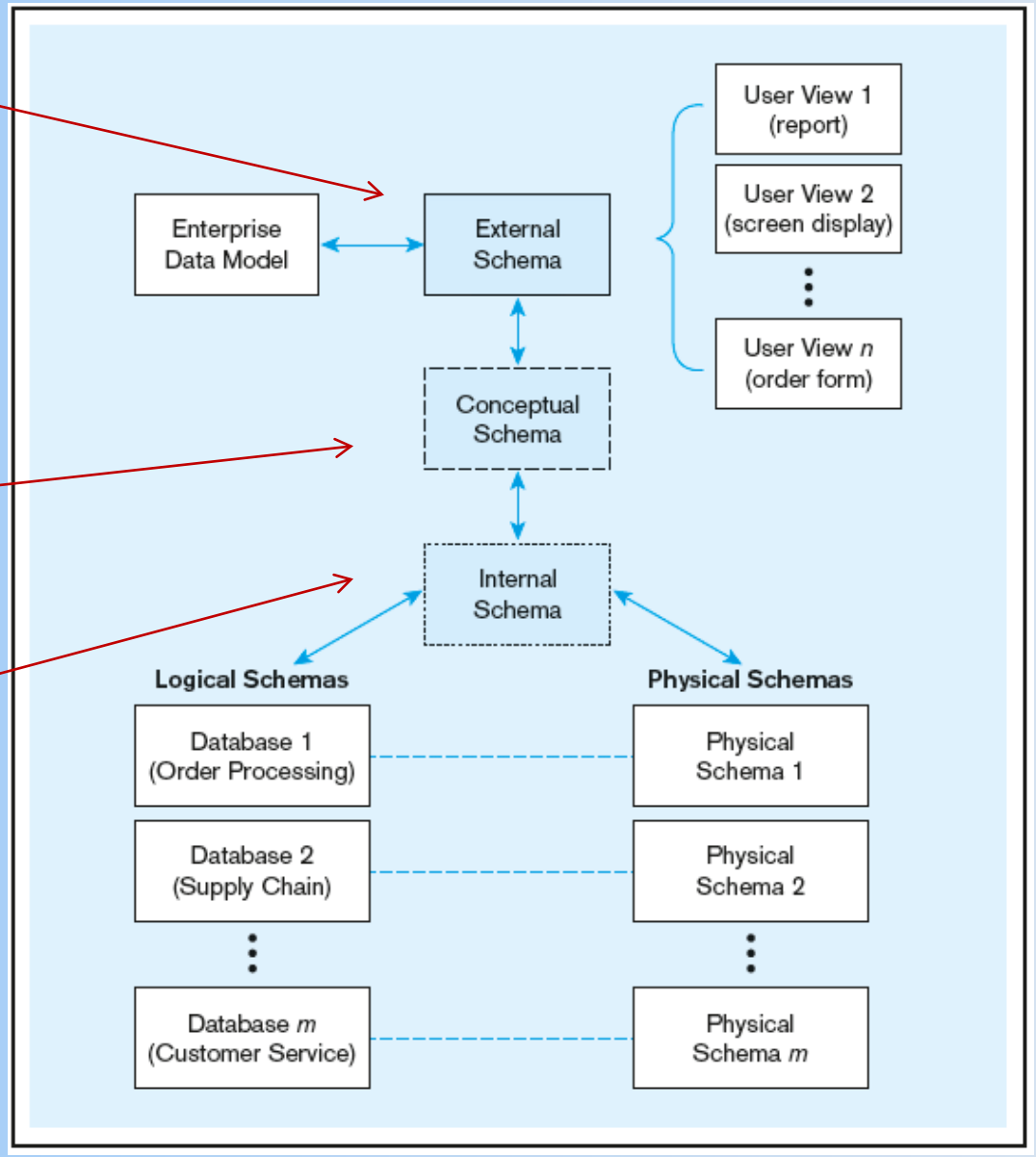
Three-schema Architecture

(Schema: a database structure containing descriptions of database objects)

Different people have different views of the same database…these are the external schema

The view of data administrator

The internal schema is the underlying design and implementation



Database Schema

- External Schema
  - User Views
  - Subsets of Conceptual Schema
  - Can be determined from business-function/data entity matrices
  - DBA determines schema for different users
- Conceptual Schema
  - E-R models–covered in Chapters 2 and 3
- Internal Schema
  - Logical structures–covered in Chapter 4
  - Physical structures–covered in Chapter 5