https://selldocx.com/products
/solution-manual-data-modeling-and-database-design-2e-umanath

Chapter 1 – Database Systems: Architecture and Components

Chapter 1 Objectives

After completing this chapter, the student will understand:

- The difference between data, metadata, and information and highlight how metadata serves as the lens by which data can become information
- How data management is a discipline that focuses on the proper acquisition, storage, maintenance, and retrieval of data
- The characteristics of file-processing systems and their limitations
- How the ANSI/SPARC Three-Schema Architecture constitutes the solution to the problems plaguing file processing systems
- What constitutes a database, a database management system, and a database
- The difference between a model and a data model
- The role of data models in database design
- The role of the three data models (conceptual, logical, and physical) in the database design life cycle portrayed in Figure 1.7

Chapter 1 Overview

This chapter begins with an introduction to the rudimentary concepts of data and how information emerges from data when viewed through the lens of metadata. Next, the discussion addresses data management, contrasting file-processing systems with database systems. This is followed by brief examples of desktop, workgroup, and enterprise databases. The chapter then presents a framework for database design in Figure 1.7 that describes the conceptual, logical, and physical tiers of data modeling and their roles in the database design life cycle. This framework serves as the roadmap to guide the reader through the remainder of the book. Finally, an example walks one through the cradle to grave life cycle of data modeling and database design in a nutshell.

Chapter 1 Key Terms

Data	Unorganized facts about things, events, activities, and transactions.
Information	Data that has been organized into a specific context such that it has value to its recipient.
Metadata	A lens through which data takes on specific meaning and yields information.
Data element	The smallest unit of data.
Record type	A group of related data elements treated as a unit.
Record	A set of values for the data elements constituting a record type.
File	A collection of records.
Data Set	Another term for a file.
Sequential access	An access approach where in order to get to the *nth* record in a data set it is necessary to pass through the previous *n-1* records in the data set.
Direct access	An access approach where it is possible to get to the *nth* record in a data set without having to pass through the previous *n-1* records in the data set.
File-processing system	The predecessor of a database system where records were stored in separate non-integrated files.
Data integrity	Ensures that data is correct, consistent, complete, and current.
ANSI/SPARC three-schema architecture	A collection of three separate schemas or views for describing data in a database: (a) external schema (or application view), (b) conceptual schema (or logical view) and (c) internal schema (or physical view).
Conceptual schema	Represents the global conceptual view of the structure of the entire database for the community of users. It is independent of any particular data structure or data representation.
External schema	Consists of a number of different user views or subschemas, each describing portions of the database of interest to a particular user or group of users. The external schema describes the data corresponding to part of the conceptual schema as seen by one or more users or programs.

Internal schema	Describes the physical structure of the stored data and the mechanism used to implement the access strategy. As opposed to the conceptual schema and external schema, which are technology-independent, the internal schema is technology-dependent.
Data independence	The ability to modify a schema definition in one level without affecting a schema definition in a higher level. For example, the conceptual schema insulates user views in the external schema from changes in the physical storage structure of the data in the internal schema.
Physical data independence	The ability to modify the internal schema without causing the application program in the external schema to be rewritten.
Logical data independence	The immunity of a user view from changes in the other user views.
Database	A self-describing collection of integrated files consisting of (1) users' data, (2) metadata, and (3) overhead data.
Database management system	A collection of general-purpose software that facilitates the processes of defining, constructing, and manipulating a database for various applications.
Distributed database	A collection of multiple logically interrelated databases that may be geographically dispersed over a computer network.
Distributed database management system	Software that manages a distributed database while rendering the geographical distribution of the data transparent to the user community.
Data warehouse	A collection of data designed to support management decision making. A data warehouse contains a wide variety of data that present a coherent picture of business conditions at a single point in time.
Data definition language (DDL)	The component of a database management system used to create the structure of database objects such as tables, views, assertions, domains, schemas, etc.
Data control language (DCL)	The component of a database management system used to control user access, facilitate backup and recovery from failures, and insure that users access only the data they are authorized to use.

Data manipulation language (DML)	The component of a database management system product that facilitates the retrieval, insertion, deletion, and modification of data in a database.
Data dictionary	The component of a database system that stores metadata that provides such information as the definitions of data items and their relationships, authorizations, and usage statistics.
Data repository	A collection of metadata about data models and application program interfaces.
Data model	A representation of a real-world phenomenon that makes use of descriptors.
Universe of interest	The aspect of the real world represented by the database.
Requirements specification	The initial step in the database design process where existing documents and systems are reviewed and prospective users are interviewed in an effort to identify the objectives to be supported by the database system.
Business rules	User-specified restrictions on the organization's activities (business processes) that must be reflected in the database or database applications.
Business rule (from Chapter 2)	A short statement of a specific condition or procedure relevant to the universe of interest being modeled expressed in a precise, unambiguous manner.
Conceptual data modeling	Involves describing the structure of the data to be stored in the database without specifying how it will be physically stored.
Logical data modeling	Involves refining the conceptual data model to (a) the point where it is more compatible with the technology intended for implementation and (b) eliminate data redundancy problems.
Physical data modeling	Involves transforming the logical data model into a form that can be implemented by some DBMS product.

Chapter 1 Solutions

1. What is the difference between data, metadata, and information?
   Answer. On this question, it is important to make sure that students read the first paragraph on page 2 in its entirety so that they can see how metadata serves as a type of lens that allows us to obtain information from what is otherwise just data. In essence, this captures the relationship among the three terms.

2. Demonstrate your understanding of data, metadata, and information using an example.
   Answer. Perhaps we did not ask the question as well as we could have here. What we are looking for is an example in a somewhat different context from what appears in the text. One possible example involves the strings 2/1/06 – 2/10/06. At first glance, this appears to include the dates from February 1, 2006 through February 10, 2006. However, in Europe, the dates are really January 2, 2006 through October 2, 2006. Without knowing the format of the dates, there is little meaning to the strings 2/1/06 – 2/10/06. As another example, consider a collection of "Customer Satisfaction Survey" responses as raw data. We can have terabytes of such data that would be worthless unless we have some sort of specification in a form of metadata that would reveal the true meaning of the responses. Only if we have such metadata that describes the data that has been collected would we be able to interpret customer responses and derive business value through information that could enable a business process to analyze and improve customer satisfaction.

3. Describe the four actions involved in data management.
   Answer. The four actions, often abbreviated as CRUD, are the creation of data, the retrieval of data, the modification or updating of data, and the deletion of data. We use DBMS tools like DDL and DML to create data. In a database system, it is important that data retrieval be fast and efficient. This can be achieved through efficient query design and through optimized DBMS query execution. Updates to data must not leave the database in an inconsistent state; the DBMS must provide facilities to check constraints that follow from business rules. Care must be exercised when deleting data. The DBMS needs to allow for the possibility of preventing a deletion if there is dependent data in the database in order to preserve database consistency.

4. Distinguish between sequential access and direct access. Give an example of a type of application for which each is particularly appropriate.
   Answer. Sequential access requires that one pass through the first n-1 records in a data set to get to the nth record. Some examples of applications that lend themselves to sequential access appear in the second paragraph in Section 1.2. Direct access allows for the retrieval of the nth record in a data set without having to look at the previous n-1 records. "Behind the scenes" this is often accomplished by the use of hashing algorithms where, for example, the physical location of a record is determined by taking its identifier (e.g., an account number) and dividing it by some prime number and using the remainder to determine the physical location of the record on the direct access storage device. Direct access is commonly used for ad hoc querying (e.g., perhaps an advisor might want to look at the demographic data the university maintains on a student prior to speaking with him or her about what courses to take during a given semester).

5. Identify a common task in a payroll system for which sequential access is more appropriate than direct access, and explain why this is so.
   Answer. Printing monthly (weekly, bi-weekly) payroll checks is the classic example since typically, virtually every employee receives a monthly (weekly, bi-weekly) payroll check.

6. What is the difference between a serial collection of data and a sequential collection of data? Which can be used for direct access?
   Answer. A serial (unordered) collection of data cannot be used meaningfully for sequential access. Although the text does not state this, certain kinds of sequential files can also be accessed directly. These are called indexed sequential files, which with the use of an external index, allow a file organized sequentially to be accessed directly.

7. What is the purpose of an external index?
   Answer. An external index allows data to be accessed in a logical order that differs from the physical order in which it is stored.

8. What is data integrity, and what is the significance of a lack of data integrity?
   Answer. Data integrity essentially involves the accuracy of the data in a database. When data integrity exists, it means that data values are correct, consistent, complete, and current. The way data files are organized significantly affects the consistency of information. For example, if redundant data is spread across multiple files, it can be difficult to preserve data integrity since we would have to make sure that updates are applied across multiple data files. However, if data is organized in a relational database and redundancy is eliminated through data normalization, data integrity is preserved and data consistency problems will be avoided. A poorly designed database can often ultimately be accompanied by a lack of data integrity.

9. Describe the limitations of file-processing systems. How do database systems make it possible to overcome these limitations?
   Answer. File processing systems store data in separate files and use custom programming code to read, write, and manipulate records in these files. These systems lack data integrity since physical separation of data can create inconsistencies in the form of redundancy (i.e., duplication of data) and outdated or incomplete data. The dependence on custom programming code and a programmer's documentation of the code creates potential problems if changes in business rules require updates and redesign. Databases solve this problem by keeping related data integrated and as a separate platform providing independence of data from the program that consumes the data. Programs can request and retrieve data through data views that are controlled by the database since a well-designed database can expose only necessary information to the client program preventing unauthorized data access. Existence of databases creates the possibility for the three-schema architecture (ANSI/SPARC) because users are now insulated from the physical location of the data and the underlying hardware. This separation creates physical and logical data independence among users immune to the changes in underlying data structures and hardware providers.

10. Using the Internet, trace the history of ANSI and ISO and their relevance to the information systems discipline. Write a summary of your findings.
Answer. *www.ansi.org/about_ansi/introduction/history.aspx?menuid=1*
www.ansi.org/
Page 14 of the book Oracle9i: SQL with an Introduction to PL/SQL written by Lannes L. Morris-Murphy, Course Technology, 2003, identifies ANSI and ISO as industry-accepted committees that set industry standards for SQL. The discussion points out that the use of industry-established standards allows the user to transfer skills among various relational database management systems and enables various applications to communicate with different databases without major redevelopment efforts.

11. Describe the structure of the ANSI/SPARC three-schema architecture. Compare this structure with that of the two-schema architecture inherent in a file-processing system.
Answer. The ANSI/SPARC three-schema architecture consists of three perspectives of metadata in a database: the external schema, the conceptual schema, and the internal schema. The external schema consists of a number of user views or subschemas – each describing portions of the database of interest to a particular user or group of users. The conceptual schema is located between the external schema and the internal schema and represents the global conceptual view of the structure of the entire database for the community of users. In other words, the conceptual schema is the consolidation of user views. The internal schema describes the physical structure of the stored data and the mechanism used to access the data. On the other hand, a two-schema architecture consists of just the internal schema and the external schema. In the absence of a conceptual schema, the internal schema structures must be mapped directly to the external schema. As a result, changes in the internal schema requiring appropriate changes in the external schema result in the loss of data independence.

12. Explain why a file-processing system may be referred to as belonging to a two-schema architecture.
Answer. A file-processing system can be referred to as a two-schema architecture because any modification to the storage structure or access strategy in the internal schema necessitates changes to the application programs in the external schema and the subsequent recompilation and testing of these programs.

13. Define data independence.
Answer. Data independence occurs as a result of the conceptual schema insulating user views (i.e., the external schema) from changes in the physical storage structure of the data in the internal schema. This is also known as physical data independence.

14. What is the difference between logical and physical data independence? Why is the distinction between the two important?
Answer. Logical data independence occurs when user views are immune to logical design changes in the conceptual schema.

15. What is the difference between a database and a database management system?
    Answer. A database is a self-describing collection of integrated files consisting of users' data, metadata, and overhead data. A database management system product, on the other hand, is a collection of general-purpose software that facilitates the processes of defining, constructing, and manipulating a database.

16. Since ANSI and ISO have adopted SQL as the standard language for database access, explore via the Internet the history and features of SQL and its appropriateness for database access. Write a summary of your findings.
    Answer. Two examples of Internet sources are:
    www.vbip.com/books/1861001800/chapter_1800_02.asp
    http://en.wikipedia.org/wiki/SQL

17. Write a short essay (one or two pages) about distributed databases using information available from Internet sources.
    Answer. A possible Internet source is: http://en.wikipedia.org/wiki/Distributed_database

18. Write a short essay (one or two pages) about data warehousing using information available from Internet sources.
    Answer. A possible Internet source is: http://en.wikipedia.org/wiki/Data_warehouse

19. Oil companies have functional databases, and the consumer-product industry tends to have product databases. How do financial institutions and the airline industry classify their enterprise database systems? Use Internet sources to find the answer, and record your findings.
    Answer. Rather than using Internet sources, responding to this question may require the student to contact an airline and a financial institution to learn how each organizes its database. They will likely learn that airlines tend to have functional databases while financial institutions are more oriented around product databases.

20. Find out and describe briefly what a CASE tool is, using Internet sources.
    Answer. A possible Internet source is: http://en.wikipedia.org/wiki/CASE_tool

21. Distinguish between a model and a data model.
    Answer. Some students may indicate that a data model is restricted to being a conceptual expression of an intangible real world object. This is just half true. Please ask them to read the last sentence of the second paragraph on page 14 in Section 1.6. It indicates that objects, ***tangible as well as intangible****, can be modeled using descriptors.*

22. What is the role of data models in database design?
    Answer. Data models serve as the blueprints for designing databases. Data modeling is a crucial part of the database design process since an accurate and comprehensive data model provides a solid basis for the future development of a flexible, scalable, and maintainable database.

23. Write a short essay (one or two pages) summarizing the content of the Harvard Business Review article on databases citied in Figure 1.4.
    Answer. Figure 1.4 no longer contains a reference to the article from which it has been adapted. Its source is the classic article referenced on page 21 as Nolan, R. L. (1973) "Computer Data Bases: The Future is Now," Harvard Business Review September-October, Vol. 51 No. 4, pp. 98-114.