

Database Systems: Design, Implementation, and Management

12TH Edition

Chapter 3

The Relational Database Model



Department of
Software Engineering
BAHRIA UNIVERSITY
Discovering Knowledge

Data Modeling and Data Models

- Data models
 - Relatively simple representations of complex real-world data structures
 - Often graphical
- Model: an abstraction of a real-world object or event
 - Useful in understanding complexities of the real-world environment
- Data modeling is iterative and progressive

Data Modeling and Data Models

- **Data modeling:** It is a process of creating a data model for the data to be stored in a Database. Whereas data model is a conceptual representation of
 - Data objects
 - The associations between different data objects
 - The rules.
- Data modeling helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data. Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data.

Data Modeling and Data Models

Data model emphasizes on what data is needed and how it should be organized instead of what operations need to be performed on the data.

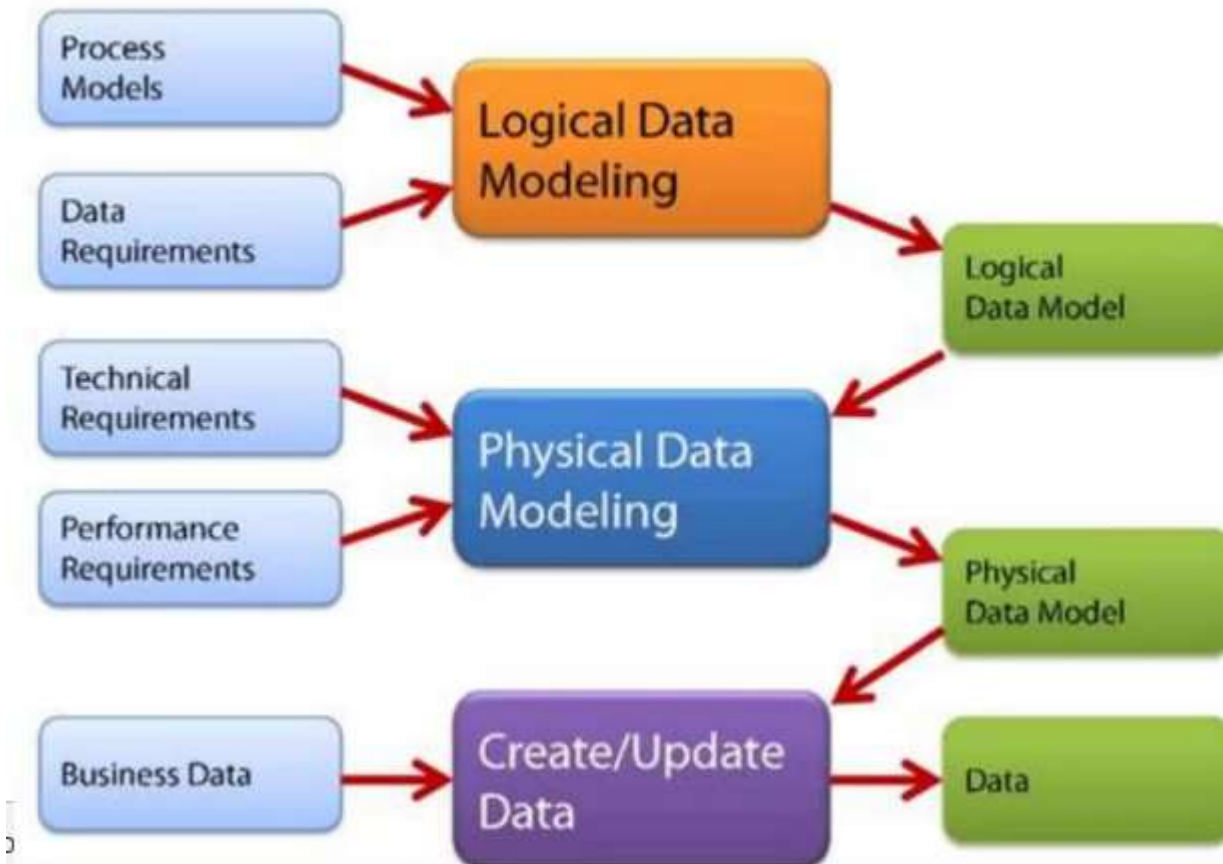
- **Data models:** Simple representations of complex real-world data structures
 - Useful for supporting a specific problem domain
- **Model** - Abstraction of a real-world object or event

- The primary goal of using data model are:
 - Ensures that all data objects required by the database are accurately represented. Omission of data will lead to creation of faulty reports and produce incorrect results.
 - A data model helps design the database at the conceptual, physical and logical levels.
 - Data Model structure helps to define the relational tables, primary and foreign keys and stored procedures.
 - It provides a clear picture of the base data and can be used by database developers to create a physical database.
 - It is also helpful to identify missing and redundant data.
 - Though the initial creation of data model is labor and time consuming, in the long run, it makes your IT infrastructure upgrade and maintenance cheaper and faster.

Types of Data Models [1/2]

- **Conceptual:** This Data Model defines **WHAT** the system contains. This model is typically created by Business stakeholders and Data Architects. The purpose is to organize, scope and define business concepts and rules.
- **Logical:** Defines **HOW** the system should be implemented regardless of the DBMS. This model is typically created by Data Architects and Business Analysts. The purpose is to developed technical map of rules and data structures.
- **Physical:** This Data Model describes **HOW** the system will be implemented using a specific DBMS system. This model is typically created by DBA and developers. The purpose is actual implementation of the database.

Types of Data Models [2/2]



Advantages of Data model

- The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately.
- The data model should be detailed enough to be used for building the physical database.
- The information in the data model can be used for defining the relationship between tables, primary and foreign keys, and stored procedures.
- Data Model helps business to communicate the within and across organizations.
- Data model helps to documents data mappings in ETL process
- Help to recognize correct sources of data to populate the model

The Importance of Data Models

- Facilitate interaction among the designer, the applications programmer, and the end user
- End users have different views and needs for data
- Data model organizes data for various users
- Data model is an abstraction
 - Cannot draw required data out of the data model

Data Model Basic Building Blocks

- Entity: anything about which data are to be collected and stored
- Attribute: a characteristic of an entity
- Relationship: describes an association among entities
 - One-to-many (1:M) relationship
 - Many-to-many (M:N or M:M) relationship
 - One-to-one (1:1) relationship
- Constraint: a restriction placed on the data

Business Rules

- Descriptions of policies, procedures, or principles within a specific organization
 - Apply to any organization that stores and uses data to generate information
- Description of operations to create/enforce actions within an organization's environment
 - Must be in writing and kept up to date
 - Must be easy to understand and widely disseminated
- Describe characteristics of data as viewed by the company

Discovering Business Rules (cont'd.)

- Standardize company's view of data
- Communications tool between users and designers
- Allow designer to understand the nature, role, and scope of data
- Allow designer to understand business processes
- Allow designer to develop appropriate relationship participation rules and constraints

Translating Business Rules into Data Model Components

- Nouns translate into entities
- Verbs translate into relationships among entities
- Relationships are bidirectional
- Two questions to identify the relationship type:
 - How many instances of B are related to one instance of A?
 - How many instances of A are related to one instance of B?

The Evolution of Data Models

**TABLE
2.1**

Evolution of Major Data Models

GENERATION	TIME	DATA MODEL	EXAMPLES	COMMENTS
First	1960s–1970s	File system	VMS/VSAM	Used mainly on IBM mainframe systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems Navigational access
Third	Mid-1970s	Relational	DB2 Oracle MS SQL Server MySQL	Conceptual simplicity Entity relationship (ER) modeling and support for relational data modeling
Fourth	Mid-1980s	Object-oriented Object/ relational (O/R)	Versant Objectivity/DB DB2 UDB Oracle 11g	Object/relational supports object data types Star Schema support for data warehousing Web databases become common
Fifth	Mid-1990s	XML Hybrid DBMS	dbXML Tamino DB2 UDB Oracle 11g MS SQL Server	Unstructured data support O/R model supports XML documents Hybrid DBMS adds object front end to relational databases Support large databases (terabyte size)
Emerging Models: NoSQL	Late 2000s to present	Key-value store Column store	SimpleDB (Amazon) BigTable (Google) Cassandra (Apache)	Distributed, highly scalable High performance, fault tolerant Very large storage (petabytes) Suited for sparse data Proprietary API