

Data Model

Hui Chen ^a

^aCUNY Brooklyn College, Brooklyn, NY, USA

February 1, 2022

Outline

- 1 Data Models
- 2 Data Modeling
- 3 Entity-Relationship Modeling

Outline

1 Data Models

2 Data Modeling

3 Entity-Relationship Modeling

Data Modeling and Data Models

- ▶ Model: simple representation of complex real-world objects and events
- ▶ Data Model: simple/mathematical representation of complex real-world data structures, useful for supporting a specific problem domain
- ▶ Data modeling: the process of creating a specific data model for a determined problem domain



Data Model Components

- ▶ Mathematical representation of the structure of data
 - ▶ Entity: person, place, thing, or event about which data will be collected and stored
 - ▶ Attribute: characteristic of an entity
 - ▶ Relationship: association among entities
 - ▶ Multiplicity: One-to-many (1:M OR 1..*), Many-to-many (M:N or *..*), and One-to-one (1:1 OR 1..1)
- ▶ Operations: a limited set of operations that can be performed on the data
- ▶ Constraint: restriction placed on data, such as, what the data can be
 - ▶ Ensures data integrity

Types of Data Models

Time	Data Model	Examples DBMS	Comments
1960s, 1970s	File System	VMS/SAM	Used mainly on IBM mainframe systems; managed records, not relationships
1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems; navigational access
Mid-1970s	Relational	DB2, Oracle MS SQL Server, MySQL	Conceptual simplicity; Entity Relationship (ER) modeling and support for relational data modeling
Mid-1980s	Object-oriented; Object/Relational(O/R)	Versant, Objectivity/DB, DB2 UDB, Oracle 12c	Object/relational supports object data types; Star Schema support for data warehousing; Web databases become common
Mid-1990s	XML Hybrid DBMS	dbXML, Tamino, DB2 UDB, Oracle 12c, 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)
Early 2000s to present	Key-value store; Column store	SimpleDB (Amazon), BigTable (Google), Cassandra (Apache), MongoDB, Riak	Distributed, highly scalable; High performance, fault tolerant; Very large storage (petabytes); Suited for sparse data; Proprietary application programming interface (API)

Outline

- 1 Data Models
- 2 Data Modeling**
- 3 Entity-Relationship Modeling

Data Modeling

- ▶ Discovering *business rules*
- ▶ Translating business rules into data model components



Discover Business Rules

Identify and document business rules to create an accurate data model

- ▶ Understand and standardize an organization's view of data
 - ▶ Understand the nature, role, scope of data, and business processes
 - ▶ Develop appropriate relationship participation rules and constraints

Translating Business Rules into Data Model Components

Based on business rules, identify entities, attributes, relationships, and constraints

- ▶ Nouns translate into entities
- ▶ Verbs translate into relationships among entities
 - ▶ Relationships are bidirectional
 - ▶ 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?

Outline

- 1 Data Models
- 2 Data Modeling
- 3 Entity-Relationship Modeling**

Entity-Relationship Model Building Blocks

- ▶ Entity sets
- ▶ Attributes, and
- ▶ Relationships

Entity Sets

An entity is an abstract object of some sort, and a collection of similar entities forms an *entity set*.

- ▶ Example: let's design a movie database (from whom?)
 - ▶ What are the nouns?
 - ▶ Each movie is an entity, and the set of all movies constitutes an entity set
 - ▶ Call the entity set *Movies*
 - ▶ Each movie star (actor or actress) is an entity, the set of all movie stars is an entity set
 - ▶ Call the entity set *Stars*
- ▶ Question: let's consider a database for an educational institution
 - ▶ What are examples of entity and entity set?

Attributes

Entity sets have associated attributes, which are properties of the entities in that set.

- ▶ Example: let's continue to design the movie database
- ▶ What attributes can entity set *Movies* and *Stars* have?
- ▶ Discussion: Are the attributes of primitive data types?

Relationships

Relationships are connections among two or more entity sets.

- ▶ Example: let's continue to design the movie database
- ▶ What are the verbs?
- ▶ Entity sets `Movies` and `Stars` can have a relationship called `Stars-in`
- ▶ Entity sets `Studios` and `Movies` can have a relationship called `Owns`

Entity-Relationship Diagrams

An Entity-Relationship Diagram (E-R Diagram) is a graph representing entity sets, attributes, and relationships. There are three popular notations to represent the graph.

- ▶ Chen notation (by Peter Chen)
- ▶ Crow's Foot notation
- ▶ UML notation

In this class, we shall follow Chen notation.

To be continued . . .