



LECTURE 6

Chapter 2

Data models

Business rules

- From a database point of view, the collection of data becomes meaningful only when it reflects properly defined 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
- written business rules are used to define entities, attributes, relationships, and constraints.

Business rules

- Business rules describe, in simple language, the main and distinguishing characteristics of the data as viewed by the company. Examples of business rules are as follows:
 1. A customer may generate many invoices.
 2. An invoice is generated by only one customer.
 3. A training session cannot be scheduled for fewer than 10 employees or for more than 30 employees.

Discovering business rules

- Sources of business rules:
 - Company managers Policy makers
 - Written documentation Department managers
 - Procedures Standards Operations manuals
 - Direct interviews with end users
- The process of identifying and documenting business rules is essential to database design for several reasons:
- It helps to standardize the company's view of data.
- It can be a communication tool between users and designers.
- It allows the designer to understand the nature, role, and scope of the data.
- It allows the designer to understand business processes.
- It allows the designer to develop appropriate relationship participation rules and constraints and to create an accurate data model.

Translating business rules into data model components

- Nouns translate into entities
- Verbs translate into relationships among entities
- For example, the business rule “a customer may generate many invoices” contains two nouns (customer and invoices) and a verb (generate) that associates the nouns.
- 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?
- For example
 - In how many classes can one student enroll? **Answer: many classes.**
 - How many students can enroll in one class? **Answer: many students.**

Naming conventions

- Naming occurs during translation of business rules to data model components
- Names should make the object unique and distinguishable from other objects
- Names should also be descriptive of objects in the environment and be familiar to users
- Proper naming:
 - Facilitates communication between parties
 - Promotes self-documentation

Standard database concepts

- **Schema** is the conceptual organization of the entire database as viewed by the Database administrator.
- **subschema** defines the portion of the database “seen” by the application programs That produce the desired information from the data within the Database.
- **data manipulation language (DML)** defines the environment in which data can Be managed and is used to work with the data in the database.
- A schema **data definition language (DDL)** enables the database administrator to Define the schema components.

Relational model

- Developed by E. F. Codd of IBM in 1970,
- The **relational model** is based on mathematical set theory and represents data as independent relations.
- Each **relation (table)** is conceptually represented as a two-dimensional structure of intersecting rows and columns.
- The relations are related to each other through the sharing of common entity characteristics (values in columns).
- **Table (relation)** a logical construct perceived to be a two-dimensional structure composed of intersecting rows (entities) and columns (attributes) that represents an entity set in the relational model.
- **Tuple** in the relational model, a table row.
- **Relational database management system (RDBMS)** a collection of programs that manages a relational database.
- The **RDBMS** software translates a user's logical requests (queries) into commands that physically locate and retrieve the requested data.

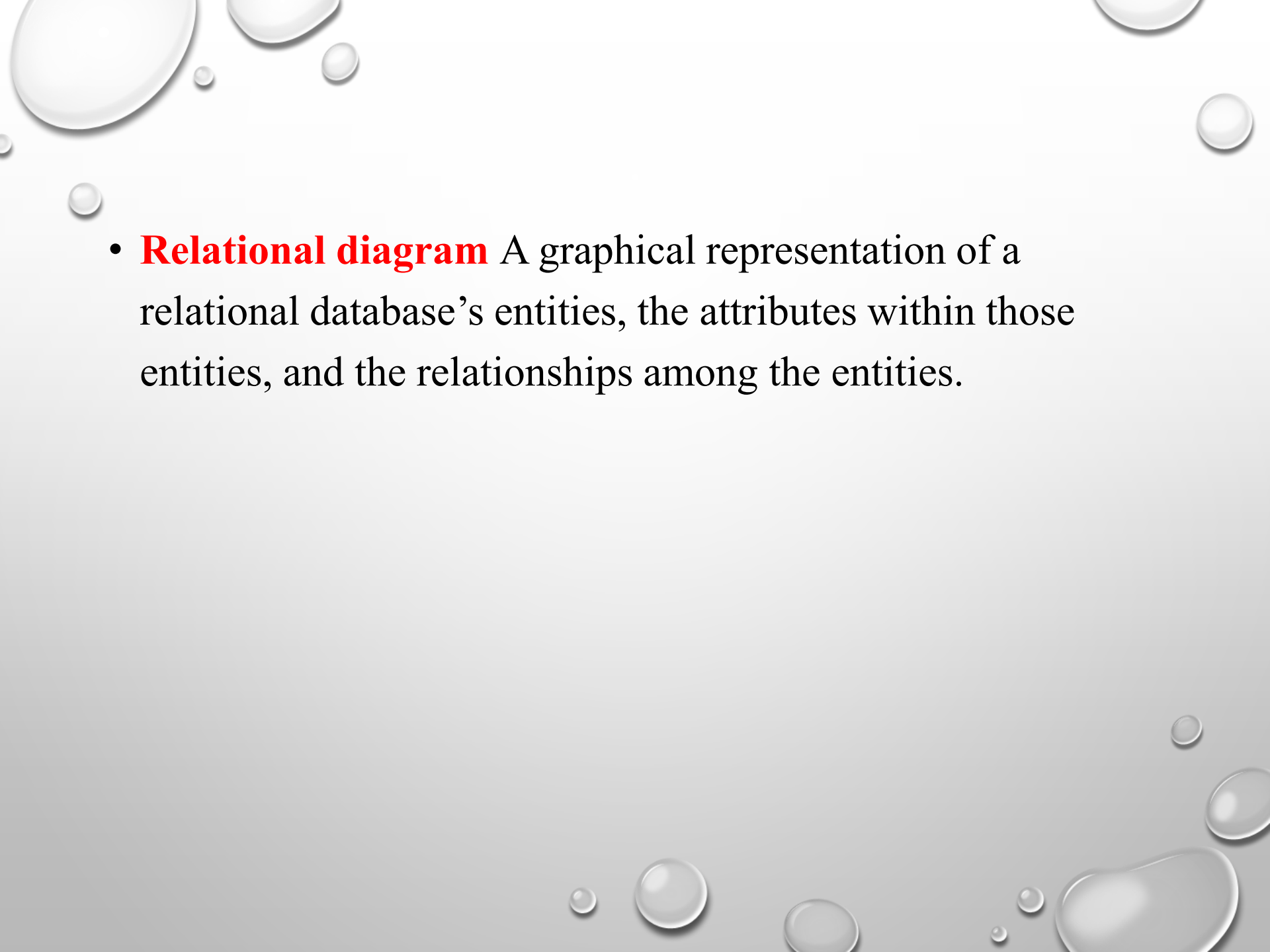
- 
- **Relational diagram** A graphical representation of a relational database's entities, the attributes within those entities, and the relationships among the entities.

FIGURE 2.1

Linking relational tables

Table name: AGENT (first six attributes)

Database name: Ch02_InsureCo

AGENT_CODE	AGENT_LNAME	AGENT_FNAME	AGENT_INITIAL	AGENT_AREACODE	AGENT_PHONE
501	Alby	Alex	B	713	228-1249
502	Hahn	Leah	F	615	882-1244
503	Okon	John	T	615	123-5589

Link through AGENT_CODE

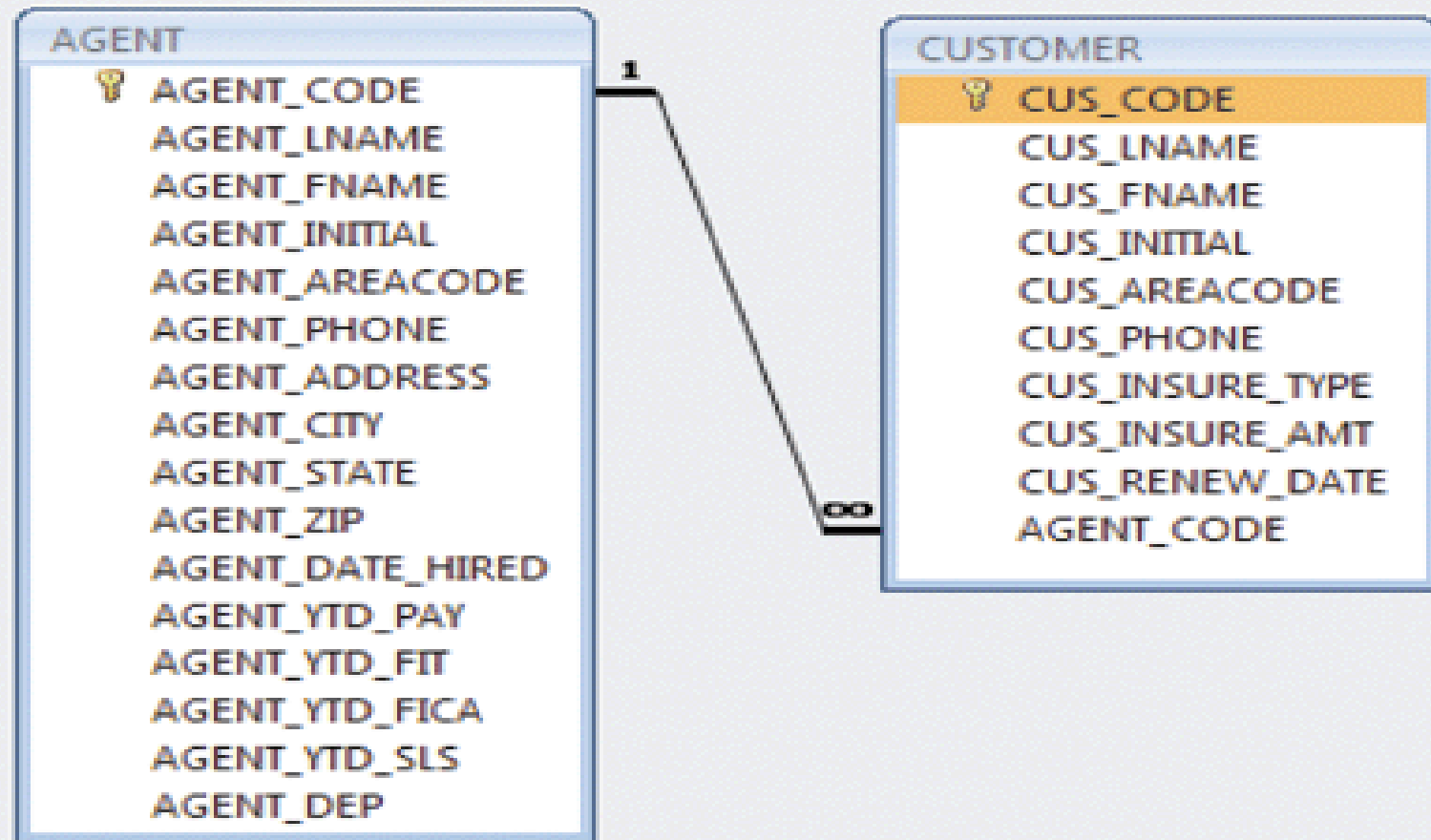
Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_INSURE_TYPE	CUS_INSURE_AMT	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	615	844-2573	T1	100.00	05-Apr-2012	502
10011	Dunne	Leona	K	713	894-1238	T1	250.00	16-Jun-2012	501
10012	Smith	Kathy	W	615	894-2285	S2	150.00	29-Jan-2013	502
10013	Olowski	Paul	F	615	894-2180	S1	300.00	14-Oct-2012	502
10014	Orlando	Myron		615	222-1672	T1	100.00	28-Dec-2013	501
10015	O'Brian	Amy	B	713	442-3381	T2	850.00	22-Sep-2012	503
10016	Brown	James	G	615	297-1228	S1	120.00	25-Mar-2013	502
10017	Williams	George		615	290-2556	S1	250.00	17-Jul-2012	503
10018	Farriss	Anne	G	713	382-7185	T2	100.00	03-Dec-2012	501
10019	Smith	Olette	K	615	297-3809	S2	500.00	14-Mar-2013	503

SOURCE: Course Technology/Cengage Learning

**FIGURE
2.2**

A relational diagram



SOURCE: Course Technology/Cengage Learning

Structured query language (SQL)

- Allows the user to specify what must be done without specifying how.
- The RDBMS uses SQL to translate user queries into instructions for retrieving the requested data.
- SQL makes it possible to retrieve data with far less effort than any other database or file environment.
- *The end-user interface*. Basically, the interface allows the end user to interact with the data (by automatically generating SQL code). Each interface is a product of the software vendor's idea of meaningful interaction with the data.
- *A collection of tables stored in the database*. In a relational database, all data is perceived to be stored in tables. The tables simply “present” the data to the end user in a way that is easy to understand. Each table is independent. Rows in different tables are related by common values in common attributes.
- *SQL engine*. Largely hidden from the end user, the SQL engine executes all queries, or data requests

- **Entity relationship (ER) model (ERM)** A data model that describes relationships (1:1, 1:M, and M:N) among entities at the conceptual level with the help of ER diagrams.
- **Entity relationship diagram (ERD)** a diagram that depicts an entity relationship model's entities, attributes, and relations.
- **Entity instance (entity occurrence)** a row in a relational table.
- **Entity set** a collection of like entities.
- **Connectivity** the type of relationship between entities.

FIGURE 2.3

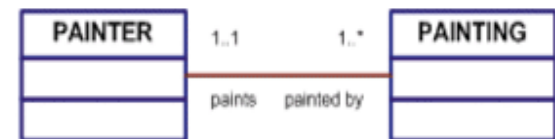
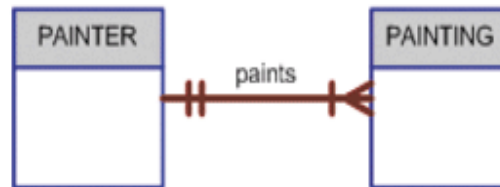
The ER model notations

Chen Notation

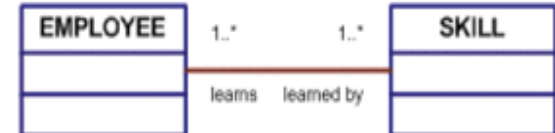
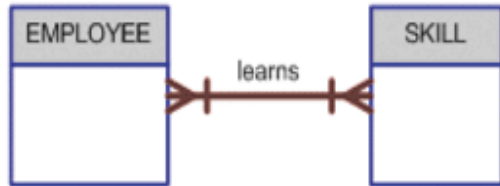
Crow's Foot Notation

UML Notation

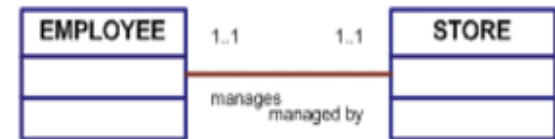
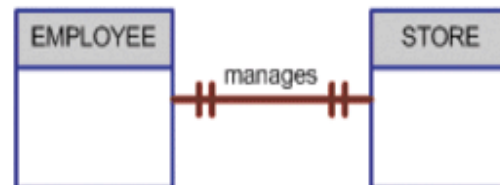
A One-to-Many (1:M) Relationship: a PAINTER can paint many PAINTINGs; each PAINTING is painted by one PAINTER.



A Many-to-Many (M:N) Relationship: an EMPLOYEE can learn many SKILLs; each SKILL can be learned by many EMPLOYEEs.



A One-to-One (1:1) Relationship: an EMPLOYEE manages one STORE; each STORE is managed by one EMPLOYEE.



Emerging data models: big data and NOSQL

- **Big data** refers to a movement to find new and better ways to manage large amounts of web- and sensor-generated data and derive business insight from it, while simultaneously providing high performance and scalability at a reasonable cost.
- The term *big data* has been used in many different frameworks, from law to statistics to economics to computing.

Basic characteristics of big data databases (4 Vs)

- *Volume* refers to the amounts of data being stored.
- *Velocity* refers not only to the speed with which data grows but also to the need to process this data quickly in order to generate information and insight.
- *Variety* refers to the fact that the data being collected comes in multiple different data formats. A great portion of these data comes in formats not suitable to be handled by the typical operational databases based on the relational model.
- *Variability* – This refers to the inconsistency which can be shown by the data at times, thus hampering the process of being able to handle and manage the data effectively.

New big data technologies.

- **Hadoop** is a java-based, open-source, high-speed, fault-tolerant distributed storage and computational framework. Hadoop uses low-cost hardware to create clusters of thousands of computer nodes to store and process data. Hadoop originated from Google's work on distributed file systems and parallel processing and is currently supported by the apache software foundation.⁵ Hadoop has several modules, but the two main components are Hadoop distributed file system (HDFS) and MapReduce.
- **Hadoop Distributed File System (HDFS)** is a highly distributed, fault-tolerant file storage system designed to manage large amounts of data at high speeds. In order to achieve high throughput, HDFS uses the write-once, read many model. This means that once the data is written, it cannot be modified. HDFS uses three types of nodes: a name node that stores all the metadata about the file system, a data node that stores fixed-size data blocks (that could be replicated to other data nodes), and a client node that acts as the interface between the user application and the HDFS.

- **MapReduce** is an open-source application programming interface (API) that provides fast data analytics services. MapReduce distributes the processing of the data among thousands of nodes in parallel. MapReduce works with structured and nonstructured data. The MapReduce framework provides two main functions: **Map and reduce**. In general terms, the map function takes a job and divides it into smaller units of work, and the reduce function collects all the output results generated from the nodes and integrates them into a single result set. Although MapReduce itself is viewed as fairly limited today, it defined the paradigm for how big data is processed.
- **NoSQL** is a large-scale distributed database system that stores structured and unstructured data in efficient ways.

- REPORT ON HADOOP