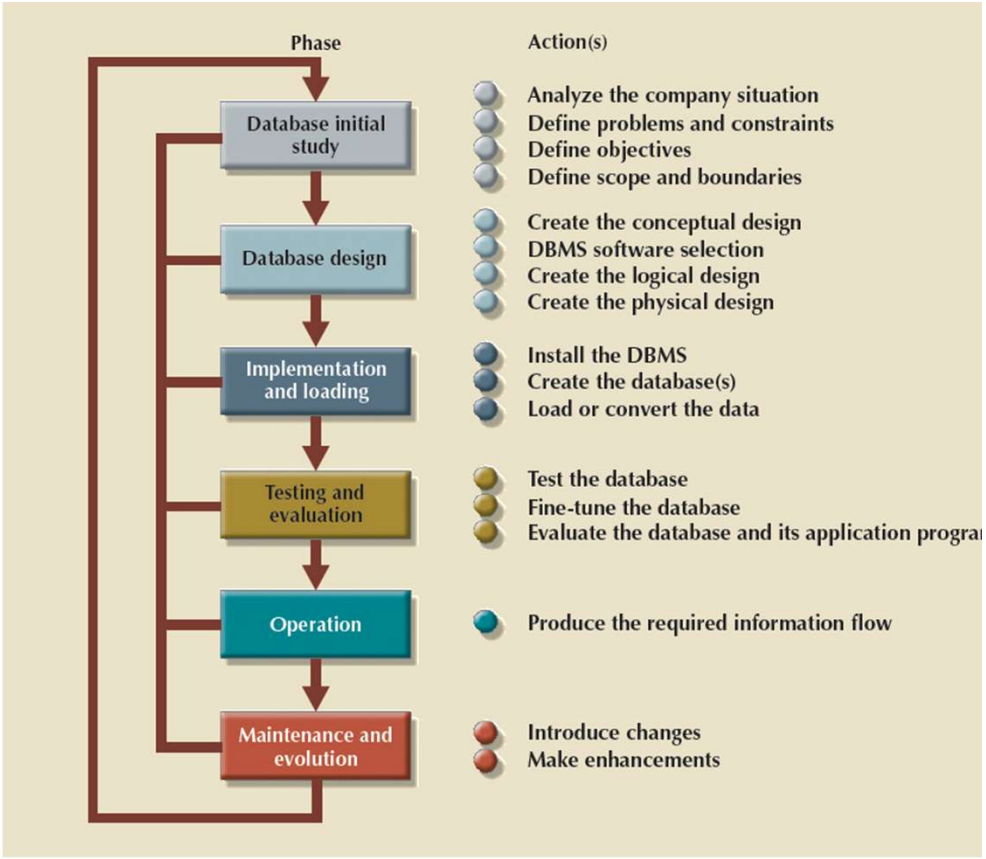
MSDS 420

Atef Bader, PhD

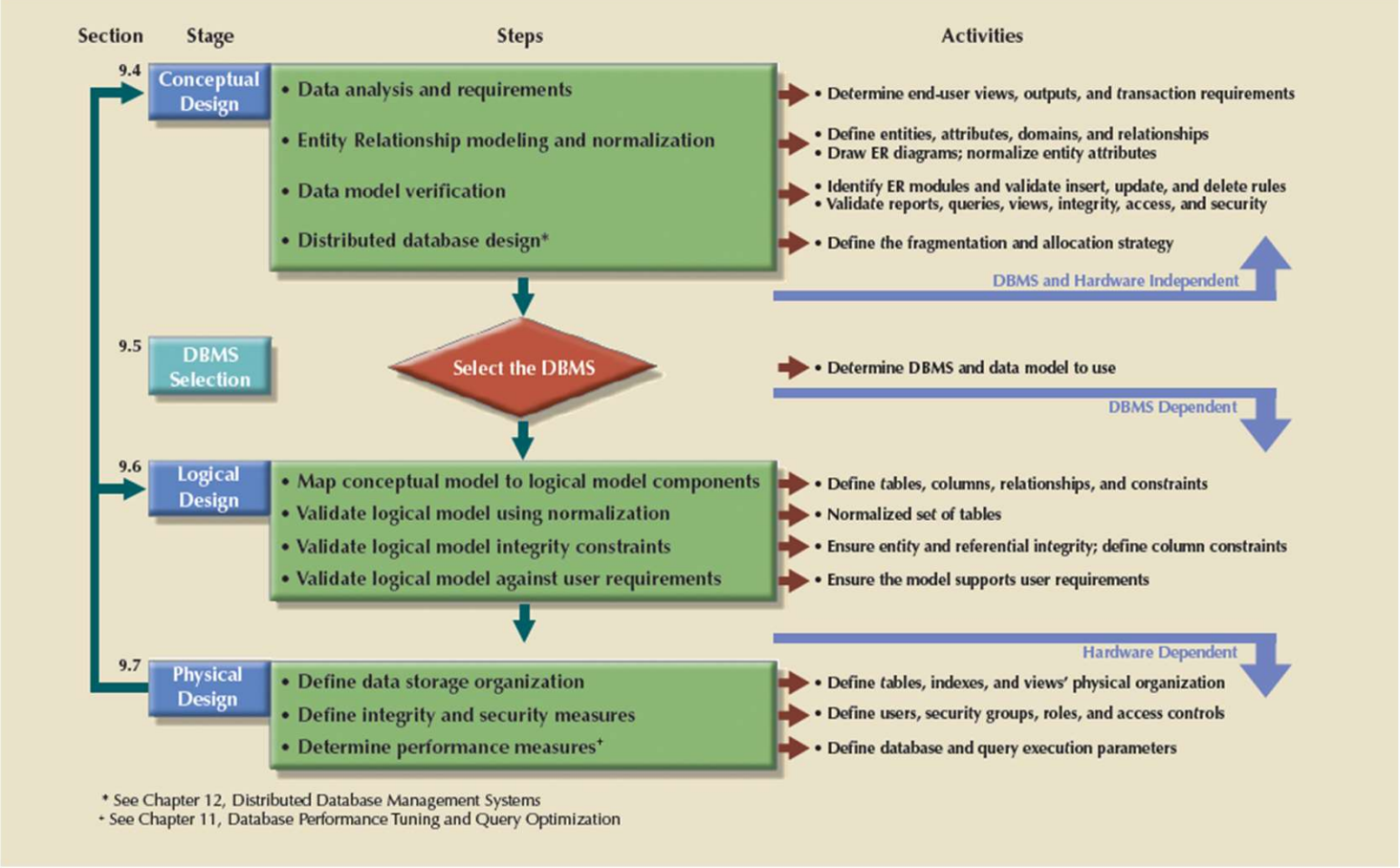
Agenda

- Database Life Cycle (DBLC)
- Database Design Process
- ERD (Entity Relationship Diagram)
- Exercise #2 Walkthrough and Deliverable

The Database Life Cycle (DBLC)



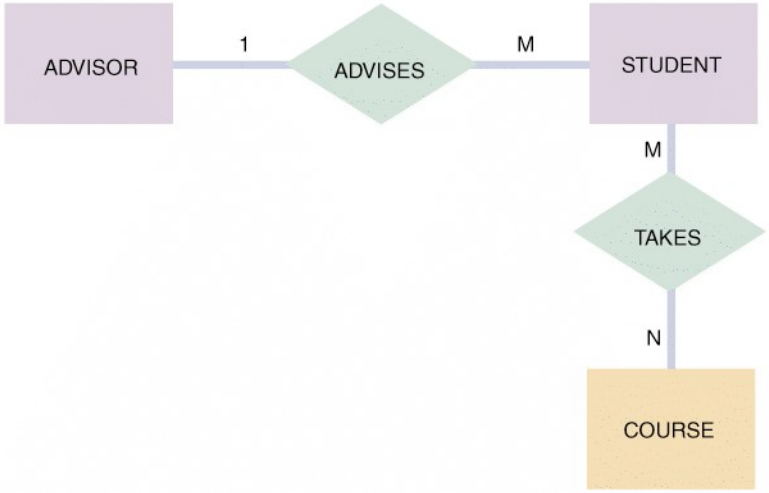
Database Design Process



Developing the Conceptual Model Using ER Diagrams

DEVELOPING THE CONCEPTUAL MODEL USING ER DIAGRAMS	
STEP	**ACTIVITY**
1	Identify, analyze, and refine the business rules.
2	Identify the main entities, using the results of Step 1.
3	Define the relationships among the entities, using the results of Steps 1 and 2.
4	Define the attributes, primary keys, and foreign keys for each of the entities.
5	Normalize the entities. (Remember that entities are implemented as tables in an RDBMS.)
6	Complete the initial ER diagram.
7	Validate the ER model against the end users' information and processing requirements.
8	Modify the ER model, using the results of Step 7.

Entity-Relationship Diagrams



An initial entity-relationship diagram for ADVISOR, STUDENT, and COURSE

Logical Design Steps

LOGICAL DESIGN STEPS	
STEP	**ACTIVITY**
1	Map the conceptual model to logical model components.
2	Validate the logical model using normalization.
3	Validate the logical model integrity constraints.
4	Validate the logical model against user requirements.

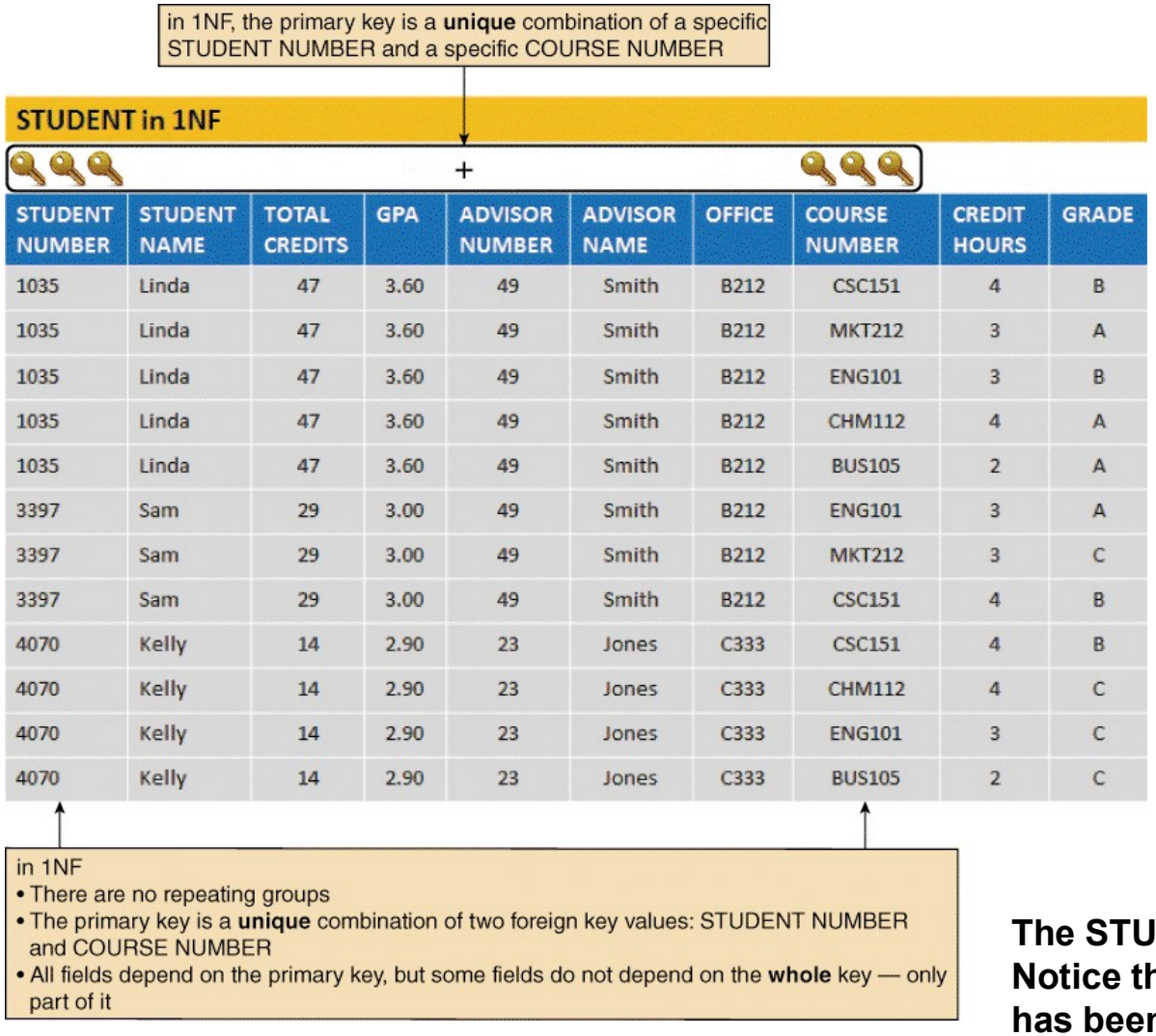
Data Normalization

STUDENT (Unnormalized)

STUDENT NUMBER	STUDENT NAME	TOTAL CREDITS	GPA	ADVISOR NUMBER	ADVISOR NAME	OFFICE	COURSE NUMBER	CREDIT HOURS	GRADE
1035	Linda	47	3.60	49	Smith	B212	CSC151	4	B
							MKT212	3	A
							ENG101	3	B
							CHM112	4	A
							BUS105	2	A
3397	Sam	29	3.00	49	Smith	B212	ENG101	3	A
							MKT212	3	C
							CSC151	4	B
4070	Kelly	14	2.90	23	Jones	C333	CSC151	4	B
							CHM112	4	C
							ENG101	3	C
							BUS105	2	C

The STUDENT table is unnormalized because it contains a repeating group that represents the courses each student has taken

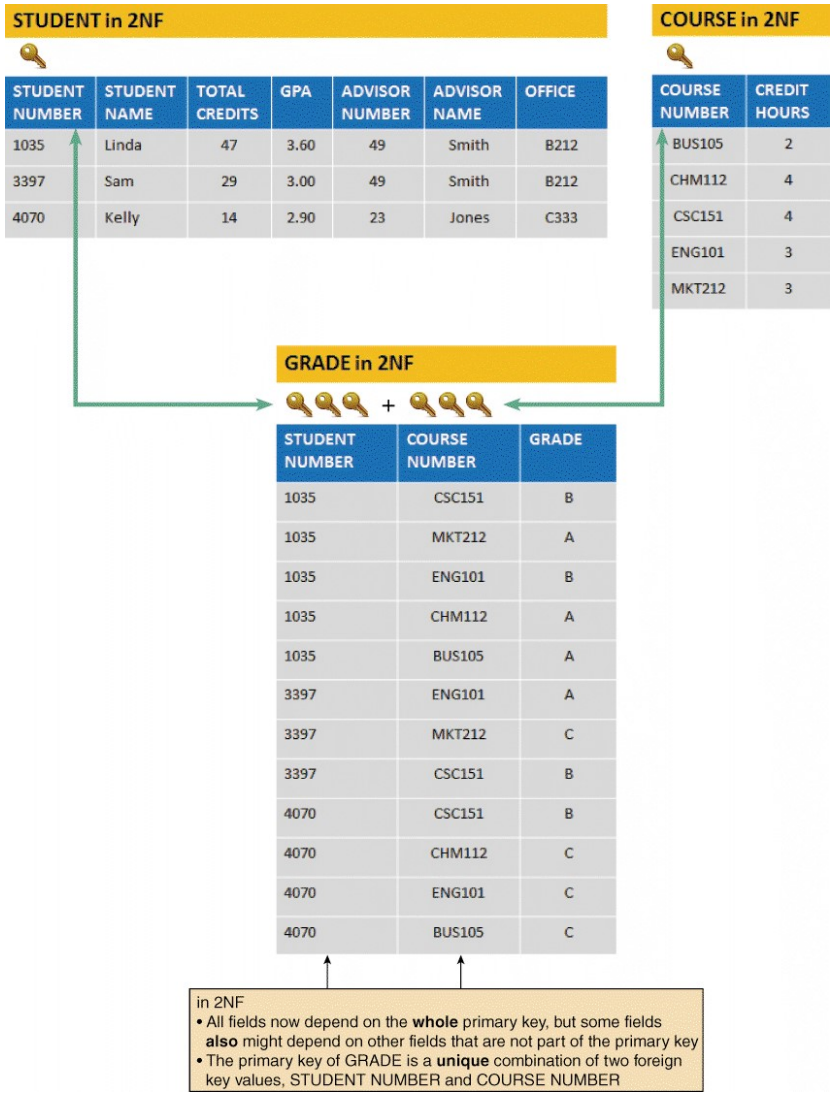
Data Normalization (Cont.)



STUDENT NUMBER	STUDENT NAME	TOTAL CREDITS	GPA	ADVISOR NUMBER	ADVISOR NAME	OFFICE	COURSE NUMBER	CREDIT HOURS	GRADE
1035	Linda	47	3.60	49	Smith	B212	CSC151	4	B
1035	Linda	47	3.60	49	Smith	B212	MKT212	3	A
1035	Linda	47	3.60	49	Smith	B212	ENG101	3	B
1035	Linda	47	3.60	49	Smith	B212	CHM112	4	A
1035	Linda	47	3.60	49	Smith	B212	BUS105	2	A
3397	Sam	29	3.00	49	Smith	B212	ENG101	3	A
3397	Sam	29	3.00	49	Smith	B212	MKT212	3	C
3397	Sam	29	3.00	49	Smith	B212	CSC151	4	B
4070	Kelly	14	2.90	23	Jones	C333	CSC151	4	B
4070	Kelly	14	2.90	23	Jones	C333	CHM112	4	C
4070	Kelly	14	2.90	23	Jones	C333	ENG101	3	C
4070	Kelly	14	2.90	23	Jones	C333	BUS105	2	C

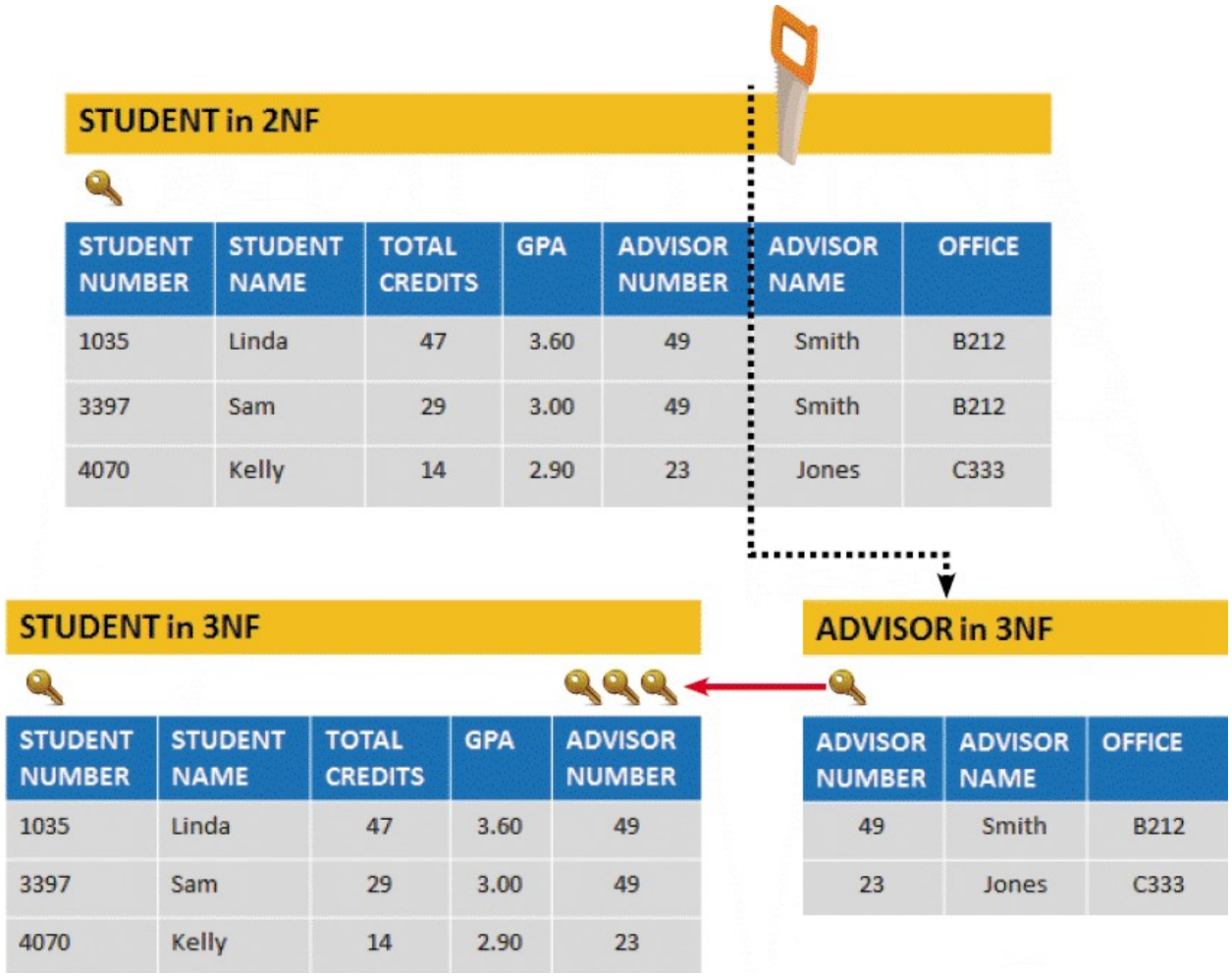
The STUDENT table in 1NF. Notice that the primary key has been expanded to include STUDENT NUMBER and COURSE NUMBER

Data Normalization (Cont.)



The STUDENT, COURSE, and GRADE tables in 2NF. Notice that all fields are functionally dependent on the entire primary key of their respective tables

Data Normalization (Cont.)



STUDENT, ADVISOR, COURSE, and GRADE tables in 3NF. When the STUDENT table is transformed from 2NF to 3NF, the result is two tables: STUDENT and ADVISOR

Physical Design Steps

PHYSICAL DESIGN STEPS	
STEP	**ACTIVITY**
1	Define data storage organization.
2	Define integrity and security measures.
3	Determine performance measurements.

Physical Design Steps

- Select the Database Engine for the implementation
  - Oracle, SQLite, PostgreSQL, MySQL, etc.

- Use SQL to
  - Create the Tables in the database Engine
  - Insert/Update/Delete Rows into the Tables
  - Answer different Queries to retrieve data and present information to the end users

Working with a Relational Database

- Suppose you work in the registration office, and you need answers for the following queries:
  - What are the courses taken by student number 3397?
  - Provide a list of students who have GPA less than 3.0
  - Who are the students advised by Jones?
  - Provide a list of students who have taken MKT212

Data Modeling and Data Models

- **Data modeling**: Iterative and progressive process of creating a specific data model for a determined problem domain

  - **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

Data Model Basic Building Blocks

- **Entity**: Unique and distinct object used to collect and store data
  - **Attribute**: Characteristic of an entity
- **Relationship**: Describes an association among entities
  - **One-to-many (1:M)**
  - **Many-to-many (M:N or M:M)**
  - **One-to-one (1:1)**
- **Constraint**: Set of rules to ensure data integrity

Business Rules

Brief, precise, and unambiguous description of a policy, procedure, or principle

Enable defining the basic building blocks

Describe main and distinguishing characteristics of the data

Sources of Business Rules

Company managers

Department managers

Written documentation

Direct interviews with end users

Reasons for Identifying and Documenting Business Rules

- Help standardize company's view of data
- Communications tool between users and designers
- Allow designer to:
  - Understand the nature, role, scope of data, and business processes
  - Develop appropriate relationship participation rules and constraints
  - Create an accurate data model

Translating Business Rules into Data Model Components

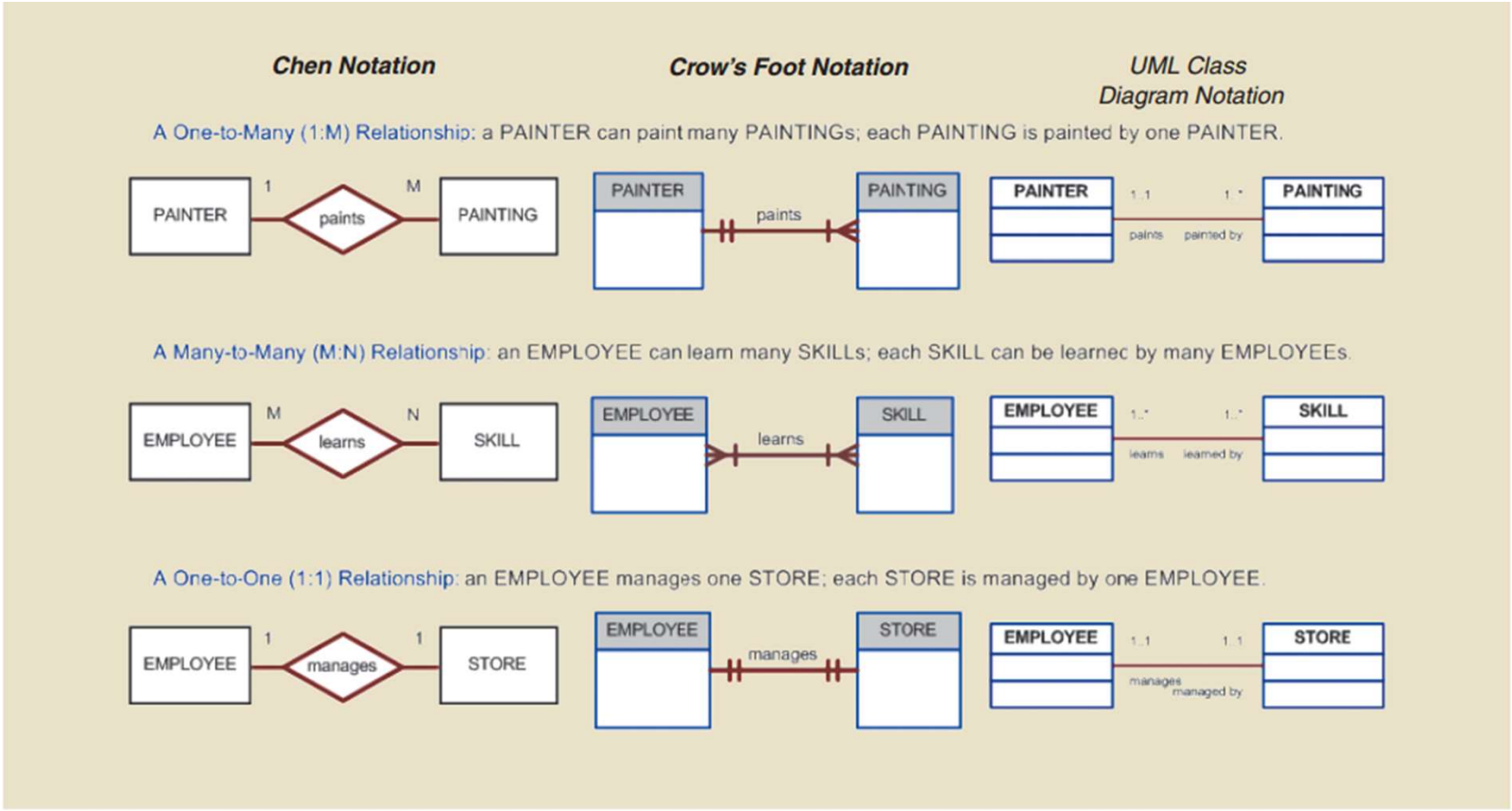
- Nouns translate into entities
- Verbs translate into relationships among entities
- Relationships are bidirectional
- Questions to identify the relationship type
  - How many instances of B are related to one instance of A?
  - How many instances of A are related to one instance of B?

The Entity Relationship Model

- Graphical representation of entities and their relationships in a database structure

- **Entity relationship diagram (ERD)**
  - Uses graphic representations to model database components
- **Entity instance or entity occurrence**
  - Rows in the relational table
- **Connectivity**: Term used to label the relationship types

The ER Model Notations



A Simple Key-Value Representation

Trucks-R-Us

Data stored using traditional relational model

DID	CERT1	CERT2	CERT3	DOB	LICTYPE
2732	80		95	1/24/1962	P
2946		92		4/11/1970	
3650	86			11/27/1963	R

Data stored using key-value model

DID	KEY	VALUE
2732	CERT1	80
2732	CERT3	95
2732	DOB	1/24/1962
2732	LICTYPE	P
2946	CERT2	92
2946	DOB	4/11/1970
3650	CERT1	86
3650	DOB	11/27/63
3650	LICTYPE	R

Driver 2732

- In the relational model:
  - Each row represents one entity instance.
  - Each column represents one attribute of the entity.
  - The values in a column are of the same data type.
- In the key-value model:
  - Each row represents one attribute/value of one entity instance.
  - The "key" column could represent any entity's attribute.
  - The values in the "value" column could be of any data type and therefore it is generally assigned a long string data type.

Relational Model

Advantages

- Structural independence is promoted using independent tables
- Tabular view improves conceptual simplicity
- Ad hoc query capability is based on SQL
- Isolates the end user from physical-level details
- Improves implementation and management simplicity

Disadvantages

- Requires substantial hardware and system software overhead
- Conceptual simplicity gives untrained people the tools to use a good system poorly
- May promote information problems

Entity Relationship Model

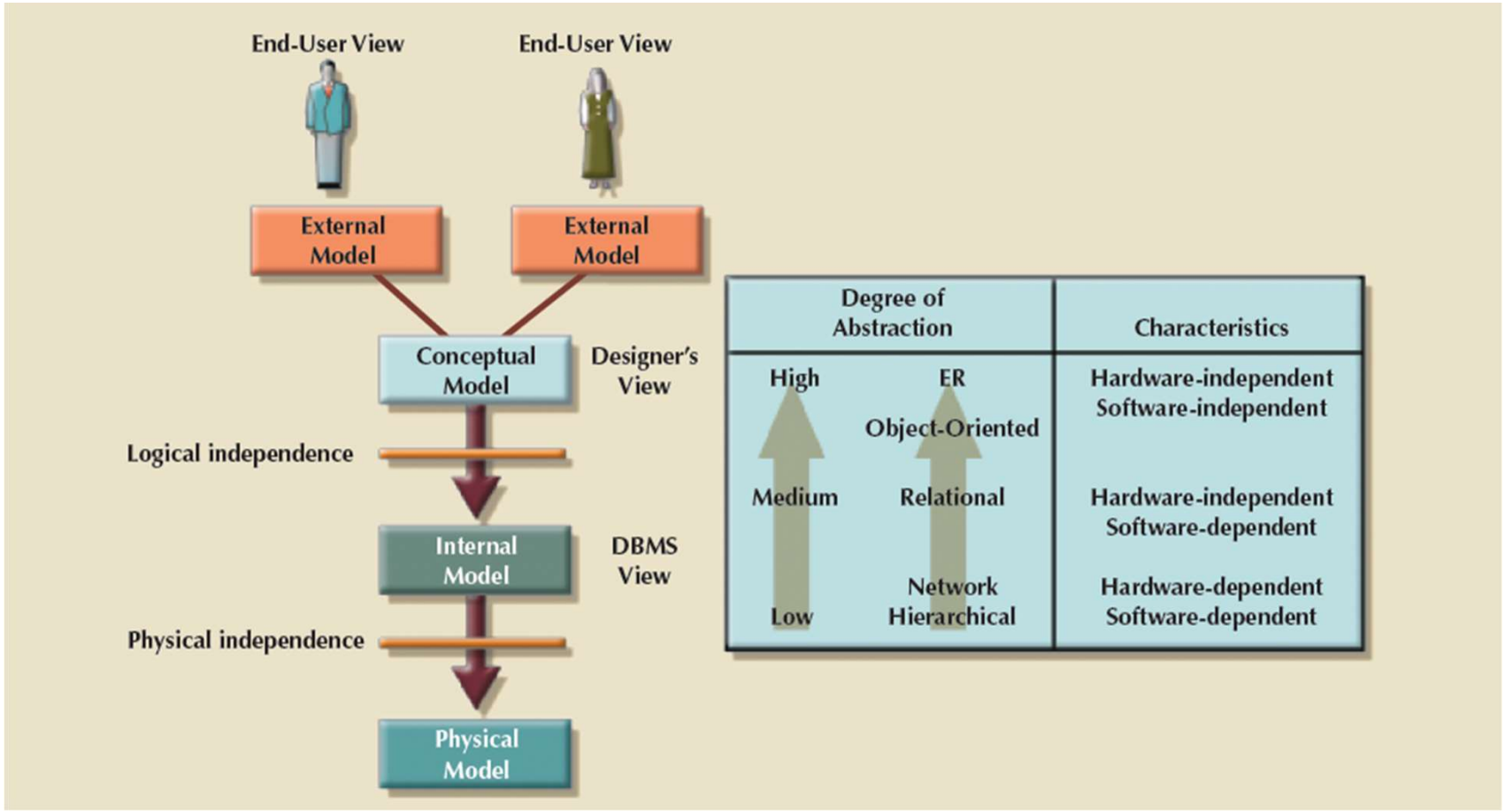
Advantages

- Visual modeling yields conceptual simplicity
- Visual representation makes it an effective communication tool
- Is integrated with the dominant relational model

Disadvantages

- Limited constraint representation
- Limited relationship representation
- No data manipulation language
- Loss of information content occurs when attributes are removed from entities to avoid crowded displays

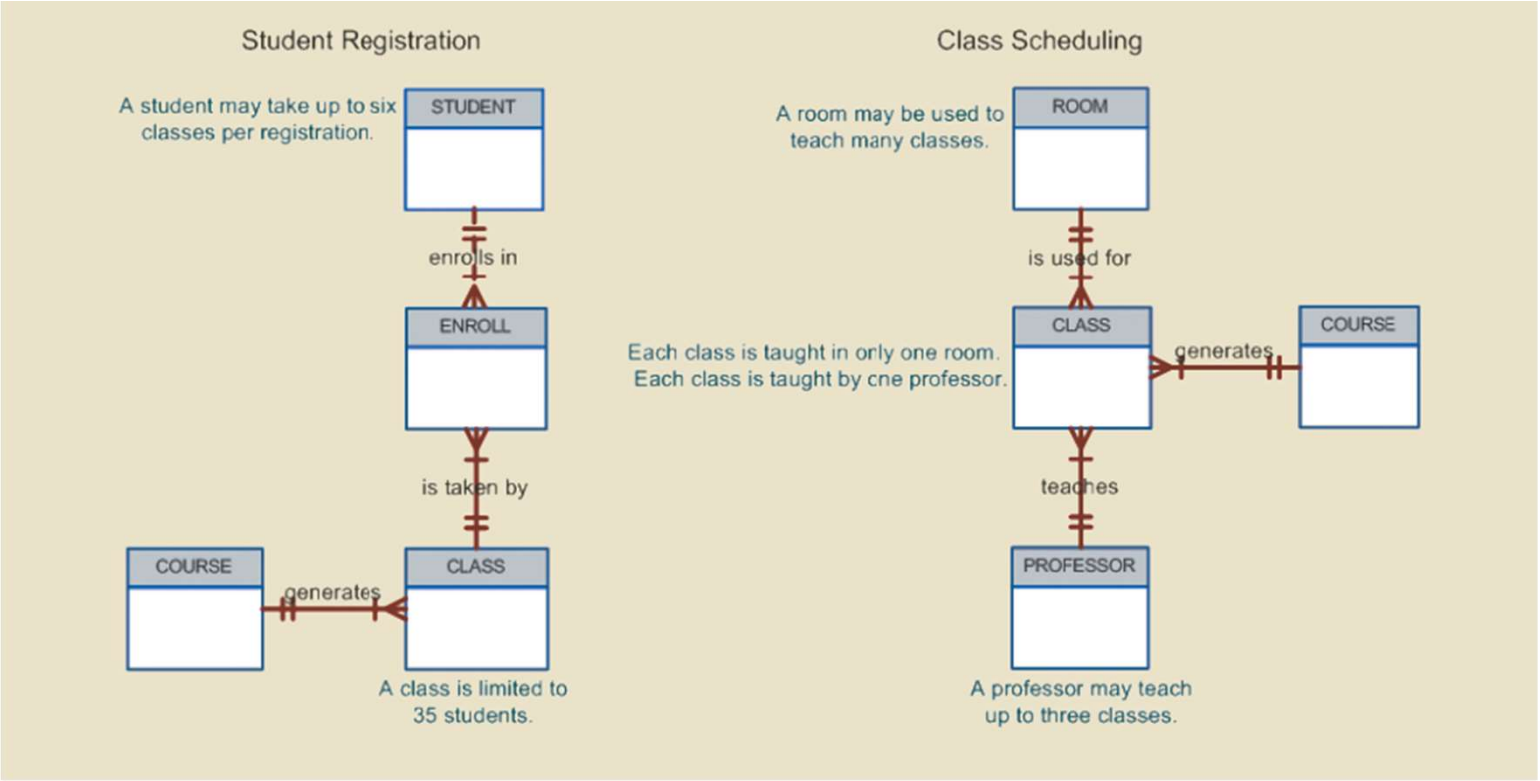
Data Abstraction Levels



The External Model

- End users' view of the data environment
- ER diagrams are used to represent the external views

External Models For Tiny College

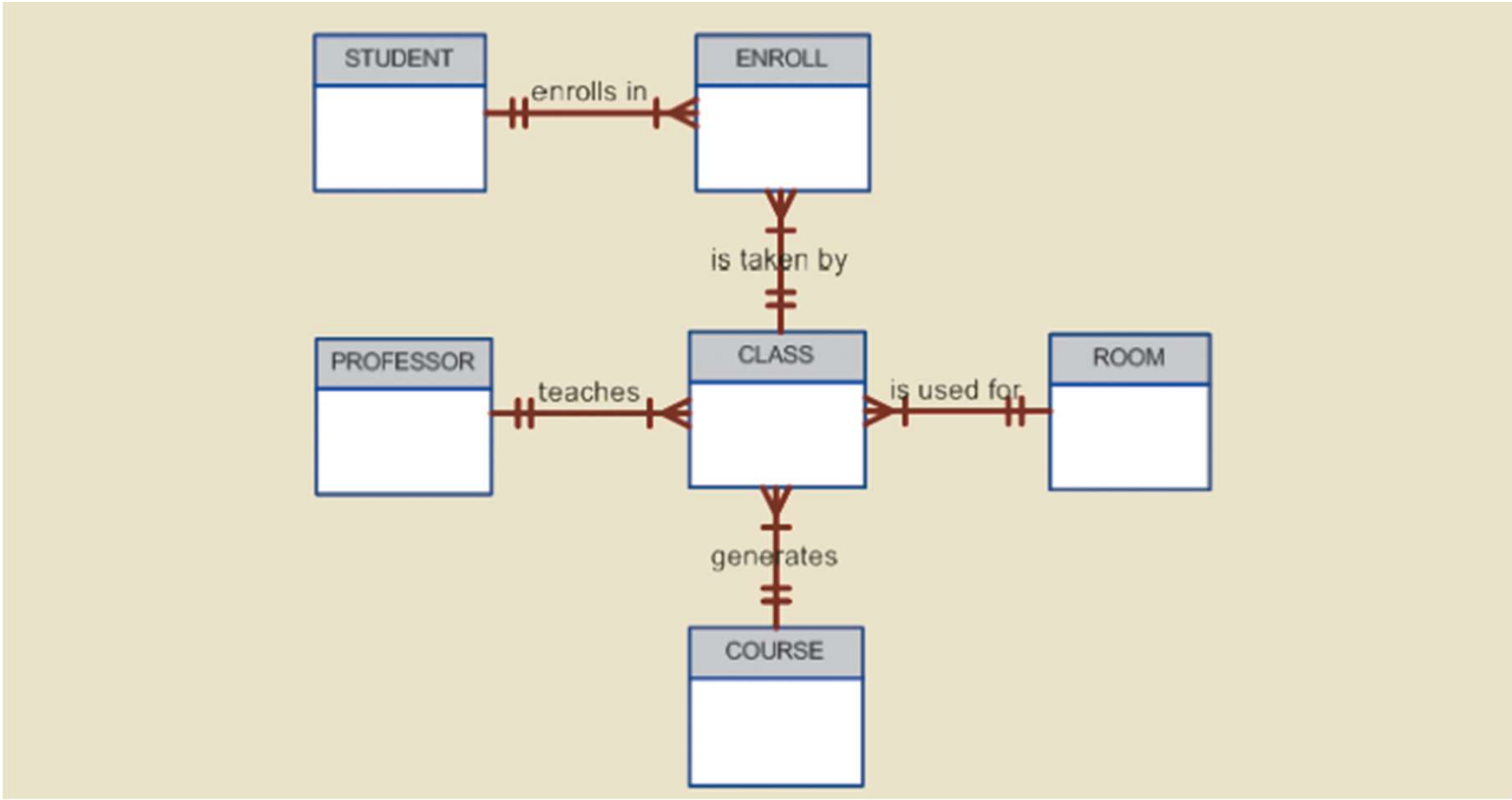


The Conceptual Model

- **Represents a global view of the entire database by the entire organization**

- **Conceptual schema**: Basis for the identification and high-level description of the main data objects
- Has a macro-level view of data environment
- Is software and hardware independent
- **Logical design**: Task of creating a conceptual data model

Conceptual Model For Tiny College

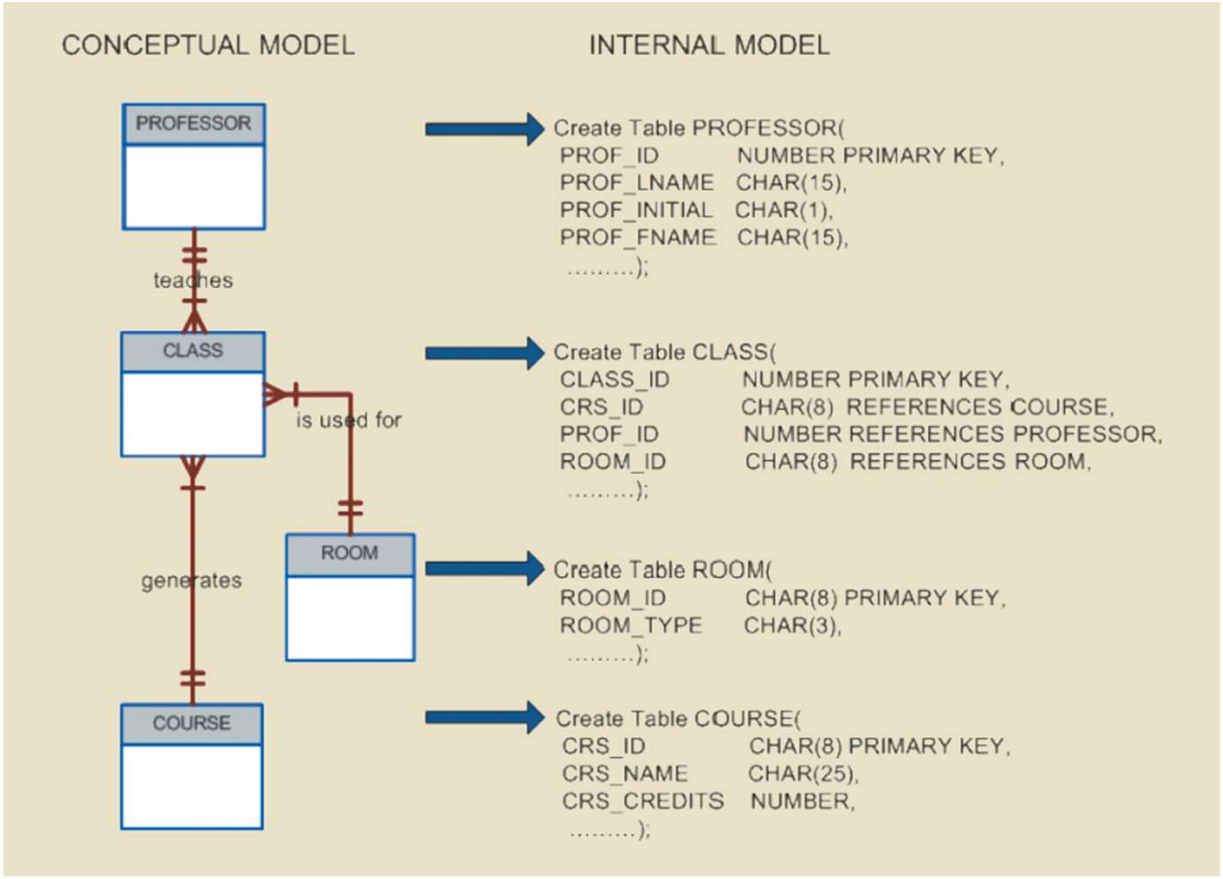


The Internal Model

- Representing database as seen by the DBMS mapping conceptual model to the DBMS

- **Internal schema**: Specific representation of an internal model
  - Uses the database constructs supported by the chosen database
- Is software dependent and hardware independent
- **Logical independence**: Changing internal model without affecting the conceptual model

Internal Model for Tiny College



The Physical Model

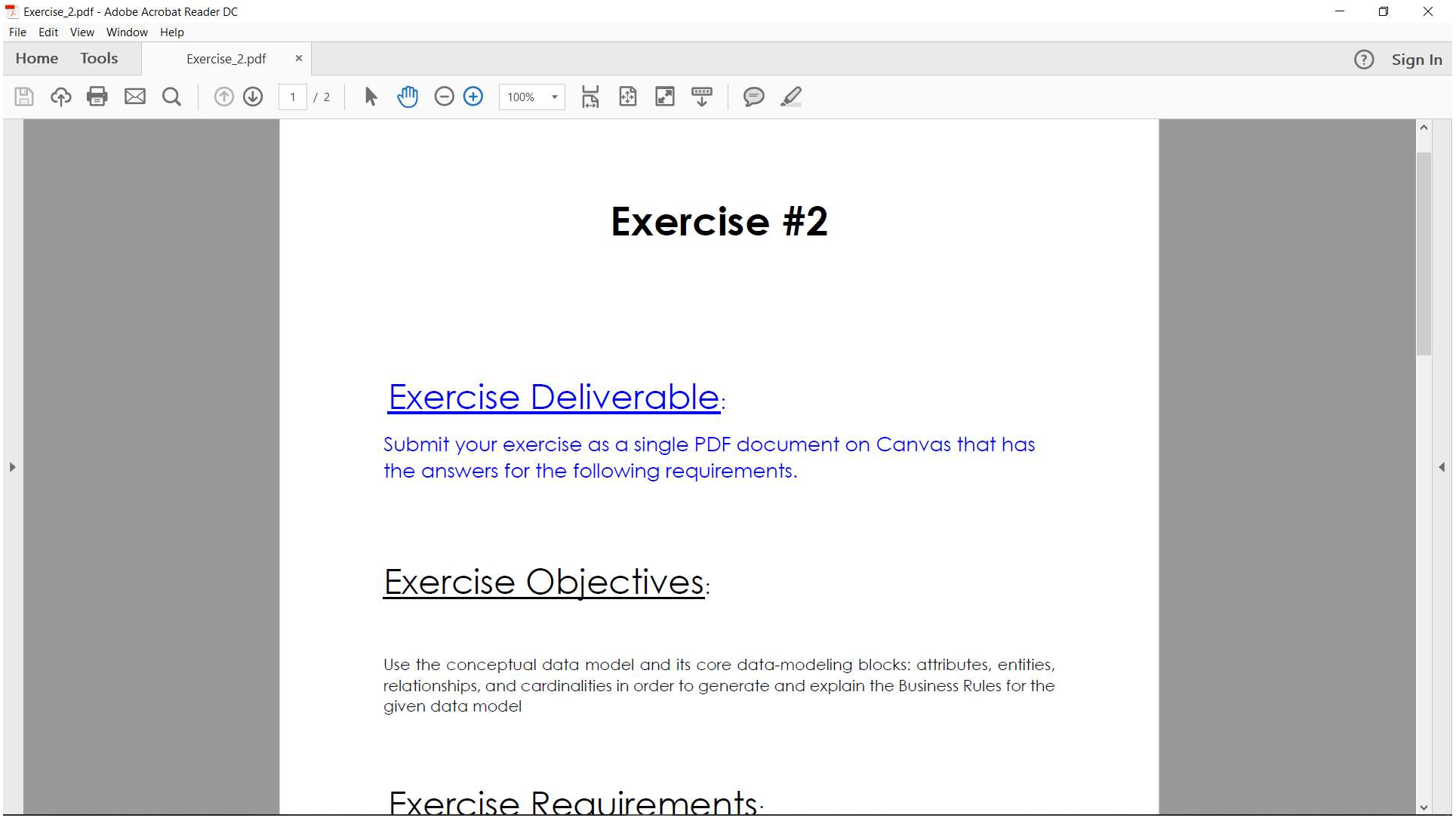
- Describes the way data are saved on storage media such as disks or tapes
- Requires the definition of physical storage and data access methods

Levels of Data Abstraction

LEVELS OF DATA ABSTRACTION

MODEL	DEGREE OF ABSTRACTION	FOCUS	INDEPENDENT OF
External	High	End-user views	Hardware and software
Conceptual	↕	Global view of data (database model independent)	Hardware and software
Internal	↕	Specific database model	Hardware
Physical	Low	Storage and access methods	Neither hardware nor software

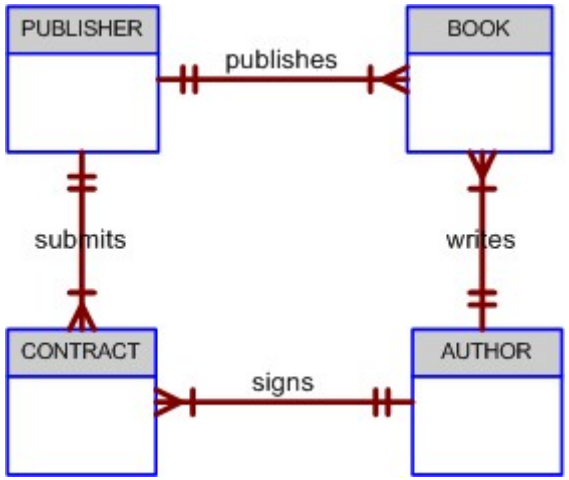
Exercise #2


Exercise_2.pdf - Adobe Acrobat Reader DC
File Edit View Window Help
Home Tools
Exercise_2.pdf
Sign In
1 / 2
100%
Exercise #2
Exercise Deliverable:
Submit your exercise as a single PDF document on Canvas that has the answers for the following requirements.
Exercise Objectives:
Use the conceptual data model and its core data-modeling blocks: attributes, entities, relationships, and cardinalities in order to generate and explain the Business Rules for the given data model
Exercise Requirements:

Exercise #2 - Walkthrough

- Assume that we have been given the following ERD, lets illustrate what you need to do for the ERD given in Exercise #2 requirements document

The Crow's Foot ERD



Exercise #2 – Walkthrough & Deliverable

Criteria	
Describe and List the Entities	Book, Author, …
Describe and List the Relationships between the entities	writes relationship: Author writes Book
Describe and List the Cardinalities of the Relationships	one-to-many cardinality of the writes relationship between Author and Book
Document the Business Rules	One publisher can publish many books.
	Each book is published by one publisher.
	A publisher can submit many (book) contracts.