**CH 1**:

**Data** : Raw-Data

**Information** : Result of processing raw data

**Metadata** :

* Data about data
* Integrates and manages business data
* Describes characteristics and relationships
* Places constraints on data entered into the database

**DBMS** : Database Management System

* The DBMS Manages the Interaction between the End User and the Database

Advantages:

* Better data integration and less data inconsistency–
  + **Data inconsistency:**

Different versions of the same data appear in different places

* Increased end-user productivity
* Improve:
  + Data sharing
  + Data security
  + Data access
  + Decision making
  + Data quality: Promoting accuracy, validity, and timeliness of data

Disadvantages:

* Cost
* Database packages are pricey
* Speed
  + Queries integrating numerous tables can be slow.
* Maintenance
  + Index rebuilding, database reengineering
* RDBMS do not scale well for big data
* Vendor Dependence\*

**DBA** : Database Administrator

**Structural Dependence**

* Access to a file is dependent on its structure
* All file system programs are modified to conform to a new file structure

**Structural Independence**

* File structure is changed without affecting the application’s ability to access the data

?: Data Structural Independence is better than Dependence.

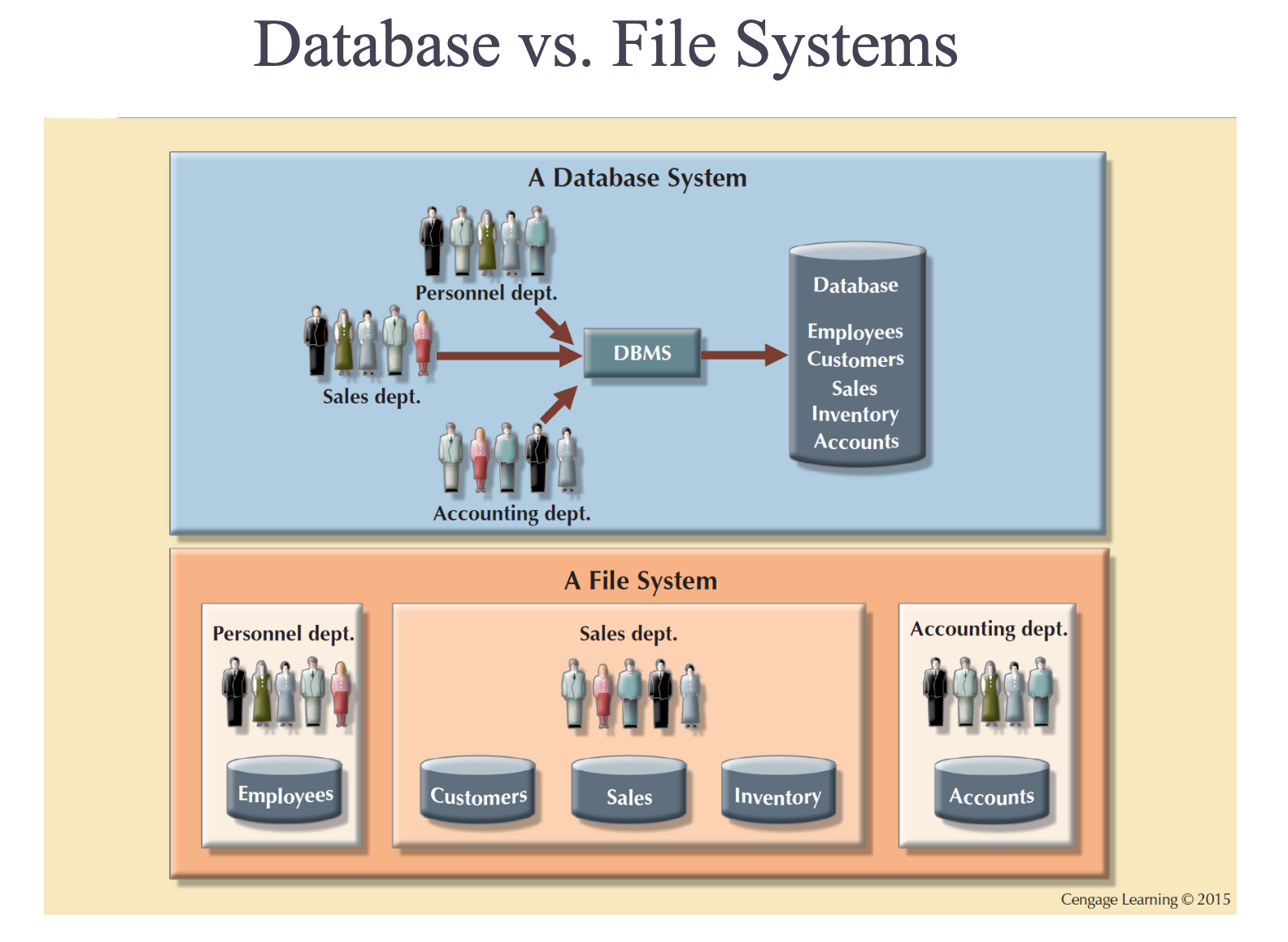
**Data Dependence**

* Data access changes when data storage characteristics change

**Data Independence**

* Data storage characteristics can be changed without affecting the program’s ability to access the data

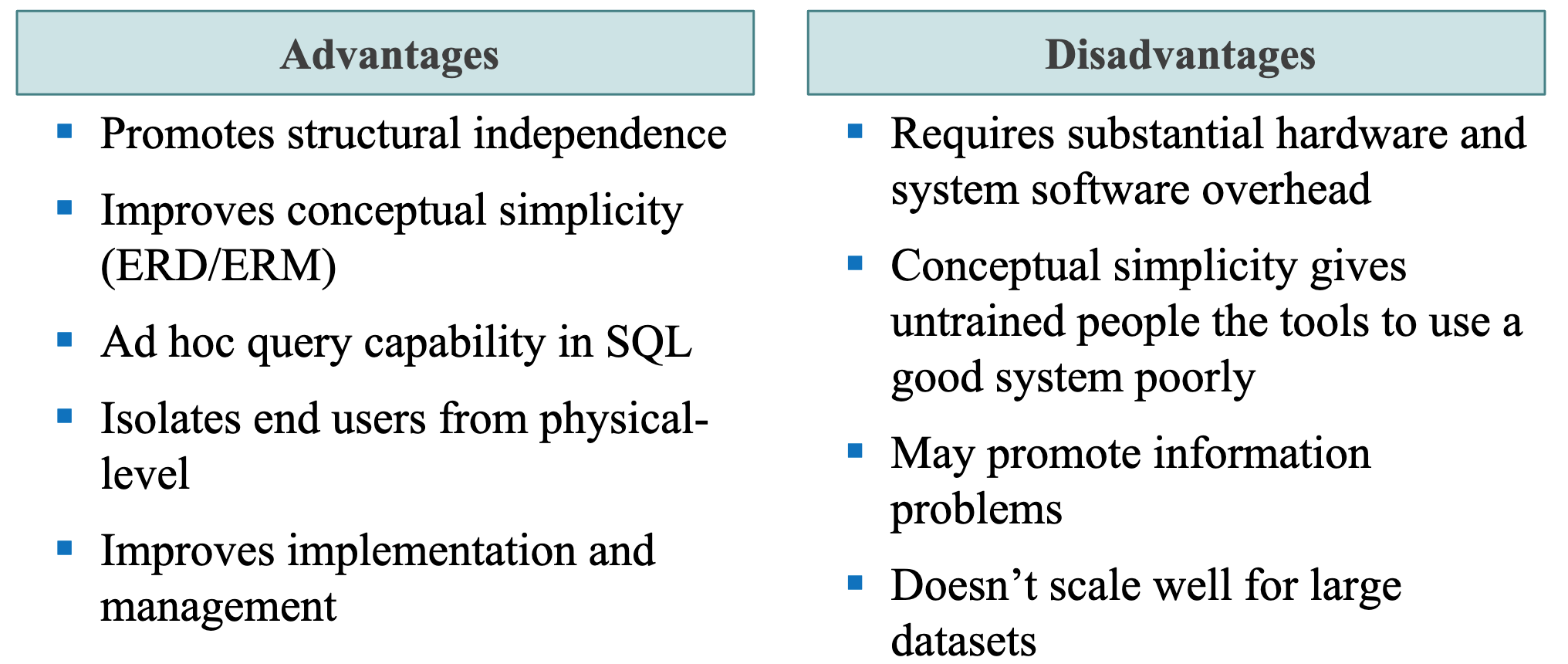
Database vs. File Systems (Google Doc)



**Disadvantages of database systems:**

* Increased costs
* Management complexity
* Maintaining currency
* Vendor dependence
* Frequent upgrade/replacement cycles

**CH 2**:

**RDBMS** : Relational Database Model System  


**Model** : Abstraction of a real-world object or event

**Data models**: Simplified representations of complex data structures

**Data modeling**: Iterative process of creating a specific data model for a problem domain

**The Entity Relationship Model**

* Introduced by Peter Chen in 1976
* Represents the conceptual model
* Graphically represents entities and relationships
* **Entity relationship diagram (ERD)** 
  + Graphical representations model database components
* Entity instance or entity occurrence
  + Rows in the relational table
* Connectivity
  + Describes relationship types

**Entities** : Distinct objects used to collect and store data.

Comprised of:

* Attributes: Characteristics about an entity
* Relationships: related attributes that link two entities.

**Attributes**

**Composite identifier** ( **Composite Key** ) : Primary key composed of more than one attribute  
  
**Composite attribute** : Attribute that can be subdivided to yield additional attributes

* Name, Address, Phone, DOB

**Simple attribute** : Attribute that cannot be subdivided

* Age

**Single-valued attribute** : Attribute that has only a single value

* JobTitle {professor}

**Derived attribute** : Attribute whose value is calculated from other attributes **Multivalued attributes**: Attributes that have many values

Hobbies {biking, hiking, clubbing}

Learn to avoid multivalued attributes

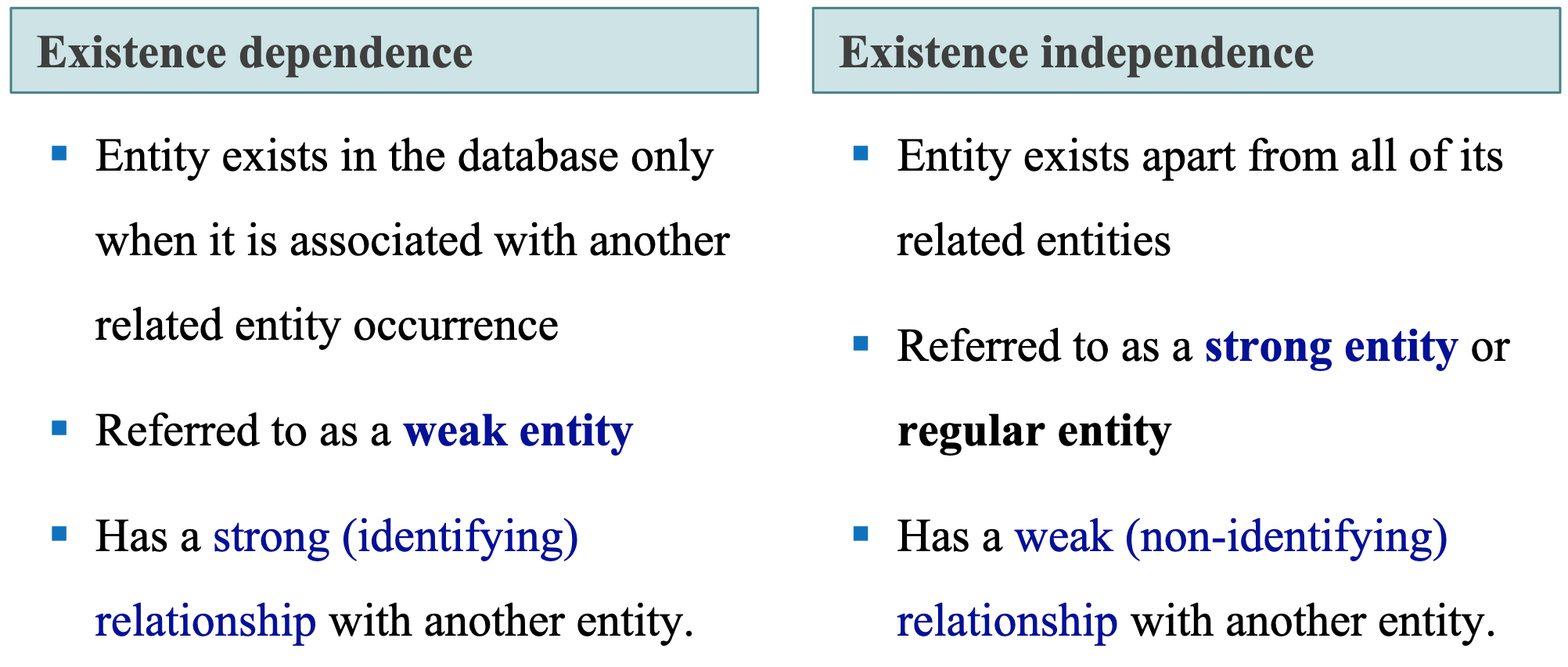
**Relationships**

Association between entities always operate in both directions **Participants** : Entities that participate in a relationship

**Connectivity** : Describes the relationship classification

**Cardinality** : Expresses the minimum and maximum number of entity occurrences associated with one occurrence of related entity

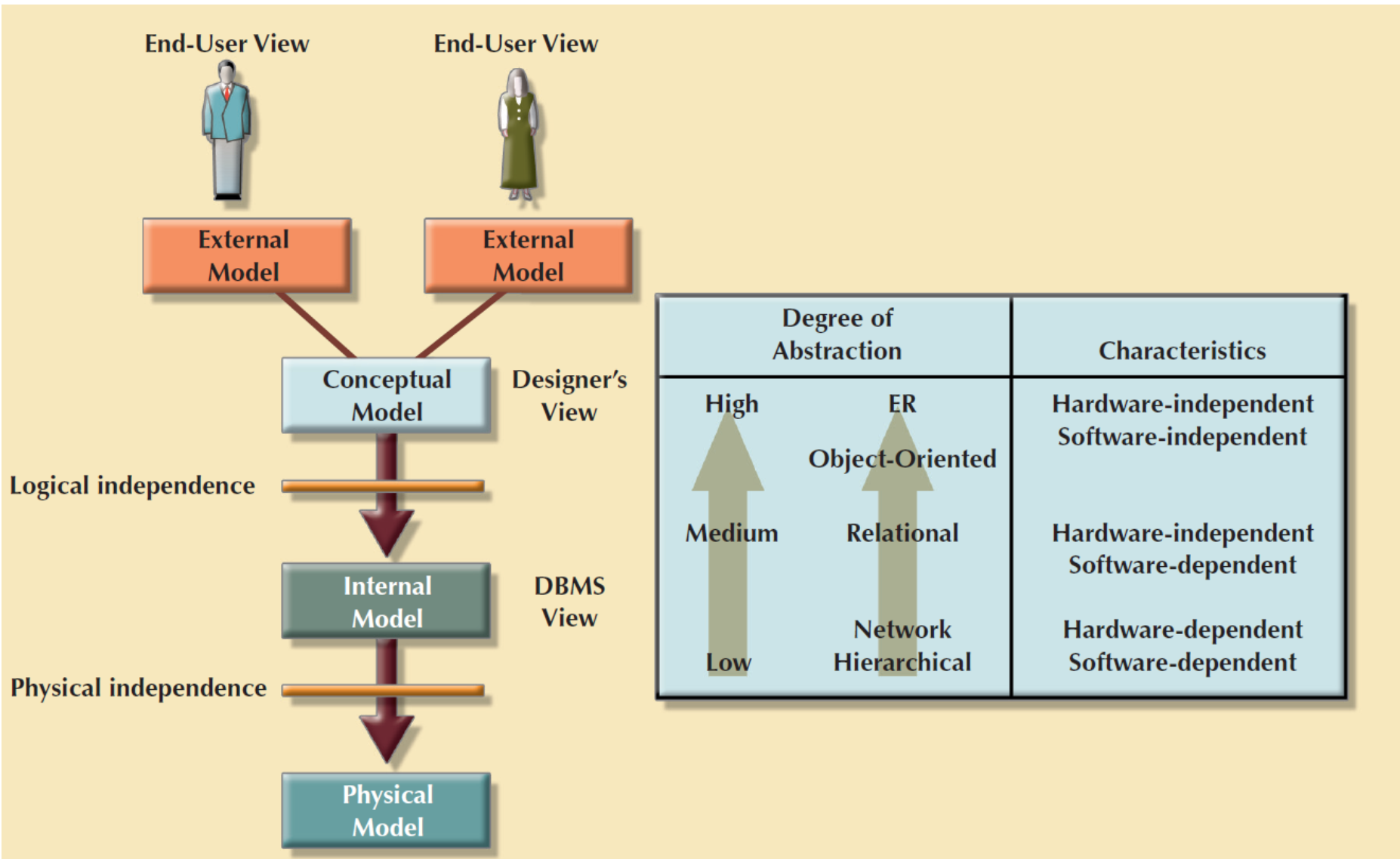
**Entity Dependence**

****

**Associative Entities**

* Also known as composite or bridge entities (Thoms calls them intersection tables)
* Used to represent an M:N relationship between two or more entities
* Is in a 1:M relationship with the parent entities
  + Composed of the primary key attributes of each parent entity
* May also contain additional attributes that play no role in connective process

**Data Abstraction Levels**



**CH: 3**

**Characteristics of a Relational Table**

**1. Values are atomic.**

**2. Column values are of the same kind.**

**3. Each row is unique.**

**4. The sequence of columns is insignificant.**

**5. The sequence of rows is insignificant.**

**6. Each column must have a unique name.**

1. **Values Are Atomic**

This property implies that columns in a relational table are not repeating group or arrays. Such tables are referred to as being in the "first normal form" (1NF). The atomic value property of relational tables is important because it is one of the cornerstones of the relational model.

The key benefit of the one value property is that it simplifies data manipulation logic.

1. **Column Values Are of the Same Kind**

In relational terms this means that all values in a column come from the same domain. A domain is a set of values which a column may have. For example, a Monthly\_Salary column contains only specific monthly salaries. It never contains other information such as comments, status flags, or even weekly salary.

This property simplifies data access because developers and users can be certain of the type of data contained in a given column. It also simplifies data validation. Because all values are from the same domain, the domain can be defined and enforced with the Data Definition Language (DDL) of the database software.

1. **Each Row is Unique**

This property ensures that no two rows in a relational table are identical; there is at least one column, or set of columns, the values of which uniquely identify each row in the table. Such columns are called primary keys and are discussed in more detail in **Relationships and Keys**.

This property guarantees that every row in a relational table is meaningful and that a specific row can be identified by specifying the primary key value.

1. **The Sequence of Columns is Insignificant**

This property states that the ordering of the columns in the relational table has no meaning. Columns can be retrieved in any order and in various sequences. The benefit of this property is that it enables many users to share the same table without concern of how the table is organized. It also permits the physical structure of the database to change without affecting the relational tables.

1. **The Sequence of Rows is Insignificant**

This property is analogous to the one above but applies to rows instead of columns. The main benefit is that the rows of a relational table can be retrieved in different order and sequences. Adding information to a relational table is simplified and does not affect existing queries.

1. **Each Column Has a Unique Name**

Because the sequence of columns is insignificant, columns must be referenced by name and not by position. In general, a column name need not be unique within an entire database but only within the table to which it belongs.

**Keys**

**Primary Key (PK)**