**Dalhousie University**
**Faculty of Computer Science**

# CSCI 2141 – Intro to Database Systems

## Week 2 – Data Modelling

# What we cover this week

- Data Modelling
  - Various types of data models
  - Data modelling notations
  - How to design a database from business requirements
- Business rules
- Bad design versus good design
- Practice

# Why Design Databases?

Let’s see an example

- We need to create a database to store information about an employee’s skill certifications:
  - What data do we need to save?
    - Employee number
    - Name
    - Title
    - Date of hiring
    - Skills (can be several)
    - Data each skill achieved
- Employee table could be created as follows:

# Poor Database Design

| ID | ENum | Name | Title | HireDate | Skill1 | Skill1Dat |
|---|---|---|---|---|---|---|
| 1 | 02345 | Brian Oates | DBA | 2/14/1995 | Basic Database Management | 2/14/20 |
| 2 | 08273 | Marco Bienz | Analyst | 7/28/2006 | Basic Web Design | 3/8/20 |
| 3 | 06234 | Jasmine Patel | Programmer | 8/10/2005 | Basic Web Design | 8/10/20 |
| 4 | 03373 | Franklin Johnson, Jr. | Purchasing Agent | 3/15/2002 | Advanced Spreadsheets | 6/20/20 |
| 5 | 13567 | Almond, Robert | Analyst | 9/30/2012 | Basic Process Modeling | 9/30/20 |
| 6 | 10282 | Richardson, Amanda | Clerk | 4/11/2011 | | |
| 7 | 09382 | Susan Mathis | Database Programmer | 8/2/2010 | Basic DB Design | 8/2/20 |
| 8 | 14311 | Duong, Lee | Programmer | 9/1/2014 | Basic Web Design | 9/1/20 |
| 9 | | | | | Master Database Programming | |
| 10 | | | | | Basic Spreadsheets | |
| 11 | 09002 | Wade Gaither | Clerk | 5/20/2010 | Advanced Spreadsheets | 5/16/20 |
| 12 | 13383 | Raymond F. Matthews | Programmer | 3/12/2012 | Basic C# Programming | 3/12/20 |
| 13 | 09283 | Chavez, Juan | Clerk | 7/4/2010 | | |
| 14 | 04893 | Patricia Richards | DBA | 6/11/2004 | Advanced Database Management | 6/11/20 |
| 15 | 13932 | Lee, Megan | Programmer | 9/29/2013 | | |

# Good Database Design

## Table name: EMPLOYEE

| Employee_ID | Employee_FName | Employee_LName | Employee_HireDate | Employee_Title |
|---|---|---|---|---|
| 02345 | Johnny | Jones | 2/14/1995 | DBA |
| 03373 | Franklin | Johnson | 3/15/2002 | Purchasing Agent |
| 04893 | Patricia | Richards | 6/11/2004 | DBA |
| 06234 | Jasmine | Patel | 8/10/2005 | Programmer |
| 08273 | Marco | Bienz | 7/28/2006 | Analyst |
| 09002 | Ben | Joiner | 5/20/2010 | Clerk |
| 09283 | Juan | Chavez | 7/4/2010 | Clerk |
| 09382 | Jessica | Johnson | 8/2/2010 | Database Programmer |
| 10282 | Amanda | Richardson | 4/11/2011 | Clerk |
| 13383 | Raymond | Matthews | 3/12/2012 | Programmer |
| 13567 | Robert | Almond | 9/30/2012 | Analyst |
| 13932 | Megan | Lee | 9/29/2013 | Programmer |
| 14311 | Lee | Duong | 9/1/2014 | Programmer |

## Table name: CERTIFIED

| Employee_ID | Skill_ID | Certified_Date |
|---|---|---|
| 02345 | 100 | 2/14/2002 |
| 02345 | 110 | 8/9/2003 |
| 02345 | 180 | 2/14/2005 |
| 03373 | 120 | 6/20/2011 |
| 04893 | 180 | 6/11/2006 |
| 04893 | 220 | 9/20/2012 |
| 06234 | 110 | 8/10/2007 |
| 06234 | 200 | 8/10/2007 |
| 06234 | 210 | 1/29/2012 |
| 08273 | 110 | 3/8/2009 |
| 08273 | 190 | 8/19/2012 |
| 09002 | 110 | 5/16/2013 |
| 09002 | 120 | 5/16/2013 |
| 09382 | 140 | 8/2/2012 |
| 09382 | 210 | 8/2/2012 |
| 09382 | 220 | 5/1/2013 |
| 13383 | 170 | 3/12/2014 |
| 13567 | 130 | 9/30/2014 |
| 13567 | 140 | 5/23/2015 |
| 14311 | 110 | 9/1/2016 |

## Table name: SKILL

| Skill_ID | Skill_Name | Skill_Description |
|---|---|---|
| 100 | Basic Database Management | Create and manage database user accounts. |
| 110 | Basic Web Design | Create and maintain HTML and CSS documents. |
| 120 | Advanced Spreadsheets | Use of advanced functions, user-defined functions, and macroing. |
| 130 | Basic Process Modeling | Create core business process models using standard libraries. |
| 140 | Basic Database Design | Create simple data models. |
| 150 | Master Database Programming | Create integrated trigger and procedure packages for a distributed environment. |
| 160 | Basic Spreadsheets | Create single tab worksheets with basic formulas |
| 170 | Basic C# Programming | Create single-tier data aware modules. |
| 180 | Advanced Database Management | Manage Database Server Clusters. |
| 190 | Advance Process Modeling | Evaluate and Redesign cross-functional internal and external business processes. |
| 200 | Advanced C# Programming | Create multi-tier applications using multi-threading |
| 210 | Basic Database Manipulation | Create simple data retrieval and manipulation statements in SQL. |
| 220 | Advanced Database Manipulation | Use of advanced data manipulation methods for multi-table inserts, set operations, and correlated subqueries. |

# Exercise 1

a) Identify the serious data redundancy problems in the file structure shown below

b) How could you improve the EMP_NAME and EMP_PHONE contents shown below?

c) Identify the various data sources in the file shown below. What new files should you create to eliminate the data redundancies?

d) Draw a data model showing an improved design

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CODE | JOB_CHG_HOUR | PROJ_HOURS | EMP_PHONE |
|---|---|---|---|---|---|---|---|
| 1 | Hurricane | 101 | John D. Newson | EE | 85.00 | 13.3 | 653-234-3245 |
| 1 | Hurricane | 105 | David F. Schwann | CT | 60.00 | 16.2 | 653-234-1123 |
| 1 | Hurricane | 110 | Anne R. Ramoras | CT | 60.00 | 14.3 | 615-233-5568 |
| 2 | Coast | 101 | John D. Newson | EE | 85.00 | 19.8 | 653-234-3254 |
| 2 | Coast | 108 | June H. Sattlemeir | EE | 85.00 | 17.5 | 905-554-7812 |
| 3 | Satellite | 110 | Anne R. Ramoras | CT | 62.00 | 11.6 | 615-233-5568 |
| 3 | Satellite | 105 | David F. Schwann | CT | 26.00 | 23.4 | 653-234-1123 |
| 3 | Satelite | 123 | Mary D. Chen | EE | 85.00 | 19.1 | 615-233-5432 |
| 3 | Satellite | 112 | Allecia R. Smith | BE | 85.00 | 20.7 | 615-678-6879 |

# Solution 1

a) Ask yourself these questions:

- How many changes if JOB_CODE = EE changes from $85.00 to $90.00?
- What information do you lose if employee Allecia R. Smith is deleted from the file?
- What is the JOB_CHG_HOUR value for the CT job?

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CODE | JOB_CHG_HOUR | PROJ_HOURS | EMP_PHONE |
|---|---|---|---|---|---|---|---|
| 1 | Hurricane | 101 | John D. Newson | EE | 85.00 | 13.3 | 653-234-3245 |
| 1 | Hurricane | 105 | David F. Schwann | CT | 60.00 | 16.2 | 653-234-1123 |
| 1 | Hurricane | 110 | Anne R. Ramoras | CT | 60.00 | 14.3 | 615-233-5568 |
| 2 | Coast | 101 | John D. Newson | EE | 85.00 | 19.8 | 653-234-3254 |
| 2 | Coast | 108 | June H. Sattlemeir | EE | 85.00 | 17.5 | 905-554-7812 |
| 3 | Satellite | 110 | Anne R. Ramoras | CT | 62.00 | 11.6 | 615-233-5568 |
| 3 | Satellite | 105 | David F. Schwann | CT | 26.00 | 23.4 | 653-234-1123 |
| 3 | Satelite | 123 | Mary D. Chen | EE | 85.00 | 19.1 | 615-233-5432 |
| 3 | Satellite | 112 | Allecia R. Smith | BE | 85.00 | 20.7 | 615-678-6879 |

# Solution 1

b) Consider these issues:

- How would you create a list of employees based on their Last name? First name?
- How would you be able to print a directory based on area codes?

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CODE | JOB_CHG_HOUR | PROJ_HOURS | EMP_PHONE |
|---|---|---|---|---|---|---|---|
| 1 | Hurricane | 101 | John D. Newson | EE | 85.00 | 13.3 | 653-234-3245 |
| 1 | Hurricane | 105 | David F. Schwann | CT | 60.00 | 16.2 | 653-234-1123 |
| 1 | Hurricane | 110 | Anne R. Ramoras | CT | 60.00 | 14.3 | 615-233-5568 |
| 2 | Coast | 101 | John D. Newson | EE | 85.00 | 19.8 | 653-234-3254 |
| 2 | Coast | 108 | June H. Sattlemeir | EE | 85.00 | 17.5 | 905-554-7812 |
| 3 | Satellite | 110 | Anne R. Ramoras | CT | 62.00 | 11.6 | 615-233-5568 |
| 3 | Satellite | 105 | David F. Schwann | CT | 26.00 | 23.4 | 653-234-1123 |
| 3 | Satelite | 123 | Mary D. Chen | EE | 85.00 | 19.1 | 615-233-5432 |
| 3 | Satellite | 112 | Allecia R. Smith | BE | 85.00 | 20.7 | 615-678-6879 |

# Solution 1

c) Consider which of the fields are Employee characteristics?

- Project name?
- Job data e.g. JOB_CHG_HOUR
- Project hours?

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CODE | JOB_CHG_HOUR | PROJ_HOURS | EMP_PHONE |
|---|---|---|---|---|---|---|---|
| 1 | Hurricane | 101 | John D. Newson | EE | 85.00 | 13.3 | 653-234-3245 |
| 1 | Hurricane | 105 | David F. Schwann | CT | 60.00 | 16.2 | 653-234-1123 |
| 1 | Hurricane | 110 | Anne R. Ramoras | CT | 60.00 | 14.3 | 615-233-5568 |
| 2 | Coast | 101 | John D. Newson | EE | 85.00 | 19.8 | 653-234-3254 |
| 2 | Coast | 108 | June H. Sattlemeir | EE | 85.00 | 17.5 | 905-554-7812 |
| 3 | Satellite | 110 | Anne R. Ramoras | CT | 62.00 | 11.6 | 615-233-5568 |
| 3 | Satellite | 105 | David F. Schwann | CT | 26.00 | 23.4 | 653-234-1123 |
| 3 | Satelite | 123 | Mary D. Chen | EE | 85.00 | 19.1 | 615-233-5432 |
| 3 | Satellite | 112 | Allecia R. Smith | BE | 85.00 | 20.7 | 615-678-6879 |

# Data Modelling

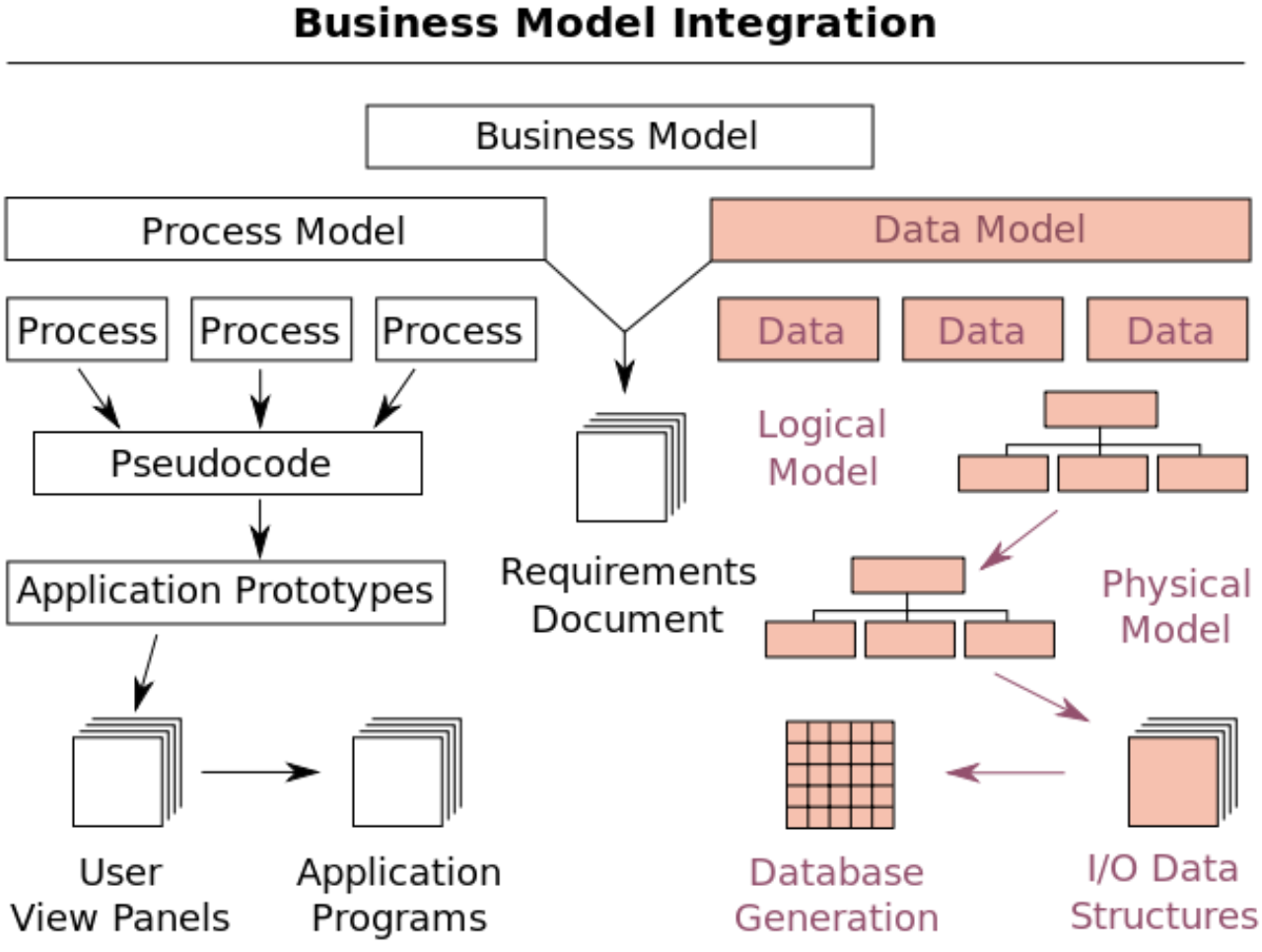
# Data Modeling

- Data modeling is a process used to define and analyze data requirements needed to support the business processes within the scope of corresponding information systems in organizations.
- 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

# Importance of Data Models

- Are a communication tool
- Give an overall view of the database
- Organize data for various users
- Are an abstraction for the creation of good database

# Process Model versus Data Model

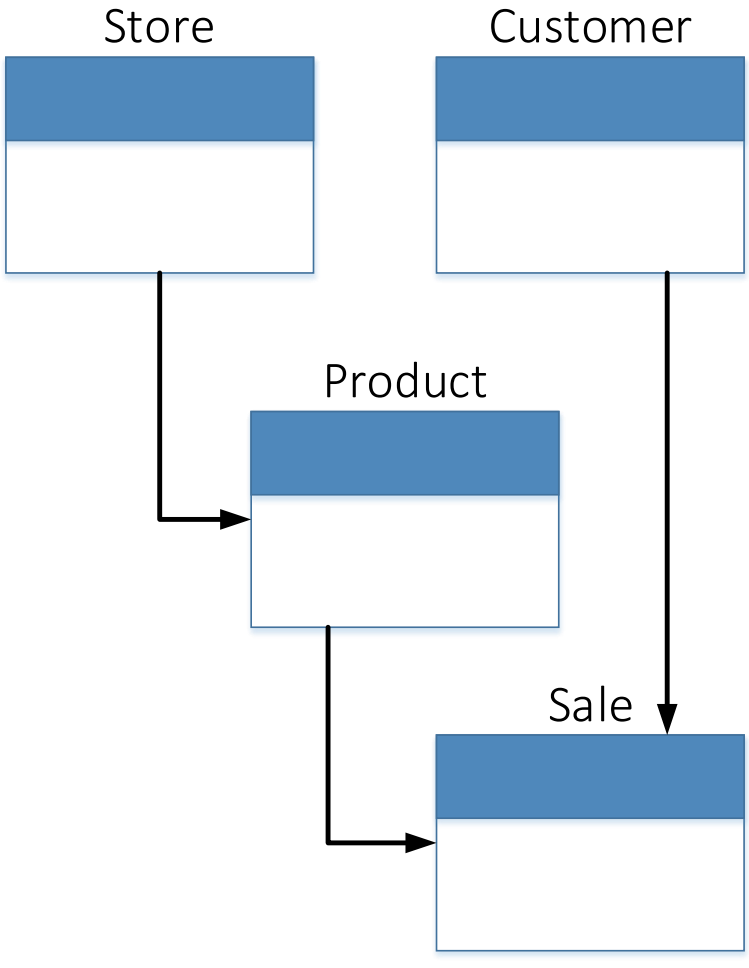



Source: Wikipedia (CCL) - Data_modeling_context.jpg

# Type of Data Models

- Conceptual Data Model
  - Identifies the highest level relationship between entities
  - Facilitates communication, and integration between systems, processes and organizations
  - Often expressed using an entity-relationship model (ERM)
- Logical Data Model
- Physical Data Model

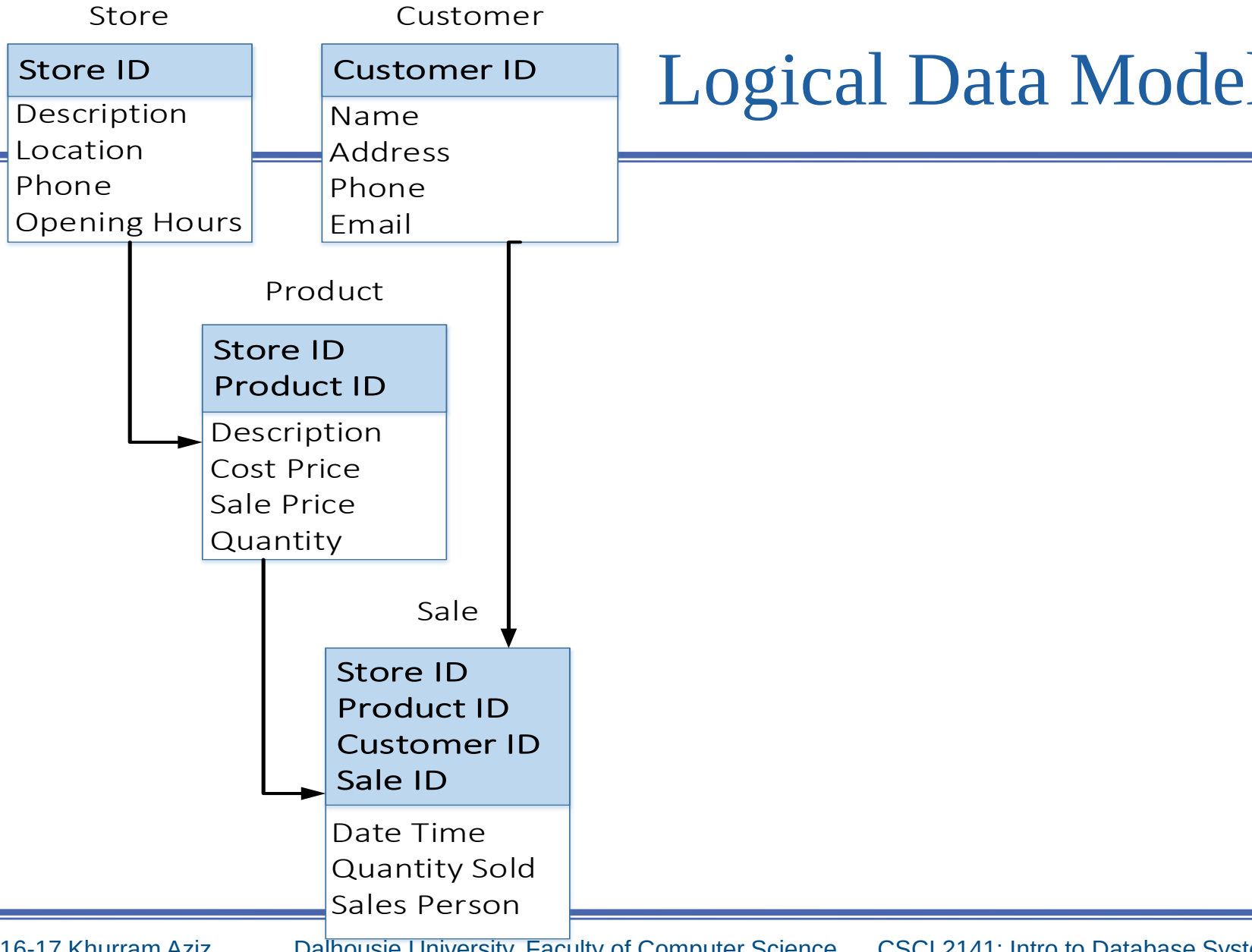
# Conceptual Data Model



# Type of Data Models

- Conceptual Data Model
- Logical Data Model
  - Expands on the CDM with more details
  - Describes the data in as much detail as possible, without regards to its physical implementation
  - Contains attributes in addition to entities and relationships
  - Normalization done at this level
- Physical Data Model

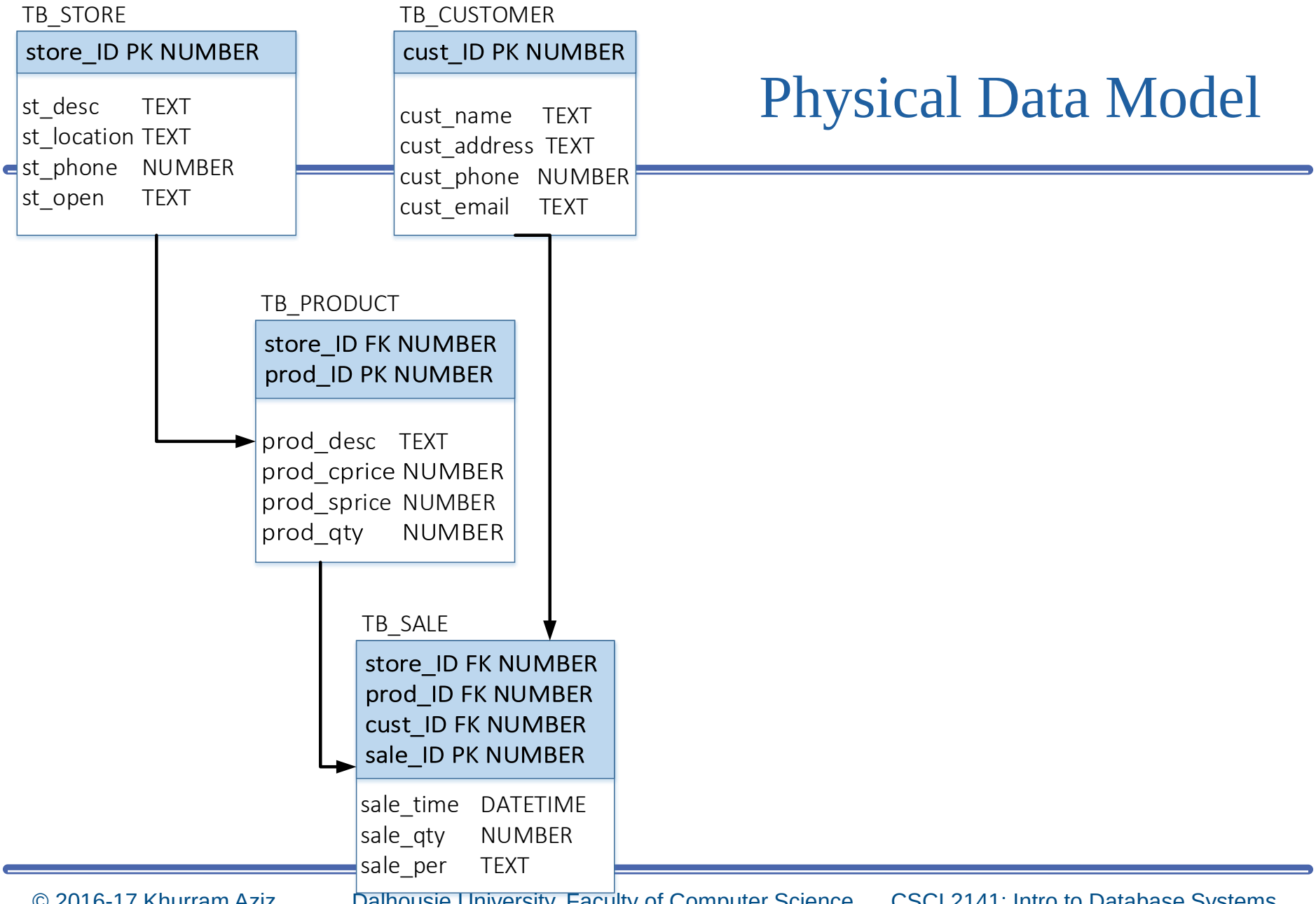
# Logical Data Model



# Type of Data Models

- Conceptual Data Model
- Logical Data Model
- Physical Data Model
  - Models how actual database will be built
  - Shows the complete table structure, keys, and relationships between tables
  - Dependent on the database management system used

# Physical Data Model



# Type of Data Models

| Model Feature | Conceptual Data Model | Logical Data Model | Physical Data Model |
|---|---|---|---|
| Entity Names | Included | Included | |
| Entity Relationships | Included | Included | |
| Attributes | | Included | |
| Keys | | Included | Included |
| Table Names | | | Included |
| Column Names | | | Included |
| Column Data Types | | | Included |
| Column Data Lengths | | | Included |

# How to do Data Modelling

# Data Modeling

- Data modeling is easy. Its like drawing a sentence
  - A course may ***contain*** many students


- Where do these sentences come from?
  - These are business rules

# 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

Policy makers

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

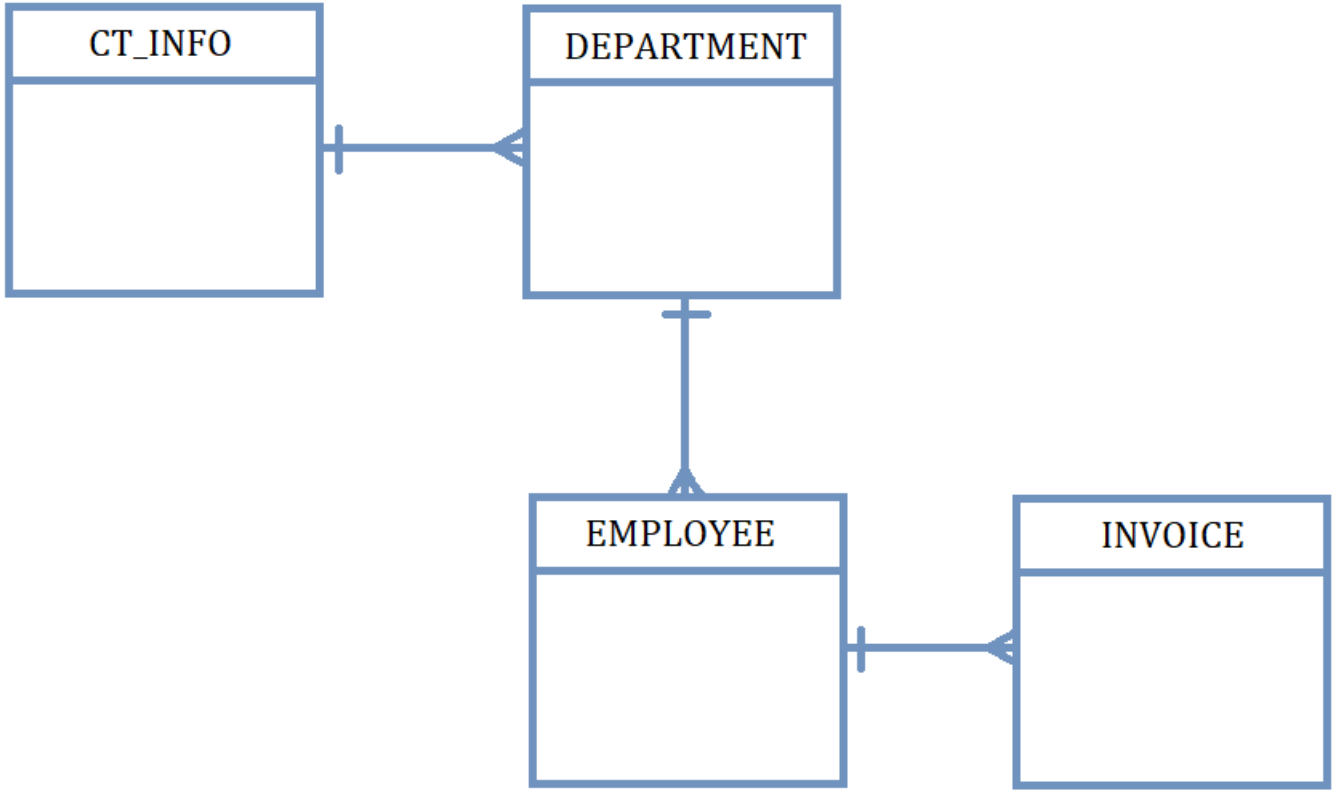
# Translating Business Rules into Data Model Components

- 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?

# Data Modelling – Example 1

- Map the following business rules to a conceptual data model:
  - Canadian Tire has many outlets
  - Each outlet has many departments
  - Each department has many employees
  - Each employee can prepare several invoices

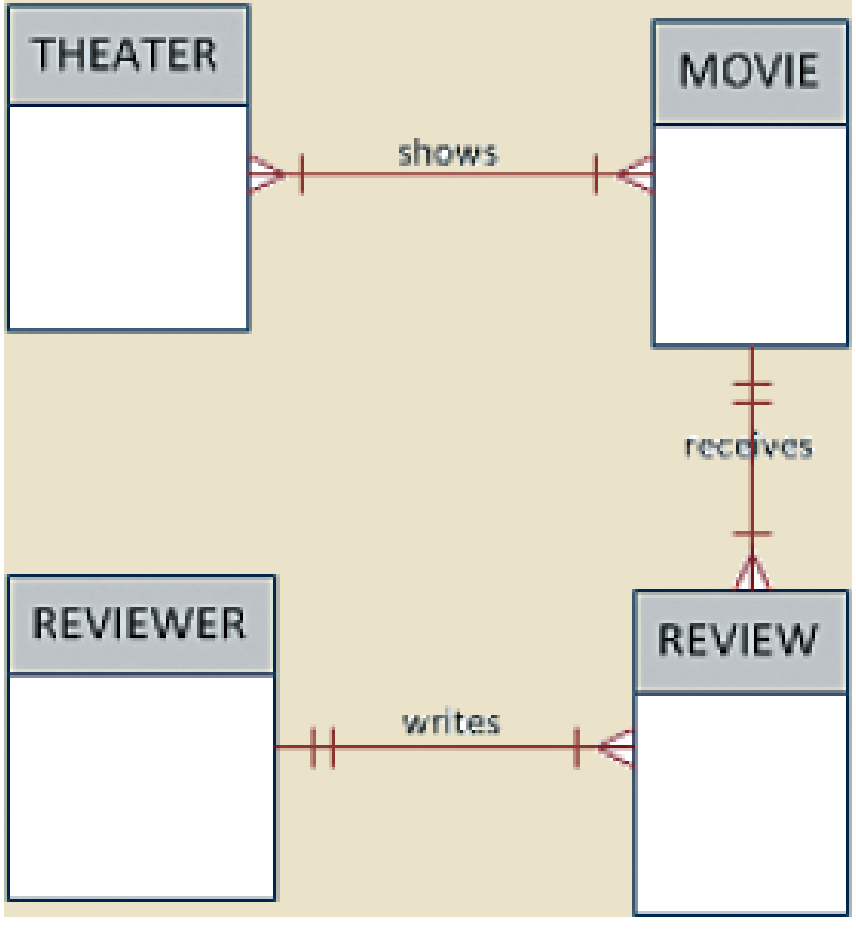
# Data Modelling – Example 1 (Solution)



# Data Modelling – Example 2

- Map the following business rules to a conceptual data model:
  - A theater show many movies.
  - A movie can be shown in many theaters.
  - A movie can receive many reviews.
  - Each review is for a single movie.
  - A reviewer can write many reviews.
  - Each review is written by a single reviewer.

# Data Modelling – Example 2 (Solution)



# Data Modelling

- Business rules are extracted from client's requirements and may not be as clear cut as shown previously
  - How to design a database in that case?

- Let's see an example

# Data Modelling

- Scenario:
  - A college has two departments. Teachers from each department list their course offerings before the beginning of each semester. These lists are collected and displayed on the enrollment website. Each department offers courses to students. Students can enroll in several courses in each semester. Fee for the courses needs to be deposited before the start of the semester

# Data Modelling – Solution

- Step 1: Identify key data things
  - Hint: Identify the nouns (things or concepts) in the scenario statement
- Step 2: Define the key “nouns”
  - Hint: Think from the perspective of all stakeholders
- Step 3: Construct a Conceptual Data Model
  - Hint: Use the key “nouns” identified and their relationships with each other
- Step 4: Construct a Logical Data Model

# Data Modelling – Solution

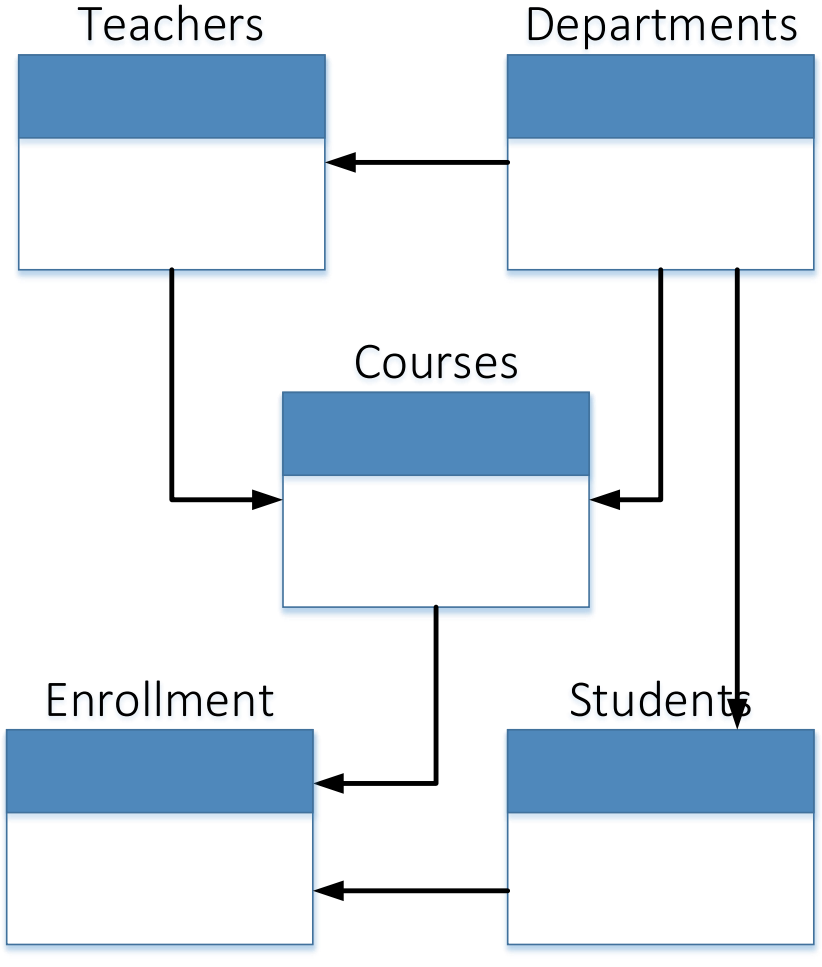
- Step 1: Identify key data things
  - A **college** has two **departments**. **Teachers** from each department list their **course** offerings before the beginning of each **semester**. These lists are collected and displayed on the **enrollment** website. Each department **offers courses to students**. **Students** can **enroll in several courses** in each semester. **Fee** for the courses needs to be deposited before the start of the semester

# Data Modelling – Solution

- Step 2: Define the key nouns, for example:
  - Students
    - Study in a department at the college
    - Can enroll in several courses
    - Have to pay fee
- Note: It is important to avoid assumptions
  - Consult relevant stake holders i.e. students, college administration, faculty, …?
  - Remember the difference between “must” and “should”

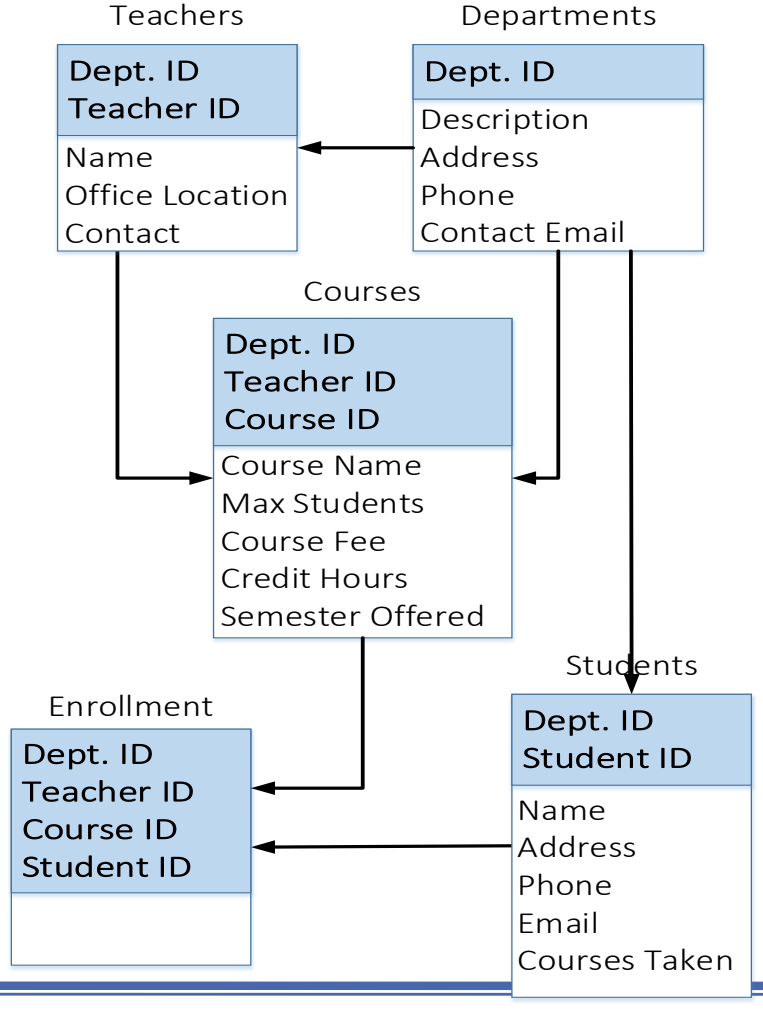
# Solution – Conceptual Data Model

Step 3:
Construct a Conceptual Data Model



# Solution – Logical Data Model

Step 4:
Construct a Logical Data Model



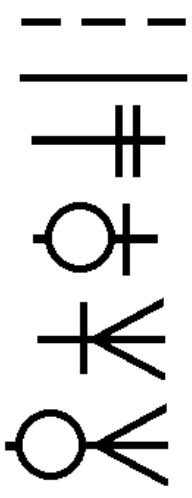
# Naming Conventions

- Entity names - Required to:
  - Be descriptive of the objects in the business environment
  - Use terminology that is familiar to the users
- Attribute name - Required to be descriptive of the data represented by the attribute
- Proper naming:
  - Facilitates communication between parties
  - Promotes self-documentation

# 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

# Types of Relationships



Weak relationship

Strong relationship

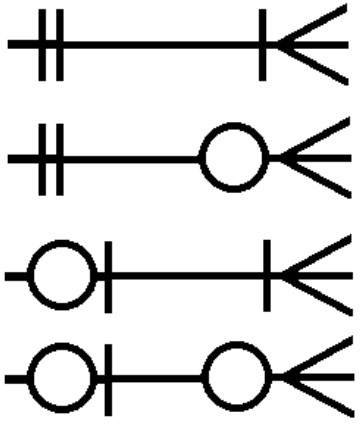
One and only one

Zero or one

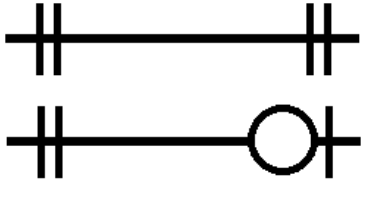
1 or more

Zero or more

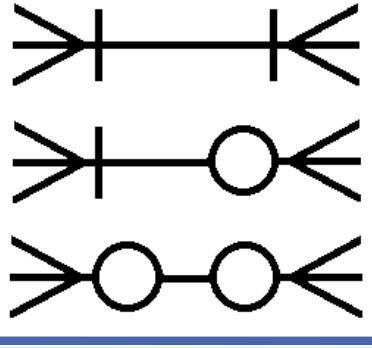
**One-to-Many (1:M)**



**One-to-One (1:1)**



**Many-to-Many (M:N)**

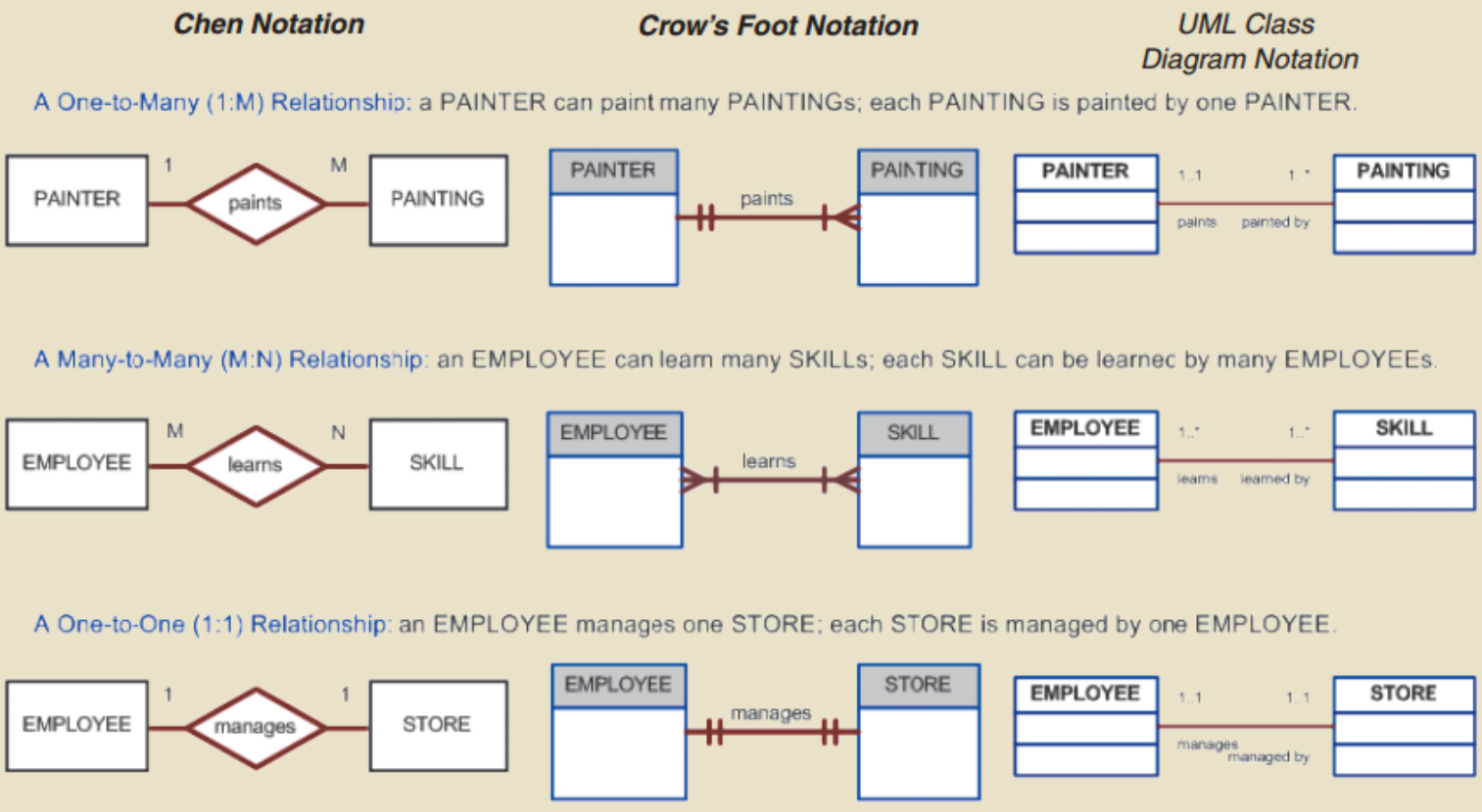


# Data Modeling Notations

- Several notations used for high-level data models
  - Choice depends on purpose and audience
  - Also depends on the choice of tool used, if any
  - Should not be tied down to using only one notation
    - Use organization's standard notation, if any
- Recommended to use tools that can generate Data Definition Language (DDL) or scripts for creating databases

# Data Modelling Notations

## FIGURE 2.3 THE ER MODEL NOTATIONS



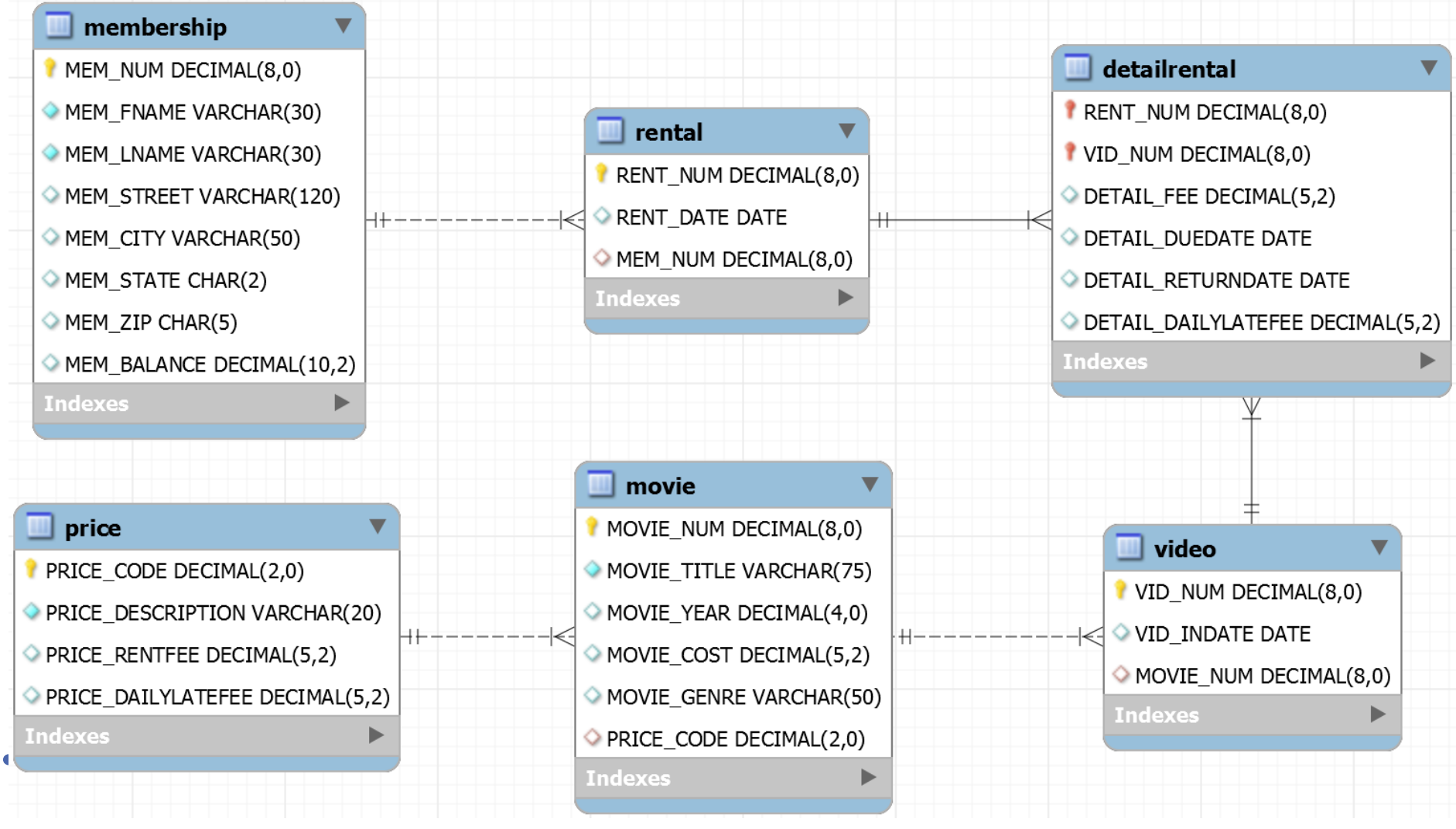
# Group Exercise

- In groups discuss and sketch a conceptual, logical and physical data model for a database that stores “all” data for a video rental store
- Step 1: Identify important business rules. For example,
  - The store may have several copies of a movie
  - Regular customers get 10% discount on new movies
  - A movie can be rented for 3 days
- Step 2: Identify entities and relationships
- Step 3: Identify attributes of each entity

# Group Exercise

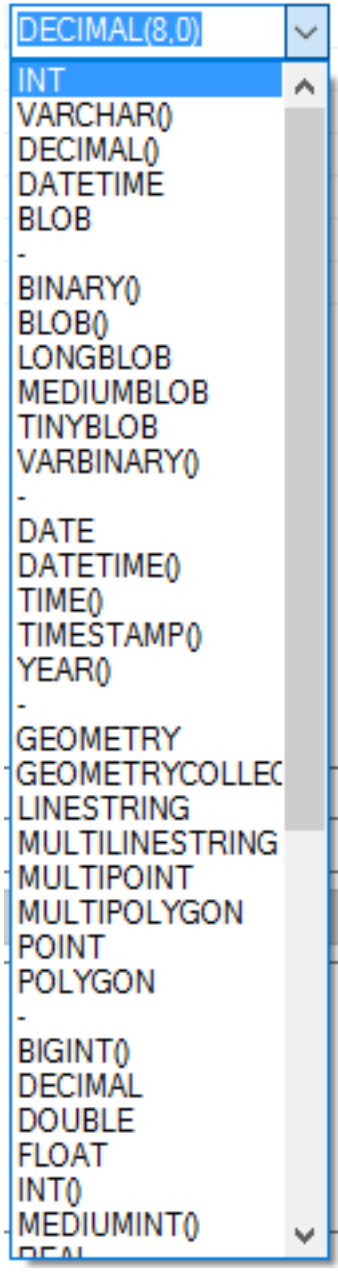
- Step 4: For a physical model, identify the constraints and the datatype for each attribute
- For this store, you might want to consider:
  - How regular customers will be identified?
  - How new movies are identified?
  - How rental price is determined?
  - What happens when movies are returned late?
  - How are movies organized?

# Example – ERD of a Video Rental Database



# Data Types

- Databases support the following data types:
  - Numeric
  - Character (Text, String)
  - Date (DateTime)
  - Logical




← MySQL data types

MS Access data types →