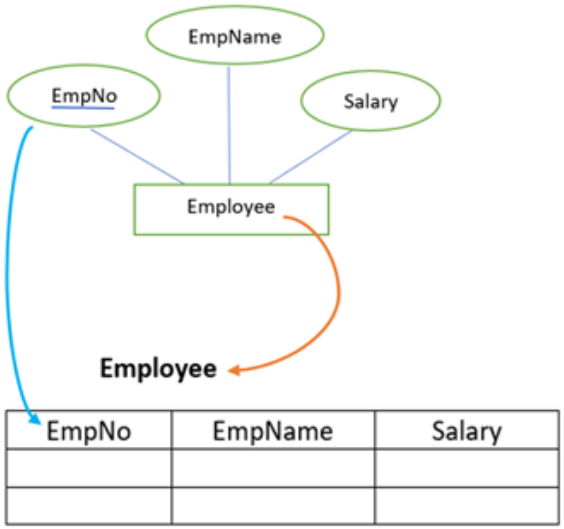


Reduction of an E-R Schema to Tables

Dr. L.M. Jenila Livingston

VIT Chennai

Steps in Database Design

- **Requirements Analysis**
  - user needs; what must database do?
- **Conceptual Design**
  - high level descr (often done with ER model)
- **Logical Design**
  - translate ER into DBMS data model
- **Schema Refinement**
  - consistency, normalization
- **Physical Design**
  - indexes, disk layout
- **Security Design**
  - who accesses what, and how

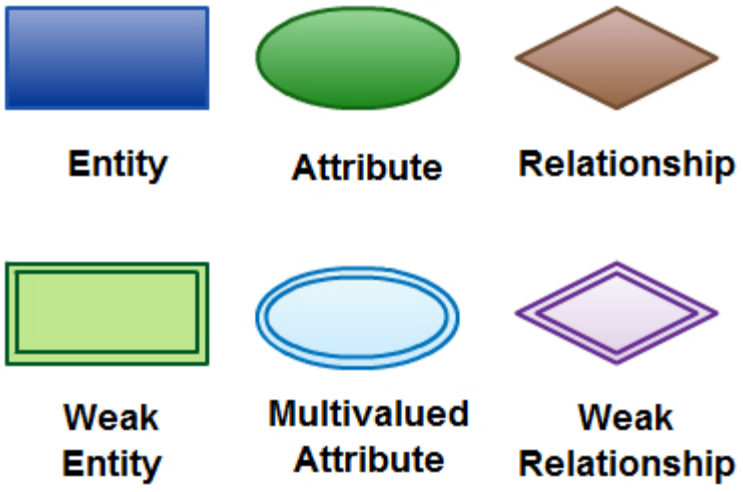
Reduction of an E-R Schema to Tables

- Converting an E-R diagram to a relational database:
  - Each entity set is converted to its' own table.
  - Each relationship *can be* (but may not be) converted to its' own table.

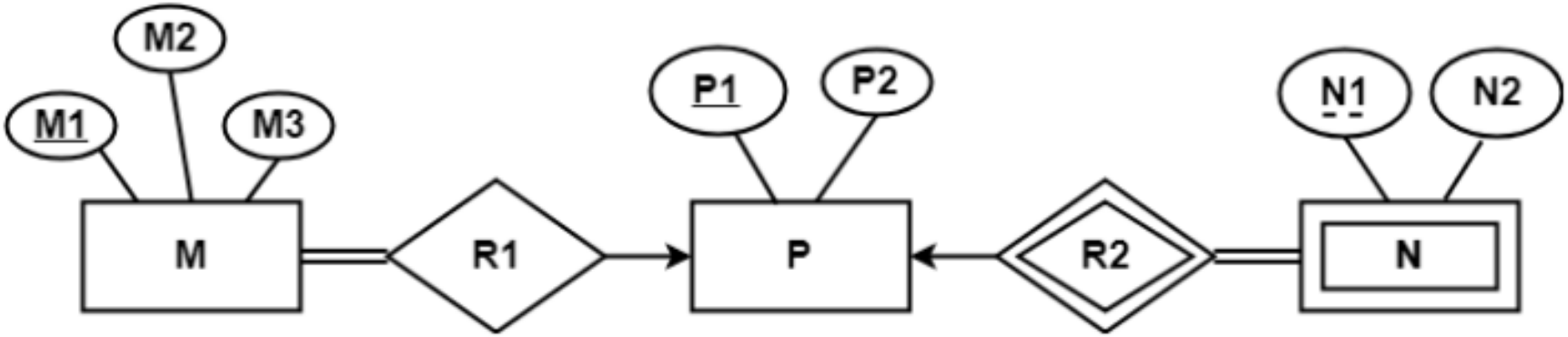
- Each table has a number of columns, which generally corresponding to the attributes in the corresponding entity or relationship set.

- The resulting tables can be modified in a variety of ways to support performance, space, or other requirements.

ER Diagram

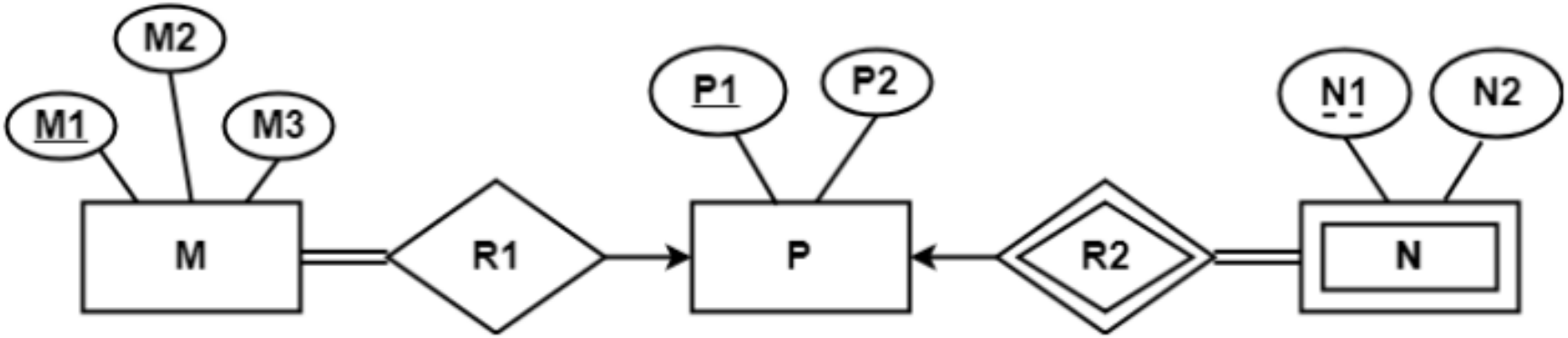


Elements in ER diagrams



1. Representing Strong Entity Sets

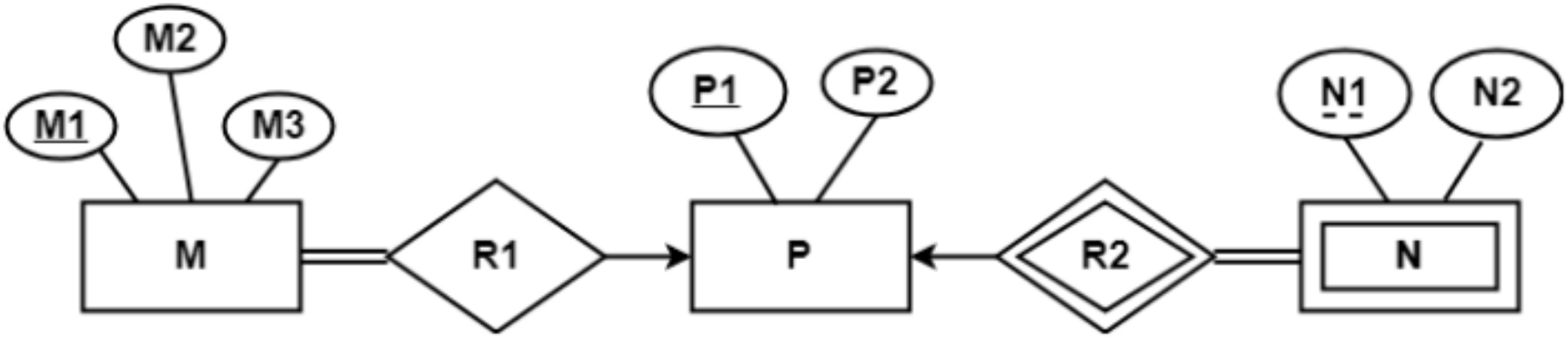
- A strong entity set reduces to a table with the same attributes.



- **M and P are strong entity sets**
- M (M1, M2, M3)
- P (P1, P2)

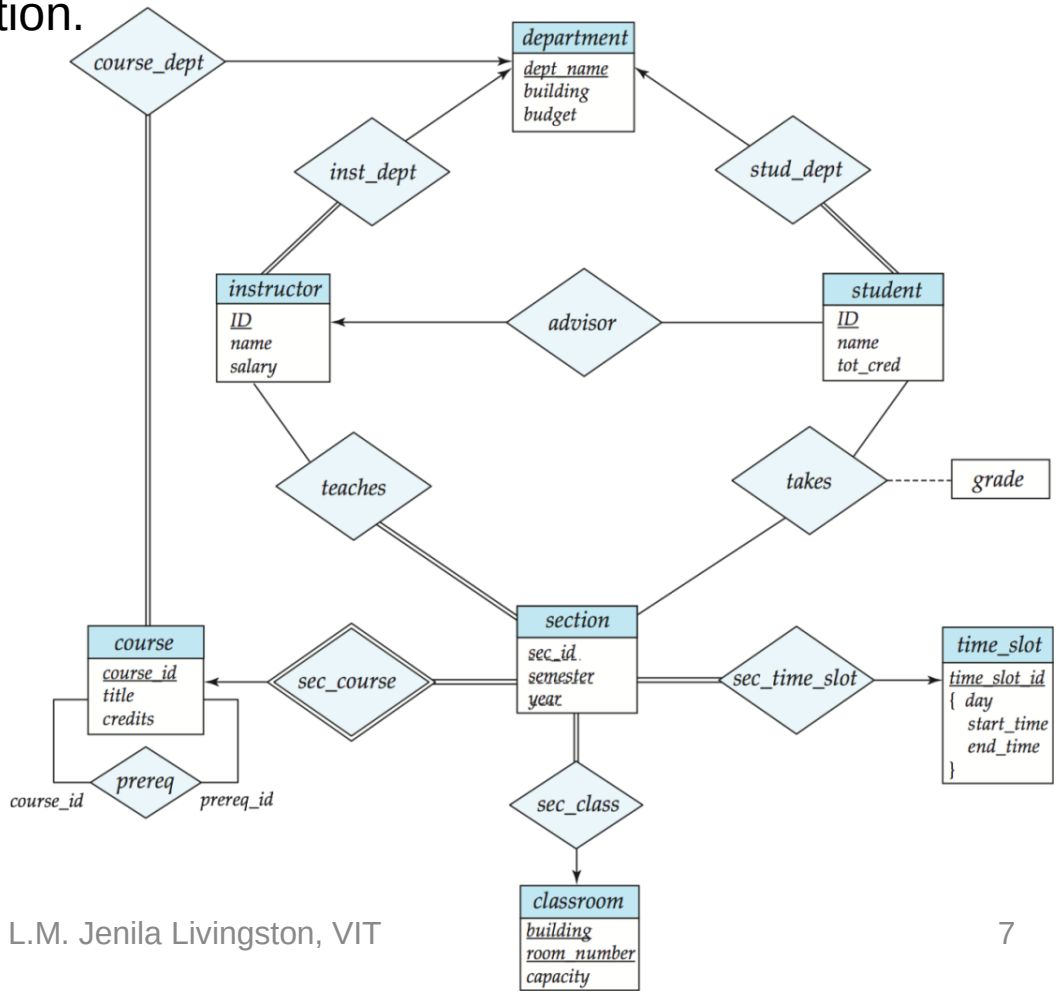
2. Representing Weak Entity Sets

- A weak entity set becomes a table that includes a column for the primary key of the identifying strong entity set.
- Weak entity identified by the double rectangle and corresponding relationship identified by the diamond
- N (N1, N2, P1)



Representing Weak Entity Sets

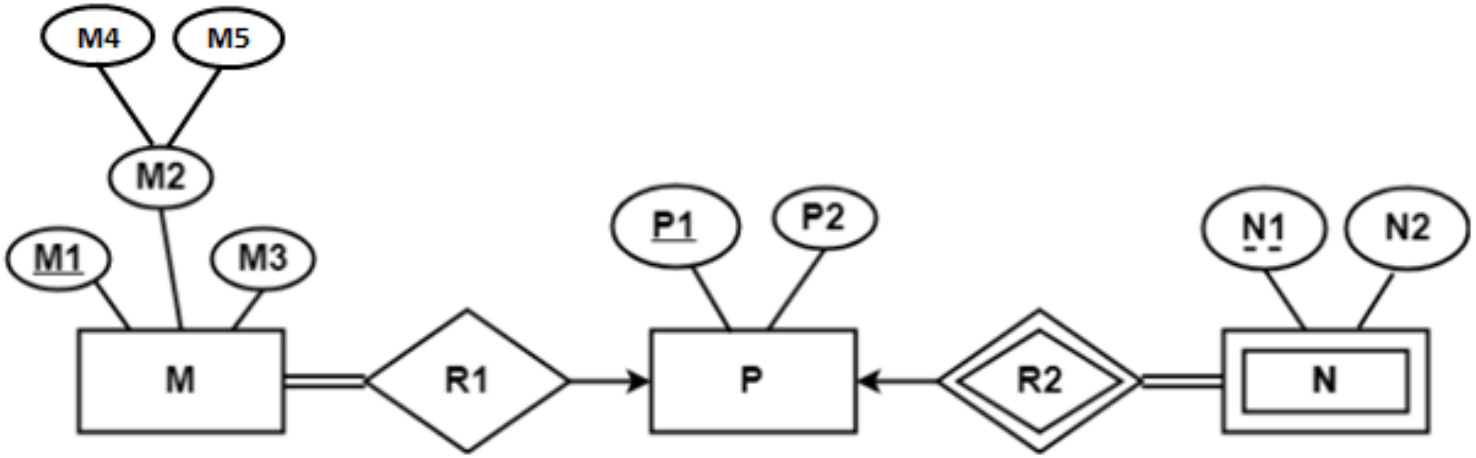
- In the above ER diagram Section is a weak entity set.
- It is identified by the strong entity set course id. So course_id to be included in the section relation.
- (identified by the double diamond relationship)



3. Composite Attributes

- **Composite attributes are broken up:**
  - Example: attribute *name* with components *first-name* and *last-name* becomes two attributes in the corresponding table – *name_first-name* and *name_last-name*.

- M (M1, M2_M4, M2_M5, M3)



4. Multi-valued Attributes

- A **multi-valued attribute** *M* of entity *E* is represented by a **new table** with the following attributes:
  - The primary key of *E*
  - An attribute corresponding to multi-valued attribute *M*

- Example:

  Entity Set:

  employee with attributes *id#, name, phone#*

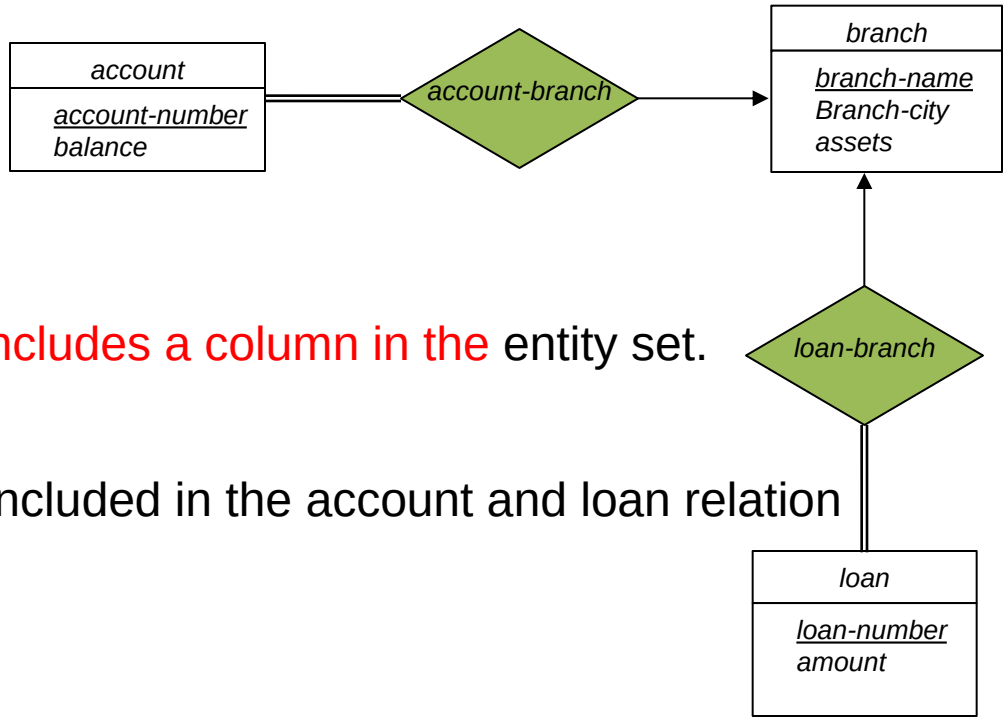
  Tables:

  employee (id#, name)

  dependent (id#, phone)

5. Representing Total Participation

- If every entity in an entity set must participate in a relationship set, then that entity set is said to have *total participation* in the relationship; indicated by a double-line.

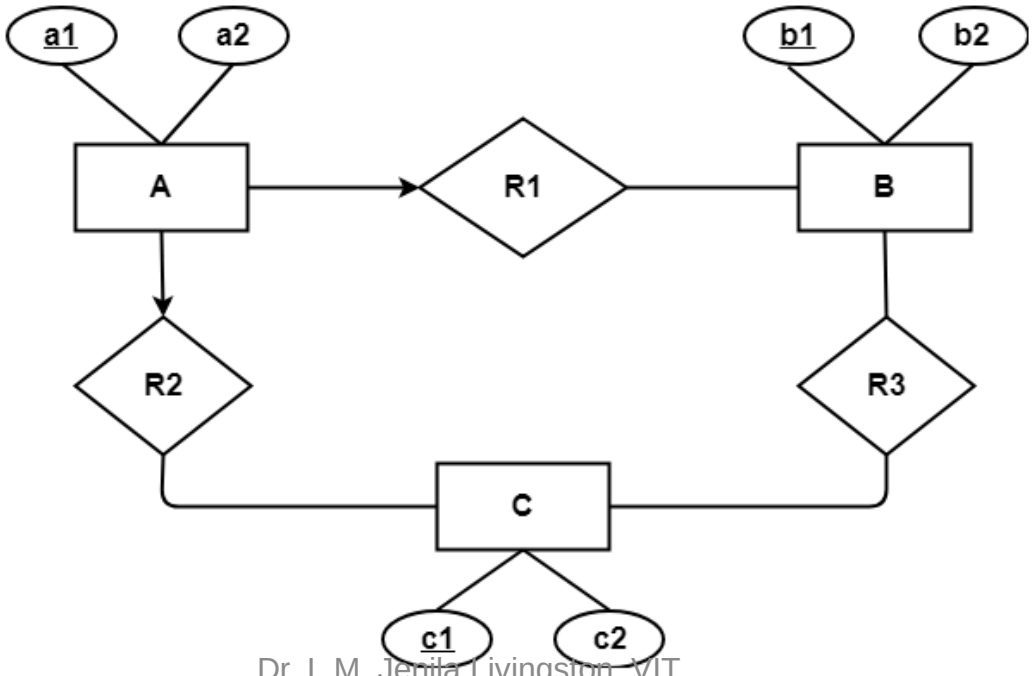


- A total participation includes a column in the entity set.
- branch_name to be included in the account and loan relation

6. Representing Relationship Sets

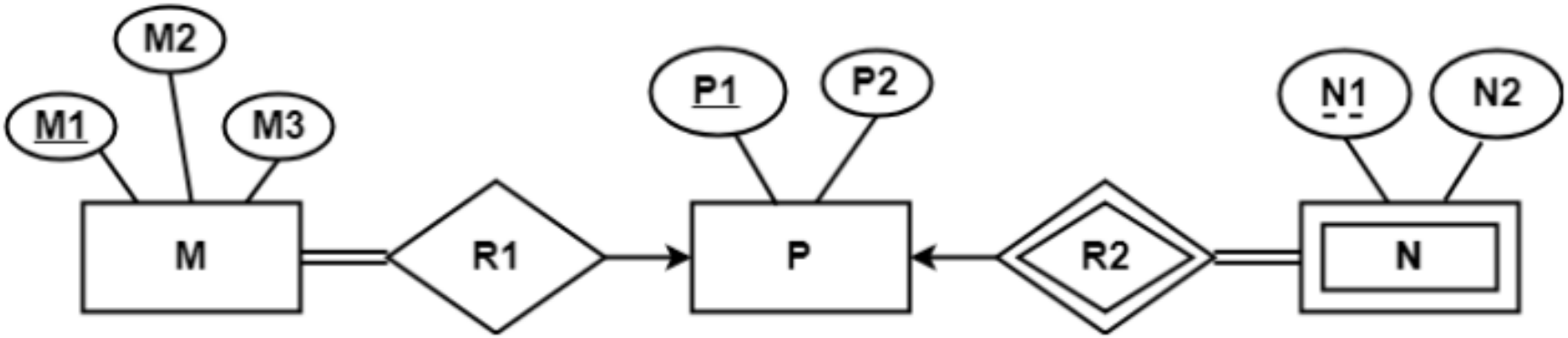
- A many-to-many relationship set is represented as a table with columns for the primary keys of the two participating entity sets, and any descriptive attributes of the relationship set.

- Example: R3 (b1, c1)



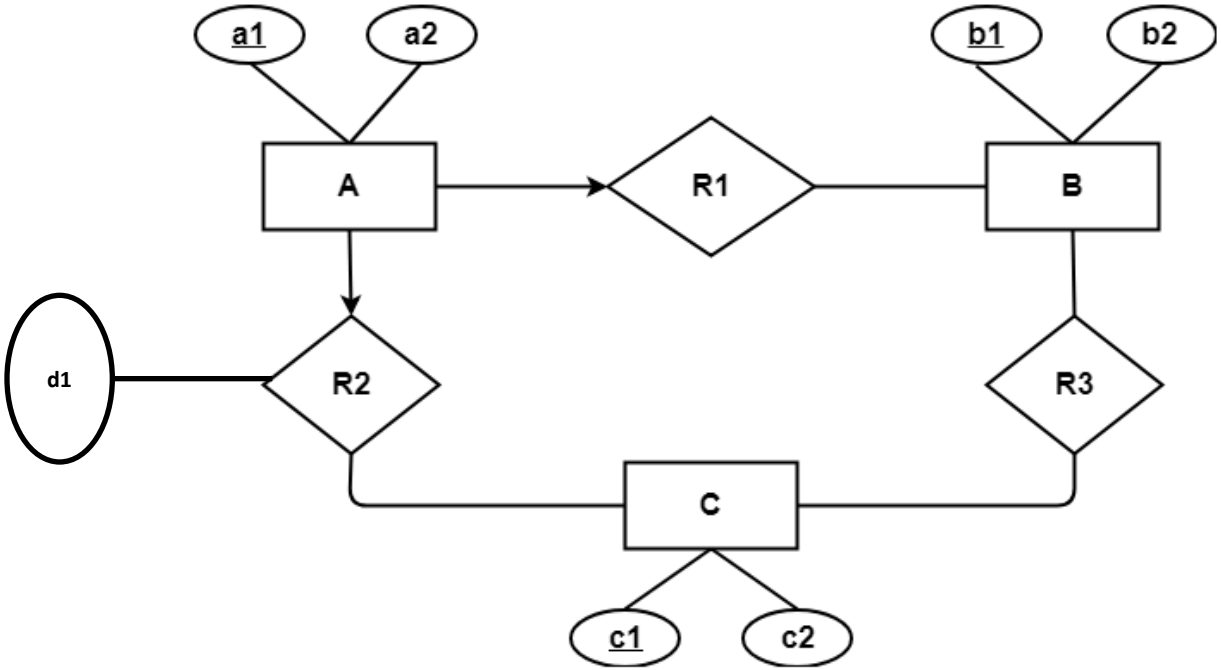
Representing Relationship Sets cont.

- A many-to-one relationship set can be represented just like a many-to-many relationship.
- Technically this is not necessary, and in some cases it does not result in a good design.
- If a relationship contains its own attribute then form a relation



Representing Relationship Sets

- In many-to-one relationship, if a relationship contains its own attribute then form a relation
- R2(d1, a1, c1)



Representing Relationship Sets, cont.

- For one-to-one relationship sets, the **extra attribute can be added to either of the tables** corresponding to the two entity sets.

- If no attribute, no need to form a table

7. Representing Specialization

- Note: This discussion assumes a 2-level inheritance hierarchy.
  - Exercise: Generalize it to an arbitrarily deep hierarchy.

- Method 1:
  - Form a table for the higher level entity set.
  - Form a table for each lower level entity set, including the primary key of the higher level entity set and local attributes.

table	attributes
person	*name, street, city*
customer	*name, credit-rating*
employee	*name, salary*

  - One Drawback: getting information about specific entities requires accessing two tables

Representing Specialization

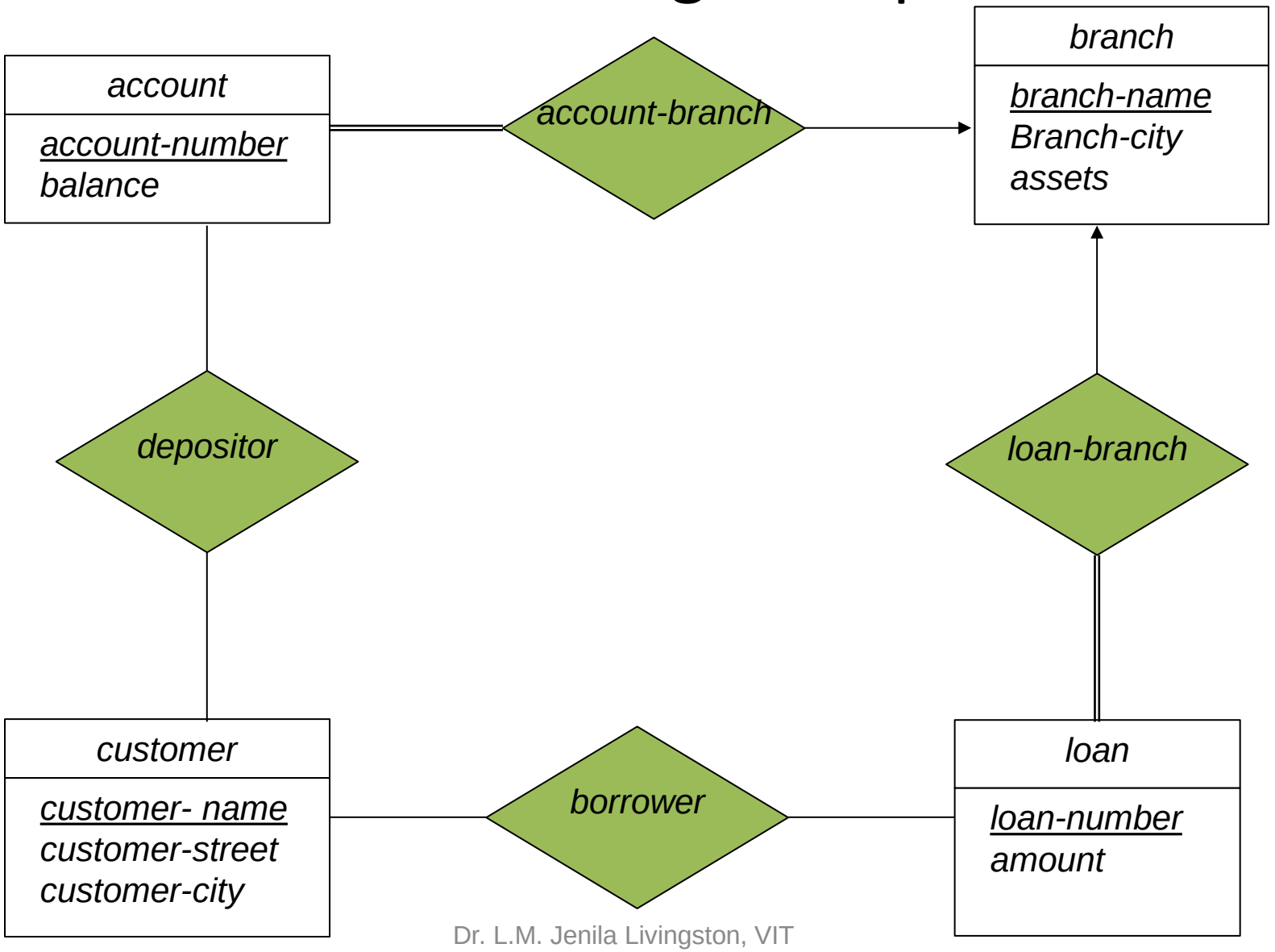
- Method 2:
  - Form a table for each entity set with all local and inherited attributes

table	table attributes
person	*name, street, city*
customer	*name, street, city, credit-rating*
employee	*name, street, city, salary*

- This method has obvious redundancies.
  - Particularly bad for persons who are both customers and employees.

- If specialization is total, the table for the generalized entity is redundant.
  - Temptation is to delete the *person* table; still might be needed for foreign key constraints.

E-R Diagram for the Banking Enterprise



Relational Schemes for the Banking Enterprise

- The following relational schemes result:

branch (branch-name, branch-city, assets)

customer (customer-name, customer-street, customer-city)

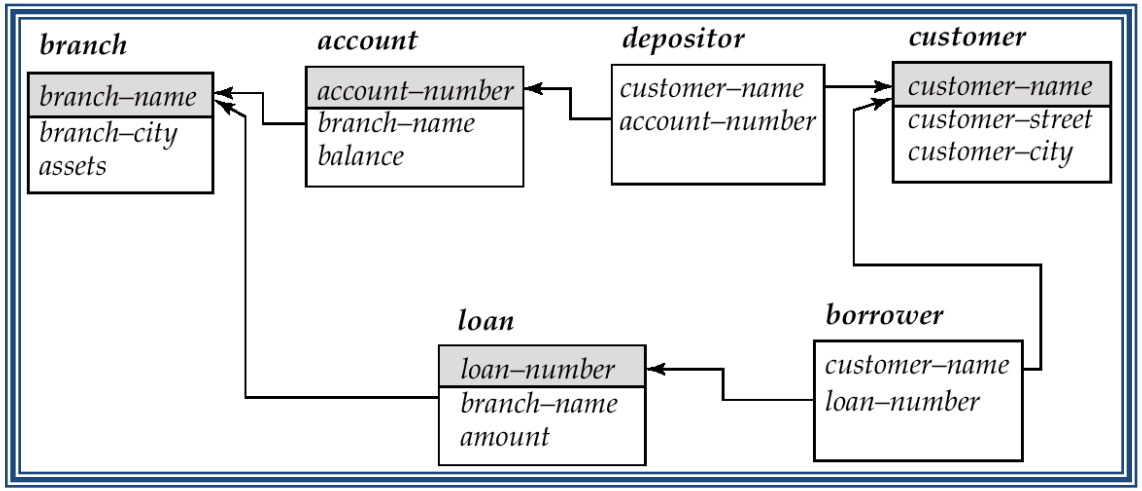
account (account-number, branch-name, balance)

loan (loan-number, branch-name, amount)

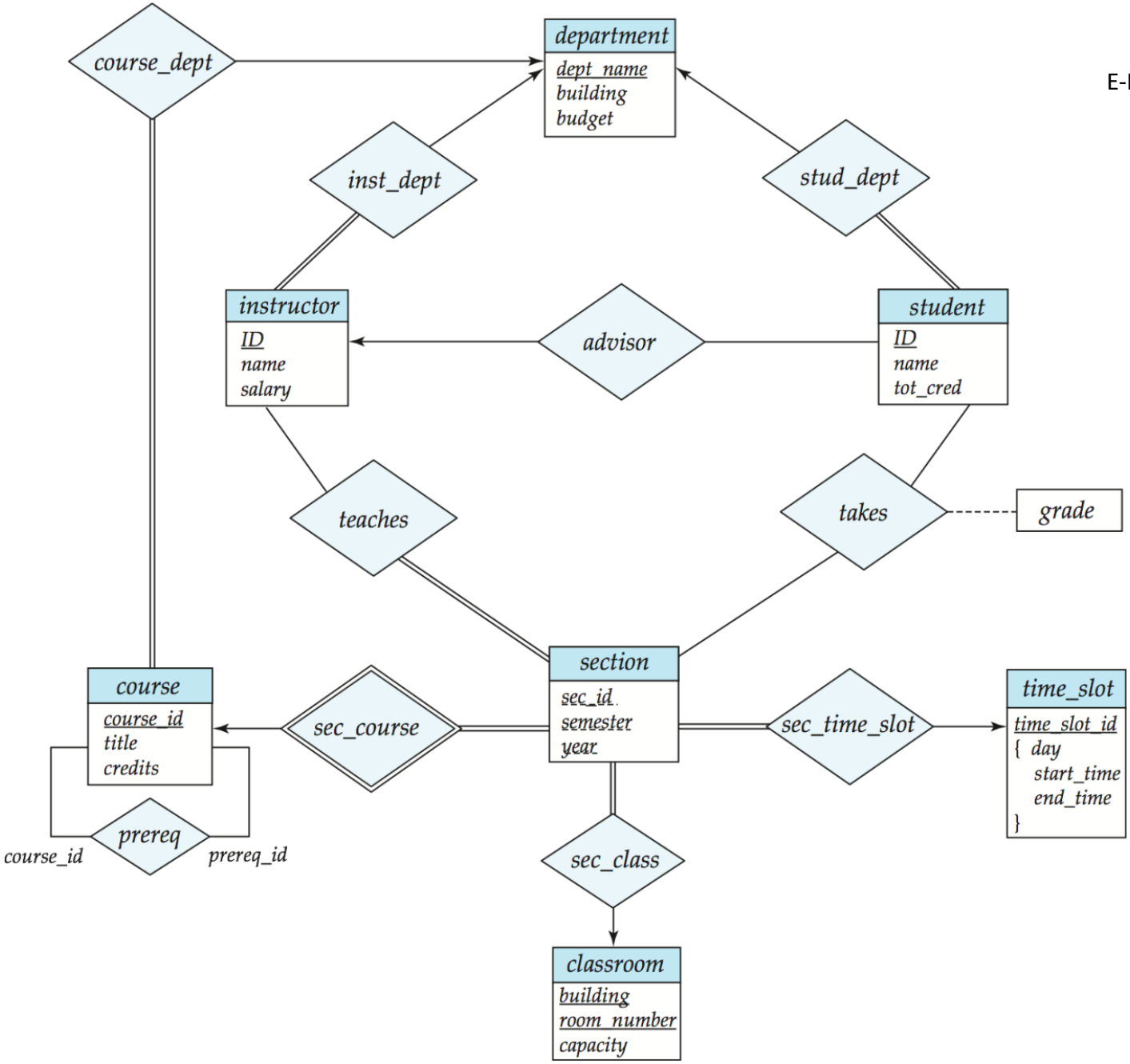
depositor (customer-name, account-number)

borrower (customer-name, loan-number)

Schema Diagram for the Banking Enterprise



E-R Diagram for a University



Relational Schema for a University

Classroom (building, room-number, capacity)

Department (dept-name, building, budget)

Course (course-id, title, dept-name, credits)

Instructor (ID, name, depart-name, salary)

Section (course-id, sec-id, semester, year, building, room-number, time-slot-id)

Teaches (ID, course-id, sec-id, semester, year)

Student (ID, name, dept-name, tot-cred)

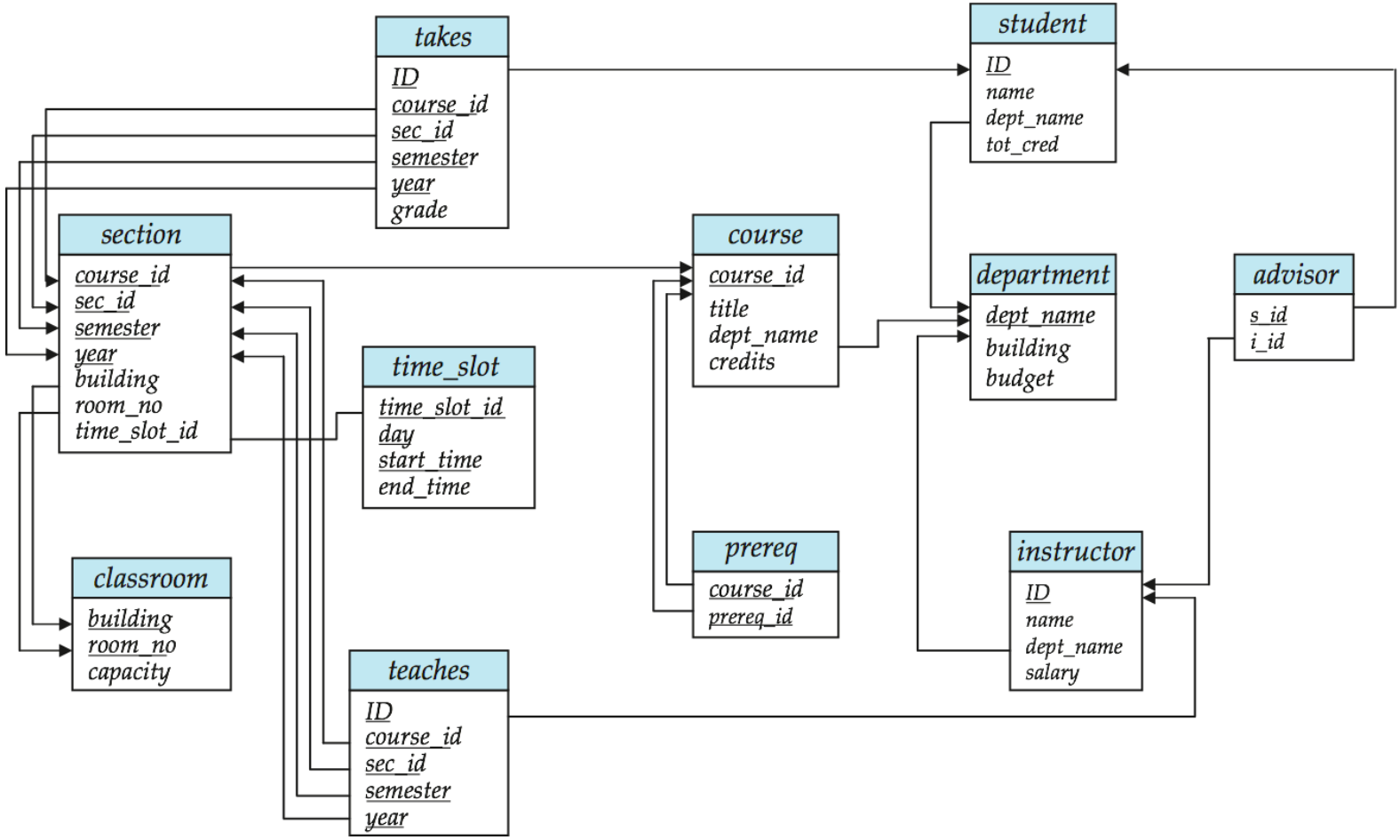
Takes (ID, course-id, sec-id, semester, year, grade)

Advisor (s-ID, i-ID)

Time-slot (time-slot-id, day, start-time, end-time)

Prereq (course-id, prereq-id)

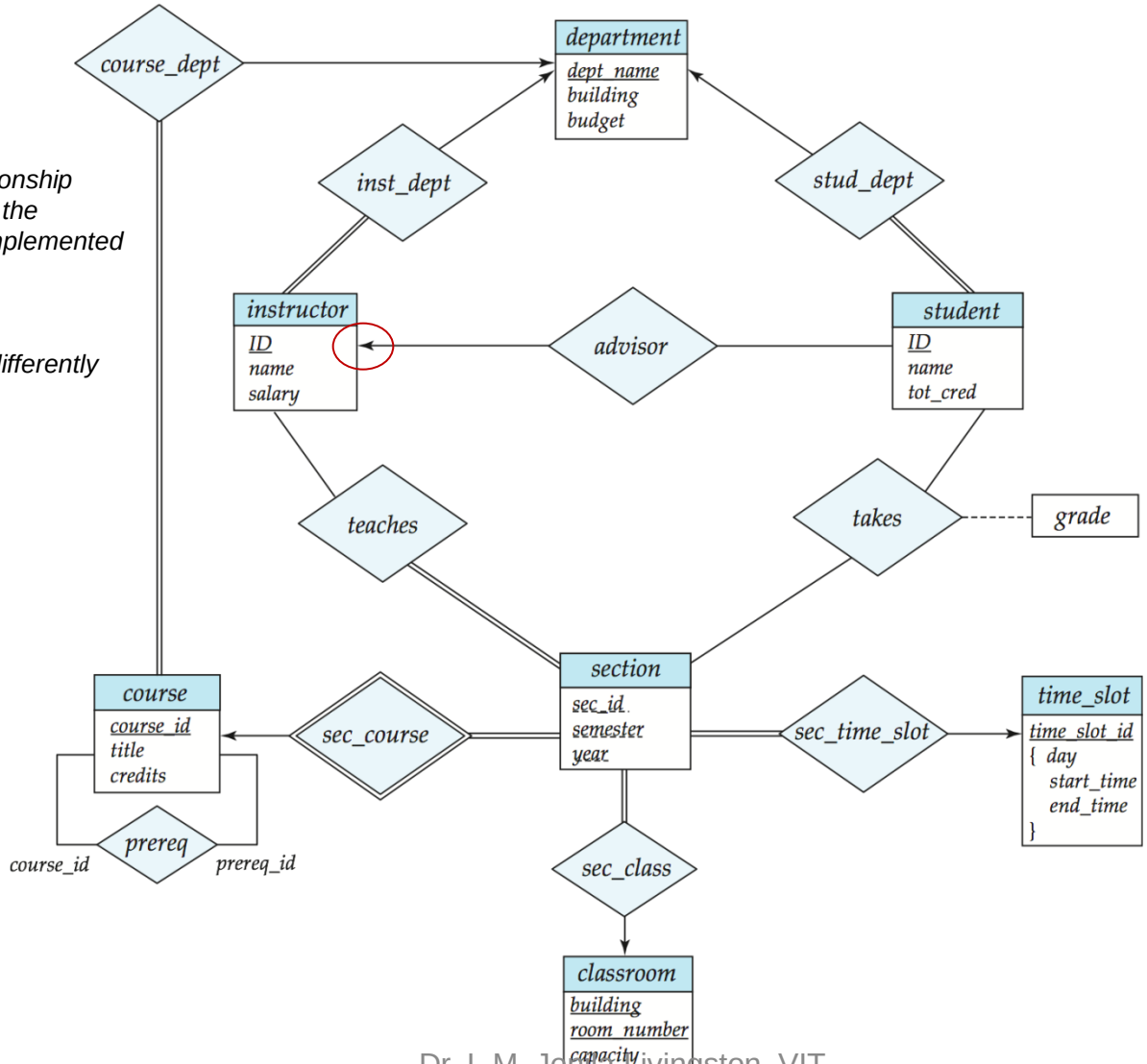
Schema Diagram for a University



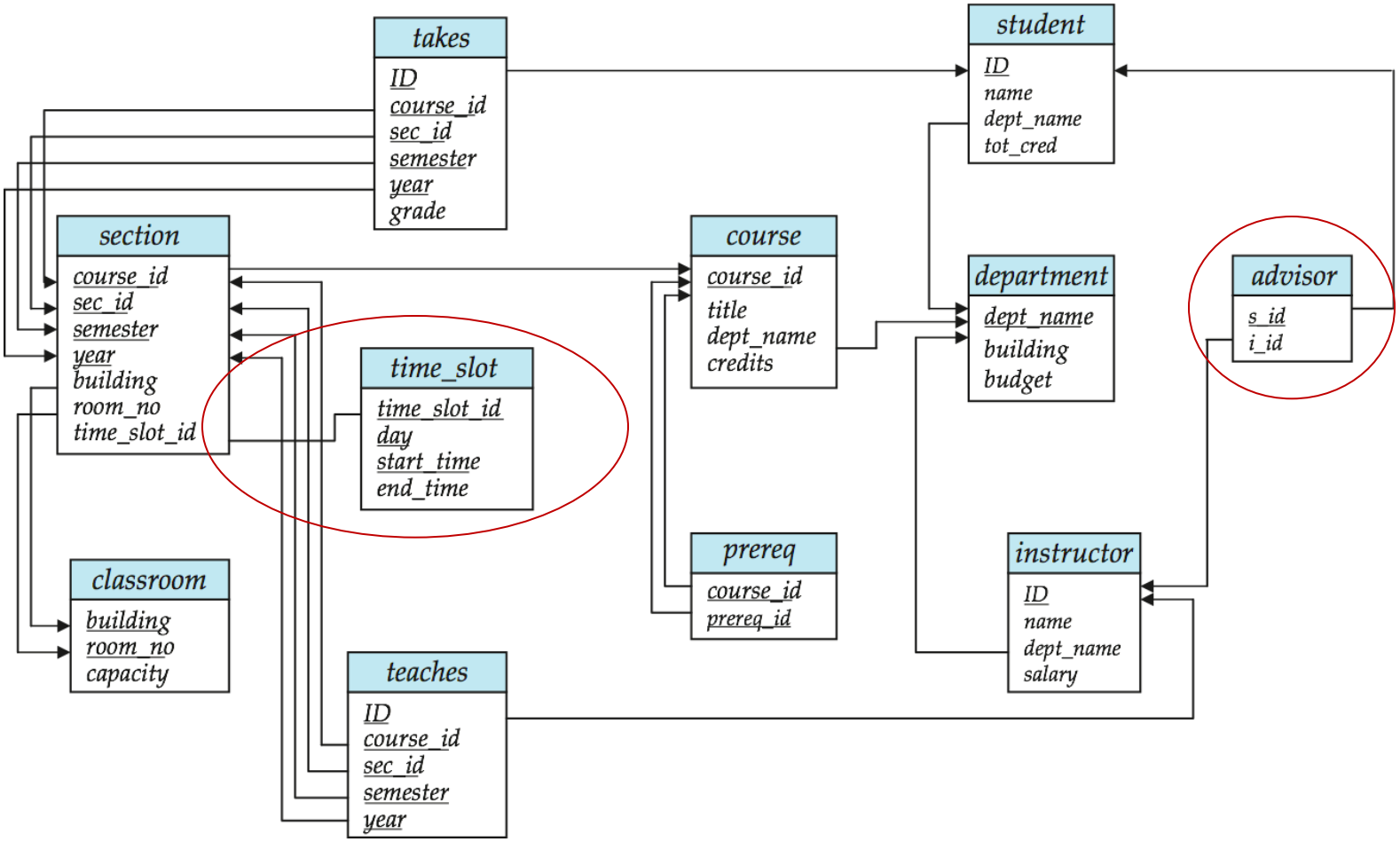
E-R Diagram for a University

Note the advisor relationship Is one-to-many, but in the schema it has been implemented as its' own table..

Similarly, time—slot is implemented slightly differently than described…



Schema Diagram for a University



Thank You