RAMAIAH
Institute of Technology

M.S. Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)
Department of Computer Science and Engineering

Course Name: Database Systems

Course Code: CS52

Credits: 3:1:0

UNIT 1

Term: October 2021– February 2022

Reference:
Elmasri, R., Shamkant B. Navathe, R.
Fundamentals of Database Systems

RAMAIAH
Institute of Technology

Unit 2 – Chapter 7

- Relational Model and Relational Algebra:
- Relational Model Concepts, Relational Model Concepts
- Relational Model Constraints and Relational Database Schema
- Update Operations, Transactions and Dealing with Constraint violations
- Unary Relational operations
- Relational Algebra Operations from Set Theory
- Binary Relational Operations, JOIN and DIVISION
- Additional Relational Operations,
- Examples of Queries in Relational Algebra
- Relational Database Design Using ER- to-Relational Mapping.

RAMAIAH
Institute of Technology

Chapter Outline

- **ER-to-Relational Mapping Algorithm**
  - Step 1: Mapping of Regular Entity Types
  - Step 2: Mapping of Weak Entity Types
  - Step 3: Mapping of Binary 1:1 Relation Types
  - Step 4: Mapping of Binary 1:N Relationship Types.
  - Step 5: Mapping of Binary M:N Relationship Types.
  - Step 6: Mapping of Multivalued attributes.
  - Step 7: Mapping of N-ary Relationship Types.

GOALS during Mapping

- Preserve all information (that includes all attributes)
- Maintain the constraints to the extent possible (Relational Model cannot preserve all constraints- e.g., max cardinality ratio such as 1:10 in ER; exhaustive classification into subtypes, e.g., STUDENTS are specialized into Domestic and Foreign)
- Minimize null values

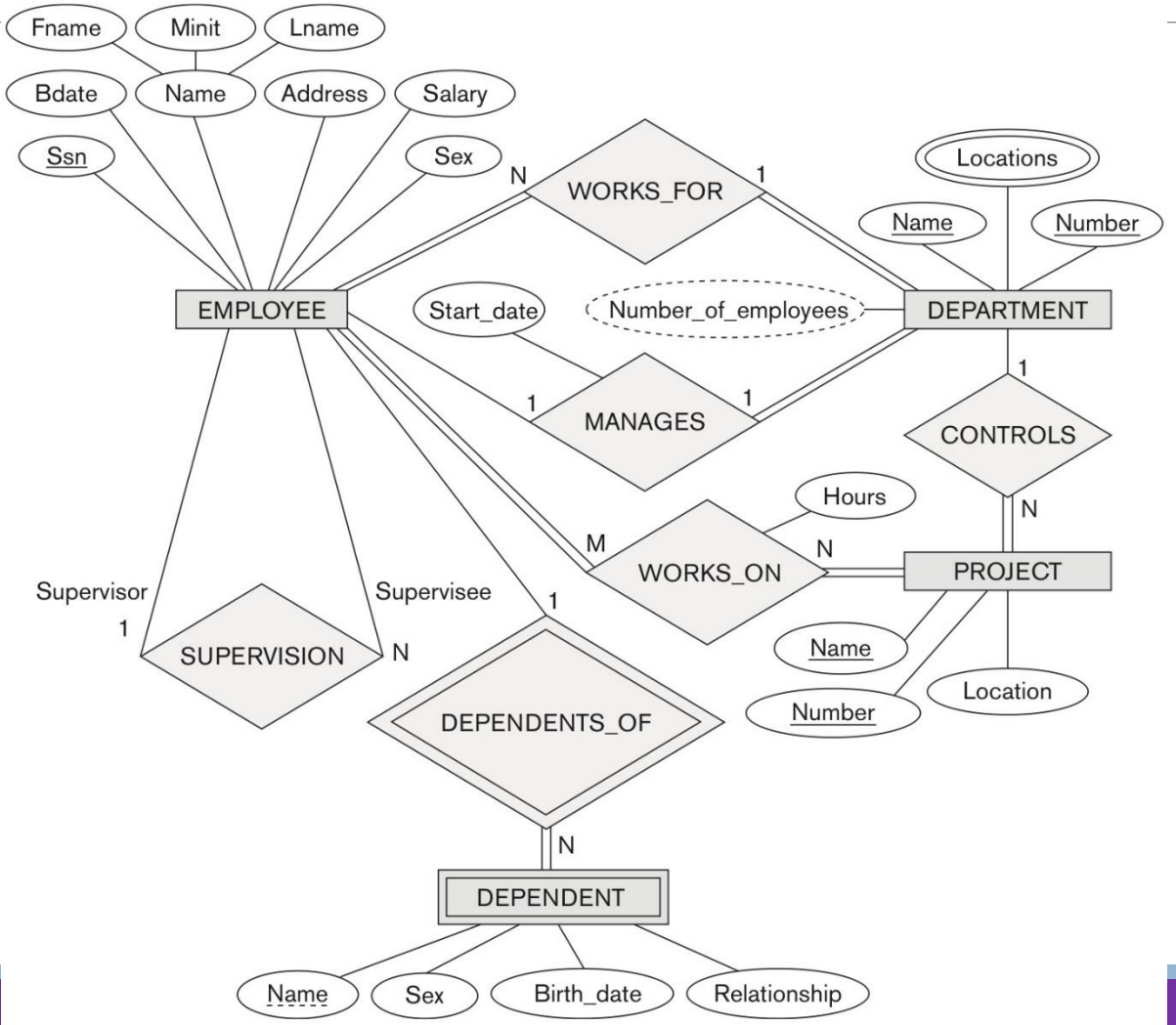
RAMAIAH
Institute of Technology

ER-to-Relational Mapping Algorithm

- **Step 1: Mapping of Regular Entity Types.**
  - For each regular (strong) entity type E in the ER schema, create a relation R that includes **all the simple attributes of E.**
  - Choose one of the key attributes of E as the **primary key** for R.
  - If the chosen key of E is **composite**, the **set of simple attributes that form it will together form the primary key** of R.
- Example: We create the relations EMPLOYEE, DEPARTMENT, and PROJECT in the relational schema corresponding to the regular entities in the ER diagram.
  - SSN, DNUMBER, and PNUMBER are the primary keys for the relations EMPLOYEE, DEPARTMENT, and PROJECT as shown.

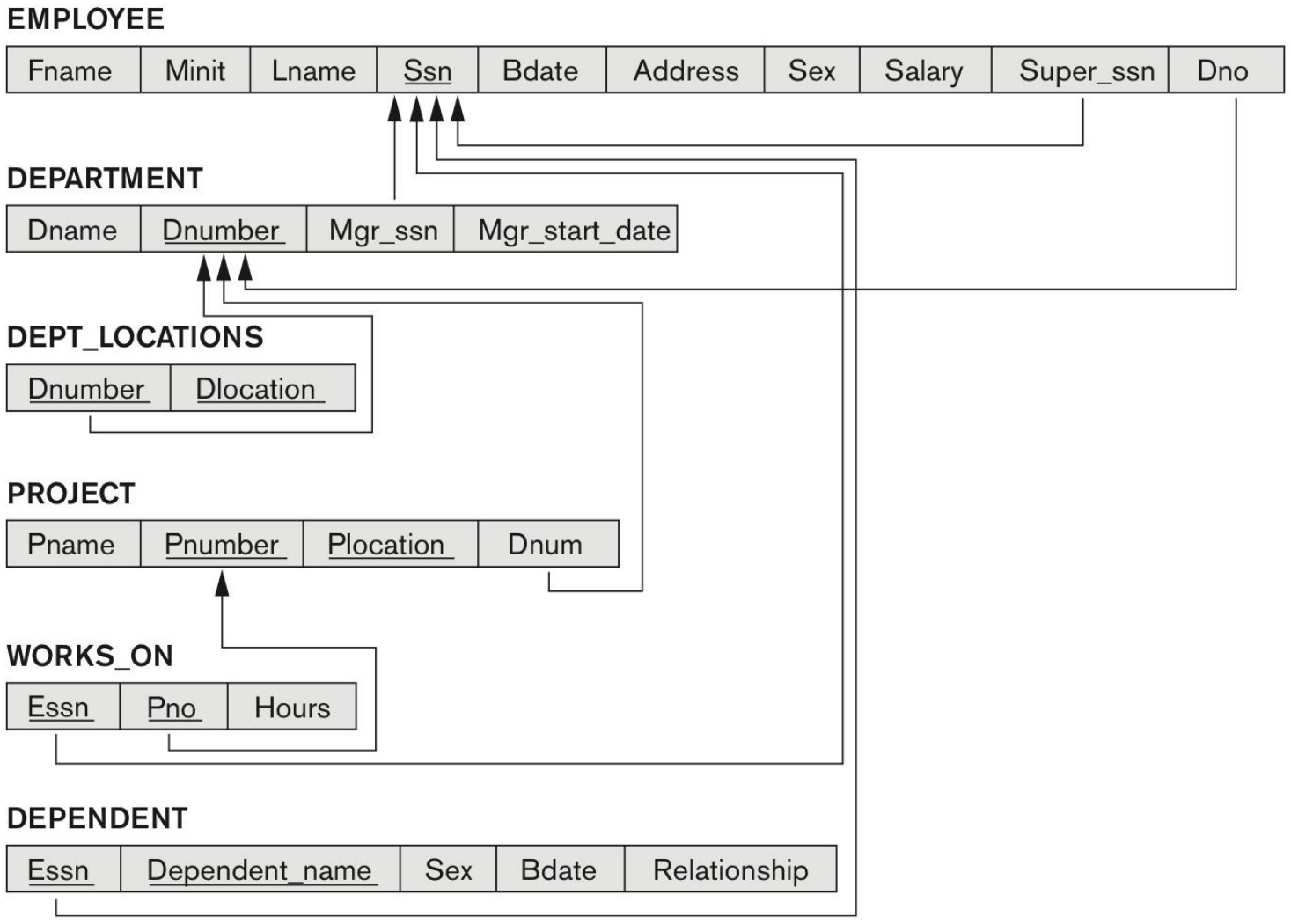
RAMAIAH
Institute of Technology

The ER conceptual schema diagram for the COMPANY database.



RAMAIAH Institute of Technology

Result of mapping the COMPANY ER schema into a relational database schema.



RAMAIAH Institute of Technology

ER-to-Relational Mapping Algorithm

- **Step 2: Mapping of Weak Entity Types**
  - For each weak entity type W in the ER schema with owner entity type E, **create a relation R & include all simple attributes** (or simple components of composite attributes) of W as attributes of R.
  - Also, **include as foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s)**.
  - The **primary key of R is the *combination of* the primary key(s) of the owner(s) and the partial key of the weak entity type** W, if any.
- **Example:** Create the relation DEPENDENT in this step to correspond to the weak entity type DEPENDENT.
  - Include the primary key SSN of the EMPLOYEE relation as a foreign key attribute of DEPENDENT (renamed to ESSN).
  - The primary key of the DEPENDENT relation is the combination {ESSN, DEPENDENT_NAME} because DEPENDENT_NAME is the partial key of DEPENDENT.

ER-to-Relational Mapping Algorithm (contd.)

- **Step 3: Mapping of Binary 1:1 Relation Types**
  - For each binary 1:1 relationship type R in the ER schema, **identify the relations S and T** that correspond to the entity types participating in R.

- There are three possible approaches:
  1. **Foreign Key (2 relations) approach:** Choose one of the relations-say S-and **include a foreign key in S the primary key of T**. It is better to choose an entity type with total participation in R in the role of S.
     - Example: 1:1 relation MANAGES is mapped by choosing the participating entity type DEPARTMENT to serve in the role of S, because its participation in the MANAGES relationship type is total.
  2. **Merged relation (1 relation) option:** An alternate mapping of a 1:1 relationship type is possible by merging the two entity types and the relationship into a single relation. This may be appropriate when both participations are total.
  3. **Cross-reference or relationship relation (3 relations) option:** The third alternative is to set up a third relation R for the purpose of cross-referencing the primary keys of the two relations S and T representing the entity types.

ER-to-Relational Mapping Algorithm (contd.)

- **Step 4: Mapping of Binary 1:N Relationship Types.**
  - For each regular binary 1:N relationship type R, **identify the relation S** that represent the participating entity type at the **N-side of the relationship** type.
  - **Include as foreign key in S the primary key of the relation T** that represents the other entity type participating in R.
  - Include **any simple attributes of the 1:N relation type as attributes of S.**
- Example: 1:N relationship types WORKS_FOR, CONTROLS, and SUPERVISION in the figure.
  - For WORKS_FOR we include the primary key DNUMBER of the DEPARTMENT relation as foreign key in the EMPLOYEE relation and call it DNO.
- An alternative approach is to use a Relationship relation (cross referencing relation) – this is rarely done.

RAMAIAH
Institute of Technology

ER-to-Relational Mapping Algorithm (contd.)

- **Step 5: Mapping of Binary M:N Relationship Types.**
  - For each regular binary M:N relationship type R, ***create a new relation*** **S** to represent R. This is a ***relationship relation.***
  - Include as **foreign key attributes in S the primary keys of the relations that represent the participating entity types**; *their combination will form the primary key* of S.
  - Also include any **simple attributes of the M:N relationship type** (or simple components of composite attributes) as attributes of S.
- Example: The M:N relationship type WORKS_ON from the ER diagram is mapped by creating a relation WORKS_ON in the relational database schema.
  - The primary keys of the PROJECT and EMPLOYEE relations are included as foreign keys in WORKS_ON and renamed PNO and ESSN, respectively.
  - Attribute HOURS in WORKS_ON represents the HOURS attribute of the relation type. The primary key of the WORKS_ON relation is the combination of the foreign key attributes {ESSN, PNO}.

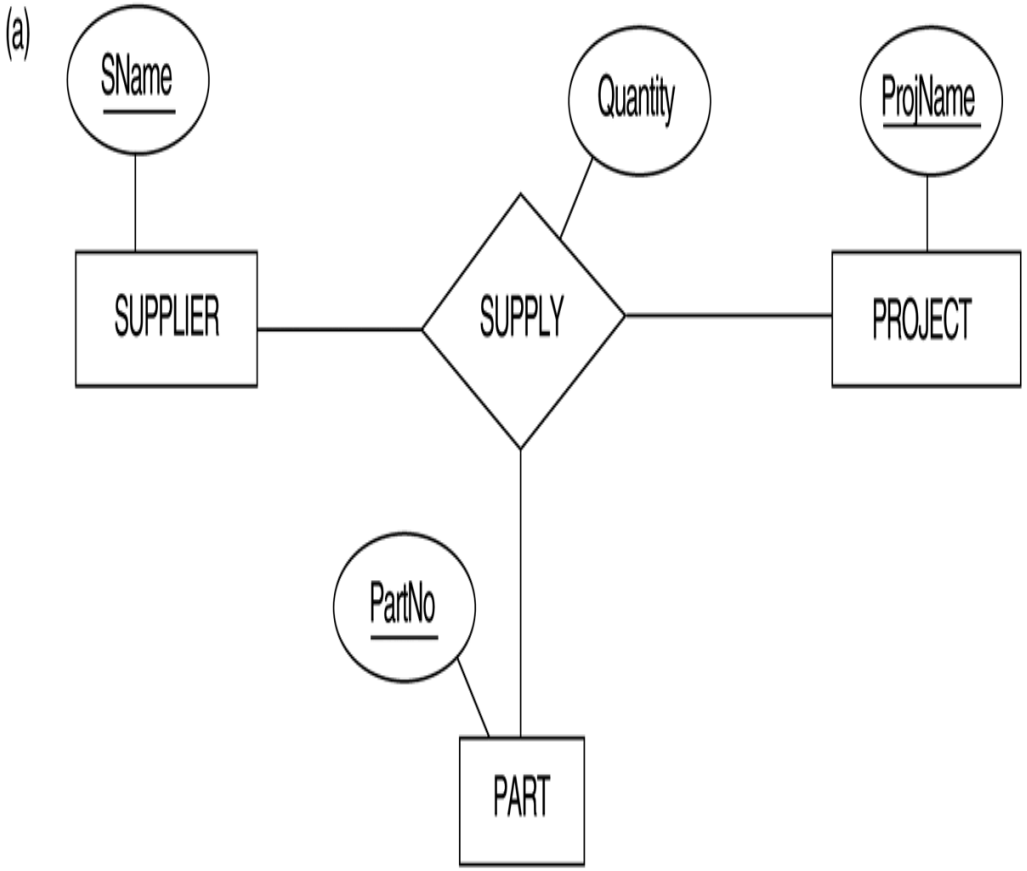
ER-to-Relational Mapping Algorithm (contd.)

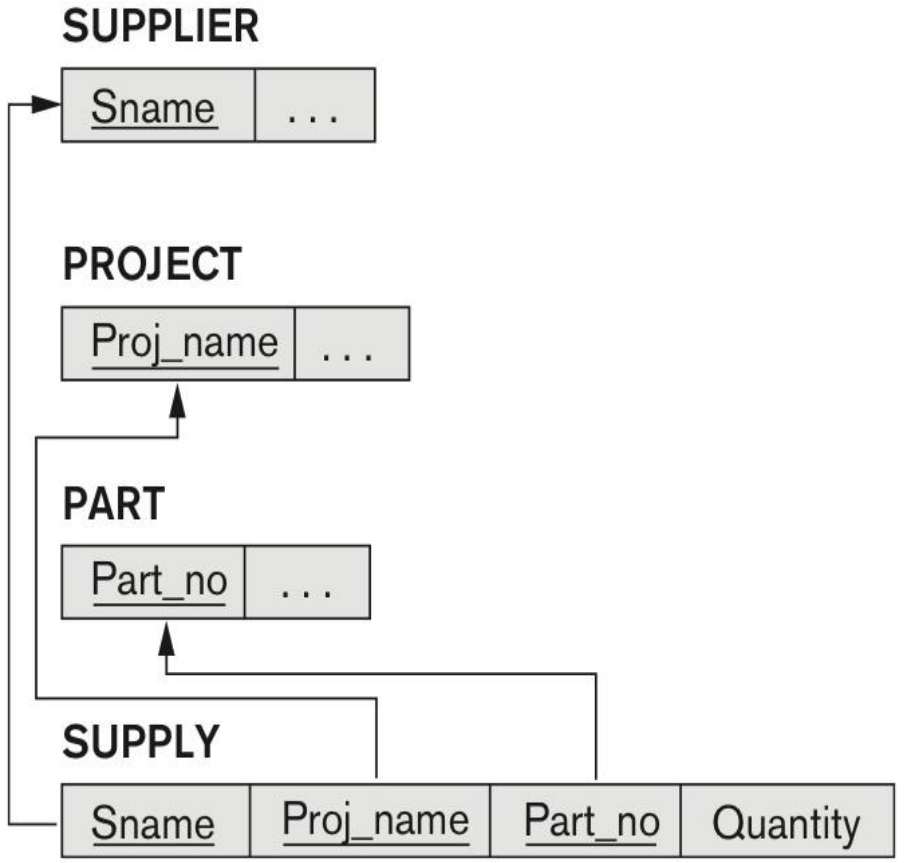
- **Step 6: Mapping of Multivalued attributes.**
  - **For each multivalued attribute A, create a new relation R.**
  - This relation R will include an **attribute** corresponding to A, plus the **primary key attribute K-as a foreign key** in R-of the relation that represents the entity type of relationship type that has A as an attribute.
  - The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.
- **Example:** The relation DEPT_LOCATIONS is created.
  - The attribute DLOCATION represents the multivalued attribute LOCATIONS of DEPARTMENT, while DNUMBER-as foreign key-represents the primary key of the DEPARTMENT relation.
  - The primary key of R is the combination of {DNUMBER, DLOCATION}.

ER-to-Relational Mapping Algorithm (contd.)

- **Step 7: Mapping of N-ary Relationship Types.**
  - For each n-ary relationship type R, where n>2, create **a new relationship** S to represent R.
  - Include as **foreign key attributes** in S the primary keys of the relations that represent the participating entity types.
  - Also **include any simple attributes of the n-ary relationship type** (or simple components of composite attributes) as attributes of S.
- **Example:** The relationship type SUPPLY in the ER on the next slide.
  - This can be mapped to the relation SUPPLY shown in the relational schema, whose primary key is the combination of the three foreign keys {SNAME, PARTNO, PROJNAME}

Mapping the *n*-ary relationship type SUPPLY

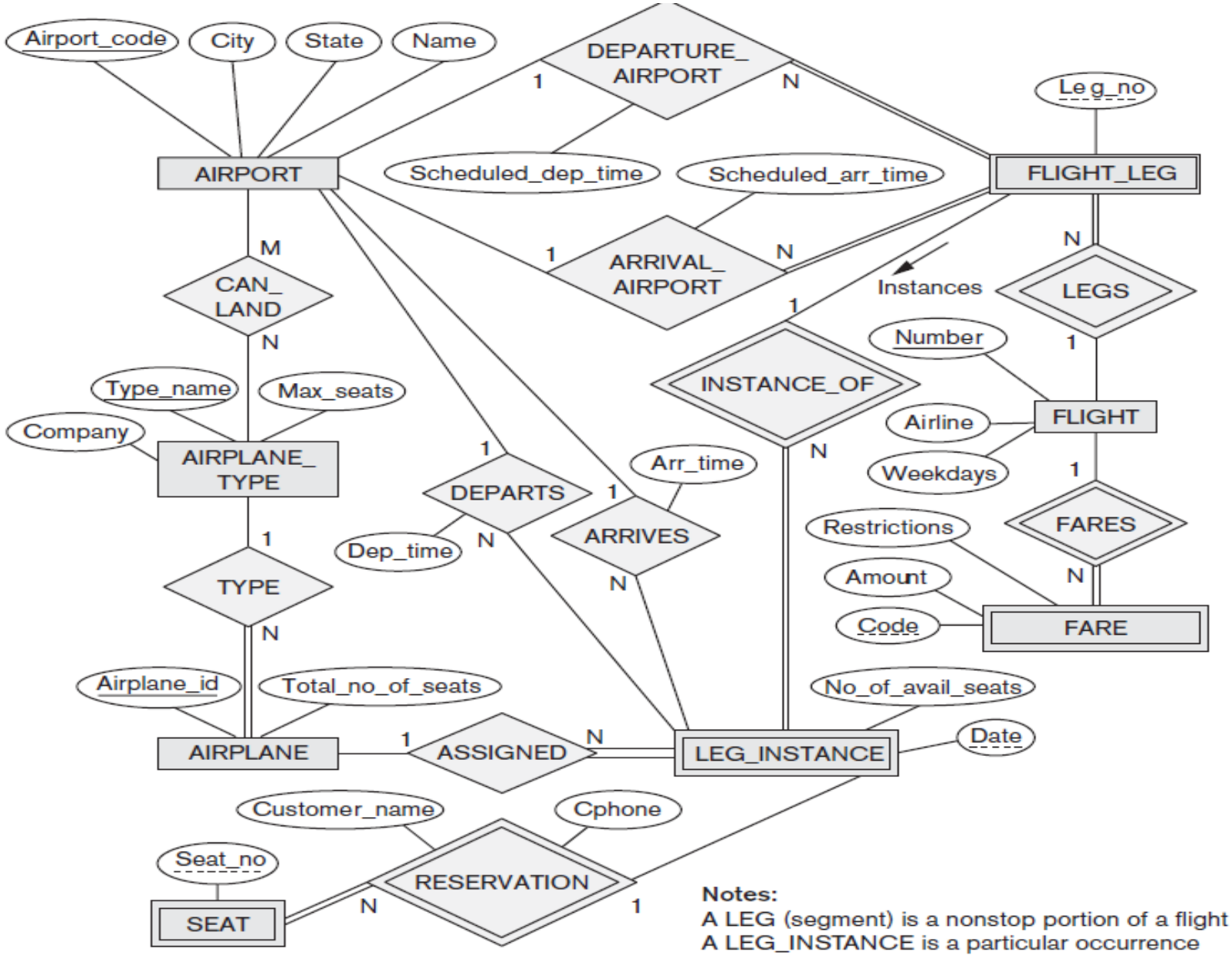




Summary of Mapping constructs and constraints

Table 9.1 Correspondence between ER and Relational Models

ER MODEL	RELATIONAL MODEL
Entity type	*Entity* relation
1:1 or 1:N relationship type	Foreign key (or *relationship* relation)
M:N relationship type	*Relationship* relation and *two* foreign keys
n-ary relationship type	*Relationship* relation and *n* foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple component attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary (or secondary) key


Airport_code
City
State
Name
AIRPORT
DEPARTURE_AIRPORT
ARRIVAL_AIRPORT
Scheduled_dep_time
Scheduled_arr_time
FLIGHT_LEG
Leg_no
Instances
LEGS
CAN_LAND
Type_name
Max_seats
Company
AIRPLANE_TYPE
TYPE
DEPARTS
Dep_time
Arr_time
ARRIVES
INSTANCE_OF
Number
FLIGHT
Airline
Weekdays
FARES
Restrictions
Amount
Code
FARE
Airplane_id
Total_no_of_seats
AIRPLANE
ASSIGNED
LEG_INSTANCE
No_of_avail_seats
Date
Customer_name
Cphone
Seat_no
RESERVATION
SEAT
Notes:
A LEG (segment) is a nonstop portion of a flight
A LEG_INSTANCE is a particular occurrence of a LEG on a particular date.

AIRPORT

Airport_code	Name	City	State

FLIGHT

Flight_number	Airline	Weekdays

FLIGHT_LEG

Flight_number	Leg_number	Departure_airport_code	Scheduled_departure_time
		Arrival_airport_code	Scheduled_arrival_time

LEG_INSTANCE

Flight_number	Leg_number	Date	Number_of_available_seats	Airplane_id
		Departure_airport_code	Departure_time	Arrival_airport_code	Arrival_time

FARE

Flight_number	Fare_code	Amount	Restrictions

AIRPLANE_TYPE

Airplane_type_name	Max_seats	Company

CAN_LAND

Airplane_type_name	Airport_code

AIRPLANE

Airplane_id	Total_number_of_seats	Airplane_type

SEAT_RESERVATION

Flight_number	Leg_number	Date	Seat_number	Customer_name	Customer_phone

END of Chapter 7