COURSE NAME: DBMS
COURSE CODE:23AD2102A

Topic: ER TO RELATIONAL MODEL

Session - 5

AIM OF THE SESSION

To familiarize students with the basic concept ER to Relational model

INSTRUCTIONAL OBJECTIVES

This Session is designed to: Characteristics of DBMS

LEARNING OUTCOMES

At the end of this session, you should be able to: basic knowledge of DBMS

ER TO RELATIONAL MODEL

An Entity-Relationship (ER) model to a Relational model involves transforming the entities, relationships, and attributes from the ER diagram into tables, keys, and relationships in a relational database.

converting an ER model to a Relational model:

1.Identify Entities: Each entity in the ER diagram becomes a table in the Relational model.

2.Identify Attributes: Each attribute of an entity becomes a column in the corresponding table.

3.Identify Relationships:

1. For one-to-one relationships, the primary key of one entity becomes a foreign key in the other entity.
2. For one-to-many and many-to-many relationshipsChoose a primary key for each table.

SESSION INTRODUCTION

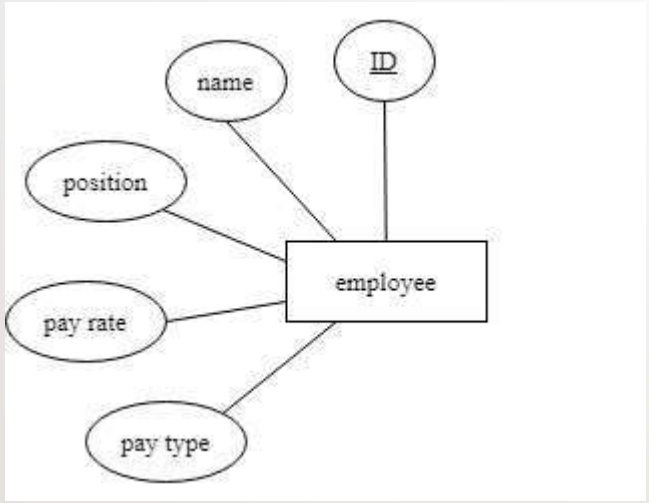
Entities

The first step in building a relational database from an ERD is creating a table from each entity in the data model. Weak entities need slightly different handling than regular entities, so we will address them separately, starting with regular entities.

Regular entities

- Most attributes for the entity should be converted to columns in the new table.
- Do not create columns for derived attributes, as these values are not intended to be stored.
- Do not create columns for multivalued attributes

Example: the entity employee might become a table named employee or employees. conversion of the employee entity into a relational table named employee

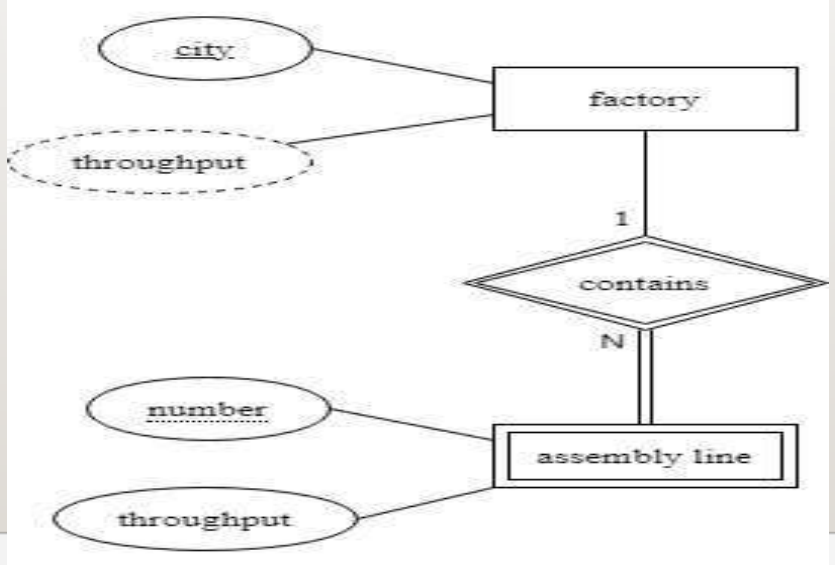


Column name	Type
id	integer
name	varchar
position	varchar
pay_rate	currency
pay_type	varchar
Keys Primary key: id	

Weak entities

Weak entities are converted into tables in nearly the same way as regular entities. However, recall that a weak entity has no identifying key attribute. Instead, it has a partial key, which must be combined with the key of the parent entity

The table created from a weak entity must therefore incorporate the key from the parent entity as an additional column. The primary key for the new table will be composed of the columns created from the parent key and



SESSION INTRODUCTION

Column name	Type
city	Varchar
Keys Primary key: city	

Column name	Type
factory_city	Varchar
number	Integer
Keys Primary key: factory_city, number Foreign key: factory_city → factory (city)	

Relationships

- Relationships can be handled using a few different approaches, depending on the cardinality ratio of the relationship.

A table to represent the relationship.

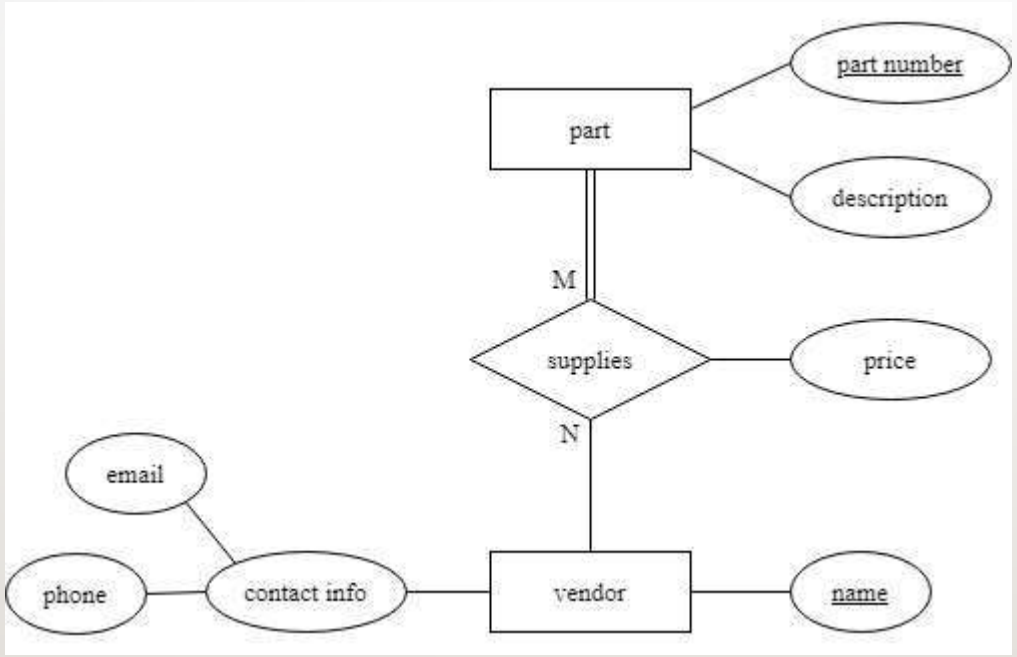
This kind of table is known as a cross-reference table, and acts as an intermediary in a three-way join with the two (or more) tables whose entities participate in the relationship.

Many-to-many

Many-to-many relationships are the most general type of relationship; a database structure accommodating a many-to-many relationship can also accommodate one-to-many or one-to-one relationships, as “one” is just a special case of “many”.

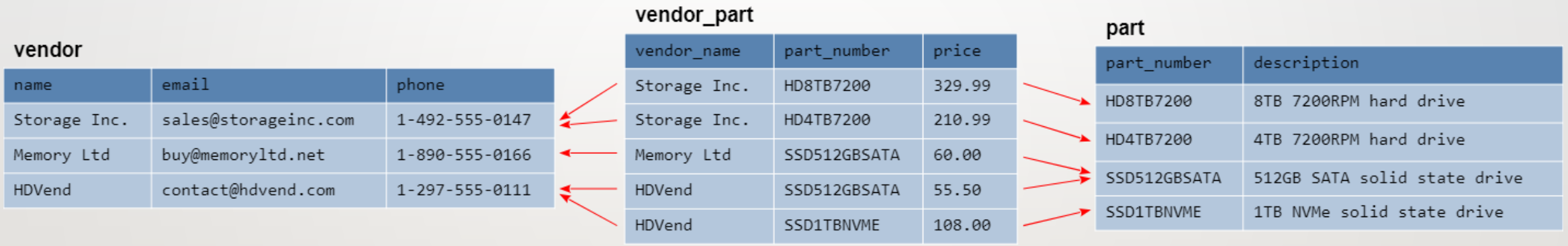
SESSION INTRODUCTION

ERD indicates a many-to-many relationship between the entities vendor and part. A computer part (such as an 8TB hard drive) can come from multiple sellers, while sellers can sell multiple different computer parts:



SESSION INTRODUCTION

- Create the cross-reference table vendor_part.
- It is common to name a cross-reference table using the names of the two tables being related, although other schemes can of course be used.



vendor

name	email	phone
Storage Inc.	sales@storageinc.com	1-492-555-0147
Memory Ltd	buy@memoryltd.net	1-890-555-0166
HDVend	contact@hdvend.com	1-297-555-0111

vendor_part

vendor_name	part_number	price
Storage Inc.	HD8TB7200	329.99
Storage Inc.	HD4TB7200	210.99
Memory Ltd	SSD512GBSATA	60.00
HDVend	SSD512GBSATA	55.50
HDVend	SSD1TBNVME	108.00

part

part_number	description
HD8TB7200	8TB 7200RPM hard drive
HD4TB7200	4TB 7200RPM hard drive
SSD512GBSATA	512GB SATA solid state drive
SSD1TBNVME	1TB NVMe solid state drive

SESSION INTRODUCTION

Table descriptions for vendor, part, and
the vendor_part cross-reference table are given below:

Table vendor

Column name	Type
name	char
phone	Integer
email	char
Keys Primary key: name	

Table part

Column name	Type
part_number	char
description	char
Keys Primary key: part_number	

Column name	Type
vendor_name	char
part_number	Integer
price	char
Keys Primary key: vendor_name, part_number Foreign key: vendor_name → vendor (name) Foreign key: part_number → part (part_number)	

One-to-many

- one-to-many relationships can be implemented in the database using a cross-reference table
- The employee entity participates in one-to-many relationships with both factory and itself

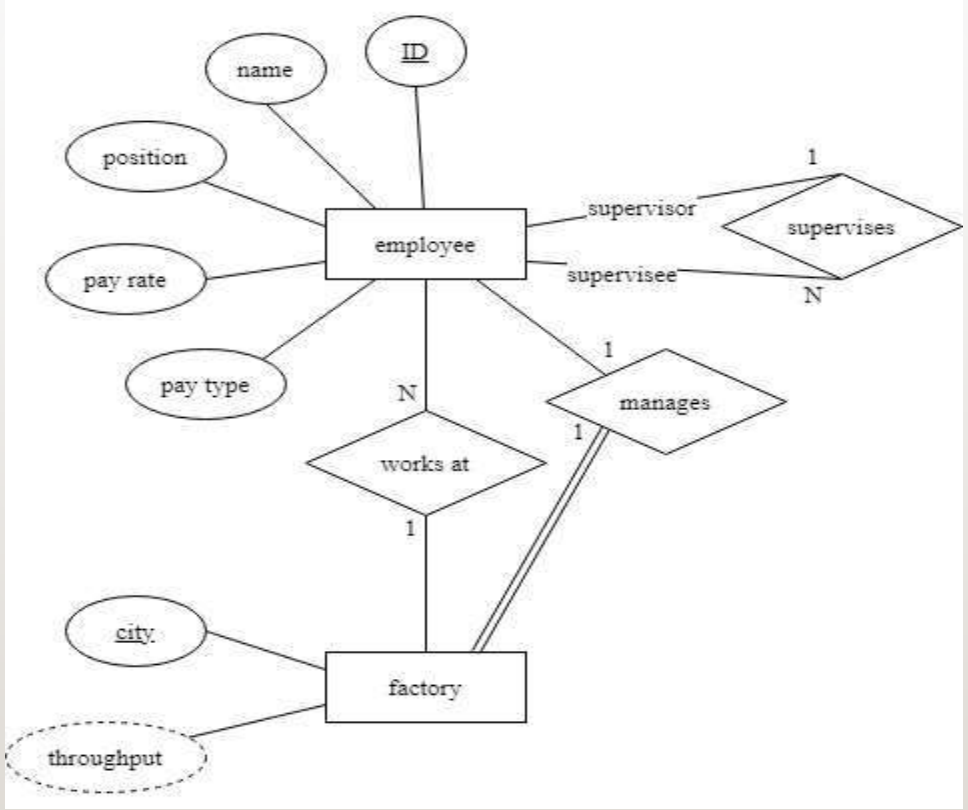


Table employee

Column name	Type
name	char
position	char
pay_rate	currency
pay_type	char
factory_city	char
supervisor_id	int
Keys Primary key: id Foreign key: factory_city → factory (city) Foreign key: supervisor_id → employee (id)	

One-to-one

A one-to-one relationship, manages, between employee and factory.

Table factory

Column name	Type
city	char
manager_id	int
Keys Primary key: city Foreign key: manager_id → employee (id)	

Multivalued attributes

- A multivalued attribute is used when a list of arbitrary values needs to be stored, but there is no particular expectation that the values will be examined in a search of the database.
- An array-valued column may be an appropriate choice for databases that support such columns.

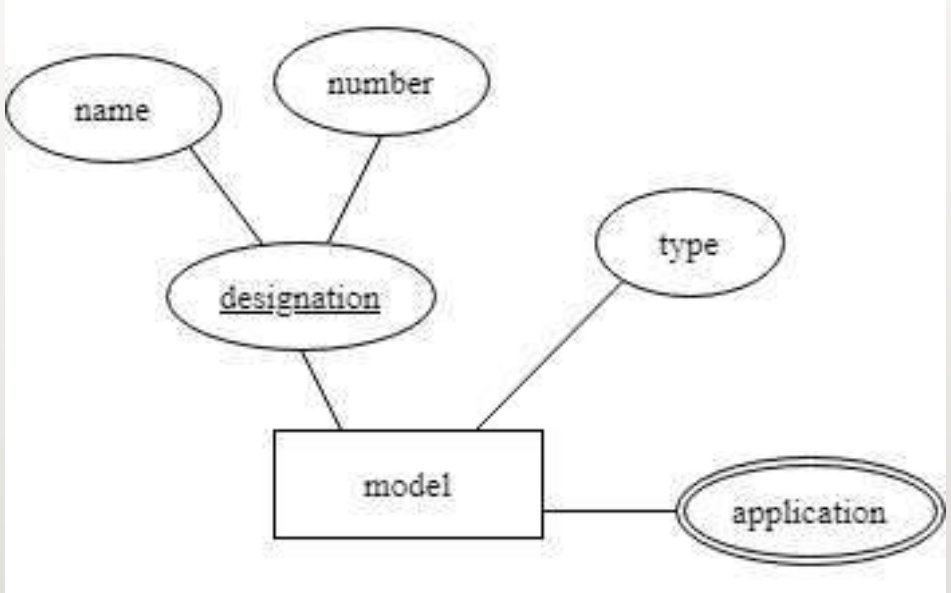


Table model

Column name	Type
number	int
type	char
Keys Primary key: name, number	

Column name	Type
model_name	char
model_number	char
application	Char
Keys Primary key: model_name, model_number, application Foreign key: (model_name, model_number) → model (name, number)	

Table application

Column name	Type
application	char
Keys Primary key: application	

SELF-ASSESSMENT QUESTIONS

1. Data is

(a) Used in decision making
(b) Raw facts or events
(c) Transformed facts
(d) Information

2. What is a database?

(a) Organized collection of information that cannot be accessed, updated, and managed
b) Collection of data or information without organizing
c) Organized collection of data or information that can be accessed, updated, and managed
d) Organized collection of data that cannot be updated

SUMMARY

- ER Model, when conceptualized into diagrams, gives a good overview of entity-relationship, which is easier to understand.
- ER diagrams can be mapped to relational schema, that is, it is possible to create relational schema using ER diagram.
- Entity-Relationship (ER) diagram to a relational model involves translating the conceptual entities and relationships into relational database tables

TERMINAL QUESTIONS

1. Describe the history of DBMS.

2. List out the application areas of DBMS.

3. Analyze DBMS and file systems.

4. Summarize the characteristics of database approach.

5. List out the different datamodels in DBMS.

REFERENCES FOR FURTHER LEARNING OF THE SESSION

Reference Books:

1. 1. Database System Concepts, Sixth Edition, Abraham Silberschatz, Yale University Henry, F. Korth Lehigh University, S. Sudarshan Indian Institute of Technology, Bombay.
2. Fundamentals of Database Systems, 7th Edition, RamezElmasri, University of Texas at Arlington, Shamkant B. Navathe, University of Texas at Arlington.

Web Link:

1. https://nptel.ac.in/courses/106105175

THANK YOU

(DEEMED TO BE UNIVERSITY)

Team – DBMS