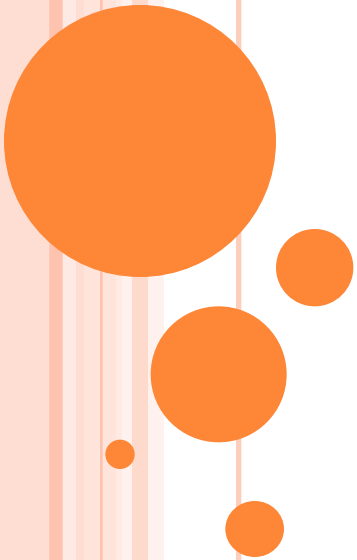


Normalisation

SIT103

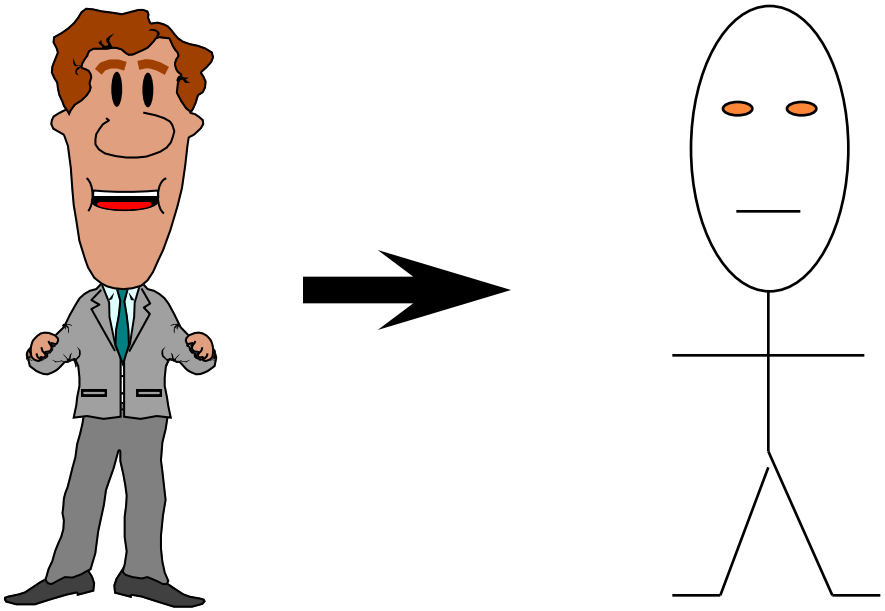
Normal Forms, Functional Dependence

Partial/Transitive Dependencies

Primary Keys, Referential Integrity

Normalisation

- This is the process of converting complex data structures into a simple, stable structure.



Normalisation - Steps

Normalization is a mechanical process to evaluate and correct the table structure, with an aim to eliminate **data redundancies**

- – Goes through a number of stages known as "**normal form**"
- – First stage is called **the first normal form** (**1NF**)
- • Generally, (n+1)-NF is better than the n-NF
- – For most situations, **3NF** is the furthest we go
- – Higher normal forms has less redundancies

Normal Forms

Normal Form	Characteristics
First Normal Form (**1NF**)	Table Format, no repeating groups, PK identified
Second Normal Form (**2NF**)	1NF and no partial dependencies
Third Normal Form (**3NF**)	2NF and no transitive dependencies
Boyce-Codd Normal Form (**BCNF**)	Special case of 3NF
Fourth Normal Form (**4NF**)	3NF and no independent multivalued dependencies

Example

- Imagine a company that sells video games

EXAMPLE(CONT.)

- NO NF

Sales Records:

Cust Name	Item	Shipping Address	Newsletter	Supplier	Supplier Phone	Price
Alan Smith	Xbox One	35 Palm St, Miami	Xbox News	Microsoft	(800) BUY-XBOX	250
Roger Banks	PlayStation 4	47 Campus Rd, Boston	PlayStation News	Sony	(800) BUY-SONY	300
Evan Wilson	Xbox One, PS Vita	28 Rock Av, Denver	Xbox News, PlayStation News	Wholesale	Toll Free	450
Alan Smith	PlayStation 4	47 Campus Rd, Boston	PlayStation News	Sony	(800) BUY-SONY	300

1st Normal Form:

- Each Cell is single valued
- Entries in a column are same type
- Rows uniquely identified-Add unique ID, or Add more columns to make it unique

Primary Key

Cust ID	Cust Name	Item	Shipping Address	Newsletter	Supplier	Supplier Phone	Price
at_smith	Alan Smith	Xbox One	35 Palm St, Miami	Xbox News	Microsoft	(800) BUY-XBOX	250
roger25	Roger Banks	PlayStation 4	47 Campus Rd, Boston	PlayStation News	Sony	(800) BUY-SONY	300
wilson44	Evan Wilson	Xbox One	28 Rock Av, Denver	Xbox News	Microsoft	(800) BUY-XBOX	250
wilson44	Evan Wilson	PS Vita	28 Rock Av, Denver	PlayStation News	Sony	(800) BUY-SONY	200
am_smith	Alan Smith	PlayStation 4	47 Campus Rd, Boston	PlayStation News	Sony	(800) BUY-SONY	300

EXAMPLE(CONT.)

- Does the price depends on customer?
- Does the supplier depends on customer?

- 2NF
  - 1NF and All attributes/fields depend on the key

Primary Key

Cust ID	Cust Name	Shipping Address	Newsletter
at_smith	Alan Smith	35 Palm St, Miami	Xbox News
roger25	Roger Banks	47 Campus Rd, Boston	PlayStation News
wilson44	Evan Wilson	28 Rock Av, Denver	Xbox News
wilson44	Evan Wilson	28 Rock Av, Denver	PlayStation News
am_smith	Alan Smith	47 Campus Rd, Boston	PlayStation News

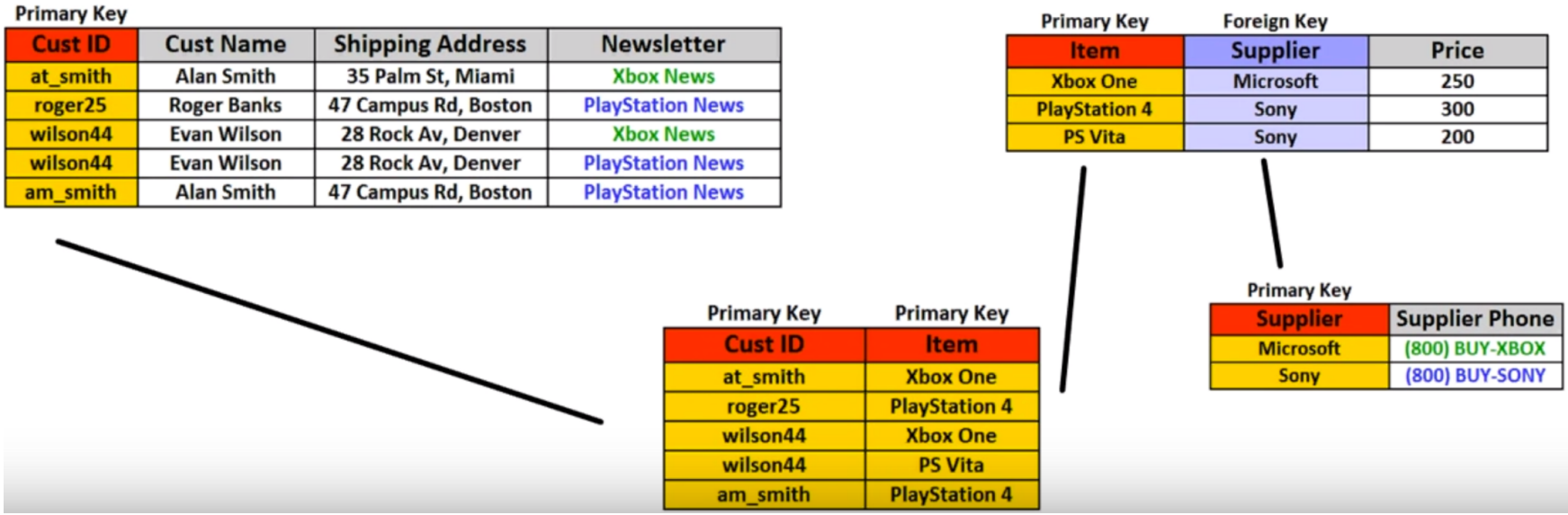
Primary Key

Item	Supplier	Supplier Phone	Price
Xbox One	Microsoft	(800) BUY-XBOX	250
PlayStation 4	Sony	(800) BUY-SONY	300
PS Vita	Sony	(800) BUY-SONY	200

EXAMPLE(CONT.)

- Assume there are 100 records of Items in 2NF. One supplier changes his phone number. How records are going to be updated?

- 3NF
  - 2NF and all fields/attributes can be determined only by the key in the table and no other key



Type of Dependencies

- Functional Dependency:
  - A relationship between two attributes
  - One field is dependent on another, the first field value would not come into existence unless the second field value does.

  - IF B depends on A there is only 1 value for B for each value of A
  - B only exists when A exists first. A ---> B
  - A is called determinant, B is called dependent

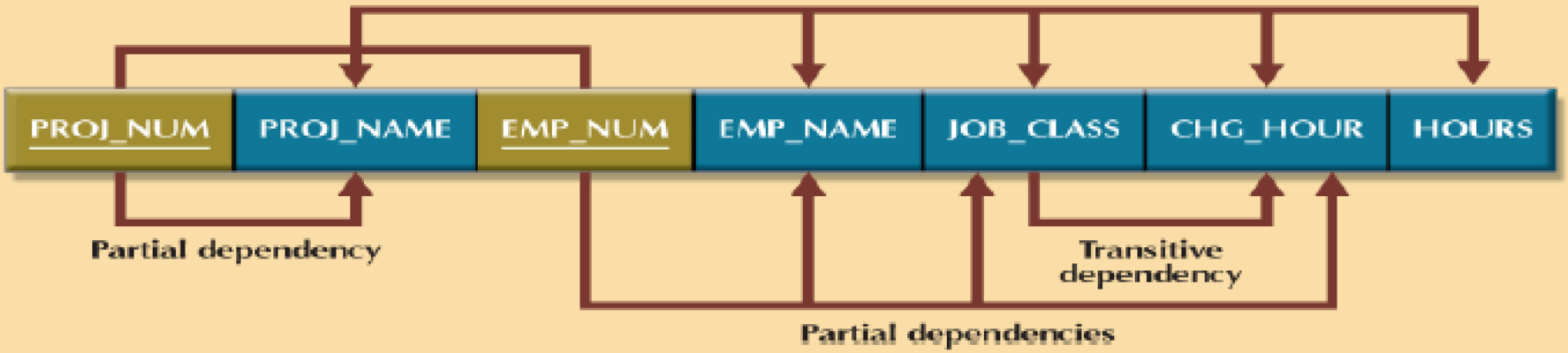
  - – Example:
    - Tax file number → name, address, birth date
    - Vehicle identification No. → make, model, colour

Type of Dependencies

- Partial Dependence
- A dependency based on only a part of a composite primary key;

- Transitive Dependency
- A dependency of one nonprime attribute on another nonprime attribute
- – Prime attribute is an attribute which is not part of a key

1NF



1NF (PROJ_NUM, EMP_NUM, PROJ_NAME, EMP_NAME, JOB_CLASS, CHG_HOURS, HOURS)

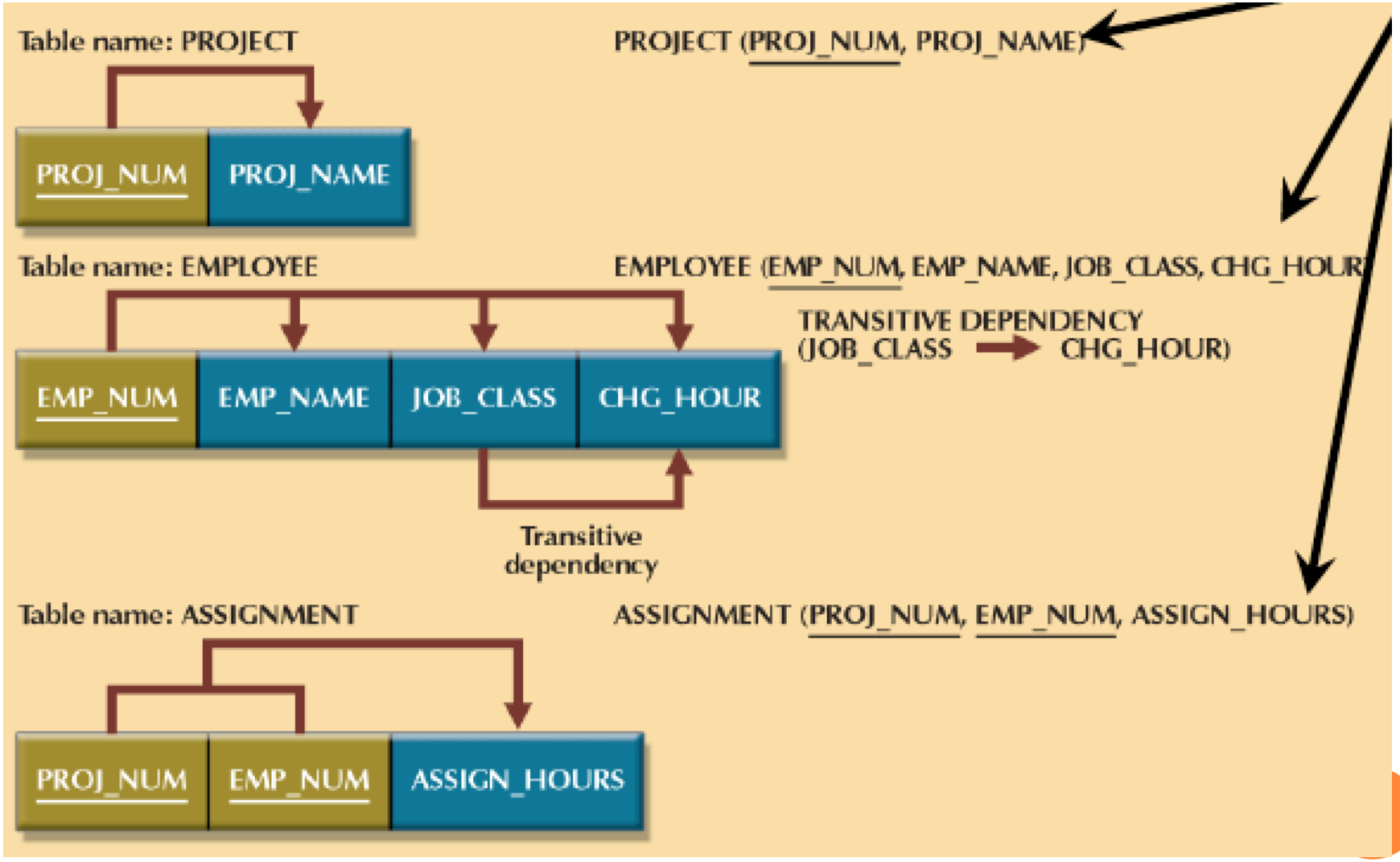
PARTIAL DEPENDENCIES:
(PROJ_NUM → PROJ_NAME)
(EMP_NUM → EMP_NAME, JOB_CLASS, CHG_HOUR)

TRANSITIVE DEPENDENCY:
(JOB CLASS → CHG_HOUR)

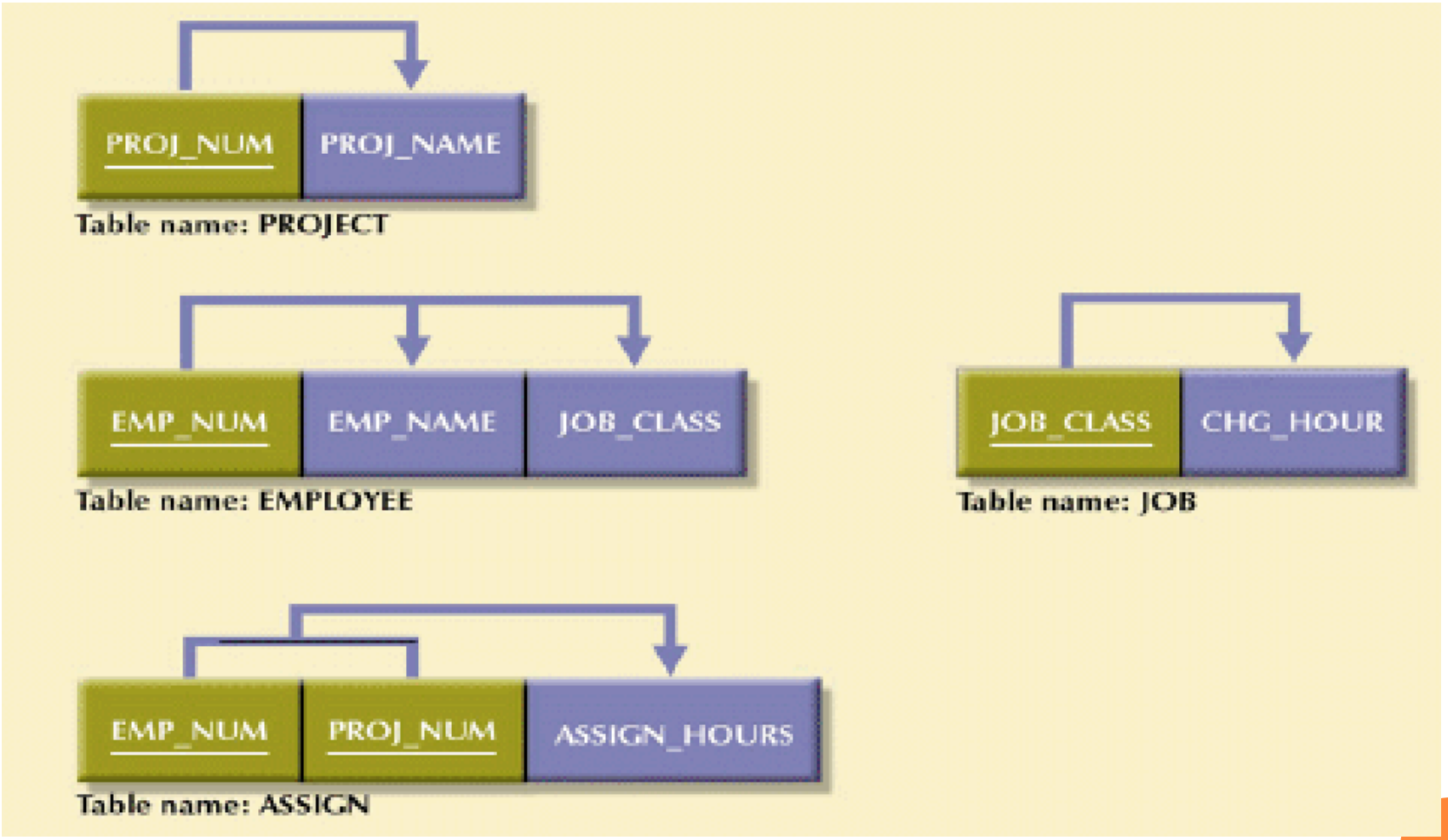
- Above figure is in **1NF**
- To have it in 2NF , get rid of **partial** dependencies
- To have it in 3NF, get rid of **transitive** dependencies

2NF

Relational Model



3NF


PROJ_NUM
PROJ_NAME
Table name: PROJECT
EMP_NUM
EMP_NAME
JOB_CLASS
Table name: EMPLOYEE
JOB_CLASS
CHG_HOUR
Table name: JOB
EMP_NUM
PROJ_NUM
ASSIGN_HOURS
Table name: ASSIGN

Summary of Normal Form

- A relation is in 1NF if and only if
  - – There is no repeating group
  - – All attributes are dependent on the PK
- • A relation is in 2NF if and only if
  - – It is in 1NF
  - – There is no partial dependency
- • A relation is in 3NF if and only if
  - – It is in 2NF
  - – There is no transitive dependency

Other Normal Forms

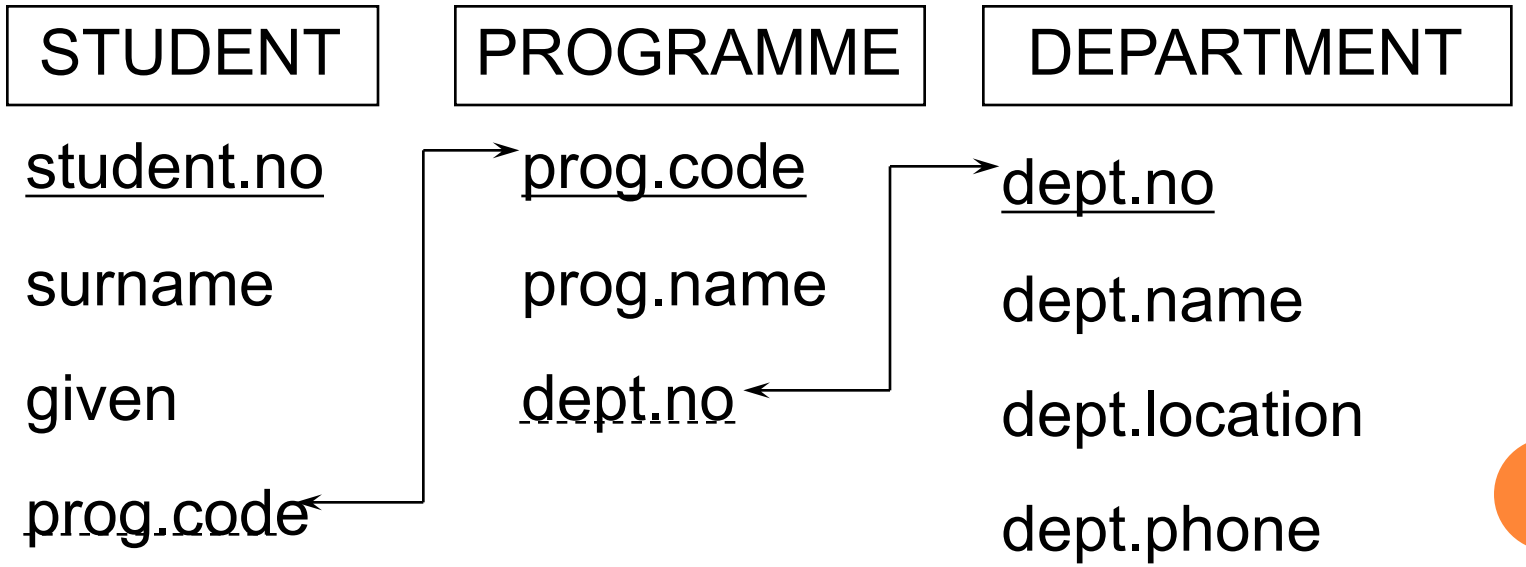
- Boyce-Codd Normal Form
  - remove remaining anomalies resulting from functional dependencies;
- Fourth Normal Form
  - remove anomalies that result from a multi-valued dependencies;
- Fifth Normal Form
  - designed to cope with dependency known as join dependency.

Keys

- Primary Key
  - Unique identifier (field or fields) of a table
  - properties of a primary key are:
    - Uniqueness
    - Availability
    - Stability
    - Minimality
- Candidate Key
  - A field or fields that could be a primary key
- Composite Key
  - a primary key containing more than one field.

Referential Integrity

- related to foreign keys only
- values in foreign key must exist in primary key of related file.

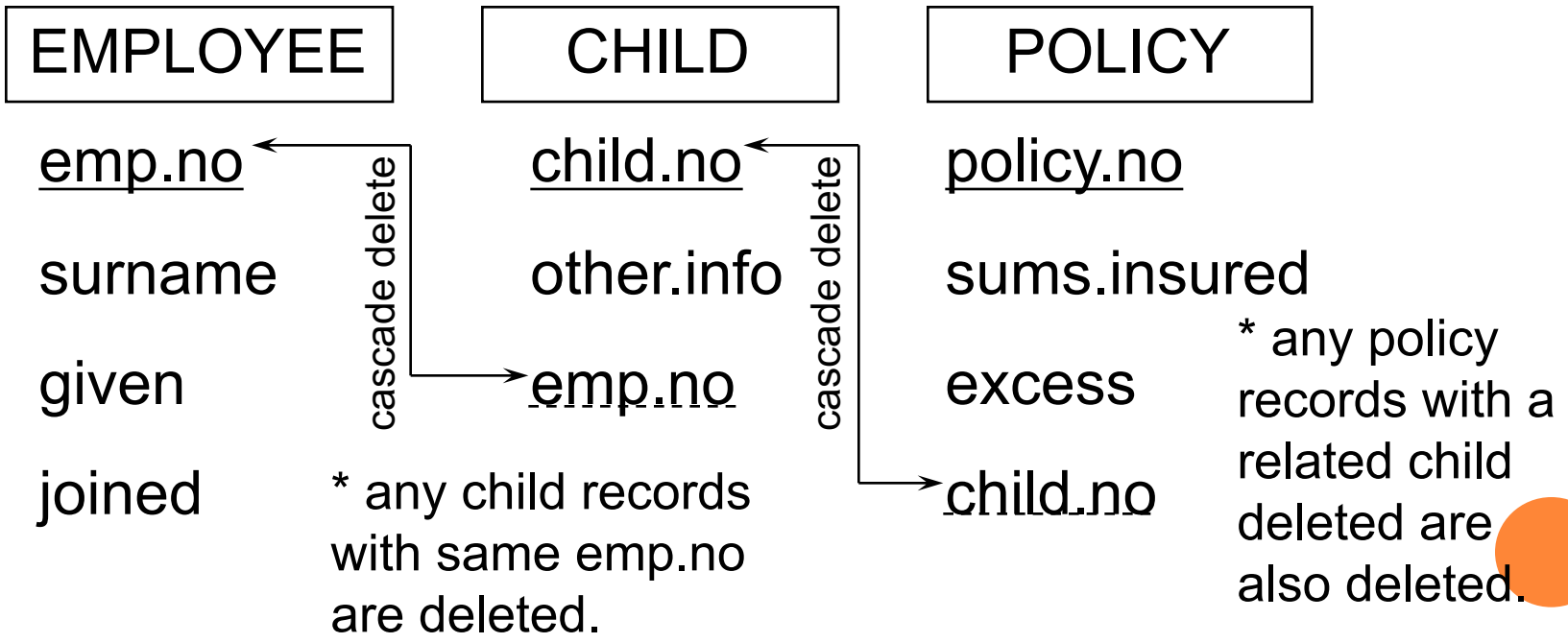


Referential Integrity- Other Issues

- Insert/Update
  - value inserted/change in foreign key must already exist in primary key of other file.
- Delete - three options:
  - not allow;
  - null out the corresponding foreign key(s);
  - cascade delete - remove entire record and any related foreign keys.

Referential Integrity - Cascade Delete

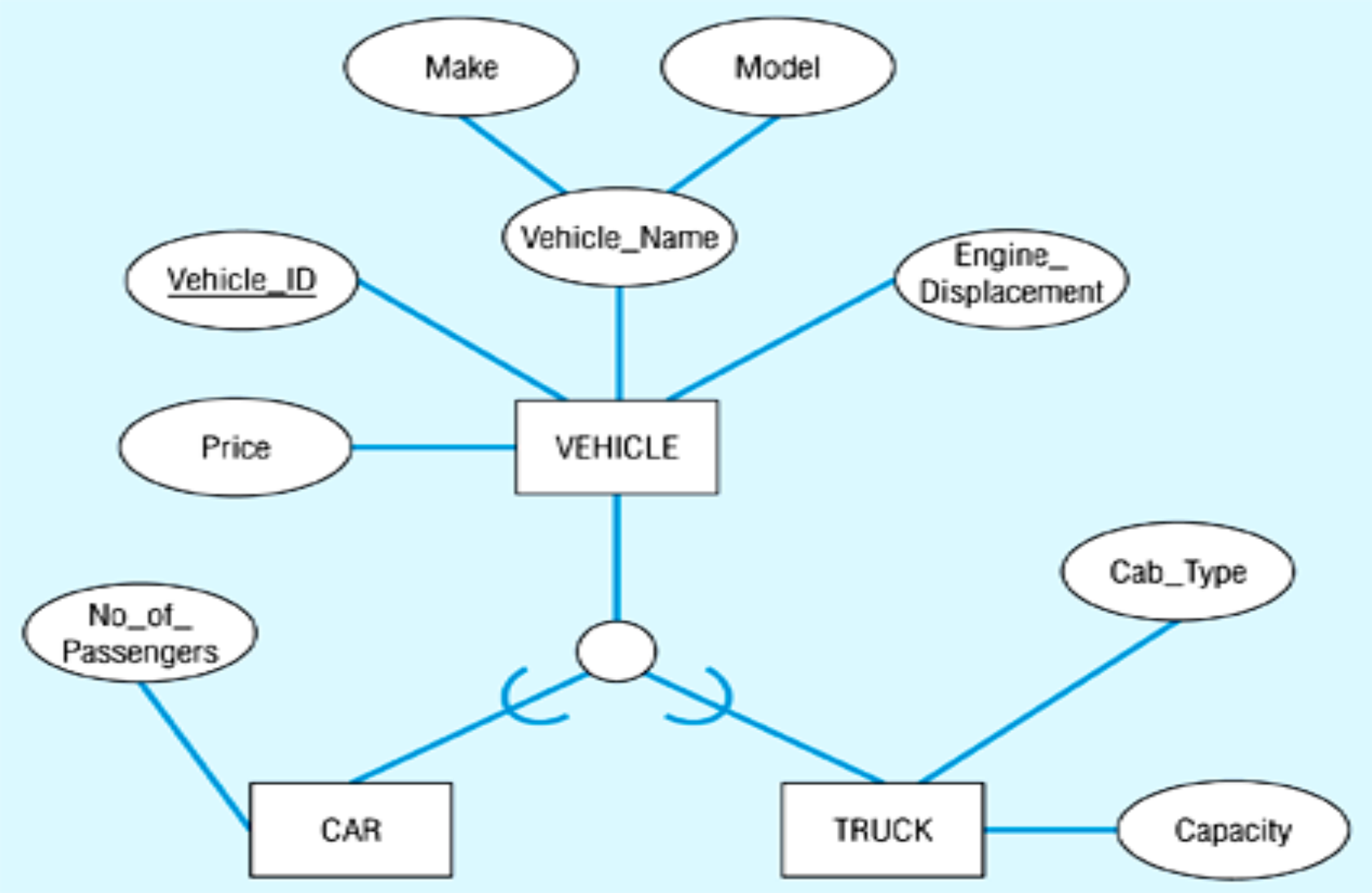
- A delete is issued to delete an employee from the EMPLOYEE file.



Enhanced ER Model (EER)

- Super and Sub Types
- The ER model has been enhanced by various people to include inheritance.
- Concepts have simply been borrowed from the Object Oriented model but drawn differently.
- In practice if you wanted to create a model involving inheritance you would probably go directly to an Object Oriented model.

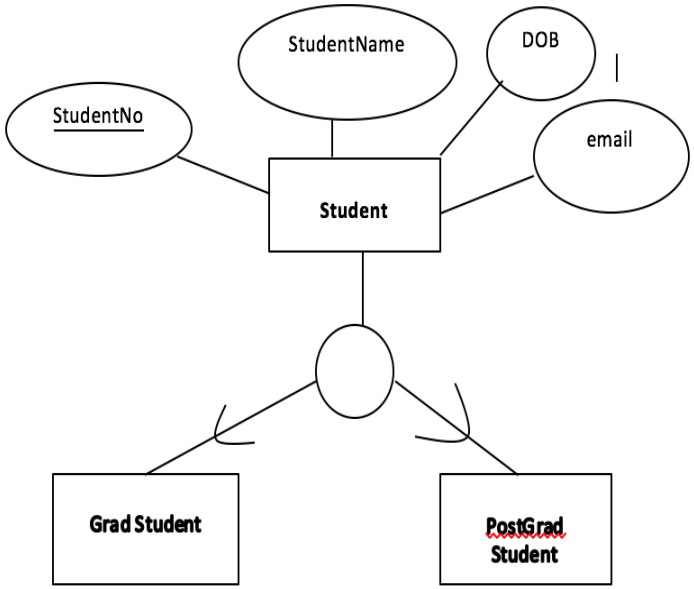
EER Model Super/Sub Types



Mapping EER model to Relational model

- There are FOUR options
  - Option 1 - Create one table one field status (not good if sub entities have attributes) – for have a new field called vehicle type
  - Option 2 - Create one table with extra fields for each sub entities (not work if sub entities have attributes)
  - Option 3 - Create one table with all the attributes of super and sub entities (not good for managing space)
  - Option 4 - Create three tables and connect them using foreign key references (the best option if sub entities and support entity has attributes)

Option 1 – Create one table

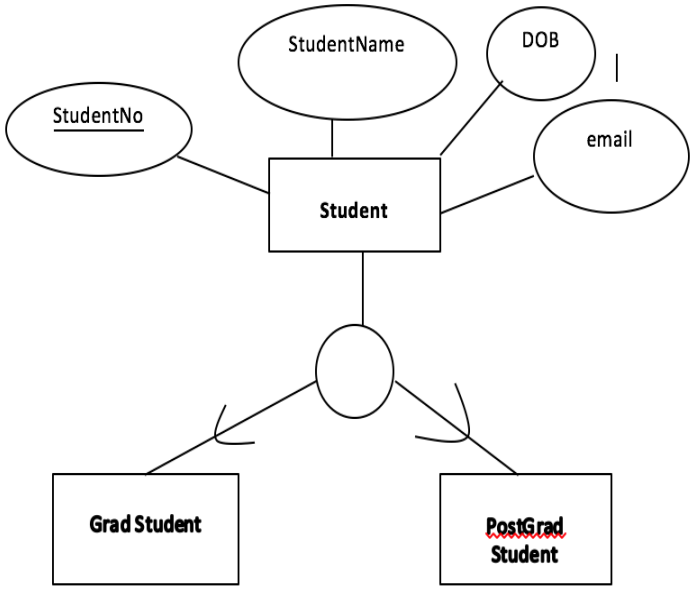


Student(StudentNo, Sname, DOB, email, Status)

This option is good since sub entities and super entities are free with attributes

Status attribute could store "grad" for graduate students and "post" for post graduate students

Option 2 – Create one table

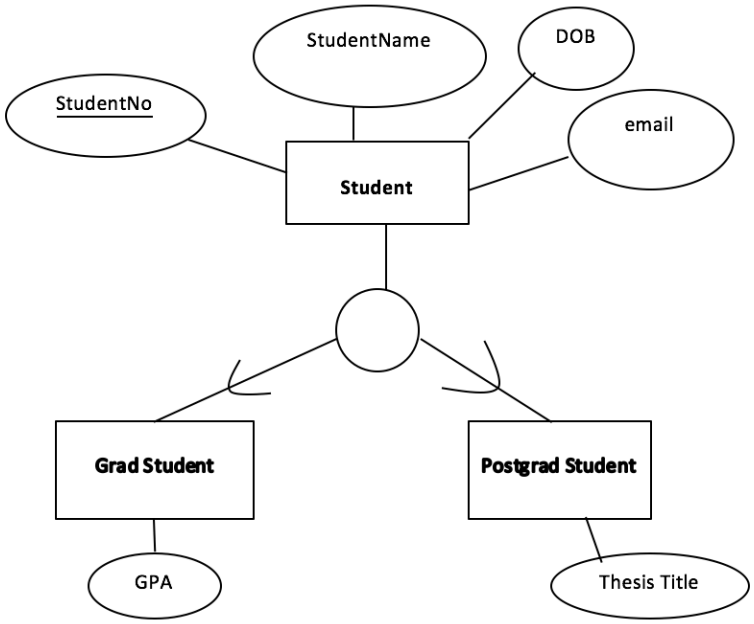


Student(StudentNo, Sname, DOB, email, GradStud, PostGrad)

This option is good since sub entities and super entities are free with attributes

For GradSud and PostGrad columns could be indicated with "Y" if based on the Student group

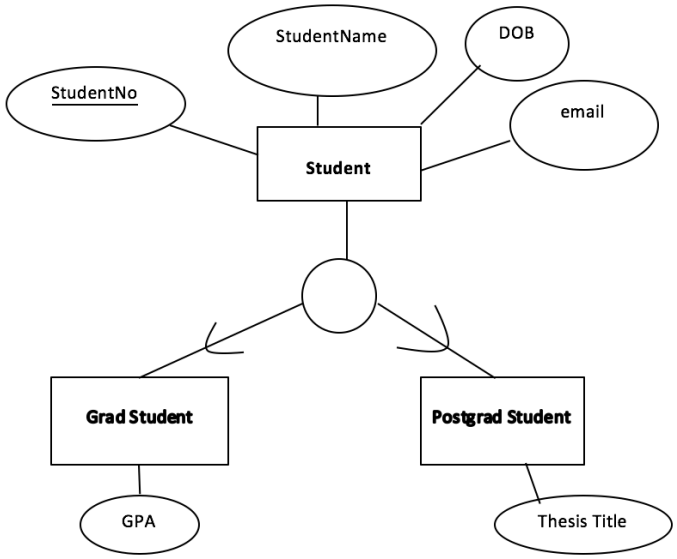
Option 3 – Create one Table



Student(StudentNo, Sname, DOB, email, gpa, thisisTitle)

This option is good since sub entities and super entities are limited with some attributes.

Option 4 – Having multiple tables



Student(StudentNo, Sname, DOB, email)

GradStudent(StudentNo, gpa)

PostGradStudent(StudentNo, ThesisTitle)

This option is good super entity and sub entities have attribute.