MONASH University

MONASH
INFORMATION
TECHNOLOGY

Week 5 – Database Design II: Logical Modelling

FIT3171 Databases
Semester 1 2022

Malaysia Campus

EARLY SEMESTER STUDENT FEEDBACK SURVEY (ESSFS)

Take the survey,
get your voice heard.

MONASH University
MALAYSIA

Reference

Several of the examples and diagrams used this week have been taken from:

Hoffer, J. A. , Prescott, M. B. & McFadden, F. R. “Modern Database Management”

Step 2 (and 3) of the Design Process

- Step 1 Conceptual Model (week 2)
  - Database Model independent
- Step 2 Logical Model (this week)
  - Select which type (model) of database you wish to implement your conceptual model in
    - Network, Relational, OO, XML, NoSQL, ...
  - Database model dependent
- Step 3 Physical Model
  - Select which specific vendor for your chosen model you will implement in
    - Oracle, MySQL, IBM DB2, SQL Server, ...
  - Database vendor dependent
  - Final output schema file to implement model (for relational model a set of tables)

Summary of Terminologies at Different Levels

Conceptual	Logical (Relational)	Physical (Relational)
Entity	Relation	Table
Attribute	Attribute	Column
Instance	Tuple	Row
Identifier (KEY)	Primary Key	Primary Key
Relationship	---	---
---	Foreign Key	Foreign Key

Music Model - Pre Class Activity

- A music database is to be created. The database will include information about ARTISTs, their ALBUMs and the TRACKs on their released albums. The only albums we wish to record are those released by individual artists, so each album will only be released by one artist.
- For each artist we will record the artist's name, the year they were born and the country in which they were born. Each artist will be assigned a unique id.
- For an album we will record the name of the album, who the album was recorded by, and when the album was released. Each album will be assigned a unique id.
- For each track on an album we will record a track number (starting at 1 on every album), the name of the track, the play length for the track and the album the track was recorded on.

Music Model - Pre Class Activity

Conceptual

Relations

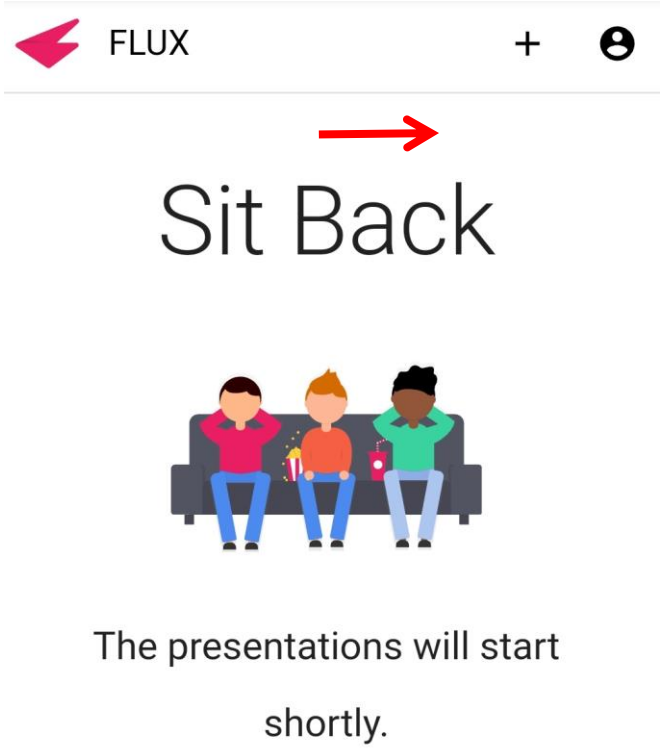
ARTIST (artist_id, artist_name, artist_born, artist_country)

ALBUM (album_id, album_name, album_release, artist_id)

TRACK (album_id, track_no, track_name, track_length)

Flux.qa: for lecture participation

flux.qa/QBGYRS



Q1. Which of the following are invalid relations (note your reasons as you make your decisions):

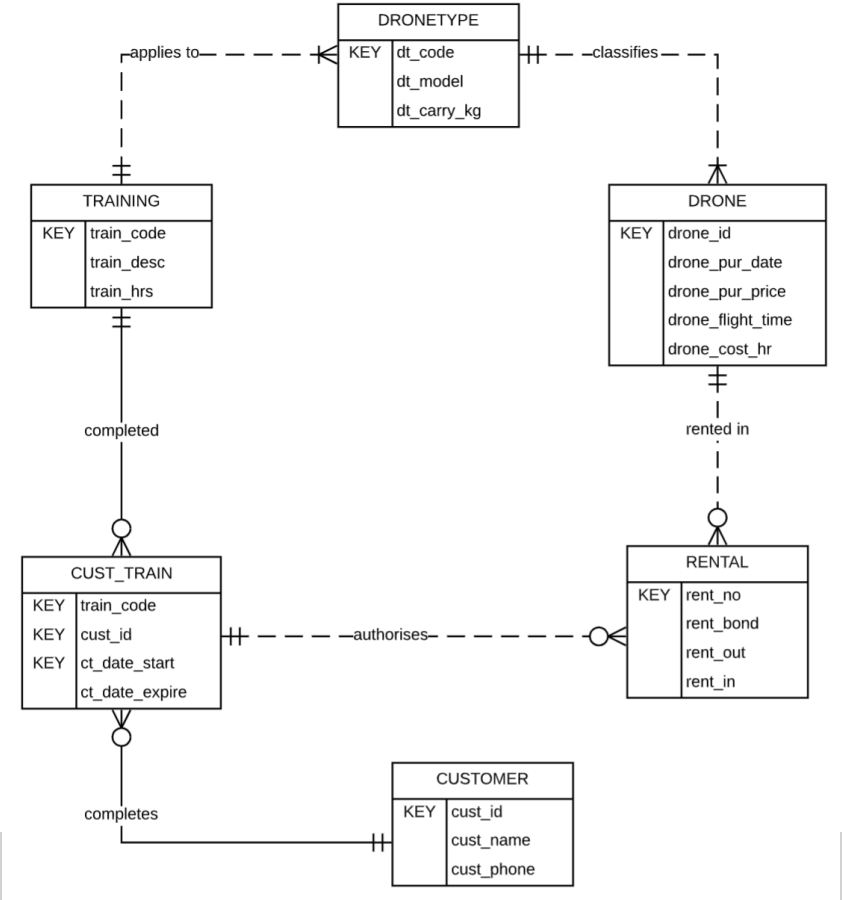
A. EMPLOYEE (empname, empdept, empno, empsalary)
B. CUSTOMER (custno, custname, custphone, custphone)
C. ORDER (orderno, orderdate, custno)
D. PRODUCT (prodno, proddesc, produprice)
E. TRIP (trip_id, driver_id, driver_name, (stop_id, stop_time))
F. LICENCE_HELD (driver_id, licence_type)

Multiple answers possible

Week 3 Relational Model Characteristics

- Each relation must have a unique name
- Each attribute of a relation must have a distinct name within the relation
- An attribute cannot be multivalued (consist of repeating values)
- All values of an attribute need to be from the same domain
- The order of attributes and tuples in a relation is immaterial
- Each relation must have a primary key
- Logical (not physical) connections are made between relations by virtue of primary/foreign key pairing

Recap Week 2 HiFlying Drone Conceptual Model



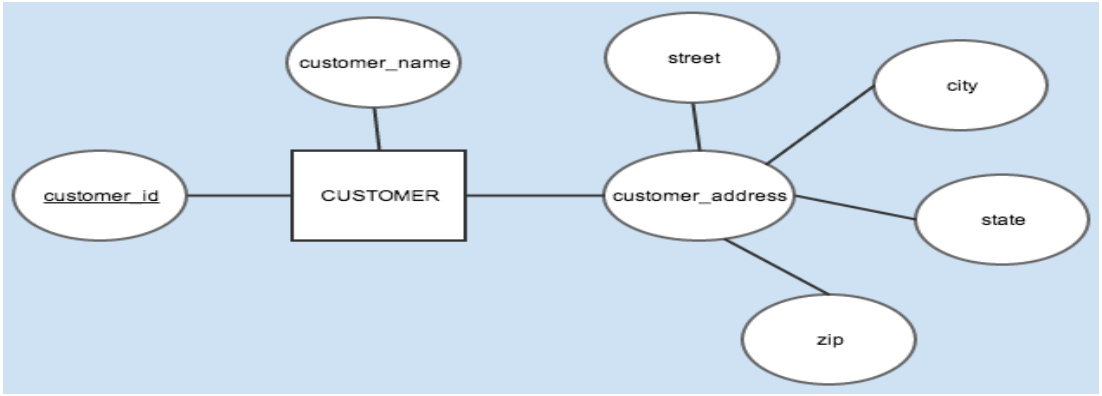
Transforming ER diagrams into relations (mapping conceptual level to logical level)

- Essentially
  - KEY to PK
  - Represent relationships with PK/FK pairs
- The steps are:
  - Map strong (regular) entities
  - Map weak entities
  - Map binary relationships
  - Map associative entities
  - Map unary relationships
  - Map ternary relationships
  - *Map supertype/subtype relationships (is not part of this unit).*

Map Regular Entities

- Composite Attributes
  - When the regular entity type contains a composite attribute, only the simple component attributes of the composite attribute are included in the new relation.
  - Compared to composite attributes, simple attributes not only improve data accessibility but also help in maintaining data quality
  - Mapping a composite to its simple component attributes is the normal action ***if no client specification, to the contrary, is available***
    - however, if in doubt ask eg phone numbers

Mapping a Composite Attribute



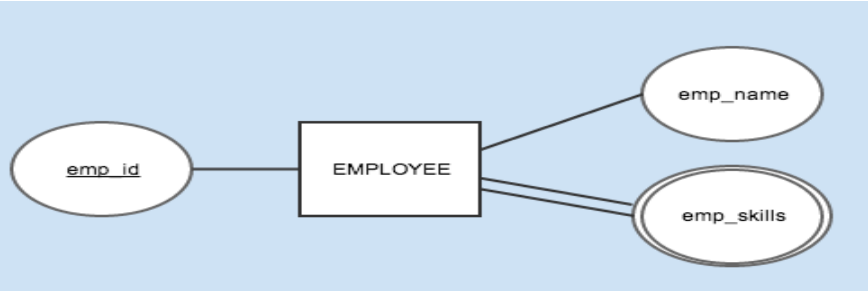
CUSTOMER	
Key	cust_id
	cust_name
	cust_street
	cust_city
	cust_state
	cust_zip

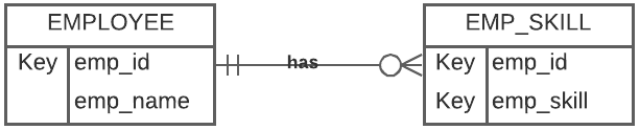
CUSTOMER		
P	*	cust_id
	*	cust_name
	*	cust_street
	*	cust_city
	*	cust_state
	*	cust_zip

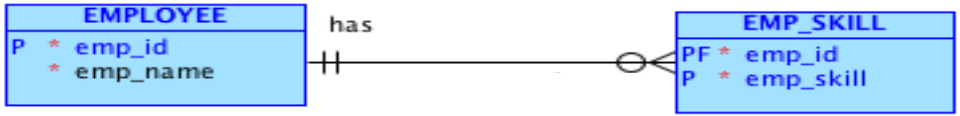
Map Regular Entities

- Multivalued Attribute
  - When the regular entity type contains a multivalued attribute, two new relations are created.
  - The first relation contains all the attributes of the entity type except the multivalued attribute itself.
  - The second relation contains two attributes that form the PK. One of the attributes is the PK from the first relation, which becomes the FK in the second relation and the other is the multivalued attribute.
  - There can also be non key attributes in the second relation depending upon the data requirements.

Mapping a Multi valued Attribute



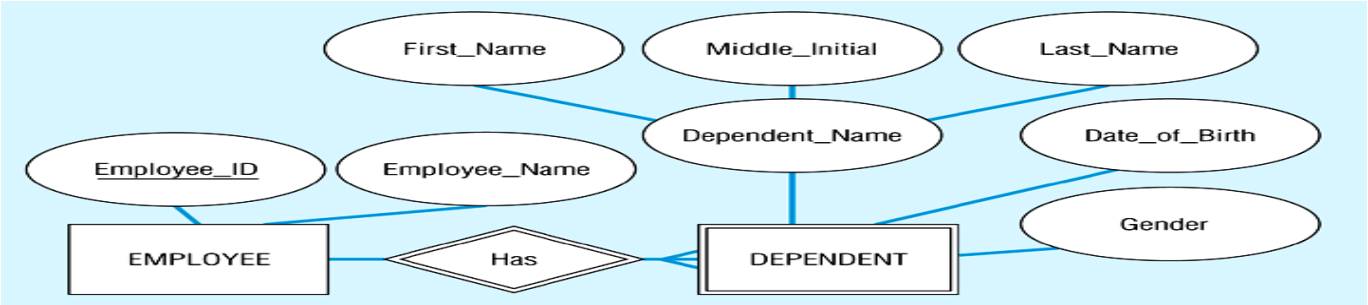


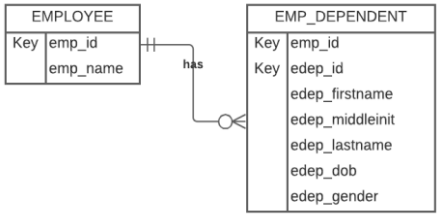


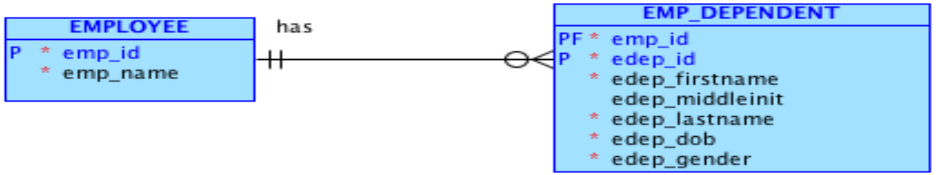
Is there a better solution than the one shown above?
What are the issues here - this was partially discussed in week 2

Mapping a Weak Entity

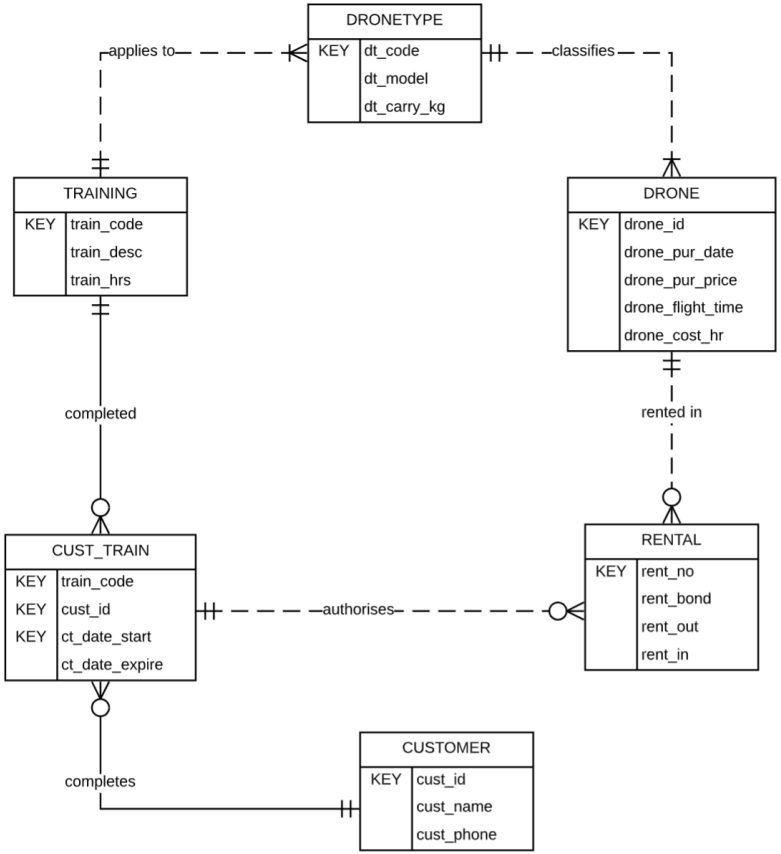
- For each weak entity type, create a new relation and include all of the simple attributes as attributes of this relation. The PK of the identifying relation is also included as the FK in this new relation.



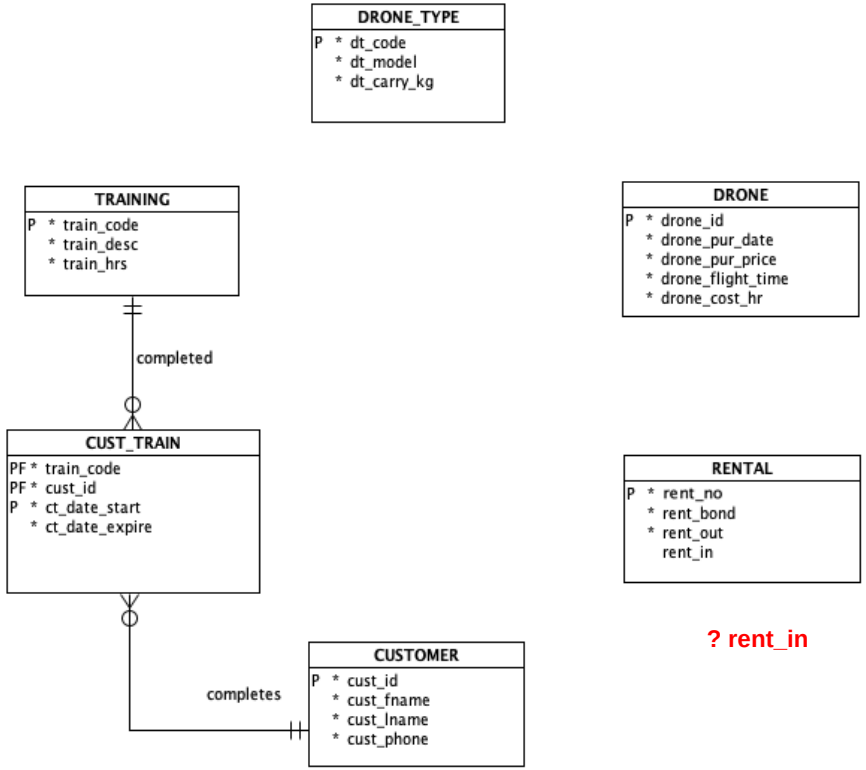




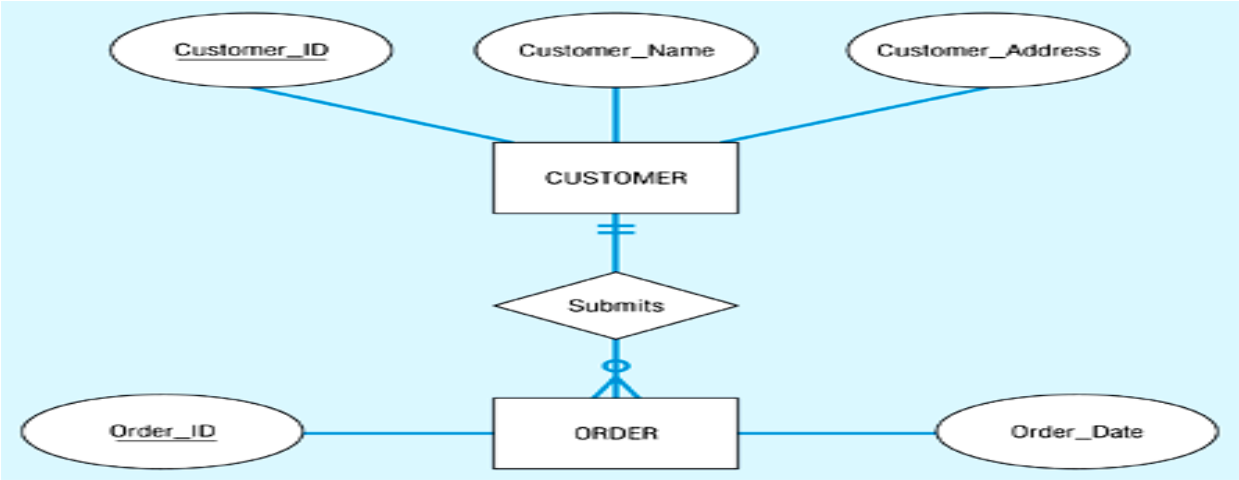
Conceptual

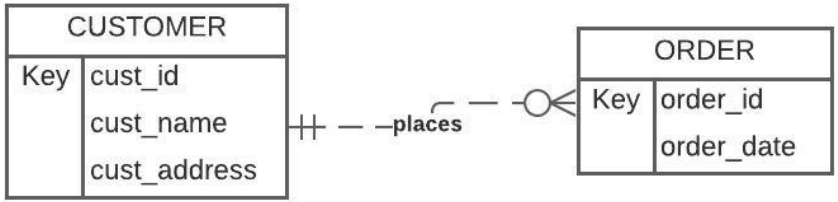


Logical

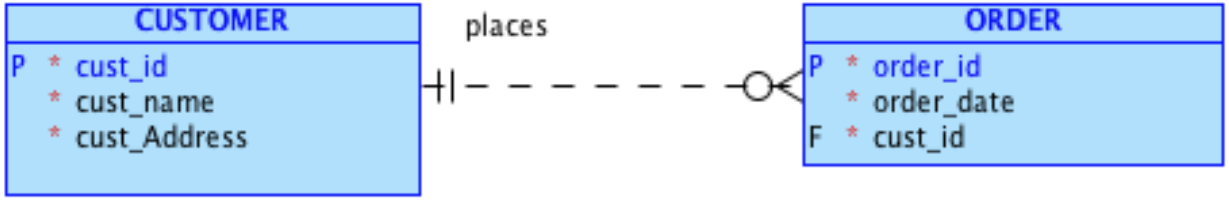


Mapping a 1:M Binary Relationship



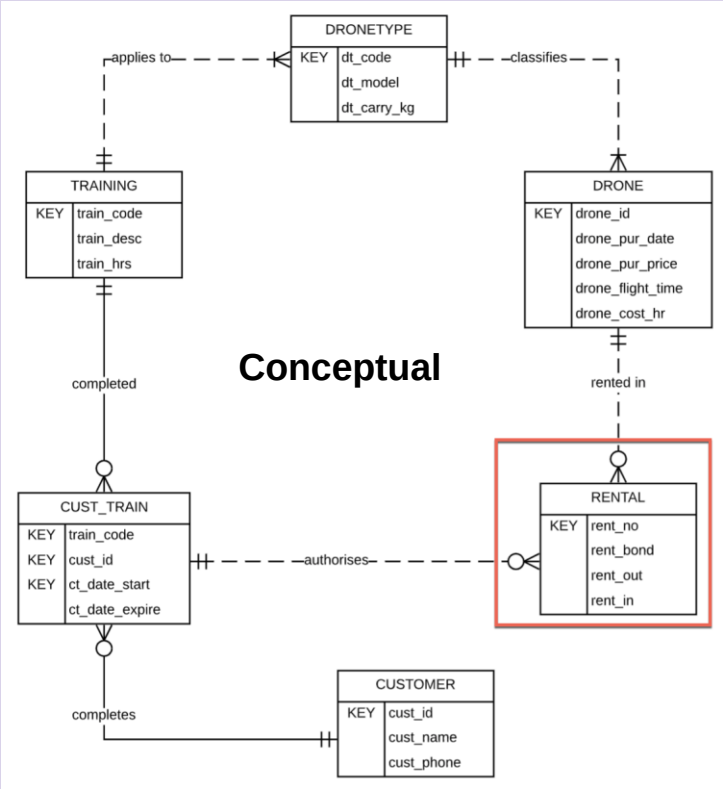


Map Binary Relationships (1:M)



For each 1:M binary relationship, first create a relation for each of the two entity types participating in the relationship. Then include the PK attribute (or attributes) of the entity on the one-side of the relationship as the FK on the many-side of the relationship.

Q2 The full set of attributes that are required in the RENTAL relation will be:



A. rent_no, rent_bond, rent_out, rent_in

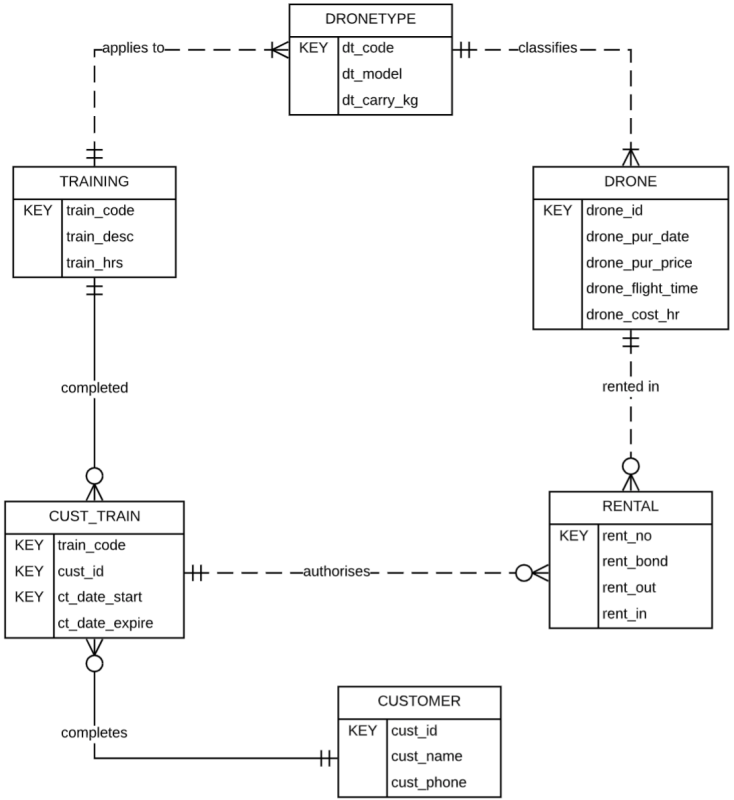
B. rent_no, rent_bond, rent_out, rent_in, drone_id

C. rent_no, rent_bond, rent_out, rent_in, train_code, cust_id, ct_date_start, drone_id

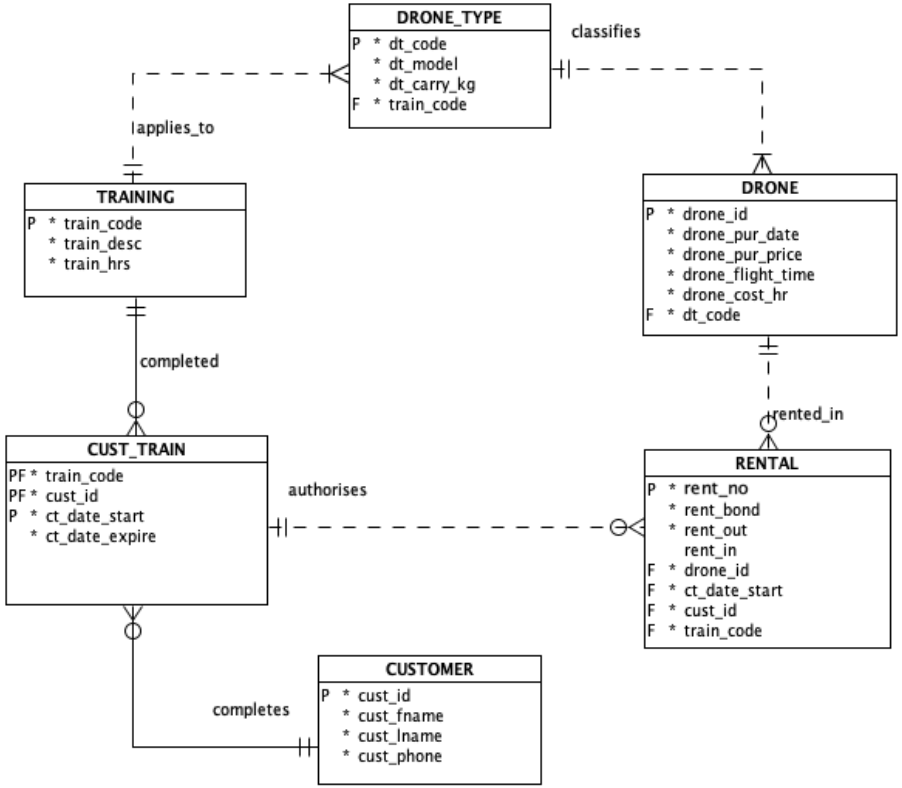
D. rent_no, rent_bond, rent_out, rent_in, train_code, drone_id

E. rent_no, rent_bond, rent_out, rent_in, train_code, cust_id, ct_date_start, drone_id, dr_code

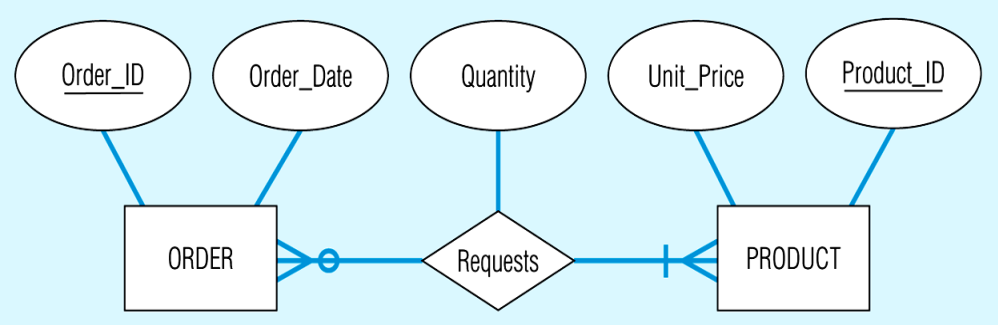
Conceptual

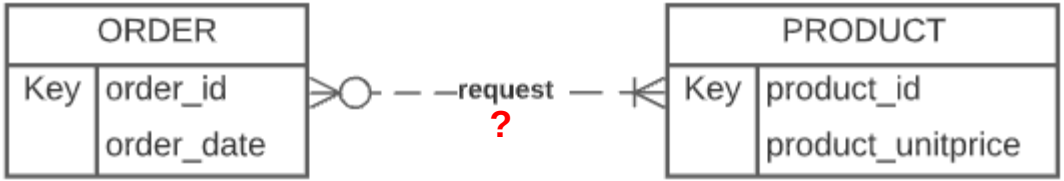


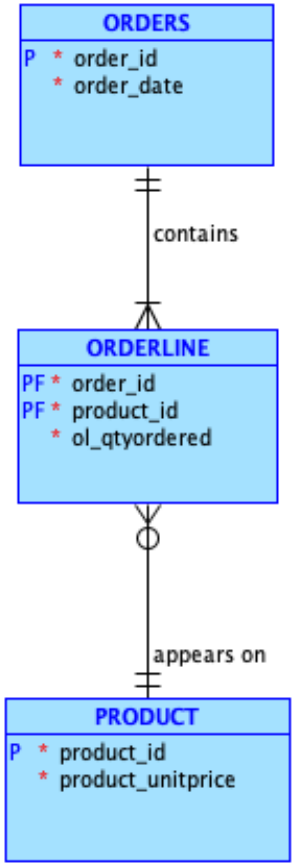
Logical



Mapping a M:N Binary Relationship



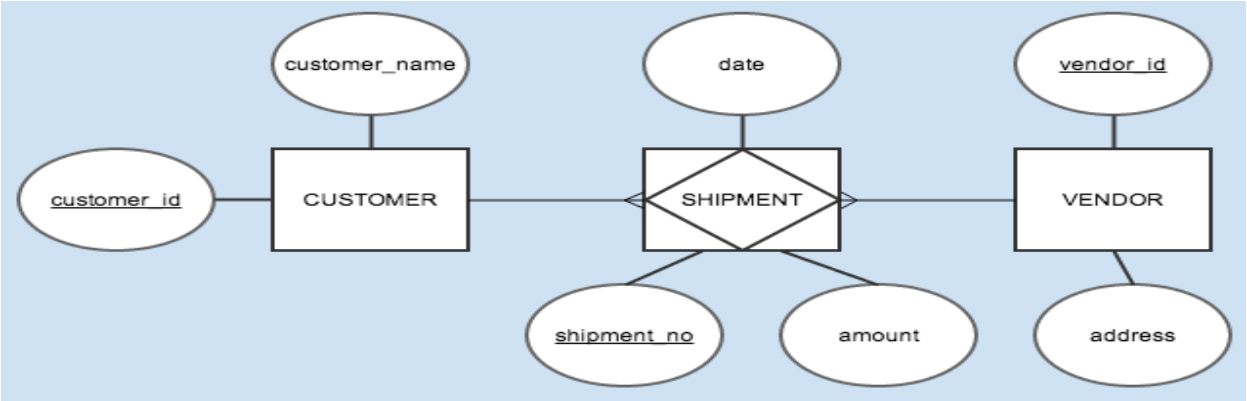


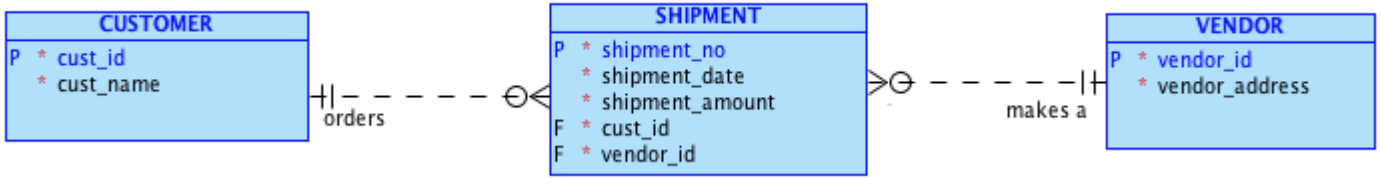

ORDERS
P * order_id
* order_date
contains
ORDERLINE
PF * order_id
PF * product_id
* ol_qtyordered
appears on
PRODUCT
P * product_id
* product_unitprice

Map Binary Relationship (M:N)

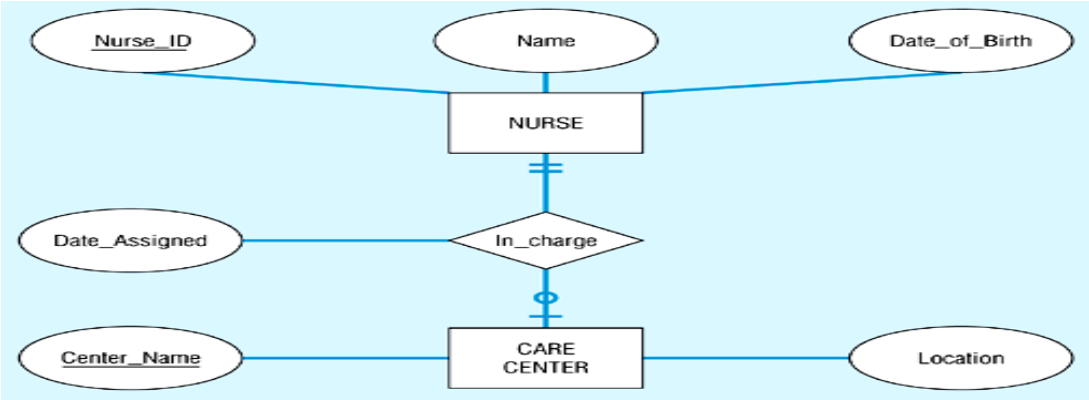
- For a M:N binary relationship
  - First create a relation for each of the two entity types participating in the relationship.
  - Then create a new relation and include as foreign key attributes, the PK attribute (or attributes) for each of the two participating entity types. These attributes become the PK of the new relation.
  - If there are any nonkey attributes associated with the M:N relationship, they are also included in the new relation.

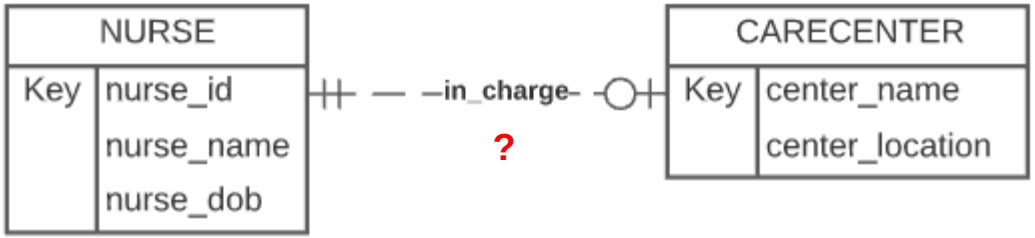
Mapping an associative entity with an Identifier



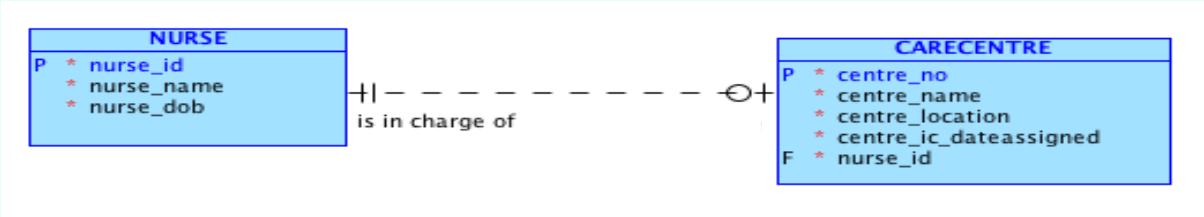


Mapping a 1:1 Binary Relationship





Q3. NURSE participation in this relationship is:



A. Optional
B. Mandatory
C. It depends on the implementation
D. 1:1
E. 0

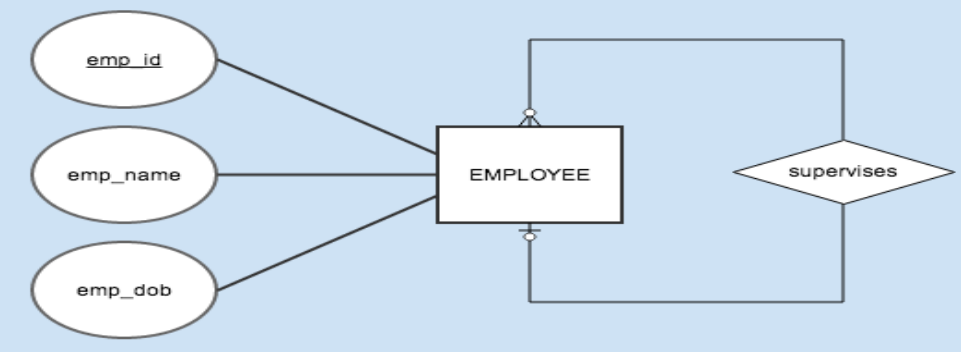
Map Binary Relationship (1:1)

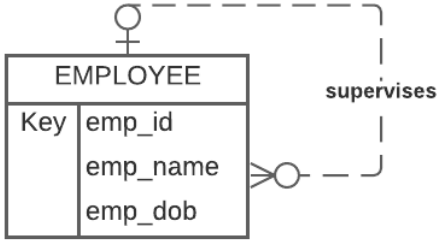
- Create two relations, one for each of the participating entity types.
  - The primary key (PK) on the mandatory side of the relationship becomes the foreign key (FK) on the optional side of the relationship.
  - where both are optional place the FK on the side which causes the fewest nulls
  - Special case: 1:1 total relationship (mandatory participation from both sides)
    - Should consolidating the two entity types into one relation

Map unary relationships

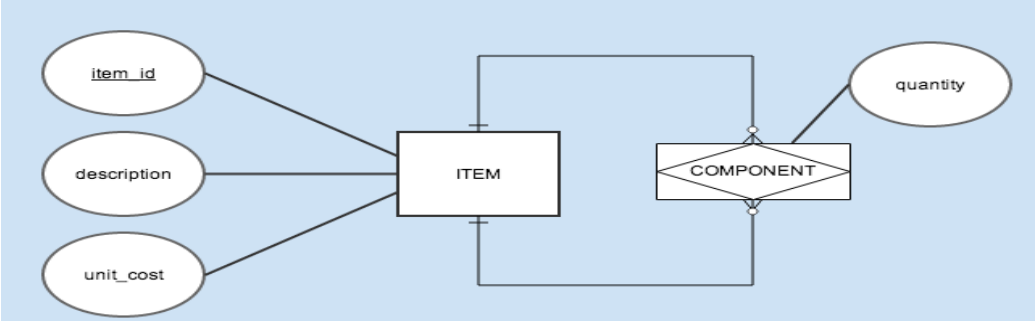
- Unary Relationship is a relationship between the instances of a single entity type.
- Unary 1:M Relationship – A relation is created for the entity type. Add a FK within the same relation that references the PK of the relation. A recursive foreign key is a FK in a relation that references the PK values of the same relation.
- Unary M:N Relationship – Two relations are created, one for the entity type in the relationship and the other as the associative relation to represent the M:N relationship itself. The PK of the associative relation consists of two attributes (with different names) taking their values from the PK of the other relation.

Mapping a 1:M Unary Relationship

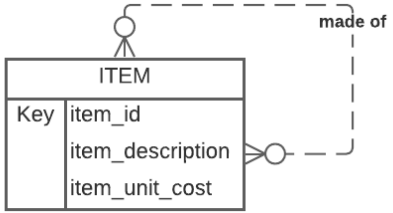


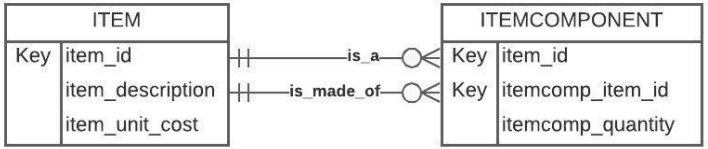


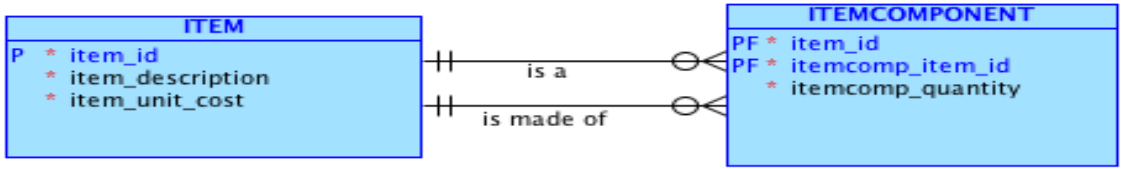
Mapping a M:N Unary Relationship



quantity?

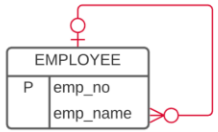


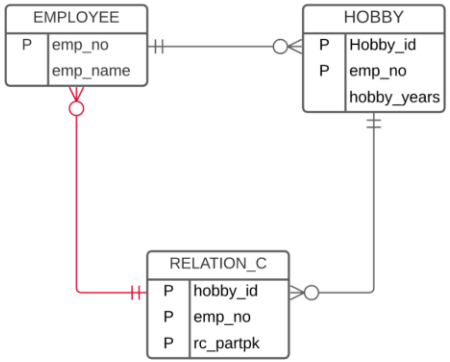




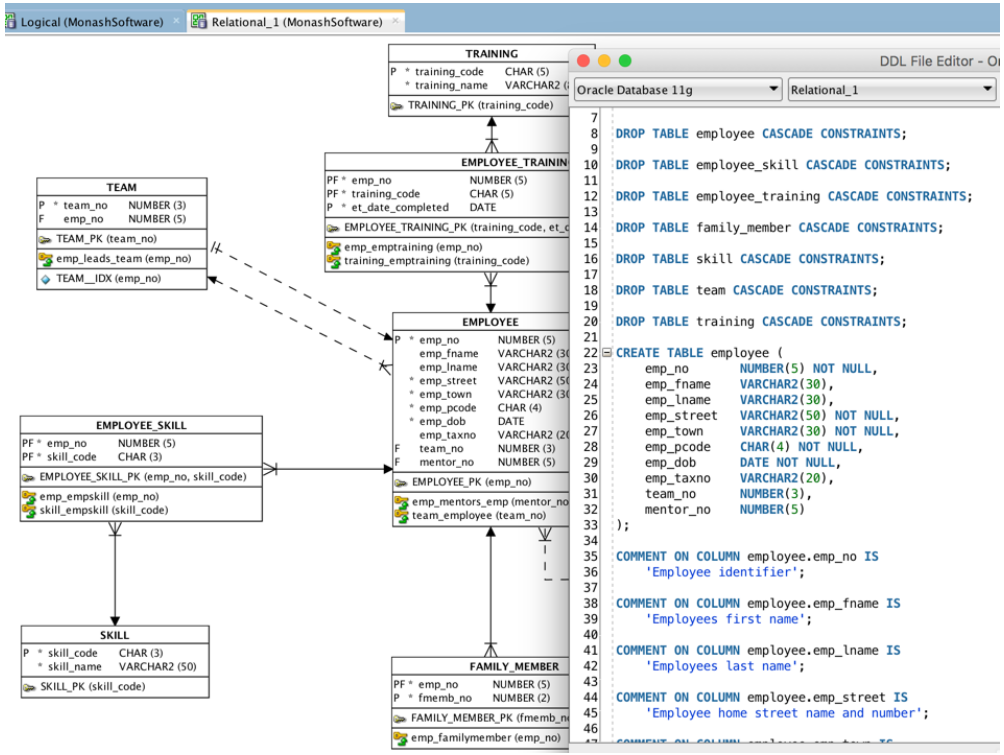
IMPORTANT NOTE

- Apply the principles we have discussed, think carefully about the consequences of relationship placement
  - Recursive identifying relationships CANNOT exist
  - 1:1 Total identifying relationships CANNOT exist
  - Take care with relationship "loops"
- What happens here:

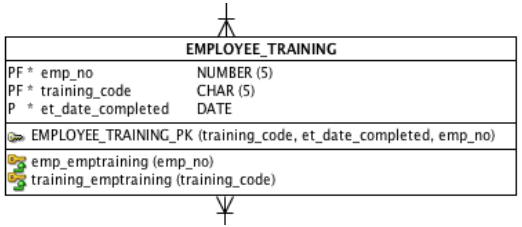




SQL Developer Data Modeler



Adding surrogate keys

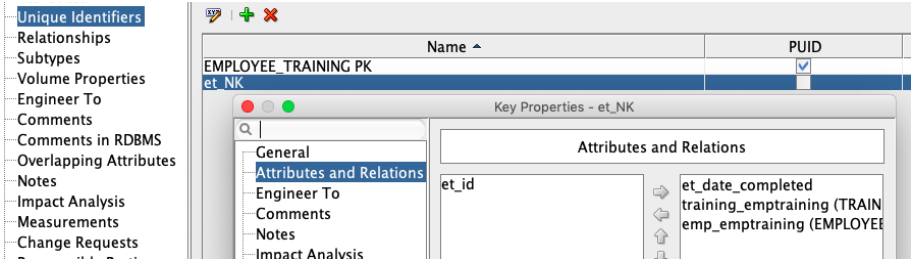


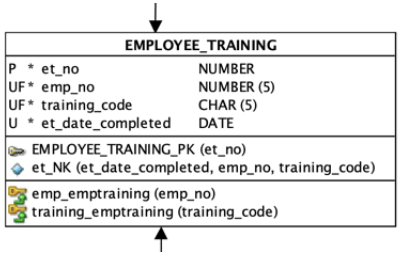
Potential problem:
Need to ensure that the identified key from the conceptual model - the natural key:
(emp_no, training_code, et_date_completed)
will still remain unique
Solution, where needed:
Define a unique index on the attributes of natural key

Surrogate PK's may be added **ONLY** on the logical model provided they are justified (include in documentation / assumptions).

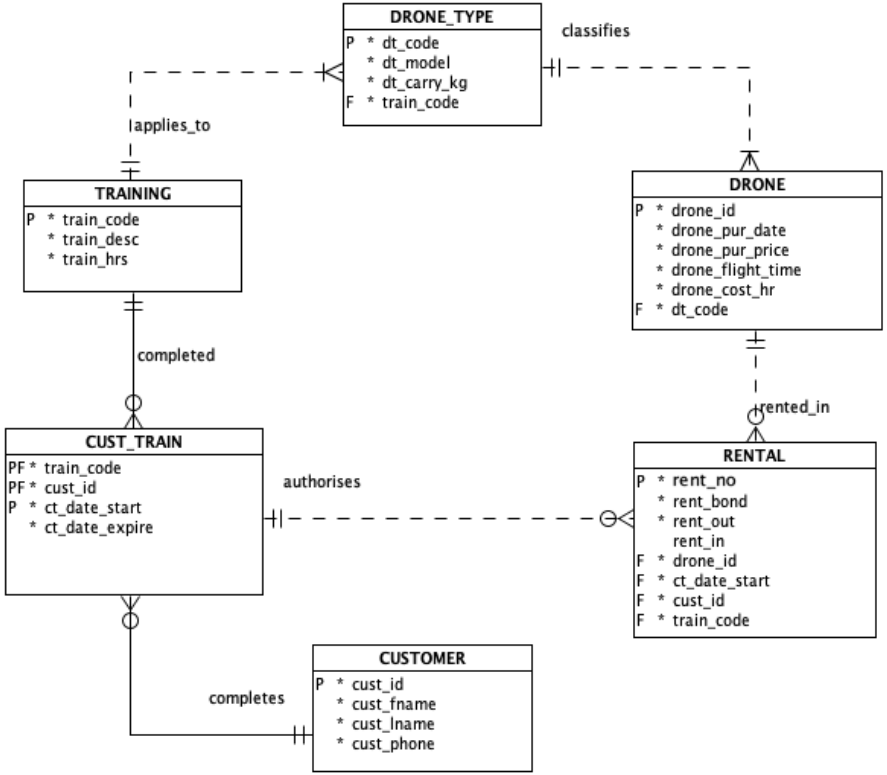
MANUALLY add new PK attribute (here et_no), **DO NOT USE** SQL Developers "Create Surrogate Key" option

et_no	emp_no	training_code	et_date_completed
1	101	ORA01	1-Oct-2016
2	101	ORA01	1-Oct-2016
3	101	ORA01	1-Oct-2016

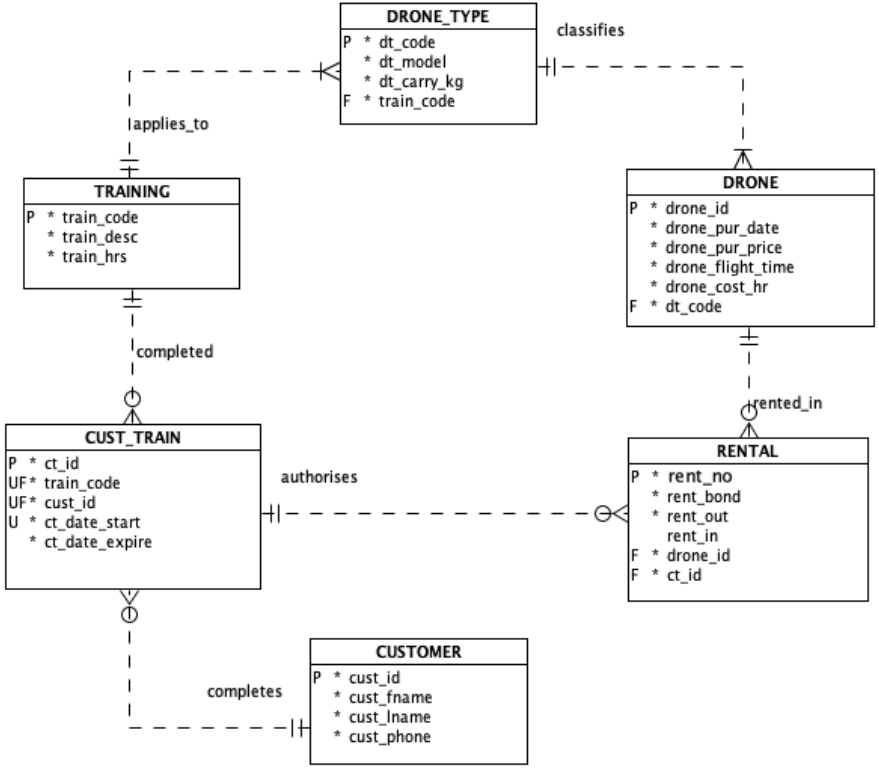




Logical

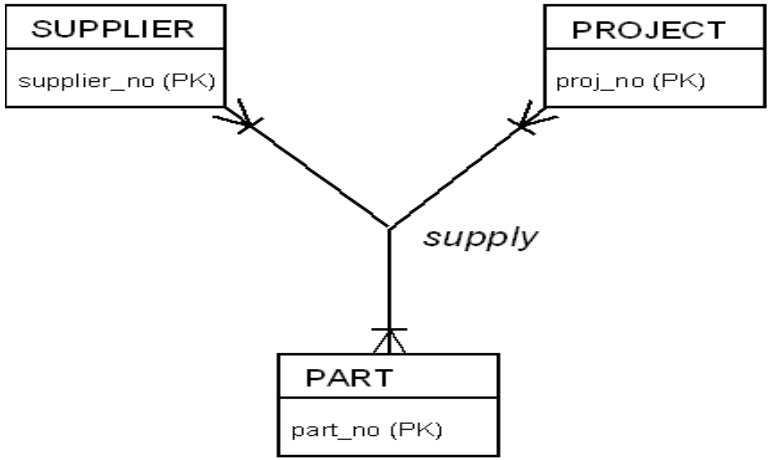


Logical - with Surrogate key

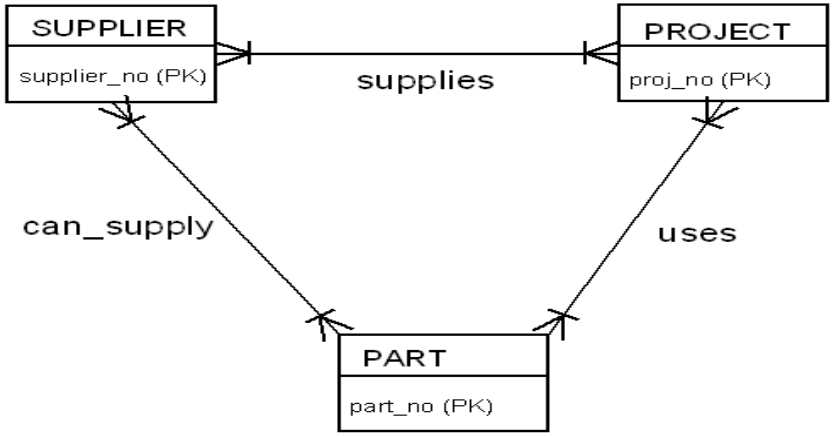


Ternary Relationships

Ternary



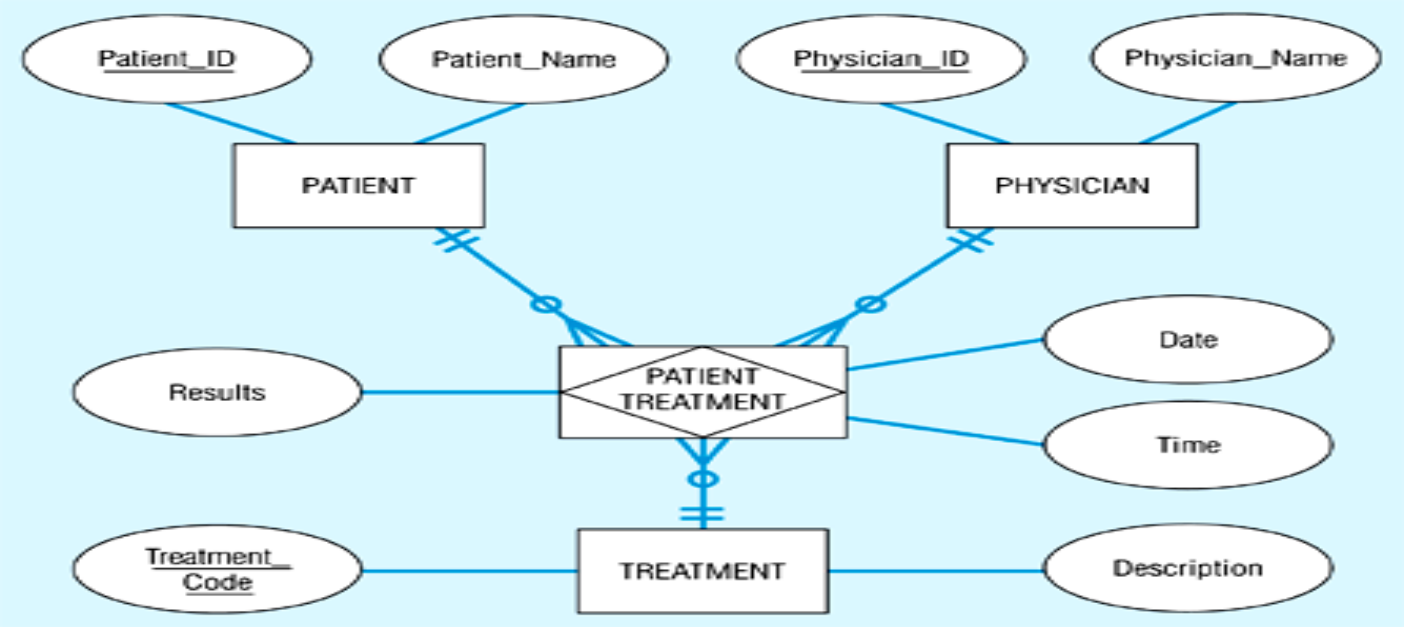
modelled as binary:



Ternary Relationships – model as binary relationships?

- Ternary represents more information than three binary relationships
- For example - Supplier 1 supplies Project 2 with Part 3 -
  - ternary
    - instance (supplier 1, project 2, part 3) exists
  - binaries
    - instances
      - (supplier1, project 2) (project 2, part 3) (supplier 1, part 3)
    - BUT does not imply (supplier 1, project 2, part 3)
- How then do we map such relationships?

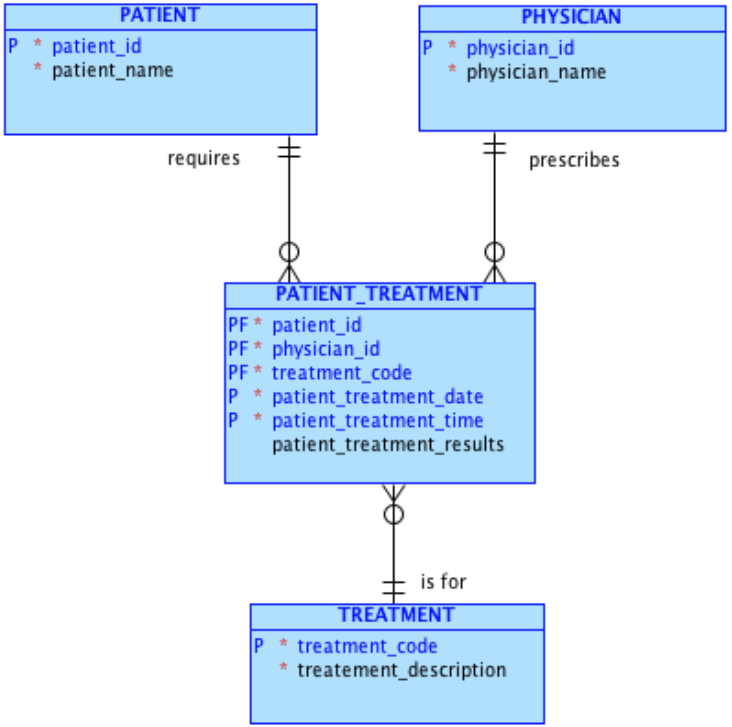
Mapping a Ternary Relationship



Map Ternary (and n-ary) Relationships

- Ternary relationship should be converted to an associative entity.
  - To map an associative entity type that links three regular entity types, an associative relation is created.
  - The default PK of this relation consists of the three PK attributes for the participating entity types.
  - Any attributes of the associative entity type become attributes of the new relation.

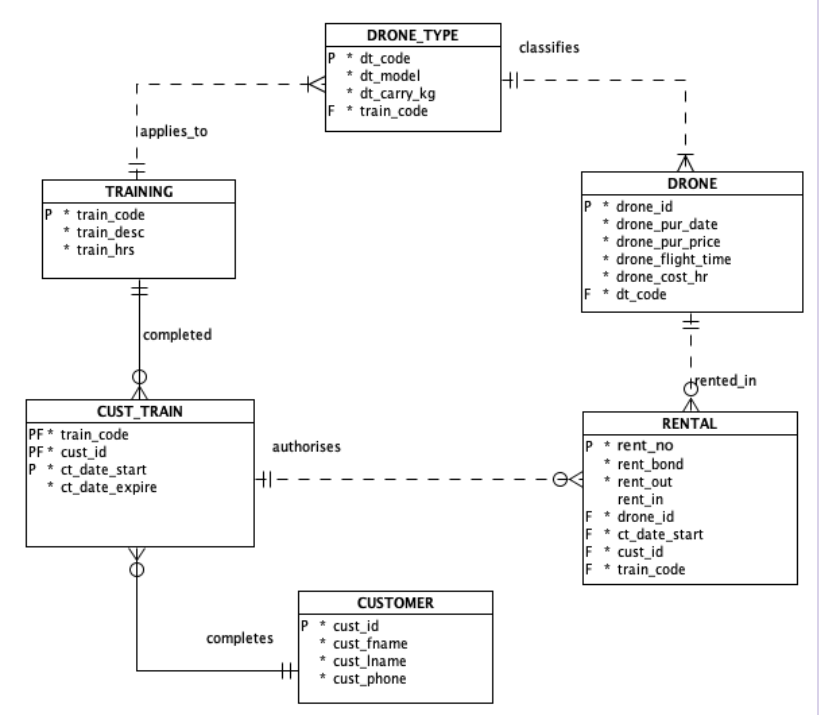
Mapping a Ternary Relationship



Overall Design Process - checklist

1. Assignment 1A
   a. complete conceptual model
2. Assignment 1B
   - free to modify submitted conceptual model in any manner - note will not be reassessed by marker (no requirement to submit again)
   a. normalise supplied forms to 3NF, one form at a time UNF->1NF->2NF->3NF
   b. carry out attribute synthesis on resultant set of 3NF relations to obtain one final set of 3NF relations
   c. map your 1A conceptual model to relational logical model
   d. integrate your final set of 3NF relations from 2b above, ensure attribute names in your normalisation are consistent with the names used in your logical model
   e. check model for no insert/update/delete anomalies
   f. check model for surrogate key requirement (if added ensure unique index on natural key created)
   g. generate physical (relational) model and from this generate the schema file, add appropriate details to the schema file (see week 6 tutorial)
   h. run the schema file and ensure no error (if there are any go back and fix logical model and repeat step 2g until removed)

Q4. What effect will the normalisation result have on the logical model?



3NF:

TRAINING (train_code, train_desc, train_active_mnths)

DRONE_TYPE (dt_code, dt_model, dt_manuf, train_code)

TRAINER (trainer_id, trainer_rego, trainer_fname, trainer_lname, trainer_category)

TRAINING_COURSE (train_code, traincourse_date, trainer_id)

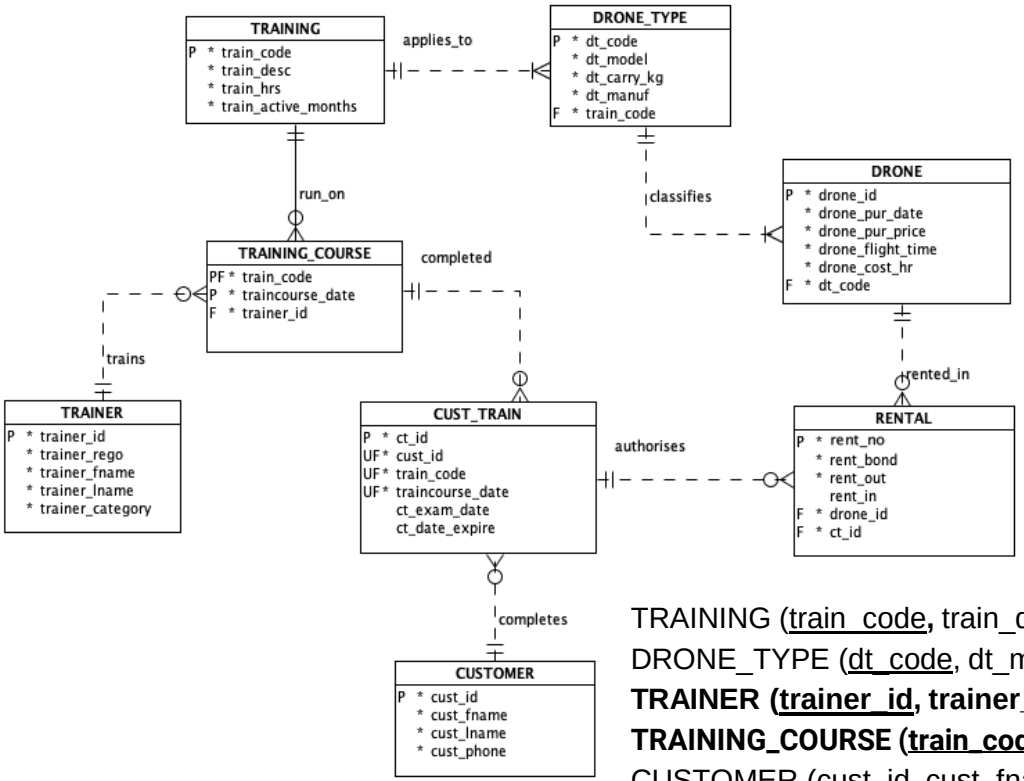
CUSTOMER (cust_id, cust_fname, cust_lname)

CUST_TRAINING (train_code, traincourse_date, cust_id, ct_exam_date, ct_date_expire)

A. Change the attributes of TRAINING
B. Change the attributes of CUSTOMER
C. Change the PK in TRAINING
D. Add a new relation TRAINER
E. Add a new relation CUST_TRAINING

Multiple answers possible

ALSO in your group list the details of the changes which will need to be made to this model based on your answer/s



Q5. Name an attribute on this model which exhibits insert/update/delete anomalies

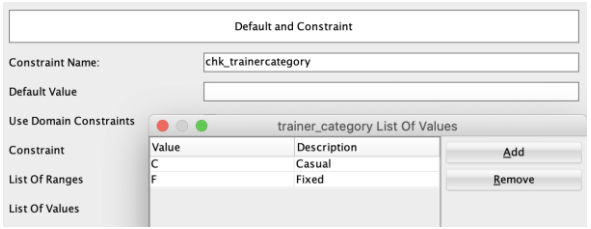
- **Free text (word cloud) response**

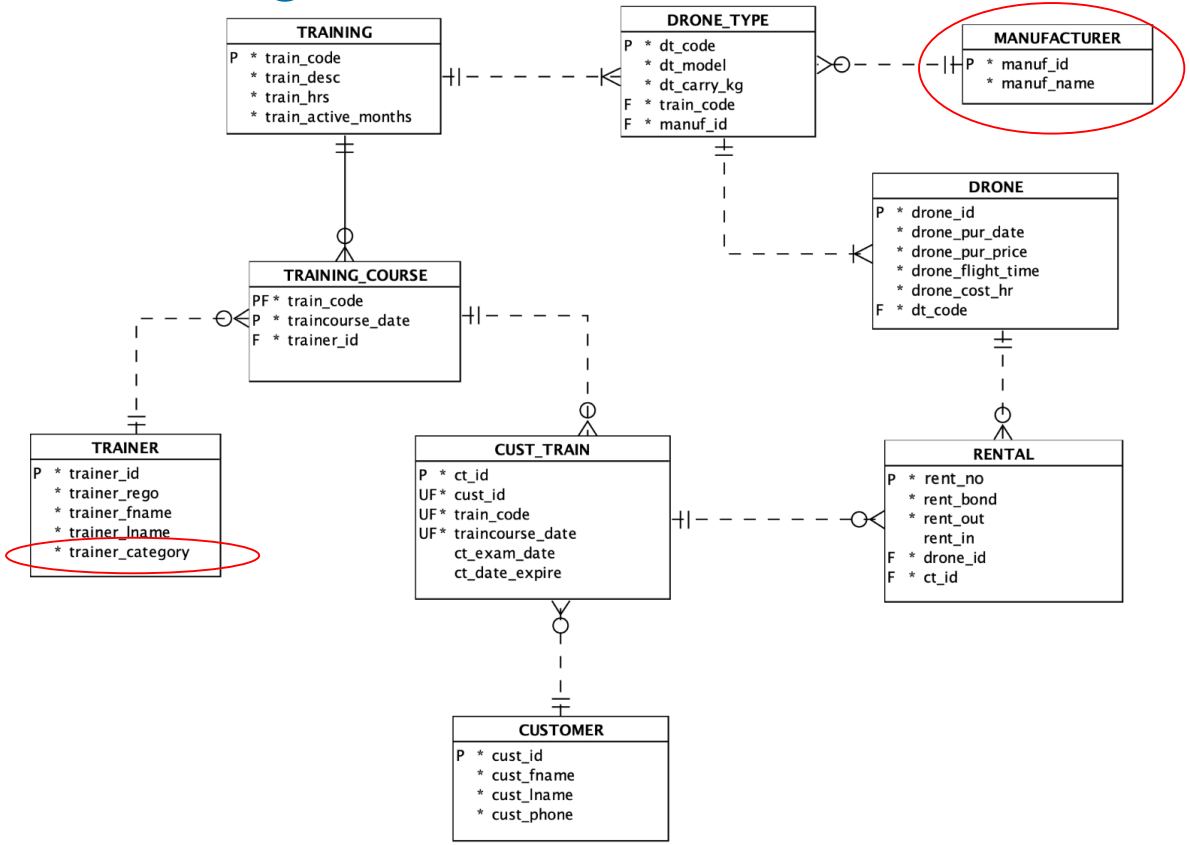
TRAINING (train_code**,** train_desc, **train_active_mnths**)
DRONE_TYPE (dt_code, dt_model, **dt_manuf**, train_code)
TRAINER (trainer_id, trainer_rego, trainer_fname, trainer_lname, trainer_category)
TRAINING_COURSE (train_code, traincourse_date, trainer_id)
CUSTOMER (cust_id, cust_fname, cust_lname)
CUST_TRAINING (train_code**, traincourse_date,** cust_id, ct_exam_date, ct_date_expire)

Controlling Values

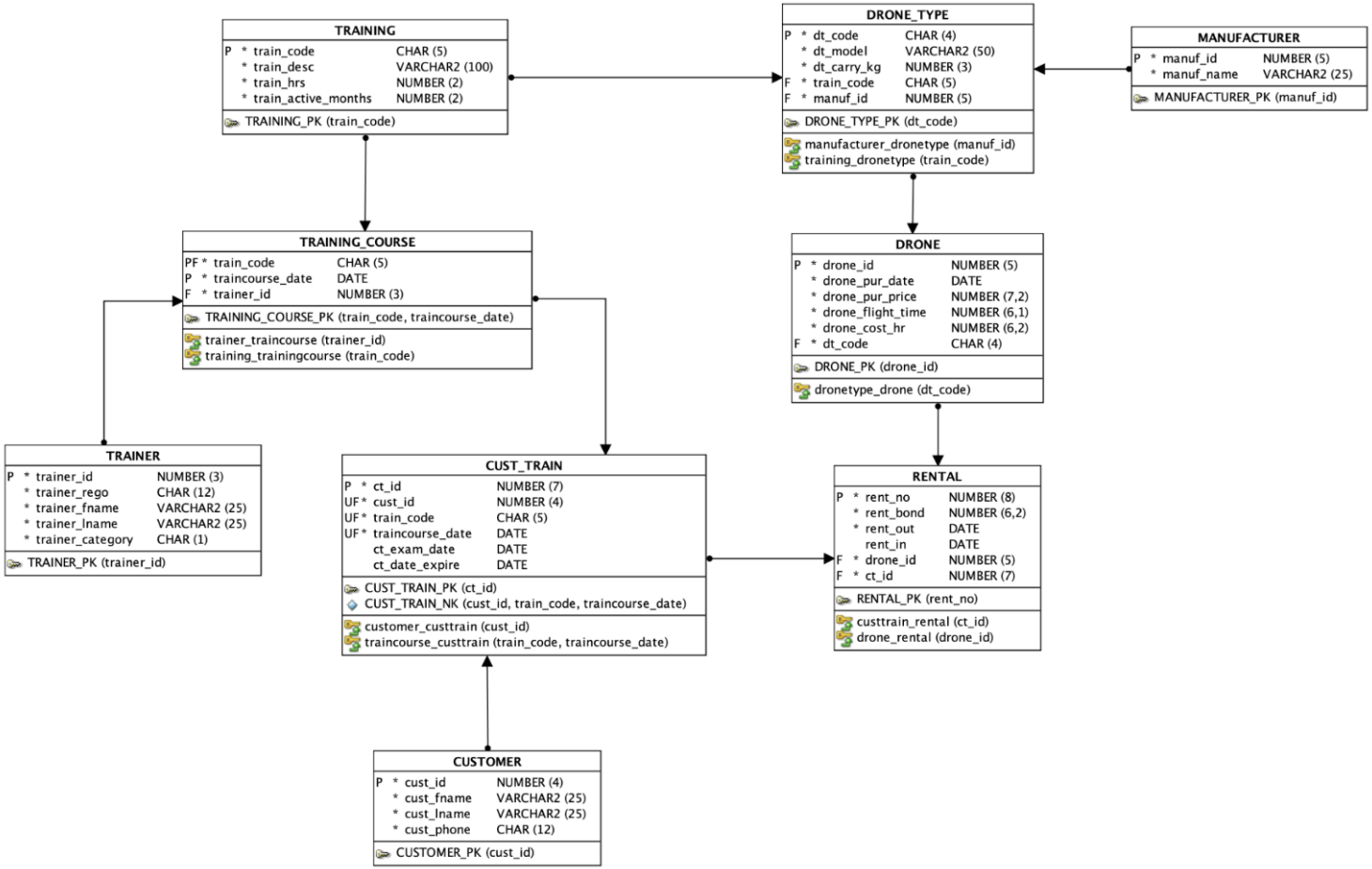
lookup table
- easily extended

check constraint
- small unchanging set => codify



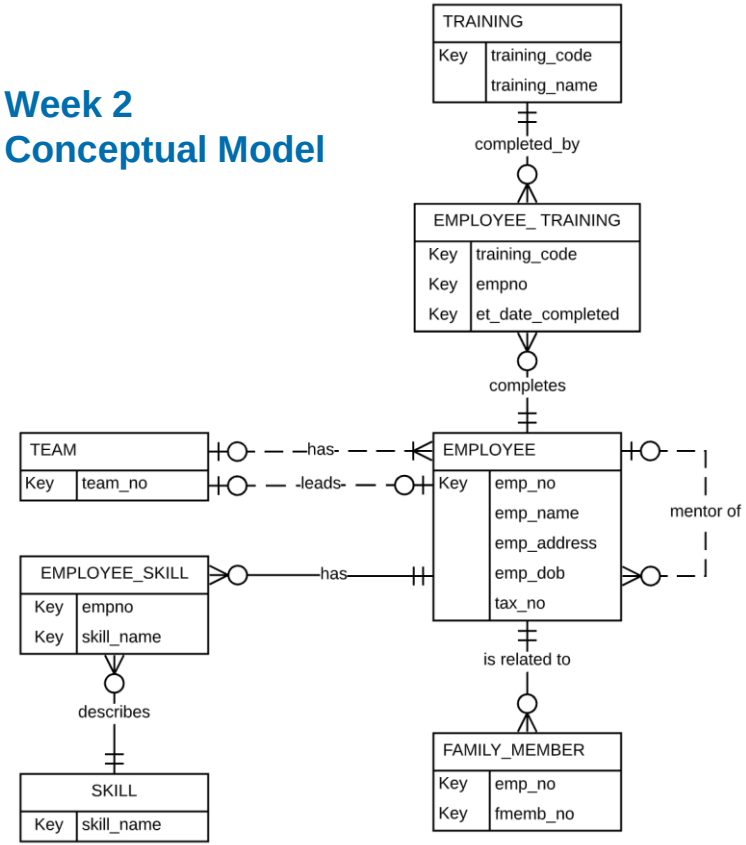


Relational (Physical) Model



Monash Software Conceptual Model

Week 2 Conceptual Model



Week 4 Normalisation

3NF

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode)

EMP_PHONE (emp_no, phone_no, phone_type)

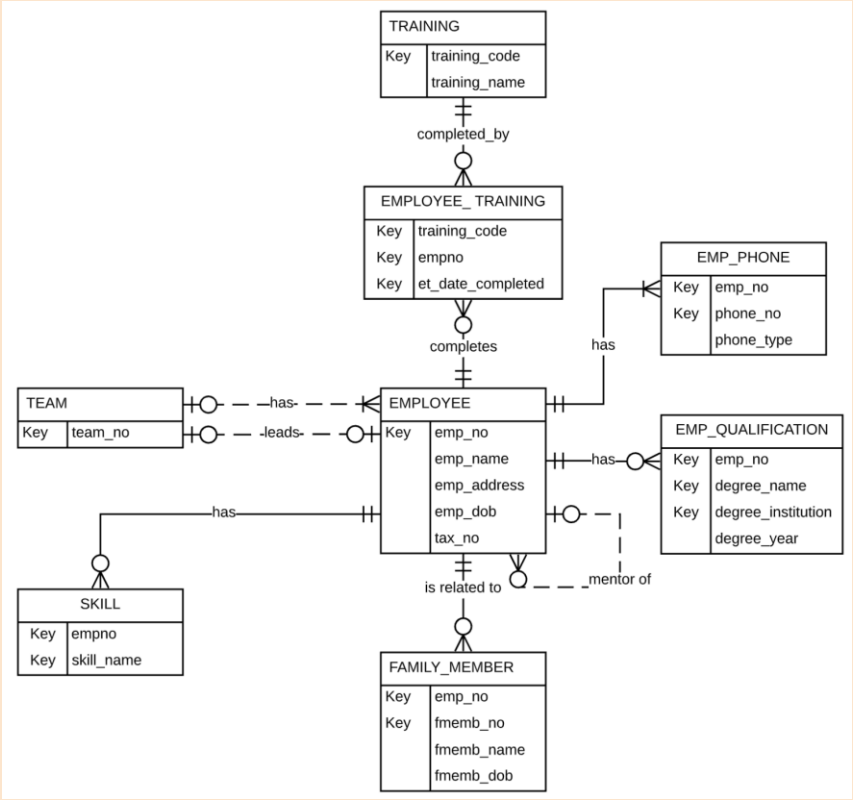
EMP_QUALIFICATION (emp_no, degree_name, degree_institution, degree_year)

FAMILY_MEMBER (emp_no, fmemb_no, fmemb_name, fmemb_dob)

EMPLOYEE_SKILL (emp_no, skill_name)

Post Forum Task

Convert this model to a logical model and then integrate the normalisation from the Week 4 Forum Session



Why participate in ESSFS?

ESSFS is easy and **only takes 5 mins**.

ESSFS helps us **improve your experience** while you are still enrolled in the unit.

ESSFS could win you a prize. Stand a chance to **win prizes** including an Ipad Mini, Airpods and Grab vouchers.

Take the survey,
get your voice heard.

MONASH University
MALAYSIA

WHAT ARE YOU WAITING FOR?

Get your voice heard.

OR

Visit

http://monash.ureview.org

MONASH University
MALAYSIA