MONASH University

MONASH
INFORMATION
TECHNOLOGY

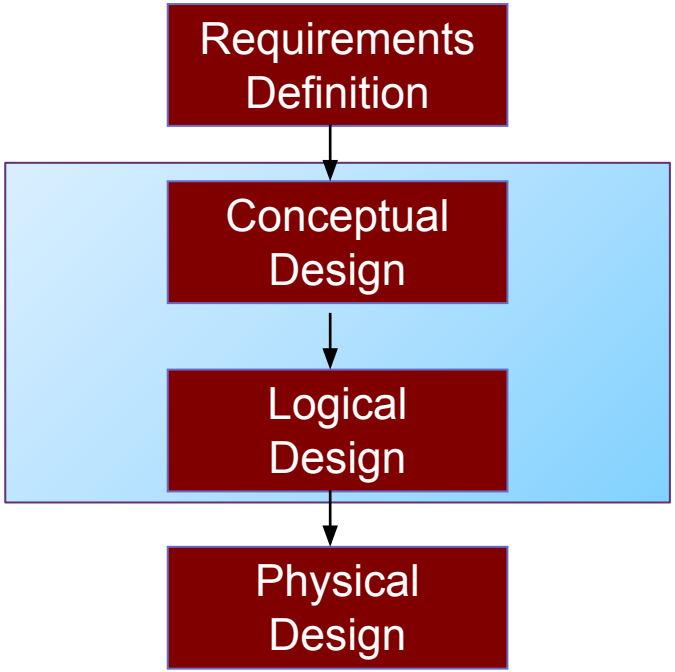
FIT2094 Databases

Week 2 - Conceptual Modelling

Please obtain a copy of the Drone case study for this workshop from the week 2 block on Moodle under "Workshop Resources" header

Workshop S1 2022

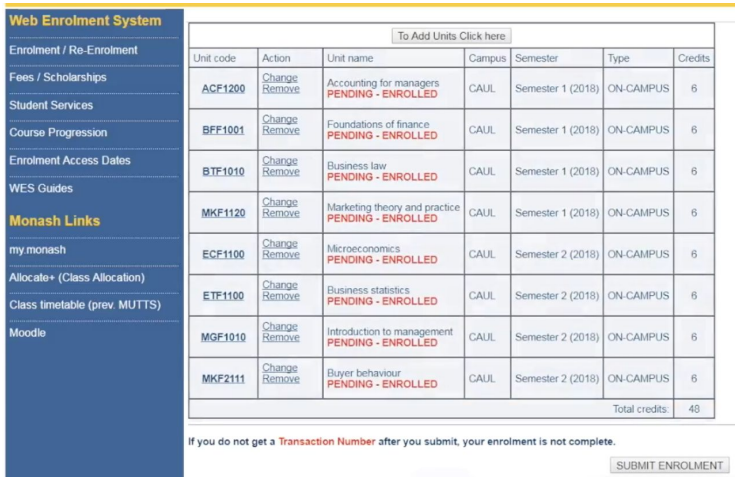
The Database Design Life Cycle



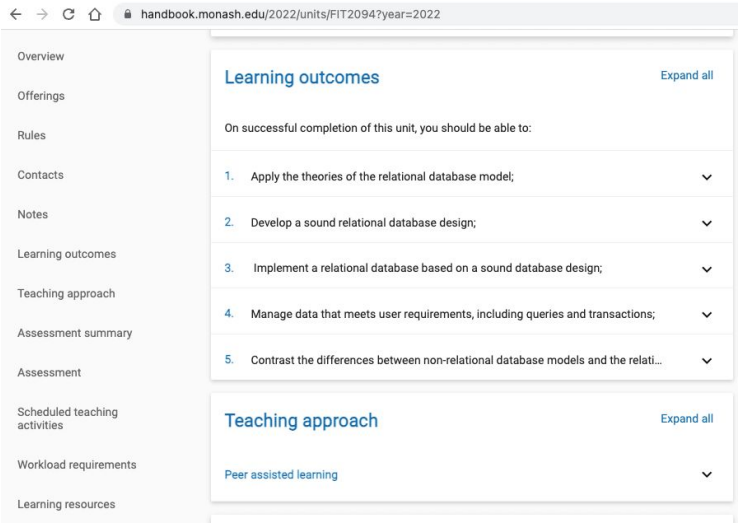
Requirements Definition

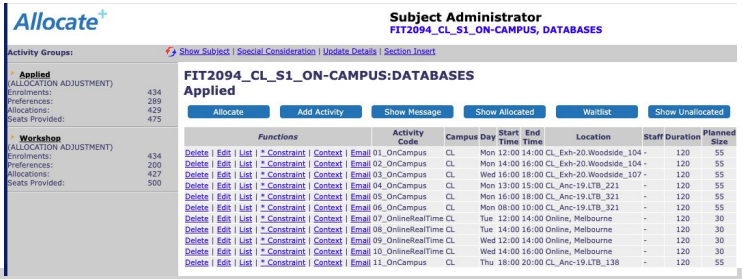
- Identify and analyse user views.
- A 'user view' may be a report to be produced or a particular type of transaction that should be supported.
- Corresponds to the external level of the ANSI/SPARC architecture.
- Output is a statement of specifications which describes the user views' particular requirements and constraints.

Different views of the underlying data



Student





Admin

Staff & Student

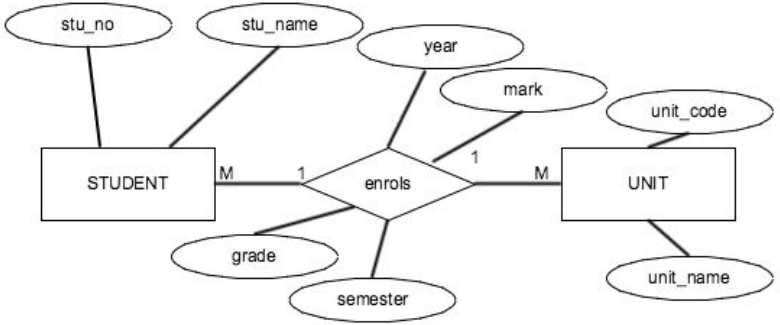
ER Modeling

- ER (Entity-Relationship) model developed by Peter Chen in 1976 to aid database design.
- used for conceptual model (ERD).
- ER diagrams give a visual indication of the design.
- Basic components:
  - Entity
  - Attribute
  - Relationship

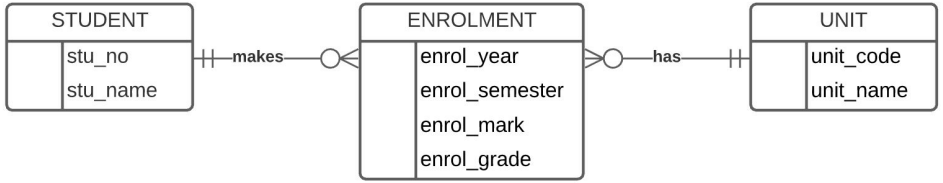
Conceptual Design

- Develop the enterprise data model.
- Corresponds to the conceptual level of the ANSI/SPARC architecture.
- *Independent of all physical implementation considerations (the type of database to be used)*.
- Various design methodologies may be employed such as UML, ER (Entity-Relationship) Modelling and Semantic Modelling.
- ER consists of ENTITIES and RELATIONSHIPS between entities
  - An ENTITY will have attributes (things we wish to record), one or more of which will identify an entity instance (called the KEY)

ERD - Notation



Chen



Information Engineering/James Martin/Crows foot

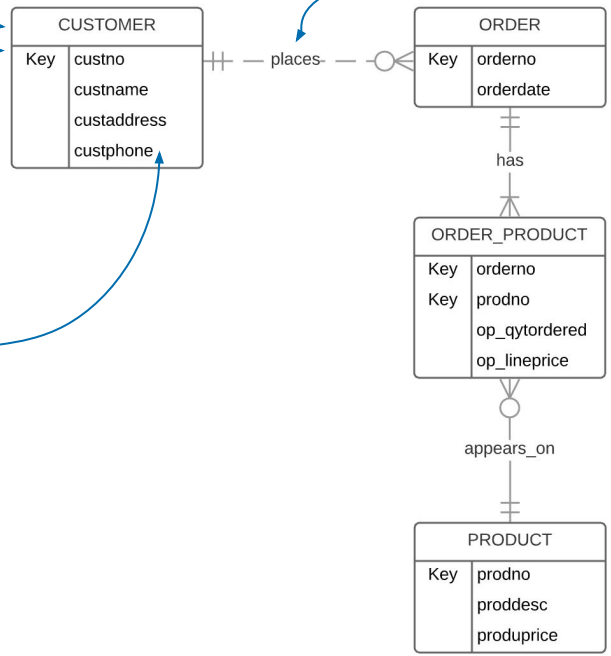
**** This is what we will be using***

Conceptual Level (ER Model)

ENTITY
Collection of "Customer(s)"

KEY ATTRIBUTE(S)
Instance identifier

NON KEY ATTRIBUTE
Other non-key attributes



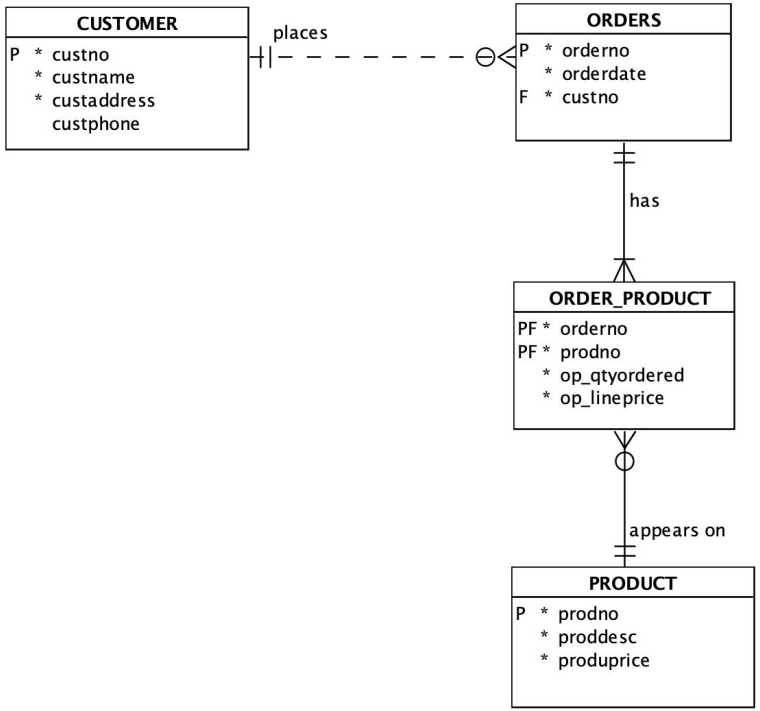
RELATIONSHIP
*Connects entities - on a conceptual model this is the **ONLY** manner in which entities are connected*

Assignment 1A

Logical Design

- Develop a data model which targets a particular database type (e.g. relational, hierarchical, network, object-oriented, noSQL).
- Independent of any implementation details which are specific to any particular vendors DBMS package.
- Normalisation technique (see week 4) is used to test the correctness of a relational logical model.

Logical Level (Logical Model - Relational)

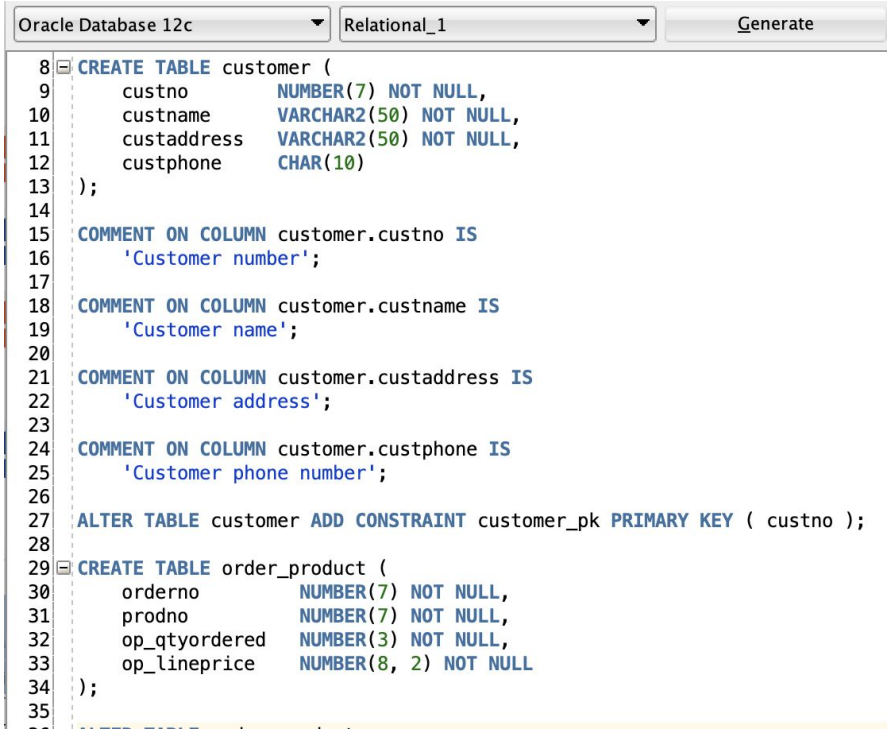


Assignment 1B

Physical Design

- Develop a strategy for the physical implementation of the logical data model.
- Choose appropriate storage structures, indexes, file organisations and access methods which will most efficiently support the user requirements (not part of unit).
- Physical design phase is dependent on the particular DBMS in use.
- ANSI/SPARC internal level.

Physical Level – Starting point



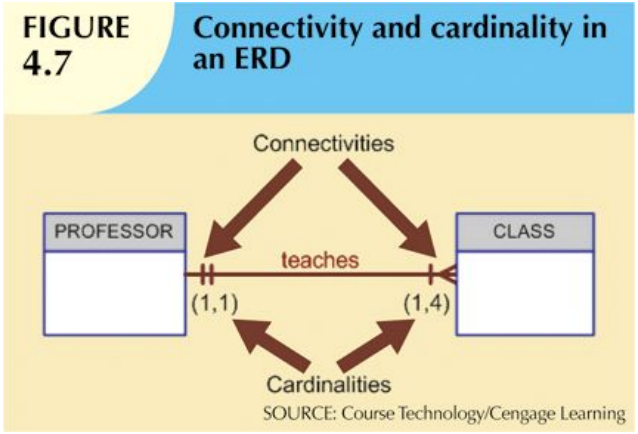

```
 8  CREATE TABLE customer (
 9      custno        NUMBER(7) NOT NULL,
10      custname      VARCHAR2(50) NOT NULL,
11      custaddress   VARCHAR2(50) NOT NULL,
12      custphone     CHAR(10)
13  );
14
15  COMMENT ON COLUMN customer.custno IS
16      'Customer number';
17
18  COMMENT ON COLUMN customer.custname IS
19      'Customer name';
20
21  COMMENT ON COLUMN customer.custaddress IS
22      'Customer address';
23
24  COMMENT ON COLUMN customer.custphone IS
25      'Customer phone number';
26
27  ALTER TABLE customer ADD CONSTRAINT customer_pk PRIMARY KEY ( custno );
28
29  CREATE TABLE order_product (
30      orderno         NUMBER(7) NOT NULL,
31      prodno          NUMBER(7) NOT NULL,
32      op_qtyordered   NUMBER(3) NOT NULL,
33      op_lineprice    NUMBER(8, 2) NOT NULL
34  );
35
```

The database *schema*

Important rule for Conceptual Modelling

- ***All that is described in the brief has been included and all that has been included was described in the brief***
  - Every entity, attribute and relationship described in the brief **has been included**, and
  - **Must not add** entities, attributes and relationships which are not included as part of the brief, and
- In a real life scenario if there are concerns about features of the brief, discuss with client
  - For assignments:
    - your client will be the ed forum
    - may make assumptions provided they do not violate this rule

CONNECTIVITY/CARDINALITY



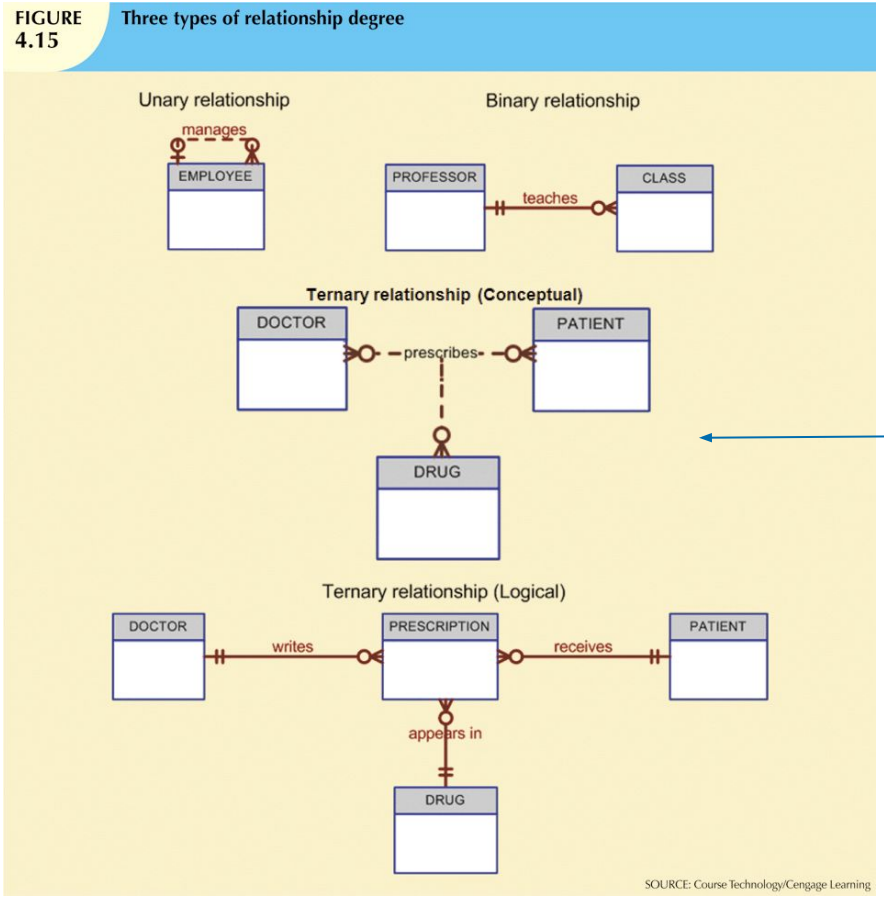
CONNECTIVITY

one to one

one to many

many to many

In general for Crows Foot notation specific cardinalities are not shown as above eg. (1,4), instead cardinality is depicted via min and max using standard symbols (Inside symbol = min, outside symbol = max)

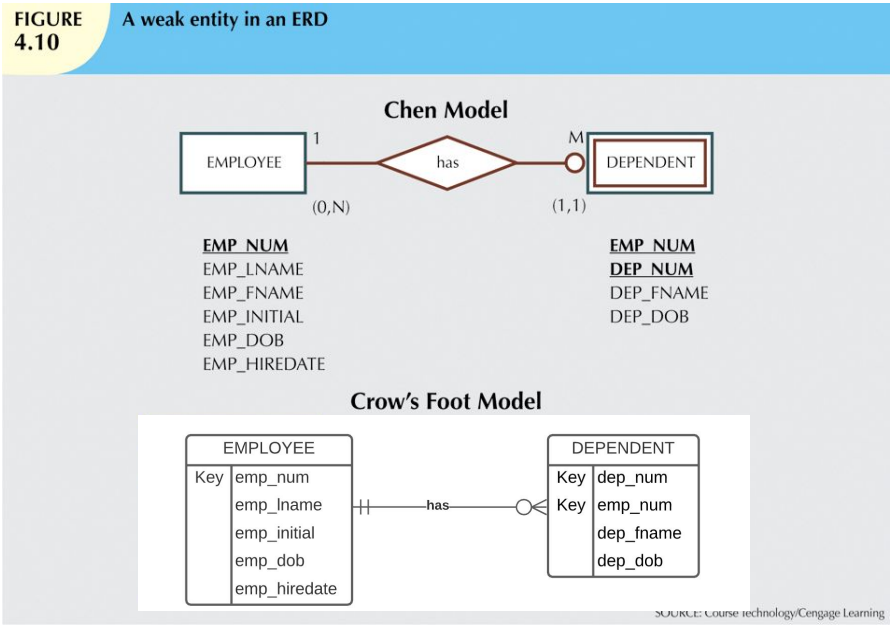


Note this is not an acceptable form of a conceptual model in Crow's Foot notation (*relationship lines cannot join*)

Weak vs Strong Entity

- Strong entity
  - Has a key which may be defined without reference to other entities.
  - For example EMPLOYEE entity.
- Weak entity
  - Has a key which requires the existence of one or more other entities.
  - For example FAMILY entity - need to include the key of employee to create a suitable key for family
- Database designer often determines whether an entity can be described as weak based on business rules
  - customer pays monthly account
    - Key: cust_no, date_paid, or
    - Key: payment_no (surrogate? – not at conceptual level)

Weak vs Strong Entity



Note the Crow's Foot model shown here has been modified from the text version

Identifying vs Non-Identifying Relationship

- **Identifying**
- Identifier of A is part of identifier of B.

- Shown with solid line
- ENROLMENT - STUDENT
  Enrolment key includes student id, which is an identifier of student.

- **Non-identifying**
- Identifier of A is NOT part of identifier of B.

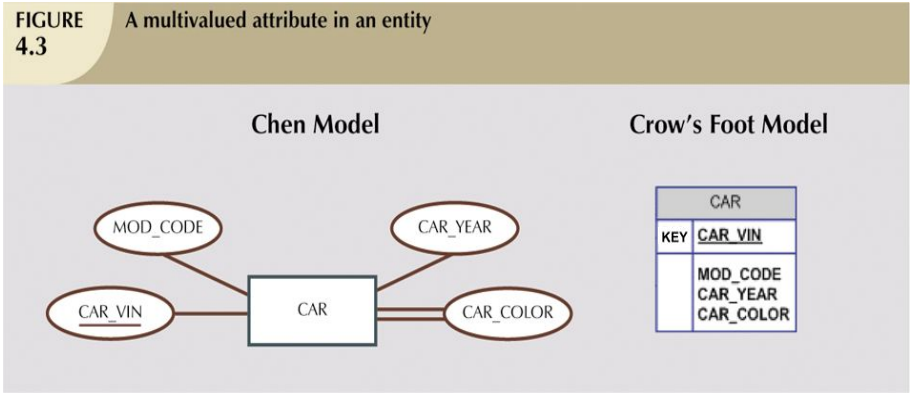
- Shown with broken line
- Department no (identifier of department) is not part of Employee's identifier.

Types of Attributes

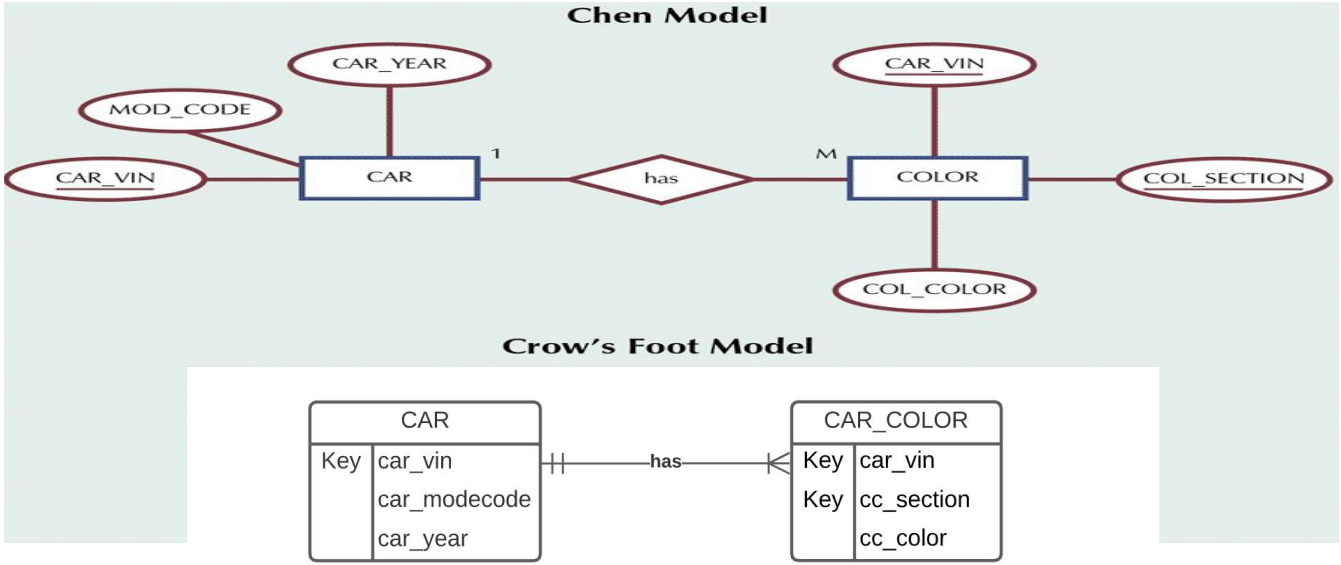
- Simple
  - Cannot be subdivided
  - Age, sex, marital status
- Composite
  - Can be subdivided into additional attributes
  - Address into street, city, zip
- Single-valued
  - Can have only a single value
  - Person has one social security number
- Multi-valued
  - Can have many values
  - Person may have several college degrees
- Derived
  - Can be derived with algorithm
  - Age can be derived from date of birth
- Attribute classification is driven by Client requirements
  - Phone Number?

Multivalued Attribute

- An attribute that has a list of values.
- For example:
  - Car colour may consist of body colour, trim colour, bumper colour.
- Crow's foot notation does not support multivalued attributes. Values are listed as a separate attribute.



Resolving Multivalued Attributes



Note the Crow's Foot model shown here has been modified from the text version

Associative (or Composite) Entity