



MONASH
University

MONASH
INFORMATION
TECHNOLOGY

FIT2094 Databases

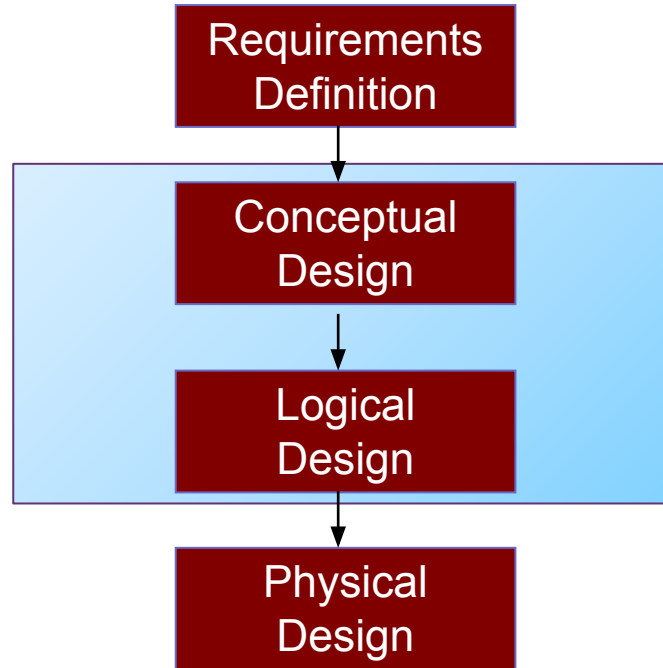
Week 2 - Conceptual Modelling

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

Web Enrolment System						
Enrolment / Re-Enrolment	To Add Units Click here					
Fees / Scholarships	Unit code	Action	Unit name	Campus	Semester	Credits
Student Services	ACF1200	Change Remove	Accounting for managers PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS 6
Course Progression	BFF1001	Change Remove	Foundations of finance PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS 6
Enrolment Access Dates	BTF1010	Change Remove	Business law PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS 6
WES Guides	MKF1120	Change Remove	Marketing theory and practice PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS 6
Monash Links	ECF1100	Change Remove	Microeconomics PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS 6
my.monash	ETF1100	Change Remove	Business statistics PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS 6
Allocate+ (Class Allocation)	MGF1010	Change Remove	Introduction to management PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS 6
Class timetable (prev. MUTTS)	MKF2111	Change Remove	Buyer behaviour PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS 6
Moodle	Total credits:					48

If you do not get a **Transaction Number** after you submit, your enrolment is not complete.

SUBMIT ENROLMENT

handbook.monash.edu/2022/units/FIT2094?year=2022	
Overview	Learning outcomes Expand all
Offerings	
Rules	On successful completion of this unit, you should be able to:
Contacts	1. Apply the theories of the relational database model; ▼
Notes	2. Develop a sound relational database design; ▼
Learning outcomes	3. Implement a relational database based on a sound database design; ▼
Teaching approach	4. Manage data that meets user requirements, including queries and transactions; ▼
Assessment summary	5. Contrast the differences between non-relational database models and the relational database model; ▼
Assessment	
Scheduled teaching activities	Teaching approach Expand all
Workload requirements	Peer assisted learning ▼
Learning resources	

Student

Allocate⁺

Activity Groups:

Applied

(ALLOCATION ADJUSTMENT)

Enrolments: 434

Preferences: 269

Allocations: 429

Seats Provided: 475

Workshop

(ALLOCATION ADJUSTMENT)

Enrolments: 434

Preferences: 200

Allocations: 427

Seats Provided: 500

Show Subject

Special Consideration

Update Details

Section Insert

FIT2094_CL_S1_ON-CAMPUS:DATABASES

Applied

Allocate

Add Activity

Show Message

Show Allocated

Waitlist

Show Unallocated

Functions	Activity Code	Campus	Day	Start Time	End Time	Location	Staff	Duration	Planned Size
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 01, OnCampus	CL	Mon	12:00	14:00	CL_Esh-20, Woodside_104	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 02, OnCampus	CL	Mon	14:00	16:00	CL_Esh-20, Woodside_104	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 03, OnCampus	CL	Wed	16:00	18:00	CL_Esh-20, Woodside_107	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 04, OnCampus	CL	Mon	13:00	15:00	CL_Anc-19, LTB_221	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 05, OnCampus	CL	Mon	16:00	18:00	CL_Anc-19, LTB_321	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 06, OnCampus	CL	Mon	08:00	10:00	CL_Anc-19, LTB_321	-	120	55
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 07, OnlineRealTime CL	CL	Tue	12:00	14:00	Online, Melbourne	-	120	30
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 08, OnlineRealTime CL	CL	Tue	14:00	16:00	Online, Melbourne	-	120	30
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 09, OnlineRealTime CL	CL	Wed	12:00	14:00	Online, Melbourne	-	120	30
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 10, OnlineRealTime CL	CL	Wed	14:00	16:00	Online, Melbourne	-	120	30
<div>Delete</div> <div>Edit</div> <div>List</div> <div>+ Constraint</div> <div>Context</div>	Email 11, OnCampus	CL	Thu	18:00	20:00	CL_Anc-19, LTB_138	-	120	55

Staff & Student

Admin



MONASH University

ER Modeling

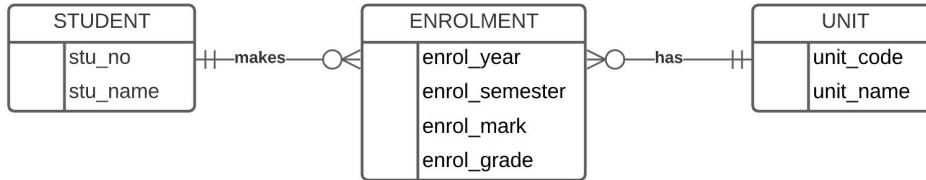
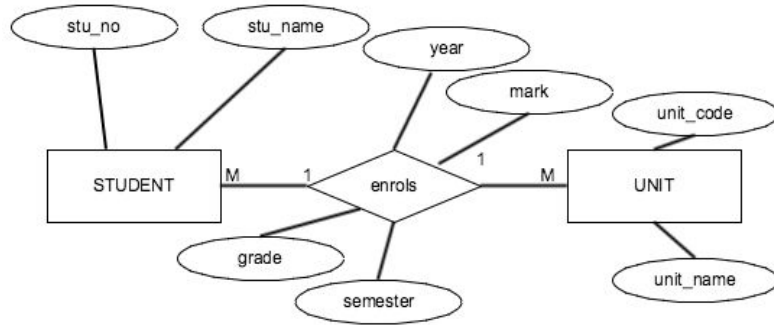
- ER (Entity-Relationship) model developed by Peter Chen in 1976 to aid database design.
- Used for conceptual model (ERD).
- ER diagrams give a visual indication of the design.
- Basic components:
 - Entity
 - Attribute
 - Relationship



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)"

RELATIONSHIP

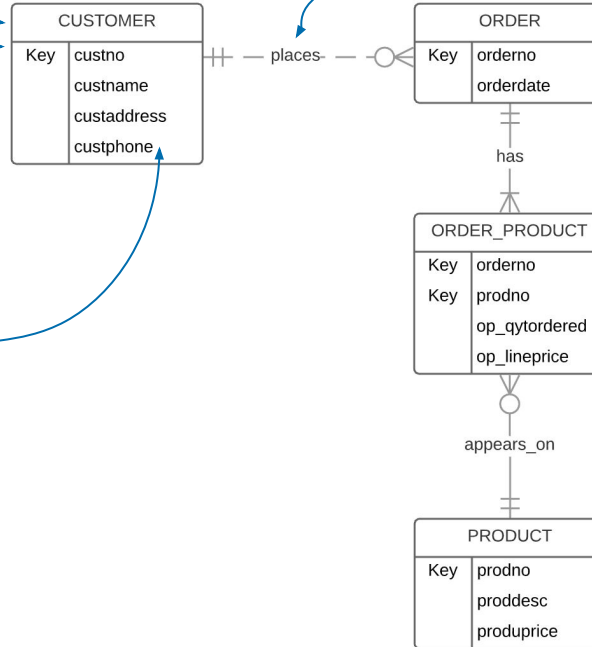
*Connects entities - on a conceptual model this is the **ONLY** manner in which entities are connected*

KEY ATTRIBUTE(S)

Instance identifier

NON KEY ATTRIBUTE

Other non-key attributes

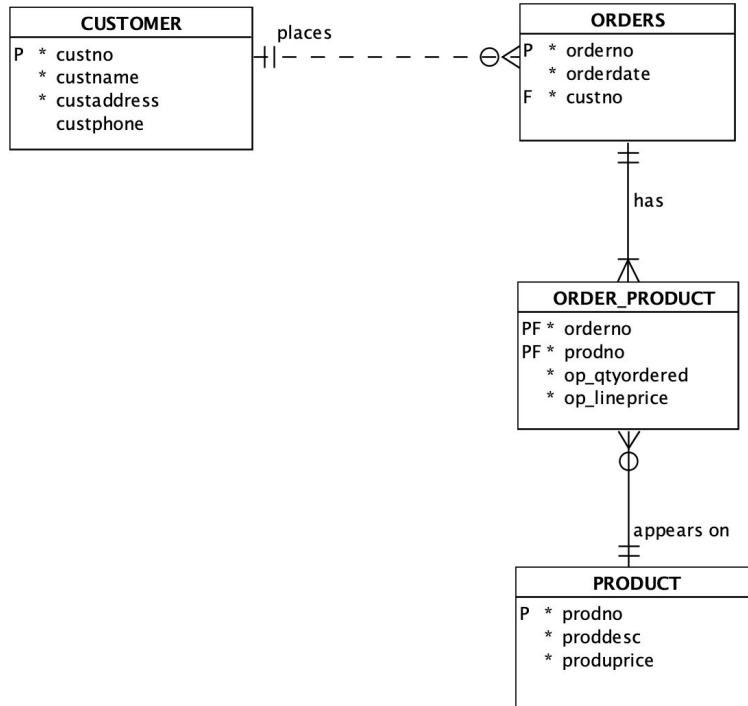


Assignment 1A

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

```
Oracle Database 12c  Relational_1  Generate

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*

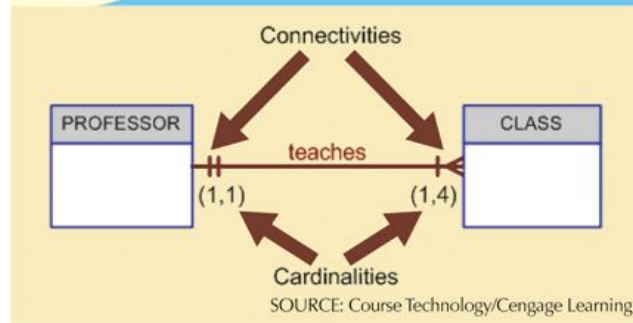
Important rule for Conceptual Modelling

- ***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

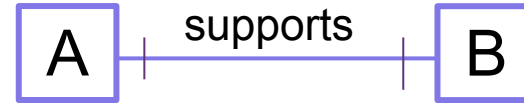
FIGURE 4.7

Connectivity and cardinality in an ERD

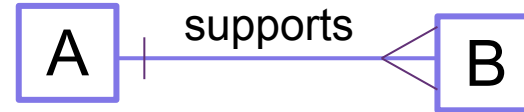


CONNECTIVITY

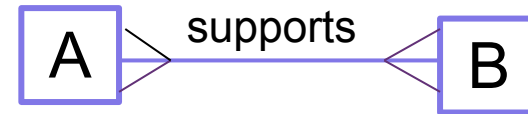
one to one



one to many



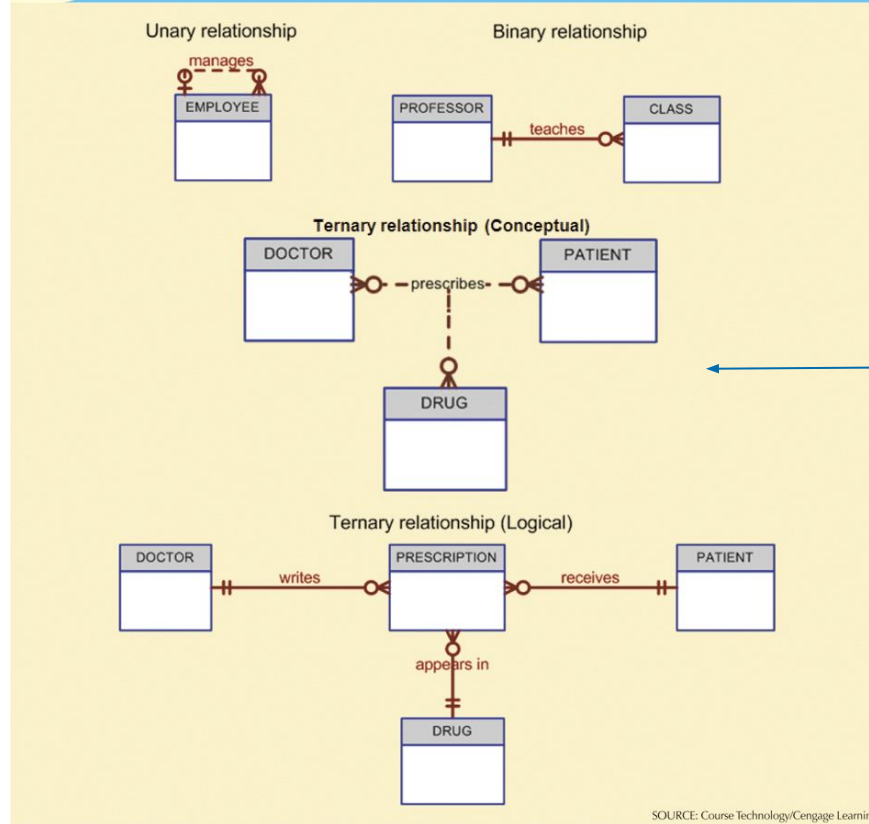
many to many



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)

FIGURE
4.15

Three types of relationship degree



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

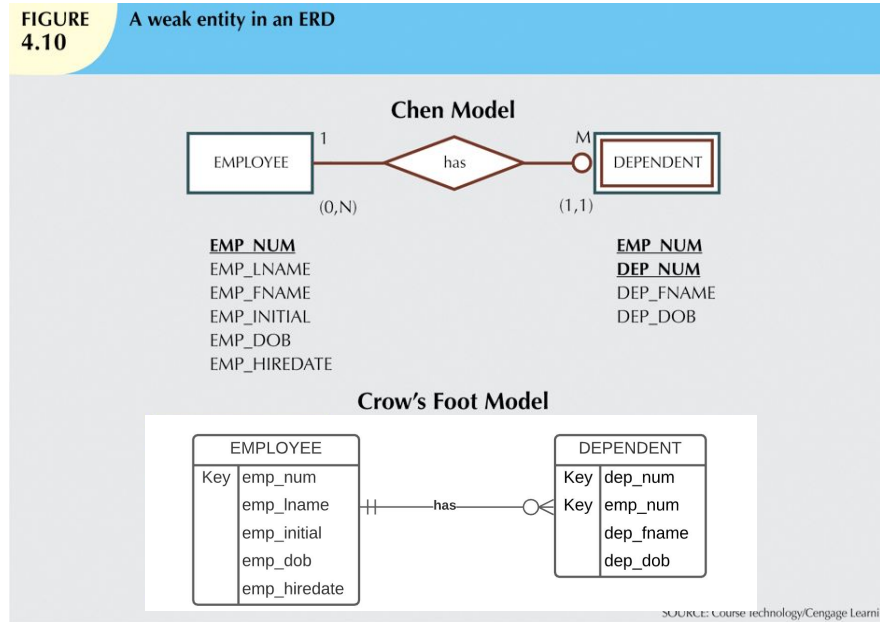
Weak vs Strong Entity

- 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

FIGURE 4.10

A weak entity in an ERD



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.



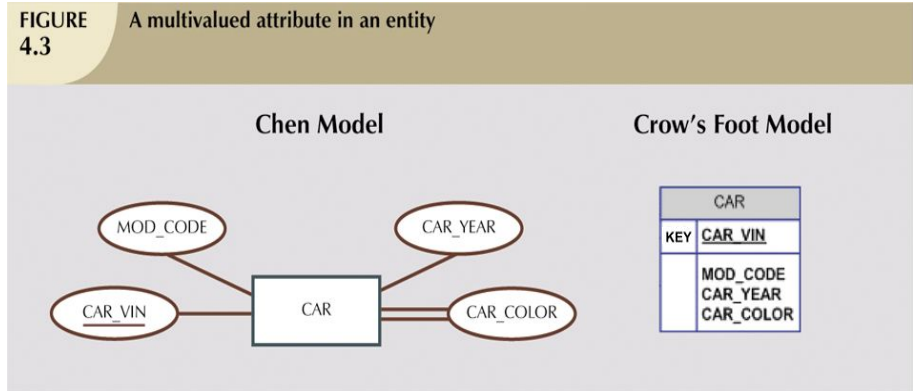
- Shown with broken line
- Department no (identifier of department) is not part of Employee's identifier.

Types of Attributes

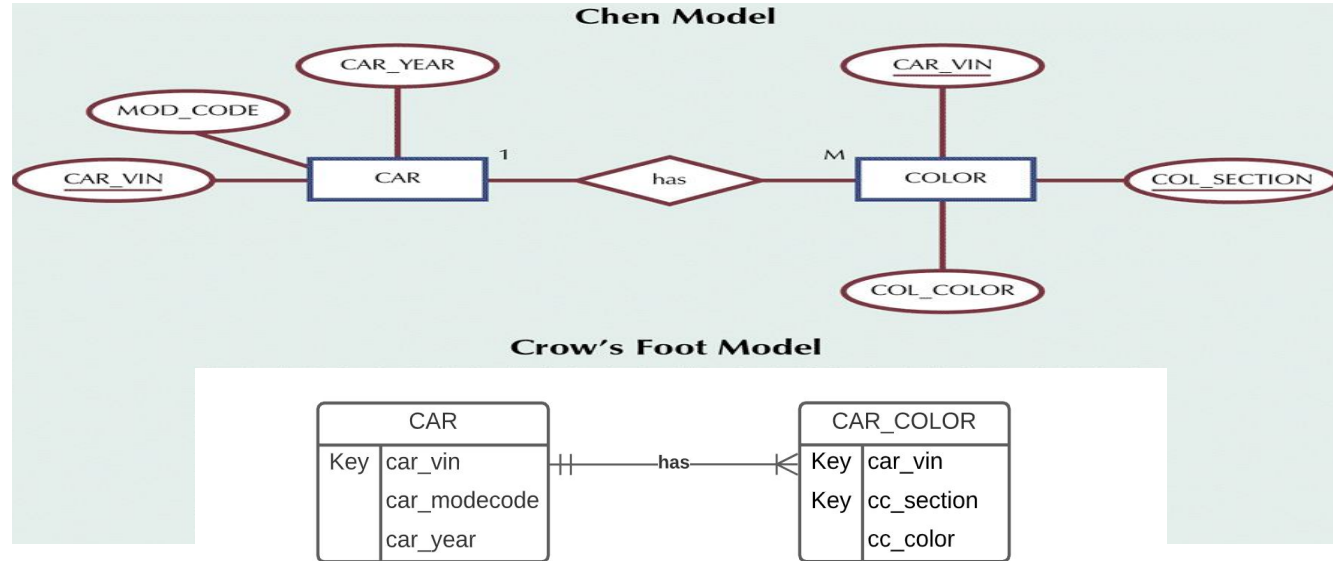
- Simple
 - Cannot be subdivided
 - Age, sex, marital status
 - Composite
 - Can be subdivided into additional attributes
 - Address into street, city, zip
 - Single-valued
 - Can have only a single value
 - Person has one social security number
- Multi-valued
 - Can have many values
 - Person may have 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

