

BT5110 Data Management and Warehousing

Tutorial 4: Entity-relationship Modelling

Mark Meng Huasong

School of Computing
National University of Singapore

13 - 17 Sep, 2021



All the materials within presentation slides are protected by copyrights.
It is forbidden by NUS to upload these materials to the Internet.

Correction of Q 2(c) of Tutorial 3

```
1  SELECT  s.department, s.name, count(*)
2  FROM    student s, loan l
3  WHERE   l.owner = s.email
4  -- The line below is buggy, but tolerated by Postgres.
5  -- GROUP BY s.department, s.email
6  -- The output is correct because s.name is not the primary key
7  --   so it does affect the group by result. The correct code is
8  --   shown below:
9  GROUP BY s.department, s.email, s.name
10 HAVING count(*) >= ALL
11      (SELECT count(*)
12       FROM student s1, loan l1
13       WHERE l1.owner = s1.email
14       AND s.department = s1.department
15       GROUP BY s1.name);
```

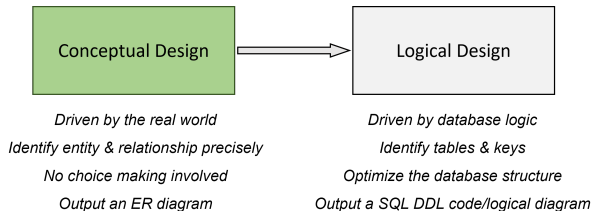
Quick Recap: End of Last Tutorial

What we have done in the last week:

- (1) Write simple queries with aggregation;
- (2) Write nested queries;
- (3) Make use of double negation and left outer joining to achieve complex queries.

(This tutorial does not use the existing book loan database)

Question 1



Question 1 (a-d)

(a) Identify the entity sets. Justify your choice by quoting the sentences in the text that support it.

Solution: There are three entity sets: member, bottle and wine.

(b) Identify the relationship sets and the entity sets that they associate. Justify your choice by quoting the sentences in the text that support it.

Solution: There are two relationship sets: taste and contain. The relationship set taste links the entity set member with the entity set bottle. The relationship set contain links the entity set wine with the entity set bottle.

Question 1 (a-d) Cont.

(c) For each entity set and relationship set identify its attributes. Justify your choice by quoting the sentences in the text that support it.

Solution: The relationship set taste has two attributes: date and rating. The relationship set contain has **no** attribute.

(d) For each entity set, identify its keys.

Solution: For the entity set member, there are three attributes: name, address and card number. The card number is the primary key.

For the entity set wine, there are 7 attributes. Three attributes appellation, name and vintage are (composite) keys; four remaining attributes are country of origin, degree of alcohol, certification and bottled details.

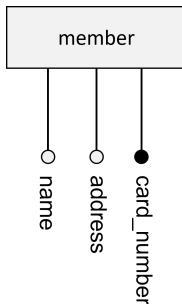
For the entity set bottle, there are **3 or 5** (let me explain later) attributes, including in cellar, bottle number and three foreign keys to identify a wine instance. The bottle number is the primary key, and the primary key(s) of the entity set wine is/are the referencing foreign keys. (The Boolean attribute can be derived by in cellar, therefore is not necessary.)

Question 1 (a-d) Cont.

Let's find the supporting text from the background information provided.

Entity *member* (Paragraph 1)

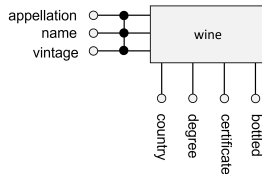
The organisation is big enough so that there could be several members with the same name. A card with a unique number is issued to **identify** each drinker. The contact address of each member is also recorded for the mailing of announcements and calls for meetings.



Question 1 (a-d) Cont.

Entity *wine* (Paragraph 3)

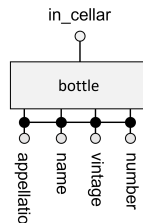
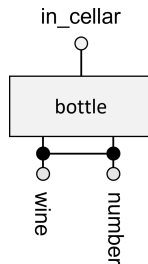
Each wine is **identified** by its name (Parade D'Amour), appellation (Bordeaux) and vintage (1990). Other information of interest about the wine is the degree of alcohol (11.5), where and by whom it has been bottled (Mis en Bouteille par Amblard-Larolphe Negociant-Eleveur a Saint Andre Cubzac (Gironde) - France), the certification of its appellation if available (Appellation Bordeaux Controlée), and the country it comes from (produce of France).



Question 1 (a-d) Cont.

Entity *bottle* (Paragraph 4)

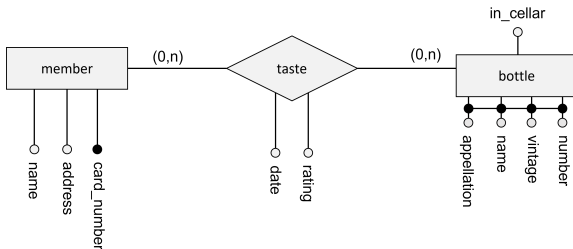
For each wine, the bottles in the wine cellar of VINO are numbered. For instance, the cellar has 20 bottles numbered 1 to 20 of a Semillon from 1996 named Rumbalara. ... The bottles are either available in the cellar or they have been tasted and emptied.



Question 1 (a-d) Cont.

Relationship *taste* (Paragraph 2)

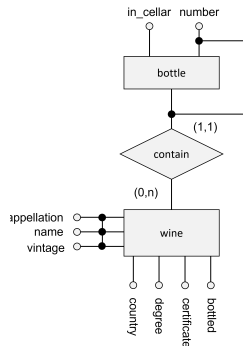
At most once a week, VINO organises a tasting **session**. At each session, the attending members taste several bottles. Each **member** records for each bottle his or her evaluation of the **quality** (very good, good, average, mediocre, bad, very bad) of each wine that she or he tastes. The evaluation may differ for the same wine from one drinker to another. Actual quality and therefore evaluation also varies from one to another **bottle** of a given wine. Every bottle that is opened during the tasting session is finished during that session.



Question 1 (a-d) Cont.

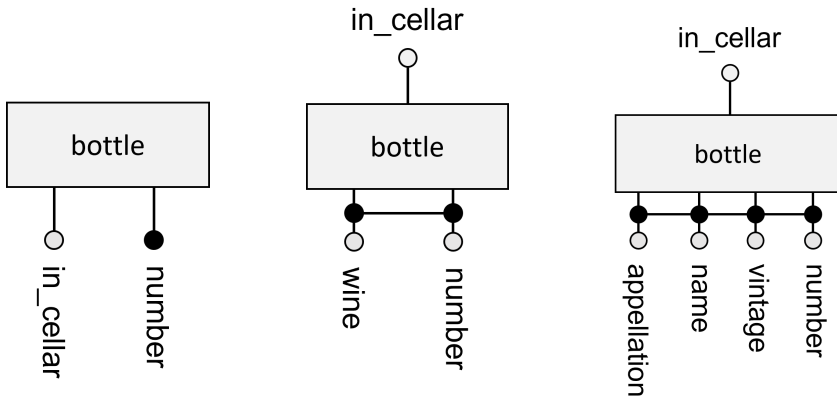
Relationship *contain* (Paragraph 4)

Generally, there are or have been several bottles of the same wine in the cellar...For documentation purposes VINO may also want to record wines for which it does not own bottles.



Question 1 (a-d) Cont.

How many attributes for the entity set “bottle”?



Question 1 (e)

For each entity set and each relationship set in which it participates, indicate the minimum and maximum participation constraints.

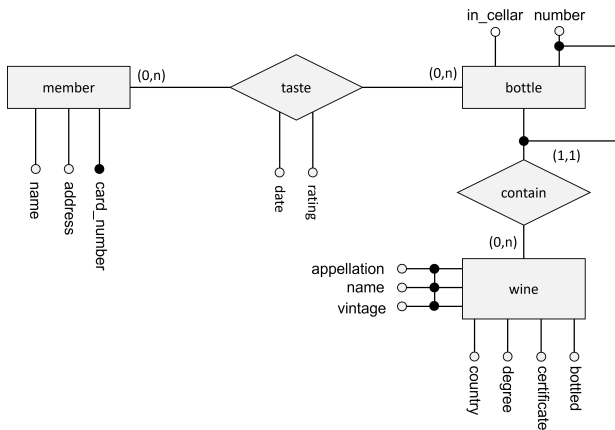
Solution: The relationship **taste** is an optional many-to-many relationship: both pairs of cardinality constraints are (0, n). (number of {minimum, maximum} outgoing links)

The relationship **contain** is a one-to-many relationship from wine to bottle. It is mandatory for the bottle entities and optional for the optional for the wine entities. The participation constraints for the entity set **wine** is (0, n). The participation constraints for the entity set **bottle** is (1, 1). It could not be otherwise since the entity set bottle is a weak entity (is weakly identified under the relationship set **contain** and the dominant entity set **wine**).

Question 1 (f)

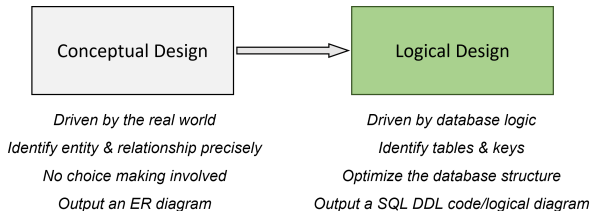
Draw the corresponding entity-relationship diagram with the key and participation constraints. Indicate in English the constraints that cannot be captured, if any.

Solution: There is no additional constraints.



Question 2

Translate your entity-relationship diagram into a relational schema. Give the SQL data description language (DDL) statements to create the schema. Declare the necessary integrity constraints. Indicate in English the constraints that cannot be captured, if any.



Question 2 Cont.

Solution:

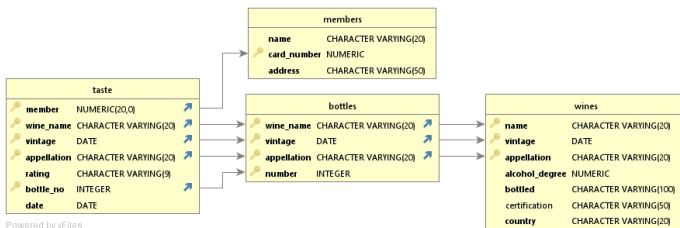
```
1  CREATE TABLE IF NOT EXISTS members (  
2      name VARCHAR(20) NOT NULL,  
3      card_number NUMERIC PRIMARY KEY,  
4      address VARCHAR(50) NOT NULL  
5  );  
6  
7  CREATE TABLE IF NOT EXISTS wines (  
8      name VARCHAR(20),  
9      vintage DATE,  
10     appellation VARCHAR(20),  
11     alcohol_degree NUMERIC NOT NULL,  
12     bottled VARCHAR(100) NOT NULL,  
13     certification VARCHAR(50),  
14     country VARCHAR(20) NOT NULL,  
15     PRIMARY KEY (name, vintage, appellation)  
16 );
```

Question 2 Cont.

```
17  -- CONTINUED WITH THE LAST SLIDE
18
19  -- Here we merge 'bottles' and 'contain' because 'bottles' is a weak entity.
20  CREATE TABLE IF NOT EXISTS bottles (
21      wine_name VARCHAR(20),
22      vintage DATE,
23      appellation VARCHAR(20),
24      number INTEGER,
25      -- Here we omit the attribute 'in_cellar' because it can be derived.
26      PRIMARY KEY (number, wine_name, vintage, appellation),
27      FOREIGN KEY (wine_name, vintage, appellation)
28      REFERENCES wines (name, vintage, appellation)
29  );
30
31  CREATE TABLE IF NOT EXISTS taste (
32      member NUMERIC(20),
33      wine_name VARCHAR(20),
34      vintage DATE,
35      appellation VARCHAR(20),
36      rating VARCHAR(9) NOT NULL,
37      bottle_no INTEGER,
38      date DATE NOT NULL,
39      PRIMARY KEY (member, bottle_no, wine_name, vintage, appellation),
40      FOREIGN KEY (member) REFERENCES members(card_number),
41      FOREIGN KEY (bottle_no, wine_name, vintage, appellation) REFERENCES
42      bottles(number, wine_name, vintage, appellation)
43  );
```

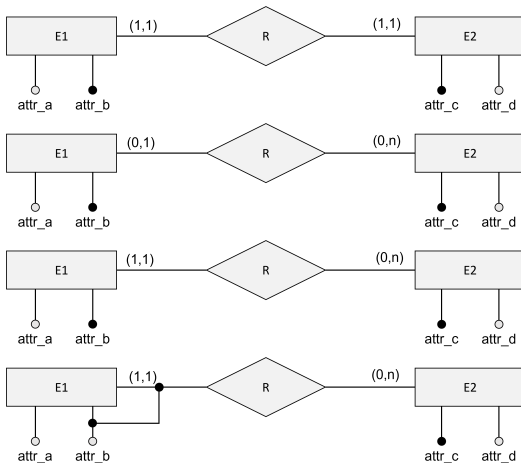
Question 2 Cont.

Logical diagram of tables created:

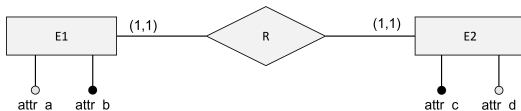


Extra Practice: Participation vs. Logical Design

Can you spot the difference among the 4 cases below? Can you explain what they look like in the logical design? Can you give an example from the real world for each case?



Extra Practice (Case 1)



E1 as **political_party**,
E2 as **secretary_general**,
R as **led_by**

People's Action	●	—	●	Lee Hsien Loong
Workers (S'pore)	●		●	Francis Yuen
Progress Singapore	●	✗	●	Pritam Singh
Reform (S'pore)	●	—	●	Kenneth Jeyaretnam

Table E1:

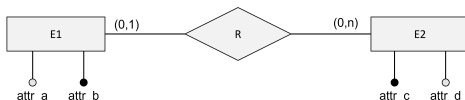
🔑 name,
secretary_general

or alternatively, we define

Table E2:

🔑 name
leading_party,

Extra Practice (Case 2)



E1 as **parliament_member**,
E2 as **political_party**,
R as **belongs_to**

Table E1:

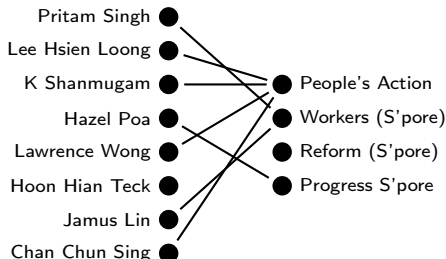
🔑 name

Table E2:

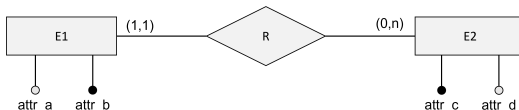
🔑 name

Table R:

🔑 → member_name (E1.name),
→ affiliation_party (E2.name)



Extra Practice (Case 3)



E1 as **government**,
E2 as **parliament_member**,
R as **head**

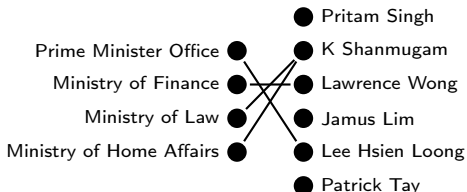


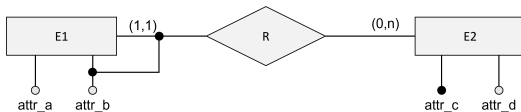
Table E1:

🔑 government_name
→ name (E2.name)

Table E2:

🔑 name

Extra Practice (Case 4)



E1 as **party**,
E2 as **country**,
R as **belongs_to**

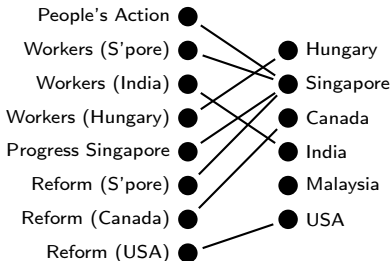


Table E1:

🔑 party_name
🔑 → country (E2.name)

Table E2:

🔑 name

For any further question, please feel free to email me:

huasong.meng@u.nus.edu

Cases in the extra practice are contributed by our students.

Copyright 2021 Mark H. Meng. All rights reserved.