

Handin 1

Thomas Vinther & Jens Kristian Refsgaard Nielsen

11. februar 2019

1 Modeling with the E-R and the Relational Model

After going through the text, we decide to make Members, bottles and wines entity sets, as these are the only written in non-cursive bold writing.

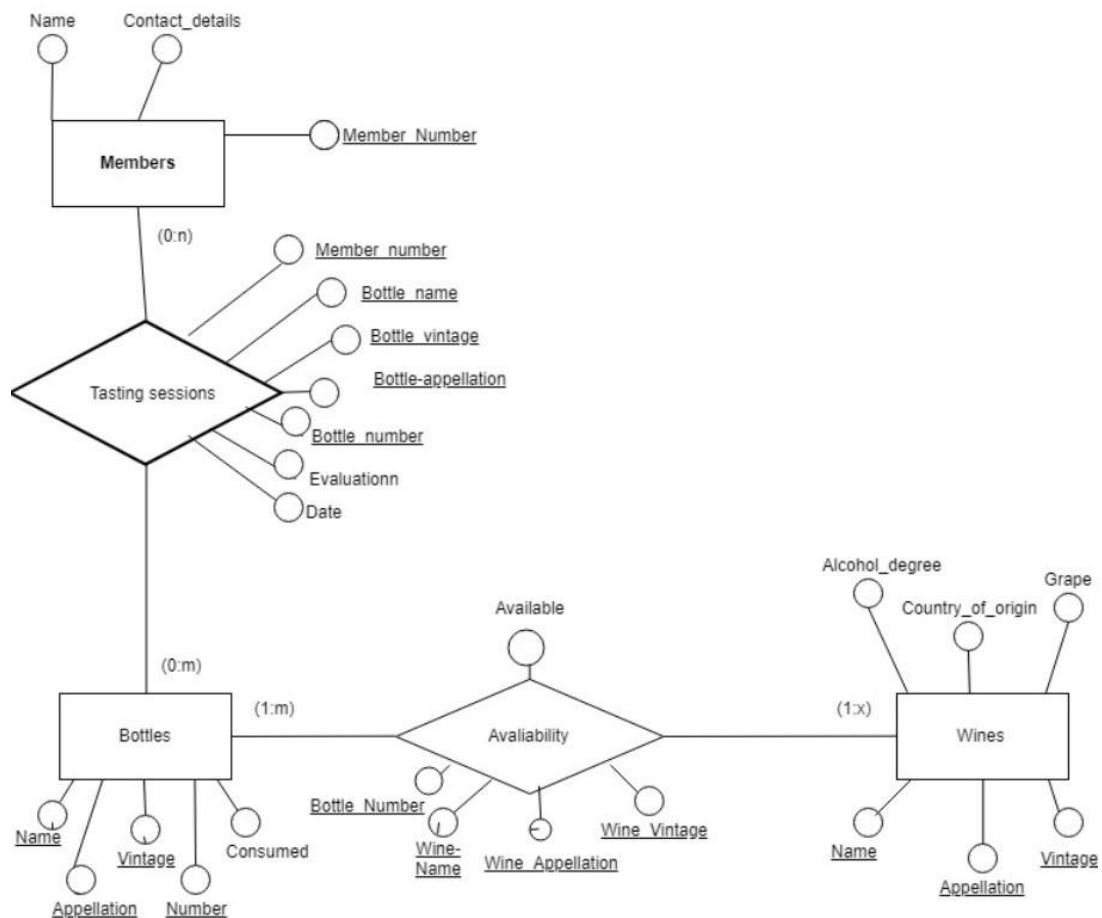
Members because it quite obviously needs to be an entity set and is quite simple, it has three attributes: 1. Name, 2. Contact_details which we just note as some String 3. a unique member_number which we use as the primary key.

Wines Because each wine is identified by its name, appellation and vintage (which we will use as part of a primary key to identify each wine) and also holds information about country of origin, grape and alcohol degree we will make wine an entity set as well. Because VINO needs to store information about wine they do not currently own, and might even never have owned a bottle of, we need to separate bottle and wine as two entity sets. The relationship between bottles and wines therefore simply become a availability relationship.

Bottles Because we want to describe the relationship between the members and the wine they drink and because a member will taste a specific bottle, and may even taste the same bottle at different dates and rank them differently, even though they are the "same" wine, we make bottles our third entity set. The relationship between members and bottles we note as the tasting sessions. In this relationship we also note the evaluation each member will make of each wine they taste. In this way, each entry in this relationship will be of a member, who have tasted a specific bottle of wine, the date they did it and the evaluation they gave the wine. An important attribute is whether or not the bottle has been consumed or not, which we again simply note in the attribute Consumed

Reasoning As mentioned above, we do not know how many wines VINO will actually have at any given time. They might like to record all the wine in world, (perhaps not) which would then only be stored in the entity set Wines and in the Availability relationship, this in turn would mean that bottles would only store any bottles they have owned or do own. But this become a necessity when a member can rank the same wine, but from different bottles, differently. This leads to some duplication in the database, but it also simplifies a potential usage of the database, as a lookup in Tastings sessions could potentially give you all the information you would need, without having to refer to another part of the database.

Constraints In the ER schema we have also noted the participation constraints. For Members to the Tasting sessions relationship, the relationship is only partial, as a new member could potentially not have been to a tasting yet and a member might never come to a tasting. The



Figur 1: VINO Er schema

"same" goes for Bottles and Tasting sessions, as a bottle could possibly be stored forever and never be drunk, even though that would be a shame. In the relationship between Bottles and Availability, each bottle that exists must be available and therefore also be associated with a wine. The same goes for wines, it must be either available or not and associated with a bottle if it is.

This gives us the following E-R Diagram. Where attributes underlined notes the primary keys of each entity set.

The SQL DDL statements to create the above schema:

Entity sets:

Members:

```
CREATE TABLE 'Members' (  
  'Name' VARCHAR(45),  
  'Member_number' INT ,  
  'Contact_details' VARCHAR(45),  
  PRIMARY KEY ('Member_number') )
```

Bottles:

```
CREATE TABLE 'Bottles' (  
  'Name' VARCHAR(45) ,  
  'Appellation' VARCHAR(45) ,  
  'Vintage' INT ,  
  'Consumed' VARCHAR(45) ,  
  'Number' INT ,  
  PRIMARY KEY ('Name', 'Appellation', 'Vintage', 'Number') )
```

Wines:

```
CREATE TABLE 'Wines' (  
  'Name' VARCHAR(45) ,  
  'Appellation' VARCHAR(45) ,  
  'Vintage' INT ,  
  'Alcohol_degree' INT ,  
  'Country_of_origin' VARCHAR(45) ,  
  'Grape_variety' VARCHAR(45) ,  
  PRIMARY KEY ('Name', 'Appellation', 'Vintage') )
```

Relationships:

Tasting Sessions:

```
CREATE TABLE 'Tasting Sessions' (  
  'Bottle_name' VARCHAR(45) ,  
  'Bottle_appellation' VARCHAR(45) ,  
  'Bottle_vintage' INT ,  
  'Bottle_number' INT ,  
  'Member_number' VARCHAR(45) ,  
  'Evaluation' VARCHAR(45) ,  
  'Date' DATE ,  
  PRIMARY KEY ('Bottle_name', 'Bottle_appellation', 'Bottle_vintage', 'Bottle_number', 'Member_number') ,  
  FOREIGN KEY (Member_number) REFERENCES members(Member_number)  
  FOREIGN KEY (Bottle_name) REFERENCES bottles(Name)  
  FOREIGN KEY (Bottle_appellation) REFERENCES bottles(Appellation)  
  FOREIGN KEY (Bottle_number) REFERENCES bottles(Number)  
  FOREIGN KEY (Bottle_vintage) REFERENCES bottles(Vintage)  
  )
```

Availability:

```

CREATE TABLE 'Availability'(
'Wines_Name' VARCHAR(45),
'Wines_Appellation' VARCHAR(45),
'Wines_Vintage' VARCHAR(45),
'Available' VARCHAR(45),
'Bottle_number' INT,
PRIMARY KEY('Wines_Name','Wines_Appellation','Wines_Vintage',Bottle_Number)
FOREIGN KEY (Wines_Name) REFERENCES wines(Name)
FOREIGN KEY (Wines_Appellation) REFERENCES wines(Appellation)
FOREIGN KEY (Wines_Vintage) REFERENCES wines(Vintage)
FOREIGN KEY (Bottle_number) References bottles(Number)
)

```

1.1 Relational Schema diagram

As requested in the announcement, a Schema Diagram:

Members:

<u>Member_number</u>	Name	Contact_Details
----------------------	------	-----------------

Tasting session

<u>Member_number</u>	<u>Bottle_name</u>	<u>Bottle_vintage</u>	<u>Bottle_appellation</u>	<u>Bottle_number</u>	Evaluation	Date
----------------------	--------------------	-----------------------	---------------------------	----------------------	------------	------

Bottles

<u>Name</u>	<u>Vintage</u>	<u>Appellation</u>	<u>Number</u>	Consumed
-------------	----------------	--------------------	---------------	----------

Availability

<u>Bottle_Number</u>	<u>Wine_Name</u>	<u>Wine_Appellation</u>	<u>Wine_Vintage</u>	Available
----------------------	------------------	-------------------------	---------------------	-----------

Wines

<u>Name</u>	<u>Appellation</u>	<u>Vintage</u>	Alcohol_degree	Country_of_origin	Grape
-------------	--------------------	----------------	----------------	-------------------	-------