SEIFERT BELMONT RED
Mock database Design

Aleksei Miros

17th April 2022

Contents

Project Description:

The overarching domain of this database project is a medium-sized cattle company, modelled after a real company, Seifert Belmont Reds. The domain will focus on the internal aspect of the business excluding any customer interactions. This includes major assets such as vehicles, cattle, properties and their paddocks, and primary staff excluding the owners but including managers.

This domain contains a few specific constraints and rules:

- Due to weather and property conditions, paddocks or entire farms may not always be utilised during a given period.
    - This involves paddocks without cattle or farms without staff or vehicles.
- All farms have at least one paddock.
- All farms and paddocks are named uniquely across the business.
- Vehicles refers to on property vehicles, not road-registered vehicles.
    - As such the vehicles do not have official registration and have an internal integer ID which is unique across all vehicles on all properties
- Staff can move between properties, as such their staff ID is unique across all properties, but they are registered to a primary location.

Following this domain, the primary functionality of this database is intended to assist in management and analytics of cattle. Though no direct analytics functionality is projected to be implemented in the database, aggregation functions are planned to be implemented in an accessible manor to assist in further insights. Secondary functionality includes systems to help manage staff, vehicle information, paddocks, and their feed bins.

To implement this project, MySQL will be used as the database management system. MySQL will be used with Python implementing the Flask library to build the application, while HTML and CSS will be used to build a graphical user interface for the application.

ER Diagram:

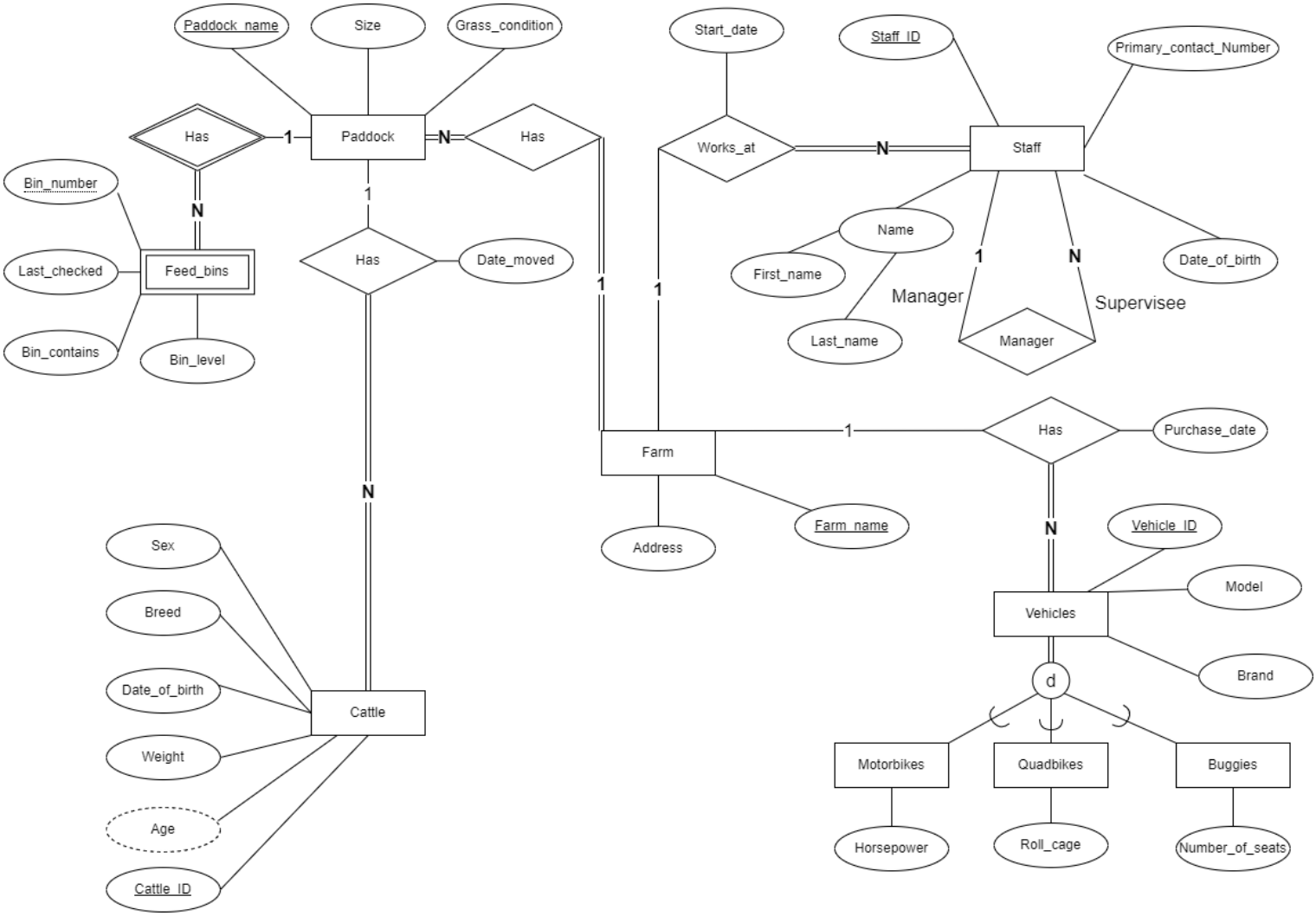

Figure 1 : ER Diagram

Candidate Keys:

Table	***Primary Key***	***Candidate Keys***
FARM	Farm_name	Address
STAFF	Staff_ID	
VEHICLES	Vehicle_ID	
PADDOCK	Paddock_name	
CATTLE	Cattle_ID	
Feed_BINS	{Paddock_name, Bin_number}	

Table 1: Candidate Keys

Schema:

FARM

[Farm_name : Variable string, Address : Variable string]

STAFF

[Staff_ID : Integer > 0, First_name : Variable string, Last_name : Varibale String, Date_of_birth : Date , **Farm_name :** Variable string**,** start_date : Date, Manager_ID : Integer > 0, Primary_contact_number Variable string]

PADDOCK

[Paddock_name : Varibale string, Size : Real number > 0 , Grass_condition : {'Dry', 'Green'}, **Farm_name** : Variable string]

CATTLE

[Cattle_ID : Integer > 0, Sex : {'Male', 'Female'}, Breed : {'Belmont Red', 'Angus', 'Cross'}, Date_of_birth : Date, Weight : Real number to two decimal places, **Paddock_name** : Varibale string, Date_moved : Date]

FEED_BINS

[**Paddock_name** : Integer > 0, Bin_number : Integer > 0, Last_checked : Date, Bin_contains : {'Wheat', 'Salt Lick', 'Sorghum'}, Bin_level : 0 ≤ real number ≤ 1]

VEHICLES

[Vehicle_ID : Integer > 0 , Model : Variable string, Brand : Varibale string, **Farm_name :** Variable string, Purchase_date : Date]

MOTORBIKES

[**Vehicle_ID** : Integer > 0 , Horsepower : Integer > 0]

QUADBIKES

[**Vehicle_ID** : Integer > 0, Roll_cage : {'Yes', 'No'}]

BUGGIES

[**Vehicle_ID** : Integer > 0, Number_of_seats : 2 ≤ Integer ≤ 4]

Foreign Keys:

FEED_BINS.Paddock_name → PADDOCK.Paddock_name

STAFF.Farm_name → Farm.Farm_name

STAFF.Manager_ID → STAFF.Staff_ID

VEHICLE.Farm_name → Farm.Farm_name

PADDOCK.Farm_name → Farm.Farm_name

CATTLE.Paddock_name → PADDOCK.Paddock_name

MOTORBIKES.Vehicle_ID → VEHICLE_Vehicle_ID

QUADBIKES.Vehicle_ID → VEHICLE_Vehicle_ID

BUGGIES.Vehicle_ID → VEHICLE_Vehicle_ID

Functional Dependencies:

FARM:

- Farm_name → Address
    - All farms have a unique name and one address. Therefore, for any given farm name, the address will be the same for every instance of said name.
- Address → Farm_name
    - All farms have a unique name and one address. Therefore, for any given address, the farm name will be the same for every instance of said name.

STAFF:

- Staff_ID → First_name, Last_name, Date_of_birth, Farm_name, Start_date, Manager_ID, Primary_contact_number
    - Staff IDs are unique across the entire business. Therefore, for any given Staff ID, the First name, last name, date of birth, the farm name the staff works at, their start date, and their managers ID will always be the same for every instance of said ID.
    - As multiple staff members may share a dwelling with a landline as their primary contact number, the contact number cannot be used as a determinant or primary key.

PADDOCK:

- Padock_name → Size, Grass_condition, Farm_name
    - All paddocks across the entire business have unique names. Therefore, for any given paddock name, the size, grass condition, and farm it's apart of will be the same for every instance of said name.

CATTLE:

- Cattle_ID → Sex, Breed, Date_of_birth, Weight, Paddock_name, Date_moved
    - All cattle across the entire business have a unique ID. Therefore, for any given cattle ID, the sex, breed, date of birth, weight, paddock they reside in, and they date they were last moved will be the same for every instance of said ID.

VEHICLES:

- Vehicle_ID → Model, Brand, Farm_name, Purchase_date
    - Vehicle IDs are unique across the entire business. Therefore, for any given vehicle ID, the model, brand, farm the vehicle is used at, and the purchase date will be the same for every instance of said ID.
- Model → Brand
    - Each model name is only apart of one brand due to copyright laws.

MOTORBIKES:

- Vehicle_ID → Horsepower
    - Vehicle IDs are unique across the entire business. Therefore, for any given vehicle ID, the horsepower will be the same for every instance of said ID.

QUADBIKES:

- Vehicle_ID → Roll_cage
    - Vehicle IDs are unique across the entire business. Therefore, for any given vehicle ID, the roll cage status will be the same for every instance of said ID.

BUGGIES:

- Vehicle_ID → Numer_of_seats
    - Vehicle IDs are unique across the entire business. Therefore, for any given vehicle ID, the number of seats will be the same for every instance of said ID.

FEED_BINS :

- {Paddock_name, Bin_number} → Last_checked, Bin_contains, Bin_level
    - Bin numbers are unique in every paddock. Therefore, for any given bin number in addition to a paddock ID, the date the bin was last checked, what it contains, and its level will be the same for every instance of said paddock ID with bin number.

Normalised Schema : BCNF

FARM:

- Farm_name → Address
- Address → Farm_name

Farm_name$^+$ → X where X is all attributes of FARM. Therefore, Farm_name is a superkey.

Address$^+$ → X where X is all attributes of FARM. Therefore, Address is a superkey.

As both Farm_name, and Address are superkeys, X is a superkey for all X → B. Therefore, FARM is in BCNF form.

FARM	
Farm_name : Variable string	Primary key
Address : Variable string	Candidate key

STAFF:

- Staff_ID → First_name, Last_name, Date_of_birth, Farm_name, Start_date, Manager_ID, Primary_contact_number

$Staff\_ID^{+}$ → X where X is all attributes of STAFF. Therefore, Staff_ID is a superkey.

As Staff_ID is a superkey, X is a superkey for all X → B. Therefore, STAFF is in BCNF form.

STAFF	
Staff_ID : Integer > 0	Primary key
First_name : Varibale string	
Last_name : Variable string	
Date_of_birth : Date	
Farm_name : Variable string	**Foreign key → FARM.Farm_name**
Start_date : Date	
Manager_ID : Integer > 0	
Primary_contact_number : Variable string	

PADDOCK:

- Paddock_name → Size, Grass_condition, Farm_name

Paddock_name$^+$ → X where X is all attributes of PADDOCK. Therefore, Paddock_name is a superkey.

As Paddock_name is a superkey, X is a superkey for all X → B. Therefore, PADDOCK is in BCNF form.

PADDOCK	
Paddock_name : Variable string	Primary key
Size : Real number > 0	
Grass_condition : {'Dry', 'Green'}	
Farm_name : Variable string	**Foreign key → FARM.Farm_name**

CATTLE:

- Cattle_ID → Sex, Breed, Date_of_birth, Weight, Paddock_name, Date_moved

Cattle_ID$^+$ → X where X is all attributes of CATTLE. Therefore, Cattle_ID is a superkey.

As Cattle_ID is a superkey, X is a superkey for all X → B. Therefore, CATTLE is in BCNF form.

CATTLE	
Cattle_ID : Int > 0	Primary key
Sex : {'Male', 'Female'}	
Breed : {'Belmont Red', 'Angus', 'Cross'}	
Date_of_birth : Date	
Weight : Real number > 0	
Paddock_name : Variable string	**Foreign key → PADDOCK.Paddock_name**
Date_moved : Date	

VEHICLES:

- Vehicle_ID → Model, Brand, Farm_name, Purchase_date
- Model → Brand.

Vehicle_ID$^+$ → X where X is all attributes of VEHICLES. Therefore, Vehicle_ID is a superkey.

Model$^+$ does not → X where X is all attributes of VEHICLES. Therefore, Model is not a superkey and Model → Brand violates BCNF.

Therefore, VEHICLES is split.

VEHICLES	
Vehicle_ID : Integer > 0	Primary key
Model : Variable string	
Farm_name : Variable string	**Foreign key → FARM.Farm_name**
Purchase_date : Date	

- Vehicle_ID → Model, Brand, Farm_name, Purchase_date

Vehicle_ID$^+$ → X where X is all attributes of VEHICLES. Therefore, Vehicle_ID is a superkey.

As Vehicle_ID is a superkey, X is a superkey for all X → B. Therefore, VEHICLES is in BCNF form.

VEHICLE_BRANDS	
<u>Vehicle_ID : Integer > 0</u>	**<u>Primary key & foreign key</u> →** **VEHICLES.Vehicle_ID**
Brand : Variable string	

- Model → Brand

$Model^+$ → X where X is all attributes of VEHICLES. Therefore, Model is a superkey.

As Model is a superkey, X is a superkey for all X → B. Therefore, VEHICLE_BRANDS is in BCNF form.

MOTORBIKES:

- Vehicle_ID → Horsepower

$Vehicle\_ID^+$ → X where X is all attributes of MOTORBIKES. Therefore, Vehicle_ID is a superkey.

As Vehicle_ID is a superkey, X is a superkey for all X → B. Therefore, MOTORBIKES is in BCNF form.

MOTORBIKES	
<u>Vehicle_ID : Integer > 0</u>	**<u>Primary key & foreign key</u> →** **VEHICLES.Vehicle_ID**
Horsepower : Integer > 0	

QUADBIKES:

- Vehicle_ID → Roll_cage

Vehicle_ID$^+$ → X where X is all attributes of QUADBIKES. Therefore, Vehicle_ID is a superkey.

As Vehicle_ID is a superkey, X is a superkey for all X → B. Therefore, QUADBIKES is in BCNF form.

QUADBIKES	
Vehicle_ID : Integer > 0	**Primary key & foreign key →** **VEHICLES.Vehicle_ID**
Roll_cage : {'Yes', 'No'}	

BUGGIES:

- Vehicle_ID → Numer_of_seats

Vehicle_ID$^+$ → X where X is all attributes of BUGGIES. Therefore, Vehicle_ID is a superkey.

As Vehicle_ID is a superkey, X is a superkey for all X → B. Therefore, BUGGIES is in BCNF form.

BUGGIES	
Vehicle_ID : Integer > 0	**Primary key & foreign key →** **VEHICLES.Vehicle_ID**
Number_of_seats : 2 ≤ Integer ≤ 4	

FEED_BINS :

- {Paddock_name Bin_number} → Last_checked, Bin_contains, Bin_level

{Paddock_name, Bin_number} $^{+}$ → X where X is all attributes of FEED_BINS. Therefore, {Paddock_name, Bin_number} is a superkey.

As {Paddock_name, Bin_number} is a superkey, X is a superkey for all X → B. Therefore, FEED_BINS is in BCNF form.

FEED_BINS	
Paddock_name : Variable string	**Primary key & foreign key →** **PADDOCK.Paddock_name**
Bin_number : Integer > 0	Primary key
Last_checked : Date	
Bin_contains : {'Wheat', 'Salt Lick', 'Sorghum'},	
Bin_level : 0 ≤ real number ≤ 1	