Demonstrate that all the relations in the relational Schema are normalized to BCNF

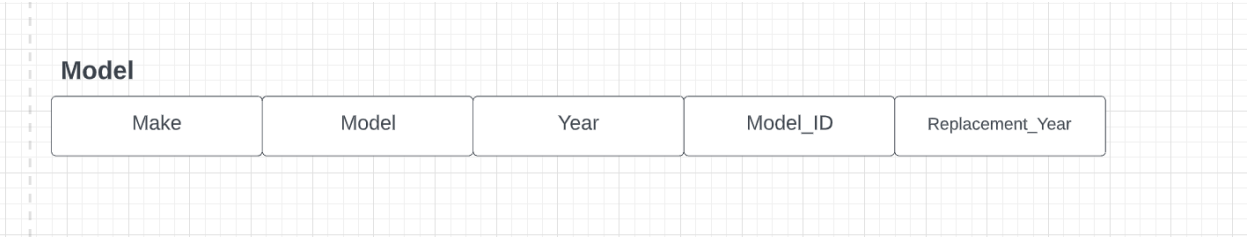
Normalization is important in database systems because it minimizes the redundancy, duplicate data, and ensures all the tables are efficient, accurate and fast. One of the normalizations is Boyce-Codd-Normal Form; The advantage of BCNF is that it reduces all the duplicate data and arranges the data such that, when we modify it, we only change it one place. BCNF is an extension of the third normal form. For the tables to be in BCNF, those must be in 3NF

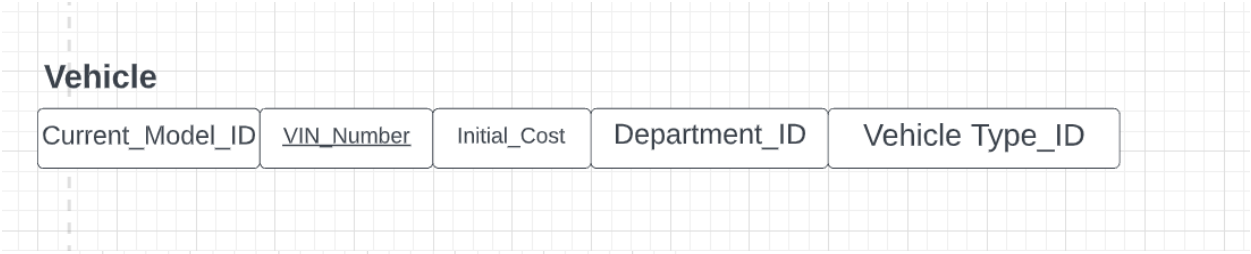
Our relational schema contains three tables, Vehicle, Emissions, and Repeated Costs, that are created using the ER are not in BCNF because there is redundancy in data.

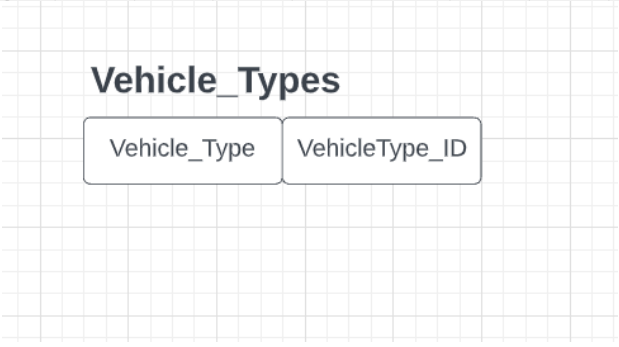
Consider the Vehicle table given below. The table is in the first normal form; however, not in BCNF. Since the Vehicle table is not converted to BCNF, then it may violate insertion, deletion, or modification anomaly. For instance, If the table has two rows that store the same information of one car, whose Current_Make is Ford, Current_Year is 2012, and Current_Model is F350 but belongs to different departments. If we want to update the department name of the car, then we have to update it in the two rows, or the data will become inconsistent. If the department name only gets updated in one row but not in the other, then the car Ford F350 will be having two different department names, which is incorrect and would lead to inconsistent data.

So to remove data redundancy and inconsistency within the data, we divided the vehicle relation into several relations. We have created a separate relation called Department that stores the department name and department ID. The attribute Department ID is added because the department relation needs a primary key. Another relation we have created is the Models table which stores the attributes make, model, year, modelID, and replacement

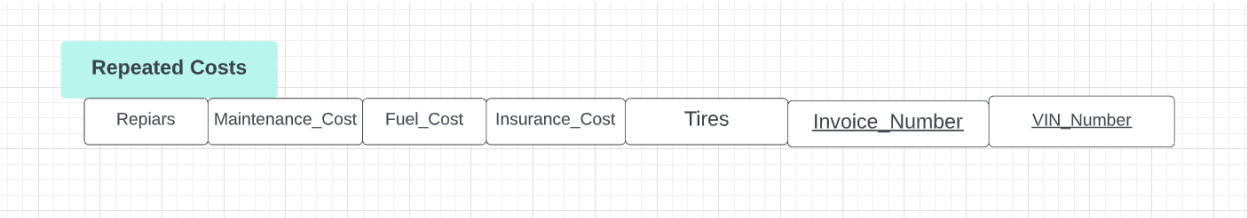
year. In this relation, modelID would be the primary key and another attribute replacement year is added because our design requires to replace all the cars manufactured before 2006. Additionally, another relation that emerged from the original Vehicle relation is Vehicle Types. This relation stores the types of vehicles along with the vehicleTypeID (primary key). Finally, the below relations are generated after normalizing the original Vehicle relation into BCNF.



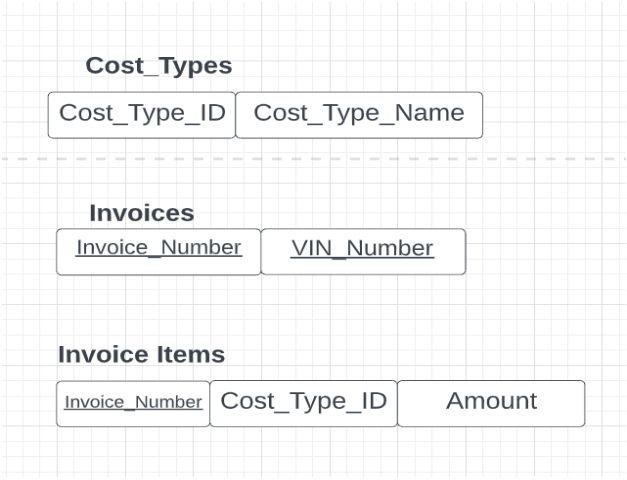




Another relation we have in the relational schema is Repeated Costs, which stores attributes such as repairs cost, maintenance cost, tires cost, insurance cost, fuel cost, VIN number and invoice number. Before normalizing the table looks like:

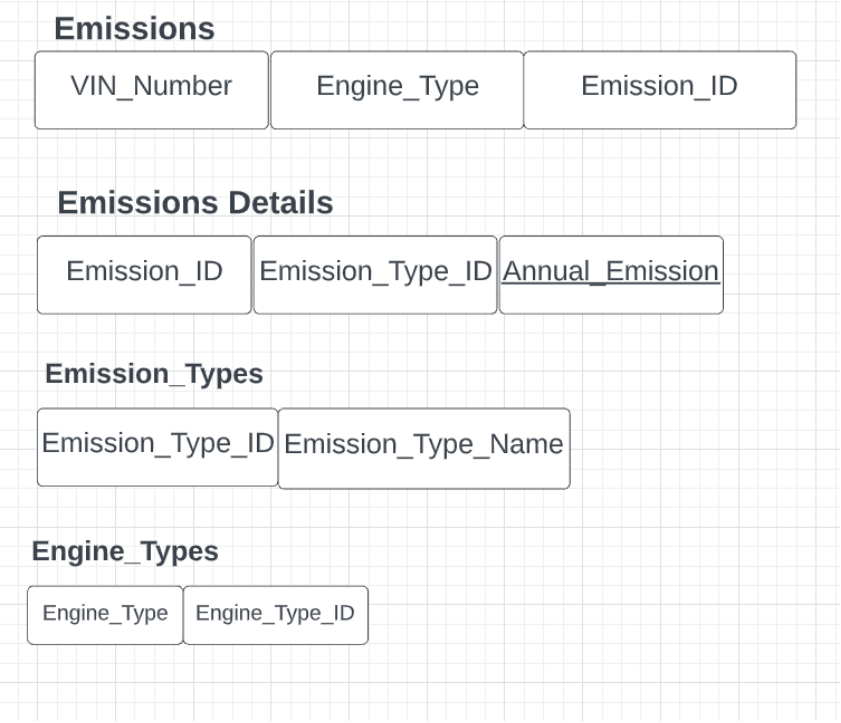


The Repeated Costs relation is in one normal form because it does not contain composite or multi-valued attributes. Every attribute in the relation is a single valued attribute. The relation, however, is not in BCNF because of the dependencies that exist in the relation which may cause complications when we implement the queries . To remove such issues, we have generated several relations based off of the Repeated Costs relation. One of the relations we have created is Cost_Types. This relation contains the attributes Cost_Type_Name, such as maintenance, fuel, or insurance. We have also added the Cost_Type_ID as a primary key. Another relation created is Invoices, which contains invoice number and VIN number. The final relation we have generated for the Repeated Costs relation is Invoice Items. This table has invoice number, cost_type_ID, and amount. Finally, the below relations are generated after normalizing the original Repeated Costs relation into BCNF.



The last relation we have in the relational schema is Emissions. Just as the above relations, the Emission relation is also in 1NF as it does not violate

the rules of having single valued attributes. As the table contains multiple attributes that are grouped together, it is crucial to separate attributes and create separate relations for those. For this table, we generated four relations in total and those are Emissions, Emissions_Details, Emission_Types, and Engine_Types. First, the Emission relation consists of VIN_Number, Engine_Type, and Emission_Id attributes. The Emissions_Details relation consists of Emission_ID, Emission_Type_ID and Annual Emission. For the Emission_Types table, the attributes we stored are Emission_Type_ID and Emission_Type_Name. The table Engine_Type contains Engine_Type attribute and Engine_Type_ID. Finally, the below relations are generated after normalizing the original Emissions relation into BCNF.



In conclusion, BCNF normalization has created more tables and established relationships among those tables.

Views (Virtual Tables):

1. **List the insurance amount for all the cars whose cost equals 800**

   CREATE VIEW VehiclesWithInsurance800 AS
   Select V. VIN_Number, M.Make, M. Model, M.Year, IT. Amount
   From Vehicles V
   JOIN Model M ON M. Model_ID = V. Model_ID
   JOIN Invoices I ON V. VIN_Number = I. VIN_NUMBER
   JOIN Invoice_Items IT ON I. Invoice_Number = IT. Invoice_Number
   JOIN Cost_Types CT ON CT.Cost_Type_ID = IT.Cost_Type_ID
   Where CT.Cost Type Name = "Insurance Cost" AND IT.Amount = '800'

2. **What is the vehicle type of Ford F350 Box Truck**

   CREATE VIEW Ford350VehicleType AS
   Select DISTINCT M. Make, M. Model, VT. Vehciel_Type
   From Vehicle_Types VT
   JOIN Model M ON VT. Vehicle_Type_ ID = M. Vehicle_Type_ ID
   Where M. Make = "Ford" AND M. Model = 'F350 Box Truck'

Queries:

1. **Count all the cars who belong to the department "Carpenter Roof".**

   Select count (VIN_Number)
   From Vehicles V
   JOIN Departments D ON D.Department_ID = V. Department_ID
   Where D. Department Name = 'Carpenter Roof';

2. **List the fuel cost and maintenance cost for Ford, Club Wagon**

   Select SUM (IT. Amount) AS Fuel_Cost, SUM (IT1. Amount) AS Maintenance _Cost

   From Vehicles V

JOIN Model M ON M. Model_ID = V. Model_ID
JOIN Invoices I ON I. VIN_Number = V. VIN_Number
JOIN Invoice_Items IT ON IT. Invoice_Number = I. Invoice_Number
JOIN Invoice_Items IT1 ON IT1. Invoice_Number = I. Invoice_Number
JOIN Cost_Types CT ON CT. Cost_Type_ID = IT. Cost_Type_ID AND CT. Cost_Type_Name = “Fuel_Cost”
JOIN Cost_Types CT1 ON CT1. Cost_Type_ID = IT1. Cost_Type_ID AND CT. Cost_Type_Name = “Maintenance_Cost”
Where M. Make = “Ford” AND M. Model = “Club Wagon”

3. **What is the annual emission for all Ford Cars?**

    Select Sum (ED. Annual Emission)
    From Model M,
    JOIN Vehicle V ON V. current_Model_ID = M. Model_ID
    JOIN Emissions E ON E. VIN_NUMBER = V. VIN_NUMBER
    JOIN Emission Details ED ON E. Emission_ID = ED. Emission_ID
    Where M. Make = “Ford”

4. **What is the emission type for all Ford Cars?**

    Select distinct ET.Emission_type_name
    FROM Vehicle V

    JOIN Model M ON M. Model_ID = V. Current_Model_ID
    JOIN Emissions E ON E. VIN_Number = V. Vin_number
    Join Emissions_Details ED ON ED.Emission_ID=E.Emission_ID
    JOIN Emission_Types ET ON ET. Emission_Type_ID = Ed. Emission_Type_ID
    WHERE M.Make= ’ford’

5. **What is the vehicle type of Dodge Caravan?**

```
Select Distinct VT. Vehcile_Type
From Model M
JOIN  Vehicle_Type VT ON VT. Vehicle_Type_ID = M. Vehicle_Type_ID
Where  M. Make = “Dodge” AND M. Model = “Caravan”
```

6. **What is the average initial cost of all Chevy cars ?**

```
Select Avg (V. Initial Cost )
FROM Vehicle V
JOIN Model M ON M. Model_ID = V. Current_Model_ID
Where M. Make = “Chevy”
```

7. **What is the engine type of all “Dodge Grand Caravan” cars?**

```
Select Distinct ENT. Engine_Type,
From  Vehicle V
JOIN Model M ON M. Model_ID = V. Current_Model_ID
JOIN Emissions E ON E. VIN_Number = V. VIN_Number
Where M. Make = “Dodge” AND M. Model = “Grand Caravan”
```

8. **How many cars were manufactured in 2018?**

```
Select Count (V. VIN_Number)
From Model M
Join Vehicle V ON V.Current_Model_ID = M. Model_ID
Where M. Year = ‘2018’
```

9. **Given all the Ford cars, which car has the maximum maintenance cost?**

```
Select distinct MAX (IT. Amount)
From Vehicle V
Join MOdel M ON M.Model_ID = V. Current_Model_ID
Join Invoices I ON I. Vin_Number = V. VIN_Number
```

Join Invoice Items IT ON IT. Invoice Number = I. Invoice Number

Join Cost_Types CT on CT. Cost_Type_ID = IT. Cost_Type_ID

Where M. Make = “Ford” AND CT. Cost_Type_Name = “Maintenance Cost”

10. Given all the Chevy cars, which car has the minimum Tires cost?

Select distinct MIN (IT. Amount)

From Vehicle V

Join MOdel M ON M.Model_ID = V. Current_Model_ID

Join Invoices I ON I. Vin_Number = V. VIN_Number

Join Invoice Items IT ON IT. Invoice Number = I. Invoice Number

Join Cost_Types CT on CT. Cost_Type_ID = IT. Cost_Type_ID

Where M. Make = “Chevy” AND CT. Cost_Type_Name = “Tires Cost”