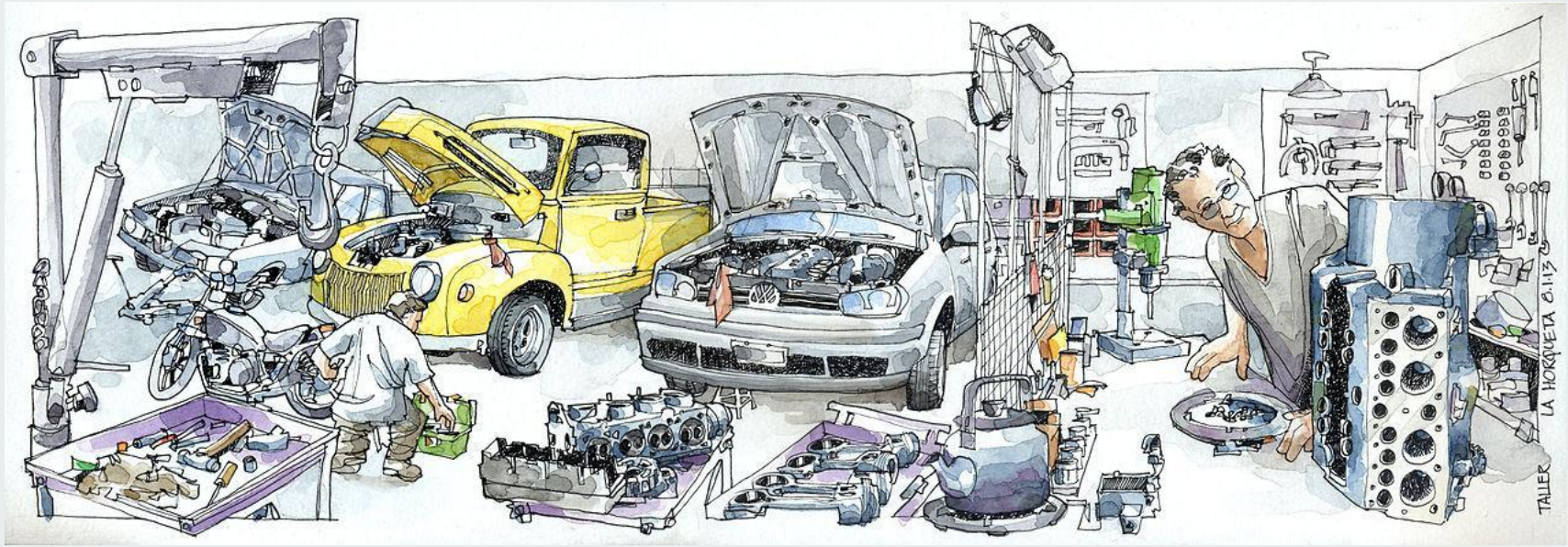
Relational Database Design Final Project

For “BestCar” body shop


TALLER
LA HORQUETA 6.1.15

Questions for My Client

- Interests/Background and Entities?
- Relationship between entities? (Cardinalities, Participation)

Background & Interest

- Hired a new data analyst. 😀
- He found data redundancy and inconsistency issues. => Reorganize database 🙀

Entities, attributes, and identifiers

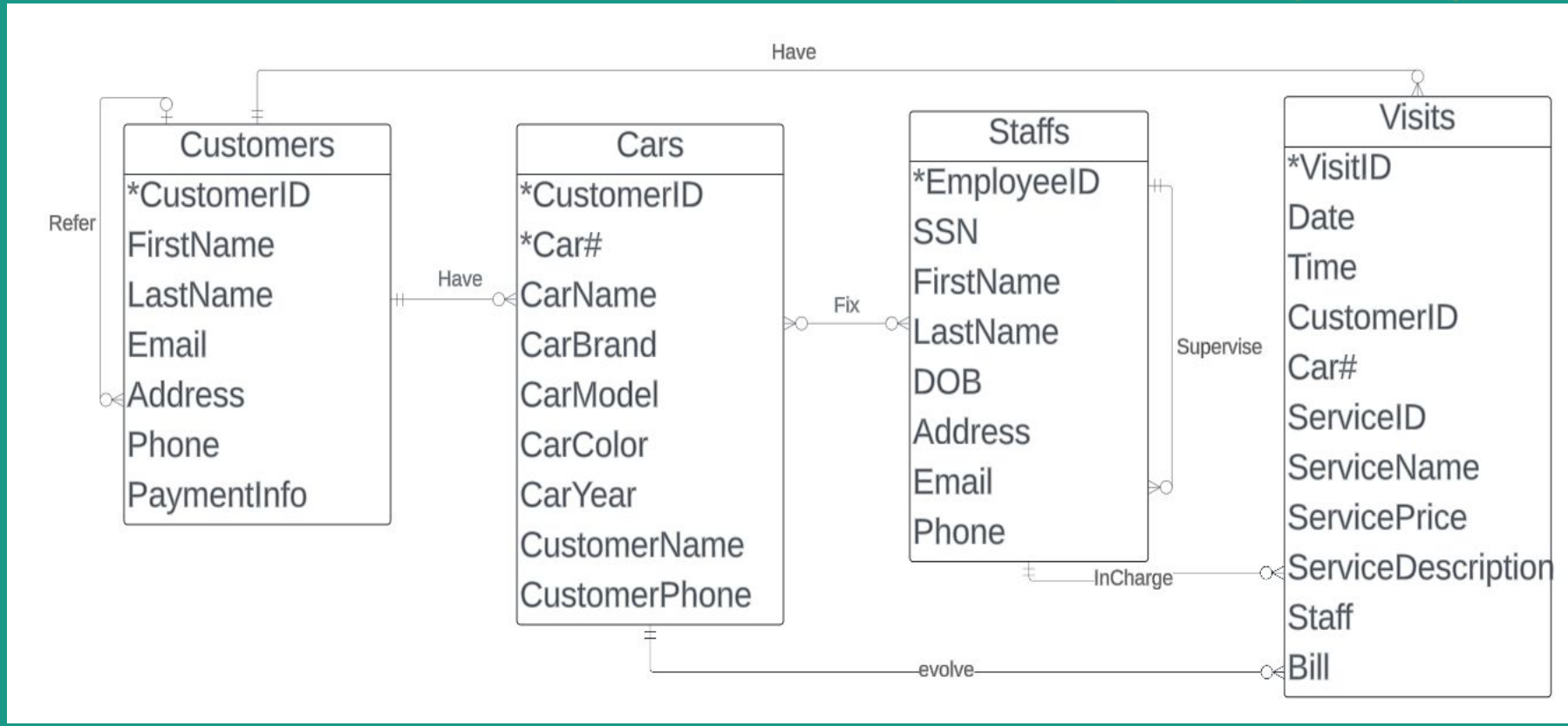
- Customers: *CustomerID, FirstName, LastName, Email, Address, Phone, PaymentInfo
- Cars: *CustomerID, *Car#, CarName, CarBrand, CarModel, CarColor, CarYear, CustomerName, CustomerPhone,
- Staffs: *EmployeeID, SSN, FirstName, LastName, DOB, Address, Email, Phone
- Visits: *VisitID, Date, Time, CustomerID, Car#, ServiceID, ServiceName, ServicePrice, ServiceDescription, Staff, Bill

(* indicate the primary key):

Relations between Entities

- A customer may have one or more cars; A car must belong to one and only one customer.
- A customer may have one or more visits; A visit must be done by one and only one customer.
- A car may have one or more visits; A visit must involve one and only one car.
- A car may be fixed with one or more staff. A staff may fix one or more cars;.
- A staff must be in charge of one or more visits; A visit must be charged by one and only one staff.
- A staff must have one and only one supervisor; A staff may supervise one or more staff.
- A customer may be referred by only one customer; A customer may refer to one or more customers.

Create the Entity Relationship Diagram (ERD)



Convert ERD to Relational Model

- Customers (CustomerID, FirstName, LastName, Email, Address, Phone, PaymentInfo, ReferredByCustomerID(fk)).
- Cars (CustomerID(fk), Car#, CarName, CarBrand, CarColor, CarYear, CustomerName, CustomerPhone).
- Staff (EmployeeID, SSN, FirstName, LastName, DOB, Address, Email, Phone, SupervisorID(fk)).
- Visit (VisitID, Date, Time, CustomerID(fk), Car#(fk), ServiceID, ServiceName, ServicePrice, ServiceDescription, EmployeeID(fk), Bill).
- Cars_Staff (CustomerID(fk), Car#(fk),EmployeeID(fk))

Normalize the Relational Model to 3NF

✋Wait, first need to get functional dependencies (FDs).

Functional Dependencies 1

- Customers (CustomerID, FirstName, LastName, Email, Address, Phone, PaymentInfo, ReferredByCustomerID(fk)).
  - FD1: CustomerID → FirstName, LastName, Email, Address, Phone, PaymentInfo, ReferredByCustomerID
- **Cars** (CustomerID(fk), Car#, CarName, CarBrand, CarColor, CarYear, CustomerName, CustomerPhone).
  - FD1: CustomerID,Car# → CarName, CarBrand, CarColor, CarYear, CustomerName, CustomerPhone
  - FD2: CustomerID → CustomerName, CustomerPhone

Functional Dependencies 2

- Staff (EmployeeID, SSN, FirstName, LastName, DOB, Address, Email, Phone, SupervisorID(fk)).
  - FD1: EmployeeID → SSN, FirstName, LastName, DOB, Address, Email, Phone, SupervisorID
- **Visit** (VisitID, Date, Time, CustomerID(fk), Car#(fk), ServiceID, ServiceName, ServicePrice, ServiceDescription, EmployeeID(fk), Bill).
  - FD1: VisitID → Date, Time, CustomerID, Car#, ServiceID, ServiceName, ServicePrice, ServiceDescription, EmployeeID, Bill
  - FD2: ServiceID → ServiceName, ServicePrice, ServiceDescription.

Functional Dependencies 3

- Cars_Staff (CustomerID(fk), Car#(fk), EmployeeID(fk))
  - There is no non-primary-key attribute.

Normalize Relational Model to 3NF 1

Customers, Staff, and Cars_Staff relations are in 3NF

- In 1NF - Each cell has one value
- In 2NF - no partial functional dependencies
- In 3NF - no transitive functional dependencies

Normalize Relational Model to 3NF 2

Cars relation is not in 2NF

- In 1NF
- Not in 2NF
  - FD2: CustomerID → CustomerName, CustomerPhone. **=>** partial functional dependency

Part of primary key

non-primary-key attributes

  - Normalize Cars to 2NF (delete CustomerName, CustomerPhone in “Cars” relation)

Normalize Relational Model to 3NF 3

- **Cars** (CustomerID(fk), Car#, CarName, CarBrand, CarColor, CarYear, ~~CustomerName, CustomerPhone~~).
  - FD1: CustomerID, Car# → CarName, CarBrand, CarColor, CarYear~~, CustomerName, CustomerPhone~~
  - ~~FD2: CustomerID → CustomerName, CustomerPhone~~

Now “Cars” is in 2NF. It’s also in 3NF because there are none transitive dependency!

Normalize Relational Model to 3NF 5

- Visit relation is not in 3NF.
  - In 1NF
  - In 2NF - No partial functional dependency
  - Not in 3NF:
    - FD2: ServiceID → ServiceName, ServicePrice, ServiceDescription. VisitID → ServiceID, and ServiceID → ServiceName, ServicePrice

Transitive functional dependency

Normalize Relational Model to 3NF 4

- We need to normalize Visit to 3NF (1. Create a new relation “Service” to put ServiceID, ServiceName, ServicePrice, ServiceDescription 2. Delete ServiceName, ServicePrice, ServiceDescription in “Visit” relation)
  - Service (ServiceID, ServiceName, ServicePrice, ServiceDescription)
    - FD1: ServiceID → ServiceName, ServicePrice, ServiceDescription
  - **Visit** (VisitID, Date, Time, CustomerID(fk), Car#(fk), ServiceID, ServiceName, ServicePrice, ServiceDescription, EmployeeID(fk), Bill).
    - FD1: VisitID → Date, Time, CustomerID, Car#, ServiceID, ~~ServiceName, ServicePrice, ServiceDescription~~, EmployeeID, Bill
    - ~~FD2: ServiceID → ServiceName, ServicePrice, ServiceDescription.~~

Finalize the relational model in 3NF

- Customers (CustomerID, FirstName, LastName, Email, Address, Phone, PaymentInfo, ReferredByCustomerID(fk)).
  - FD1: CustomerID → FirstName, LastName, Email, Address, Phone, PaymentInfo, ReferredByCustomerID
- Cars (CustomerID(fk), Car#, CarName, CarBrand, CarColor, CarYear).
  - FD1: CustomerID, Car# → CarName, CarBrand, CarColor, CarYear
- Staff (EmployeeID, SSN, FirstName, LastName, DOB, Address, Email, Phone, SupervisorID(fk)).
  - FD1: EmployeeID → SSN, FirstName, LastName, DOB, Address, Email, Phone, SupervisorID
- Service (ServiceID, ServiceName, ServicePrice, ServiceDescription)
  - FD1: ServiceID → ServiceName, ServicePrice, ServiceDescription
- Visit (VisitID, Date, Time, CustomerID(fk), Car#(fk), ServiceID(fk), EmployeeID(fk), Bill).
  - FD1: VisitID → Date, Time, CustomerID, Car, ServiceID, EmployeeID, Bill
- Cars_Staff(CustomerID(fk), Car#(fk), EmployeeID(fk))
  - There is no non-primary-key attribute.


THANK YOU THANK YOU