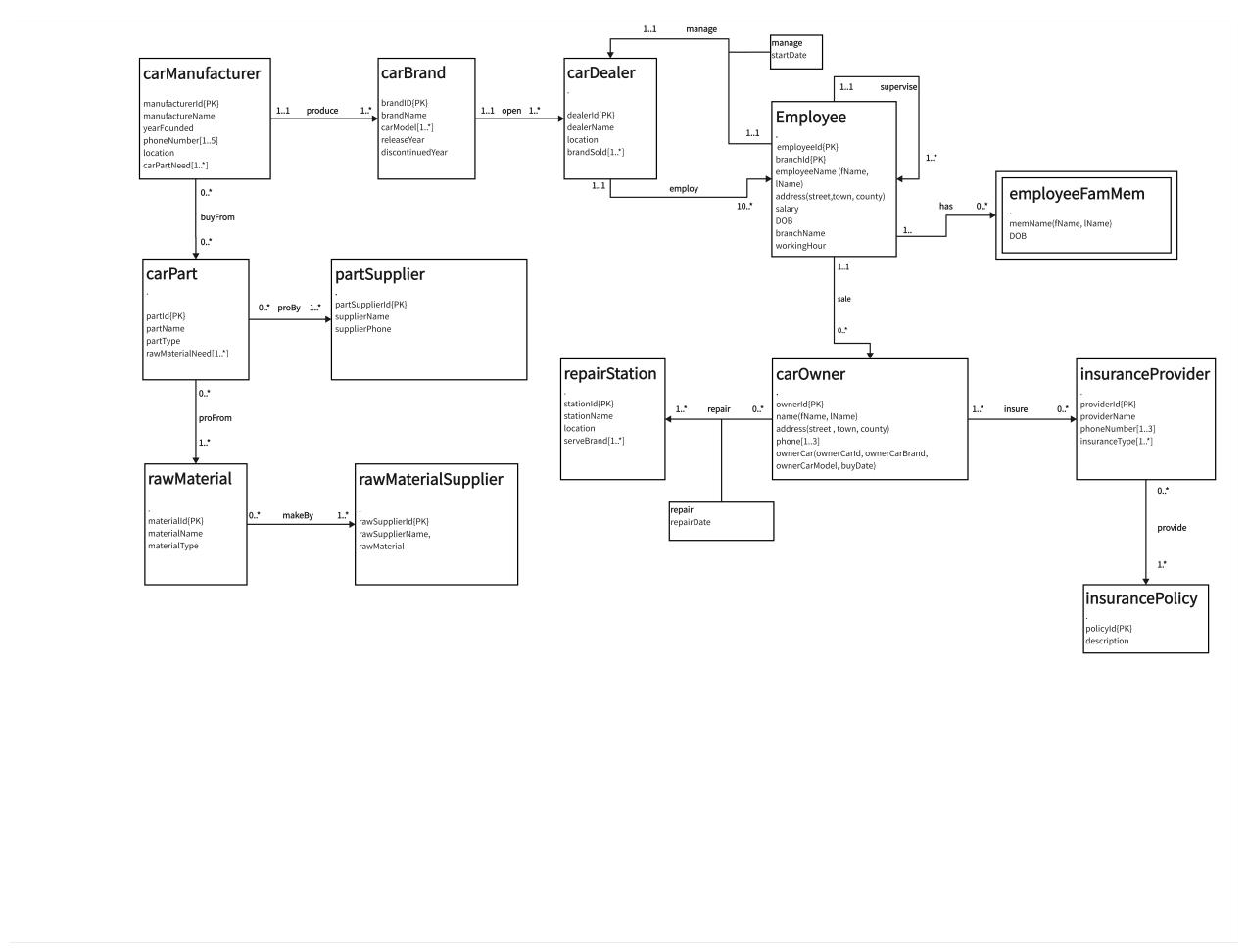
******

***Step1***

1. carManufacturer (manufacturerId,manufacturerName,yearFounded,location)

Pk: manufacturerId

1. carPart (partId,partName,partType)

Pk: partId

1. rawMaterial (materialId,materialName,materialType)

Pk: materialId

1. partSupplier (partSupplierId,supplierName,supplierPhone)

Pk: partSupplier

1. rawMaterialSupplier (rawSupplierId,rawSupplierName,rawMaterial)

Pk: rawSupplierId

1. carBrand (brandID,brandName,releaseYear,discontinuedYear)

Pk: brandID

1. carDealer (dealerId,dealerName,location)

Pk: dealerId

1. Employee (employeeId,branchId,fName,lName,street,Town,county,salary,DOB,branchName, workingHour)

Pk: employeeId,branchId

1. carOwner (ownerId,fName,lName,street,Town,county,ownerCarId,ownerCarBrand,owenerCarModel,butDate)

Pk: ownerId

1. insuranceProvider (providerId,providerName)

Pk: providerId

1. insurancePolicy (policyId,description)

Pk: policyId

1. RepairStation(stationId,stationName,location)

Pk: stationId

***Step2 Mapping***

rawMatiralSupplier and rawMatiral is \*:\* relation , so we make new relation.

make (matieralId,rawSupplierId)

Primary key:matieralId,rawSupplierId

Foreign key:matieralId reference rawMatiral(matieralId)

Foreign key:rawSupplierId reference rawMatiralSupplier(rawSupplierId)

carPart and rawMatieral is \*:\* relation,so we make new relation.

proFrom(partId,matieralId)

Primary key:matieralId,partId

Foreign key:matieralId reference rawMatiral(matieralId)

Foreign key:partId reference carPart(partId)

carPart and partSupplier is \*:\* relation,so we make new relation.

proBy(partId,partSupplierId)

Primary key:partId,partSupplierId

Foreign key:partId reference carPart(partId)

Foreign key:partSupplierId reference partSupplier(partSupplierId)

carPart and carManufacturer is \*:\* relation,so we make new relation.

buyFrom(partId,manufacturerId)

Primary key:partId,manufacturerId

Foreign key:partId reference carPart(partId)

Foreign key:manufacturerId reference carManufacturer(manufacturerId)

carManufacturer and carBrand is 1:\* relation ,and carManufacturer is 1 side,carBrand is \* side ,so carManufacturer is parent carBrand is child.

carBrand (brandID,brandName,releaseYear,discontinuedYear,manufacturerId)

Primary key: brandID

Foreign key:manufacturerId reference carManufacturer(manufacturerId)

carDealer and carBrand is 1:\* relation ,and carBrand is 1 side,carDealer is \* side ,so carBrand is parent, carDealer is child.

carDealer (dealerId,dealerName,location,brandId)

Primary key: dealerId

Foreign key:brandId reference carBrand(brandId)

Employee is recursive relation,so we add new pre attributes.

Employee (employeeId,fName,lName,street,Town,county,salary,DOB,supervisor)

Primary key: employeeId

Foreign key:supervisor reference employee(supervisor)

employeeFamMem is weak entity and depend on Employee,so we add Employee’s pk into employeeFamMem

employeeFamMem(fName,lName,DOB,employeeId)

Primary key: employeeId

Foreign key:employeeId reference employee(employeeId)

Employee and carOwner is 1:\* relation ,and Employee is 1 side,carOwner is \* side ,so Employee is parent ,carOwner is child.

carOwner(ownerId,fName,lName,street,Town,county,ownerCarId,ownerCarBrand,owenerCarModel,butDate,employeeId)

Primary key:ownerId

Foreign key:employeeId reference employee(employeeId)

carOwner and repairStation is \*:\* relation,so we make new relation.

Repair(repairDate,stationId,ownerId)

Primary key:stationId,ownerId

Foreign key:ownerId referemce carOwner(ownerId)

Foreign key:stationId referemce repairStation(stationId)

carOwner and insuranceProvider is \*:\* relation,so we make new relation.

Insure(ownerId,providerId)

Primary key:ownerId,providerId

Foreign key:ownerId referemce carOwner(ownerId)

Foreign key:providerId referemce insuranceProvider(providerId)

insurancePolicy and insuranceProvider is 1:\*relation,so we make new relation

Provide(providerId,policyId)

Primary key:providerId,policyId

Foreign key:providerId referemce insuranceProvider(providerId)

Foreign key:policyId referemce insurancePolicy(policyId)

carDealer and employee is 1:1 relation,and Mandatory participation on both sides，so both side can be parent,there we choose employee as parent.

carDealer (dealerId,dealerName,location,brandId,employeeId)

Primary key: dealerId

Foreign key:brandId reference carBrand(brandId)

Foreign key:employeeId reference employee(employeeId)

Employee and carDealer is 1:\* relation, employee is \* side,carDealer is 1 side.

Employee (employeeId,fName,lName,street,Town,county,salary,DOB,supervisor,dealerId)

Primary key: employeeId

Foreign key:supervisor reference employee(supervisor)

Foreign key:dealerId reference carDealer(dealerId)

***Step3 Normalization***

**UNF—→1NF**

1. phoneNumber and carPartNeed in carManufacturer are multi-valued attributes

So we need to create new relations for them.

manufacturerPhone(phoneNumber,manufacturerId)

Pk:phoneNumber

Foreigner key manufacturer references carManufacturer(manufactureId)

manufacturerPart(carPartNeed,manufacturerId)

Pk:carPartNeed

Foreigner key manufacturer references carManufacturer(manufactureId)

1. rawMaterialNeed in carPart is a multi-valued attribute

So we need to create new relations for it.

carMaterialNeed(rawMaterialNeed,partId)

Pk:rawMaterialNeed

Foreigner key partId references carPart(partId)

1. carModel in carBrand is a multi-valued attribute

So we need to create new relations for it.

Models(carModel,brandID)

Pk:carModel

Foreigner key brandID references carBrand(brandID)

1. brandSold in carDealer is a multi-valued attribute

So we need to create new relations for it.

haveSold(brandSold,dealerId)

Pk:brandSold

Foreigner key dealerId references carDealer(dealerId)

1. serveBrand in repairStation is a multi-valued attribute

So we need to create new relations for it.

haveServed(serveBrand,stationId)

Pk:serveBrand

Foreigner key stationId references repairStation(stationId)

1. phone in carOwner is a multi-valued attribute

So we need to create new relations for it.

ownerPhone(phone,ownerId)

Pk:phone

Foreigner key ownerId references carOwner(ownerId)

1. phoneNumber and insuranceType in insuranceProvider are multi-valued attributes

So we need to create new relations for them.

insurancePhone(phoneNumber,providerId)

Pk:phoneNumber

Foreigner key providerId references insuranceProvider(providerId)

provideType(insuranceType,providerId)

Pk:insuranceType

Foreigner key providerId references insuranceProvider(providerId)

**1NF—→2NF**

In entity Employee,there are some problems we need to normalize

Attributes fName,lName,street,town,county,salary and DOB are worked out from only employeeId

Attribute branchName is worked out from only branchId

attribute workingHour is worked out from employeeId and branchId

So we need to separate Employee

Employee (employeeId,fName,lName,street,Town,county,salary,DOB,supervisor,dealerId)

Primary key: employeeId

Foreign key:supervisor reference employee(supervisor)

Foreign key:dealerId reference carDealer(dealerId)

branch(branchId,branchName)

Pk:branchId

workingHours(employeeId,branchId,workingHour)

Pk:employeeId,branchId

Foreigner key employeeId references employee(employeeId)

Foreigner key branchId references branch(branchId)

**2NF—→3NF**

In entity carOwner,there are some problems we need to normalize

Attributes ownerCarBrand,ownerCarModel and buyDate are worked out from ownerCarId

But ownerCarId is also worked out from ownerId

So we need to separate carOwner.

carOwner(ownerId,fName,lName,street,Town,county)

Pk:ownerId

ownerCar(ownerId,ownerCarBrand,ownerCarModel,buyDate)

Pk:ownerCarId

***Step4 Full of all set***

rawMaterialSupplier (rawSupplierId,rawSupplierName,rawMaterial)

Primary key: rawSupplierId

make (matieralId,rawSupplierId)

Primary key:matieralId,rawSupplierId

Foreign key:matieralId reference rawMatiral(matieralId)

Foreign key:rawSupplierId reference rawMatiralSupplier(rawSupplierId)

rawMaterial (materialId,materialName,materialType)

Primary key: materialId

proFrom(partId,matieralId)

Primary key:matieralId,partId

Foreign key:matieralId reference rawMatiral(matieralId)

Foreign key:partId reference carPart(partId)

carPart (partId,partName,partType)

Primary key: partId

carMaterialNeed(rawMaterialNeedId, rawMaterialNeed,partId)

Pk:rawMaterialNeedId

Foreigner key partId references carPart(partId)

proBy(partId,partSupplierId)

Primary key:partId,partSupplierId

Foreign key:partId reference carPart(partId)

Foreign key:partSupplierId reference partSupplier(partSupplierId)

partSupplier (partSupplierId,supplierName,supplierPhone)

Primary key: partSupplier

buyFrom(partId,manufacturerId)

Primary key:partId,manufacturerId

Foreign key:partId reference carPart(partId)

Foreign key:manufacturerId reference carManufacturer(manufacturerId)

carManufacturer (manufacturerId,manufacturerName,yearFounded,location)

Primary key: manufacturerId

manufacturerPhone(phoneNumber,manufacturerId)

Pk:phoneNumber

Foreigner key manufacturer references carManufacturer(manufactureId)

manufacturerPart(needPartId, partName,manufacturerId)

Primary key:needPartId

Foreigner key manufacturer references carManufacturer(manufactureId)

carBrand (brandID,brandName,releaseYear,discontinuedYear,manufacturerId)

Primary key: brandID

Foreign key:manufacturerId reference carManufacturer(manufacturerId)

Models(carModelId, carModel,brandID)

Primary key:carModelId

Foreigner key brandID references carBrand(brandID)

carDealer (dealerId,dealerName,location,brandId,employeeId)

Primary key: dealerId

Foreign key:brandId reference carBrand(brandId)

Foreign key:employeeId reference employee(employeeId)

haveSold(brandSoldId, brandSold,dealerId)

Primary key:brandSoldId

Foreigner key dealerId references carDealer(dealerId)

Employee (employeeId,fName,lName,street,Town,county,salary,DOB,supervisor,dealerId)

Primary key: employeeId

Foreign key:supervisor reference employee(supervisor)

Foreign key:dealerId reference carDealer(dealerId)

branch(branchId,branchName)

Pk:branchId

workingHours(employeeId,branchId,workingHour)

Pk:employeeId,branchId

Foreigner key employeeId references employee(employeeId)

Foreigner key branchId references branch(branchId)

employeeFamMem(fName,lName,DOB,employeeId)

Primary key: employeeId

Foreign key:employeeId reference employee(employeeId)

carOwner(ownerId,fName,lName,street,Town,county,employeeId)

Primary key:ownerId

Foreign key:employeeId reference employee(employeeId)

ownerPhone(phoneNum, phone,ownerId)

Pk:phoneNum

Foreigner key ownerId references carOwner(ownerId)

ownerCar(ownerCarId,ownerCarBrand,ownerCarModel,buyDate,ownerId)

Pk:ownerCarId

Foreign key: ownerId references carOwner(ownerId)

RepairRel(repairDate,stationId,ownerId)

Primary key:stationId,ownerId

Foreign key:ownerId referemce carOwner(ownerId)

Foreign key:stationId referemce repairStation(stationId)

RepairStation(stationId,stationName,location)

Pk: stationId

haveServed(serveBrandId, serveBrand,stationId)

Pk:serveBrandId

Foreigner key stationId references repairStation(stationId)

insure(providerId,ownerId)

Primary key:providerId,ownerId

Foreign key:providerId referemce insuranceProvider(providerId)

Foreign key:ownerIdreferemce carOwner(ownerId)

insuranceProvider (providerId,providerName)

Pk: providerId

insurancePhone(phoneNumber,providerId)

Pk:phoneNumber

Foreigner key providerId references insuranceProvider(providerId)

provideType(insuranceType,providerId)

Pk:insuranceType

Foreigner key providerId references insuranceProvider(providerId)

Provide(providerId,policyId)

Primary key:providerId,policyId

Foreign key:providerId referemce insuranceProvider(providerId)

Foreign key:policyId referemce insurancePolicy(policyId)

insurancePolicy (policyId,descriptionOfPolicy)

Pk: policyId