

2. Релационен модел на данни

Първо преобразуваме isa-йерархиите, после множествата от същности, после връзките много-много и накрая връзките много-един. Връзките много-един оптимизираме.

За преобразуването на isa-йерархия може да използваме три подхода: E/R, ООП и NULL. За isa-йерархията Clients-Owners-Buyers, най-подходящи са ООП и NULL-подхода, защото може собственик на автомобил да бъде и купувач на автомобил в автосалона.

Ако използваме NULL подхода всички клиенти – купувачи и собственици ще бъдат в една таблица. В таблицата няма да имаме атрибут, по който да различаваме клиента дали е купувач или собственик, но по връзките с другите таблици ще става ясно дали клиента е купувач или собственик. Ако използваме ООП подхода ще имаме една таблица за клиентите- собственици една таблица за клиентите-купувачи и една таблица за тези клиенти, които са едновременно и собственици и купувачи, т.е. ще имаме три таблици, в които ще бъде информацията за клиентите на агенцията. Таблицата само с клиенти, които не са нито купувачи, нито собственици не я разглеждаме, защото в автосалона няма такива клиенти. За преобразуването на тази isa-йерархия ще използваме NULL подхода. Така получаваме само една релационна схема:

Clients (phoneNumber, name, email, location).

За преобразуването на isa-йерархията за автомобилите (Automobiles) подходящи са ООП подхода и ER подхода.

В базата от данни на автосалона не може да има автомобили, които да не са нито леки автомобили, нито камиони, нито автобуси, както и не може да има автомобили,

които да са едновременно лек автомобил и камион или лек автомобил и автобус или камион и автобус. Така ако използваме ООП подхода от 8-те възможни (2 на степен броя на наследниците, в случая 3) релационни схеми за isa-йерархията с Automobiles , смисъл имат само три: Cars, Trucks и Buses. Като използваме ООП-подхода и игнорираме празните таблици (релационните схеми, в които никога няма да има редове) получаваме:

1. Automobiles (brand, category, colour, price, horsePower, euroEmissionStandard, yearManufacture, engineType , transmissionType, spareTyre) **X (не създаваме таблица, защото няма такива)**

2. AutomobilesCars(clientPhoneNumber , brand, category, colour, price, horsePower, euroEmissionStandard, yearManufacture, engineType , transmissionType, spareTyre, spoiler, sunroof)

3. AutomobilesTrucks(clientPhoneNumber , brand, category, colour, price, horsePower, euroEmissionStandard, yearManufacture, engineType , transmissionType, spareTyre, loadCapacity, numberOfAxles, numberOfSeats)

4. AutomobilesBuses(clientPhoneNumber , brand, category, colour, price, horsePower, euroEmissionStandard, yearManufacture, engineType , transmissionType, spareTyre, sunroof, loadCapacity, numberOfAxles, numberOfSeats)

5. AutomobilesCarsTrucks(атрибутите на Automobiles, Cars и Trucks) **X (не създаваме таблица, защото няма такива)**

6. AutomobilesCarsBuses(атрибутите на Automobiles, Cars и Buses) **X (не създаваме таблица, защото няма такива)**

7. AutomobilesTrucksBuses(атрибутите на Automobiles, Trucks и Buses) **X (не създаваме таблица, защото няма такива)**

8. AutomobilesCarsTrucksBuses(атрибутите на Automobiles, Cars, Trucks и Buses) **X (не създаваме таблица, защото няма такива)**

Може да преименуваме релационните схеми AutomobilesCars на Cars, AutomobilesTrucks на Trucks и AutomobilesBuses на Buses.

Ако изберем E/R към REL подход ще имаме 4 таблици в БД и всеки път когато ни трябва повече информация за даден автомобил трябва да връзваме две таблици Automobiles и съответния автомобил.

Клиентите купувачи обаче най-често търсят по определни параметри автомобила – да бъде дизел, да има ръчна скоростна кутия и т.н., тогава ООП – подхода е по-подходящ, защото в една таблица ще има цялата информация. За разлика при E/R подхода ще трябва да се свързват всеки път по две таблици Automobiles и съответния автомобил.

За решението на нашата задача за Automobiles ще използваме ООП подхода.

За isa-йерархията Staff-Sales Assistants-Mechanics, най-подходящи са ООП и NULL-подхода.

Ако използваме NULL подхода всички асистенти-продавачи и монтьори ще бъдат в една таблица. В таблицата няма да имаме атрибут, по който да различаваме служителя дали е асистент по продажби или монтьор, но по връзките с другите таблици ще става ясно дали служителя е асистент по продажби или монтьор. Така получаваме само една

релационна схема:

Staff(**workNumber**,name,phoneNumber,region)

След преобразуване на трите isa-йерархии получихме релационните схеми:

- Clients (**phoneNumber**, name, email,location)
- Staff(**workNumber**,name,phoneNumber,region)
- Cars(**clientPhoneNumber** , **brand**, category, colour, price, horsePower,euroEmissionStandard,yearManufacture,engineType , transmissionType, spareTyre,spoiler,sunroof)
- Trucks(**clientPhoneNumber** , **brand**, category, colour, price, horsePower,euroEmissionStandard,yearManufacture,engineType , transmissionType, spareTyre,loadCapacity,numberOfAxles,numberOfSeats)
- Buses(**clientPhoneNumber** , **brand**, category, colour, price, horsePower,euroEmissionStandard,yearManufacture,engineType , transmissionType, spareTyre,sunroof,loadCapacity,numberOfAxles,numberOfSeats)

Продължаваме с другите множества от същности и връзки:

- Autostores(**name**,address)
- Services(**type**,price)
- WorksIn(**name** ,**workNumber**) X (оптизираме)
- GoesToSuggest(**clientPhoneNumber**,**autostoreName**, priceAutomobile) X (оптизираме)
- Receives(**clientPhoneNumber**,**type**)

- Requires(clientPhoneNumber, autostoreName, type, maxPrice)
- Fulfills(staffWorkNumber, servicesType) X (оптизираме)
- ServesForPurchase(clientsPhoneNumber, staffWorkNumber) X (оптизираме)

Окончателно схемата на базата от данни се състои от следните релационни схеми:

- Clients (phoneNumber :INT NOT NULL, name : VARCHAR(50), email : VARCHAR(30), location : VARCHAR(40), staffWorkNumber, autostoreName, priceAutomobile)
- Staff(workNumber, name : VARCHAR(15) NOT NULL, phoneNumber : VARCHAR(10), region : VARCHAR(40), autostoreName : VARCHAR(20) NOT NULL, servicesType)
- Cars(clientPhoneNumber : INT NOT NULL , brand : VARCHAR(20) NOT NULL , category : VARCHAR(10) NOT NULL, price : FLOAT NOT NULL, colour : VARCHAR(15), horsePower : INT, euroEmissionStandard : SMALLINT , yearManufacture : INT, engineType : VARCHAR(10), transmissionType : VARCHAR(15), spareTyre : SMALLINT, spoiler : SMALLINT, sunroof : SMALLINT)
- Trucks(clientPhoneNumber : INT NOT NULL , brand : VARCHAR(20) NOT NULL , category : VARCHAR(10) NOT NULL, price : FLOAT NOT NULL, colour : VARCHAR(15), horsePower : INT, euroEmissionStandard : SMALLINT , yearManufacture : INT, engineType : VARCHAR(10), transmissionType : VARCHAR(15), spareTyre : SMALLINT,

loadCapacity : FLOAT,numberOfAxles :
SMALLINT,numberOfSeats : SMALLINT)

- Buses(**clientPhoneNumber** : INT NOT NULL , **brand** :
VARCHAR(20) NOT NULL , category : VARCHAR(10) NOT
NULL, price : FLOAT NOT NULL, colour : VARCHAR(15),
horsePower : INT,euroEmissionStandard : SMALLINT
,yearManufacture : INT,engineType : VARCHAR(10),
transmissionType : VARCHAR(15), spareTyre : SMALLINT,
sunroof : SMALLINT, loadCapacity : FLOAT,numberOfAxles :
SMALLINT,numberOfSeats : SMALLINT)

- Autostores(**name** : VARCHAR(20),address : VARCHAR (40))

- Services(**type** : VARCHAR(30) ,price : FLOAT)

- Receives(**clientsPhoneNumber** :
VARCHAR(10),**servicesType** : VARCHAR(30))

- Requires(**clientsPhoneNumber** : VARCHAR(10),
autostoreName : VARCHAR(20),type: VARCHAR(30),maxPrice :
FLOAT)