SRS Documentation

Notown Record Management System

Prepared by,

DEBOPRIYO GHOSH	Roll: 35
SAIKAT JANA	Roll: 50
MOUSUMI MONDAL	Roll: 40
SARADINDU RANA	Roll: 28

B.Tech., Sem - V, Group- 1

Dated: 07/04/2021

Contents

Draw the ER diagram; Design the relational schema with minimum redundancy.

Notown Records has decided to store information about musicians who perform on its albums (as well as other company data) in a database. The company has wisely chosen to hire you as a database designer (at your usual consulting fee of $2,500/day).

Each musician that records at Notown has an SSN, a name, an address, and a phone number. Poorly paid musicians often share the same address, and no address has more than one phone.
Each instrument that is used in songs recorded at Notown has a name (e.g., guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, E-flat).
Each album that is recorded on the Notown label has a title, a copyright date, a format (e.g., CD or MC), and an album identifier.
Each song recorded at Notown has a title and an author.
Each musician may play several instruments, and a given instrument may be played by several musicians.
Each album has a number of songs on it, but no song may appear on more than one album.
Each song is performed by one or more musicians, and a musician may perform a number of songs.
Each album has exactly one musician who acts as its producer. A musician may produce several albums, of course.

1.Data Requirement

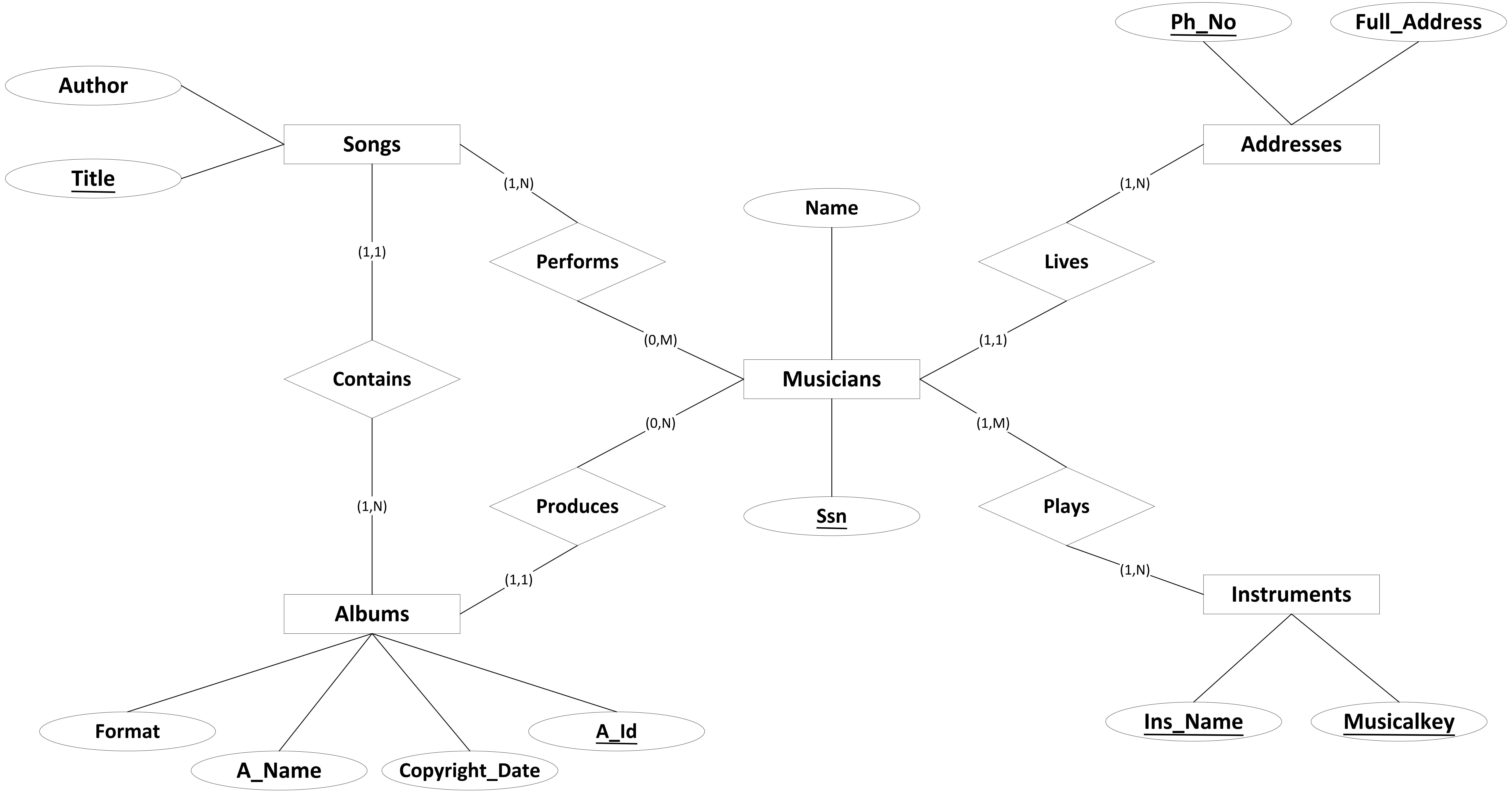
- **Songs:** Every song has a unique title. Each song is written by specific author.
- **Albums:** Every album has an unique id and respective album name. Eachone has specific format (e.g. CD or MC) and date of copyright.
- **Musicians:** Every musician has an unique Ssn number and their respective name .
- **Instruments:** Each instrument is identified by an unique name (e.g. guiter, synthesizer, flute) and its musical key (e.g. C,B-flat ,E-flat).
- **Addresses:** Each address of musicians identified by a contact phone number. Each one has full address.

2. Assumption

1) An album may contain multiple songs. But the recorded albums has atleast one song.
2) A musician may or may not perform songs . Musician may produce one or more albums.
3) A musician can play one or more instruments.
4) A musician lives only in one address identified by it's phone number.
5) In one address there may be more than one poorly paid musicians.
6) An instrument can be played by one or more than one customer.
7) Every album has only one producer.
8) A song can be performed by one or more than one musician.

3. Entity Relationship Diagram

A. Diagram


Ph_No
Full_Address
Author
Songs
Addresses
Title
(1,N)
(1,N)
Name
(1,1)
Performs
Lives
(0,M)
(1,1)
Musicians
Contains
(0,N)
(1,M)
Produces
Plays
(1,N)
Ssn
(1,N)
(1,1)
Instruments
Albums
Format
A_Name
Copyright_Date
A_Id
Ins_Name
Musicalkey

B. Description

a) In this diagram the entities are Albums, Musicians , Songs , Instruments , Addresses.

b) Albums contains songs so they are connected by the relationship ‘Contains’.

c) Musicians performs songs. They are connected by the relationship ‘Performs’.

d) Musicians produces album, so they are connected by the relationship ‘Produces’.

e) A musician lives in an addresses. So musician and addresses are connected by the relationship ‘Lives’.

f) Musician plays instruments, so musician and instruments is connected by the relationship ‘Plays’.

C. Attributes

Albums: {A_Id, A_Name, Format,Copyright_Date}
Songs: {Title , Author}
Musicians: {Ssn, Name }
Instruments: {Ins_Name ,MusicalKey}
Addresses: {Ph_no, FullAddress}

4.Relationships

a. Musicians-Addresses (N:1) comprises.
b. Musicians-Instruments (N:M) comprises.
c. Musicians-Albums (1:N) comprises.
d. Musicians-Songs (N:M) comprises.
e. Albums-Songs (1:N) holds.

5.Relational Schema

Albums:

Albums

- <A_Id>
  - Album title: <A_Name>
  - Date of copyright : <Copyright_Date>
  - Album format : <Format>
  - Producer id : <Ssn> (Forgien key from ‘Musicians’ table)

Songs:

Songs

- <Title>
  - Author name : <Author>
  - Album id : <A_Id> (Forgien key from ‘Albums’ table)

Musicians:

Musicians

- <Ssn>
  - Musician name : <Name>
  - Musician contact number : <Ph_No>(Forgien key from ‘Addresses’ table)

Instruments:

Instruments

- <Ins_Name>
    - Musical key type : <Musicalkey>

Addresses:

Addresses

- <Ph_No>
    - Musician address : <FullAddress>

Plays:

Plays

- Instrument name : <Ins_name> (Forgien key from ‘Instruments’ table)
- Musician id : <Ssn> (Forgien key from ‘Musicians’ table)

Performs:

Performs

- Musician id : <Ssn> (Forgien key from ‘Musicians’ table)
- Song title : <Title> (Forgien key from ‘Songs’ table)

6. Normalization of Relational Schema

I)Musicians {Ssn, Name, Ph_No }

Ssn -> { Name,Ph_No }

Ssn is the candidate key.

There is no multi valued attribute so the table is in 1NF.

Ssn is the primary key and there is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

II)Songs { Title, Author, A_Id }

Title -> { **Author, A_Id** }

Title is the candidate key.

There is no multi valued attribute so the table is in 1NF.

Title is the primary key and there is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

III)Albums{A_id, A_Name, Format, Copyright_Date, Ssn}

A_Id -> { A_name, Format, Copyright_Date ,Ssn}

A_Id is the candidate key.

There is no multi valued attribute so the table is in 1NF.

A_Id is the primary key and there is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

IV)Instruments { Ins_Name, Musicalkey}

(Ins_Name, MusicalKey} is the composite primary key.

There is no multi valued attribute so the table is in 1NF.

There is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

V)Addresses {Ph_No, FullAddress}

Ph_No -> { FullAddress }

Ph_No is the candidate key.

There is no multi valued attribute so the table is in 1NF.

Ph_No is the primary key and there is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

VI)Performs{Ssn,Title}

{Ssn,title} is a composite primary key.

There is no multi valued attribute so the table is in 1NF.

There is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.

VII)Plays{Ssn,Ins_Name}

{Ssn, Ins_Name} is a composite primary key.

There is no multi valued attribute so the table is in 1NF.

There is no partial dependency, so the relation is in 2NF.

The table is in 3NF as there is no transitive dependency.