# Model ER choices

Every entity contains only its own attributes (*ID*, *name, etc.* but not the *ID* of other entities). We chose to create relations instead of giving the *ID* of an other entities as an attribute of the entity.

This relation design is especially suitable in this case because if several entities share their *ID* this means that a relation exists between them.

The *Song* relation is a ternary relation that represents a song on a support. It relates a support (*Medium*) with the position on this medium (*Track*) and the recording of the song (*Recording*).

*Physical\_Song* contains all the pairs of *Release* (album, single…) and on which support it is displayed.

Those choices are made in consideration of the given data. For example, for *Physical\_Song* relationship there is an existing relationship between Medium and Release. Medium should contain the ID of Release, so we made a relation of it.

# Constraints Explanation

There is some complex constraints between entities because the musical universe is also complex: collaborations (multiple artists per track), compilations (multiple artists per album), etc.

It is difficult to define more constraints because the available data are often incomplete.

## Area

An artist can have at most one area. This constraint is given by the available data. Indeed, instinctively we may think that one artist have to come from one or more than one area (for example: New-York (*City*) and America (*Continent*)). But in the given data, the situation is different: an artist can have only one area or no area if the data is missing.

## Tracks

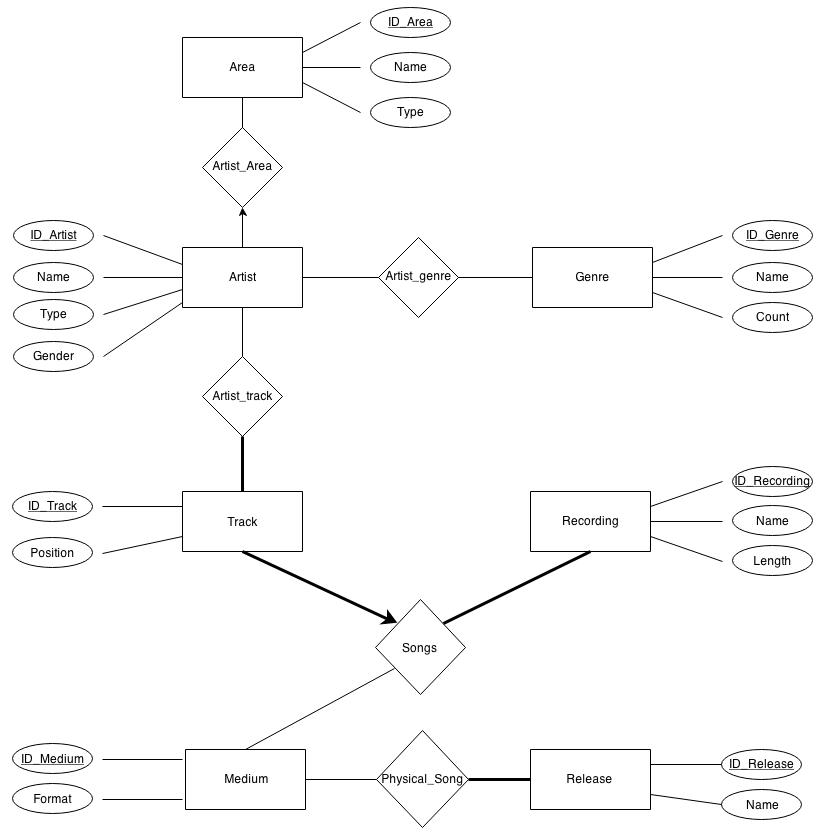
All tracks have to be represented in the relation *Artist\_Track* at least once. It means that all tracks were made by at least one artist.

All tracks have to be represented exactly once in the relation *Song*. It means that all tracks (all positions in a CD for example) are related to exactly one music (i.e there is no position without a music) and one support.

## Recordings

All recordings have to be at least once represented in the *Song* relation. Indeed, recording is the logical song. It does not make sense if it does not stand on any support. But the same recording may be on different supports, and be referred by several tracks etc.

## Realeases

All releases must be at least once represented in the *Physical\_Song* relation. For the same reason as for recording. It does not make sense that an album, single … does not stand on any support.

**Schéma 1:**

**Area** : ID\_Area, Name, Type

**Artist** : ID\_Artist, Name, Type, Gender

**Genre** : ID\_Genre, Name, Count

**Medium** : ID\_Medium, Format

**Recording** : ID\_Recording, Name, Length

**Release** : ID\_Release, Name

**Track** : ID\_Track, Position

**Artist\_Area** : ID\_Artist, ID\_Area

**Artist\_Genre** : ID\_Artist, ID\_Genre

**Artist\_Track** : ID\_Artist, ID\_Track

**Physical\_Song** : ID\_Medium, ID\_Recording

**Song** : ID\_Track, ID\_Medium, ID\_Recording

## Table Creation

-- Area :

CREATE TABLE Area

(ID\_Area CHAR(60),

Name CHAR(60),

Type CHAR(60),

PRIMARY KEY(ID\_Area))

-- Artist :

CREATE TABLE Artist

(ID\_Artist CHAR(60),

Name CHAR(60),

Type CHAR(60),

Gender CHAR(60),

PRIMARY KEY(ID\_Artist))

-- Genre :

CREATE TABLE Genre

(ID\_Genre CHAR(60),

Name CHAR(60),

Count CHAR(60),

PRIMARY KEY(ID\_Genre))

-- Medium :

CREATE TABLE Medium

(ID\_Medium CHAR(60),

Format CHAR(60),

PRIMARY KEY(ID\_Medium))

-- Recording :

CREATE TABLE Recording

(ID\_Recording CHAR(60),

Name CHAR(60),

Length CHAR(60),

PRIMARY KEY(ID\_Recording))

-- Release :

CREATE TABLE Release

(ID\_Release CHAR(60),

Name CHAR(60),

PRIMARY KEY(ID\_Release))

-- Track :

CREATE TABLE Track

(ID\_Track CHAR(60),

Position CHAR(60),

PRIMARY KEY(ID\_Track))

-- Artist\_Area :

CREATE TABLE Artist\_Area

(ID\_Artist CHAR(60),

ID\_Area CHAR(60),

PRIMARY KEY(ID\_Artist),

FOREIGN KEY(ID\_Artist) REFERENCES Artist,

FOREIGN KEY(ID\_Area) REFERENCES Area)

-- Artist\_Genre :

CREATE TABLE Artist\_Genre

(ID\_Artist CHAR(60),

ID\_Genre CHAR(60),

PRIMARY KEY(ID\_Artist, ID\_Genre),

FOREIGN KEY(ID\_Artist) REFERENCES Artist,

FOREIGN KEY(ID\_Genre) REFERENCES Genre)

-- Artist\_Track :

CREATE TABLE Artist\_Track

(ID\_Artist CHAR(60),

ID\_Track CHAR(60),

PRIMARY KEY(ID\_Artist, ID\_Track),

FOREIGN KEY(ID\_Artist) REFERENCES Artist,

FOREIGN KEY(ID\_Track) REFERENCES Track)

-- Physical\_Song :

CREATE TABLE Physical\_Song

(ID\_Medium CHAR(60),

ID\_Release CHAR(60),

PRIMARY KEY(ID\_Medium, ID\_Release),

FOREIGN KEY(ID\_Medium) REFERENCES Medium,

FOREIGN KEY(ID\_Release) REFERENCES Release)

-- Song :

CREATE TABLE Song

(ID\_Track CHAR(60),

ID\_Recording CHAR(60),

ID\_Medium CHAR(60),

PRIMARY KEY(ID\_Track),

FOREIGN KEY(ID\_Medium) REFERENCES Medium,

FOREIGN KEY(ID\_Track) REFERENCES Track,

FOREIGN KEY(ID\_Recording) REFERENCES Recording)