

ER diagram normalization

Group project - CSCI466

Converting entities into relations: each strong, non-subtype entity becomes its own relation (primary keys underlined)

Song(songID, title)

Artist(artistID, name)

User(userID, name, timeSung)

KaraokeVersion(versionID, versionName)

Handling relationships:

- has (1,1) <-> (1,m) links song to karaoke version files
- workedOn (1,m) <-> (0,m) links song entity to artist entity
- freeQueuedIn (0,m) <-> (0,m) links user to the song they are free queuing in
- paidQueuedIn (0,m) <-> (0,m) links user to the song they are paid queuing in

Converting relationships into entities: one to many relationships are added to existing tables. many to many relationships become their own entity, with the primary key of the new relation being a concatenation of the identifiers of the entities this relationship links. (Foreign key represented by *)

KaraokeVersion(versionID, songID*, versionName)

WorkedOn(songID*, artistID*, contribution)

FreeQueuedIn(versionID*, userID*, dateTime)

PaidQueuedIn(versionID*, userID*, dateTime, amountPaid)

Final Relations:

Song(songID, title)

Artist(artistID, name)

User(userID, name, timeSung)

KaraokeVersion(versionID, songID*, versionName)

WorkedOn(songID*, artistID*, contribution)

FreeQueuedIn(versionID*, userID*, dateTime)

PaidQueuedIn(versionID*, userID*, dateTime, amountPaid)