ENSF 608 – Final Project

Cirque Du Soleil Database

03 Dec 2023

Chris Brunet

Sean Buchanan

Christopher Proc

Design Decisions

Tables – The design used 13 tables including:

- Three representing the major entities in our system and their attributes (Performer, Production, and Venue).
- Two representing relationships between those entities (Schedule, Crew)
- One representing the weak entity of Emergency Contact
- Three representing specializations of performer (Musician, Entertainer, Aerialist)
- Three representing multi-valued attributes (Instrument, Aerial Equipment, Medication)
- One to enforce strict performer categories – Performer Type.

Assumptions:

- We assume that a production is only scheduled at a single venue and location. Different productions could be simultaneously hosted at the same venue with a different location.
- We assume that a venue can host the same production multiple times.
- We assume that performer types are totally disjoint – no overlaps.
- We assume that an aerialist has only one sport, but may use multiple pieces of equipment.

Disjoint specialization - We used method 8A – subclass tables to handle the extra attributes, with a type attribute in the main table. This was chosen over 8B for two reasons: One, it allowed us to use auto-increment for performer_id numbers while still guaranteeing unique and mappable ids in the sub-classes. And two, it centralizes the majority of common attributes shared between performer types into one table for easier queries, and allows us to extend the database in the future for overlapping performer types – we can simply add flags for each type and register a performer in multiple sub-tables.

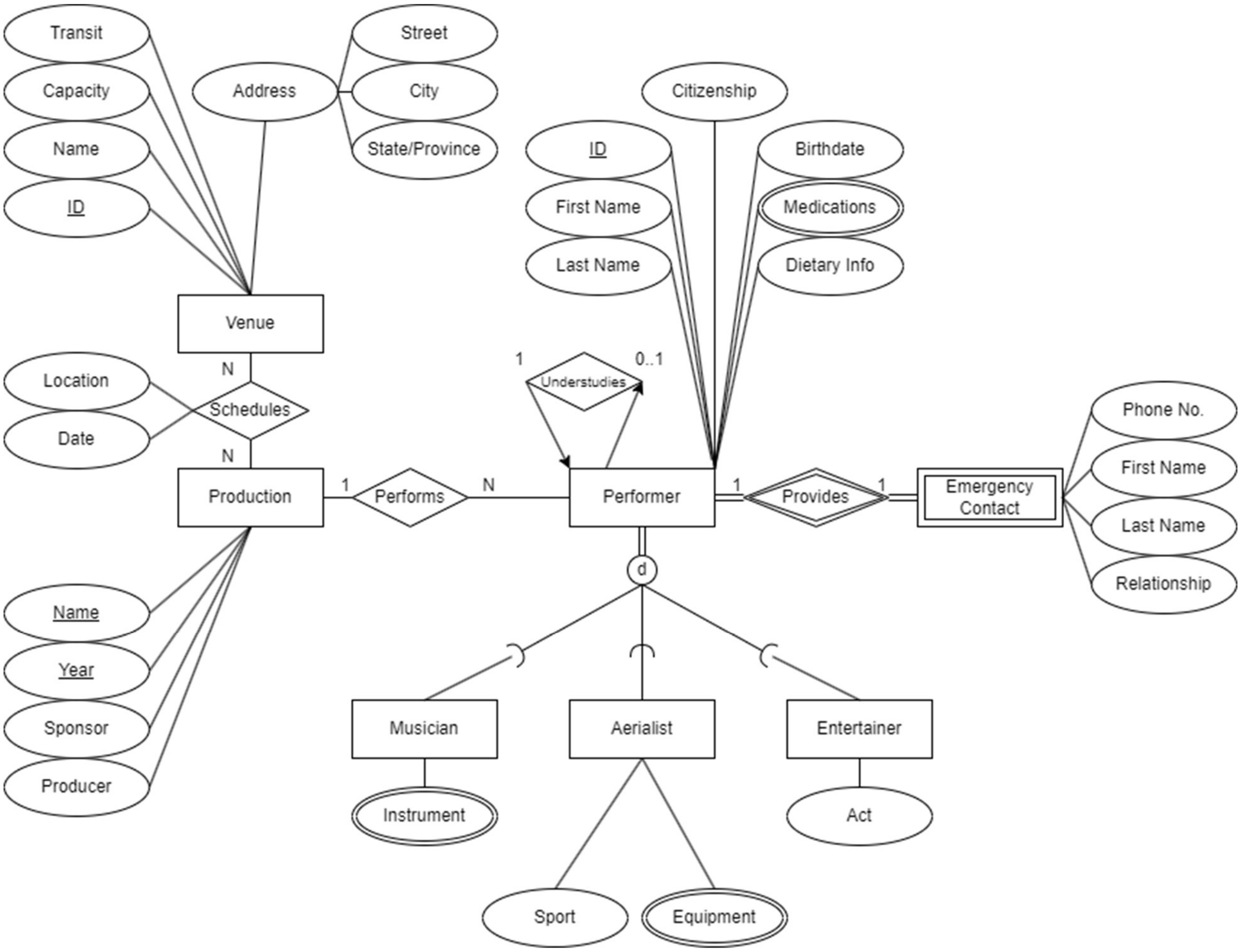
Foreign Key constraints are used on Medication, Emergency Contact, Subclass Tables, and Equipment tables in order to enforce the relationship with a specific performer. Foreign Key constraints are used on Schedule and Crew to enforce the relationships with specific Venues, Productions, and Performers.

Primary Key constraints - Schedule has four attributes in the primary key in order to satisfy our assumption that a venue can host a show multiple times. Due to the 1-1 relationship between Performer and Emergency Contact, we use performer_id as the primary key of emergency contact.

Triggers were used for this database to allow us to automatically insert, or reassign performers to the sub-class tables when created or updated.

Other design decisions:

- Deletions on Performer, or Production will cascade to the tables that reference them as foreign keys.
- Deletions on Venue will not cascade – you must adjust the schedule first.
- Deletions on Understudy will set to Null.
- The Performer Type table was added in order to strictly enforce the performer_type attribute of the Performer – knowing the possible values for performer allows us to more easily write triggers.


Transit
Capacity
Name
ID
Address
Street
City
State/Province
Venue
N
Location
Date
Schedules
N
Production
1
Performs
N
Name
Year
Sponsor
Producer
Citizenship
ID
First Name
Last Name
Birthdate
Medications
Dietary Info
1
Understudies
0..1
Performer
1
Provides
1
Emergency Contact
Phone No.
First Name
Last Name
Relationship
d
Musician
Aerialist
Entertainer
Instrument
Act
Sport
Equipment