

We mapped each ER entity (for example., Users, Events, Tickets) to its own table with a surrogate primary key ID and the entity's attributes for the columns. For ISA hierarchies of the subentities We used table per subclass: each subtype (for example : Organizer, Attendee) is a separate subtable whose Primary Key is also a Foreign Key to the parent, enforcing existence dependency and avoiding NULLs, We enforce referential integrity with FKs with ON DELETE CASCADE to prevent entities without parents

All the relationships between entities are represented as separate tables that connect the two sides, with a composite primary key formed from the primary keys of the participating entities, and foreign keys enforcing referential integrity. We kept junctions to allow many-to-many and store relationship future attributes if needed.

Instead of having for example a special attribute in each entity, for example instead of having an event that has as an attribute organizer_Id, we have a junction table "creates", to do the job