

In my database, there will be relations between the entities of Players, Seasons, Teams, and Awards. Players participate in a season, and all of their base attributes are the same regardless of what positions they play, but in their relationship with the Season entity, their stats for that season require the Players to be split up into Pitchers and Hitters. To keep Players from becoming too divided, my database will not record defensive stats.

The Award entity will hold individual and team awards. Individual awards will be like Hall of Fame, Gold Glove, etc and team awards will be postseason series wins like Pennant, World Series, etc. These two types of Awards seem similar enough to me in terms of what identifies them, so they will be part of the same entity. The Team/Award relationship is defined by the season it occurs in, as is the Player/Award relationship.

In order to allow Players to join and leave and rejoin teams, which is very common with prospects who cyclically get hot, move to the majors, go cold, move down to AAA, get hot... etc., I am adding a start_year attribute to the Player/Team relationship. This will allow players to have different stints with the same team.

I am planning on keeping track of all of the MLB teams and how long Giants players played on those teams, but I don't plan on keeping track of how those non-Giants teams did during the season, so the Season/Team relation is a many-to-many relationship with the potential to store Teams without a Season.

What I am confused about is whether or not I should include Team into the attributes of Pitcher and Hitter, or if I should make a ternary relationship with Player and Team and Season.