University of Manitoba

Project Report

Data Discovery and Database Design

COMP 3380: Databases Concepts and Usage

Authors:
Garik Avagyan - 7893763
Ivan Balkashynov - 7900097
Tim Sarvis - 7866126
Uday Shekhawat - 7923995

Instructor:	*Anifat Olawoyin*
Department:	Computer Science
Date:	April 11, 2023

Contents

1 Summary of Dataset

The database used in this project is created using data from the 2021 NBA season. Various aspects of the season such as players, coaches, and teams are contained and related within the database.

The data for this database design was chosen because of the relational nature of basketball and the NBA. Since the NBA focuses heavily on relationships and provides an interesting domain for queries, it proved to be a good choice for database implementation.

The datasets themselves are a combination of publicly available datasets and the manual insertion of data to expand and fill each dataset. There are four main datasets from which the database is filled from: Awards, Players, Teams, and Colleges.

Awards.csv

This dataset contains information about the various player-based awards for the 2021 NBA season. This is a small dataset that contains information about the title of awards, which player won the award, and any bonus money that comes from winning the award. An example row in this dataset consists of the following:

Title	Pid	Fname	Lname	Bonus
MVP	450	Nikola	Jokic	1000000

Table 1: Awards.csv

This dataset is 7 rows by 5 columns for a total of 35 entries and file size of 221 bytes.

Players.csv

This dataset contains information about each player in the 2021 NBA season. This is the largest dataset used for this project, containing information for 605 NBA players. For each player, the dataset describes their name, physical information, statistics, nationality, salary, college they played for, what head coach they play for, and any draft information(if they were drafted). An example row in this dataset consists of the following:

Pid	Fname	Lname	Pos	Age	Tm	Stats	Height	Weight	Country
4	Aaron	Nesmith	SF	22	Boston Celtics	...	195.58	97.52228	USA

Salary	CollegeName	Hcid	Draft_year	Draft_round	Draft_number
3631200	Vanderbilt	2	2014	1	4

Table 2: Player.csv

This dataset is 605 rows by 41 columns for a total of 24805 entries and file size of 117KB.

Teams.csv

This dataset contains information about the various teams in the NBA and their associated statistics for the 2021 NBA season. In addition to information about the team itself, this dataset also describes some of the logistical information relating to each team such as their stadium and

where they're located. For each team in the dataset, their name, statistics, head coach, stadium, geographical information, and owner of the team is described. An example row in this dataset consists of the following:

Teamname	Stats	Head Coach	Hcage	Sid	Stadium	Capacity	State
Atlanta Hawks	41	1	58	1	State Farm Arena	18118	Georgia

City	Cid	CEOfname	CEOlname	networth	League Name
Atlanta	1	Tony	Ressler	6.2	NBA

Table 3: Team.csv

This dataset is 30 rows by 23 columns for a total of 690 entries and file size of 5KB.

Colleges.csv

This dataset contains information about the various colleges that NBA players attended. This is the second smallest dataset, only containing the geographical information about each of the colleges. For every college, their country and US state are described. An example row in this dataset consists of the following:

College	Country	State
Alabama	USA	Alabama

Table 4: College.csv

This dataset is 139 rows by 3 columns for a total of 417 entries and file size of 4KB.

2 ER Model

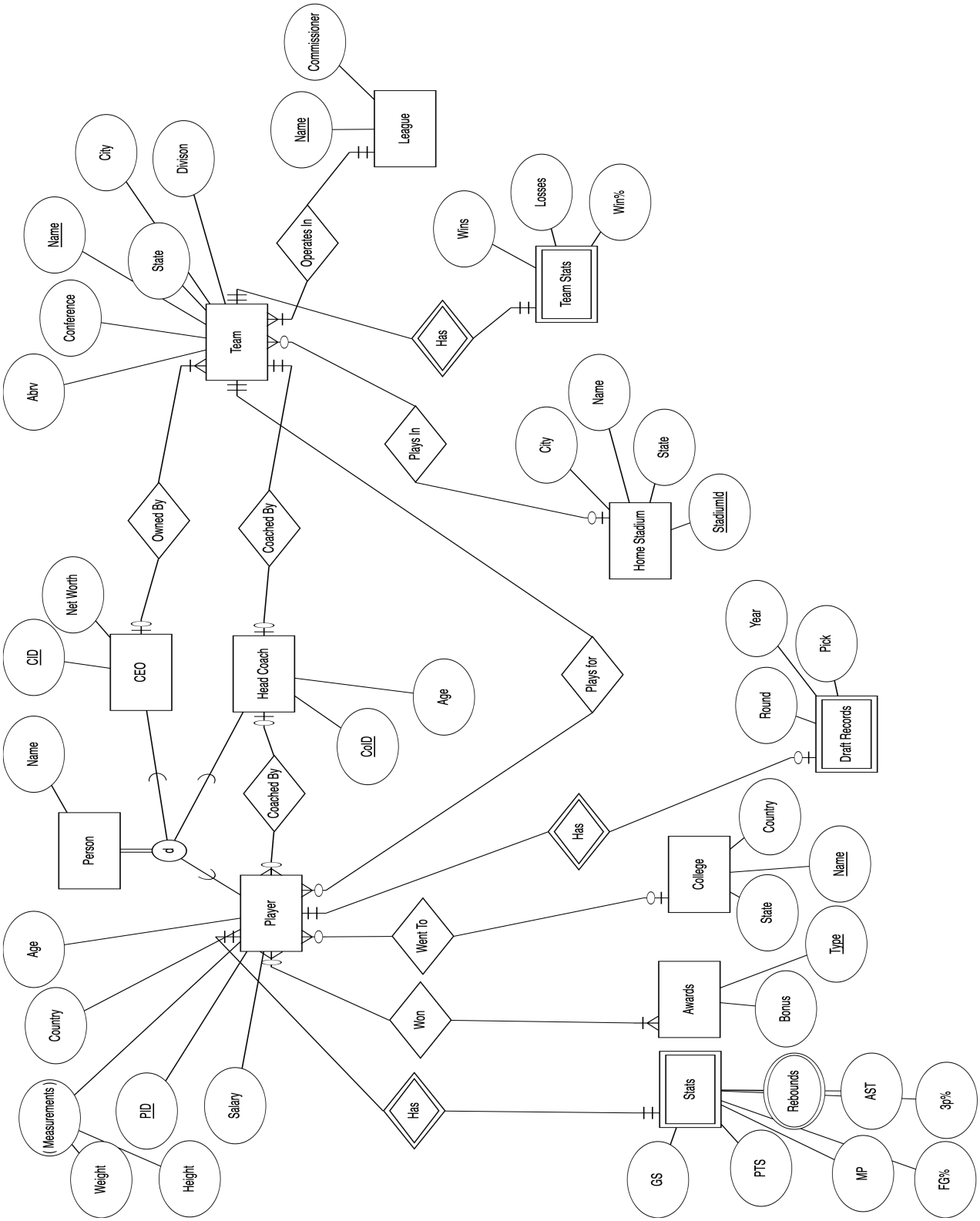

Figure 1: ER Model

Explanation/Assumptions of Various Components

- Rebounds can be a multi-valued attribute since they can include offensive and defensive rebounds.
- For US Colleges, there are no duplicate college names in our dataset so name can be used a primary key.
- For Team, no NBA team can have a duplicate name so name is a suitable primary key. Similar ideas are used for the league.
- Draft records, Stats, and Team Stats have no partial keys

3 Discussion of Data Model

The overarching goal of the database is to provide information surrounding the NBA as a whole rather than detailing specific statistics and details. The process we used to break this data model down into these certain tables started with deriving the two most important components of the NBA: players and teams. Within the NBA, there are teams that play against each other, each using their own roster of players. Since these two components are the focus of our database, we also made them the two largest tables(in terms of attributes). Following that, we then worked on expanding both the players and teams entities such that we covered all relationships that involve them. For the Player table, in addition to playing basketball, players win awards, players get drafted(usually), and players go to a certain college before getting drafted. These three relationships expanded into multiple new tables relating to the Player table. In terms of the Team table, in addition to playing against other teams, teams are coached, teams are owned, teams are physically located somewhere, and teams are part of a given league. These relationships became new tables relating to the Team table. The coaching of a team was also shared with the Player table since players are coached by a Head Coach.

When designing the model, there was some difficulty in determining keys to be used for the various entities mainly due to the logistics of how things worked. One such example, NBA players and NBA coaches have no unique attributes and thus surrogate keys were needed to be inserted.

In terms of difficulty with participation/cardinality ratio of relationships, the only difficulty was with determining the Player ->Team and Coach ->Team relationships. In this relationship, both players and coaches can exist without being a part of any team since they are labelled free agents. However, a team cannot exist without players and/or a coach since they are required(by rules of the league) to have a certain number of players and coaches on the roster. To realise this, a deeper knowledge of the NBA logistics was required.

To translate the data model to a relational model, it was a smooth transition. The datasets, with the help of some manual information addition, supported each of the tables we had designed. Without considering the addition of the surrogate keys, the mapping process to convert an ER model to the relational model was straightforward and simplistic.

The design of the data model from Part 1 was the exact same as produced in Part 3, except for the addition of two attributes and the removal of another attribute. The attribute additions were adding a capacity and stadium id field to the Stadium entity. The capacity attribute opened up some interesting query options by allowing us to rank stadiums in terms of their capacity. The stadium ID was necessary since stadiums could share the same name in the future while being in different locations. The attribute deletion was removing the city attribute from College. From our dataset, we found that there were no duplicate college names and thus the city field wasn't needed for key purposes or any queries we wanted to implement. In terms of regrets, the only regret was the inclusion of the League entity. Since this is a NBA database and likely not to be expanded, all teams should be assumed to be the same league, rendering this entity useless to our queries.

4 Remodelling

Could this data be modelled in a different way, why or why not? Given the work completed, would you choose this model?

This data could be modelled in many different ways due to the fact that the NBA is a deep and complex system of relationships. One possible model could place a larger emphasis on the statistics of various players and teams. While our database attempts to provide information about the NBA as a whole, a more statistical oriented database could provide analysis into the performance of players or teams. For example, in a statistical-based NBA database, possible queries could be "How does player X perform when on the floor with player Y" or "How does team X perform with various permutations of its starting lineup". In addition to this model, a possible model could include the NBA fan and game interactions. This model would include more information about fans, popularity, game attendance, ticket prices, jersey sales, and more. With this model, some questions could be asked such as "What teams have the highest amount of attendance and sales" or "What player is the most popular based on their jersey sales".

With the work already completed, we would continue to use our current data model. Most analysts and common fans are interested in questions that pertain to the NBA as a whole. With our data model, we sufficiently capture that information and provide an avenue for those interested to get the answers to whatever queries they have.

5 Summary of the Interface

The purpose of the interface was to allow possible analysts to answer certain queries with our database in a more intuitive fashion. For the interface, we chose to build a GUI in Python as pictured below.

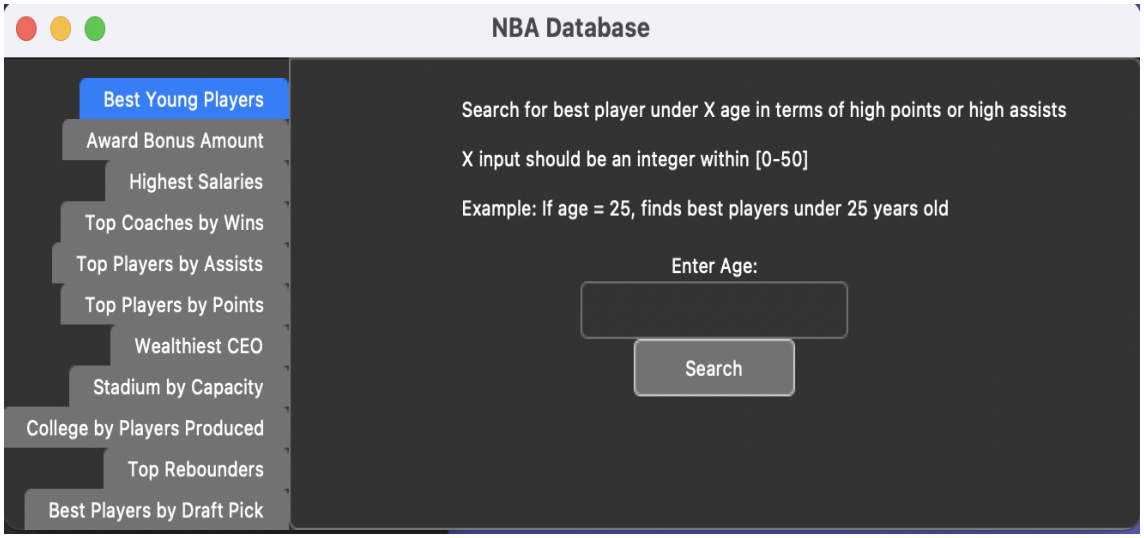

Figure 2: Interface

Within the GUI, we have 11 different tabs representing 11 different queries for analysts to run. For each query, an analyst is allowed to enter in specific information, such as age of players, to only select the specific information they want to analyse.

6 List of Interesting Queries

Our group has created 11 queries that can be implemented using the interface. We have found them interesting, useful and practical for our database.

1. Query that finds the list of colleges that have produced X (entered value) or more players. Returns the college information and sorts the lists by the number of players produced.

2. A query that finds stadiums with a capacity of X (entered value) or more. Sorts by capacity.

3. A query that finds team CEOs with a wealth of X (entered value) or more. The wealth (Net Worth) is displayed in billions. Returns the CEO's, their wealth, and their respective teams. Sorts by wealth.

4. A query that finds the top X (entered value) players in terms of points scored. Returns information about players and sorts them by points scored.

5. A query that finds the top X (entered value) players in terms of assists. Returns information about players and sorts them by assists.

6. A query that finds the top X(entered value) NBA head coaches in terms of number of wins. Returning information about coaches and sorting them by number of games won.

7. A query that finds players who receive a salary of X (entered value) or more. Returns player information and salary. Sorts by salary.

8. A query that finds players who have received an award (bonuses) worth X (entered value) or more. Returns the player information, the award type and the bonus amount. Sorts by the bonus amount.

9. A query that finds the best players under the age of X(entered value). Returns only players with >20 points per game or >8 assists per game(or both). Sorts by age.

10. A query that finds the best X (entered value) rebounders with the most total rebounds. Returns the player information, with the respective rebound information. Sorts by the total rebounds.

11. A query that returns the best players in terms of achieved points for a given X (entered value) draft pick and sorts by the achieved points. Returns the player information. Does not count those players that have not played any games.

7 Relational Database

Does this dataset require a relational database? Would other database systems be a better choice in modelling this data? Why or why not? Would the "interesting queries" you wrote be easier or harder to recreate if you were using an alternative database?

This dataset does not require a relational database, but a relational database is amongst the best options to model it. Firstly, in our dataset, the data is not a tightly connected graph and ends up sparsely connected. Due to this sparse connection, this dataset is more suited towards relational databases over something like graph databases. Secondly, the data model is quite rigid and consistent. The overarching rules of basketball and what stats are tracked are rarely changed, leading to tables that rarely have attribute additions or attribute deletions. Lastly, within the NBA database, there are specific starting points to query that require the key/value storage that a relational database offers. Usually when querying the NBA database, analysts will often want to look for certain players, certain teams, or certain stats rather than some of the connections that are between them. Therefore the NBA database can be modelled in many different ways such as a graph database, but a relational model would be one of the better database systems for it.

On average, the queries would be of similar difficulty to recreate using an alternative database. Examples of queries that would benefit the most from an alternative database such as a graph database, would be the college players and award winners since these queries focus on the relationships between entities. Since they focus on the relationships between entities, a graph database would be a natural choice to use to implement these queries. Other queries such as top players or stadium capacity, are quite simple to implement in a relational database and would be of similar difficulty in other database systems like a graph database.

8 References

"2021–22 NBA Season." Wikipedia, Wikimedia Foundation, 3 Feb. 2023, https://en.wikipedia.org/wiki/2021%E2%80%9322_NBA_season.

Büyüknacar, Muhammet Ali. \2021–22 NBA Season Active NBA Players." Kaggle, 29 Sept. 2021, https://www.kaggle.com/datasets/buyuknacar/202122-nba-season-active-nba-players?select=active_players_2.csv.

Cirtautas, Justinas. \NBA Players." Kaggle, 6 Aug. 2022, https://www.kaggle.com/datasets/justinas/nba-players-data.

\NBA History – 2021 Awards." ESPN, ESPN Internet Ventures, http://www.espn.com/nba/history/awards/_/year/2021.

Vinco, Vivo. \2021–2022 NBA Player Stats." Kaggle, 18 June 2022, https://www.kaggle.com/datasets/vivovinco/nba-player-stats?select=2021-2022%2BNBA%2BPlayer%2BStats%2B-%2BPlayoffs.csv.