Data Integration and Cloud Services

Course Project on

IPL Cricket Team Performance and Player Statistics

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Team No: 1

Ananya Deshpande	01FE21BCS059	511
Poorva G Khatawate	01FE21BCS302	246
Adwait S	01FE21BCS222	238
Amaan Khan	01FE21BCS212	212

Faculty In charges

Ms Neha Tarannum

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

HUBLI–580 031 (India).

Academic year 2023-24

Table of Content

1. Introduction

- Informatica PowerCenter efficiently extracts vast amounts of IPL data, including player statistics, match results, and historical data, providing a complete data set for analysis.
- Through data cleaning and validation processes, Informatica ensures high-quality, accurate, and consistent data, essential for reliable analysis and reporting.
- The ability to integrate data from multiple sources allows for a unified view of IPL information, enabling more comprehensive analysis and insights.
- Complex transformations, such as aggregations, calculations, and enrichments, are handled effortlessly, turning raw data into valuable insights.
- Informatica PowerCenter's scalability ensures that as the volume of IPL data grows, the ETL processes can handle increased loads without compromising performance.
- By loading transformed data into real-time dashboards and reporting tools, stakeholders can access up-to-date information for timely decision-making.
- Historical IPL 2023 data, when processed through advanced analytics, enables predictive modeling to forecast future performances and match outcomes.
- Data-driven insights derived from ETL processes support strategic decisions regarding player selections, game strategies, and overall team management, leading to optimized performance.

1.1 Preamble

The Indian Premier League (IPL) is a premier cricket tournament that generates vast amounts of data, encompassing player statistics, match results, and historical performance metrics. This extensive and diverse dataset is crucial for teams, analysts, and fans who seek to derive actionable insights and make informed decisions. However, the complexity and volume of this data pose significant challenges in terms of management, integration, and real-time analysis. Leveraging advanced ETL (Extract, Transform, Load) capabilities provided by Informatica PowerCenter can streamline the data processing workflow, ensuring data quality, consistency, and timely availability of insights. This approach not only enhances the analytical capabilities surrounding IPL cricket data but also optimizes performance and strategic decision-making for stakeholders involved in the league.

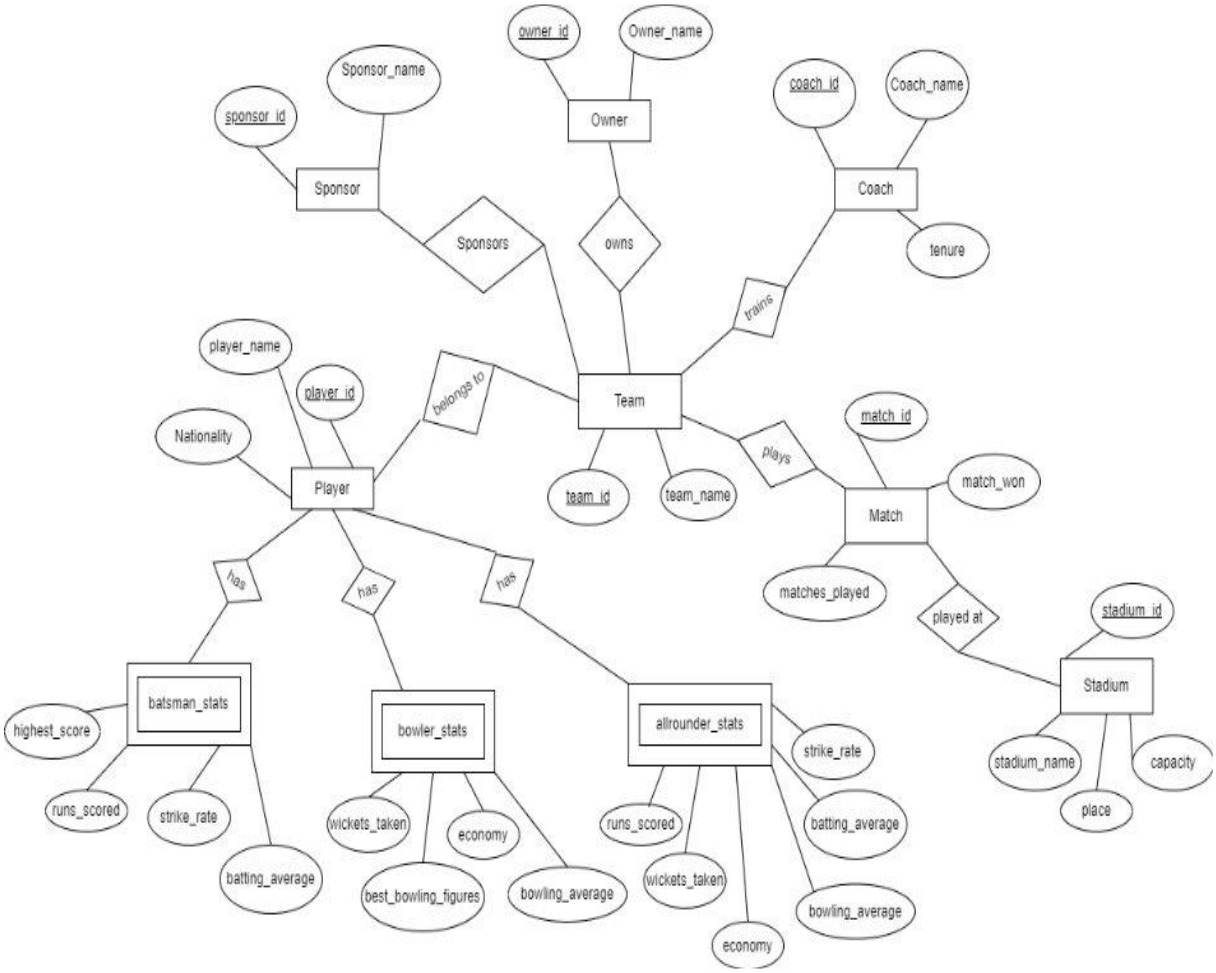
1.2 Problem Definition

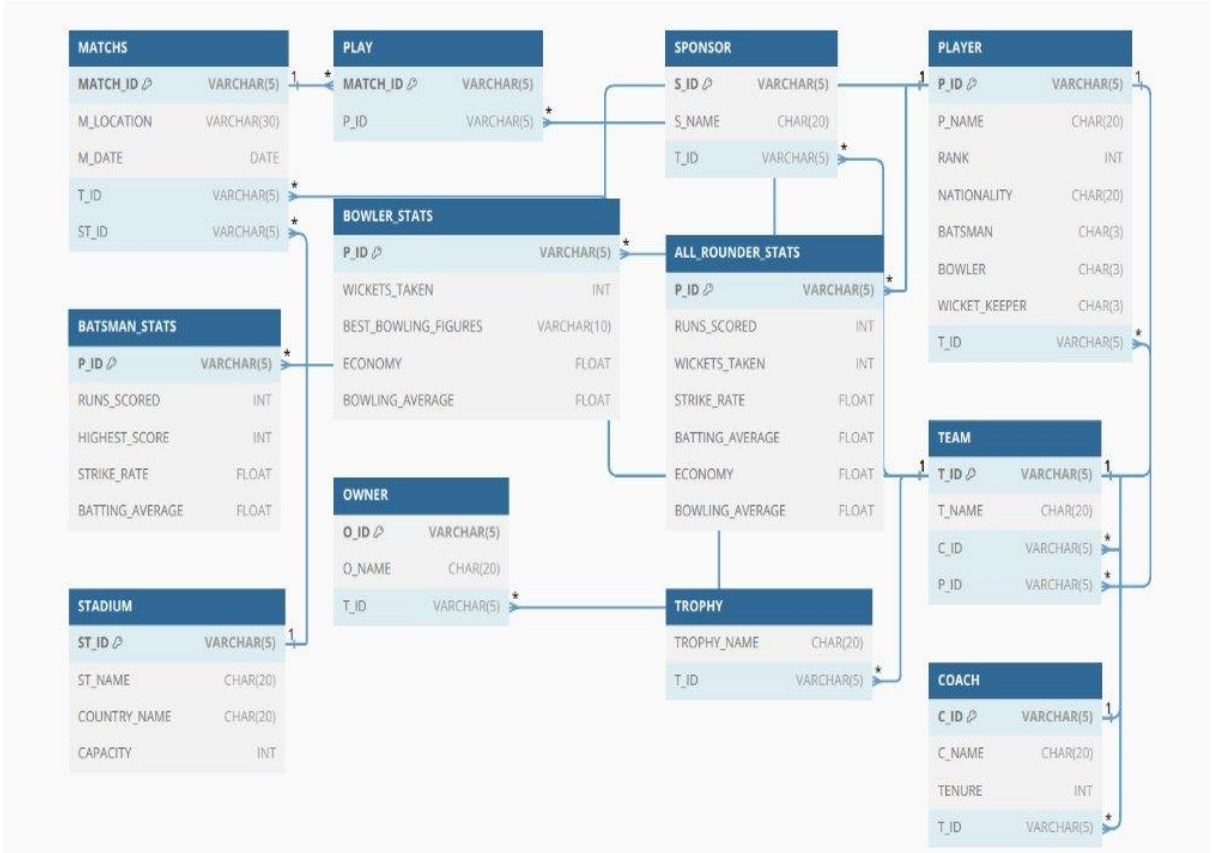
The IPL cricket database contains extensive, diverse data that is challenging to manage, integrate, and analyze effectively for actionable insights and real-time decision-making. Leveraging Informatica PowerCenter's ETL capabilities will streamline data processing, ensure data quality, and provide comprehensive analytics and reporting for the IPL.

1.3 Objectives

- Streamline IPL cricket data extraction from diverse sources using Informatica PowerCenter.
- Enable real-time reporting and analytics, providing up-to-date information for informed decision-making.
- Leveraging historical data trends and predictive analytics to optimize player selection and team strategies, enhancing overall IPL performance

2. ER diagram


owner_id
Owner_name
Owner
Sponsor_name
sponsor_id
Sponsor
Sponsors
owns
coach_id
Coach_name
Coach
tenure
trains
player_name
player_id
belongs to
Team
Nationality
Player
team_id
team_name
plays
match_id
match_won
Match
matches_played
played at
stadium_id
Stadium
stadium_name
capacity
place
has
has
has
batsman_stats
highest_score
runs_scored
strike_rate
batting_average
bowler_stats
wickets_taken
economy
best_bowling_figures
bowling_average
allrounder_stats
runs_scored
wickets_taken
economy
strike_rate
batting_average
bowling_average


MATCHS
MATCH_ID VARCHAR(5)
M_LOCATION VARCHAR(30)
M_DATE DATE
T_ID VARCHAR(5)
ST_ID VARCHAR(5)
PLAY
MATCH_ID VARCHAR(5)
P_ID VARCHAR(5)
SPONSOR
S_ID VARCHAR(5)
S_NAME CHAR(20)
T_ID VARCHAR(5)
PLAYER
P_ID VARCHAR(5)
P_NAME CHAR(20)
RANK INT
NATIONALITY CHAR(20)
BATSMAN CHAR(3)
BOWLER CHAR(3)
WICKET_KEEPER CHAR(3)
T_ID VARCHAR(5)
BOWLER_STATS
P_ID VARCHAR(5)
WICKETS_TAKEN INT
BEST_BOWLING_FIGURES VARCHAR(10)
ECONOMY FLOAT
BOWLING_AVERAGE FLOAT
ALL_ROUNDER_STATS
P_ID VARCHAR(5)
RUNS_SCORED INT
WICKETS_TAKEN INT
STRIKE_RATE FLOAT
BATTING_AVERAGE FLOAT
ECONOMY FLOAT
BOWLING_AVERAGE FLOAT
BATSMAN_STATS
P_ID VARCHAR(5)
RUNS_SCORED INT
HIGHEST_SCORE INT
STRIKE_RATE FLOAT
BATTING_AVERAGE FLOAT
TEAM
T_ID VARCHAR(5)
T_NAME CHAR(20)
C_ID VARCHAR(5)
P_ID VARCHAR(5)
OWNER
O_ID VARCHAR(5)
O_NAME CHAR(20)
T_ID VARCHAR(5)
STADIUM
ST_ID VARCHAR(5)
ST_NAME CHAR(20)
COUNTRY_NAME CHAR(20)
CAPACITY INT
TROPHY
TROPHY_NAME CHAR(20)
T_ID VARCHAR(5)
COACH
C_ID VARCHAR(5)
C_NAME CHAR(20)
TENURE INT
T_ID VARCHAR(5)

3. Data set description

Database Schema Description

The database consists of various tables designed to manage information related to ILP teams, sponsors, stadiums, owners, trophies, coaches, matches, players, and player statistics. Below is a detailed description of each table and its attributes.

1. TEAM Table

Table Name: TEAM

- TEAM_ID VARCHAR(5), Primary Key. Unique identifier for each team.

- TEAM_NAME: CHAR(20). The name of the team.

This table stores the basic information about teams.

2. SPONSOR Table

Table Name: SPONSOR

- SPONSOR_ID: VARCHAR(5), Primary Key. Unique identifier for each sponsor.

- SPONSOR_NAME: CHAR(20). The name of the sponsor.

- TEAM_ID: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates which team the sponsor is associated with.

This table contains information about the sponsors associated with each team.

3. STADIUM Table

Table Name: STADIUM

- STADIUM_ID: VARCHAR(5), Primary Key. Unique identifier for each stadium.

- STADIUM_NAME: CHAR(60). The name of the stadium.

- CAPACITY: INT. The seating capacity of the stadium.

This table holds information about different stadiums.

4. OWNER Table

Table Name: OWNER

- OWNER_ID: VARCHAR(5), Primary Key. Unique identifier for each owner.

- OWNER_NAME: CHAR(50). The name of the owner.

- TEAM_ID: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates

which team the owner is associated with.

This table stores information about the owners of the teams.

5. TROPHY Table

Table Name: TROPHY

- TEAM_ID: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates which team the trophy information pertains to.

- NO_OF_TROPHIES_WON: INT. The number of trophies won by the team.

This table contains the number of trophies won by each team.

6. COACH Table

Table Name: COACH

- COACH_ID: VARCHAR(5), Primary Key. Unique identifier for each coach.

- COACH_NAME: CHAR(30). The name of the coach.

- TEAM_ID: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates which team the coach is associated with.

This table holds information about the coaches of the teams.

7. MATCH Table

Table Name: MATCH

- MATCH_ID: VARCHAR(5), Primary Key. Unique identifier for each match.

- MATCH_DATE: DATE. The date when the match was held.

- MATCH_WON: VARCHAR(50). The result of the match (which team won).

- TEAM_ID1: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates the first team in the match.

- TEAM_ID2: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates the second team in the match.

-STADIUM_ID: VARCHAR(5). Foreign Key referencing STADIUM(STADIUM_ID). Indicates the stadium where the match was held.

This table contains details about each match played.

8. PLAYER Table

Table Name: PLAYER

- PLAYER_ID: VARCHAR(5), Primary Key. Unique identifier for each player.
- PLAYER_NAME: CHAR(20). The name of the player.
- NATIONALITY: CHAR(20). The nationality of the player.
- BATSMAN: CHAR(3). Indicates if the player is a batsman ('YES' or 'NO').
- BOWLER: CHAR(3). Indicates if the player is a bowler ('YES' or 'NO').
- ALL_ROUNDER: CHAR(3). Indicates if the player is an all-rounder ('YES' or 'NO').
- TEAM_ID: VARCHAR(5). Foreign Key referencing TEAM(TEAM_ID). Indicates which team the player belongs to.

This table holds information about the players.

9. BATSMAN_STATS Table

Table Name: BATSMAN_STATS

-PLAYER_ID: VARCHAR(5), Primary Key. Foreign Key referencing PLAYER(PLAYER_ID). Indicates which player the batting statistics belong to.

- RUNS_SCORED: INT. The total runs scored by the batsman.
- HIGHEST_SCORE: INT. The highest score achieved by the batsman in a match.
- STRIKE_RATE: FLOAT. The strike rate of the batsman.
- BATTING_AVERAGE**: FLOAT. The batting average of the batsman.

-MATCH_ID: VARCHAR(5). Foreign Key referencing MATCH(MATCH_ID). Indicates the match in which the stats were recorded.

This table contains statistical information about batsmen.

10. BOWLER_STATS Table

Table Name: BOWLER_STATS

-PLAYER_ID: VARCHAR(5), Primary Key. Foreign Key referencing PLAYER(PLAYER_ID). Indicates which player the bowling statistics belong to.

-WICKETS_TAKEN: INT. The total number of wickets taken by the bowler.

-BEST_BOWLING_FIGURES: VARCHAR(10). The best bowling figures of the bowler in a match.

-ECONOMY: FLOAT. The economy rate of the bowler.

-BOWLING_AVERAGE: FLOAT. The bowling average of the bowler.

-MATCH_ID: VARCHAR(5). Foreign Key referencing MATCH(MATCH_ID). Indicates the match in which the stats were recorded.

This table contains statistical information about bowlers.

11. ALL_ROUNDER_STATS Table

Table Name: ALL_ROUNDER_STATS

-PLAYER_ID: VARCHAR(5), Primary Key. Foreign Key referencing PLAYER(PLAYER_ID). Indicates which player the all-rounder statistics belong to.

- RUNS_SCORED: INT. The total runs scored by the all-rounder.

- WICKETS_TAKEN: INT. The total number of wickets taken by the all-rounder.

- STRIKE_RATE: FLOAT. The strike rate of the all-rounder.

- BATTING_AVERAGE: FLOAT. The batting average of the all-rounder.

- ECONOMY: FLOAT. The economy rate of the all-rounder.

- BOWLING_AVERAGE: FLOAT. The bowling average of the all-rounder.

-MATCH_ID: VARCHAR(5). Foreign Key referencing MATCH(MATCH_ID). Indicates the match in which the stats were recorded.

This table contains statistical information about all-rounders.

These tables are created in SQL developed, data is entered for each table and Informatica PowerCenter tool is used for creating mappings and applying various transformations for real time scenarios using the ILP cricket database of 2023.

4. Transformations

Various transformations of Informatica PowerCenter tool are used for the following real time scenarios.

Scenario 1:

Players count in each team according to nationality:

Utilizing the Players and Team data to find the count of players that belong to a particular nationality in each team.

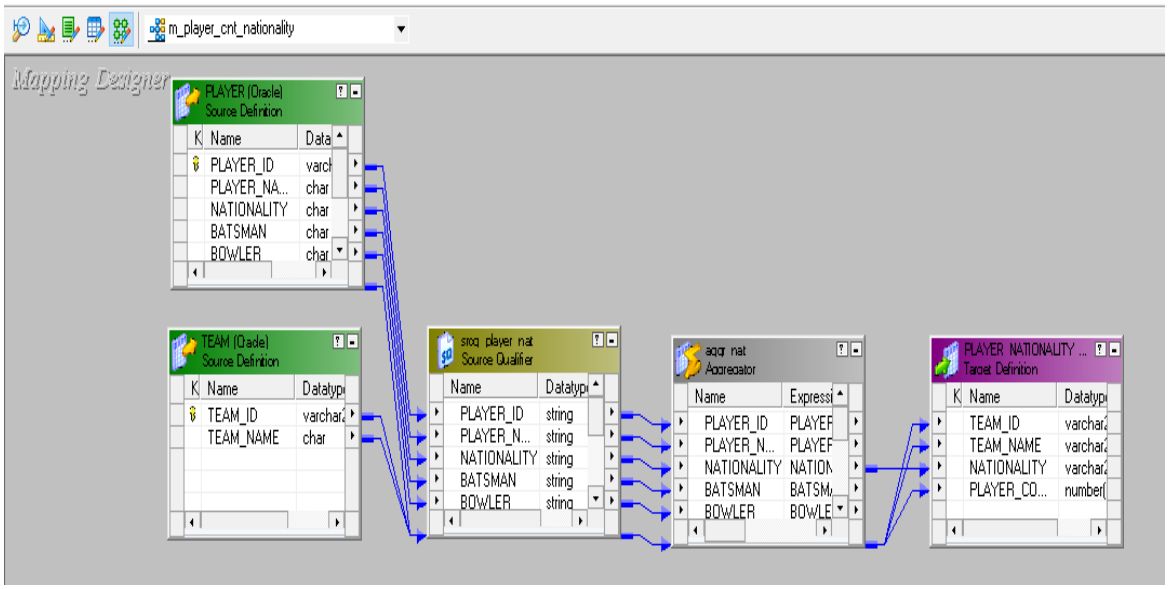
Transformations used:

- Source qualifier: Extracts data from the PLAYER and TEAM tables.
- Aggregator: Groups the data by TEAM_ID, NATIONALITY, and counts the number of players for each combination.

Target table has:

- Team ID
- Team Name
- Nationality
- Count of Players

The below figure shows the mapping of scenario 1



Scenario 2:

Top 5 teams with highest number of sponsors:

Utilizing the data from Sponsor and Team tables to find the top 5 teams having a higher number of sponsors.

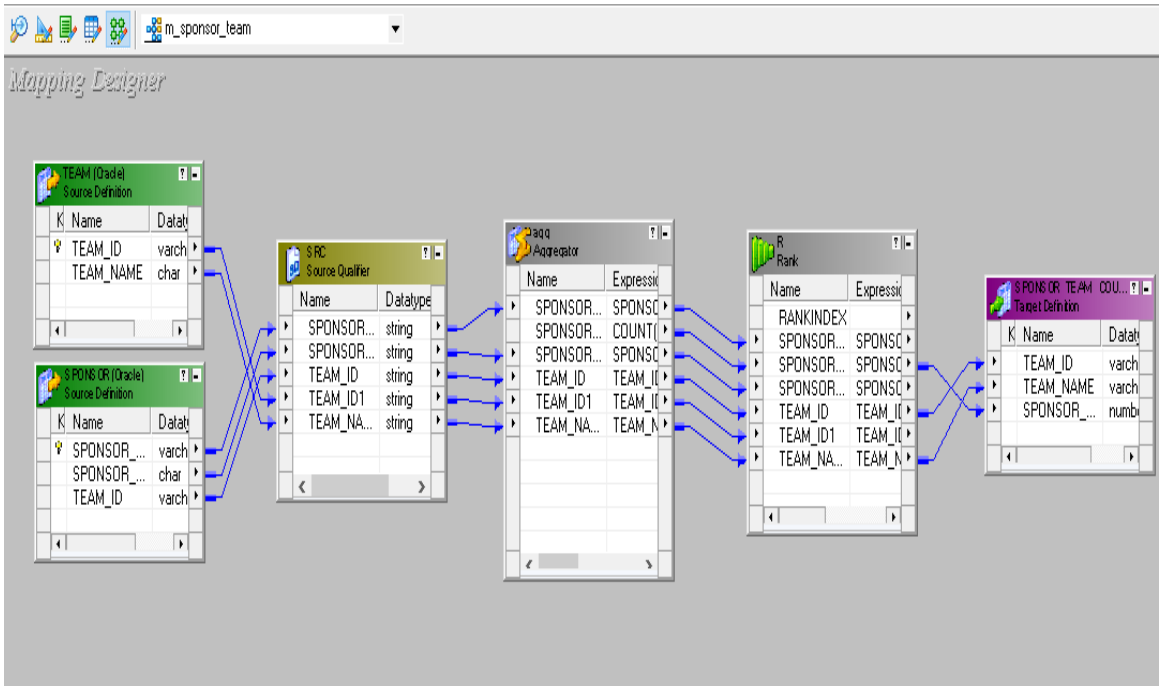
Transformations used:

- Source qualifier: Extracts data from the SPONSOR and TEAM tables.
- Aggregator: Groups the data by TEAM_ID and counts the number of sponsors for each team.
- Rank: Orders the teams by sponsor count in descending order and selects the top 5 teams.

Target table has:

- Team ID
- Team Name
- Sponsors count

The below figure shows the mapping of scenario 2



Scenario 3:

Top 5 Bowlers:

Utilizing the data from Players and Bowler_stats tables to find the top 5 bowlers.

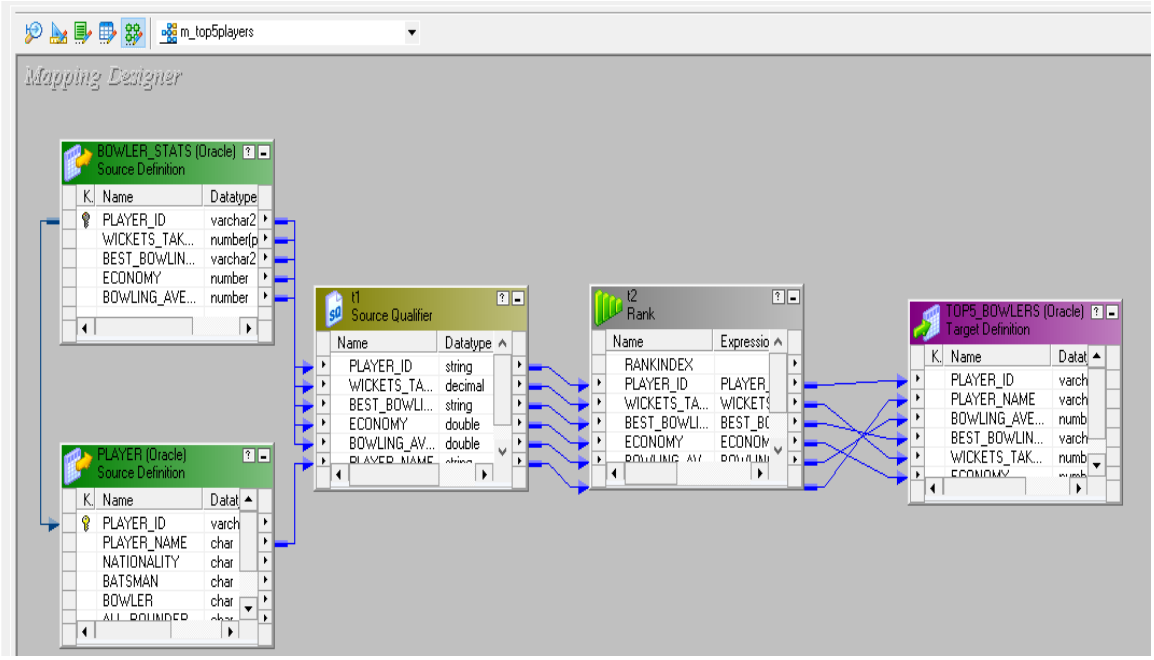
Transformations used:

- Source qualifier: Extracts data from the PLAYER and BOWLER_STATS tables.
- Rank: Ranks the bowlers based on specific criteria such as wickets taken or bowling average to determine the top 5 bowlers.

Target table has:

- Player ID
- Player Name
- Bowling average
- Best bowling figure
- Wickets taken
- Economy

The below figure shows the mapping of scenario 3



Scenario 4:

Group stadium based on number of matches played:

Utilizing the data from Match and Stadium tables to create two groups where Group A denotes more than 7 matches played at a particular stadium and Group B denotes less than 7 matches played at a particular stadium.

Transformations used:

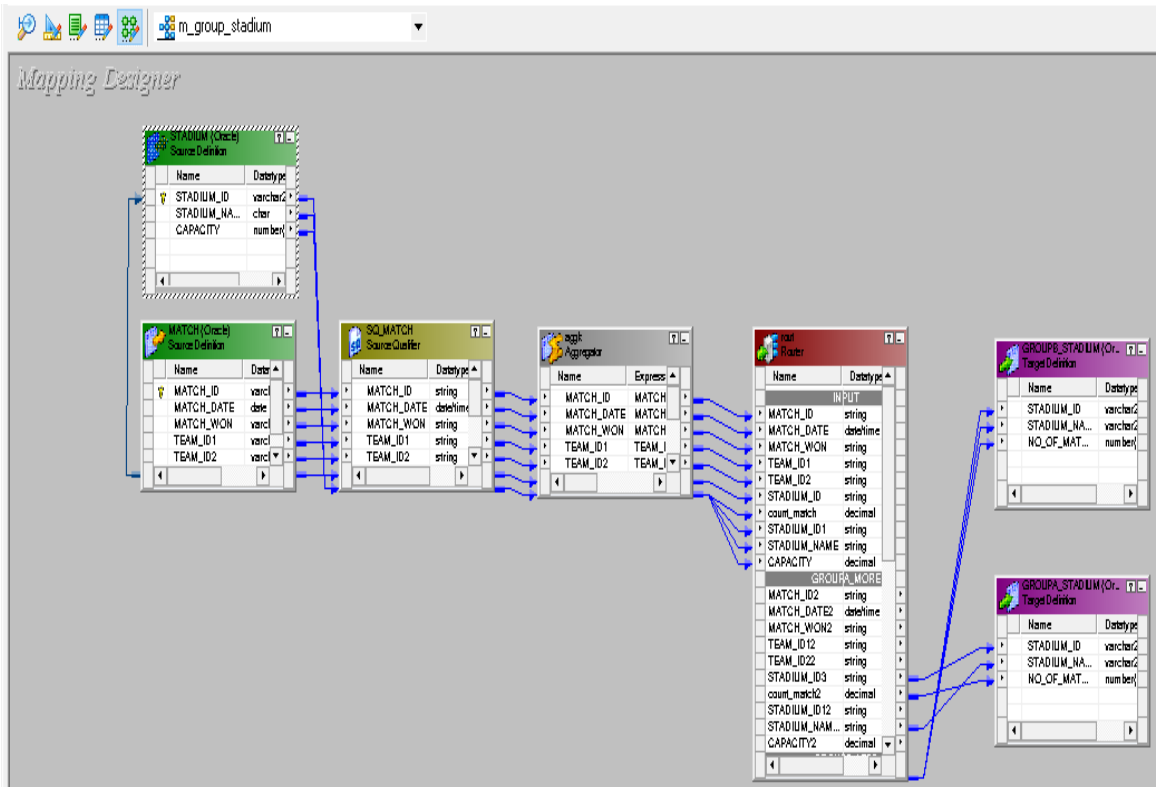
- Source qualifier: Extracts data from the MATCH and STADIUM tables.
- Aggregator: Groups the data by STADIUM_ID and counts the number of matches played at each stadium.
- Router: Routes the aggregated data into two groups based on the count of matches:

  Group A: Stadiums where the count of matches is greater than 7.
  Group B: Stadiums where the count of matches is 7 or fewer.

2 Target table have:

- Stadium ID
- Stadium Name
- Number of matches played

The below figure shows the mapping of scenario 4



Scenario 5:

Find the best team (Top 4, middle and bottom order):

Utilizing the data from Players, Batsmen_stats, Bowler_stats and All_rounder_stat tables to find a team with players having better performace and thus finding the top4, bottom4 and middle order of the team.

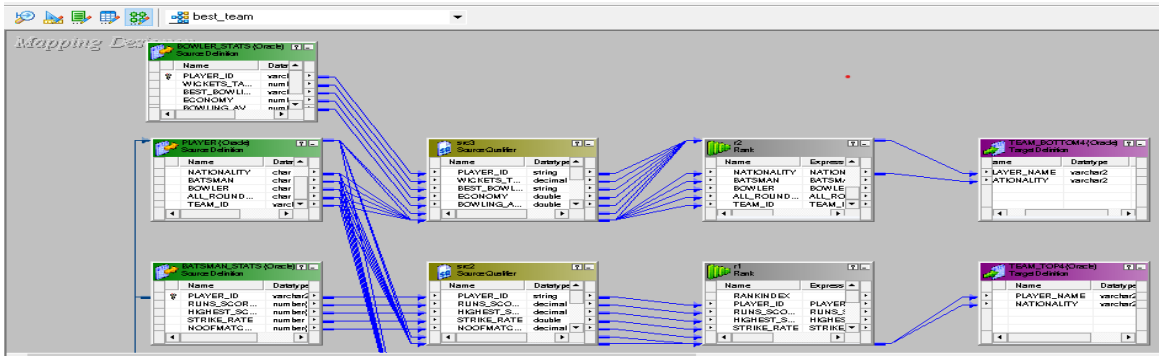
Transformations used:

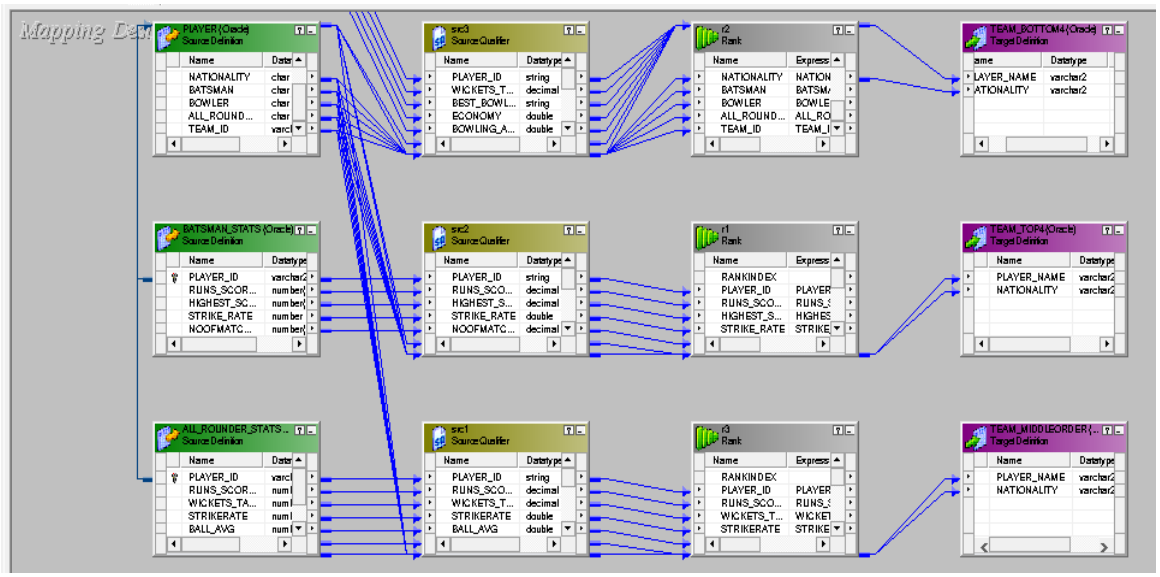
- Source qualifier: Extracts data from the PLAYER, BATSMAN_STATS, BOWLER_STATS, and ALL_ROUNDER_STATS tables to gather player performance metrics.
- Rank: Rank players based on their overall performance metrics (e.g., batting average, bowling average, all-rounder performance).

3 Target table have:

- Player name
- Nationality

The below figure shows the mapping of scenario 5





Scenario 6:

Allrounders Role Optimization**:**

Utilizing the data from Players and All_rounder_stat tables to find the statistics of player based on performance metrics.

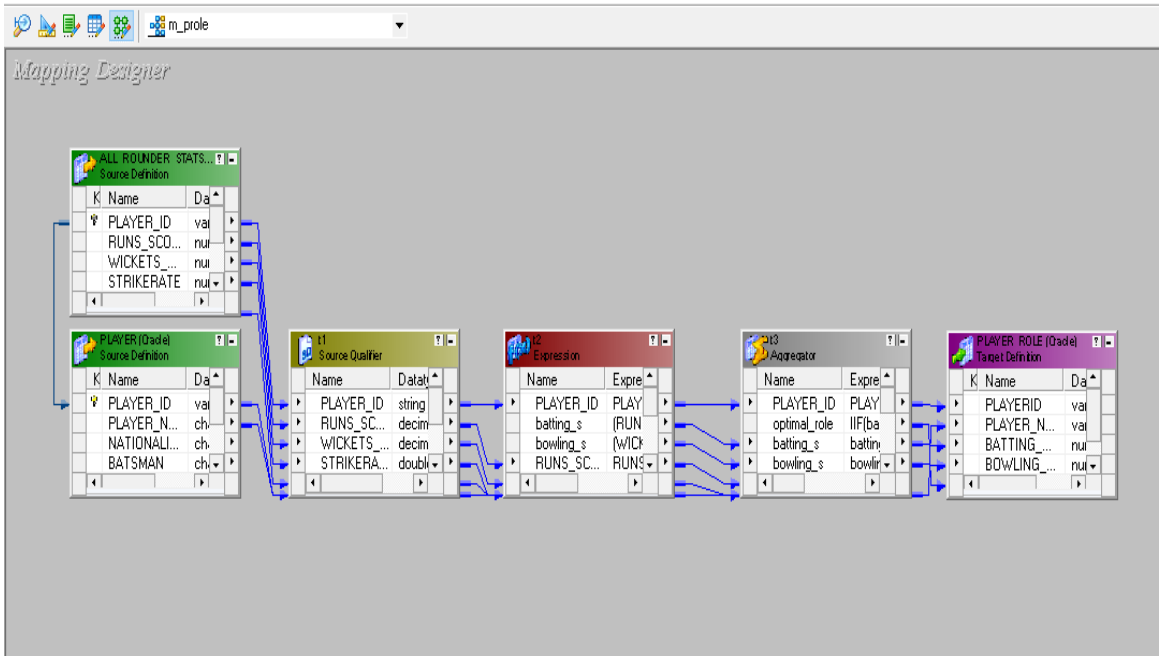
Transformations used:

- Source qualifier: Extracts data from the PLAYER and ALL_ROUNDER_STATS tables.
- Expression: Calculate suitable metrics to determine batting and bowling capabilities based on performance statistics like runs scored, wickets taken, strike rate, batting average, and economy.
- Aggregator: Aggregate player data to derive overall suitability scores for both batting and bowling.

Target table has:

- Player ID
- Player name
- Batting suitablity
- Bowling suitablity
- Optimal Role

The below figure shows the mapping of scenario 6



Scenario 7:

Average of Batsmen players:

Utilizing the date from Players, batsmen_stats and All_rounder_stat tables to find the average of batsmen and all-rounder players.

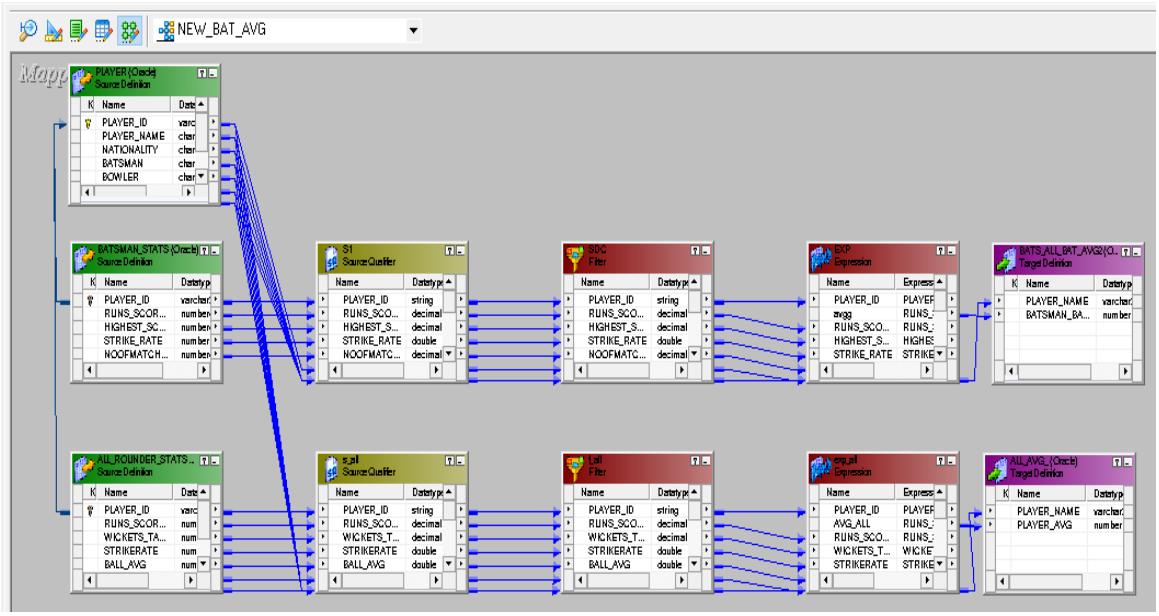
Transformations used:

- Source qualifier: Extracts data from the PLAYER and BATSMAN_STATS tables
- Filter: Filters the data to select only batsmen players (where BATSMAN = 'YES') from the PLAYER table.
- Expression: Calculates the average statistics (such as runs scored/no_of_matches_played) for each batsman player using the data from the BATSMAN_STATS and ALLROUNDER_STATS tables.

Target table has:

- Player name
- Average

The below figure shows the mapping of scenario 6



5. Conclusion

Implementing different scenarios by creating mappings using Informatica PowerCenter for IPL cricket data extraction from diverse sources, ensuring a cohesive dataset for analysis. This integration improved data processing efficiency and accuracy, laying a robust foundation for informed decision-making. Real-time reporting and analytics capabilities provide with up-to-date insights during the IPL season, facilitating agile responses to match dynamics and player performances.

Additionally, leveraging historical data trends and predictive analytics optimize player selection and team strategies. By analyzing past performance data, it enables data-driven decisions aimed at maximizing team efficiency and competitive advantage. It emphasizes how effectively Informatica PowerCenter was utilized to elevate IPL cricket performance by optimizing data management, providing real-time insights, and leveraging advanced analytics.