NHL Data Integration and Team DNA Profiling Pipeline

Building a scalable data pipeline to generate Team DNA profiles and quantify performance traits

Table of Content

1. Introduction

1.1 Overview

This project transforms raw National Hockey League (NHL) game data into Team DNA Profiles — data-driven representations of how teams play and evolve across seasons. By integrating multiple datasets, the project identifies distinct Team Archetypes — Attackers, Defenders, Strategists, and Balancers— which were defined by the team using key performance metrics such as efficiency, defense, and aggressiveness.
The outcome is an interactive visualization that highlights how team styles shift over time and how different archetypes influence success. Data was consolidated, cleaned, and modeled using Microsoft Fabric and the Medallion architecture to ensure a scalable, analytics-ready foundation, demonstrating skills in data pipeline design, ETL, and data modeling.

1.2 Problem statement

Although NHL team-level statistics exist for each game, analysts and fans face difficulty in quickly understanding how teams play and evolve over seasons, as the data is fragmented across multiple sources and lacks a framework for comparing playing styles. Users must manually interpret raw metrics, such as goals, penalties, and defensive stats, to identify team patterns or strategic tendencies.
This project addresses the problem by providing Team DNA Profiles and defined Team Archetypes (Attackers, Defenders, Strategists, and Balancers), based on curated performance metrics. The solution enables analysis of team playing styles over time, revealing trends and associations between archetypes and success in the league.

2. Data Sources

2.1 Dataset

NHL Game Dataset

- **Source link:** NHL Game Dataset
- **Description**: This repository contains 13 interrelated datasets covering NHL games across multiple seasons, including game, team, and player-level information such as scores, penalties, shots, and performance metrics.

2.2 Data Dictionary

2.2.1 game_team_stats.csv – team-level stats per game

Field Name	Data Type	Data Format	Description	Example
game_id	String	YYYYNNNN	Unique identifier for the NHL game. Format: Year + Game Number	2016020045
team_id	String	Numeric	NHL team identifier (internal team ID)	4
HoA	object	home/away	Indicates if the team played at home or away.	away
won	bool	TRUE / FALSE	Whether the team won the game	TRUE
settled_in	object	REG / OT / SO	How the game was settled: REG = Regulation, OT =	REG
head_coach	object	Text	Name of the team's head coach at the time of the game.	Dave Hakstol
goals	int64	Numeric	Number of goals scored by the team	4
shots	Int64	Numeric	Number of shots taken by the team	20
hits	Int64	Numeric	Number of hits recorded by the team.	11
pim	Int64	Numeric	Penalty minutes incurred by the team	3
powerPlayOpportunities	Int64	Numeric	Number of power play opportunities the team	5
powerPlayGoals	Int64	Numeric	Number of power play goals scored by the team.	2

faceOffWinPercentage	float64	Percentage (0.0 – 100.0)	Faceoff win percentage for the team in the game.	50.9
giveaways	Int64	Numeric	Number of giveaways committed by the team	12
takeaways	Int64	Numeric	Number of takeaways committed by the team	9
blocked	Int64	Numeric	Number of opponent shots blocked by the	11
startRinkSide	object	left/right	Side of the rink the team started on in the first period.	left

2.2.2 game_goalie_stats.csv – goalie performance per game

Field Name	Data Type	Data Format	Description	Example
game_id	String	Whole number	Unique identifier for the NHL game.	2016020045
player_id	String	Whole number	Unique identifier for the goalie player.	8473607
team_id	String	Whole number	Identifier for the team the goalie played for.	4
timeOnIce	Integer	Seconds	Total time on ice for the goalie during the game, in seconds.	1504
assists	Integer	Whole number	Number of assists credited to the goalie.	0
goals	Integer	Whole number	Number of goals scored by the goalie.	0
pim	Integer	Minutes	Penalty minutes incurred by the goalie.	0
shots	Integer	Whole number	Total shots against the goalie.	16

saves	Integer	Whole number	Total saves made by the goalie.	12
powerPlaySaves	Integer	Whole number	Saves made by the goalie during power play situations.	1
shortHandedSaves	Integer	Whole number	Saves made by the goalie during shorthanded (penalty kill) situations.	0
evenSaves	Integer	Whole number	Saves made by the goalie during even strength play.	11
shortHandedShotsAgainst	Integer	Whole number	Shots against the goalie during shorthanded situations.	0
evenShotsAgainst	Integer	Whole number	Shots against the goalie during even strength play.	13
powerPlayShotsAgainst	Integer	Whole number	Shots against the goalie during power play situations.	3
decision	String	Text	Game decision for the goalie: W (win), L (loss), or empty (no decision).	L
savePercentage	Float	Percentage	Overall save percentage (saves / shots * 100).	75.000000
powerPlaySavePercentage	Float	Percentage	Save percentage on power play shots (powerPlaySaves / powerPlayShotsAgainst * 100).	33.333333
evenStrengthSavePercentage	Float	Percentage	Save percentage during even strength play (evenSaves / evenShotsAgainst * 100).	84.615385

2.2.3 game.csv – general game information (date, teams, venue, scores)

Field Name	Data Type	Data format	Description	Example
game_id	String	Numeric	Unique identifier for each NHL game	2016020045
season	String	Numeric	Season year notation	20162017
type	String	Text	Game type: regular (R),Playoffs(P),Preseason(A)	R
date_time_GMT	datetime	timestamp	Date/time in GMT	2016-10-19T00:30:00Z
away_team_id	Integer	Whole number	4	Away team numeric code
home_team_id	Integer	Whole number	16	Home team numeric code
away_goals	Integer	Whole number	4	Away team goals scored
home_goals	Integer	Whole number	7	Home team goals scored
outcome	String	Text	Home win REG	Result with type (REG, OT)
home_rink_side_start	String	Text	right	Rink side home team
venue	String	Text	United Center	Game venue name
venue_link	String	Text	/api/v1/venues/null	Venue API placeholder

venue_time_zone_id	String	Text	America/Chicago	Venue timezone location
venue_time_zone_offset	Integer	Whole number	-5	Time offset from UTC
venue_time_zone_tz	String	Text	CDT	Timezone abbreviation

2.2.4 team_info.csv – team metadata

Field	Data Type	Data format	Description	Example
team_id	String	Numeric	Unique identifier for each NHL team	1
franchise_id	String	Numeric	Numeric code representing the franchise associated with the team	23
shortName	String	Text	Short city or location name for the team	New Jersey
teamName	String	Text	Full nickname or designation of the team	Devils
abbreviation	String	Text	Official 2-4 letter code used to abbreviate team names	NJD
link	String	Text	API endpoint suffix specific to the team	/api/v1/teams/1

2.2.5 game_goals.csv - metadata about hockey game goals

Field Name	Data Type	Data Format	Description	Example
play_id	String	Text	Unique identifier for each play, typically in the format YYYYMMDDGG_PPP.	2016020045_6

strength	String	Categorical	Indicates the game situation when the goal was scored.	Even
gameWinningGoal	Boolean	True, False, or Missing	Indicates whether the goal was the game-winning goal.	TRUE
emptyNet	Boolean	True, False, or Missing	Indicates whether the goal was scored against an empty net.	FALSE

2.2.6 game_plays_players.csv - metadata about hockey game plays

Field Name	Data	Data Format	Description	Example
play_id	String	Text	Unique identifier for each play, typically in the format YYYYMMDDGG_PPP.	2016020045_4
game_id	String	Numeric (YYYYMMDDGG format)	Unique identifier for each game, typically a 10-digit number.	2016020045
player_id	String	Numeric (7-8 digits)	Unique identifier for each player involved in the play.	8473604
playerType	String	Categorical	Role of the player in the specific play (e.g., Shooter, Goalie, Assist).	Winner

2.2.7 game_officials.csv - metadata about hockey game officials

Field Name	Data Type	Data Format	Description	Example
game_id	String	Numeric	Unique identifier for each game, typically a 10-digit number.	2016020045
official_name	String	Text	Full name of the official involved in the game.	Dan O'Rourke

official_type	String	Categorical	Role of the official in the game (Referee or Linesman).	Referee

2.2.8 penalties.csv - metadata about hockey game penalties

Field Name	Data Type	Data Format	Description	Example
play_id	String	Text	Unique identifier for each play, typically in the format YYYYMMDDGG_PPP.	2016020045_41
penaltySeverity	String	Categorical	Type of penalty assessed (e.g., Minor, Major, Misconduct).	Minor
penaltyMinutes	Integer	Numeric	Duration of the penalty in minutes.	2

2.2.9 game_plays.csv - play-by-play data for NHL games

Field Name	Data Type	Data Format	Description	Example
play_id	String	Text	Unique identifier for each play in the game.	2016020045_1
game_id	String	Numeric	Unique identifier for the game.	2016020045
team_id_for	Float	Numeric	Team ID performing the action (NaN for non-team events).	16
team_id_against	Float	Numeric	Opposing team ID (NaN for non-team events).	4
event	String	Categorical	Type of event (e.g., Goal, Shot, Faceoff).	Goal
secondaryType	String	Text	Subtype of the event (e.g., Wrist Shot, often missing).	Wrap-around
x	Float	Numeric	X-coordinate of the event location on the rink.	-88
y	Float	Numeric	Y-coordinate of the event location on the rink.	5

period	Integer	Numeric	Game period (1-3 for regulation, 4 for OT, 5 for shootout).	1
periodType	String	Categorical	Type of period (REGULAR, OVERTIME, SHOOTOUT).	REGULAR
periodTime	Integer	Numeric	Time elapsed in the period in seconds.	56
periodTimeRemaining	Float	Numeric	Time remaining in the period in seconds.	1144
dateTime	String	DateTime	Timestamp of the event.	10/19/2016 1:41
goals_away	Integer	Numeric	Current goals scored by the away team.	0
goals_home	Integer	Numeric	Current goals scored by the home team.	1
description	String	Text	Textual description of the play.	Patrick Kane (1) Wrap-around, assists: Artem Anisimov (2), Brent Seabrook (2)
st_x	Float	Numeric	Standardized or flipped X-coordinate.	88
st_y	Float	Numeric	Standardized or flipped Y-coordinate.	-5

2.2.10 game_scratches.csv - records of players scratched or not dressed for NHL games

Field Name	Data Type	Data Format	Description	Example
game_id	String	Numeric	Unique identifier for each game.	2016020045

team_id	String	Numeric	Identifier for the team the player belongs to.	16
player_id	String	Numeric	Unique identifier for the scratched player (not participating in the game).	8477845

2.2.11 **game_shifts.csv** - records of player shifts in NHL games

Field Name	Data Type	Data Format	Description	Exampl
game_id	String	Numeric	Unique identifier for each game.	2018020001
player_id	String	Numeric	Unique identifier for the player.	8466139
period	Integer	Numeric	The period of the game during which the shift occurred (e.g., 1 for first period).	1
shift_start	Integer	Numeric	The start time of the player's shift in seconds from the beginning of the period.	0
shift_end	Float	Numeric	The end time of the player's shift in seconds from the beginning of the period.	42

2.2.12 **game_skater_stats.csv** - per-game statistics for NHL skaters

Field Name	Data Type	Data Format	Description	Example
game_id	String	Numeric	Unique identifier for each game.	2016020045
player_id	String	Numeric	Unique identifier for the player.	8468513
team_id	String	Numeric	Identifier for the team the player belongs to.	4
timeOnIce	Integer	Numeric	Total time on ice in seconds.	955

assists	Integer	Numeric	Number of assists made by the player.	1
goals	Integer	Numeric	Number of goals scored by the player.	0
shots	Integer	Numeric	Number of shots taken by the player.	0
hits	Float	Numeric	Number of hits delivered by the player (missing in some records).	2
powerPlay Goals	Integer	Numeric	Number of goals scored during power plays.	0
powerPlay Assists	Integer	Numeric	Number of assists during power plays.	0
penaltyMin utes	Integer	Numeric	Total penalty minutes served by the player.	0
faceOffWin s	Integer	Numeric	Number of faceoffs won by the player.	0
faceoffTak en	Integer	Numeric	Total number of faceoffs taken by the player.	0
takeaways	Float	Numeric	Number of takeaways by the player (missing in some records).	1
giveaways	Float	Numeric	Number of giveaways by the player (missing in some records).	1
shortHand edGoals	Integer	Numeric	Number of goals scored while short-handed.	0
shortHand edAssists	Integer	Numeric	Number of assists while short-handed.	0
blocked	Float	Numeric	Number of shots blocked by the player (missing in some records).	1
plusMinus	Integer	Numeric	Plus/minus rating for the player.	1
evenTime OnIce	Integer	Numeric	Time on ice at even strength in seconds.	858

shortHandedTimeOnIce	Integer	Numeric	Time on ice while short-handed in seconds.	97
powerPlayTimeOnIce	Integer	Numeric	Time on ice during power plays in seconds.	0

2.2.13 **player_info.csv** - player metadata

Field Name	Data Type	Data Format	Description	Example
player_id	String	Numeric	Unique identifier for each player.	8466148
firstName	String	Text	Player's first name.	Marian
lastName	String	Text	Player's last name.	Hossa
nationality	String	Categorical	Player's nationality code (e.g., ISO country code).	SVK
birthCity	String	Text	City where the player was born.	Stará Lubovna
primaryPosition	String	Categorical	Player's primary playing position (e.g., RW for Right Wing, D for Defense).	RW
birthDate	String	DateTime	Player's date of birth in YYYY-MM-DD HH:MM:SS format.	1/12/1979 0:00
birthStateProvince	String	Text	State or province where the player was born (missing for some players).	ON
height	String	Text	Player's height in feet and inches format.	6' 1"
height_cm	Float	Numeric	Player's height in centimeters.	185.42
weight	Float	Numeric	Player's weight in pounds.	207
shootsCatches	String	Categorical	Player's shooting or catching hand (L for Left, R for Right, or NaN).	L

3. Methodology / Approach

3.1 Tech Stack / Tools

Component	Tool / Platform	Purpose
Data Platform	Microsoft Fabric (Lakehouse)	Unified workspace for storage, transformation, and analytics
Data Processing	PySpark (Fabric Notebook)	Used for Silver-layer transformations: cleaning and wrangling
Data Querying	SparkSQL (Fabric SQL Endpoint)	Used in Gold layer for aggregations, KPI computation, and modeling
Visualization	Power BI (Integrated in Fabric)	Builds interactive dashboards on top of Gold-layer datasets
Architecture	Medallion Architecture	Structured data flow (Bronze → Silver → Gold) to ensure scalability and data quality

3.2 Medallion Architecture

The project follows the Medallion Architecture within Microsoft Fabric's Lakehouse environment, implementing a structured ELTL (Extract, Load, Transform, Load) process to ensure scalability, modularity, and analytical readiness.

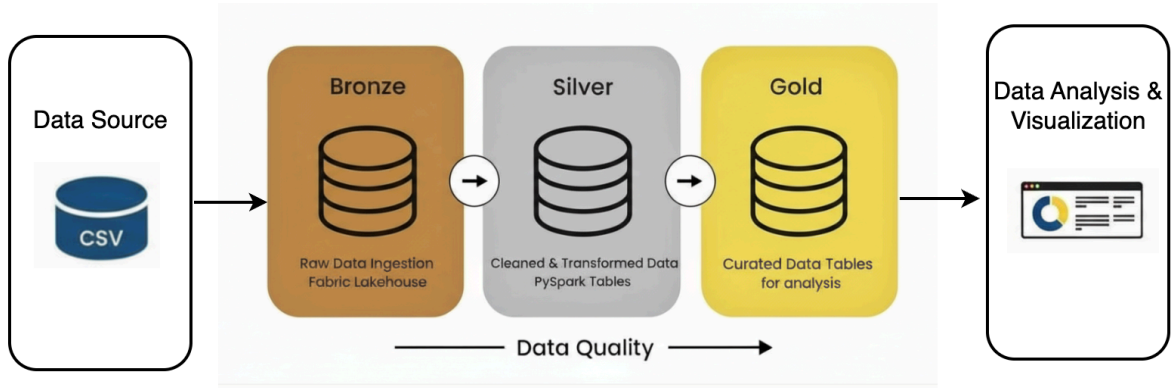
Layer	Description	Key Activities	Tools
Bronze (Raw Layer)	Stores the 13 original NHL datasets from Kaggle in their raw form.	Data ingestion, schema validation, and file organization.	Fabric Lakehouse
Silver (Cleaned & Modeled Layer)	Contains standardized and integrated data ready for analytics.	Data cleaning, type casting, renaming, joining, and filtering.	PySpark
Gold (Analytical Layer)	Aggregated, business-ready tables for insights generation.	Metric computation, archetype classification, and KPI derivation.	SparkSQL

Visualization Layer	Presents analytical insights interactively.	Create dashboards showing team archetypes and evolution.	Power BI

This layered approach supports reproducibility, scalability, and data lineage transparency, ensuring each stage of the pipeline serves a clear purpose.

3.3 Architecture Diagram

The following diagram provides a high-level overview of the project's architecture within Microsoft Fabric. It illustrates the logical flow of data through the Medallion Architecture (Bronze–Silver–Gold), from raw data ingestion to final visualization.



This architecture ensures:

- Centralized data storage and management within the Fabric Lakehouse
- Layered data refinement through Bronze (raw), Silver (cleaned), and Gold (analytical) stages
- Seamless connection to Power BI for reporting and interactive visualization

3.4 ELT Workflow

Step 1: Extract and Load (Bronze Layer)

- Retrieved 13 NHL datasets from Kaggle NHL Game Data.
- Uploaded CSV files to the Fabric Lakehouse.

Step 2: Transform (Silver Layer – Data Cleaning and Standardization)

- The Silver layer focuses on data standardization, cleaning, and integrity validation to ensure consistency across all 13 raw datasets before analytical modeling.
- Transformations were performed using PySpark notebooks within Microsoft Fabric, guided by a standardized Data Cleaning Checklist applied to all 13 raw datasets as shown below:

Category	Checks & Actions
1. Schema & Initial Inspection	- Reviewed data types (numeric, categorical, date, text) - Validated dataset size and schema alignment with dictionary
2. Data Type Casting	- Converted columns to correct data types (e.g., game_id as String, date_time_GMT as Timestamp) - Standardized naming conventions (e.g., Game Date → game_date)
3. Missing Values	- Identified null values using .isNull() - Dropped incomplete rows where necessary; retained strategic nulls for analysis
4. Duplicates	- Checked for exact duplicates using .dropDuplicates()- Ensured unique primary keys (e.g., game_id, player_id)
5. Outliers & Invalid Values	- Flagged or excluded logically invalid records (e.g., negative goals, impossible ages) - Validated game_id structure and team associations
6. Data Consistency	- Standardized categorical fields (e.g., “NY” vs “New York”)- Unified date/time formats (YYYY-MM-DD)
7. Data Integrity	- Verified foreign key relationships between tables (e.g., team_id, player_id) - Ensured referential integrity before joining
8. Business Logic Validation	- Confirmed game_id prefix aligns with season type - Verified goals and outcomes consistency
9. Output Validation	- Checked final row count against source - Previewed Silver tables in Fabric to confirm schema correctness

Step 3: Transform & Load (Gold Layer – Analytical Modeling using SparkSQL)

- The Gold layer focuses on aggregating, enriching, and modeling the cleaned Silver tables to produce business-ready analytical tables for reporting and visualization.

Key Workflow and Highlights:

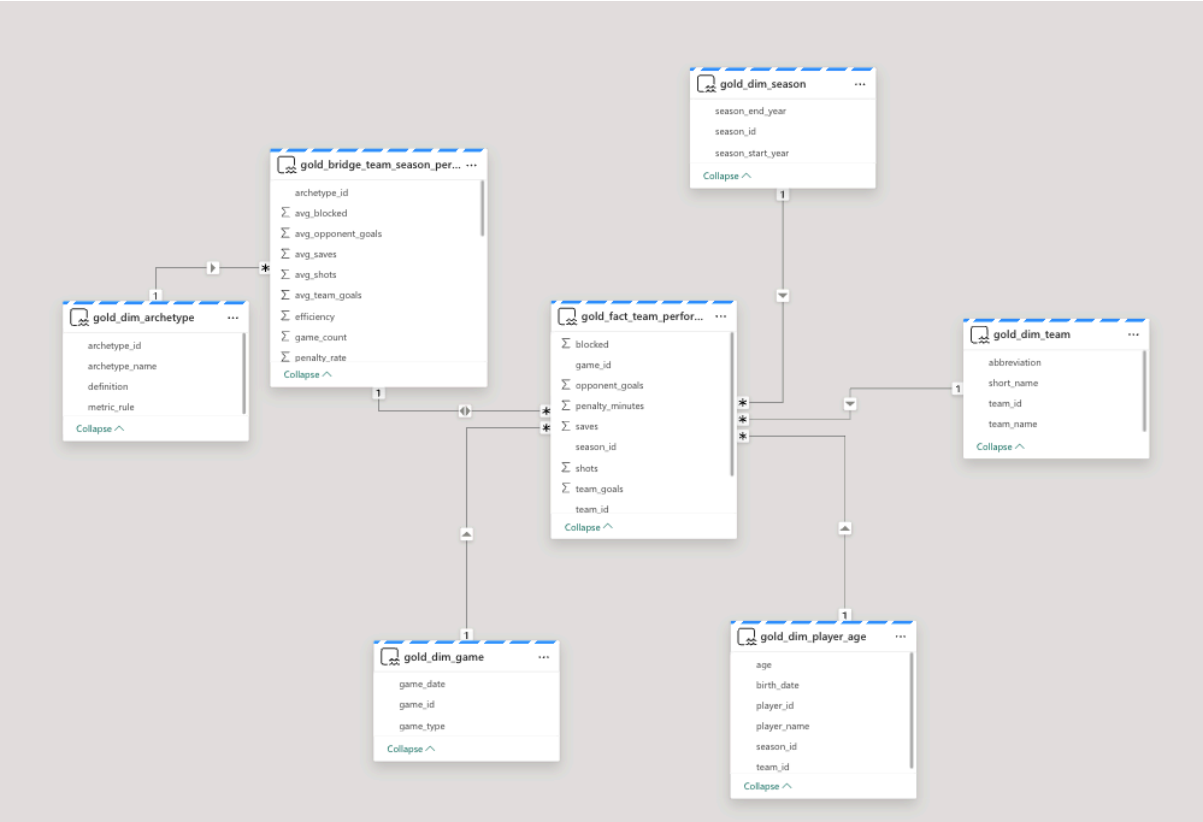
1. **Fact Table**
    a. Stores team- and game-level performance metrics such as goals, shots, penalties, saves, and blocked shots.
    b. Serves as the primary source for aggregation in downstream tables.
2. **Dimension Tables**
    a. Created for teams, players, seasons, and archetypes.
    b. Enable structured joins and support slicing and reporting by attributes like season, team, player, or archetype
3. **Bridge Table for Team-Season Performance**
    a. Aggregates team-season metrics (e.g., average goals, shots, penalty minutes, wins, efficiency).
    b. Composite key (team_season_id) ensures uniqueness.
    c. Precomputes **team archetypes** based on weighted scoring logic:
        i. **Attackers:** emphasize shots and penalty minutes
        ii. **Defenders:** emphasize saves, blocked shots, and limiting opponent goals
        iii. **Strategists:** emphasize efficiency and win percentage
        iv. **Balancers:** reward well-rounded teams
    d. Simplifies analysis and visualization in Power BI.

Step 4: Visualization (Power BI)

- A semantic model was first created in the Lakehouse using the Gold tables, which was then exposed as a Power BI dataset.
- Built interactive dashboards displaying:
    - Team archetypes distribution
    - Seasonal evolution of team archetype distribution
    - Win-rate for each archetype

- Performance metrics for each archetype
- Dashboards allow slicing by season and team, supporting data-driven storytelling.

3.5 Database Schema / Semantic Model



This follows a star schema optimized for analytical queries in Power BI. The fact table stores detailed game-level performance metrics, while dimension tables provide contextual information such as teams, players, and seasons. A bridge table is introduced to model team-season relationships and aggregate key performance indicators, allowing efficient retrieval of archetype insights over multiple seasons.

3.6 Table Structures

Table Name	Primary Key	Column Name	Description
Gold_Fact_Team_Performance	game_id	game_id	Unique identifier for each game
		season_id (FK)	Season identifier
		team_id (FK)	Identifier for each team
		team_season_id(FK)	Identifier for each team–season combination
		team_goals	Number of goals scored by team
		opponent_goals	Number of goals scored by opponent team
		won	1 = True, 0 = False
		shots	Total shots attempted
		penalty_minutes	Total penalty minutes
		blocked	Total blocked by goalies/ team/game
		saves	Total saves by goalies/ team/game
		efficiency	Goals / Shots
		penalty_rate	Avg penalties per game
		defensive_strength	Save% at team level
Gold_Dim_Game	game_id	game_id	Unique identifier for each game
		game_type	Type of the game
		game_date	Date of the game
Gold_Dim_Team	team_id	team_id	Unique identifier for each team
		team_name	Full name of the team
		shortname	Short name of the team
		abbreviation	Abbreviation of the team name
Gold_Dim_Season	season_id	season_id	Unique identifier for each season

		season_start_year	Starting year of the season
		season_end_year	Ending year of the season
Gold_Dim_Date	date_id	date_id	Unique identifier for each date
		year	Year of the date
		month	Month of the date
		day_of_month	Day of the month
		day_of_week	Day of the week
		season_id (FK)	Season identifier
Gold_Dim_Archetype	archetype_id	archetype_id	Unique identifier for each archetype
		archetype_name	Name of the archetype
		definition	Description of the archetype
		metric_rule	Rule to determine the archetype
Gold_Dim_PlayerAge	Gold_Fact_Team_Performance	player_id	Unique identifier for each player
		season_id (FK)	Season identifier
		team_id (FK)	Team identifier
		player_name	Full name of the player
		birth_date	Birth date of the player
		age	Age at the start of the season
Gold_Team_Season_Performance	team_season_id	team_season_id	Unique identifier for each team–season combination
		season_id (PK, FK)	Season identifier
		team_id (PK, FK)	Team identifier
		game_count	Total number of games played by the team
		avg_team_goals	Average goals scored by team per game

		avg_opponent_goals	Average goals conceded per game
		avg_shots	Average shots per game
		total_penalty_minutes	Total penalty minutes
		avg_saves	Average saves per
		avg_blocked	Average blocked per game
		win_count	Total number of wins
		win_percentage	Percentage of games won
		efficiency	Goals / Shots

3.7 Data validation

After transforming and aggregating data into Gold tables, the following validation checks were performed to ensure correctness and integrity:

- **Row Count / Record Consistency:** Verified Gold table row counts match expectations from Silver tables.
- **Aggregation Checks:** Cross-checked summed and averaged metrics (goals, shots, wins, penalties) against source tables.
- **Foreign Key / Referential Integrity:** Ensured all dimension keys exist in fact and bridge tables.
- **Archetype Assignment Validation:** Confirmed all teams have assigned archetypes for each season; distributions checked for reasonableness.
- **Metric Consistency (Heatmap Validation):** Aggregated key metrics (goals, shots, blocked, saves, penalties, efficiency) by archetype and visualized in a heatmap. This ensures that the Gold layer calculations reflect the intended performance profiles for each archetype.
- **Business Logic Validation:** Verified derived metrics (efficiency, win percentage) correctly reflect underlying data.
- **Spot-Checking:** Sampled records to ensure transformations and joins were applied correctly.

These checks confirm that the Gold tables are accurate, consistent, and ready for downstream visualization and analysis.

4. Challenges / Solution / Limitations

4.1 Challenge #1: Narrowing the Project Scope and Understanding Domain Context

Challenge:
At the start of the project, the team faced difficulty defining a focused problem statement due to limited familiarity with the NHL domain and the complexity of the datasets.
With 13 raw datasets covering diverse aspects such as game events, player attributes, and team statistics — each containing unfamiliar hockey-specific terms (e.g., “pim,” “powerPlayOpportunities,” “faceOffWinPercentage”) — it was challenging to identify which variables were most relevant for meaningful analysis.
Without proper scoping, the risk was building a technically sound but directionless pipeline with unclear business value.

Solution / Workaround:
The team conducted an initial data exploration phase, reviewing all datasets to map relationships between entities (e.g., team, player, season).
To reduce scope creep, we aligned on a central analytical focus — understanding *team playstyle archetypes* — which leveraged both statistical metrics and contextual interpretation of game data.
This narrowed the scope to datasets most relevant to team-level performance while maintaining a manageable pipeline and clear deliverables.

Limitations:

- Domain understanding was largely derived from statistical relationships rather than expert hockey knowledge.
- Some nuanced metrics (e.g., giveaways, takeaways) were omitted due to lack of clarity or data completeness.
- Future iterations could benefit from domain consultation to refine metric selection and improve interpretability.

4.2 Challenge #2: Creating the ‘Perfect Archetype Metric’

Challenge:
One major challenge was devising an algorithm to accurately assign each NHL team to an archetype (Attacker, Defender, Strategist, or Balancer).

Initially, the team used fixed metric thresholds (e.g., ≥ 80th percentile in shots/goals → Attacker). However, this rigid approach left a portion of teams unclassified (NA/NULL) because they didn't neatly fit any archetype. Adjusting thresholds or adding a fifth "Generalist" archetype improved coverage but proved unsustainable as datasets and team performances evolved across seasons.

Solution / Workaround:
We shifted to a Continuous Scoring Model that calculates a score for each archetype and assigns the highest-scoring one.
For example:

- *Aggressive_Score* = percentile_rank(shots) + percentile_rank(goals_for) + percentile_rank(penalty_minutes)
- *Defensive_Score* = percentile_rank(–goals_against) + percentile_rank(blocked) + percentile_rank(save%)
- *Efficient_Score* = percentile_rank(goals_for_per_shot) + percentile_rank(win%)
  This ensures every team receives an archetype classification while maintaining fairness across different statistical distributions.

Limitations:

- Results remain sensitive to metric weighting; small changes can shift archetype labels.
- The model may miss qualitative play-style nuances not captured by stats.
- Continuous scoring assumes stable metric relationships across seasons, which may not always hold.

4.3 Challenge #3: Handling Invalid or Missing Data Without Compromising Dataset Size

Challenge:
During preprocessing, several fields contained invalid or missing values.
Removing all rows with inconsistencies initially seemed safest but led to a 40**% data loss**, leaving too small a dataset for reliable analysis.
Further investigation showed that the missing fields were mostly **non-critical variables**, not required for archetype scoring.

Solution / Workaround:
The team decided to retain rows with invalid or missing values in non-essential columns, maintaining roughly 85% data retention.

Only key metrics affecting archetype computation (e.g., goals, shots, saves) were cleaned or imputed.
Less relevant fields were left unchanged to preserve overall dataset representativeness.

Limitations:

- Retaining imperfect data introduces minor noise in secondary metrics.
- Assumes missing values are randomly distributed, which might not always be true.
- Future work could include targeted imputation or data-quality scoring to refine results.

5. Finding / Analysis

All Power BI visuals were designed with interactive slicers — Team Name, Archetype Name, and Season Year — allowing users to dynamically filter and explore specific teams, archetypes, or time periods.

Objective 1: Overall Team Archetype Distribution (Donut Chart)

Approach:

- Aggregated team-season archetypes from the Gold table and visualized the distribution in Power BI using a donut chart.

Key Findings:

- All NHL teams were successfully classified into four archetypes.
- Balancers dominate with 44.66%, reflecting the prevalence of well-rounded strategies.
- Attackers account for 18.79%, Strategists 12.59%, and Defenders 23.97%, showing that specialized archetypes are less common but significant.

Visualization:

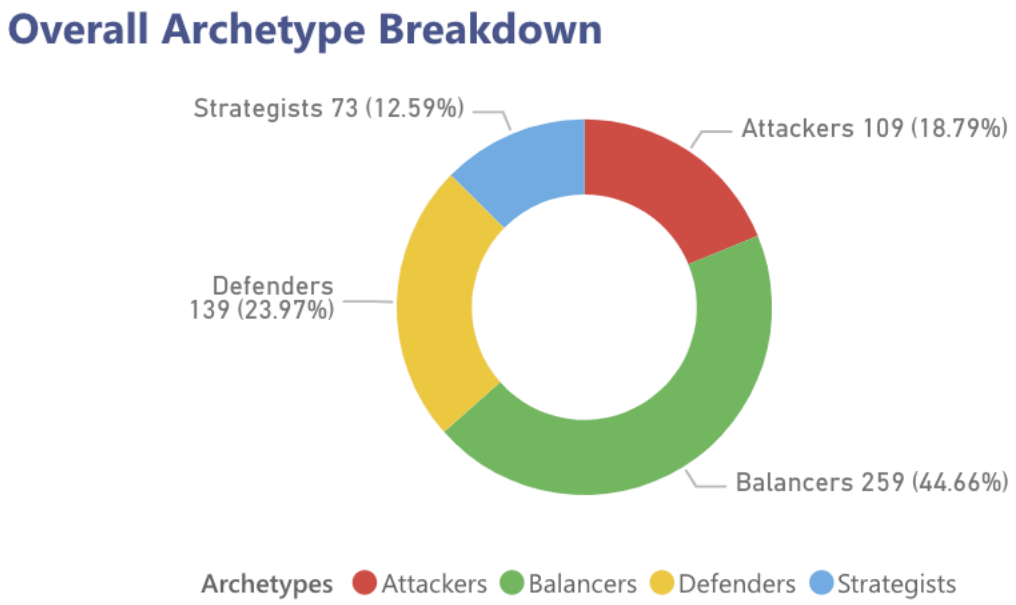


Figure 1: Donut chart showing overall NHL team archetype distribution across all seasons.

Objective 2: Team Archetype Trends Across Seasons (Stacked Column Chart)

Approach:

- Aggregated the count of teams per archetype for each season from the Gold layer.
- Visualized seasonal changes using a stacked column chart in Power BI to track shifts in team strategies from 2000 to 2020.

Key Findings:

- Balancers remained largest in seasons, showing the dominance of well-rounded strategies.
- Proactive playstyle (Strategists + Attackers) increased after 2015, indicating trend toward indicating a trend toward more offensive and initiative-driven team strategies.
- Overall, total team counts grew after 2017, reflecting a greater presence of competitive balance, team diversity & team participation across the league.

Visualization:

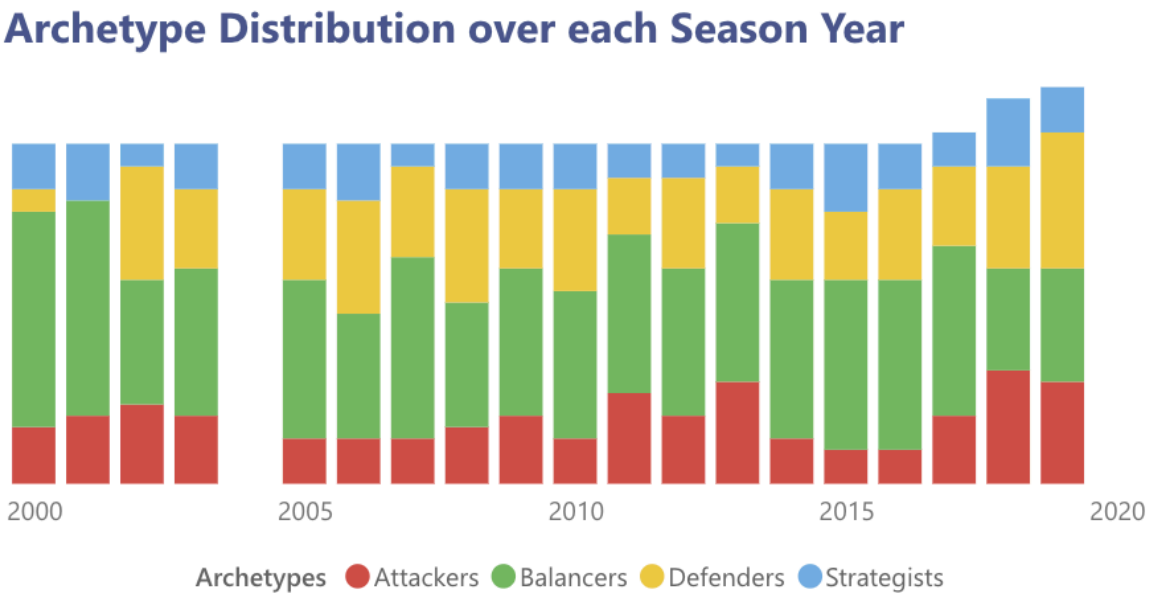


Figure 2: Stacked column chart showing team archetype distribution by season (2000–2020).

Objective 3: Archetype Win Rate and Game Count (Line and Clustered Column Chart)

Approach:
Used Gold-layer aggregated metrics to analyze the relationship between average games played and win rate per archetype from 2000 to 2020. Visualized using a line and clustered column chart, where columns represent average games played and the line represents average win percentage.

Key Findings:

- **Strategists** achieved the highest average win rate (~53.31%), validating the effectiveness of efficiency-based, low-penalty play.
- **Attackers** followed at ~51.10%, while **Balancers** and **Defenders** trailed at ~46.59% and ~46.23%, respectively.
- Average games played per archetype remained similar (~77–83 per season), suggesting that win-rate differences stem from play style rather than game volume.

Visualization:

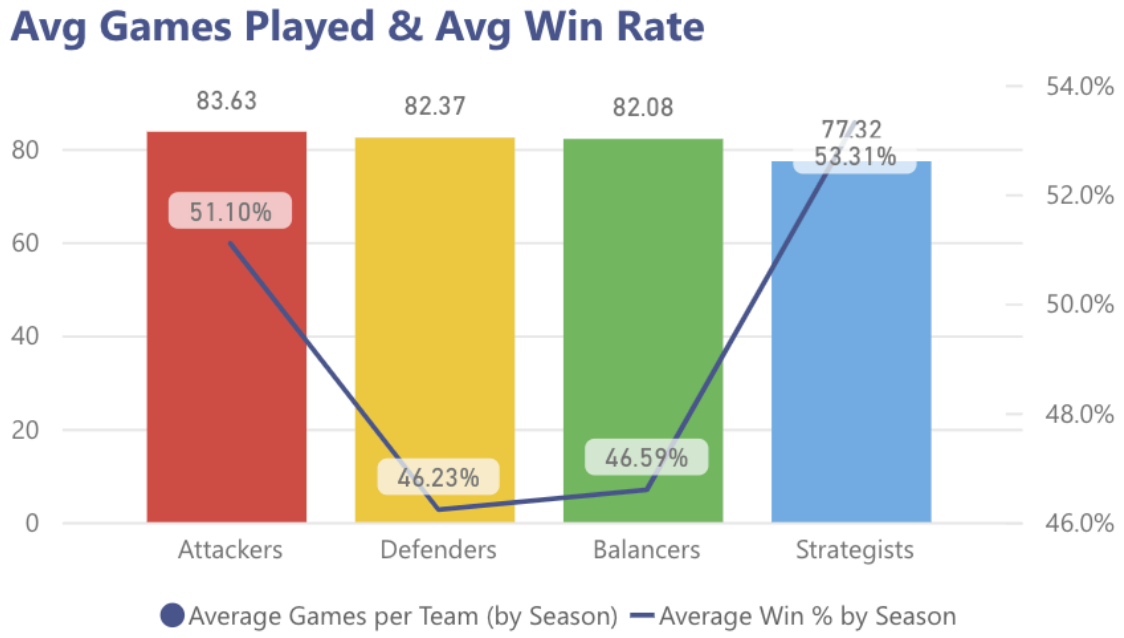


Figure 3: Line and clustered column chart comparing average win rate and games played per archetype (2000–2020).

6. Next Steps & Conclusion

6.1 Next Steps

1. Data Engineering Enhancements

- **Automated Pipeline Refreshes:** Implement scheduled ELTL processes to keep Power BI dashboards up-to-date with new season data.

2. Machine Learning / AI Application

- **Dynamic Metric Weighting:** Apply regression, clustering, or other ML approaches to dynamically adjust metric weights based on performance trends, improving adaptability and predictive power of archetype assignments.

3. Data Analysis Expansion

- **Player-Level Analysis:** Extend the archetype framework to include player-level metrics, enabling insights into individual contributions and informing team strategy and player evaluation.

- **Opponent-Adjusted Performance:** Refine team-level insights by accounting for opponent context, providing more nuanced and actionable analysis of archetype effectiveness.

6.2 Conclusion

This project successfully built an end-to-end NHL analytics pipeline that transforms raw game data into interpretable and actionable team “DNA profiles.”
The continuous-scoring algorithm ensures every team receives an archetype classification that is scalable, data-driven, and adaptable across seasons.
Interactive Power BI dashboards provide insights into archetype distribution, seasonal trends, performance efficiency, and team-level tendencies, enabling evidence-based decision-making for analysts, coaches, and management.
By integrating both team-level and season-level analyses, the project demonstrates how structured data modeling and visualization can uncover hidden strategic identities and guide future performance evaluation in professional hockey.