

Progetto finale Basi di Dati

Gestione del campionato di calcio di Prima Categoria Veneta

GIULIO FANTUZZI [EC2100728]

1 Descrizione della realtà di riferimento

Si vuole realizzare una base di dati per gestire una stagione di campionato di calcio di Prima Categoria Veneta. In generale, si dovranno registrare le partite disputate nei diversi gironi, con i relativi risultati, tenendo traccia di varie informazioni riguardanti squadre e calciatori.

1.1 Requisiti del database

- Si considera una singola stagione sportiva, la quale è articolata in 8 gironi (denominati con le lettere A-H), ciascuno dei quali è composto da 14 squadre;
- Ogni squadra è caratterizzata da un nome societario, dal paese di appartenenza, dallo stadio di proprietà (di cui si conosce la capienza e il costo per biglietto), dall'allenatore (di cui interessano alcuni dati anagrafici ed il possesso del patentino di base da allenatore) e dal girone di campionato a cui appartiene. Di ogni squadra sono noti i giocatori, ognuno dei quali caratterizzato da nome, numero di maglia, data e città di nascita e ruolo principale (NB: i ruoli dei giocatori però possono cambiare di partita in partita);
- Ogni partita è caratterizzata dal numero progressivo della giornata di campionato, dalla data in cui è stata disputata (espressa in anno-mese-giorno) e dalle 2 squadre che si sono affrontate nella gara (è importante registrare se una squadra ha giocato la partita in casa o in trasferta). Oltre al risultato finale della partita, si vuole tenere traccia del numero di spettatori presenti e del direttore di gara che ha arbitrato (esso è caratterizzato da nome, cognome, codice fiscale, data di nascita e paese di residenza);
- Per ogni giornata di campionato (di un fissato girone) si vogliono sapere i risultati di tutte le partite, i punti relativi ad ogni squadra, i giocatori di ogni squadra che hanno effettivamente giocato e quali di loro sono risultati marcatori, ammoniti e/o espulsi.

1.2 Operazioni di interesse

Alcune operazioni piuttosto standard, tra cui:

- Inserimento dei dati sulle partite ;
- Trasferimento di giocatori ;
- Visualizzazione delle partite di una determinata giornata e girone;

Altre operazioni più interessanti (le riprenderemo nel seguito), tra cui:

- Designazioni arbitrali per le partite di una giornata di campionato. Nello specifico, si vuole che un arbitro non possa dirigere una gara in cui gioca una squadra (se esiste) della suo paese di residenza;
- Visualizzare le statistiche di una Squadra: Goal Fatti, Goal Subiti, Partite Vinte, Partite Perse e Partite Pareggiate(120 volte a settimana);
- Visualizzare la classifica provvisoria di un dato girone (50 volte al giorno);
- Visualizzare la classifica marcatori di tutto il campionato (50 volte al giorno);
- Valutazione dell'affluenza del pubblico e del relativo incasso per le partite di ogni giornata di campionato(100 volte a settimana)

2 Glossario dei termini

| Termine | Descrizione | Sinonimi | Collegamenti |
|------------|---|---------------|--------------------------------|
| Partita | Incontro tra 2 squadre nel campionato | Gara,Incontro | Giornata, Squadra, Arbitro |
| Giornata | Insieme di più partite | Turno | Partita |
| Squadra | Squadra che gioca nel campionato | Società | Partita, Giocatore, Allenatore |
| Girone | Insieme di più squadre | Gruppo | Squadra |
| Stadio | Impianto sportivo in cui gioca la squadra | Campo | Squadra,Partita |
| Allenatore | Colui che allena una squadra | Tecnico | Squadra |
| Giocatore | Atleta di una squadra | Calciatore | Squadra,Partita |
| Arbitro | Colui che arbitra la partita | Direttore | Partita |

3 Suddivisione in frasi omogenee

NOTA: in questa sezione alcune frasi saranno semplificate rispetto la descrizione iniziale!

- Frase relative al girone

”8 gironi (denominati con le lettere A-H), ciascuno dei quali è composto da 14 squadre”;

- Frase relative alla partita

”Ogni partita è caratterizzata dal numero della giornata, dalla data in cui è stata disputata e dalle 2 squadre che si sono affrontate (distinguendo se in casa o in trasferta). Oltre al risultato finale, si vuole tenere traccia del numero di spettatori e del direttore di gara”;

- Frase relative alla giornata

”Per ogni giornata (di un fissato girone) si vogliono sapere i risultati delle partite, i punti relativi ad ogni squadra, i giocatori che hanno giocato e quali di loro sono stati marcatori, ammoniti e/o espulsi”;

- Frase relative alla squadra

”Ogni squadra è caratterizzata da un nome societario, paese di appartenenza, stadio di proprietà, allenatore e giocatori”;

- Frase relative allo stadio

”dello stadio si conosce la capienza e il costo per biglietto”;

- Frase relative all'arbitro

”il direttore di gara è caratterizzato da nome, cognome, codice fiscale, data di nascita e paese di residenza...non può arbitrare una gara in cui gioca una squadra (se esiste) della suo paese di residenza”;

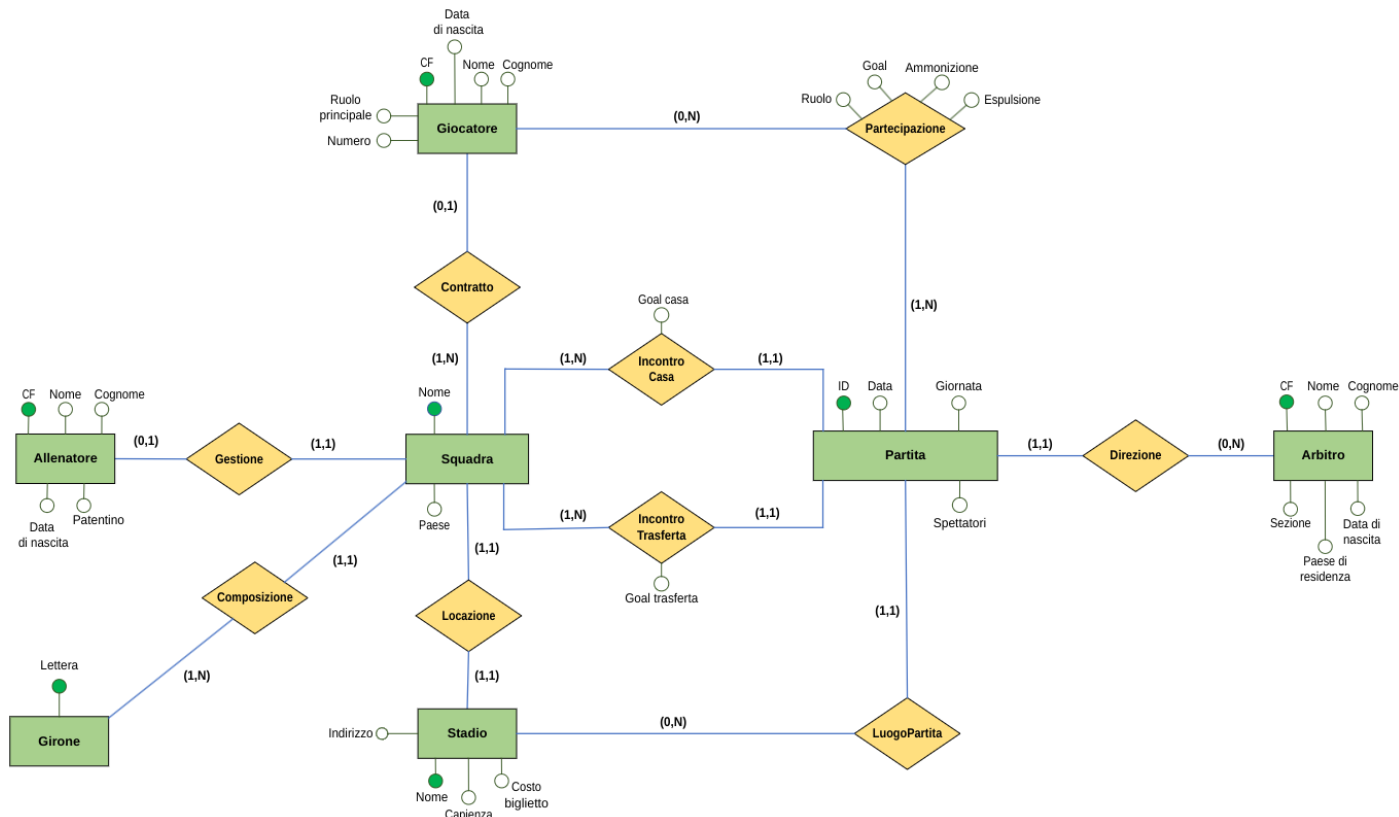
- Frase relative all'allenatore

”dell'allenatore interessano i dati anagrafici e il possesso o meno del patentino di base”;

- Frase relative ai giocatori

”ognuno è caratterizzato da nome, data e città di nascita e ruolo principale...i ruoli dei giocatori possono cambiare di partita in partita...si vogliono sapere i giocatori che hanno effettivamente giocato e quali di loro sono stati marcatori, ammoniti e/o espulsi”

4 Schema Entity-Relationship



Alcune osservazioni:

- Nel database si considerano solo gli stadi delle squadre coinvolte, quindi la relazione "Locazione" ha cardinalità (1,1). Per semplicità, si assume che i nomi degli stadi siano univoci e che squadre diverse non condividano lo stesso stadio;
- L'opzionalità in "Contratto" e "Gestione" deriva dalla possibilità che alcuni giocatori e allenatori siano svincolati. Assumendo inoltre che il numero di arbitri nel database sia strettamente maggiore di quello minimo per far fronte a tutte le partite, segue l'opzionalità in "Direzione". L'opzionalità in "LuogoPartita", infine, deriva dal fatto che in ogni giornata di campionato ci saranno stadi dove non vengono disputate gare (per essere precisi, quelli delle squadre in trasferta).
- Una partita di calcio prevede che si affrontino 2 squadre. Per rappresentare la relazione tra squadre e partite, ho valutato di inserire le 2 relazioni "Incontro Casa" e "Incontro Trasferta", così da poter distinguere sempre se una squadra gioca in casa o in trasferta;

5 Dizionario dei dati

5.1 Dizionario delle entità

| Entità | Descrizione | Attributi | Identificatore |
|------------|-----------------------------------|---|----------------|
| Partita | Incontro tra 2 squadre | ID,Data, Giornata, Spettatori | ID |
| Squadra | Squadra che gioca nel campionato | Nome, Paese | Nome |
| Arbitro | Colui che arbitra la partita | CF, Nome, Cognome, Data di nascita, Sezione, Paese di residenza | CF |
| Allenatore | Colui che allena la squadra | CF, Nome, Cognome, Data di nascita, Patentino | CF |
| Giocatore | È un calciatore | CF, Nome, Cognome, Data di nascita, Ruolo principale, Numero | CF |
| Stadio | Impianto sportivo in cui si gioca | Nome,Indirizzo, Capienza, Costo biglietto | Nome |
| Girone | Gruppo di squadre che competono | Lettera | Lettera |

5.2 Dizionario delle relazioni

| Relazione | Descrizione | Composizione | Attributi |
|--------------------|---|--------------------|---|
| Locazione | Associa le squadre al loro stadio di proprietà | Squadra,Stadio | |
| Contratto | Associa i giocatori alla loro squadra | Giocatore,Squadra | |
| Gestione | Associa gli allenatori alle squadre che dirigono | Allenatore,Squadra | |
| Partecipazione | Associa i giocatori alle partite in cui sono stati effettivamente schierati | Giocatore,Partita | Ruolo,Goal, Ammonizione, Espulsione |
| Direzione | Associa gli arbitri alle partite che dirigono | Partita,Arbitro | |
| LuogoPartita | Associa lo stadio alla partita che ospita | Stadio, Partita | |
| Incontro Casa | Associa una partita con la squadra che gioca in casa | Partita,Squadra | Goal Casa |
| Incontro Trasferta | Associa una partita con la squadra che gioca in trasferta | Partita,Squadra | Goal Trasferta |
| Composizione | Associa una squadra al girone di cui fa parte | Squadra,Girone | |

6 Vincoli non esprimibili graficamente

- Il numero di punti guadagnato da una squadra che vince/pareggia/perde una partita deve essere rispettivamente 3/1/0;
- Se una squadra gioca in casa una partita, allora è ospite nella partita successiva;
- Non possono affrontarsi squadre di gironi diversi;
- Un direttore di gara non può arbitrare una partita in cui gioca la squadra (se esiste) del suo stesso paese di residenza;
- Una squadra può essere gestita solo ad allenatori in possesso del patentino di base;
- Un giocatore non può essere espulso più di una volta a partita, come non può essere ammonito più di due volte nella stessa partita (in caso di 2 ammonizioni si registra direttamente l'espulsione). Inoltre, un giocatore espulso non può partecipare alla giornata successiva (supponiamo ci sia una squalifica standard per 1 solo turno);
- Ogni squadra può effettuare fino a un massimo di 5 sostituzioni a partita. Dato che in una partita si gioca 11 contro 11, ciò equivale a dire che in ogni partita non possono partecipare più di 16 giocatori per ogni squadra;
- Il ruolo di un giocatore può essere: Portiere, Difensore, Centrocampista, Attaccante;

7 Tavola dei volumi

Si vuole ora fornire una panoramica dettagliata sulle dimensioni e sulle quantità di dati presenti nel sistema. Alcune osservazioni preliminari:

- 14 squadre per girone $\implies (14 - 1) \cdot 2 = 26$ giornate e $14 \cdot 8 = 112$ squadre;
- Se ogni squadra ha mediamente 20 giocatori $\implies 2240$ giocatori. Siccome ammettiamo l'esistenza di giocatori svincolati, consideriamo un totale di 2500 giocatori;
- In ogni girone si giocano 7 partite per giornata, quindi in totale $7 \cdot 8 \cdot 26 = 1456$ partite;
- Ogni squadra deve avere l'allenatore e tutte le partite di una giornata devono avere un arbitro \implies allenatori ≥ 112 e arbitri ≥ 56 . Ammettendo allenatori senza squadra e più arbitri di quelli strettamente necessari, consideriamo 140 allenatori e 80 arbitri;
- Se in una partita sono sfruttate tutte le sostituzioni(5), ogni squadra schiera 16 giocatori;

| Concetto | Tipo | Volume |
|------------|------|--------|
| Squadra | E | 112 |
| Stadio | E | 112 |
| Allenatore | E | 140 |
| Giocatore | E | 2500 |
| Partita | E | 1456 |
| Arbitro | E | 80 |
| Girone | E | 8 |

| Concetto | Tipo | Volume |
|--------------------|------|---------------------|
| Locazione | R | 112 |
| Gestione | R | 112 |
| Contratto | R | 2240 |
| Partecipazione | R | 23296 (= 16 · 1456) |
| Direzione | R | 1456 |
| Incontro Casa | R | 1456 |
| Incontro Trasferta | R | 1456 |
| LuogoPartita | R | 1456 |
| Composizione | R | 112 |

8 Ristrutturazione del modello E-R

Il modello E/R proposto precedentemente mostra diverse carenze ed imprecisioni, per cui sarà necessario affrontare una fase di ristrutturazione ed ottimizzazione dello schema concettuale.

8.1 Eliminazione delle generalizzazioni

Lo schema realizzato non presenta alcuna generalizzazione, dunque si passa allo step successivo

8.2 Analisi delle ridondanze

In questa sezione si verificherà la presenza di eventuali ridondanze nello schema concettuale. Più precisamente, tramite un'analisi del carico di alcune operazioni di interesse (si veda la sezione 1.2), si valuterà se rimuovere, mantenere, o addirittura introdurre alcune ridondanze.

8.2.1 Operazione di visualizzazione statistiche squadra

Questa importante operazione si pone di calcolare alcune statistiche di una squadra, in particolare: goal fatti, goal subiti, partite vinte, partite perse e partite pareggiate.

In assenza di ridondanza, la tavola degli accessi risulterebbe:

| Concetto | Costrutto | Accessi | Tipo |
|--------------------|-----------|---------|------|
| Incontro Casa | R | C | L |
| Incontro Trasferta | R | T | L |

in cui C [T]: numero di partite giocate in casa [in trasferta] dalla squadra considerata

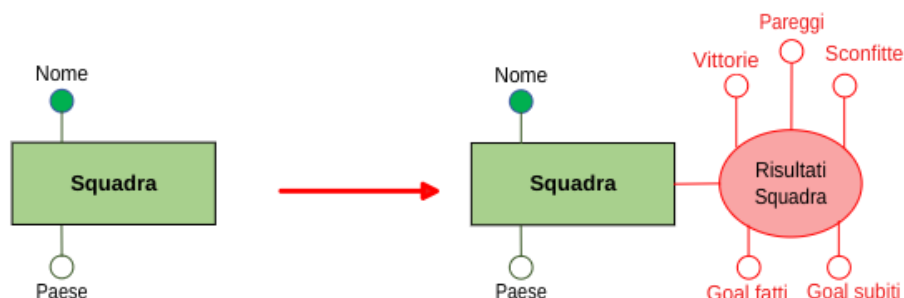
Fissiamo una generica squadra e valutiamo la peggiore situazione possibile, ossia l'ultima giornata di campionato. In questo caso si avrebbe $C=13$ e $T=13$, per cui se l'operazione venisse eseguita 120 volte a settimana, si dovrebbero effettuare $120 \cdot 13 = 1560$ accessi in lettura alla relazione "Incontro Casa" e 1560 accessi in lettura alla relazione "Incontro Trasferta", per un totale di **3120 accessi settimanali**.

Vediamo ora se, inserendo una ridondanza, sia possibile snellire il carico di lavoro. Per fare ciò si potrebbe introdurre un attributo composto (Risultati Squadra) alla squadra, contenente le statistiche d'interesse. Ad ogni inserimento del risultato di una Partita si dovrà quindi aggiornare anche l'attributo "Risultati Squadra" (per farlo basterà leggere il suo valore corrente ed effettuare una riscrittura tenendo conto della partita appena inserita). Con questo tipo di approccio, ogni volta che si intenderà calcolare le "statistiche" di una squadra (120 volte a settimana), non servirà più effettuare una ricerca su tutte le partite effettuate, ma basterà semplicemente andare a fare una lettura di Risultati Squadra. Considerato che l'inserimento del risultato di una partita avviene 1 volta a settimana per tutte le squadre (in totale sono 112), dalle osservazioni precedenti si ottiene la seguente tavola degli accessi:

| Concetto | Costrutto | Accessi | Tipo |
|----------|-----------|---------|------|
| Squadra | E | 112 | L |
| Squadra | E | 112 | S |
| Squadra | E | 120 | L |

Ricordando che, per convenzione, gli accessi in scrittura vengono conteggiati doppi, l'operazione richiederebbe un totale di $112 + 2 \cdot 112 + 120 = \mathbf{456 \text{ accessi settimanali}}$ anziché 3120.

Possiamo dunque concludere che l'introduzione di una ridondanza riduce notevolmente il numero di accessi al database, senza che il nuovo attributo ingombri troppa memoria in più.



8.2.2 Operazione di visualizzazione classifica provvisoria dei vari gironi

Risulta quasi immediato verificare come le conclusioni per questa operazione siano, in realtà, già implicate da quelle dedotte per l'operazione precedente. Infatti, per estendere a livello complessivo il risultato precedente (era valido a livello di singola squadra!) basterebbe:

1. In assenza di ridondanza: moltiplicare per il numero di squadre il totale degli accessi
 $\Rightarrow 3120 \cdot 112$ accessi totali
2. In presenza di ridondanza: moltiplicare per il numero di squadre solo gli accessi della terza riga della tabella
 $\Rightarrow 112 + 112 + 120 \cdot 112$ accessi totali

Più in generale, indicando con X : numero di accessi senza ridondanza e Y : numero di accessi con ridondanza (con $Y = y_1 + y_2 + y_3$), risulta sempre vero (essendo $y_1, y_2, y_3 > 0$) che:

$$X > Y = (y_1 + y_2 + y_3) \Rightarrow n \cdot X > (y_1 + y_2 + n \cdot y_3) \quad \forall n > 0$$

In sintesi, il calcolo della classifica diventa così un'operazione molto semplice da implementare, in quanto per ogni girone basterà selezionare le squadre ordinandole in base ai loro punti.

NOTA: i punti possono essere ricavati dal numero di vittorie e pareggi della squadra. Inoltre, gli accessi a Girone (necessari per raggruppare le squadre) non sono stati considerati in questa analisi in quanto sarebbero in ugual numero sia in assenza che in presenza di ridondanza.

8.2.3 Operazione di visualizzazione classifica marcatori

Come per la classifica delle squadre, si può pensare di semplificare il problema analizzando dapprima la questione a livello di singolo giocatore. Ricordando che nella relazione "Partecipazione" è presente l'attributo "Goal", si potrebbe pensare di sfruttarla per ricavare i goal segnati per un fissato giocatore. In assenza di ridondanza:

| Concetto | Costrutto | Accessi | Tipo |
|----------------|-----------|---------|------|
| Partecipazione | R | P | L |

in cui P : numero di partite a cui ha partecipato il giocatore.

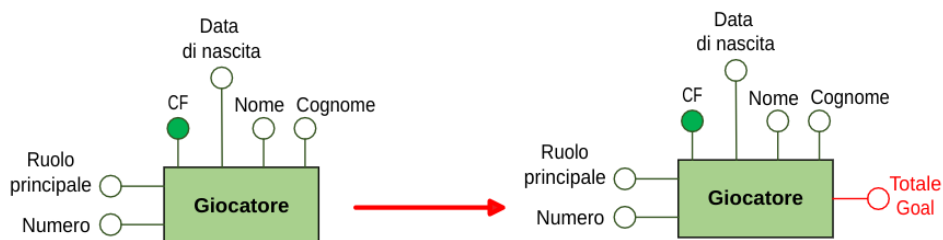
In questo contesto il caso "peggiore" si ottiene a fine campionato se il giocatore considerato ha partecipato a tutte le partite ($P=26$). Dato che l'operazione ha una frequenza di 50 volte al giorno (ossia 350 volte settimanali), si ha un totale di $350 \cdot 26 = \mathbf{9100}$ accessi settimanali.

Allo stesso tempo, si potrebbe pensare di introdurre un attributo "Totale goal" all'entità "Giocatore". Ogni volta che verrà inserita la Partecipazione di un giocatore a una Partita, se il giocatore in questione è risultato marcatore, ecco che si dovrà aggiornare anche il suo attributo "Totale goal" (per farlo basterà leggere il suo valore corrente ed effettuare una riscrittura tenendo conto della partecipazione appena inserita).

NOTA: se il giocatore non risulta marcatore, il campo "Totale goal" ovviamente non verrà riscritto. Nella valutazione degli accessi, però, considereremo un caso particolarmente estremo in cui ad ogni partita dev'essere effettuata una riscrittura per ogni giocatore che gioca (anche se è decisamente improbabile che ogni partita finisca almeno 16-16...). Con questo tipo di approccio non servirà più effettuare una ricerca su tutte le partite effettuate, ma basterà semplicemente fare una lettura di "Totale goal". In presenza di ridondanza si ottiene dunque:

| Concetto | Costrutto | Accessi | Tipo |
|-----------|-----------|-----------------------|------|
| Giocatore | E | $112 \cdot 20 = 2240$ | L |
| Giocatore | E | $112 \cdot 16 = 1792$ | S |
| Giocatore | E | 350 | L |

Ricordando che, per convenzione, gli accessi in scrittura vengono conteggiati doppi, l'operazione richiederebbe un totale di $2240 + 2 \cdot 1792 + 350 = \mathbf{6174}$ accessi settimanali anziché 9100. Siccome nella pratica verranno effettuati molti meno accessi (6174 si ricava da un caso estremo irrealistico) possiamo concludere che l'introduzione di una ridondanza riduce notevolmente il numero di accessi. Anche per questa operazione, l'estensione da livello di singolo giocatore a livello complessivo risulta quasi immediata (si veda il risultato di pag.9). Il calcolo della classifica marcatori diventa così un'operazione molto semplice da implementare, in quanto basterà selezionare i giocatori ordinandoli per Totale goal. Sarà opportuna la seguente modifica:



Si potrebbe fare un discorso identico per tenere traccia anche del totale di ammonizioni ed espulsioni di ogni giocatore. Tuttavia, non essendo questa una delle operazioni d'interesse per il progetto, si è valutato di non apportare ulteriori modifiche allo schema.

8.2.4 Operazione di visualizzazione incassi

In realtà, la relazione "LuogoPartita" non è strettamente necessaria per ricavare informazioni sugli incassi delle partite. Infatti, a partire dall'attributo "Spettatori" di "Partita", si potrebbero sfruttare le altre relazioni ed entità per risalire al costo dei biglietti, così da determinare gli incassi. Ricordando che ogni settimana vengono disputate 56 partite, si avrebbe:

| Concetto | Costrutto | Accessi | Tipo |
|--------------|-----------|---------|------|
| Partita | E | 56 | L |
| LuogoPartita | R | 56 | L |
| Stadio | E | 56 | L |

Table 1: Mantenendo la relazione

| Concetto | Costrutto | Accessi | Tipo |
|---------------|-----------|---------|------|
| Partita | E | 56 | L |
| Incontro Casa | R | 56 | L |
| Squadra | E | 56 | L |
| Locazione | R | 56 | L |
| Stadio | E | 56 | L |

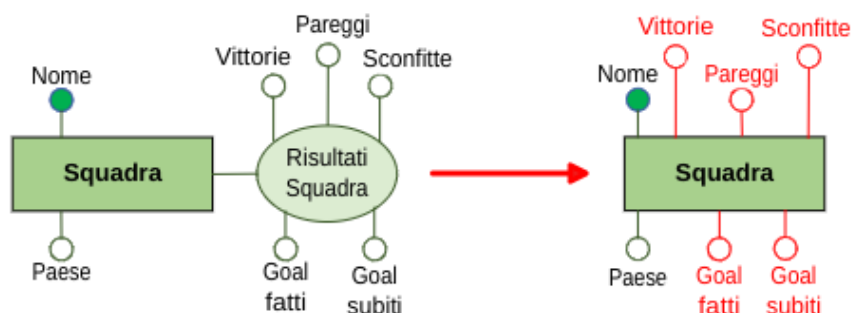
Table 2: Rimuovendo la relazione

Considerato che la frequenza dell'operazione è di 100 volte a settimana, rimuovendo la relazione sarebbero necessari $56 \cdot 5 \cdot 100 = 28000$ accessi settimanali, mentre mantenendola ne basterebbero 16800. Si conclude pertanto di mantenere la relazione "LuogoPartita".

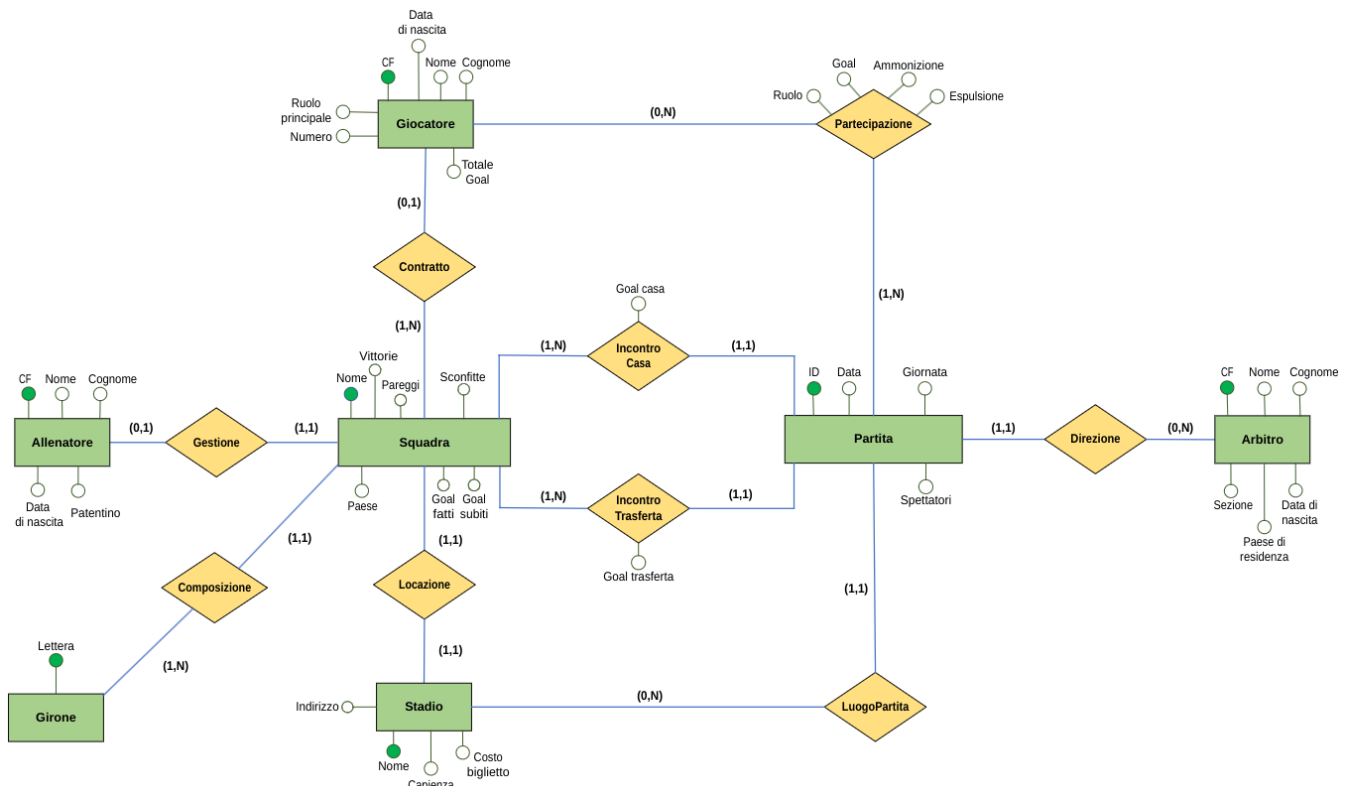
NB: similmente a quanto effettuato per "Giocatore", si poteva pensare di aggiungere anche un attributo "Totale Incassi" all'entità "Stadio", con l'idea di aggiornarlo progressivamente. Tuttavia, non risultando (per noi) un dato d'interesse frequente, si è valutato di non introdurlo.

8.3 Partizionamento degli attributi composti

Gli attributi composti del nostro modello sono coinvolti molto frequentemente nelle operazioni d'interesse. Si è dunque valutato di separare i sotto-attributi in campi separati, in modo da rendere più agevole l'accesso alle informazioni specifiche (utile anche per una futura 1NF).



9 Schema Entity-Relationship ristrutturato



10 Schema logico

- Girone(Lettera);
- Allenatore(CF, Nome, Cognome, Data di nascita, Patentino);
- Stadio(Nome, Indirizzo, Capienza, Costo biglietto);
- Arbitro(CF, Nome, Cognome, Data di nascita, Paese di residenza, Sezione);
- Squadra(Nome, Paese, Girone, Vittorie, Pareggi, Sconfitte, Goal Fatti, Goal Subiti, Stadio, Allenatore);
- Giocatore(CF, Nome, Cognome, Data di nascita, Numero, Ruolo Principale, Totale goal, Squadra);
- Partita(ID, Data, Giornata, Spettatori, Squadra casa, Squadra trasferta, Goal casa, Goal trasferta, Stadio, Arbitro);
- Partecipazione(Giocatore, Partita, Ruolo, Goal, Ammonizione, Espulsione)

11 SQL

Per non appesantire troppo il progetto, ho valutato di creare una repository su Github in cui ho inserito tutto il mio codice SQL. Nello specifico, è presente il codice per la creazione delle tabelle e l'inserimento di alcuni dati (il minimo necessario!), oltre che al codice per le varie query, stored procedure e trigger. La descrizione di tutti gli script, oltre che alcune note ed istruzioni importanti per l'uso, è stata documentata direttamente all'interno della repository.

Cliccare di seguito per accedere alla [Repository](#)

In alternativa, il link è: "https://github.com/giuliofantuzzi/Progetto-finale-Basi-di-Dati"