

Progetto Business Intelligence

Giorgio Simone 214575
Pasceri Francesco 204963

1. Descrizione dello scenario

Il nostro committente è Bet2020: uno dei maggiori bookmaker in Italia e nel Regno Unito.

Attualmente opera esclusivamente attraverso una piattaforma online che offre la possibilità, una volta registrati, di scommettere su diversi sport.

Fino ad oggi la società ha collezionato i dati su diversi database, senza la necessità di effettuare analisi approfondite sui propri trascorsi.

Gli utenti principali del sistema informativo oltre ai clienti registrati, sono gli amministratori del sistema che si occupano di aggiornare risultati live e altre statistiche. Una volta che i dati sono inseriti nei database, qui vengono conservati per diverso tempo (circa 8 anni) e vengono sfruttati dagli analisti per prospetti aziendali e statistici.

Gli aggiornamenti risultano essere quindi sia quotidiani che con cadenza trimestrale e annuale. Con il passare del tempo quindi, grandi moli di dati vengono conservati nei database che risultano sempre più appesantiti e, con le normali procedure operazionali, rendono più complicato l'accesso ai dati da parte degli addetti ai lavori, siano essi amministratori o manutentori delle basi.

Siamo stati contattati dalla società per la creazione di un data warehouse in grado di supportare procedure di analisi (data mining, analisi OLAP) e reportistica. Questo nasce dalla necessità di affrontare la crescente quantità di dati e la difficoltà nel portare avanti analisi dati necessarie per l'integrazione con i processi decisionali dell'azienda stessa.

La nostra soluzione prevede un'architettura a 3 livelli che inizia dall'acquisizione di dati operazionali e non, provenienti dalle sorgenti, per poi concludersi con la costruzione del data warehouse.

2. Analisi e Riconciliazione delle fonti dati

2.1 Ricognizione

La soluzione data warehouse proposta inizia proprio con l'acquisizione dei dati a partire dai database iniziali dell'azienda. I database a disposizione per la creazione del data warehouse sono due: il primo, chiamato Database_ES, contiene gli eventi e gli incontri dei vari sport; il secondo, Database_SS, è un altro database relativo alle scommesse sportive effettuate sulla piattaforma.

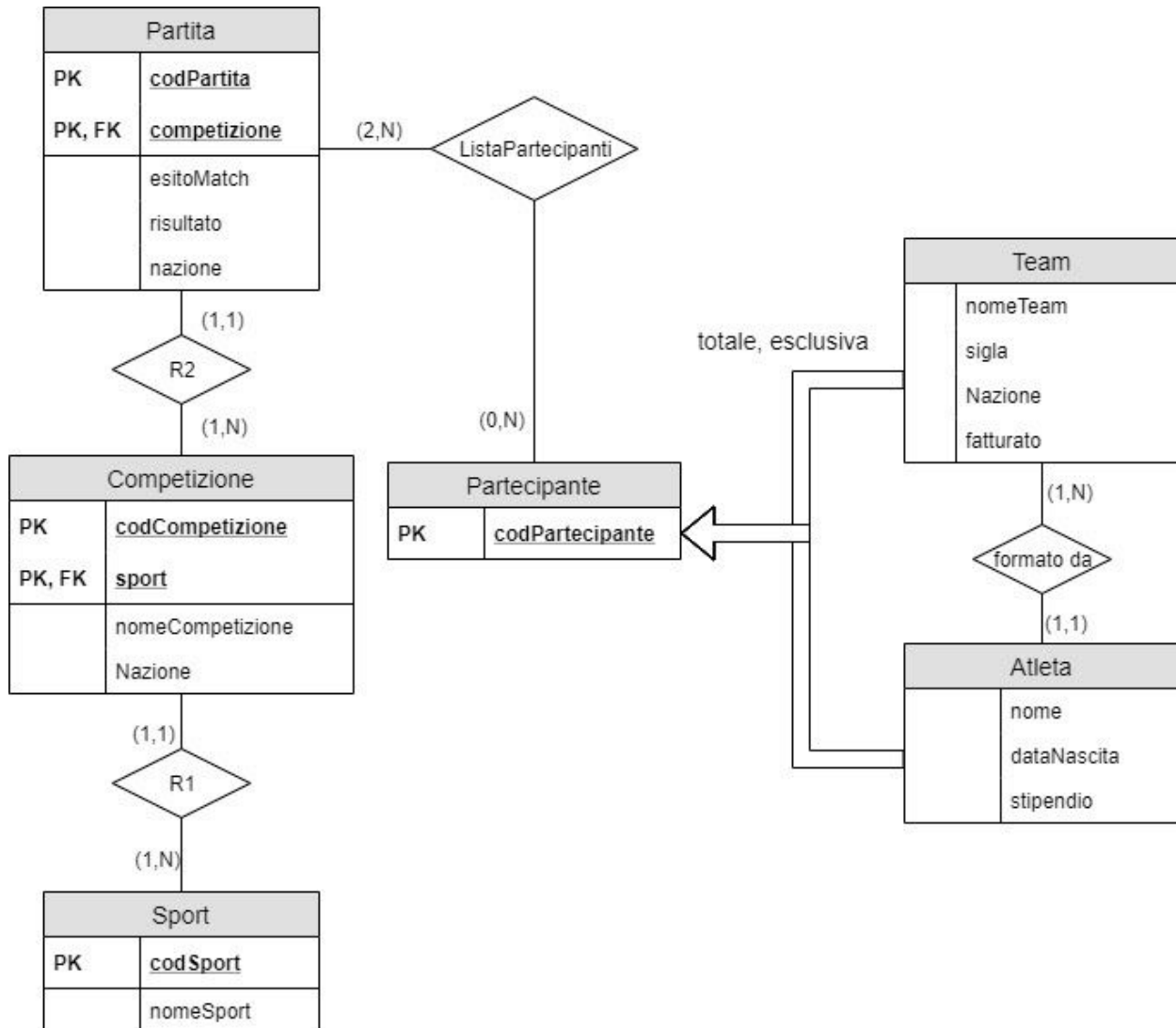
Il Database_ES è stato costruito mediante la stesura degli eventi di tutti gli sport gestiti dalla piattaforma. Questo serve a fornire la base sulla quale vengono modellate le quote dei pronostici andando ad evidenziare sia gli atleti coinvolti, sia i team nel caso di sport a squadre; come ulteriori informazioni sono inseriti anche dati più precisi in base all'evento e a chi è coinvolto in esso.

Entità	Descrizione
Sport	include tutti gli sport gestiti e ne fornisce un codice univoco
Competizione	rappresenta le competizioni di ogni sport soggette a scommessa
Evento	è un evento sportivo per cui è possibile scommettere
Partecipante	dati generali relativi al team o atleta partecipante all'evento
Team	società che partecipa ad uno sport, con informazioni aggiuntive
Atleta	entità di un singolo atleta, a volte coinvolto in una squadra

Il Database_SS, contiene gli utenti registrati sulla piattaforma e lo storico delle schedine giocate con i vari tipi di pronostici assegnati agli eventi.

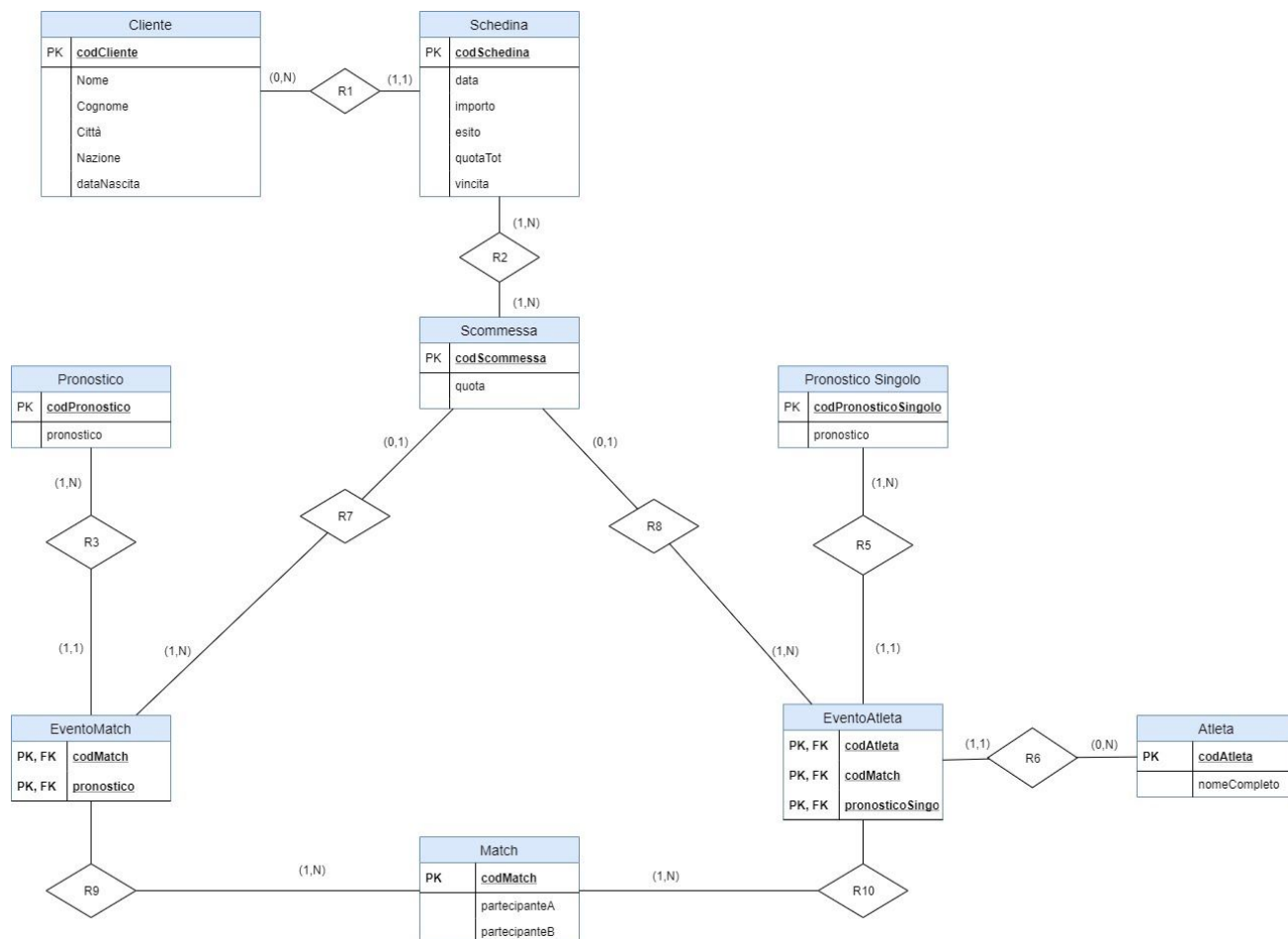
Entità	Descrizione
Cliente	contiene tutte informazioni degli utenti registrati
Schedina	rappresenta la schedina giocata dal cliente
Scommessa	scommessa all'interno di una schedina con la sua quota
Partita	partita scommessa con esito e altre informazioni
Pronostico	segno associato alla partita giocata
EventoPartita	evento speciale scommesso all'interno di una partita
PronosticoSingolo	segno associato alla scommessa su evento
Partecipante	soggetto protagonista dell'evento

2.2 Normalizzazione



A partire dallo schema del database Database_ES, abbiamo fatto una pulizia dei dati mantenendo solo le informazioni più interessanti.

A tal proposito si può evidenziare la possibilità di rimuovere l'entità "Sport": ci si accorge che oltre la chiave, il nome stesso è un fattore di identificazione univoca. Per cui nella continuazione della progettazione elimineremo tale tabella per inserire l'informazione direttamente nella competizione senza un'ulteriore aggiunta di informazione.



L'immagine sovrastante rappresenta il modello della seconda fonte, ovvero il database DB_SS. La sua analisi ci permette di sottolineare una sola peculiarità che nei passi successivi della progettazione verrà risolta. Le entità "Pronostico" e "Pronostico Singolo" risultano essere identificate univocamente dal codice del pronostico scelto, ma come nel caso dello "Sport" nella prima fonte, anche queste due entità sono univocamente identificate dal segno del pronostico stesso. Questo ragionamento porta alla loro eliminazione a seguito della fusione dei due schemi, la loro informazione sarà inserita in modo diretto durante l'alimentazione del riconciliato nelle due entità con le quali hanno una relazione: "EventoMatch" ed "EventoAtleta".

2.3 Integrazione

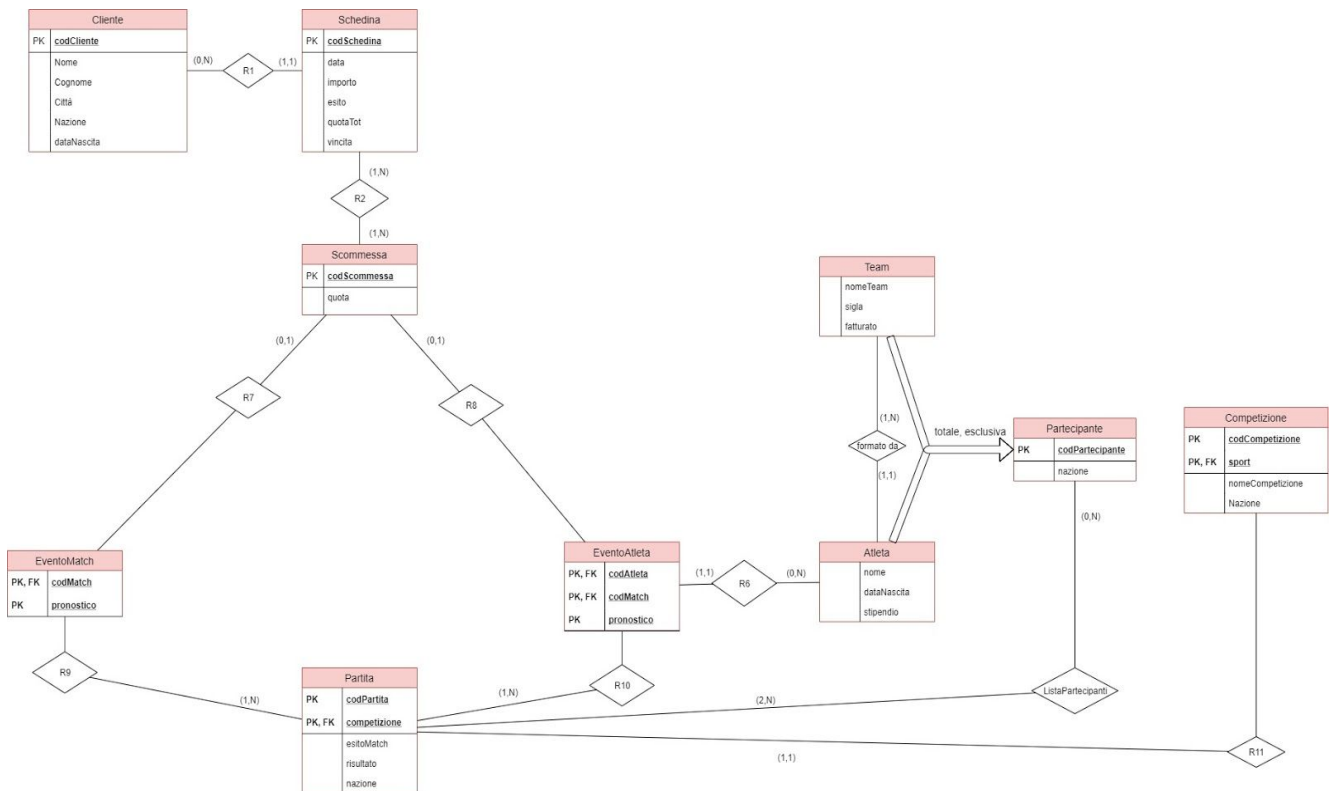
A partire degli schemi dei due database sorgente, passiamo alla loro fusione per ottenere quello che sarà il livello riconciliato.

La tecnica utilizzata è quella binaria essendo due i database di partenza e per eventuali conflitti si deciderà di privilegiare il database Database_ES. La scelta della fonte da cui prendere i dati risulta banale in quanto tra i concetti comuni dei due schemi, ovvero “Atleta” e “Match”, il database degli eventi sportivi contiene informazioni con una maggiore rilevanza per la progettazione del data warehouse.

Dopo aver comparato i due schemi risulta evidente, anche a seguito della descrizione delle tabelle del paragrafo 2.1, il caso di:

- sinonimia tra “Match” e “Partita”: le due entità rappresentano lo stesso soggetto ma differenziano per il numero e la tipologia di attributi. L'attributo *risultato* è in entrambe le rappresentazioni, sotto forma di pronostico, del risultato finale dell'evento. Per risolvere questo conflitto consideriamo di fondere le due entità mantenendo le informazioni e le relazioni di “Partita” per i motivi espressi nell'introduzione del paragrafo;
- sinonimia tra “Atleta_Partecipante” (Database_ES) e “Atleta”(Database_SS): seppur differenti nella rappresentazione le due entità hanno lo stesso significato. E' sinonimia in quanto l'atleta sul quale la scommessa è stata effettuata è il medesimo di quello registrato nel database degli eventi sportivi. Anche in questo caso riportiamo l'insieme degli attributi dell'entità della fonte Database_ES per risolvere il conflitto sul nome.

Dopo aver integrato e risolto i conflitti, si può erigere lo schema concettuale del livello riconciliato.



2.4 Progettazione del livello riconciliato

A termine di tutti i processi di analisi e ricognizione delle fonti siamo in grado di costruire un modello organizzato dello schema del livello riconciliato.

A questo punto è importante definire la storicizzazione dei dati e la corrispondenza tra gli schemi locali delle fonti e le relazioni dello schema riconciliato.

Il livello di storicizzazione è il medesimo di quello proveniente dalle sorgenti. Infatti, le marche temporali presenti nelle fonti sono semplicemente quelle riferite alle “Schedine” e all’anno della stagione di riferimento di una “Partita”; queste sono riportate anche nel livello riconciliato e nel data warehouse.

Per quanto riguarda il mapping, l’approccio seguito è quello *Global-As-View*: ad ogni concetto dello schema globale associamo una vista il cui significato è definito in base ai concetti che risiedono sugli schermi sorgenti.

Scegliamo questo approccio perchè è ovviamente più semplice e naturale scrivere le interrogazioni ed inoltre perchè abbiamo appreso che non ci saranno altre sorgenti da aggiungere, le quali porterebbero inevitabilmente alla modifica di tutti i concetti dello schema globale che le utilizzano.

A seguire un esempio del mapping del concetto Partita:

CREATE VIEW Partita **AS**

SELECT P1.partita, P1.competizione, P1.esitoMatch, P1.risultato, P1.nazione

FROM db_es.Partita as P1, db_ss.Match as P2

WHERE P1.codPartita=P2.codMatch;

3. Analisi dei Requisiti Utente

Arrivati a questo punto è stata effettuata una riunione nella sede del cliente in cui erano presenti diverse figure facenti parte della società.

Per superare la riluttanza iniziale è stato scelto un approccio di intervista ad imbuto, partendo da domande generali come “Quali sono i fattori che influenzano il processo decisionale?” fino ad arrivare a domande più specifiche per sapere ad esempio se c’è intenzione di analizzare le schedine del singolo cliente per incentivarlo a giocare con bonus specifici. Siamo venuti inoltre a conoscenza delle caratteristiche delle fonti dati disponibili e gli strumenti utilizzati fin’ora per analizzare i dati.

A seguito del colloquio apprendiamo l’interesse per 2 fatti: *Schedina* e *Scommessa*.

Schedina verrà utilizzato dagli utenti che si occupano di analizzare le schedine giocate dai vari clienti per avere un’idea su quella che è la situazione contabile del reparto.

Scommessa invece verrà utilizzato dagli utenti del sistema che si occupano di analizzare statisticamente le scommesse effettuate all’interno delle schedine, per fare analisi sugli eventi più giocati e prendere le decisioni su quelle che saranno le quote assegnate in futuro.

Inoltre nella riunione si è optato di comune accordo tra gli utenti e i gestori per un intervallo di aggiornamento con cadenza mensile per entrambi i fatti di interesse.

Per quanto riguarda la dinamicità delle gerarchie è naturale pensare di voler effettuare interrogazioni secondo gli scenari temporali oggi o ieri.

A termine del colloquio abbiamo redatto un breve glossario dei requisiti utente emersi.

Fatto	Possibili Dimensioni	Possibili misure	Storicità
Schedina	Cliente, Esito, Data, Orario	numeroSchedine vincita totale vincita avg importo giocato tot importo giocato avg bilancio	8 anni
Scommessa	Team, pronostico, Atleta, Data, risultato, Competizione	occorrenze	8 anni

In particolare, sul fatto Schedina saranno effettuate analisi in merito alle sue possibili dimensioni volendo ottenere informazioni sulle misure:

- numero schedine: rappresenta il numero di schedine giocate ed è misura di flusso che a seguito delle analisi risulta cumulativa;
- vincita totale e media: rappresenta il potenziale incasso del cliente a seguito di una schedina con esito positivo;
- bilancio: a seconda dell'esito della schedina il bilancio rappresenta l'incasso effettivo della società come differenza tra importo giocato e vincita della schedina. Sarà positivo qualora la schedina risulti con esito "false", negativo altrimenti.

Sul secondo fatto, Scommessa, l'unica misura rilevante è "occorrenze" la quale indica per ogni evento il numero di volte in cui è apparso in una schedina giocata.

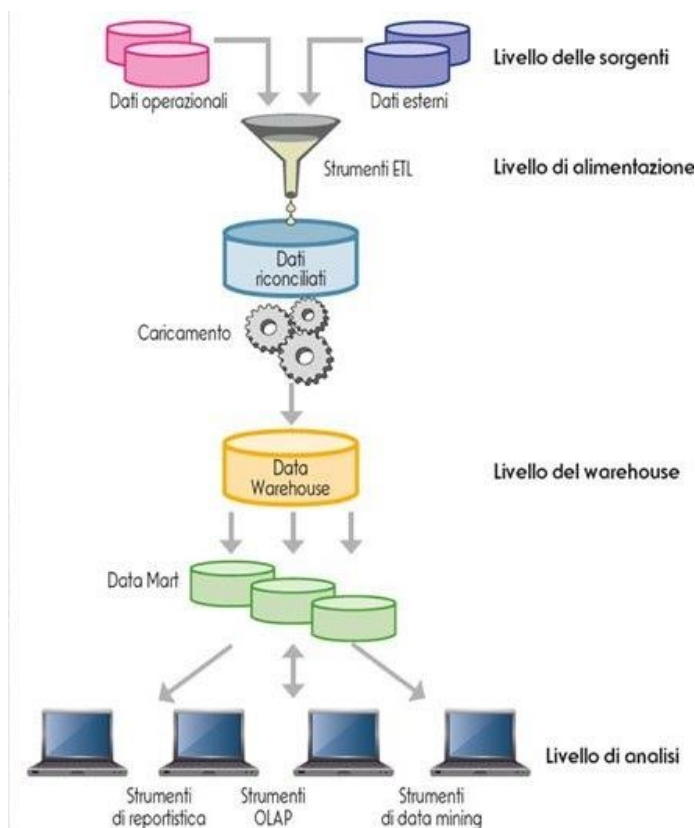
Inoltre, a seguito di domande specifiche agli utenti del sistema è stato estratto il campione del carico di lavoro preliminare seguente:

Fatto	Interrogazione
Schedina	Incasso dovuto a schedine con esito negativo dei clienti negli anni. Quantità di schedine vinte in una determinata città. Importo giocato per fasce d'Età.
Scommessa	Analisi su scommesse di una squadra in particolare o di un atleta.. Pronostici maggiormente giocati dagli utenti.

4. Pianificazione del DW

Come introdotto precedentemente, l'architettura scelta per supportare il nostro data warehouse è quella a tre livelli.

Adottiamo quindi uno schema classico per la sua realizzazione.



L'adozione di questa architettura comporta il vantaggio di escludere il data warehouse dai problemi legati all'integrazione e all'estrazione dei dati dalle sorgenti e introduce quindi un grado maggiore di "separabilità" dalle fonti.

Rilevante per la pianificazione del data warehouse è stata la scelta dell'approccio *bottum-up*. Così facendo il focus è posto sulla realizzazione del primo data mart, quello delle Scommesse Online, che si può ottenere in maniera quasi algoritmica dallo schema concettuale. Inoltre, come metodo di progettazione è stato scelto quello *data-driven*, ovvero un approccio guidato dai dati. Questa scelta è stata effettuata poiché semplifica il processo di progettazione dell'ETL, l'analisi dei requisiti e grazie anche al buon grado di normalizzazione e poca complessità delle fonti operazionali.

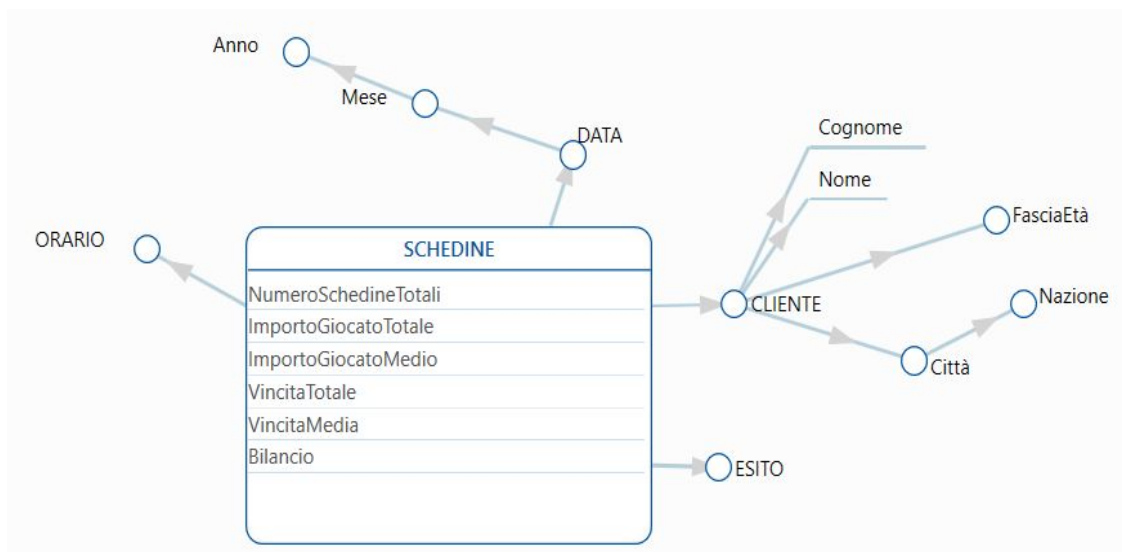
Per il livello del warehouse possiamo aggiungere ulteriori dettagli riguardanti la sua architettura in quanto supportiamo nella soluzione data mart dipendenti. Questo vuol dire che essi risultano essere delle porzioni di dati provenienti da quelli raccolti nel data warehouse e che, in scenari futuri, potranno essere inseriti concentrandosi solo sul loro sviluppo.

I software utilizzati in supporto alla progettazione sono: Pentaho Data Integration, Pentaho Schema Workbench (Mondrian), Pentaho Aggregator Designer e MySQL. Infine, le analisi OLAP potranno essere eseguite dagli utenti mediante l'utilizzo di Pentaho Server con lo strumento Saiku.

5. Progettazione Concettuale

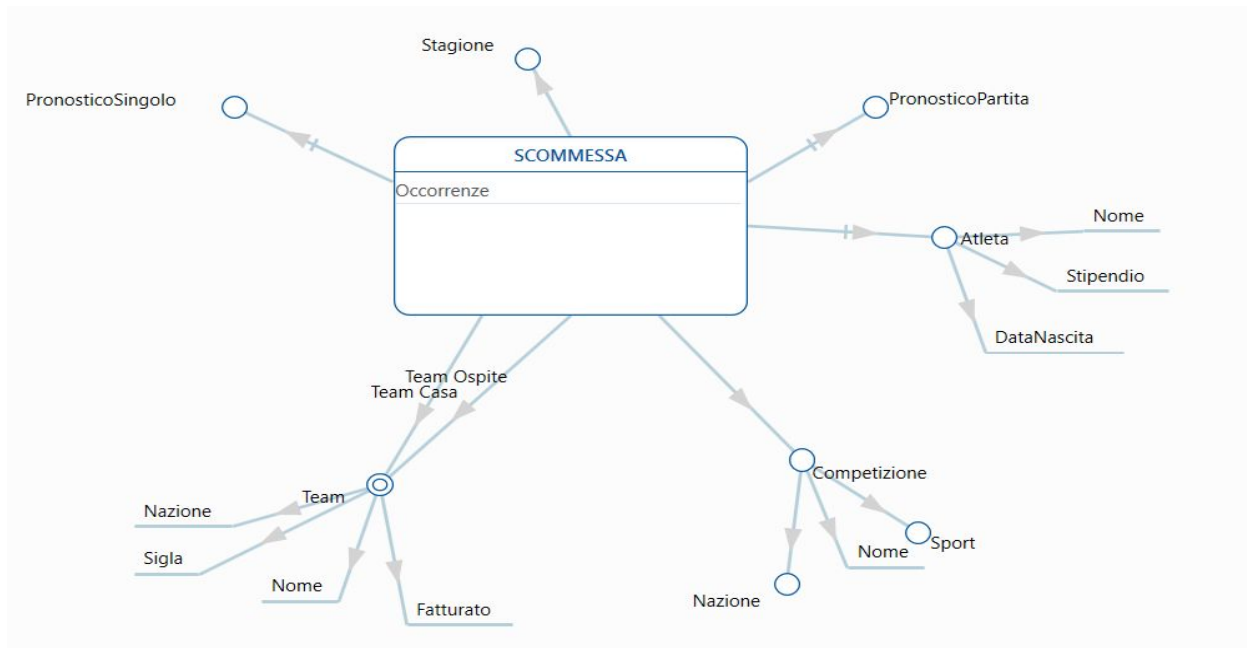
I fatti di interesse venuti fuori a seguito dell'analisi dei requisiti utenti sono Schedina e Scommessa, corrispondenti alle omonime relazioni.

Il processo di progettazione concettuale iniziato con operazioni di aggiustamento dell'albero degli attributi, ha portato alla realizzazione dei due seguenti DFM:



Rispetto ai dati del livello riconciliato, si è deciso di modellare alcuni attributi dell'entità come attributi dimensionali del fatto e di lavorare sulle loro gerarchie. In particolare, la gerarchia data è composta dal mese e dall'anno mentre in quella del cliente abbiamo deciso di mantenere descrittivi gli attributi "Nome" e "Cognome" in quanto non

importanti per eventuali analisi. Per concludere il cliente è legato ad un attributo dimensionale “Città” con dipendenza funzionale su “Nazione”, utile nel caso di aggregazione su zone geografiche.



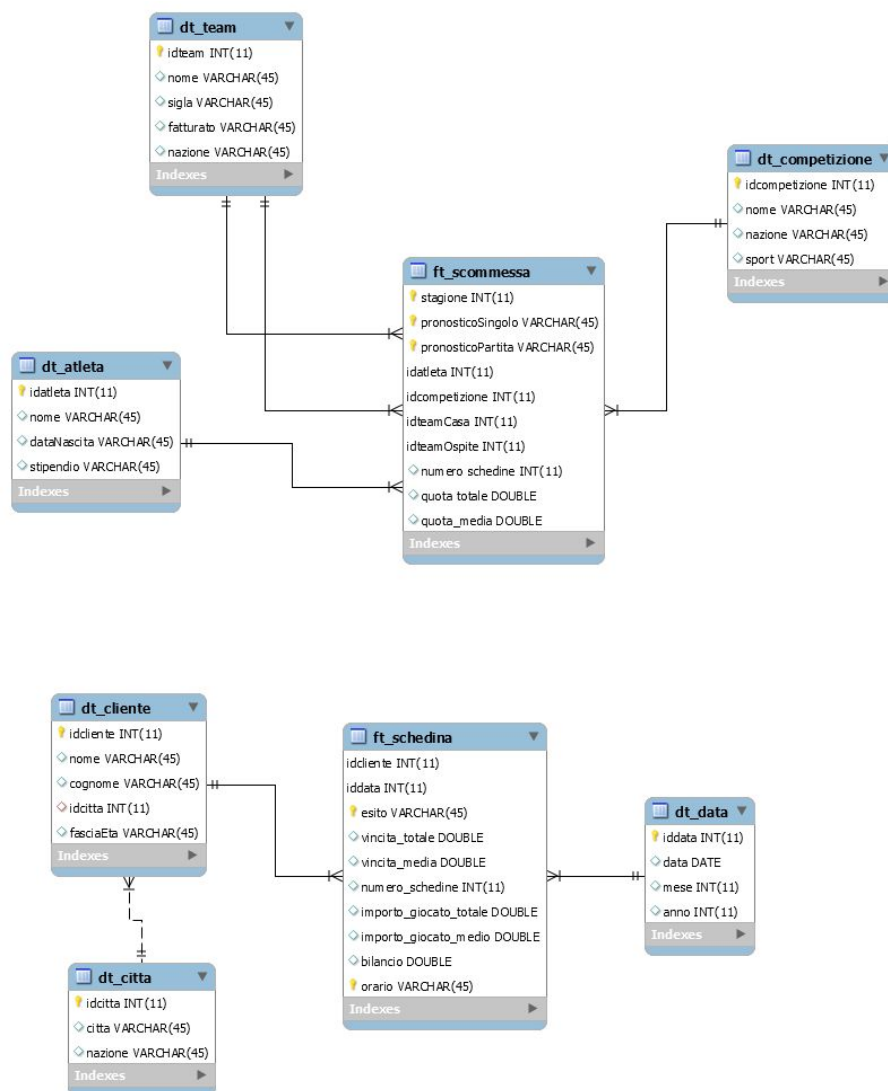
Per il secondo DFM abbiamo deciso di innestare l'attributo “Partita” direttamente sul fatto d'interesse collegandogli i due “team” partecipanti, l'anno della “stagione” e la competizione, procedendo con l'eliminazione degli attributi residui (risultato, esitoMatch) in quanto dimensioni poco interessanti dal punto di vista dell'analisi. Per il medesimo motivo, abbiamo rimosso il riferimento al team d'appartenenza dell'atleta.

Per natura stessa del fatto, la scommessa può riferirsi o ad un pronostico giocato su una partita, rappresentato dalla dimensione “PronosticoPartita”, o ad un pronostico scelto riguardante un atleta, rappresentato dalla dimensione “PronosticoSingolo”. Per questo motivo i loro archi sono opzionali.

6. Progettazione Logica

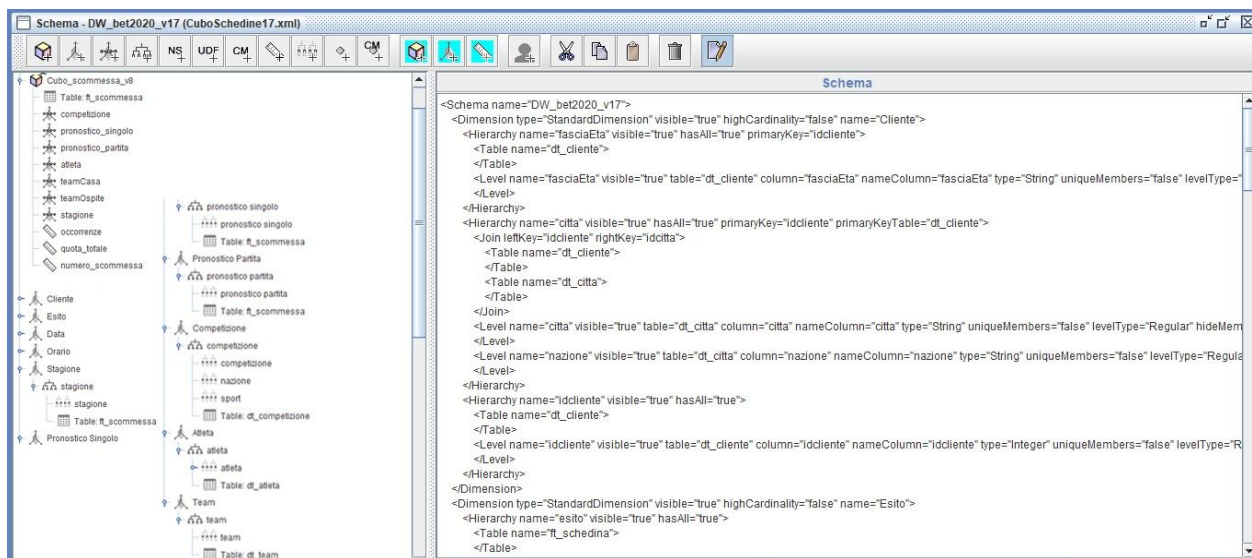
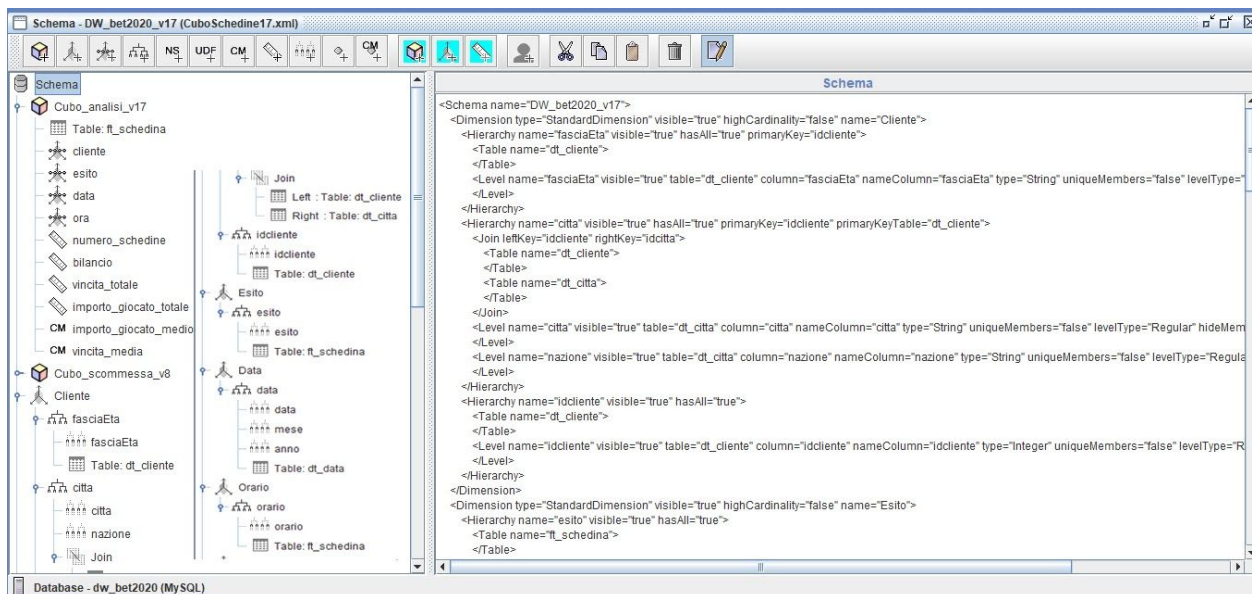
A seguito della progettazione concettuale abbiamo ottenuto i due DFM che descrivono i fatti di interesse, andremo ora a creare quello che sarà il modello logico per il nostro data warehouse.

Abbiamo deciso di adottare una soluzione ROLAP che ci permetterà di memorizzare enormi quantità di dati, andando a implementare una soluzione con schema logico Snowflake per quanto riguarda il fatto schedina, poichè si è deciso di normalizzare la gerarchia di “città” da “cliente”(eliminando la ridondanza e riducendo lo spazio richiesto per la memorizzazione, poichè la cardinalità delle città è di molto inferiore ai clienti) e di mantenere lo Star Schema naturale per il fatto Scommessa.



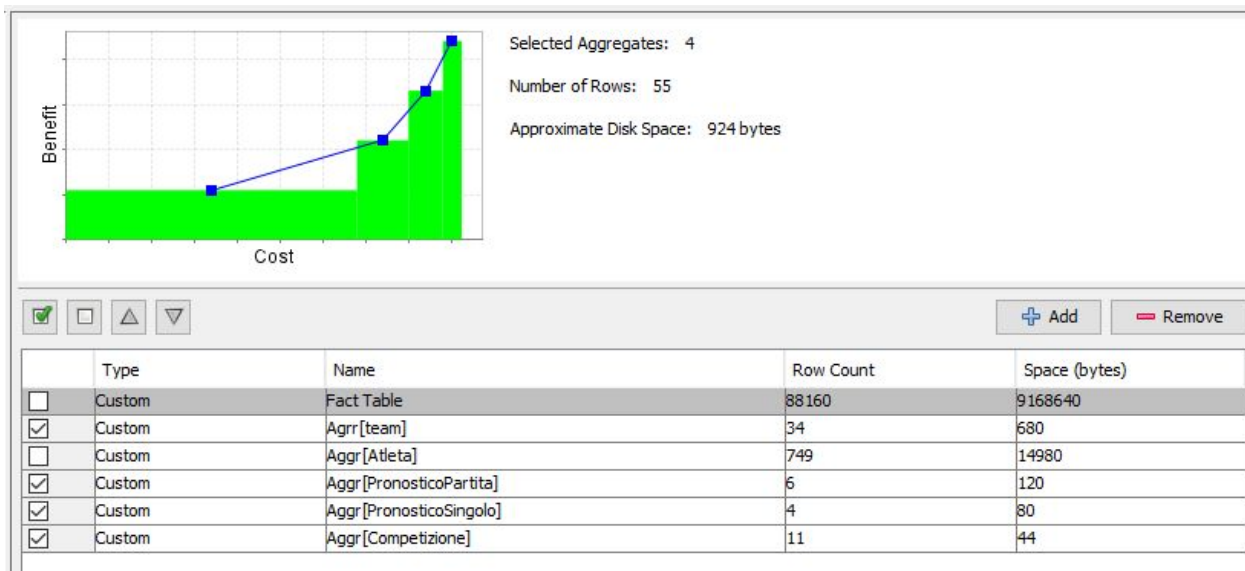
Una volta ottenuto lo schema logico di riferimento, con il software Pentaho Schema Workbench, siamo passati alla creazione dei cubi per ottenere l'XML descriptor che verrà utilizzato da Pentaho Aggregator Designer per la materializzazione delle viste, ma successivamente anche da Pentaho Server per le analisi con Saiku.

A seguire qualche screen dell'area di lavoro di PSW per la creazione dei due cubi.

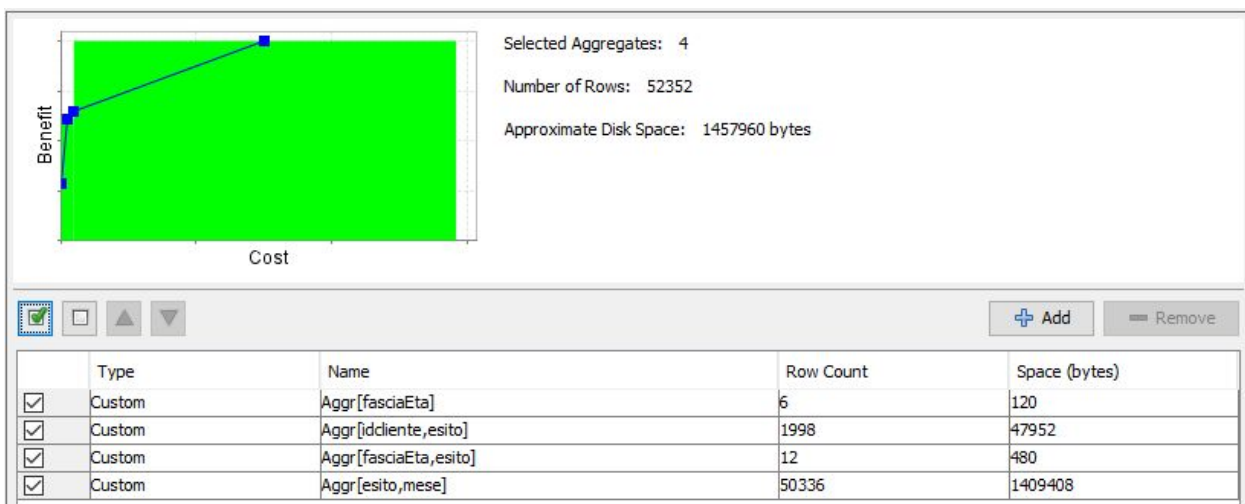


Successivamente siamo passati ad analizzare i colloqui avuti con gli utenti per identificare un carico di lavoro preliminare, ovvero quelle interrogazioni a cui il sistema verrà sottoposto più frequentemente.

Utilizzando in nostro supporto il software Pentaho Aggregator Designer, abbiamo fatto delle analisi costi/benefici sulle viste più utili da materializzare, come possiamo vedere nei grafici sottostanti.



Materializzando queste viste abbiamo la possibilità di avere la pre-aggregazione sulle scommesse di una squadra, un'atleta o una competizione in particolare. Altrettanto utile risulta avere la pre-aggregazione sulle tipologie di pronostico.



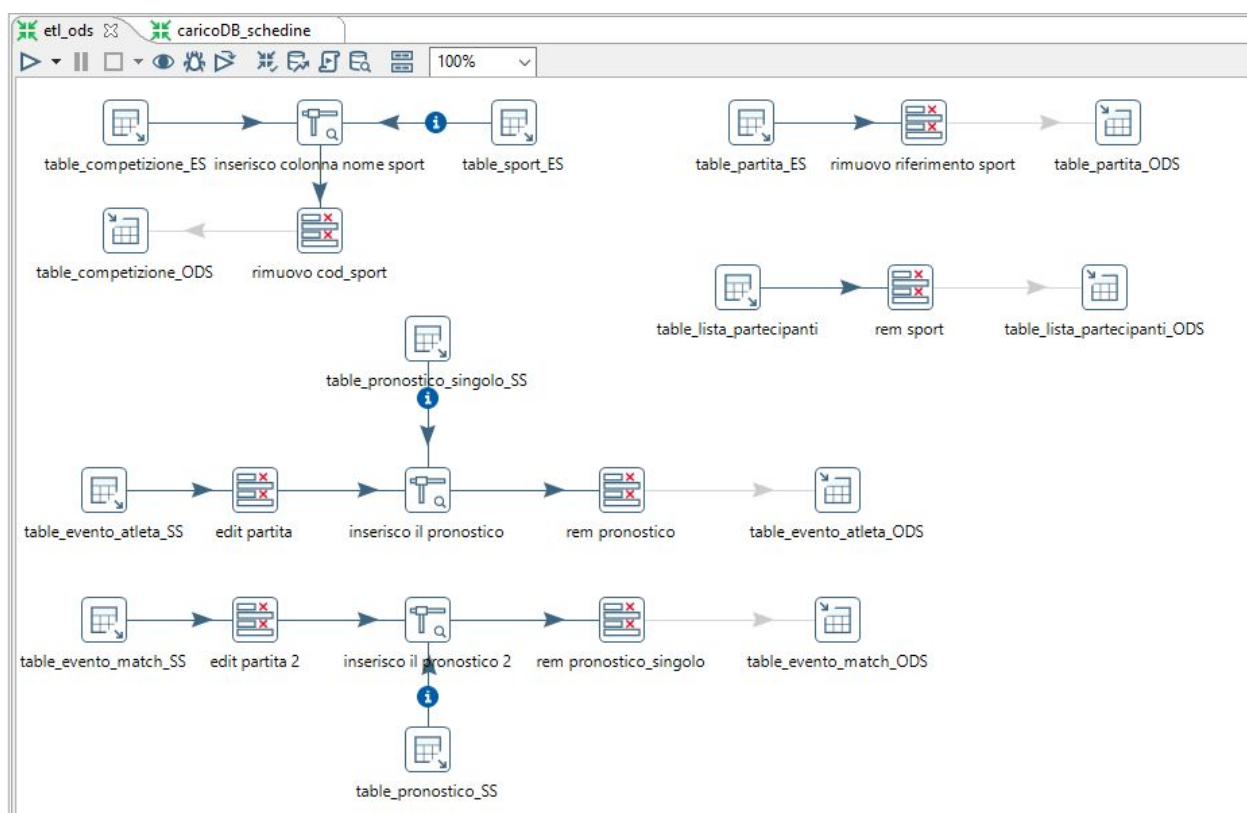
Molto frequenti erano inoltre le interrogazioni per fascia d'Età e per quanto riguarda l'esito in modo da avere dei feedback numerici su quello che sarà l'introito della società. Per questo motivo abbiamo deciso di materializzare anche queste viste che, a fronte di un'occupazione spaziale ampiamente nei range consentiti dall'hardware aziendale, apporta grossi benefici in termini di velocità di esecuzione delle interrogazioni.

7. Progettazione dell'Alimentazione

La scelta di adottare un'architettura a tre livelli si ripercuote nella necessità di effettuare due fasi di alimentazione dei dati, ovvero: alimentazione del livello riconciliato e alimentazione dei data mart. Le operazioni di alimentazione eseguite sono state le classiche di estrazione, trasformazione (compresa di pulizia) e caricamento.

La prima fase riguarda l'estrazione dei dati e il loro successivo caricamento nel livello ODS (*Operational Data Source*). Il punto di partenza del progetto, ovvero con data warehouse inesistente, ci ha costretti nell'adozione di una strategia d'estrazione statica durante il quale tutti i dati provenienti dalle sorgenti operazionali sono stati estratti mediante delle "lookup" sulle tabelle. Avendo inoltre scelto un approccio *data-driven* la fase di trasformazione e di pulizia è stata semplificata: solo alcune tabelle sono state oggetto di trasformazione.

Nella figura sottostante sono riportate le operazioni effettuate mediante il software Pentaho Data Integration (PDI).

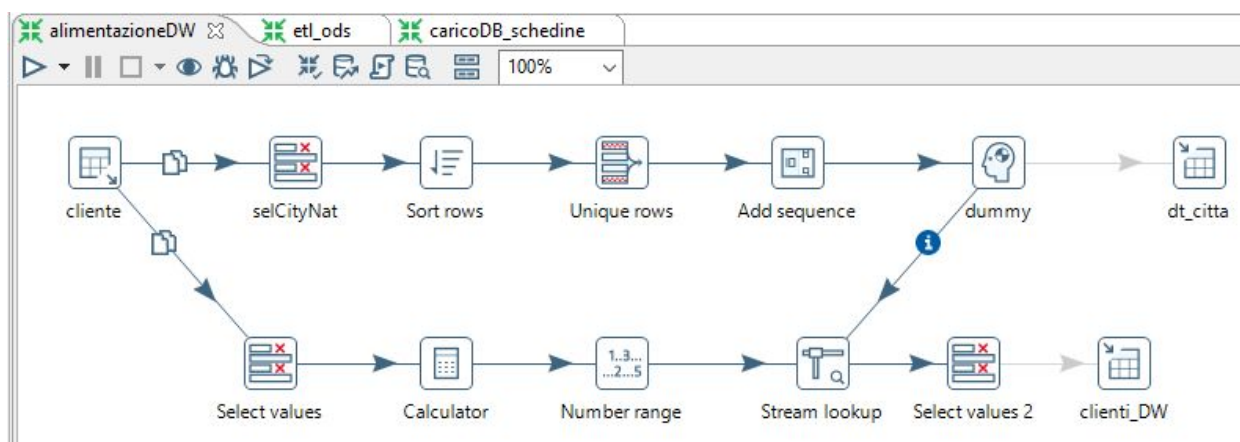


Come si evince dalla figura, le tabelle delle fonti sono state soggette a pulizia dal riferimento dello sport in quanto, come evidenziato nel Capitolo 2, può essere inserito nella competizione. Diverse operazioni sono state effettuate sulle tabelle “Evento Match” ed “Evento Atleta” nelle quali il riferimento al codice del pronostico è stato sostituito con il pronostico stesso.

La fase di caricamento si risolve nell’ultima freccia degli schemi di PDI le quali indicano che le informazioni pulite e trasformate sono state inserite nelle rispettive tabelle del livello riconciliato. Le altre tabelle omesse dall’immagine descritta non hanno necessitato di operazioni particolari e il loro caricamento è stato molto semplice all’interno delle rispettive tabelle ODS.

La seconda fase dell’alimentazione è risultata invece più complessa in vista della necessità di adattare i dati ai fatti del data mart discussi precedentemente. Inoltre, a causa dei vincoli stringenti del data warehouse riguardo dimension e fact table, l’alimentazione del data warehouse ha seguito due *step*:

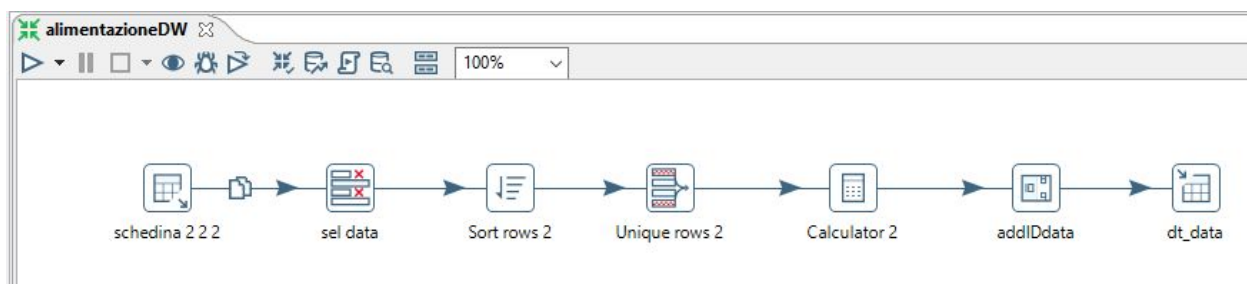
- alimentazione delle dimension table: in cui abbiamo caricato i dati, a valle di ulteriori operazioni di etl, nelle dimension table primarie e secondarie;
- alimentazione delle fact table: durante questa fase abbiamo inserito i dati nelle fact table dei fatti Schedina e Scommessa.



La precedente figura mostra i passi che descrivono, in PDI, le operazioni di ETL per alimentare le dimension table di città e clienti. Dapprima viene eseguito il ramo che porta al caricamento in “dt_citta” in quanto dimension table secondaria e successivamente quello che inserisce i clienti all’interno del data warehouse. Le città a cui i clienti appartengono sono estratte dalla tabella “Cliente” del livello riconciliato e sono associate ad una chiave surrogata, mediante il passo “Add sequence”. Per

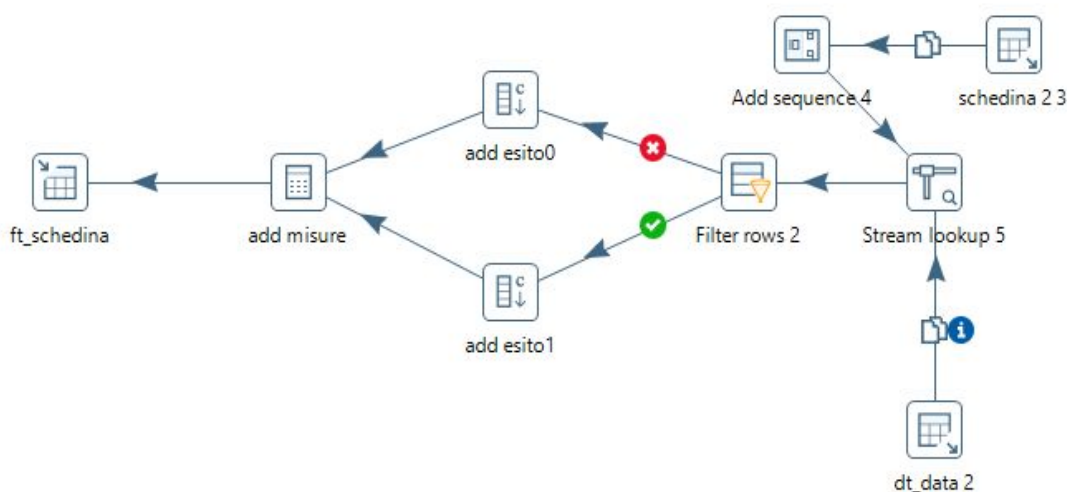
ottenere una corretta normalizzazione sono state filtrate (*Sort rows* e *Unique Rows*) in modo che per ogni città appaia una sola voce nella dimension table.

Per il cliente abbiamo iniziato le operazioni di trasformazione ricavando la sua età dalla data di nascita e associando la fascia d'età d'appartenenza necessaria per le analisi; successivamente, sono ripuliti tutti i dati non necessari e ridondanti. La città d'appartenenza è stata sostituita con la chiave surrogata.



Questa seconda figura mostra la realizzazione della dimension table delle date. Questa necessita di una chiave surrogata per esprimere le gerarchia di data, mese ed anno. Per riuscire a realizzare tale tabella abbiamo: estratto i dati dall'entità Schedina dell'ODS; ripulito tutti i campi senza informazioni sulla data stessa; ordinato la colonna rimanente; infine, estratto i dati sul mese e sull'anno. Il caricamento dei dati estratti è poi avvenuto a seguito dell'assegnamento di una chiave surrogata per ogni riga unica della tabella.

L'ultima trasformazione necessaria per completare il primo fatto, Schedina, del data mart è stata quella mostrata nella figura sottostante.



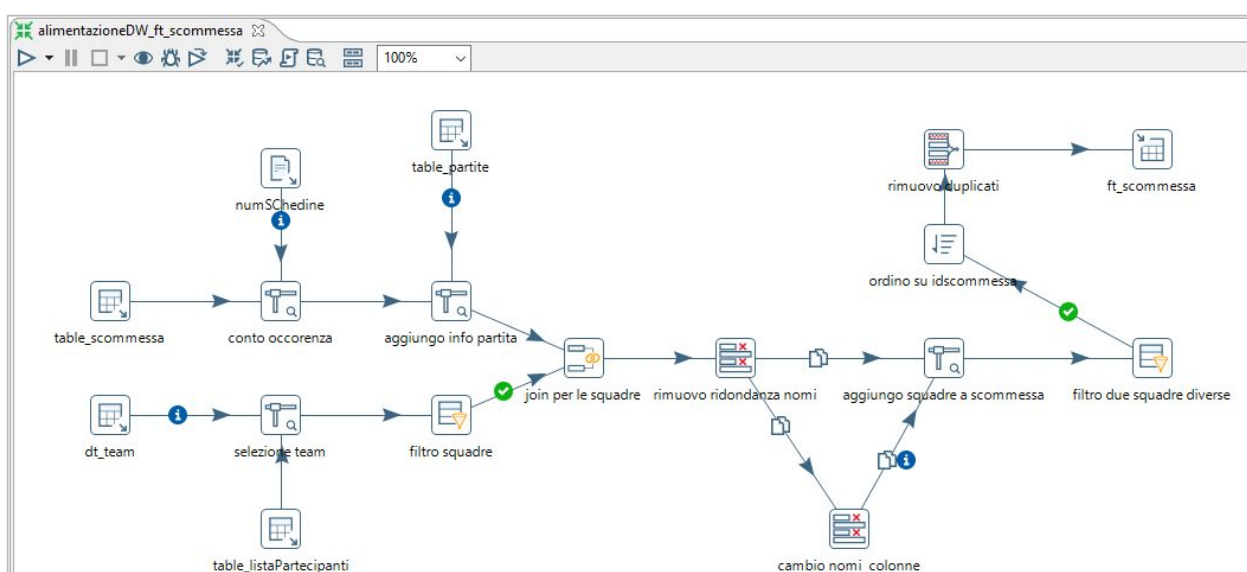
Questa descrive le operazioni ETL effettuate per inserire i dati nella fact table. Estratti staticamente i dati dal livello riconciliato tramite una *lookup* sulle date abbiamo sostituito il campo con la chiave surrogata della dimension table già presente nel data warehouse. Le informazioni mancanti nel livello riconciliato sono quelle delle misure necessarie alle analisi. Per inserirle nella fact table abbiamo inserito un passo di PDI che consente di performare calcoli matematici ("*Add measure*"). In particolare, siamo riusciti per la singola schedina a definire in maniera rapida la maggior parte delle misure mentre il bilancio ci ha costretto a suddividere, temporaneamente, le schedine in vinte e perse. Questo perché la misura bilancio è stata ottenuta mediante la seguente formula:

$$bilancio = importo_{giocato} - esitoI * vincita_{potenziale}$$

dove *esitoI* è un intero a due valori: 0, se l'esito della schedina risulta negativo, 1 altrimenti. L'inserimento nelle operazioni ETL di questo intero è stato fatto a valle del filtro sul reale esito della schedina ("*Filter rows*").

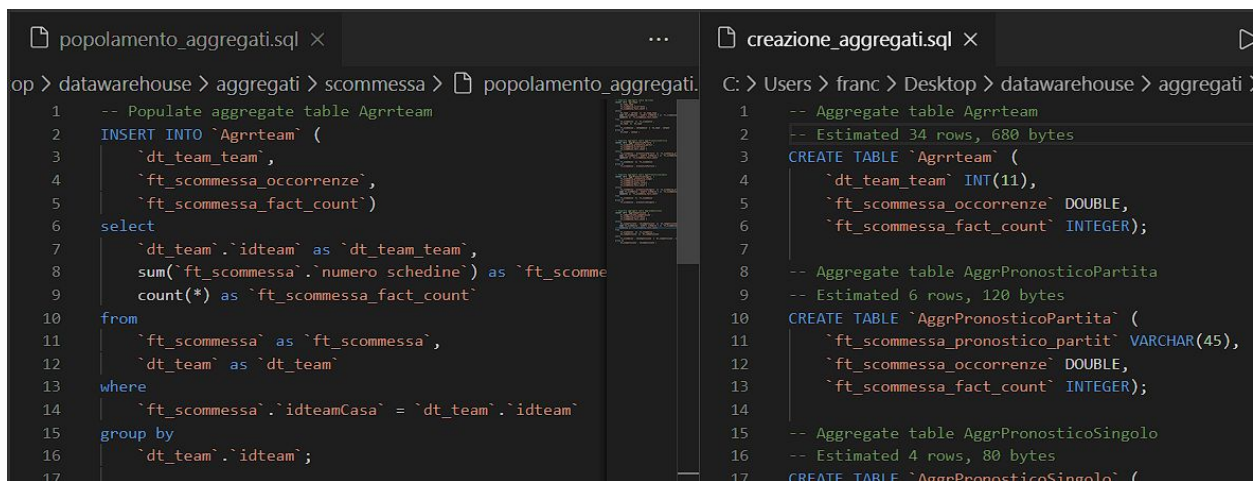
L'alimentazione del secondo fatto è stata più immediata in quanto le dimension table necessarie hanno avuto bisogno di meno operazioni di ETL. Per le dimension table di Atleta, Competizione e Team abbiamo dovuto semplicemente caricare i dati in quanto già ben formattati e organizzati; l'unico accorgimento che abbiamo dovuto sostenere è stato quello relativo all'eliminazione dalla tabella dell'atleta del team appartenente per rispecchiare il DFM proposto durante la progettazione concettuale.

Le operazioni di ETL maggiori sono state poi effettuate prima della fase di alimentazione della fact table della Scommessa. Il processo PDI che porta al caricamento è mostrato nella figura sotto riportata.



Il processo è partito da sorgenti contenenti le tabelle del livello ODS, successivamente sono stati effettuati passi per contare le occorrenze di una scommessa nelle schedine dell'anno e, infine, inseriti gli id dei due team partecipanti, oltre a quello dell'atleta a cui la scommessa eventualmente si riferisce. Poiché a seguito di queste operazioni per un match si ripetevano più volte gli stessi valori (causato dalla "join per le squadre") l'introduzione di un passo di filtraggio sui valori unici ("*Unique rows*") è stato necessario prima di poter inserire i dati nella fact table.

Il data warehouse nella sua principale progettazione è stato così completato. Il passo successivo è stato quello di determinare l'alimentazione delle viste scelte da materializzare. Per effettuare tale operazione ci siamo serviti di Pentaho Aggregator Designer il quale ci ha consentito di rendere più veloci queste operazioni mediante due funzioni: esportazione del codice SQL per la creazione e per il loro popolamento. La figura seguente mostra due esempi del codice estratto ed eseguito poi in PDI per la creazione automatizzata delle viste.



```
popolamento_aggregati.sql ×
op > datawarehouse > aggregati > scommessa > popolamento_aggregati.
1  -- Populate aggregate table Agrrteam
2  INSERT INTO `Agrrteam` (
3    `dt_team_team`,
4    `ft_scommessa_occorrenze`,
5    `ft_scommessa_fact_count`)
6  select
7    `dt_team`.`idteam` as `dt_team_team`,
8    sum(`ft_scommessa`.`numero schedine`) as `ft_scommessa_fact_count`
9  from
10   `ft_scommessa` as `ft_scommessa`,
11   `dt_team` as `dt_team`
12  where
13   `ft_scommessa`.`idteamCasa` = `dt_team`.`idteam`
14  group by
15   `dt_team`.`idteam`;
16
17

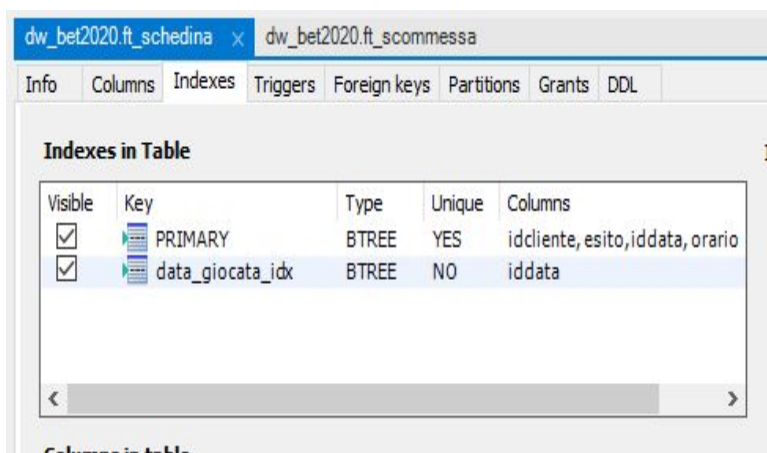
creazione_aggregati.sql ×
C: > Users > franc > Desktop > datawarehouse > aggregati
1  -- Aggregate table Agrrteam
2  -- Estimated 34 rows, 680 bytes
3  CREATE TABLE `Agrrteam` (
4    `dt_team_team` INT(11),
5    `ft_scommessa_occorrenze` DOUBLE,
6    `ft_scommessa_fact_count` INTEGER);
7
8  -- Aggregate table AggrPronosticoPartita
9  -- Estimated 6 rows, 120 bytes
10 CREATE TABLE `AggrPronosticoPartita` (
11   `ft_scommessa_pronostico_partit` VARCHAR(45),
12   `ft_scommessa_occorrenze` DOUBLE,
13   `ft_scommessa_fact_count` INTEGER);
14
15 -- Aggregate table AggrPronosticoSingolo
16 -- Estimated 4 rows, 80 bytes
17 CREATE TABLE `AggrPronosticoSingolo` (
```

8. Progettazione fisica

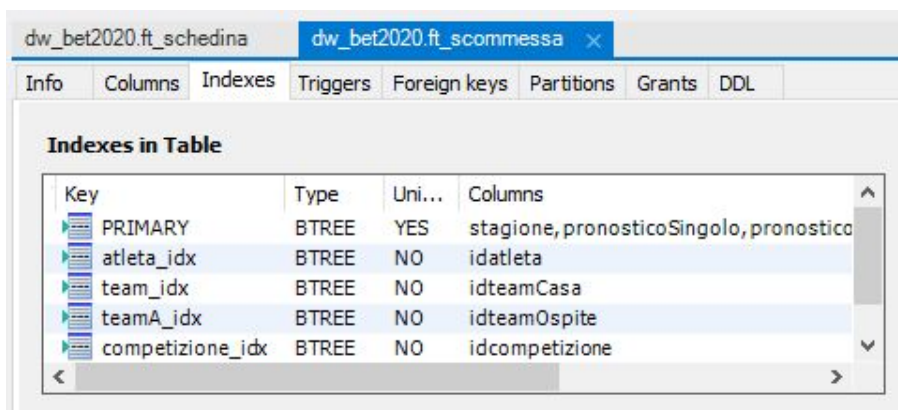
In questa fase ci siamo preoccupati di analizzare quali tra gli indici possono supportare meglio la creazione del data warehouse ed in particolare la costruzione di fact e dimension table.

Data la scelta di utilizzare come DBMS il software MySQL e come motore di salvataggio dati InnoDB, siamo venuti incontro ad un vincolo di tale software: tutti gli indici vengono realizzati solo mediante B+Tree.

Abbiamo deciso di implementarli sulle chiavi primarie delle dimension table e sulla chiave composta delle due fact table. In particolare, ogni dimension table ha indice sulla propria chiave primaria ad eccezione del cliente che conterrà il riferimento alla dimension table delle città. Le fact table, invece, manterranno sia un indice sulla propria chiave composta che i singoli riferimenti.



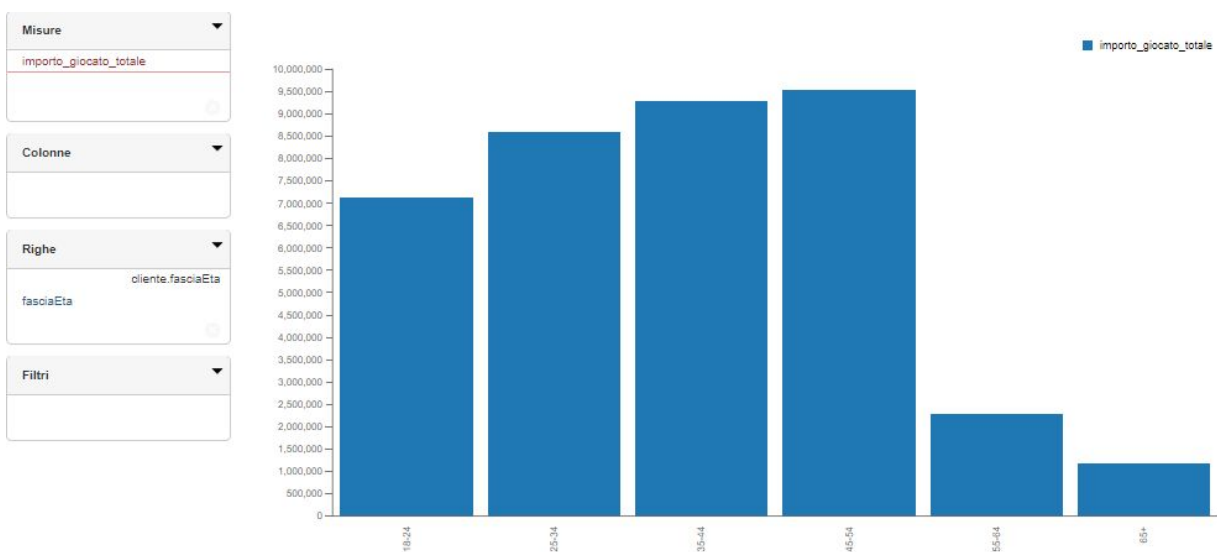
Visible	Key	Type	Unique	Columns
<input checked="" type="checkbox"/>	PRIMARY	BTREE	YES	idcliente, esito, iddata, orario
<input checked="" type="checkbox"/>	data_giocata_idx	BTREE	NO	iddata



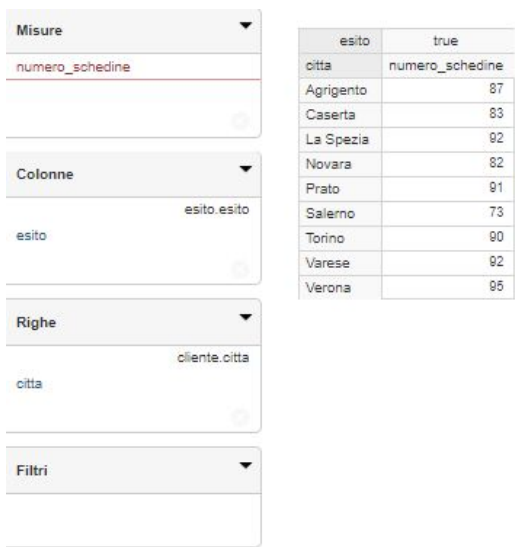
Key	Type	Uni...	Columns
PRIMARY	BTREE	YES	stagione, pronosticoSingolo, pronostico
atleta_idx	BTREE	NO	idatleta
team_idx	BTREE	NO	idteamCasa
teamA_idx	BTREE	NO	idteamOspite
competizione_idx	BTREE	NO	idcompetizione

9. Conclusione

Finita la fase di progettazione siamo passati alla fase di testing insieme agli utenti che usufruiranno degli strumenti di analisi, per chiarire i dubbi riguardo l'utilizzo. Con l'ausilio di Pentaho Server e lo strumento Saiku abbiamo effettuato qualche analisi sui dati a disposizione con ottimi risultati. Alleghiamo per concludere qualche risultato grafico.

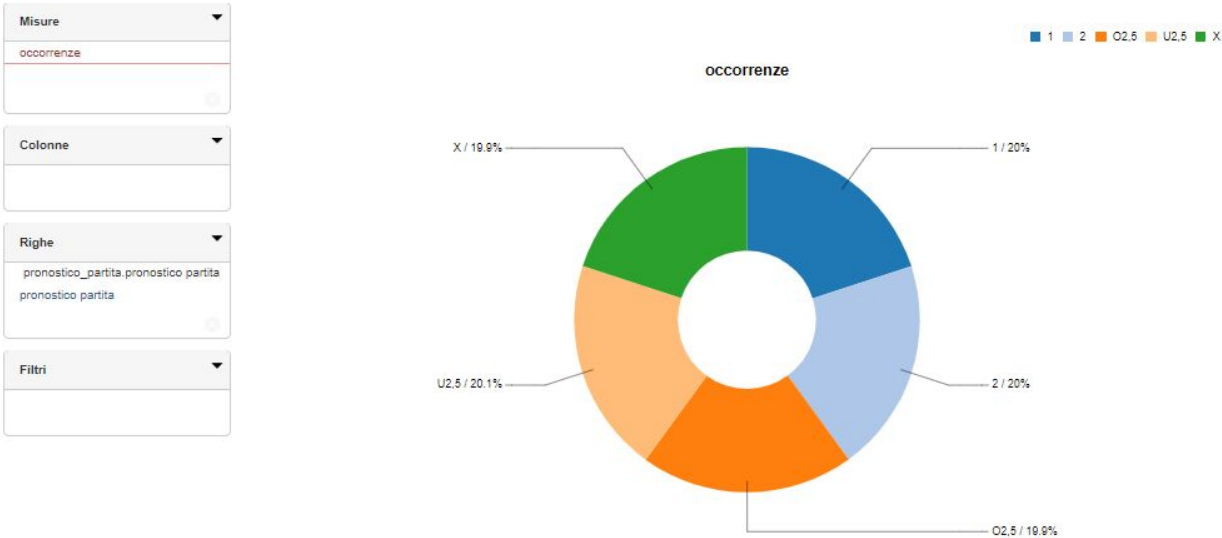


Nel grafico sovrastante analizziamo l'importo giocato per fasce d'Età, chiaramente dominato dalla fascia 45-54.

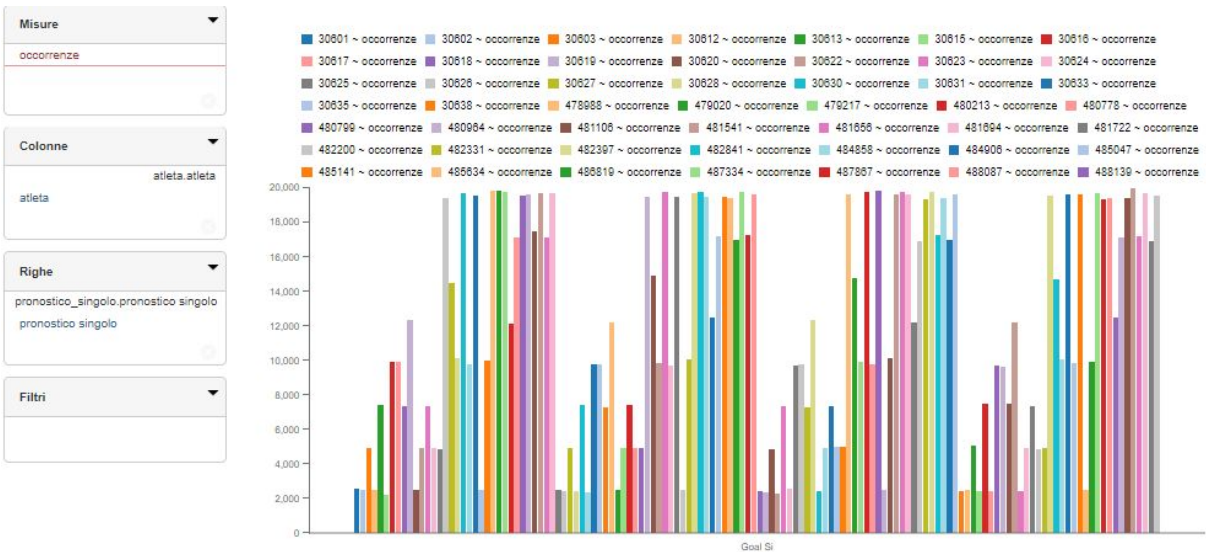


A sinistra un'analisi sul numero di schedine vinte su un sottoinsieme delle città.

A seguire la percentuale del numero di occorrenze dei pronostici della partita giocati.



Volendo analizzare per una serie di Atleti selezionati il numero di volte sui quali è stato pronosticato “Goal Si” otteniamo il seguente grafico:



Rimanendo in tema pronostici possiamo osservare l'analisi delle occorrenze dei pronostici scommessi su alcuni team selezionati:

Misure

occorrenze

Colonne

stagione.stagione

stagione

pronostico_partita.pronostico partita

pronostico partita

Righe

teamCasa.team

team

Filtri

stagione	2009					2014				
pronostico partita	1	2	O2,5	U2,5	X	1	2	O2,5	U2,5	X
team	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze	occorrenze
8191	1.243	1.182	1.204	1.285	1.247	1.237	1.220	1.190	1.260	1.249
8455	1.284	1.266	1.224	1.237	1.224	1.204	1.177	1.270	1.233	1.256
8462	1.184	1.220	1.239	1.181	1.208	-	-	-	-	-
8528	1.238	1.253	1.261	1.224	1.298	-	-	-	-	-
8602	1.248	1.293	1.197	1.254	1.225	-	-	-	-	-
8659	-	-	-	-	-	1.235	1.190	1.233	1.239	1.194
10252	1.203	1.208	1.202	1.231	1.248	1.254	1.293	1.250	1.258	1.278

Ultimo ma non per ordine di importanza l'incasso relativo a schedine con esito negativo nei vari anni.

