DATA WAREHOUSING: FRONT-END E OPERAZIONI OLAP

Slide 2 - Sommario

Benvenuti!

In questa lezione vedremo per prima cosa quali strumenti possono essere utilizzati per analizzare i dati memorizzati in un Data Warehouse.

Ci concentreremo poi sugli strumenti OLAP e introdurremo le principali operazioni utili a fini analitici.

Infine, descriveremo le principali estensioni dello standard SQL per eseguire operazioni OLAP su dati relazionali.

Cominciamo…

Slide 3 - Strumenti per interagire con un Data Warehouse

Esistono diversi strumenti per interagire con un Data Warehouse e si differenziano sia per il tipo di interazione proposto sia per il tipo di informazione che permettono di estrarre.

Tra questi ricordiamo:

- Strumenti per la reportistica che permettono di creare report di dati utilizzando format predefiniti
- Strumenti OLAP che permettono di analizzare i dati attraverso l'uso di specifiche operazioni analitiche
- Strumenti statistici che permettono di analizzare i dati in base ad analisi di tipo statistico
- Strumenti di Data Mining che permettono di estrarre, dai dati, conoscenza precedentemente sconosciuta e rappresentata in termini di modelli e pattern.

In questa lezione ci concentreremo sugli strumenti OLAP.

Slide 4 – Strumenti OLAP

Gli strumenti OLAP sono utili quando non è possibile utilizzare report predefiniti per rappresentare i dati di interesse perché le richieste analitiche potrebbero non essere note a priori.

Questi strumenti sono invece molto flessibili e permettono di specificare operazioni OLAP arbitrarie che, nel caso siano eseguite in ambienti ROLAP, verranno tradotte in comandi SQL.

In una sessione di lavoro, le operazioni vengono applicate in sequenza ai risultati ottenuti al passo precedente.

Gli strumenti OLAP permettono quindi di raffinare via via i risultati analitici ottenuti e di analizzare il fatto di interesse secondo punti di vista e livelli di dettaglio differenti.

Slide 5 - Operazioni OLAP

Le operazioni OLAP offrono diverse funzionalità importanti da un punto di vista analitico. Tra queste ci sono:

- Il calcolo di funzioni di aggregazione delle misure rispetto a una o più dimensioni
- L'esecuzione di operazioni di confronto, essenziali per comparare l'andamento dei fatti considerati
- La presentazione efficace dei risultati usando modalità di visualizzazione alternative
- L'esplorazione dei dati secondo l'organizzazione gerarchica delle dimensioni.

Slide 6 - Operazioni OLAP

Più in dettaglio, le principali operazioni OLAP sono 6:

- Roll up
- Drill down
- Slice and dice
- Pivoting
- Sorting
- Drill across

Tra queste, l'operazione di **Sorting**, cioè di **ordinamento**, non richiede ulteriori chiarimenti, mentre è opportuno approfondire le prime 5 operazioni.

Tutte le operazioni prendono in input uno o più Data mart e restituiscono in output un nuovo Data mart.

Slide 7 – Roll up

Iniziamo con l'operazione di Roll up.

Questa operazione permette di ridurre il livello di aggregazione in input secondo due modalità:

- per prima cosa, è possibile ridurre il livello di dettaglio di una o più dimensioni, tra quelle considerate nell'aggregazione di partenza, navigando le gerarchie corrispondenti. Per esempio, si può passare da una aggregazione rispetto a negozio e mese a una aggregazione rispetto a città — attributo dimensionale meno dettagliato rispetto a negozio — e mese;
- una seconda possibilità consiste nell'eliminare alcune dimensioni dall'aggregazione, ad esempio, passando dall'aggregazione rispetto a prodotto e città all'aggregazione rispetto al solo prodotto.

In entrambi i casi, come schematizzato in figura, si riduce il numero di aggregati da calcolare e ciascun aggregato verrà calcolato su insiemi più grandi di fatti.

Slide 8 – Drill down

Viceversa, l'operazione di Drill down permette di aumentare il livello di aggregazione dei fatti agendo in due direzioni:

- per prima cosa, si può aumentare il livello di dettaglio di una o più dimensioni, tra quelle considerate nell'aggregazione di partenza, navigando le gerarchie corrispondenti. Per esempio, si può passare da una aggregazione rispetto a città e mese a una aggregazione rispetto a negozio — attributo dimensionale più dettagliato rispetto a città — e mese;

- una seconda possibilità consiste nell'aggiungere alcune dimensioni all'aggregazione, ad esempio passando dall'aggregazione rispetto a città all'aggregazione rispetto a città e prodotto.

In entrambi i casi, come schematizzato in figura, si aumenta il numero di aggregati da calcolare e ciascun aggregato verrà calcolato su insiemi più piccoli di fatti.

Slide 9 – Slice and dice

Le operazioni di Roll up e Drill down cambiano il livello di aggregazione, ma calcolano comunque gli aggregati rispetto a tutti i fatti disponibili.

Al contrario, l'operazione di Slice and dice permette di selezionare un sottoinsieme dei fatti da aggregare.

Questo può avvenire in due modi:

- si possono selezionare i fatti che soddisfano un certo predicato ottenendo una 'slice' cioè una fettina del cubo di partenza. Per esempio, si potrebbero considerare solo le vendite effettuate nel 2021;
- oppure si possono selezionare i fatti che soddisfano una certa combinazione di predicati ottenendo un 'dice', cioè un morso del cubo di partenza. Per esempio, si potrebbero considerare solo le vendite effettuate nel 2021 a Genova.

Slide 10 – Pivoting

Introduciamo adesso l'operazione di Pivot.

In questo caso, l'insieme dei fatti in input e il relativo livello di aggregazione non vengono modificati.

Al contrario, il Data Mart viene riorganizzato per migliorare la visualizzazione dei dati.

L'operazione di Pivot ha quindi un impatto solo sulla presentazione dei risultati.

Slide 11 – Drill across

Infine, l'operazione di Drill across prende in input due Data Mart e ne combina il contenuto generando un nuovo Data Mart come risultato.

Slide 12 – Estensioni OLAP di SQL

Poiché i Data Mart vengono spesso implementati nei sistemi relazionali, il linguaggio standard per la definizione e la manipolazione dei dati relazionali SQL è stato esteso con operatori OLAP.

Queste estensioni riguardano sia gli operatori di raggruppamento sia gli operatori di aggregazione.

Vediamoli insieme.

Slide 13 – Nuovi operatori di raggruppamento

Partiamo dai nuovi operatori di raggruppamento.

Il raggruppamento di righe in SQL viene specificato tramite la clausola GROUP BY.

Questa clausola permette di definire gruppi di righe che condividono gli stessi valori per una lista di colonne.

Gli operatori OLAP di raggruppamento, invece, permettono di definire gruppi di righe rispetto a più di una lista di colonne e corrispondono quindi all'esecuzione simultanea di molteplici clausole GROUP BY. Questi operatori vengono implementati in modo molto efficiente. Infatti, i risultati aggregati ottenuti per una certa lista di attributi vengono riutilizzati, se necessario, per calcolare aggregati più generali.

Esistono tre operatori di raggruppamento OLAP:

- **GROUP BY ROLLUP** per calcolare aggregati rispetto ai valori di insiemi specifici di colonne ottenute rimuovendo una colonna alla volta da un insieme dato;
- **GROUP BY CUBE** per calcolare aggregati rispetto a tutte le combinazioni di un insieme di colonne specificato;
- **GROUP BY GROUPING SETS** per calcolare aggregati rispetto a una lista specificata di insiemi di colonne.

Vediamo un esempio per ciascun operatore.

Slide 14 – GROUP BY ROLLUP

L'esempio si riferisce al Data Mart per l'analisi delle vendite introdotto nelle lezioni precedenti. In questo primo esempio, si vuole calcolare il ricavo delle vendite nell'anno 2000 per quattro combinazioni di attributi:

- città, mese e codice prodotto
- città e mese
- solo città
- totale generale

In questo esempio, ogni lista di attributi si ottiene dalla precedente eliminando l'attributo dimensionale a destra.

GROUP BY ROLLUP con lista di attributi pari a (città, mese e codice prodotto) realizza in modo efficiente il raggruppamento rispetto a tutti e quattro gli insiemi di attributi e per ciascun raggruppamento calcola la funzione aggregata, cioè la somma dei ricavi.

Notiamo che il risultato contiene un valore nullo nelle colonne di alcune righe. Questo avviene quando la riga si riferisce all'aggregazione di un gruppo per cui il valore della colonna non è definito.

Slide 15 – GROUP BY CUBE

Quando invece si vuole calcolare il ricavo delle vendite nell'anno 2000, per tutti i possibili sottoinsiemi di città, mese e codice prodotto, si può utilizzare l'operatore GROUP BY CUBE passandogli la lista degli attributi di interesse. Come si può vedere dall'esempio la query SQL differisce dalla precedente solo per l'operatore di raggruppamento utilizzato.

Slide 16 – GROUP BY GROUPING SETS

Infine, supponiamo di essere interessati a calcolare il ricavo delle vendite nell'anno 2000 rispetto a insiemi di colonne specifiche. Per esempio, vogliamo calcolare i ricavi rispetto ai mesi e rispetto alle combinazioni di

(città, mese e codice prodotto). In questo caso, si può utilizzare l'operatore GROUP BY GROUPING SETS, in modo analogo a quanto visto negli esempi precedenti.

Slide 17 – Nuovi meccanismi di aggregazione

Oltre agli operatori OLAP di raggruppamento, SQL offre anche meccanismi alternativi per il calcolo degli aggregati.

Il primo meccanismo permette di specificare finestre di calcolo (**window**) per definire, in modo flessibile, l'insieme delle righe sulle quali calcolare una certa funzione aggregata.

La definizione delle finestre si basa su tre concetti fondamentali: **partizionamento, ordinamento e framing**.

Il secondo meccanismo corrisponde all'introduzione in SQL di nuove funzioni di aggregazione, rispetto a quelle usuali, che rendono ancora più flessibile l'uso delle finestre.

Analizziamo adesso come è possibile definire una finestra.

Slide 18 – Partizionamento

Il partizionamento divide le righe di una tabella in gruppi, senza collassare ciascun gruppo in una singola riga del risultato, come avviene invece per la clausola GROUP BY.

Su ogni partizione si può calcolare una funzione di aggregazione.

In output, il risultato dell'aggregazione su una partizione viene associato a ciascuna riga della partizione.

Nell'esempio in figura, i dati vengono partizionati rispetto alla città. Per ogni città viene quindi calcolato il ricavo totale di vendita, che viene poi restituito in output, come colonna di ciascuna riga di risultato.

Slide 19 – Ordinamento

Il secondo concetto fondamentale per la definizione delle finestre è quello di ordinamento.

Infatti, le righe all'interno di una partizione possono essere ordinate.

Questo è utile per utilizzare particolari funzioni di aggregazione OLAP che dipendono dall'ordinamento delle righe in input.

Tra le funzioni di aggregazione introdotte in SQL per OLAP, distinguiamo due insiemi di funzioni:

- funzioni che generano, come già discusso in precedenza, **un solo valore aggregato**, uguale per tutte le righe della partizione
- funzioni che generano **valori aggregati diversi** per ciascuna riga della partizione.

Slide 20 – Ordinamento (segue)

Tra le funzioni che generano **un solo valore aggregato**, uguale per tutte le righe della partizione, e dipendono dall'ordinamento delle righe nelle partizioni, ricordiamo la funzione che restituisce il primo o l'ultimo valore di una colonna, rispetto all'ordinamento considerato.

Nell'esempio in figura, per ogni partizione, quindi per ogni città, le righe vengono ordinate rispetto al mese di vendita (anche se sono già ordinate) e viene restituito il ricavo corrispondente al primo mese nell'ordinamento, quindi febbraio 2022.

Slide 21 – Ordinamento (segue)

Invece, tra le funzioni che generano **valori aggregati diversi** per ciascuna riga della partizione e dipendono sempre dall'ordinamento, ricordiamo le funzioni che restituiscono la posizione ordinale di ciascuna riga della partizione rispetto all'ordinamento specificato, secondo diverse definizioni.

Nell'esempio in figura, la finestra è stata definita senza la clausola di partizionamento, quindi viene considerata una singola partizione che contiene tutte le righe. Questa partizione viene poi ordinata rispetto al mese di vendita, infine vengono calcolate le funzioni aggregate ROW_NUMBER(), RANK() e RANK_DENSE(), che restituiscono diversi numeri ordinali per ciascuna riga nella partizione ordinata.

Slide 22 - Framing

L'ultima clausola che può essere utilizzata per definire le finestre è quella di **framing** che permette di associare a ciascuna riga di una partizione un valore aggregato, calcolato su un sottoinsieme delle righe della partizione.

Per ogni riga, l'insieme delle righe a cui applicare la funzione di aggregazione viene definito attraverso una **finestra mobile**, chiamata **frame**.

I frame possono essere definiti secondo diverse modalità e sono utili per il calcolo di aggregati mobili.

Nell'esempio in figura, le righe vengono partizionate rispetto alla città; quindi le righe di ogni partizione vengono ordinate rispetto al mese. Viene poi definita una finestra mobile che, per ogni riga in una partizione, calcola un aggregato considerando il ricavo di vendita associato alla riga in oggetto e a tutte quelle precedenti nella partizione, secondo l'ordinamento specificato.

Slide 23 – Clausola di definizione di window

Oltre alla sintassi introdotta negli esempi precedenti, in SQL è anche possibile attribuire un nome alle finestre e richiamarlo nella clausola SELECT per la specifica del calcolo aggregato.

In presenza di window, l'ordine di esecuzione delle clausole SQL non cambia: prima la clausola FROM, poi WHERE, quindi GROUP BY e HAVING; successivamente SELECT, incluso il calcolo delle finestre, e per ultimo ORDER BY.

Slide 24 – Riepilogo e conclusioni finali

Bene, siamo giunti alla fine di questa video lezione.

Ti ricordo che abbiamo introdotto gli strumenti di front-end per l'analisi dei dati e descritto le principali operazioni OLAP.

In particolare, abbiamo visto:

- Gli strumenti che possono essere utilizzati per analizzare i dati memorizzati in un Data Warehouse

- Le principali operazioni OLAP utili a fini analitici
- Le estensioni dello standard SQL per specificare operazioni OLAP su dati relazionali

Grazie per l'attenzione!