Sistemi di gestione di basi di dati

22 febbraio 2011

1. (6 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

   AUTORE(CodA, NomeA, CittàA, DataNascita)
   CASA_EDITRICE(CodCE, NomeCE, Indirizzo, CittàCE)
   LIBRO(ISBN, Titolo, CodA, CodCE, Tipo, Prezzo)
   VENDITE(ISBN, Data, NumeroCopieVendute)

   Si ipotizzino le seguenti cardinalità per le tabelle:

   - card(AUTORE)= 10^5 tuple,
     MIN(DataNascita) = 1-1-1949, MAX(DataNascita) = 31-12-1978,
     numero di CittàA $\simeq 10$,
   - card(CASA_EDITRICE)= 10^3 tuple,
     numero di CittàCE $\simeq 10$,
   - card(LIBRO)= 10^7 tuple,
     MIN(Prezzo) = 5, MAX(Prezzo) = 44,
     numero di Tipo $\simeq 20$,
   - card(VENDITE)= 10^9 tuple,
     MIN(Data) = 01-01-2010, MAX(Data) = 31-12-2010,

   Inoltre si ipotizzi il seguente fattore di riduzione per la condizione di group by:

   - having sum(NumeroCopieVendute)$\geq$1.000 $\simeq \frac{1}{10}$.

   Si consideri la seguente query SQL:

```
select L1.ISBN, L1.Titolo, A.NomeA
from LIBRO L1, AUTORE A
where L1.CodA=A.CodA and A.CittàA='Milano'
      and L1.Tipo<>'Romanzo storico'
      and L1.ISBN NOT IN (select V.ISBN
                          from VENDITE V, LIBRO L
                          where V.ISBN=L.ISBN and L.Prezzo > 40
                          and V.Data ≥ 01/04/2010 and V.Data ≤ 30/04/2010
                          group by V.ISBN
                          having sum(NumeroCopieVendute)≥1.000)
```

   Per l'interrogazione SQL

   (a) Si scriva l'espressione algebrica corrispondente, indicando le operazioni svolte, la cardinalità e la selettività di ogni operazione. Dove necessario, si ipotizzi la distribuzione dei dati. Discutere la possibilità di anticipare l'operatore GROUP BY.

   (b) Si scelgano le strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il piano di esecuzione (ordine e tipo dei join, accesso alle tabelle e/o indici, etc.).

2. (7 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali hanno l'asterisco):

   GIURATO(CodGiurato, NomeG, Città)
   VALUTAZIONE_GIURATO(CodGiurato, CodCanzone, Punteggio)
   PUNTEGGIO_COMPLESSIVO_PER_CANZONE(CodCanzone, NumeroGiurati,PunteggioComplessivo, CanzoneVincitrice)

   Si scrivano i trigger per gestire le seguenti attività nell'ambito di una gara canora. Durante la gara sono presentate 14 canzoni. Le canzoni vengono valutate da una giuria, che elegge la canzone vincitrice.

   (1) *Vincolo di integrità sulla composizione della giuria.* La tabella `GIURATO` contiene la composizione della giuria. La giuria deve includere al massimo 300 giurati. Tutte le operazioni di modifica della tabella `GIURATO` che causano la violazione del vincolo non devono essere eseguite. Si scriva il trigger per la gestione del vincolo di integrità.

   (2) *Selezione della canzone vincitrice.* Ciascun giurato assegna un punteggio a ogni canzone. La tabella `PUNTEGGIO_COMPLESSIVO_PER_CANZONE` contiene, per ogni canzone, il numero di giurati che hanno assegnato un punteggio alla canzone (attributo `NumeroGiurati`) e il punteggio complessivo assegnato da tali giurati (attributo `PunteggioComplessivo`). Il processo di valutazione è terminato quando, per tutte le 14 canzoni, tutti i giurati hanno assegnato il loro punteggio. A questo punto si seleziona la canzone vincitrice, ossia la canzone che ha ottenuto il punteggio complessivo più alto.

   Si scriva il trigger per propagare le modifiche alla tabella `PUNTEGGIO_COMPLESSIVO_PER_CANZONE` quando viene assegnato il punteggio ad una canzone da parte di un giurato (inserimento di un nuovo record nella tabella `VALUTAZIONE_GIURATO`). Il trigger deve inoltre verificare se il processo di valutazione è terminato, e in questo caso, per la canzone vincitrice, impostare l'attributo `CanzoneVincitrice`, a 'Sì'. Si supponga che non ci siano mai due canzoni che hanno ottenuto lo stesso punteggio complessivo.

3. Progettazione Data Warehouse

Una società di analisi di mercato è interessata ad analizzare le attività degli utenti dei siti di social networking.

La società vuole valutare alcune statistiche sugli utenti iscritti (nuovi e totali), negli ultimi anni in diversi social networks e al variare delle caratteristiche degli utenti stessi. Inoltre si vuole analizzare il numero medio di contatti (amici o followers) per utente.

La società ha avuto accesso alle basi di dati dei social networks più diffusi. Tali basi di dati devono essere integrati in un data warehouse che permetta di svolgere le seguenti analisi in modo efficiente.

Deve essere possibile analizzare il numero di utenti totali, il numero di nuovi utenti iscritti e il numero medio di contatti (amici o followers) per utente, in funzione di:

- giorno dell'anno (da 1 a 366), giorno della settimana (lunedì-domenica), giorno del mese (1-31), giorni feriali o festivi, settimana dell'anno (1-52),
- data, mese, bimestre, trimestre, anno,
- sito di social networking, anno di fondazione del sito, stato (nazione) in cui è stato fondato il sito,
- anno di nascita, sesso, nazione di origine degli utenti.

Nel data warehouse saranno contenuti i dati relativi agli anni 2006-2010. Sono inoltre note le seguenti statistiche (le informazioni ritenute necessarie ma non presenti in questa lista possono essere ipotizzate e stimate dal candidato):

- sono presi in considerazione 20 siti di social networking fondati dal 2004 al 2006 in 10 stati diversi;
- l'età degli utenti varia tra 15 anni e 65 anni;
- gli utenti appartengono a circa 200 stati diversi;
- non sono prese in considerazione eventuali disiscrizioni da parte degli utenti, quindi un utente iscritto può smettere di usare il sito ma non può cancellare la propria iscrizione (il numero di utenti iscritti totali è monotono crescente).

Sono riportate di seguito alcune delle interrogazioni frequenti di interesse per la società:

(a) Separatamente per ogni social network, calcolare il totale mensile di nuovi utenti iscritti e la percentuale mensile di crescita del social network. La percentuale mensile di crescita è definita come il rapporto tra i nuovi iscritti nel mese considerato e il totale degli utenti nel mese considerato.

(b) Per ogni social network e per ogni mese, calcolare quanti nuovi utenti si sono iscritti in media al giorno, separatamente per maschi e femmine. per ogni data, e per ogni anno di nascita degli utenti.

(c) Calcolare il numero totale di nuovi utenti iscritti nei diversi giorni della settimana e la percentuale di nuovi utenti iscritti in ciascun giorno della settimana rispetto al totale dei nuovi iscritti, separatamente per maschi e femmine. Assegnare un rank alle coppie (sesso, giorno della settimana) per numero di nuovi utenti decrescente.

Progettazione

(a) (7 Punti) Progettare il data warehouse in modo da soddisfare le richieste descritte nelle specifiche del problema. Il data warehouse progettato deve inoltre permettere di rispondere in modo efficiente a tutte le interrogazioni frequenti indicate.

(b) (4 Punti) Esprimere l'interrogazione frequente (b) utilizzando il linguaggio SQL esteso.

(c) (*Opzionale:* 5 Punti) Esprimere l'interrogazione frequente (a) utilizzando il linguaggio SQL esteso.