

① Algebra e calcolo

DEPUTATI (Codice, Cognome, Nome, Commissione, Provincia, Collegio)

COLLEGI (Provincia, Numero, Nome)

PROVINCE (Sigla, Nome, Regione)

REGIONI (Codice, Nome)

COMMISSIONI (Numero, Nome, Presidente)

② Formulare in algebra relazionale, in calcolo dei domini e in calcolo delle tuple l'interrogazione che elenchi nome e cognome dei deputati che sono stati eletti in Lombardia e che siano anche presidenti di commissione.

Soluzione

$\Pi_{\text{Nome, Cognome}} (\Pi_{\text{Presidente}}(\text{Commissioni}) \triangleright \triangleleft_{\text{Presidente=Codice}}$

$\Pi_{\text{Nome, Cognome, Codice}} (\Pi_{\text{Nome, Cognome, Codice, Provincia}} (\text{Deputati})$

$\triangleright \triangleleft_{\text{Provincia=Sigla}} \Pi_{\text{Sigla}} (\Pi_{\text{Sigla, Regione}} (\text{Province}))$

$\triangleright \triangleleft_{\text{Regione=Codice}} \Pi_{\text{Codice}} (\sigma_{\text{Nome='Lombardia'}}(\text{REGIONI}))$

$\{\text{Cognome:c, Nome:n} \mid \text{Deputati}(\text{Codice:cd, Cognome:c, Nome:n, Provincia:p,...}) \wedge \text{Commissioni}(\text{Presidente:cd, ..}) \wedge \text{Province}(\text{Sigla:p, Regione:r,...}) \wedge \text{Regioni}(\text{Codice:r, Nome:nr}) \wedge \text{nr='Lombardia'}\}$

③ $\{i.(\text{Cognome, Nome}) \mid i(\text{Deputati}), p(\text{Province}), c(\text{Commissioni})$
 $r(\text{Regioni}) \mid (i.\text{Provincia}=p.\text{Sigla}) \wedge (c.\text{Presidente}=i.\text{Codice})$
 $\wedge (p.\text{Regione}=r.\text{Codice}) \wedge (r.\text{Nome}=\text{'Lombardia'})\}$

4

- Si consideri uno schema relazionale contenente le relazioni $R1(ABC)$, $R3(EF)$. Formulare in calcolo relazionale sui domini e sulle tuple l'interrogazione realizzata in algebra relazionale dalla seguente espressione:

$$(\rho_{DG \leftarrow AC}(\pi_{ACEF}(R1 \triangleright \triangleleft_{B=F} R3)))$$

8

8

1

Soluzione

- $\{D:d, G:g, E:e, F:f \mid R1(A:d, B:b, C:g) \wedge R3(E:e, F:b) \}$
- $\{i.(D,G), s.(E,F) \mid i(R1), g(R1), s(R3) \mid g.B=s.F \wedge i.G=g.C \wedge i.D=g.A\}$

6

• Si consideri il seguente schema di base di dati
Aeroporto (Città, Nazione)

Volo (IdVolo, TipoAereo, GiornoSettimana,
CittàPartenza, OraPartenza, CittàArrivo,
OraArrivo)

Aereo (TipoAereo, NumPasseggeri, QuantMerci)

Scrivere un'espressione in algebra relazionale
che elenchi tutte le città con cui è collegata
direttamente Pisa sia come città di arrivo che
come città di partenza.

13

13

Soluzione

$$\Pi_{\text{Città}} \left(\sigma_{\text{CittàArrivo}=\text{"Pisa"}} \left(\rho_{\text{Città} \leftarrow \text{CittàPartenza}} (\text{Volo}) \right) \right) \cup$$

$$\Pi_{\text{Città}} \left(\sigma_{\text{CittàPartenza}=\text{"Pisa"}} \left(\rho_{\text{Città} \leftarrow \text{CittàArrivo}} (\text{Volo}) \right) \right)$$

Definire la query precedente
anche nel calcolo relazionale sulle tuple.

14

14

2

- $\{i.(C) \mid i(\text{Volo}) \mid (i.CA = \text{'Pisa'} \wedge i.C = i.CP) \vee (i.CP = \text{'Pisa'} \wedge i.C = i.CA)\}$

9

- Dato il seguente schema di base di dati
 - AEROPORTO(Città, Nazione, NumPiste)
 - VOLO(IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)
 - AEREO(TipoAereo, NumPasseggeri, QtaMerci)

Scrivere in algebra relazionale una interrogazione che permette di determinare gli aeroporti italiani da cui partono solo voli interni.

21

21

Soluzione

$$\pi_{CittàP} (\pi_{Città} (\sigma_{Nazione='Italia'} (Aeroporto))) \\ \triangleright \triangleleft_{CittàP=Città} \pi_{CittàP} (Volo) -$$

$$\pi_{CittàP} ((\pi_{Città} (\sigma_{Nazione=Italia} (Aeroporto))) \\ \triangleright \triangleleft_{CittàP=Città} \pi_{CittàP, CittàA} (Volo)) \\ \triangleright \triangleleft_{CittàA=CittàK} \rho_{CittàK \leftarrow Città} \\ (\pi_{Città} (\sigma_{Nazione \neq 'Italia'} (Aeroporto))))$$

22

22

- h
- $\{ CP:cp \mid \text{Aeroporto}(C:cp, \text{Nazione}:n,..) \wedge n='Italia' \wedge \neg \exists cv', ca'.. \text{Volo}(CV:cv', CP:cp, CA:ca',..) \wedge \text{Aeroporto}(C:ca', \text{Nazione}:n',..) n' \neq 'Italia' \}$

23

23

Esercizio 8

Considerare la seguente base di dati relazionale:

CLIENTI (Codice, Nome, Indirizzo, Città)

NOLEGGI (Cliente, Auto, DataPrelievo, DataRestituzione)

AUTOVETTURE (Targa, Modello, Colore, AnnoImmatricolazione, CostoGiornaliero)

con vincolo di integrità referenziale fra l'attributo Auto e la chiave della relazione AUTOVETTURE e fra l'attributo Cliente e la chiave della relazione CLIENTI; formulare in algebra relazionale:

1. l'interrogazione che restituisce i clienti che hanno noleggiato più di un'autovettura
2. l'interrogazione che restituisce i clienti che hanno noleggiato autovetture di un solo modello

24

Soluzione:

1. $\pi_{\text{Cliente}} (\pi_{\text{Cliente}, \text{DataP}}(\text{NOLEGGI}))$

$\triangleright \triangleleft_{(\text{Cliente}=\text{Cliente}') \wedge (\text{DataP} \neq \text{DataP}')}$

$\rho_{X' \leftarrow X} (\pi_{\text{Cliente}, \text{DataP}}(\text{NOLEGGI}))$

2. $V = \text{NOLEGGI} \triangleright \triangleleft_{\text{Auto}=\text{Targa}} \text{AUTOVETTURE}$

$V1 = V \triangleright \triangleleft_{(\text{Cliente}=\text{Cliente}') \wedge (\text{Modello} \neq \text{Modello}')} (\rho_{X' \leftarrow X} V)$

$\pi_{\text{Cliente}} \text{NOLEGGI} - \pi_{\text{Cliente}}(V1)$

Esercizio 11

- Si consideri una base di dati sulle relazioni:
 $R1(A,B,\underline{C})$ e $R2(D,\underline{E}, F)$.
- Dire se le seguenti interrogazioni sono equivalenti

33

1. $\pi_{AB}(R1) - \pi_{AB}(R1 \bowtie_{C=D} R2)$ *Esclude completamente*

2. $\pi_{AB}(\sigma_{D \text{ IS NULL}}(R1 \bowtie_{C=D} R2))$

34

34

Uso del join esterno

Data la base di dati precedente, trovare la matricola dei capi degli impiegati che guadagnano tutti più di 40000 euro

$\pi_{\text{Capo}}(\sigma_{\text{Matricola}=\text{Null}}(\text{Supervisione} = \triangleright \triangleleft \text{Impiegato}=\text{Matricola}$
 $\sigma_{\text{Stipendio} \leq 40000}(\text{Impiegati}))$
???

8

Oppure si usa il -

```

 $\pi_C (\sigma_{\text{Capo}=\text{Null}}$ 
   $(\rho_C \leftarrow \text{Capo}(\pi_{\text{Capo}}(\text{Supervisione})))$ 
   $= \triangleright \triangleleft_{C=\text{Capo}}$ 
   $\pi_{\text{Capo}}(\text{Supervisione} \triangleright \triangleleft_{\text{Impiegato}=\text{Matricola}}$ 
     $\sigma_{\text{Stipendio} \leq 40000}(\text{Impiegati}))$ 
   $)$ 
 $)$ 

```

44

9

Esercizio 12

- Si consideri la seguente basi di dati
 - Skipper(codiceV, nome, cognome, codiceB)
 - Barca (codiceB, TipoB, porto)
- Scrivere una espressione in algebra relazionale per elencare i codici dei velisti che hanno fatto da skipper su ogni barca a vela ormeggiata a Lerici.

45

45

9

Tre possibili soluzioni

Soluzione

$$\text{Velista} = \Pi_{\text{codiceV}, \text{codiceB}} (\sigma_{\text{TipoB}='Vela'} (\text{Barca}) \triangleright \triangleleft \text{Skipper})$$

$$\text{BarcaLerici} = \Pi_{\text{codiceB}} (\sigma_{\text{porto}='Lerici' \wedge \text{TipoB}='Vela'} (\text{Barca}))$$

$$\Pi_{\text{codiceV}} (\text{Velista}) - \Pi_{\text{codiceV}} (\Pi_{\text{codiceV}} (\text{Velista}) \triangleright \triangleleft \Pi_{\text{codiceB}} (\text{BarcaLerici}) \div \text{Velista})$$

$$\text{Velista} \div \text{BarcaLerici}$$

① →

② →

$$\text{Velisti} - \Pi_{\text{codiceV} \neq \text{NULL}} \left(\text{Velisti} \times \text{BarcaLerici} \right) \div \text{Velisti}$$

③ →

$$\text{Velisti} - \left(\text{Velisti} \times \text{BarcaLerici} \right) \div \text{Velisti}$$

2) Calcolo Volumi

Si consideri lo schema concettuale in figura nel quale l'attributo Saldo di una occorrenza di CONTOCORRENTE è ottenuto come somma dei valori dell'attributo Importo per le occorrenze di OPERAZIONE ad essa correlate tramite la relationship MOVIMENTO.



Valutare se convenga o meno mantenere la ridondanza, tenendo conto che i volumi delle due entità sono $V_{CC} = 2.000$ e $V_{OP} = 20.000$, $V_{MOV} ?$ e che le operazioni più importanti sono:

Op₁: effettuazione di un'operazione di movimento su un conto corrente esistente, con frequenza giornaliera $f_1 = 10$

Op₂: lettura del saldo di un conto corrente con frequenza giornaliera $f_2 = 1000$.

Soluzione

Op₁ senza Saldo richiede 40 operazioni elementari al giorno

Op₁ con Saldo richiede 70 operazioni elementari al giorno

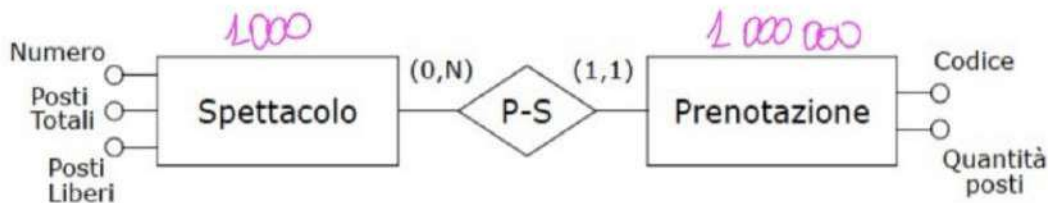
Op₂ senza Saldo richiede 20000 operazioni elementari al giorno

Op₂ con Saldo richiede 1000 operazioni elementari al giorno

Handwritten notes:
 10 (4) Scrittura su Operazione e su Movimento
 Lettura e Scrittura del Saldo
 40 + 10 (2+2)
 ~10 letture di Movimento e
 ~10 letture di importo
 1000 (1) lettura del Saldo
 1000 (20+10)

Esercizio 11

Lo schema concettuale seguente rappresenta un insieme di spettacoli e un insieme di prenotazioni ognuna delle quali fa riferimento (attraverso la relazione P-S) ad uno spettacolo. In particolare, l'attributo PostiLiberi di una occorrenza di Spettacolo è pari alla differenza fra il valore di PostiTotali per lo stesso Spettacolo e la somma del numero di posti prenotati (ottenibili dalla somma dei valori dell'attributo Quantità Posti delle occorrenze dell'entità Prenotazione cui l'occorrenza di Spettacolo è correlata tramite P-S).



Valutare se convenga o meno mantenere la ridondanza PostiLiberi, tenendo conto del fatto che i volumi delle due entità sono $V_{sp} = 1.000$ e $V_{pre} = 1.000.000$ ($V_{P-S} ?$) e che le operazioni più importanti sono:

Op₁ lettura del numero di posti disponibili per uno spettacolo, con frequenza giornaliera $f_1 = 100$

Op₂ inserimento di una prenotazione per un certo spettacolo, con frequenza giornaliera $f_2 = 10.000$

Soluzione

Op₁ senza PostiLiberi richiede 200000 operazioni elementari al giorno

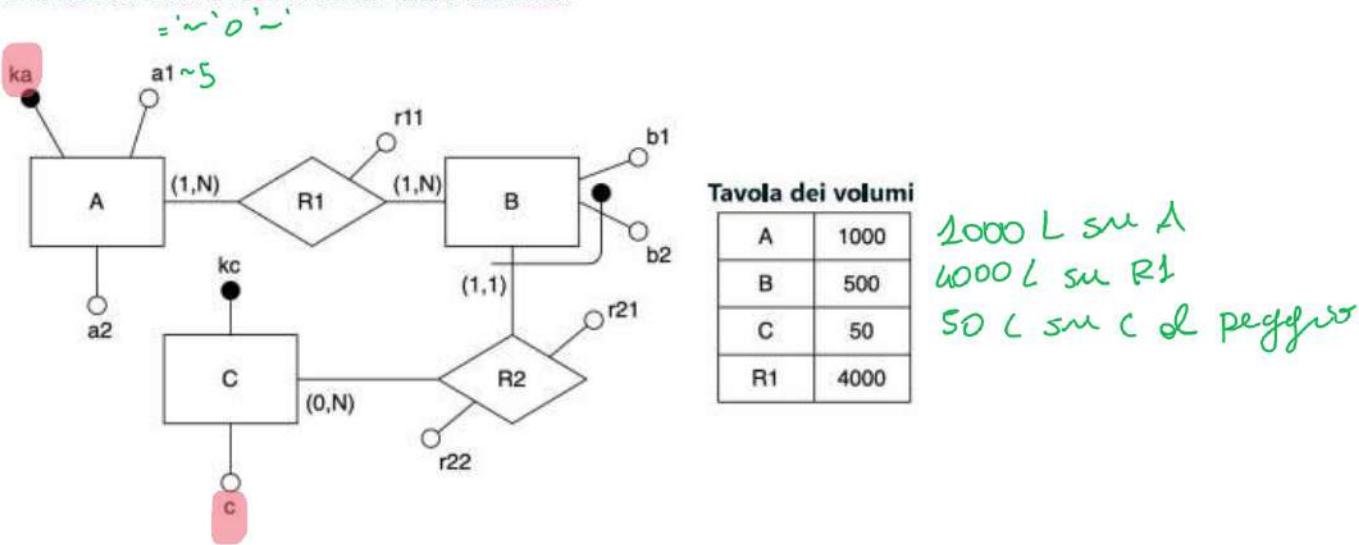
Op₁ con PostiLiberi richiede 100 operazioni elementari al giorno

Op₂ senza PostiLiberi richiede 40000 operazioni elementari al giorno

Op₂ con PostiLiberi richiede 70000 operazioni elementari al giorno

Handwritten notes:
 100 (1000 + 1000) Lettura su P-S e Prenotazione
 100 (1) Letture posti
 1000 (2+2) Scrittura P-S e P
 40000 + 10000 (1+2) Lettura e scrittura posti

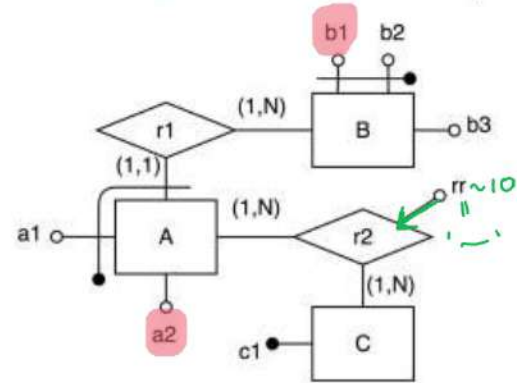
Considerare il diagramma E-R e la tabella dei volumi sottostanti. Supponendo che i valori di $a1$ siano 5 ed equi-probabili, indicare il numero di accessi necessari, nel caso peggiore, per restituire, per ciascuna occorrenza di A tale che $a1='xxx'$ o $a1='yyy'$, il codice ka dell'occorrenza, e il numero di valori diversi di c a cui essa è associata.



5050

☐ a. nessuna alternativa è corretta

Dati il diagramma E-R e la tabella dei volumi sottostanti, quanti sono, nel caso peggiore, gli accessi per determinare i valori di a_2 e b_1 relativi a occorrenze di C associate a un valore di rr pari a 'xxx', qualora i valori diversi di rr siano 10, ed equi-probabili?



| Concetto | Volume |
|----------|--------|
| A | 300 |
| B | 10 |
| C | 100 |
| r2 | 400 |

400 L su R2 per 40
con $rr = 'x'$
40 L su A per
ottenere B1 e A2

- ☐ a. non si può sapere, date le informazioni a disposizione
- ☒ b. 440

③ Dipendenze

Esercizio 7

Si considerino lo schema di relazione $R(A,B,C,D,E,F)$ e l'insieme di dipendenze associato: $G = \{A \rightarrow B, C \rightarrow AD, AF \rightarrow EC\}$.

(1) Si determinino le chiavi di R

Poiché l'attributo F non compare nella parte destra di alcuna DF , ne segue che F deve appartenere ad ogni chiave. Al contrario, D, E , e B compaiono solo nella parte destra di alcune DF . Ne segue che D, E , e B non appartengono ad alcuna chiave. Da:

- $F^+ = F$ *→ Inizio con solo quelli a sx*

- $AF^+ = AFBDEC = R$ *Aggiungo a turno quelli*

- $CF^+ = CFADEC = R$ *che compaiono a sx anche se non sempre*

segue che AF e CF sono chiavi di R .

Esercizio 8

Considerare uno schema di relazione $R(E, N, L, C, S, D, M, A)$, con le dipendenze $F = \{E \rightarrow NS, N \rightarrow MD, EN \rightarrow LCD, C \rightarrow S, D \rightarrow M, M \rightarrow D, ED \rightarrow A, NLC \rightarrow A\}$.

Calcolare una copertura ridotta F' per tale insieme e determinare le possibili chiavi.

Soluzione

I passi per calcolare la copertura ridotta di una relazione sono i seguenti:

1. sostituzione l'insieme di dipendenze funzionali con un insieme equivalente che ha i secondi membri costituiti da un singolo attributo;
2. per ogni dipendenza verifica dell'esistenza di attributi eliminabili dal primo membro.
3. eliminazione delle dipendenze ridondanti

Il primo passo porta all'individuazione delle seguenti dipendenze funzionali:

1. $E \rightarrow N$
2. $E \rightarrow S$
4. $N \rightarrow M$
5. $N \rightarrow D$
6. $EN \rightarrow L$
7. $EN \rightarrow C$
8. $EN \rightarrow D$
9. $C \rightarrow S$
10. $D \rightarrow M$
11. $M \rightarrow D$
12. $ED \rightarrow A$
13. $NLC \rightarrow A$

Il secondo passo porta alle seguenti dipendenze funzionali:

1. $E \rightarrow N$
2. $E \rightarrow S$
3. $N \rightarrow M$
5. $N \rightarrow D$
6. $E \rightarrow L$
7. $E \rightarrow C$
8. $E \rightarrow D$
9. $C \rightarrow S$
10. $D \rightarrow M$
11. $M \rightarrow D$
12. $E \rightarrow A$
13. $NLC \rightarrow A$

Le dipendenze evidenziate sono ridondanti e si possono quindi eliminare; la chiave è E.

$$F' = \{ E \rightarrow N, L, C, N \rightarrow D, C \rightarrow S, D \rightarrow M, M \rightarrow D, NLC \rightarrow A \}$$

4

Normalizzazione

Esercitazione 5 Normalizzazione

Esercizio 4.6

Dato il seguente schema: $R(ABCDE)$, con le seguenti dipendenze funzionali: $A \rightarrow B$, $A \rightarrow C$, $D \rightarrow E$. Verificare (formalmente) se ACE è superchiave o meno della relazione R . Giustificare la risposta.

ACE NON è una superchiave, in quanto la sua chiusura $ACE^+ = \{ABCE\}$ non contiene tutti gli attributi della relazione (manca D). Chiave? $\{AD\}$

b) La relazione è in forma normale di Boyce - Codd (BCNF)? È in terza forma normale (3FN)? Giustificare le risposte.

La relazione NON è in BCNF in quanto nella dipendenza funzionale: $A \rightarrow B$, A non è una superchiave della relazione. La relazione NON è in 3FN in quanto nella dipendenza funzionale: $A \rightarrow B$, A non è una superchiave della relazione, e B non è parte della chiave (AD).

c) Nel caso la relazione non risulti in 3FN, decomporla in terza forma normale. Una possibile decomposizione è la seguente: $R_1=ABC$ $R_2=DE$ $R_3=AD$

5

LOG

Esempio di file LOG

DUMP B(T1) B(T2) I(T1,O1,A1) D(T2,O2,B2) B(T3) B(T4) U(T3,O3,B3,A3) C(T2)
CK(T1,T3,T4) U(T1,O4,B4,A4) A(T3) B(T5) D(T4,O5,B5) C(T1) C(T4) I(T5,O6,A6) GUASTO

Ripresa a caldo

Ultimo checkpoint

UNDO → T1, T3, T4

REDO → Vuota

Istante del guasto

UNDO → T3, T5

REDO → T1, T4

Operazioni da disfare (*ordine cronologico inverso*)

I(T5,O6,A6) → Delete O6

U(T3,O3,B3,A3) → Update O3 = B3

Operazioni da rifare (*ordine cronologico*)

I(T1,O1,A1) → Insert O1 := A1

U(T1,O4,B4,A4) → Update O4 = A4

D(T4,O5,B5) → Delete O5

Ripresa a freddo

Supponiamo un guasto su O1, O2, O3

DUMP → Ripristino O1, O2, O3

I(T1,O1,A1) → Insert O1 := A1

D(T2,O2,B2) → Delete O2

U(T3,O3,B3,A3) → Update O3 := A3

Checkpoint → Inizio una ripresa a caldo

Esercizio1

Descrivere la ripresa a caldo, indicando la costituzione progressiva degli insiemi di UNDO e REDO e le azioni di recovery, a fronte del seguente log:

DUMP, B(T1), B(T2), B(T3), I(T1, O1, A1), D(T2, O2, B2), B(T4), U(T4, O3, B3, A3), U(T1, O4, B4, A4), C(T2), CK(T1, T3, T4), B(T5), B(T6), U(T5, O5, B5, A5), A(T3), CK(T1, T4, T5, T6), B(T7), A(T4), U(T7, O6, B6, A6), U(T6, O3, B7, A7), B(T8), A(T7), guasto

O1, O2, O3

Soluzione:

1) Per prima cosa bisogna percorrere il log a ritroso fino al più recente record di check-point: CK(T1,T4,T5,T6)

Si costruiscono gli insiemi di UNDO e di REDO:

UNDO= { T1, T4, T5, T6 } REDO={}

2) Il log viene percorso in avanti, aggiornando i due insiemi:

B(T7) UNDO= { T1, T4, T5, T6, T7 } REDO={}

A(T4) UNDO= { T1, T4, T5, T6, T7 } REDO={}

B(T8) UNDO= { T1, T4, T5, T6, T7, T8 } REDO={}

A(T7) UNDO= { T1, T4, T5, T6, T7, T8 } REDO={}

3) Il log viene ripercorso ancora a ritroso, fino all'operazione I(T1,O1,A1), eseguendo le seguenti operazioni:

O3=B7

O6=B6

O5=B5

O4=B4

O3=B3

Delete O1

4) Il log viene ripercorso in avanti per rieseguire le operazioni di REDO, ma essendo vuoto questo insieme, nessuna operazione verrà eseguita.

Esercizio 2

Si supponga che nella situazione precedente si verifichi un guasto di dispositivo che coinvolge gli oggetti O1, O2, O3; descrivere la ripresa a freddo.

DUMP, B(T1), B(T2), B(T3), I(T1, O1, A1), D(T2, O2, B2), B(T4), U(T4, O3, B3, A3), U(T1, O4, B4, A4), C(T2), CK(T1, T3, T4), B(T5), B(T6), U(T5, O5, B5, A5), A(T3), CK(T1, T4, T5, T6), B(T7), A(T4), U(T7, O6, B6, A6), U(T6, O3, B7, A7), B(T8), A(T7), guasto

Soluzione:

La ripresa a freddo è articolata in tre fasi successive.

1. Il log viene percorso a ritroso fino al primo record DUMP e si ricopia selettivamente la parte deteriorata della base dati.

2. Si ripercorre in avanti il log, applicando relativamente alla parte deteriorata della base di dati sia le azioni sulla base di dati sia le azioni di commit o abort e riportandosi così nella situazione precedente al guasto.

Insert O₁=A₁

Delete O₂

O₃=A₃

Commit (T₂)

Abort(T₃)

Abort (T₄)

O₃=A₇

Abort(T₇)

3. Si svolge una ripresa a caldo.

6 Schedule

a. Si considerino i due seguenti schedule e si verifichi se siano view serializzabili.

1. $r1(x), w1(x), w3(x), r2(y), r3(y), w3(y), w1(y), r2(x)$

2. $r1(y), w2(z), w1(z), w3(z), w3(x), w1(x)$

Soluzione

a.1. $r1(x), w1(x), w3(x), r2(y), r3(y), w3(y), w1(y), r2(x)$

Questo schedule è NonVSR. In uno schedule seriale view-equivalente a questo schedule la transazione 1 dovrebbe seguire la transazione 3 a causa delle SCRITTURE FINALI su y, ma dovrebbe anche precedere la transazione 3 a causa della relazione LEGGE - DA su x.

a.2. $r1(y), w2(z), w1(z), w3(z), w3(x), w1(x)$

La transazione 1 ha due scritture, una su z ed un'altra su x, ma anche la transazione 3 ha due scritture, una su z ed un'altra su x. Nello schedule di partenza, le scritture finali su x e z sono originate rispettivamente dalle transazioni 1 e 3. Nessuno schedule seriale potrà esibire le stesse scritture finali. Questo schedule non è quindi VSR.

Esempio

Provare che il seguente schedule è CSR e trovarne uno seriale equivalente

$S: R_1(Y) W_3(Z) R_1(Z) R_2(Z) W_3(X) W_1(X) W_2(X) R_3(Y)$

A. Separare le operazioni in base all'oggetto su cui operano per trovare i conflitti

$X \rightarrow W_3 W_1 W_2$

$Y \rightarrow R_1 R_3$

$Z \rightarrow W_3 R_1 R_2$

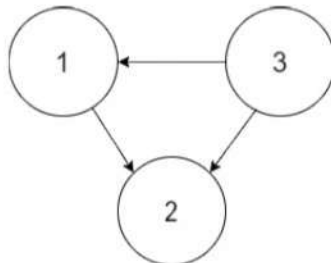
B. Individuare i conflitti e rimuovere i duplicati

Per X i conflitti sono: $3 \rightarrow 1, 3 \rightarrow 2, 1 \rightarrow 2$

Per Y non ci sono conflitti (solo letture)

Per Z i conflitti sono: $3 \rightarrow 1, 3 \rightarrow 2$ (duplicati in X)

C. Disegnare il grafo orientato dei conflitti



D. Verificare se il grafo è aciclico (*in questo caso lo è*)

E. Se richiesto trovare uno schedule seriale equivalente

$T_3 T_1 T_2$ è uno schedule seriale equivalente perché rispetta l'ordine imposto dal grafo

Esempio

Applica 2PL Strict e TS allo schedule $S: R_1(Y) W_3(Z) R_1(Z) R_2(Z) W_3(X) W_1(X) W_2(X) R_3(Y)$

2PL Strict

$R_1(Y) \rightarrow R_lock(Y)$
 $W_3(Z) \rightarrow W_lock(Z)$
 $R_1(Z) \rightarrow T_1 \text{ wait } T_3 \rightarrow \text{Queue } (T_1)$
 $R_2(Z) \rightarrow T_2 \text{ wait } T_3 \rightarrow \text{Queue } (T_2)$
 $W_3(X) \rightarrow W_lock(X)$
 $W_1(X)$ ignorata perchè T_1 in attesa
 $W_2(X)$ ignorata perchè T_2 in attesa
 $R_3(Y) \rightarrow R_lock(Y)$
 $\text{Commit } (T_3) \rightarrow \text{unlock}(Z, X, Y)$
 $\text{Dequeue } (T_1, T_2)$
 $R_1(Z) \rightarrow R_lock(Z)$
 $R_2(Z) \rightarrow R_lock(Z)$
 $W_1(X) \rightarrow W_lock(X)$
 $\text{Commit } (T_1) \rightarrow \text{unlock}(Y, Z, X)$
 $W_2(X) \rightarrow W_lock(X)$
 $\text{Commit } (T_2) \rightarrow \text{unlock}(Z, X)$

TS

$R_1(Y) \rightarrow RTM(Y) = 1$
 $W_3(Z) \rightarrow WTM(Z) = 3$
 $R_1(Z) \rightarrow \text{Abort} \rightarrow T_1 \Rightarrow T_4$
 $R_4(Y) \rightarrow RTM(Y) = 4$
 $R_4(Z) \rightarrow RTM(Z) = 4$
 $R_2(Z) \rightarrow \text{Abort} \rightarrow T_2 \Rightarrow T_5$
 $R_5(Z) \rightarrow RTM(Z) = 5$
 $W_3(X) \rightarrow WTM(X) = 3$
 $W_4(X) \rightarrow WTM(X) = 4$
 $W_5(X) \rightarrow WTM(X) = 5$
 $R_3(Y) \rightarrow RTM(Y) = 4$

equivale
ai conflitti

7 Ottimizzazione

Database

C \rightarrow Customer (Id_{cust}, Name, Age, City_{cust})

I \rightarrow Insurance (Id_{com}, Id_{director}, NumberEM, city)

P \rightarrow Policy (Id_{pol}, Id_{cust}, Id_{com}, Date)

Tabella dei volumi

| | | |
|-------------|-------------|---------------------|
| $mC = 2000$ | $mI = 20$ | $mP = 100000$ |
| | $mCity = 5$ | $mId_{cust} = 2000$ |
| | | $mId_{com} = 20$ |
| | | $mDate = 20$ |

Query da ottimizzare

$\pi_{Name}(\sigma_{City='n' \wedge Date='m'}(C \bowtie P \bowtie I))$

Ottimizzazione algebrica

$\pi_{Name}(\pi_{Name, Id_{cust}}(C) \bowtie \pi_{Id_{cust}, Id_{com}}(\sigma_{Date='m'}(P)) \bowtie \pi_{Id_{com}}(\sigma_{City='n'}(I)))$

Ottimizzazione basata sui costi

- 1 La proiezione contiene la chiave dunque gli elementi restano gli stessi (2000)
- 2 La selezione lascia in media $\frac{Record}{Date} = \frac{100000}{20} = 5000$ elementi.
La proiezione non contiene la chiave, dunque è uguale al minimo tra i record che ho e i record diversi che posso proiettare con tali attributi.
 $\min(5000, 20 \cdot 2000) = \min(5000, 40000) = 5000$
nota che non possono essere più di 5000 singolarmente!
- 3 La selezione lascia in media $\frac{Record}{City} = \frac{20}{5} = 4$ elementi.
La proiezione contiene la chiave dunque restano 4 elem.

Con questi risultati posso ora calcolare l'ordine di join

- ① \bowtie ② Il join viene fatto sulla chiave di ① e ② ha un vincolo di integrità referenziale dunque si producono esattamente ② record ossia 5000
- ① \bowtie ③ Il join non contiene attributi comuni dunque equivale a fare il prodotto cartesiano ossia $2000 \cdot 4 = 8000$ elementi
- ② \bowtie ③ Il join viene fatto sulla chiave di ③ ma non vale più il vincolo di integrità referenziale tra ② e ③ perché in entrambi ho effettuato selezioni.
 Il join produce $\min(\text{elem}(\textcircled{2}) \cdot \text{elem uguali}(\textcircled{3}), \text{elem}(\textcircled{3}) \cdot \text{elem uguali}(\textcircled{2}))$
elem uguali sull'attributo di join
 Dunque $\min(5000 \cdot \frac{4}{4}, 4 \cdot \frac{5000}{20}) = (5000, 1000) = 1000$

L'ordine di esecuzione dei join è dunque $(\textcircled{2} \bowtie \textcircled{3}) \bowtie \textcircled{1}$

Query finale

$$\pi_{\text{Nome}} \left(\pi_{\text{Idcust}, \text{Idcom}} (\sigma_{\text{note} = 'm'}(P)) \bowtie \pi_{\text{Idcom}} (\sigma_{\text{city} = 'm'}(I)) \right. \\ \left. \bowtie \pi_{\text{Nome}, \text{Idcust}}(C) \right)$$

Esercizio 10

Determinare un possibile piano di esecuzione per la seguente query effettuata sulle relazioni $T1(\underline{A}, B, C)$, $T2(\underline{D}, E)$ e $T3(\underline{F}, L)$.

```
select distinct A, L
from T1 join T2 on T1.C = T2.D join T3 on T2.E = T3.F
where T1.B = 3
```

$\Pi_{A,L} (\Pi_{A,E} (\sigma_{B=3}(T1) \bowtie \triangleleft_{C=D} T2) \bowtie \triangleleft_{E=F} T3)$ *no Query fissa*

con riferimento alle seguenti informazioni sulla base dati:

- la relazione $T1(\underline{A}BC)$ ha 800.000 tuple e 100.000 valori diversi per l'attributo B, distribuiti uniformemente
- la relazione $T2(\underline{D}E)$ ha 500.000 tuple; è definito un vincolo di integrità referenziale fra l'attributo E e la chiave F della relazione T3
- la relazione $T3(\underline{F}L)$ ha 1.000 tuple e ha una struttura hash sulla chiave F.

Soluzione

1. Selezione, tramite una scansione, dell'attributo $T1.B = 3$.

Questa soluzione ha un costo molto elevato, 800.000 accessi in memoria, ma permette di estrarre un numero molto basso di record, in quanto possiede 100.000 valori diversi e mediamente ci saranno 8 record per ogni occorrenza. $T1'$ ha 8 elementi.

2. JOIN di tipo nested loop sulle relazioni $T1'$ e T2, ponendo $T1.C = T2.D$ e usando $T1'$ come tabella interna e T2 come esterna. D è chiave in T2, quindi il join avrà un numero di elementi ≤ 8 . la proiezione avrà ancora 8 elementi al massimo.

3. JOIN usando l'accesso hash su $T3.F$ e, essendo F chiave e riferita da E, il risultato è di 8 elementi al massimo.