

Atomicità

Una transazione è una unità di esecuzione INDIVISIBILE. O viene eseguita completamente o non viene eseguita affatto.

Se una transazione viene interrotta prima del commit, il lavoro fin qui eseguito dalla transazione deve essere disfatto ripristinando la situazione in cui si trovava la base di dati prima dell'inizio della transazione.

Se una transazione viene interrotta all'esecuzione del commit (commit eseguito con successo), il sistema deve assicurare che la transazione abbia effetto sulla base di dati.

Consistenza

L'esecuzione di una transazione non deve violare i vincoli di integrità.

Al verificarsi della violazione di un vincolo il sistema può: VERIFICA IMMEDIATA:

Viene abortita l'ultima operazione e il sistema restituisce all'applicazione una segnalazione d'errore. L'applicazione può quindi reagire alla violazione.

VERIFICA DIFFERITA: Al commit se un vincolo di integrità viene violato la transazione viene abortita senza possibilità da parte dell'applicazione di reagire alla violazione.

Isolamento

L'esecuzione di una transazione deve essere INDIPENDENTE dalla contemporanea esecuzione di altre transazioni.

Il rollback di una transazione non deve creare rollback a catena di altre transazioni che si trovano in esecuzione contemporaneamente.

Il sistema deve regolare l'esecuzione concorrente.

Persistenza

L'effetto di una transazione che ha eseguito il commit non deve andare perso.

Il sistema deve essere in grado, in caso di guasto, di garantire gli effetti delle transazioni che al momento del guasto avevano già eseguito un commit.

Relazione

Rappresentano legami logici significativi tra due o più entità. In uno schema E-R, ogni relazione ha un nome che la identifica univocamente e viene rappresentata graficamente mediante un rombo, con il nome all'interno. Un'occorrenza di relazione è una ennupla costituita da occorrenze di entità, una per ciascuna delle entità coinvolte.

Attributo

Descrivono le proprietà elementari di entità o relazioni che sono di interesse ai fini dell'applicazione. Un attributo associa a ciascuna occorrenza di entità (o relazione) un valore appartenente a un insieme, detto dominio, che contiene i valori ammissibili per l'attributo. Si possono raggruppare attributi affini e questi verranno detti attributi composti.

Superchiave

Una chiave è un insieme di attributi utilizzato per identificare univocamente le tuple di una relazione. Un insieme k di attributi è SUPERCHIAVE di una relazione r se r non contiene due tuple distinte t_1 e t_2 con $t_1[k] = t_2[k]$.

Chiave candidata

Per chiave candidata intendiamo un campo singolo o composto che soddisfa i requisiti di una chiave primaria. A dispetto di una chiave primaria, vi possono esser più chiavi candidate che competono per essere chiavi primarie.

Identificatore

Specificati per ciascuna entità di uno schema e descrivono i concetti (attributi e/o entità) dello schema che permettono di identificare in maniera univoca le occorrenze dell'entità. Uno o più attributi di una entità sono sufficienti a individuare un identificatore: si parla di identificatore interno (detto chiave). Nei casi in cui l'identificatore di una entità è ottenuta utilizzando altre entità si parla di identificatore esterno.

Entità debole

Un'entità debole è tale perché la sua esistenza dipende da un'altra entità. In un'entità debole non è possibile trovare una chiave primaria, ma solo una chiave parziale. Le istanze sono parti componenti dell'entità da cui dipendono, es appartamenti di un palazzo.

Conflict-serializzabilità

Uno schedule S è conflict-serializzabile (CSR) se esiste uno schedule seriale S' tale che essi sono conflict-equivalenti, ossia possiedono le stesse operazioni e gli stessi conflitti.

View-serializzabilità

Uno schedule S è view-serializzabile (VSR) se esiste uno schedule seriale S' tale che essi sono view-equivalenti, ossia possiedono le stesse relazioni LEGGE_DA e le stesse scritture finali.

Locking a due fasi stretto

È il metodo applicato ai sistemi commerciali. Gli aspetti sono tre: il meccanismo di base per la gestione dei lock, la politica di concessione dei lock sulle risorse e la regola che garantisce la serializzabilità. Il meccanismo di base si basa su alcune primitive di lock: r_lock, w_lock e unlock K. Ogni lettura deve esser preceduta da una r_lock e seguita da una unlock K e ogni scrittura preceduta da una w_lock e seguita da una unlock. Il gestore dei LOCK mantiene per ogni risorsa x due informazioni: lo stato e le transizioni in r_lock, ossia le transizioni che bloccano in lettura la risorsa. La regola che garantisce la serializzabilità dice che una transazione dopo aver rilasciato un LOCK non può acquisirne altri. Aggiuntiva: una transazione può rilasciare i LOCK solo quando ha eseguito correttamente un COMMIT o un ROLLBACK.

Metodo di accesso calcolato(Hashing)

Si basano su una funzione di hash che mappa le chiavi sugli indirizzi di memorizzazione delle tuple nelle pagine dati della memoria secondaria; Uso:

Si stima il numero di valori chiave che saranno contenuti nella tabella;

Si alloca un numero di bucket di puntatori (B) uguale al numero stimato;

Si definisce una funzione di FOLDING che trasforma i valori chiave in numeri interi positivi $f: K \rightarrow Z^+$

Si definisce una funzione di HASHING: $h: Z^+ \rightarrow B$

Bisogna gestire inoltre le collisioni per evitare la saturazione dei BUCKET.

Indici

Per aumentare le prestazioni degli accessi alle tuple memorizzate nelle strutture fisiche (FILE SEQUENZIALE) si introducono strutture ausiliarie dette INDICI. Vi sono due tipi di indici INDICE PRIMARIO: in questo caso la chiave di ordinamento del file sequenziale coincide con la chiave di ricerca dell'indice. Coppia V, puntatore al primo record nel file sequenziale con chiave V. INDICE SECONDARIO (SOLO DENSO): in questo caso invece la chiave di ordinamento e la chiave di ricerca sono diverse. Coppia V puntatore al bucket di puntatori delle tuple con chiave V. Esistono due varianti dell'indice primario INDICE DENSO: per ogni occorrenza della chiave presente nel file esiste un corrispondente record nell'indice. INDICE SPARSO: solo per alcune occorrenze della chiave presenti nel file esiste un corrispondente record nell'indice, tipicamente una per pagina.