

HINTS

CONSEJOS CONCEPTUALES

- Una transacción siempre termina, aún en la presencia de fallas. Si ésta termina de manera exitosa, se dice que la transacción hace un commit. Mientras que si se detiene sin terminar su tarea, se dice que la transacción aborta.
- Cuando la transacción es abortada, su ejecución es detenida, y todas sus acciones ejecutadas hasta el momento son deshechas, regresando la base de datos al estado antes de su ejecución. A esta operación también se le conoce como rollback.
- Se dice que los elementos de datos que lee una transacción, constituyen el conjunto de lectura (RS). Similarmente, a los elementos de datos que una transacción escribe se les denomina conjunto de escritura (WS).
- Las transacciones pueden pertenecer a varias clases. Aun cuando los problemas fundamentales son los mismos para todas, los algoritmos y técnicas que se usan para tratarlas si pueden ser considerablemente diferentes. Éstas pueden ser agrupadas a lo largo de las siguientes dimensiones:
 - Áreas de aplicación: las transacciones se pueden ejecutar en aplicaciones no distribuidas. Aquellas que operan en datos distribuidos se les conoce como transacciones distribuidas. Por otro lado, dado que los resultados de una transacción que realiza un commit son durables, la única forma de deshacer sus efectos es mediante otra transacción. A este tipo se les conoce como transacciones compensatorias. Finalmente, en ambientes heterogéneos, se presentan transacciones heterogéneas sobre los datos.
 - Tiempo de duración: tomando en cuenta el tiempo que transcurre desde que se inicia una transacción, hasta que se realiza un commit o se aborta, éstas pueden ser de tipo batch o en línea. También se pueden diferenciar como transacciones de corta y larga vida. Las transacciones en línea se caracterizan por tiempos de respuesta muy cortos, y por acceder una porción relativamente pequeña de la base de datos. Por otro lado, las transacciones de tipo batch toman tiempos relativamente largos, y acceden grandes porciones de la base de datos.
 - Estructura: considerando la estructura que puede tener una transacción, se examinan dos aspectos: si ésta puede contener a su vez subtransacciones, o el orden de las acciones de lectura y escritura dentro de una transacción.

- Los siguientes son los aspectos más importantes relacionados con el procesamiento de transacciones:
 - Modelo de estructura de transacciones: es importante considerar si las transacciones son planas, o si pueden estar anidadas.
 - Consistencia de la base de datos: los algoritmos de control de datos semántico tienen que satisfacer siempre las restricciones de integridad cuando una transacción pretende hacer un commit.
 - Protocolos de confiabilidad: en transacciones distribuidas, es necesario introducir medios de comunicación entre los diferentes nodos de una red, para así garantizar la atomicidad y durabilidad de ellas. También se requieren protocolos para la recuperación local, y para efectuar los compromisos (commit) globales.
 - Algoritmos de control de concurrencia: éstos deben sincronizar la ejecución de transacciones concurrentes bajo el criterio de correctitud. La consistencia entre transacciones se garantiza mediante el aislamiento de las mismas.
 - Protocolos de control de réplicas: el control de réplicas se refiere a cómo garantizar la consistencia mutua de datos replicados.