



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO

FACULTAD DE INGENIERÍA

BASES DE DATOS

GRUPO:1

PROFESOR: ING. FERNANDO ARREOLA
FRANCO

ALUMNO: CALVILLO MARTÍNEZ ERIK
JONATHAN

TAREA 23



Concurrencia

A nivel sistema operativo

La concurrencia es una propiedad de los sistemas en la cual los procesos de un cómputo se hacen simultáneamente, y pueden interactuar entre ellos, es decir son procesados al mismo tiempo, de manera que, para ejecutar uno de ellos, no hace falta que se haya ejecutado otro.

En otras palabras es la ejecución de varias tareas al mismo tiempo, esto relativo de acuerdo a la arquitectura del ordenador, si nosotros persivimos la ilusión de que varios programas se ejecutan al mismo tiempo (Multiprogramación) es debido a que el sistema operativo intercala las tareas que se ejecutan en el procesador a una velocidad sumamente rápida.

Los procesos son concurrentes si existen simultáneamente. Los procesos concurrentes pueden funcionar en forma totalmente independiente unos de otros, o pueden ser asíncronos, lo cual significa que en ocasiones requieren cierta sincronización o cooperación.

La concurrencia puede presentarse en tres contextos diferentes:

- Varias aplicaciones: para este caso se invento la multiprogramación, esto para permitir que el tiempo de procesador de la máquina fuese compartido dinámicamente entre varios trabajos o aplicaciones activas, cada programa es independiente del otro.
- Aplicaciones estructuradas: Algunas aplicaciones pueden implementarse eficazmente como un conjunto de procesos concurrentes. Esto quiere decir, que existen programas que por lo regular para funcionar requieren ejecutar varias tareas a la vez o tener varios hilos de ejecución.
- Estructura del sistema operativo: Un sistema operativo es basicamente un conjunto de programas que se ejecutan concurrentemente para ofrecerte la mejor experiencia de trabajo

Tipos de Procesos Concurrentes:

Proceso independiente: Es aquel que ejecuta sin requerir la ayuda o cooperación de otros procesos. Un claro ejemplo de procesos independientes son los diferentes shells que se ejecutan de forma simultánea en un sistema. En español es todo aquel proceso que trabaja por si solo y que no requiere de ayuda de otro para funcionar adecuadamente.

Procesos son cooperantes: Son aquellos que están diseñados para trabajar conjuntamente en alguna actividad, para lo que deben ser capaces de comunicarse e interactuar entre ellos. Las interacciones pueden ser motivadas porque los procesos comparten o compiten por el acceso a recursos físicos o lógicos. También porque puede darse el caso de que los procesos se comunican y sincronizan entre sí para alcanzar un objetivo común

A nivel base de datos

La concurrencia de bases de datos es la capacidad de una base de datos para permitir que varios usuarios afecten a varias transacciones. Esta es una de las principales propiedades que separa una base de datos de otras formas de almacenamiento de datos, como las hojas de cálculo.

Las hojas de cálculo no pueden ofrecer a varios usuarios la capacidad de ver y trabajar con los diferentes datos en el mismo archivo, porque una vez que el primer usuario abre el archivo, está bloqueado para otros usuarios. Otros usuarios pueden leer el archivo, pero no pueden editar los datos.

Por ejemplo, cuando un usuario está cambiando datos, pero aún no ha guardado (confirmado) esos datos, entonces la base de datos no debe permitir que otros usuarios que consultan los mismos datos vean los datos cambiados y no guardados. En cambio, el usuario solo debe ver los datos originales.

Una vez que se guarda, se escribe en el almacenamiento físico de la base de datos en lugar de los datos originales. Siempre que el usuario que realiza el cambio no haya guardado los datos, solo él debería poder ver los datos que está cambiando. Todos los demás usuarios que soliciten los mismos datos deben ver los datos que existían antes del cambio. Una vez que el usuario guarda los datos, las nuevas consultas deberían revelar el nuevo valor de los datos.

Lo expuesto se logra cumpliendo con las propiedades ACID:

Atomicidad: una transacción debe ser una unidad atómica de trabajo, es decir, que se realicen todas sus modificaciones en los datos o no se realice ninguna.

Coherencia: cuando una transacción finaliza debe dejar todos los datos en un estado coherente. Deberán aplicarse todas las reglas a las modificaciones de la transacción permitiendo de esa manera mantener la integridad de los datos.

Aislamiento: las modificaciones realizadas por transacciones simultáneas se deben bloquear de las llevadas a cabo por otras transacciones simultáneas.

Durabilidad: el SGBD asegura que perduren los cambios realizados por una transacción que termina con éxito.

En ocasiones los usuarios tienen acceso a los datos de manera simultánea, es decir, leen o modifican los mismos datos al mismo tiempo. Cuando esto no se controla pueden suceder algunas de las siguientes situaciones problemáticas:

Actualizaciones perdidas: cuando dos o más transacciones seleccionan los mismos datos. Con la última actualización se sobrescriben las actualizaciones realizadas por otras transacciones produciendo pérdida de datos

Dependencia no confirmada (lectura no actualizada): cuando una transacción selecciona datos que están siendo actualizadas por otra transacción, lo que conlleva a que la segunda transacción acceda a datos que todavía no han sido confirmados.

Análisis contradictorios (lectura irrepensible): cuando una transacción obtiene acceso a los mismos datos varias veces y en cada ocasión accede a datos diferentes. Esto se parece a la dependencia confirmada, la diferencia radica en que los datos ya han sido confirmados por la segunda transacción, con lo cual la lectura se hace irrepensible

Lecturas fantasmas: es una situación particular que se produce en un SGBD cuando se ejecutan dos consultas idénticas al mismo tiempo y la recopilación de los datos devueltos por la segunda consulta es diferente a lo que se obtiene con la primera

La teoría del control de simultaneidad tiene dos clasificaciones:

Control de simultaneidad pesimista: un sistema de bloqueos impide que los usuarios modifiquen los datos de forma que afecte a otros usuarios. Cuando un usuario lleve a cabo una acción que da lugar a que se aplique un bloqueo, los demás usuarios no podrán realizar acciones que crearían conflictos con el bloqueo, hasta que sea liberado. Se utiliza principalmente en entornos donde existen conflictos por la obtención de datos, y en los que el costo de la protección de los mismos con bloqueos es menor que el de revertir las transacciones si se producen conflictos de simultaneidad.

Control de simultaneidad optimista: no se bloquean los datos cuando se acceden a ellos para la lectura. Cuando un usuario realiza una actualización de datos, el sistema comprueba si otro usuario los modificó después de la lectura. Si se actualizaron los datos, se produce un error. Normalmente, el usuario que recibe el error revierte la transacción y comienza de nuevo. Se utiliza principalmente en entornos donde hay pocos problemas de contención por la obtención de datos y en los que el costo de revertir ocasionalmente una transacción es menor que el de bloquear los datos cuando son leídos.

A continuación se mencionan las técnicas con la que se logran evitar problemas de concurrencia aplicadas a la simultaneidad de transacciones en un SGBD. Como se indicó anteriormente entre las técnicas de concurrencia existen dos grandes grupos:

Control de simultaneidad pesimista:
Técnicas basadas en bloqueos
Técnicas Multi-Granulares
Técnicas basadas en marcas temporales

Control de simultaneidad optimista:
Técnicas basadas en Validación.
Técnica original de control de concurrencia optimista

Referencias

- [1] "Concurrencia en base de datos". [Online]. Disponible en:
http://sedici.unlp.edu.ar/bitstream/handle/10915/73561/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y
- [2] "Concurrencia de la base de datos". [Online]. Disponible en:
<https://techinfo.wiki/concurrencia-de-la-base-de-datos/>
- [3] "Concurrencia y Secuencialidad (Sistemas Operativos)". [Online]. Disponible en:
<https://studentplace98.blogspot.com/2018/09/concurrencia-y-secuencialidad-sistemas.html#:text=Concurrencia>