



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE INFORME DE INVESTIGACIÓN / PRÁCTICA

PROFESOR	NOMBRE DE LA INVESTIGACIÓN /PRÁCTICA:	ASIGNATURA
Ing. Gonzalo Borja	Concurrencia de datos	SISTEMA AVANZADO DE BASE DE DATOS-9990

NOMBRES (ALUMNOS)

Pablo Guevara Fabara

Thalia Torres

2023-05-04

OBJETIVOS

Objetivo general:

- El objetivo general de este informe es proporcionar una visión general del control de concurrencia en las bases de datos en el desarrollo de software, destacando su importancia y los diferentes métodos y estrategias utilizados para garantizar la integridad y consistencia de los datos en entornos concurrentes.

ANÁLISIS Y DESARROLLO

El control de concurrencia es un aspecto crítico en el desarrollo de software que involucra el acceso y manipulación de datos en una base de datos por parte de múltiples usuarios o procesos de manera simultánea. El objetivo principal del control de concurrencia es garantizar la integridad de los datos y evitar problemas como las lecturas y escrituras inconsistentes o la pérdida de actualizaciones. Este informe proporcionará una visión



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
INFORME DE INVESTIGACIÓN / PRÁCTICA

general del control de concurrencia en las bases de datos y su importancia en el desarrollo de software.

1. Control de concurrencia: El control de concurrencia es un conjunto de técnicas y mecanismos utilizados para coordinar y gestionar el acceso concurrente a los datos en una base de datos por parte de múltiples usuarios o procesos. Su objetivo principal es garantizar la integridad y consistencia de los datos, evitando problemas como lecturas y escrituras inconsistentes, pérdida de actualizaciones y conflictos entre transacciones.
2. Transacciones: Una transacción es una unidad lógica de trabajo que agrupa una serie de operaciones de base de datos. Puede consistir en lecturas y escrituras de datos y debe cumplir con las propiedades ACID: Atomicidad, Consistencia, Aislamiento y Durabilidad. La atomicidad garantiza que todas las operaciones de una transacción se completen con éxito o ninguna se lleve a cabo. La consistencia asegura que la base de datos pasa de un estado válido a otro estado válido después de una transacción. El aislamiento proporciona a cada transacción una vista consistente de los datos, como si se estuviera ejecutando en forma aislada. La durabilidad garantiza que los cambios realizados por una transacción sean permanentes y persistentes, incluso en caso de fallos del sistema.
3. Problemas de concurrencia: En un entorno de múltiples transacciones concurrentes, pueden surgir varios problemas de concurrencia. Algunos de los problemas comunes incluyen:
 - Lecturas Sucias (Dirty Reads): Ocurren cuando una transacción lee datos no confirmados o temporales de otra transacción que aún no se ha completado. Esto puede llevar a la toma de decisiones basadas en información incorrecta.



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
INFORME DE INVESTIGACIÓN / PRÁCTICA

- **Escrituras Perdidas (Lost Updates):** Se producen cuando dos transacciones intentan actualizar los mismos datos de manera concurrente, y una de las actualizaciones se pierde, lo que resulta en una pérdida de datos.
 - **Lecturas No Repetibles (Non-Repeatable Reads):** Ocurren cuando una transacción realiza múltiples lecturas de un mismo dato durante su ejecución y obtiene resultados diferentes debido a que otros procesos han modificado esos datos.
 - **Lecturas Fantasma (Phantom Reads):** Ocurren cuando una transacción realiza una consulta y obtiene un conjunto de resultados, pero posteriormente, al repetir la misma consulta, se obtienen resultados adicionales o diferentes debido a las inserciones o eliminaciones realizadas por otras transacciones.
4. **Métodos de control de concurrencia:** Existen varios métodos y técnicas utilizadas para el control de concurrencia en bases de datos:
- **Bloqueo (Locking):** Es el enfoque más común y ampliamente utilizado. Consiste en adquirir bloqueos (locks) en los datos para evitar que otras transacciones los modifiquen mientras una transacción está en curso. Los bloqueos pueden ser exclusivos (escritura) o compartidos (lectura) y se liberan al finalizar la transacción.
 - **Control de Versiones (Versioning):** En lugar de bloquear los datos, este enfoque mantiene múltiples versiones de un mismo dato. Cada transacción trabaja con una versión consistente de los datos y no bloquea a otras transacciones. Esto permite una mayor concurrencia, pero puede aumentar el consumo de almacenamiento.
 - **Estampado de Tiempo (Timestamping):** Asigna un sello de tiempo (timestamp) a cada transacción y/o dato. Se utilizan para determinar el orden de ejecución de las transacciones y resolver conflictos. Las transacciones más recientes pueden sobrescribir los cambios de transacciones anteriores.



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE INFORME DE INVESTIGACIÓN / PRÁCTICA

5. Estrategias de control de concurrencia: Existen diferentes estrategias utilizadas para controlar la concurrencia en bases de datos:

- Bloqueo Optimista (Optimistic Locking): Esta estrategia supone que los conflictos de concurrencia son poco frecuentes. Permite a las transacciones operar sin bloqueos y verifica al final si ha habido algún conflicto. Si se detecta un conflicto, se revierten los cambios y se reinicia la transacción.
- Bloqueo Pesimista (Pessimistic Locking): Esta estrategia supone que los conflictos de concurrencia son comunes. Adquiere bloqueos antes de realizar operaciones y los mantiene hasta que la transacción finaliza. Puede causar mayor bloqueo y reducir la concurrencia, pero garantiza la consistencia de los datos.
- Control de Concurrencia Multiversión (Multiversion Concurrency Control - MVCC): Esta estrategia utiliza el control de versiones para permitir que las transacciones trabajen con diferentes versiones de los datos. Cada transacción ve una "instantánea" de los datos en el momento en que comenzó.

Herramientas:

1. Sistemas de Gestión de Bases de Datos (SGBD) transaccionales:

- MySQL: Un popular sistema de gestión de bases de datos relacional que ofrece soporte para transacciones y control de concurrencia a través de bloqueo y control de versiones.
- PostgreSQL: Otro SGBD relacional que ofrece un sólido soporte para transacciones y control de concurrencia utilizando diferentes métodos como bloqueo, MVCC y estampado de tiempo.



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
INFORME DE INVESTIGACIÓN / PRÁCTICA

- Oracle Database: Un SGBD empresarial que proporciona funciones avanzadas de control de concurrencia, incluyendo bloqueo, MVCC y control de transacciones distribuidas.

2. ORM (Object-Relational Mapping):

- Hibernate: Un popular framework de mapeo objeto-relacional para Java que ofrece funcionalidades de control de concurrencia, como bloqueo optimista y pesimista.
- Entity Framework: Un ORM de Microsoft para .NET que proporciona soporte para control de concurrencia a través de bloqueo optimista y pesimista.

3. Bibliotecas y Frameworks:

- Java Concurrency API: Proporciona una serie de clases y utilidades para manejar la concurrencia en aplicaciones Java, como locks, semáforos y barreras.
- .NET Task Parallel Library: Una biblioteca de .NET que permite la programación paralela y concurrente, proporcionando soporte para control de concurrencia a través de locks y monitores.
- Spring Framework: Un framework de desarrollo de aplicaciones Java que ofrece soporte para transacciones y control de concurrencia mediante el uso de anotaciones y configuración declarativa.

4. Control de versiones y herramientas de gestión de código:

- Git: Un sistema de control de versiones distribuido ampliamente utilizado que permite a los equipos de desarrollo gestionar y coordinar los cambios en el código fuente de manera concurrente.
- SVN (Subversion): Otro sistema de control de versiones ampliamente utilizado que permite el control de cambios y la gestión concurrente de proyectos de desarrollo de software.



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE INFORME DE INVESTIGACIÓN / PRÁCTICA

CONCLUSIONES

El control de concurrencia desempeña un papel fundamental en el desarrollo de software que involucra bases de datos. Garantiza la integridad y consistencia de los datos en entornos concurrentes, evitando problemas como lecturas y escrituras inconsistentes, pérdida de actualizaciones y conflictos entre transacciones. Los métodos y estrategias de control de concurrencia, como el bloqueo, el control de versiones y el estampado de tiempo, ofrecen enfoques efectivos para abordar los desafíos de la concurrencia y deben seleccionarse cuidadosamente según las necesidades del sistema para asegurar una operación eficiente y sin conflictos en el desarrollo de software basado en bases de datos.

7. BIBLIOGRAFÍA

[Arduino, G., & Alfonzo, P. L. (2018). Técnicas de control de concurrencia en base de datos: implementación en un sistema de gestión. In *XXIV Congreso Argentino de Ciencias de la Computación (La Plata, 2018)*..

Canché Euán, M. Control de concurrencia basado en componentes para una base de datos distribuida- Edición Única.

Kern, V. M. Una técnica de baja tecnología para el aprendizaje de control de concurrencia en bancos de datos.