



Universidad de Huelva



Escuela Técnica Superior de Ingeniería
Universidad de Huelva

Concurrencia de Datos y Consistencia

Administración de Bases de Datos

Grado en Ingeniería Informática

Autor:

José Ortega Lepe

Gonzalo Muñoz Quintero

20 de abril de 2018

Índice general

1. INTRODUCCIÓN	4
1.1. CONSISTENCIA DE LECTURA MULTIVERSIONAL	5
1.1.2. CONSISTENCIA DE LECTURA A NIVEL DE TRANSACCIÓN.....	5
1.1.3. CONSISTENCIA DE LECTURA Y SEGMENTOS DE DESHACER	5
2. VISIÓN GENERAL DE LOS NIVELES DE AISLAMIENTO DE TRANSACCIÓN	7
2.1. NIVEL DE AISLAMIENTO DE LECTURA CONFIRMADA.....	7
2.1.1. ESCRITURAS CONFLICTIVAS EN TRANSACCIONES DE LECTURAS CONFIRMADAS.....	7
2.2. NIVEL DE AISLAMIENTO SERIALIZABLE.....	7
2.3. NIVEL DE AISLAMIENTO DE SOLO LECTURA.....	8
3. DESCRIPCIÓN GENERAL DEL MECANISMO DE BLOQUEO EN ORACLE	9
3.1. MODOS DE BLOQUEO	9
3.2. DURACIÓN DEL BLOQUEO.....	10
3.3. BLOQUEOS E INTERBLOQUEO (DEADLOCK)	10
3.4. BLOQUEOS DML	10
3.4.1. BLOQUEOS DE FILA.....	11
3.4.2 BLOQUEOS DE TABLAS	11
3.4.3. BLOQUEOS Y CLAVES AJENAS (FOREIGN KEYS)	13
3.5. BLOQUEOS DDL	15
3.6. BLOQUEOS DE SISTEMA	15
4. DESCRIPCIÓN GENERAL DE LOS BLOQUEOS DE DATOS MANUALES.....	15
5. CONCLUSIÓN	16
6. BIBLIOGRAFÍA	17

Índice de figuras

Figura 1. Consulta que usa datos de deshacer.....	6
Figura 2. Bloqueo de fila.....	11
Figura 3. Bloqueos y claves externas (sin índices)	13
Figura 4. Bloqueos y claves externas (con índices)	14

1. INTRODUCCIÓN

En una base de datos de usuario único, un usuario puede modificar datos sin preocuparse de que otros usuarios estén modificando los mismos datos al mismo tiempo. Sin embargo, en una base de datos multiusuario, las declaraciones dentro de múltiples transacciones simultáneas pueden actualizar los mismos datos. Las transacciones que se ejecutan simultáneamente deben producir resultados consistentes. Por lo tanto, una base de datos multiusuario debe proporcionar lo siguiente:

- **Concurrencia de datos**, lo que garantiza que los usuarios puedan acceder a los datos al mismo tiempo.
- **Consistencia de los datos**, lo que garantiza que cada usuario vea una vista coherente de los datos, incluidos los cambios visibles realizados por las propias transacciones del usuario y las transacciones confirmadas de otros usuarios.

Para describir el comportamiento de transacción consistente cuando las transacciones se ejecutan al mismo tiempo, los investigadores de bases de datos han definido un modelo de aislamiento de transacciones llamado **serializabilidad**. Una **transacción serializable** opera en un entorno que hace que parezca que otros usuarios no están modificando datos en la base de datos.

Si bien este grado de aislamiento entre transacciones es generalmente deseable, se ejecuta muchas aplicaciones en modo serializable pueden comprometer seriamente la aplicación rendimiento. En resumen, las consideraciones del mundo real generalmente requieren un compromiso entre un perfecto aislamiento y un buen rendimiento de las transacciones.

Oracle Database mantiene la coherencia de los datos mediante el uso de una consistencia multiversional y varios tipos de bloqueos y transacciones. Las transacciones pueden leer la versión de los datos confirmados en el momento requerido por una consulta y devolver resultados que son consistentes con un único punto en el tiempo.

1.1. CONSISTENCIA DE LECTURA MULTIVERSIONAL

En Oracle Database, multiversioning (multiversiónar) es la capacidad de mantener simultáneamente múltiples versiones de datos. Esto significa que las consultas a la base de datos tienen las siguientes características:

- **Consultas de lectura consistente.** Los datos devueltos por una consulta están confirmados y son consistentes con respecto a un único punto en el tiempo.
- **Consultas sin bloqueo.** Los lectores y escritores de datos no se bloquean entre sí.

1.1.2. CONSISTENCIA DE LECTURA A NIVEL DE TRANSACCIÓN

Oracle Database también puede proporcionar coherencia de lectura a todas las consultas en una transacción, conocida como consistencia de lectura a nivel de transacción. En este caso, cada instrucción en una transacción ve datos del mismo punto en el tiempo, que es el momento en que la transacción comenzó.

Las consultas realizadas por una transacción serializable ven los cambios realizados por la transacción en sí. Por ejemplo, una transacción que actualiza a los empleados y luego consulta a los empleados ve las actualizaciones.

1.1.3. CONSISTENCIA DE LECTURA Y SEGMENTOS DE DESHACER

Para gestionar el modelo de coherencia de lectura multiversional, la base de datos debe crear un conjunto de datos consistentes con la lectura cuando una tabla se consulta y actualiza simultáneamente. Oracle Database logra este objetivo mediante datos de deshacer.

Cada vez que un usuario modifica los datos, Oracle Database crea las entradas de deshacer, que escribe para deshacer segmentos. Los segmentos de deshacer contienen valores antiguos de datos que han sido modificados, pero no confirmados o de transacciones recientes. Por lo tanto, múltiples versiones de los mismos datos, todos en diferentes momentos en el tiempo, puede existir en la base de datos.

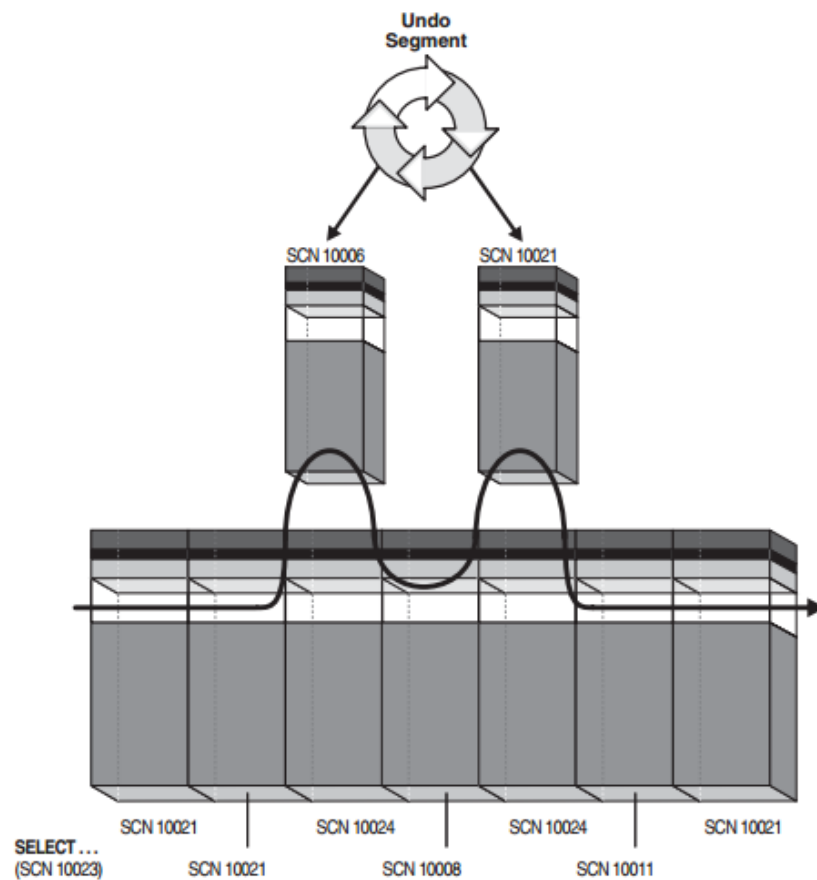


Figura 1. Consulta que usa datos de deshacer

La base de datos usa un mecanismo llamado SCN para garantizar el orden de las transacciones. A medida que la declaración `SELECT` ingresa a la fase de ejecución, la base de datos determina el SCN registrado en el momento en que la consulta comenzó a ejecutarse. En la **Figura 1**, este SCN es 10023. La consulta solo ve los datos confirmados con respecto a SCN 10023.

En la Figura 1, los bloques con SCN después de 10023 indican datos modificados, como se muestra en la figura dos bloques con SCN 10024. La declaración `SELECT` requiere una versión del bloque que es consistente con los cambios confirmados.

2. VISIÓN GENERAL DE LOS NIVELES DE AISLAMIENTO DE TRANSACCIÓN

Oracle Database proporciona los niveles de aislamiento de transacción:

- Nivel de aislamiento de lectura confirmada.
- Nivel de aislamiento serializable.
- Nivel de aislamiento de solo lectura.

2.1. NIVEL DE AISLAMIENTO DE LECTURA CONFIRMADA

En el nivel de aislamiento de lectura confirmada, que es el valor predeterminado, cada consulta ejecutada por una transacción solo ve los datos confirmados antes de que comenzara la consulta, no la transacción. Este nivel de aislamiento es apropiado para entornos de bases de datos en los que es poco probable que las transacciones entren en conflicto.

2.1.1. ESCRITURAS CONFLICTIVAS EN TRANSACCIONES DE LECTURAS CONFIRMADAS

En una transacción de lectura confirmada, se produce una escritura conflictiva cuando la transacción intenta cambiar una fila actualizada por una transacción concurrente no confirmada, a veces llamado una transacción de bloqueo. La transacción de lectura confirmada espera el bloqueo de la transacción para finalizar y liberar su bloqueo de fila.

2.2. NIVEL DE AISLAMIENTO SERIALIZABLE

En el nivel de aislamiento serializable, una transacción solo ve los cambios confirmados cuando comenzó la transacción, no la consulta, y los cambios realizados por la transacción. Una transacción serializable opera en un entorno que lo hace parecer como si ningún otro usuario estuviera modificando datos en la base de datos.

El aislamiento serializable es adecuado para entornos:

- Con grandes bases de datos y transacciones cortas que solo actualizan unas pocas filas.
- Donde la posibilidad de que dos transacciones simultáneas modifiquen las mismas filas es relativamente baja.
- Donde las transacciones de larga duración son principalmente de solo lectura.

Oracle Database permite que una transacción serializable modifique una fila solo si los cambios en la fila hechos por otras transacciones ya se confirmaron cuando la transacción serializable comenzó. La base de datos genera un error cuando una transacción serializable intenta actualizar o eliminar datos modificados por una transacción diferente que se haya confirmado después de que la transacción serializable comenzara:

ORA-08177: Cannot serialize access for this transaction

Cuando una transacción serializable falla con el error ORA-08177, una aplicación puede realizar varias acciones, incluidas las siguientes:

- Confirmar el trabajo ejecutado hasta ese punto.
- Ejecutar declaraciones adicionales (pero diferentes), tal vez después de retroceder a un savepoint establecido antes en la transacción.
- Revertir la transacción completa.

2.3. NIVEL DE AISLAMIENTO DE SOLO LECTURA

El nivel de aislamiento de solo lectura es similar al nivel de aislamiento serializable, pero de solo lectura. Las transacciones no permiten que los datos se modifiquen en la transacción a menos que el usuario sea SYS. Por lo tanto, las transacciones de solo lectura no son susceptibles al error ORA-08177. Las transacciones de solo lectura son útiles para generar informes en los que los contenidos deben ser consistentes respecto al momento en que comenzó la transacción.

3. DESCRIPCIÓN GENERAL DEL MECANISMO DE BLOQUEO EN ORACLE

Un bloqueo es un mecanismo que evita las interacciones destructivas, que son interacciones que actualiza los datos compartidos incorrectamente. Los bloqueos juegan una fila crucial en el mantenimiento de la concurrencia y consistencia de la base de datos.

La base de datos mantiene varios tipos de bloqueos, dependiendo de la operación que creó el bloqueo. En general, la base de datos utiliza dos tipos de bloqueos: **bloqueos exclusivos** y **bloqueos compartidos**. Solo se puede obtener un bloqueo exclusivo en un recurso como una fila o una tabla, pero muchos bloqueos compartidos se pueden obtener en un solo recurso.

Las siguientes reglas resumen el comportamiento de bloqueo de Oracle Database para lecturas y escrituras:

- Una fila se bloquea solo cuando la modifica una escritura.
- Una escritura de una fila bloquea a una escritura concurrente de la misma fila.
- Una lectura nunca bloquea a una escritura.
- Una escritura nunca bloquea a una lectura.

3.1. MODOS DE BLOQUEO

Oracle Database utiliza dos modos de bloqueo en una base de datos multiusuario:

- Modo de bloqueo exclusivo

Este modo evita que el recurso asociado sea compartido. Una transacción obtiene un bloqueo exclusivo cuando modifica los datos.

- Modo de bloqueo compartido

Este modo permite compartir el recurso asociado, dependiendo de las operaciones involucradas. Múltiples usuarios que leen datos pueden compartir los datos y compartir el bloqueo para evitar el acceso simultáneo de una escritura que necesita un bloqueo exclusivo.

3.2. DURACIÓN DEL BLOQUEO

Oracle Database libera automáticamente un bloqueo cuando se produce algún evento, de modo que la transacción ya no requiere el recurso.

Oracle Database libera todos los bloqueos adquiridos por las declaraciones dentro de una transacción cuando se confirma o revierte. Oracle Database también libera bloqueos adquiridos después de un savepoint cuando retrocede al punto de rescate. Sin embargo, solo las transacciones que no esperan para los recursos previamente bloqueados pueden adquirir bloqueos en los recursos disponibles ahora. Las transacciones en espera continúan esperando hasta después de que la transacción original se confirme o revierta por completo.

3.3. BLOQUEOS E INTERBLOQUEO (DEADLOCK)

Un interbloqueo o deadlock es una situación en la que dos o más usuarios esperan datos bloqueados tanto por uno como por otro. Los interbloqueos evitan que algunas transacciones continúen funcionando.

Oracle Database detecta automáticamente interbloqueos y los resuelve al retroceder una declaración involucrada en el interbloqueo, liberando un conjunto de bloqueos de filas en conflicto. La declaración revertida pertenece a la transacción que detecta el interbloqueo.

3.4. BLOQUEOS DML



Un bloqueo DML, también llamado bloqueo de datos garantiza la integridad de los datos a los que se accede concurrentemente por múltiples usuarios. Por ejemplo, un bloqueo DML impide que dos clientes compren la última copia de un libro disponible de una librería en línea. Las declaraciones DML adquieren automáticamente los siguientes tipos de bloqueos:

- Bloqueos de fila (TX).
- Bloqueos de tabla (TM).

3.4.1. BLOQUEOS DE FILA

Un bloqueo de fila, también llamado bloqueo de TX es un bloqueo en una sola fila de la tabla. Una transacción adquiere un bloqueo de fila para cada fila modificada por INSERT, UPDATE, DELETE, MERGE o SELECT ... FOR UPDATE. El bloqueo de fila existe hasta que la transacción se confirma o revierte.

Si una transacción obtiene un bloqueo para una fila, entonces la transacción también adquiere un bloqueo para la tabla que contiene la fila.

Table EMPLOYEES 						
EMPLOYEE_ID	LAST_NAME	EMAIL	HIRE_DATE	JOB_ID	MANAGER_ID	DEPARTMENT_ID
 100	King	SKING	17-JUN-87	AD_PRES		
101	Kochhar	NKOCHHAR	21-SEP-89	AD_VP	100	90
102	De Hann	LDEHANN	13-JAN-93	AD_VP	100	90
103	Hunold	AHUNOLD	03-JAN-90	IT_PROG	102	60




 Table lock acquired
 Exclusive row lock (TX) acquired
 Row being updated

Figura 2. Bloqueo de fila

Oracle Database también tiene **almacenamiento de bloqueos de fila**. A diferencia de algunas bases de datos, que usan un administrador de bloqueos para mantener una lista de bloqueos en la memoria, Oracle Database almacena información de bloqueo en el bloque de datos que contiene la fila bloqueada. La base de datos utiliza un mecanismo de cola para la adquisición de bloqueos de fila.

3.4.2 BLOQUEOS DE TABLAS

Un bloqueo de tabla, también llamado bloqueo TM, es adquirido por una transacción cuando una tabla es modificada por INSERT, UPDATE, DELETE, MERGE, SELECT con la cláusula FOR UPDATE, o la sentencia LOCK TABLE.

Se puede mantener un bloqueo de tabla en cualquiera de los siguientes modos:

- **Bloqueo de fila compartido (RS).** Este bloqueo, también llamado bloqueo de tabla subcompartido (SS), indica que la transacción que tiene el bloqueo de la tabla tiene filas bloqueadas en la tabla y tiene la intención de actualizarlas. Un bloqueo de fila compartido es el modo menos restrictivo de bloqueo de tabla, que ofrece el mayor grado de concurrencia para una tabla.
- **Bloqueo de tabla exclusivo de fila (RX).** Este bloqueo, también llamado bloqueo de tabla subexclusivo (SX), generalmente indica que la transacción que tiene el bloqueo ha actualizado las filas de la tabla o emitido `SELECT ... FOR UPDATE`. Un bloqueo SX permite que otras transacciones consulten, inserten, actualicen, eliminen o bloqueen filas al mismo tiempo en la misma tabla. Por lo tanto, los bloqueos SX permiten múltiples transacciones para obtener bloqueos SX y SS simultáneos para la misma tabla.
- **Bloqueo de tabla compartido (S).** Un bloqueo de tabla compartido obtenido por una transacción permite que otras transacciones consulten la tabla (sin usar `SELECT ... PARA UPDATE`), sin embargo, las actualizaciones solo se permiten si una sola transacción contiene el bloqueo de tabla compartido.
- **Bloqueo de tabla exclusivo de fila compartido (SRX).** Este bloqueo, también llamado bloqueo de tabla subexclusivo compartido (SSX), es más restrictivo que un bloqueo de tabla compartido. Solo una transacción a la vez puede adquirir un bloqueo SSX en una tabla determinada. Un bloqueo de SSX obtenido por una transacción permite que otras transacciones consulten la tabla (excepto `SELECT ... FOR UPDATE`) pero no actualizar la tabla.
- **Bloqueo de tabla exclusivo (X).** Este bloqueo es el más restrictivo, lo que impide que otras transacciones ejecuten cualquier tipo de declaración de DML u obtengan cualquier tipo de bloqueo en la tabla.

3.4.3. BLOQUEOS Y CLAVES AJENAS (FOREIGN KEYS)

Oracle Database maximiza el control de concurrencia de las relaciones de claves primarias con claves ajenas dependientes. El comportamiento del bloqueo depende de si las columnas de clave ajena están indexadas (tienen índices). Si las claves ajenas no están indexadas, entonces la tabla secundaria probablemente será bloqueada con más frecuencia, se producirán bloqueos y se reducirá la concurrencia. Es por esta razón que las claves ajenas casi siempre deben indexarse.

Cuando las dos condiciones siguientes se cumplen, la base de datos adquiere un bloqueo de tabla completo en la tabla secundaria:

- No existe un índice en la columna de clave ajena de la tabla secundaria.
- Una sesión modifica una clave primaria (por ejemplo, elimina una fila o modifica atributos de clave primaria) o combina filas en la tabla principal. Las inserciones en la tabla principal no requieren bloqueos de tabla en la tabla secundaria.

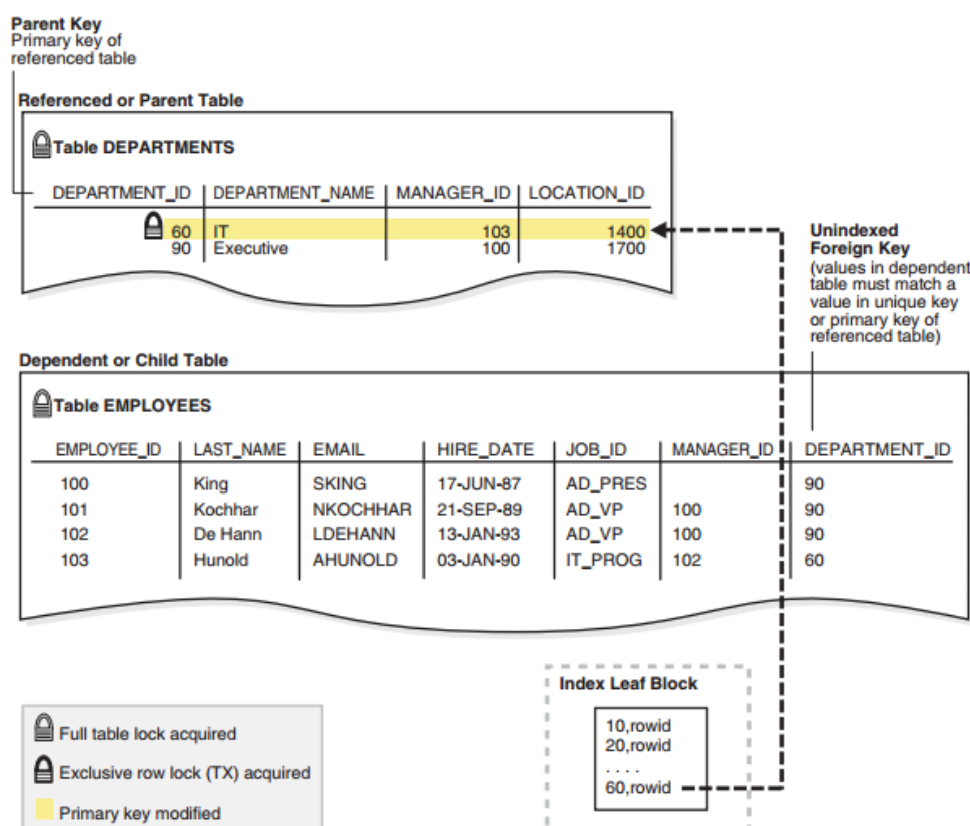


Figura 3. Bloqueos y claves externas (sin índices)

Cuando las dos condiciones siguientes se cumplen, la base de datos no requiere un bloqueo de tabla completo en la tabla secundaria:

- Existe un índice en la columna de clave ajena en la tabla secundaria.
- Una sesión modifica una clave primaria en la tabla principal (por ejemplo, elimina una fila o modifica atributos de clave primaria) o combina filas en la tabla principal.

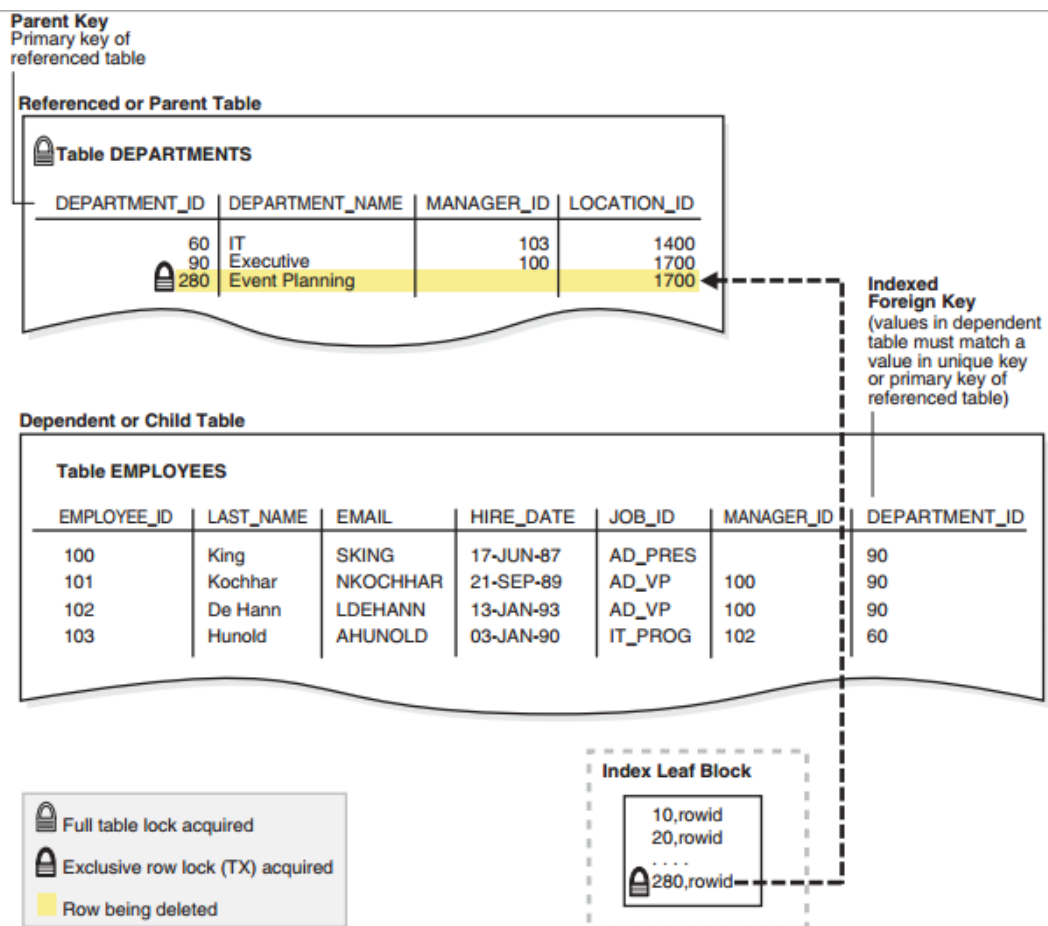


Figura 4. Bloqueos y claves externas (con índices)

3.5. BLOQUEOS DDL

Un bloqueo de diccionario de datos (DDL) protege la definición de un objeto de esquema mientras la operación continua de DDL actúa sobre el objeto o se refiere al objeto. Solo objetos de esquema individuales que están modificados o referenciados están bloqueados durante las operaciones DDL. La base de datos nunca bloquea todo el diccionario de datos.

Los usuarios no pueden solicitar bloqueos DDL explícitamente. Por ejemplo, si un usuario crea un procedimiento almacenado, Oracle Database adquiere automáticamente los bloqueos DDL para todos objetos de esquema a los que se hace referencia en la definición del procedimiento. Los bloqueos DDL protege a estos objetos de ser alterados o eliminados antes de que se complete la compilación del procedimiento.

3.6. BLOQUEOS DE SISTEMA

Oracle Database utiliza varios tipos de bloqueos de sistema para proteger la base de datos interna y la estructura de la memoria. Estos mecanismos son inaccesibles para los usuarios porque los usuarios no tienen control sobre su ocurrencia o duración.

4. DESCRIPCIÓN GENERAL DE LOS BLOQUEOS DE DATOS MANUALES

Oracle Database realiza el bloqueo automáticamente para garantizar la concurrencia de datos, integridad de los datos y la consistencia de lectura a nivel de declaración. Sin embargo, se pueden anular manualmente los mecanismos de bloqueo predeterminados de Oracle Database. La modificación del bloqueo predeterminado es útil en situaciones como las siguientes:

- Las aplicaciones requieren consistencia de lectura a nivel de transacción o lecturas repetibles.
- Las aplicaciones requieren que una transacción tenga acceso exclusivo a un recurso para que la transacción no tenga que esperar a que se completen otras transacciones.

Se puede anular el bloqueo automático de Oracle Database a nivel de sesión o a nivel de transacción. En el nivel de sesión, una sesión puede establecer el nivel de aislamiento de transacción requerido con la sentencia ALTER SESSION. En el nivel de transacción, las transacciones que incluyen las siguientes instrucciones SQL anulan el bloqueo predeterminado de la Base de Datos Oracle:

- La sentencia SET TRANSACTION ISOLATION LEVEL.
- La sentencia LOCK TABLE (que bloquea una tabla o, cuando se usa con vistas, las tablas base).
- La sentencia SELECT... FOR UPDATE.

Los bloqueos adquiridos por los enunciados anteriores se liberan después de que finaliza la transacción o se realice un rollback. Si el bloqueo predeterminado de Oracle Database se deshabilita en cualquier nivel, entonces el administrador de la base de datos o el desarrollador de la aplicación debe asegurarse de que los bloqueos deshabilitados funcionen correctamente.

5. CONCLUSIÓN

Oracle Database mantiene una concurrencia y consistencia de los datos haciendo uso de las transacciones serializables. Para ello usa diferentes niveles de aislamiento y modos de bloqueos que proporcionan cierta consistencia de los datos y estructuras de la base de datos.

Este trabajo nos ha ayudado a comprender mejor como Oracle gestiona la concurrencia de las diferentes transacciones realizadas dentro de la base de datos. También hemos aprendido los problemas que pueden ocasionar dicha concurrencia y como Oracle soluciona dichos problemas.

6. BIBLIOGRAFÍA

<https://docs.oracle.com/database/121/CNCPT/consist.htm#CNCPT020>

<https://fredericktang.wordpress.com/2008/10/10/foreign-keys-and-locking-2/>

<http://www.orafaq.com/wiki/SCN>