

## ADVERTENCIAS:

- Para el desarrollo de esta práctica se necesitan los contenidos teóricos aportados en los temas 1, 2 y 6.
- Usar Enterprise Manager (conectado con usuario SYS como SYSDBA) para todos los pasos descritos, mientras no se mencione **explícitamente** otra aplicación o conexión.

## PRÁCTICA 6

### MOVER LA TABLA A OTRO SEGMENTO.

### RECONSTRUCCIÓN DE ÍNDICES INVÁLIDOS.

- 1) ALTER TABLE ... MOVE permite mover los datos de una tabla, o de una partición, a un nuevo segmento, modificando en el transcurso de su operación **cualquiera de los atributos de almacenamiento** de la tabla o partición, incluso aquellos que no pueden ser modificados con un alter table.

No permite LMD concurrente sobre la tabla mientras se está ejecutando. Causa que cambien los identificadores (rowids) de las filas de la tabla y, dado que MOVE no se ocupa de mantener válidos los índices, invalida los índices de la tabla que deberán ser reconstruidos. Asimismo, invalida cualquier estadística recolectada previamente para la tabla.

Como se trata de mover la tabla a un nuevo segmento, va a necesitar asignación temporal de espacio extra, aunque posteriormente libere el espacio correspondiente al antiguo segmento.

Usar ISQL\*Plus, conectado con usuario SYSTEM, para borrar la tabla A sin que vaya a la papelera (drop table A purge). Cargar el guión 'Práctica 6 Complemento 1' y ejecutarlo. En realidad sólo hacía falta volver a crear la tabla A, puesto que los procedimientos almacenados siguen existiendo; pero como las sentencias de creación de los procedimientos usan la cláusula OR REPLACE, no hay error al recrearlos. Observe que se elige borrar la tabla y recrearla en lugar de truncarla, para cobrar conciencia de que la tabla A ya no tiene el permiso de movimiento de filas que se le había concedido.

Usar el procedimiento almacenado 'inserta\_en\_A' para insertar 450000 registros con valores del campo 'a' entre 1 y 450000. Comprometer la transacción y comprobar el contenido de la tabla A, para ello contar su número de registros (select count(\*) from A).

- 2) Comprobar el contenido del tablespace PRUEBA y su mapa de extensiones.
- 3) Usar ISQL\*Plus, conectado con usuario SYSTEM, para vaciar la tabla A (delete from A) (comparar el tiempo empleado en el delete frente al del truncate y buscar una respuesta sobre el porqué de esta diferencia). Comprometer la transacción y comprobar el contenido de la tabla A, para ello contar su número de registros (select count(\*) from A).

**Pregunta:** ¿Cuál es la razón de las diferencias en tiempo entre ejecutar un ‘truncate’ y un ‘delete’?

- 4) Comprobar el contenido del tablespace PRUEBA y su mapa de extensiones.
- 5) Usar ISQL\*Plus, conectado con usuario SYSTEM, para determinar con UNUSED\_SPACE la marca de marea alta del segmento de la tabla A. Observar que por encima de dicha marca queda espacio asignado al segmento de la tabla que no está usado.

**Pregunta:** ¿El valor de qué parámetros son los que nos permiten conocer la situación de la marca de marea alta?

- 6) Usar ISQL\*Plus, conectado con usuario SYSTEM, para cambiar la tabla A de segmento (alter table A move). Observar el contenido del tablespace PRUEBA y su mapa de extensiones. El nuevo segmento de la tabla A tiene el tamaño apropiado para su contenido (ninguno). Comprobar que el índice CLAVE\_PRIMARIA\_DE\_A sigue siendo válido, dado que la tabla está vacía.
- 7) Usar ISQL\*Plus, conectado con usuario SYSTEM, para determinar con UNUSED\_SPACE la marca de marea alta del segmento de la tabla A.

Usar ISQL\*Plus, conectado con usuario SYSTEM, para truncar la tabla A. Usar el procedimiento almacenado 'inserta\_en\_A' para insertar 450000 registros con valores del campo 'a' entre 1 y 450000. Ejecutar el procedimiento almacenado 'borra\_en\_A', a fin de eliminar los registros cuyo valor del campo 'a' estén entre 1 y 300000, ambos inclusive. Comprometer la transacción y comprobar el contenido de la tabla A, para ello contar su número de registros (select count(\*) from A). Comprobar el contenido del tablespace PRUEBA y su mapa de extensiones.

Usar ISQL\*Plus, conectado con usuario SYSTEM, para determinar con UNUSED\_SPACE la marca de marea alta del segmento de la tabla A. Observar que por encima de dicha marca queda espacio asignado al segmento de la tabla que no está usado.

**Pregunta:** ¿Qué valores de los parámetros obtiene antes y después de la inserción y posterior borrado de los elementos de la tabla A?

- 8) Usar ISQL\*Plus, conectado con usuario SYSTEM, para cambiar la tabla A de segmento (alter table A move) y determinar con UNUSED\_SPACE la marca de marea alta del segmento de la tabla A.

**Pregunta:** ¿Qué valores de los parámetros obtiene tras cambiar la tabla ‘A’ de segmento?

- 9) Observar el contenido del tablespace PRUEBA y su mapa de extensiones. El nuevo segmento de la tabla A tiene el tamaño apropiado para su contenido. Comprobar que el índice CLAVE\_PRIMARIA\_DE\_A ya no es válido.
- 10) En la última ejecución del UNUSED\_SPACE se observó que por encima de la marca de marea alta queda espacio asignado al segmento de la tabla que no está usado. Usar

ISQL\*Plus, conectado con usuario SYSTEM, para desasignarlo (alter table A deallocate unused) y comprobar sus consecuencias.

**Pregunta:** ¿Qué valores de los parámetros obtiene?

- 11) Usar ISQL\*Plus, conectado con usuario SYSTEM, para reconstruir el índice inválido CLAVE\_PRIMARIA\_DE\_A (alter index CLAVE\_PRIMARIA\_DE\_A rebuild) y comprobar su validez y el tamaño de su segmento.

**Pregunta 1:** ¿En qué estado se encuentra el índice una vez reconstruido? ¿Cuál es el tamaño del índice antes y después de la reconstrucción?

**Pregunta 2:** ¿Es el *REBUILD* una operación *in-situ*?

- 12) Usar ISQL\*Plus, conectado con usuario SYSTEM, para truncar la tabla A. Usar el procedimiento almacenado 'inserta\_en\_A' para insertar 650000 registros con valores del campo 'a' entre 1 y 650000. Ejecutar el procedimiento almacenado 'borra\_en\_A', a fin de eliminar los registros cuyo valor del campo 'a' estén entre 1 y 100000, ambos inclusive. Comprometer la transacción y comprobar el contenido de la tabla A, para ello contar su número de registros (select count(\*) from A).
- 13) Comprobar el contenido del tablespace PRUEBA y su mapa de extensiones.
- 14) Usar ISQL\*Plus, conectado con usuario SYSTEM, para cambiar la tabla A de segmento (alter table A move). Falla porque Oracle es incapaz de asignar el espacio necesario para el nuevo segmento en el tablespace PRUEBA.
- 15) Crear un nuevo tablespace permanente denominado PRUEBA\_TEMP, manejado localmente, con gestión automática del espacio del segmento y asignación automática de las mismas. Asignarle un único archivo de datos de tamaño inicial 100M, autoextensible en incrementos de 10M y tamaño máximo 1G, llamado PRUEBA\_TEMP. Aprovechar para mirar la sentencia SQL correspondiente y observar las cláusulas nuevas.

**Pregunta:** ¿Cuál es la instrucción SQL exacta empleada en la creación de PRUEBA\_TEMP?

- 16) Usar ISQL\*Plus, conectado con usuario SYSTEM, para mover la tabla A a un segmento en el nuevo tablespace PRUEBA\_TEMP (alter table A move tablespace PRUEBA\_TEMP).

**Pregunta:** Observe que tarda muy poco en realizarse. Dado que la tabla movida tiene 550000 registros explique por qué al hacer una reducción de segmento en una tabla, incluso con menos registros, tarda substancialmente más.

- 17) Comprobar el contenido y el mapa de extensiones de los tablespaces PRUEBA y PRUEBA\_TEMP. Verificar que el índice CLAVE\_PRIMARIA\_DE\_A está marcado como inservible.
- 18) Usar ISQL\*Plus, conectado con usuario SYSTEM, para mover de nuevo la tabla A (una vez compactada) a un segmento en el tablespace PRUEBA (alter table A move tablespace PRUEBA).

19) Comprobar el contenido y el mapa de extensiones de los tablespaces PRUEBA y PRUEBA\_TEMP. Verificar que el índice CLAVE\_PRIMARIA\_DE\_A sigue marcado como inservible.

20) Usar ISQL\*Plus, conectado con usuario SYSTEM, para reconstruir el índice CLAVE\_PRIMARIA\_DE\_A (`alter index CLAVE_PRIMARIA_DE_A rebuild`). Falla porque Oracle es incapaz de asignar el espacio necesario para la reconstrucción en el tablespace PRUEBA. Usamos, pues, espacio temporal en el nuevo tablespace PRUEBA\_TEMP para reconstruir el índice (`alter index CLAVE_PRIMARIA_DE_A rebuild tablespace PRUEBA_TEMP`) y comprobamos su validez y el tamaño de su segmento.

**Pregunta:** El *rebuild* que se acaba de realizar ha trasladado el segmento del índice a otro tablespace ¿Se ha logrado alguna compactación?

21) Usar ISQL\*Plus, conectado con usuario SYSTEM, para reconstruir de nuevo el índice CLAVE\_PRIMARIA\_DE\_A en el tablespace PRUEBA (`alter index CLAVE_PRIMARIA_DE_A rebuild tablespace PRUEBA`).

22) Comprobar el contenido y el mapa de extensiones de los tablespaces PRUEBA y PRUEBA\_TEMP. Verificar que el índice CLAVE\_PRIMARIA\_DE\_A sigue marcado como válido.

23) Desasignar, si es menester, el espacio no usado por encima de la marca de marea alta en ambos segmentos.

**Pregunta:** En su caso, cuantifique en bytes el espacio recuperado.