

## ADVERTENCIAS:

- Para el desarrollo de esta práctica se necesitan los contenidos teóricos aportados en los temas 1, 2, 6, 8 y 9.

## PRÁCTICA N° 8 PARTICIONADO DE TABLAS

- 1) Eliminar el usuario ALUMNO creado en la práctica anterior, así como todos sus objetos.
  - Recrear el usuario ALUMNO:
    - asociado al tablespace por defecto PRUEBA y
    - asociado al tablespace temporal TEMP.
  - Concederle los roles
    - CONNECT y
    - RESOURCE.
  - Observar las consecuencias de la creación.
- 2) Usar ISQL\*Plus, conectado con usuario ALUMNO para cargar el complemento 2, observar su contenido reiteradamente hasta su comprensión y, a continuación, ejecutarlo. Emplear el procedimiento inserta\_en\_b para insertar 3000 elementos. Revisar el contenido de la tabla observando el funcionamiento de las secuencias.
- 3) Eliminar la tabla B así como las secuencias; para ello emplear el complemento 1. A continuación recrear la situación anterior pero con 400.000 tuplas.
- 4) A partir de este punto, van a manejar los siguientes apartados de los siguientes manuales:
  - Data Warehousing Guide
    - desde 5.1(pág. 59) Types of Partitioning
    - hasta 5.10(pág. 68) (Partitioning and Table Compression **NO**)
  - Concepts
    - desde 18.1(pág. 327) Introduction to Partitioning
    - hasta 18.8(pág. 334) (Overview of Partitioned Indexes **NO**)
  - SQL Reference: Sintaxis y semántica de todo lo relativo al particionado de tablas en las sentencias
    - CREATE TABLE y
    - ALTER TABLE
- 5) Usar ISQL\*Plus, conectado con usuario SYSTEM para, aplicando los conocimientos de redefinición de tablas alcanzados en la práctica anterior y los conocimientos de particionado recién adquiridos, redefinir adecuadamente la tabla B como una tabla con cuatro particiones definidas en rango de la siguiente forma:
  - Partición 'añoencurso':
    - tuplas con fecha del año actual o posteriores, en el tablespace prueba\_temp.
  - Partición 'añoyyyy' (siendo yyyy el año pasado):
    - tuplas con fecha del año pasado, en el tablespace prueba.
  - Partición 'añoyyyy' (siendo yyyy el año ante-pasado):

- tuplas con fecha del año ante-pasado, en el tablespace prueba.
- Partición ‘restoaños’:
  - tuplas con el resto de fechas, en el tablespace prueba\_temp.

Tenga en cuenta que el tablespace de cada partición se encuentra en la tabla del diccionario *user\_tabs\_partitions*. Por otra parte, el tablespace por defecto de cada tabla particionada se encuentra en la tabla del diccionario *user\_part\_tables*. Las particiones en las que no se especifique tablespace residirán en PRUEBA, que es el tablespace por defecto de la tabla (en este caso coincide con el tablespace por defecto del propietario, dado que no se especificó ningún tablespace por defecto para la tabla).

**Pregunta:** Enumere y explique la secuencia completa de instrucciones que debería ejecutar para conseguir el objetivo fijado. Si aparecieran errores, remítase al siguiente apartado de la práctica.

- 6) Corregir el error haciendo autoextensible el archivo de datos prueba01 del tablespace prueba.
- 7) Borrar la tabla provisional con purge y ver el desequilibrio de las particiones.
- 8) Si se desea añadir una partición al rango de particiones, solamente es posible tras la última. No es posible, en este caso, porque la última ha sido creada con maxvalue.
  - Ventajas de maxvalue:
    - acoge al null y
    - evita valores no aceptados en ninguna partición.
  - Desventajas de maxvalue:
    - impide el add partition.

Una posible solución para disminuir el desequilibrio es partirlas.

#### **Preguntas:**

1. ¿Qué instrucción SQL emplearía para dividir la última partición en dos llamadas ‘primer\_semestre’ y ‘resto\_añoencurso’?
  2. Para el caso de la partición ‘restoaños’, ¿qué instrucción SQL emplearía para dividirla en ‘resto\_siglos’ y ‘siglo\_XXI’?
  3. Describa con exactitud el contenido de estas cuatro nuevas particiones.
- 9) Recrear la situación resultante de los puntos 3, 5 y 7 y pasar a la siguiente configuración:
    - Una tabla ‘añoencurso’ con la misma estructura relacional que la tabla ‘B’ (campos, tipos y restricciones), pero sin particionar y con los datos de la partición homónima de la tabla ‘B’ y en el mismo tablespace.
    - Una tabla ‘B’ como la resultante del punto 7, pero sin la última partición.

Tenga en cuenta que la restricción de la clave primaria de la tabla ‘B’ está en estado no validada porque el procedimiento de copia de los objetos dependientes la crea así en la tabla interina, dado que de esta forma evita tener que validar todos los datos que el procedimiento de inicio de la redefinición ha depositado en la tabla interina; la finalización de la redefinición causa, al intercambiar, que la restricción de la clave primaria de la tabla ‘B’ quede no validada.

**Pregunta:** Enumere y explique la secuencia completa de instrucciones SQL que debería ejecutar para conseguir el objetivo fijado.

- 10) Suponer que es el último día del año en curso, son las 23:30 y en lo que queda de año no van a ocurrir nuevas modificaciones en la tabla 'añoencurso'. Se desea archivar en la tabla 'B' lo referente al año que termina, creando para ello una nueva partición y, al mismo tiempo, preparar la tabla 'añoencurso' para el nuevo año.

Esta operación constituye un ejemplo de lo que se suele denominar una ventana rodante (rolling window), de tiempo en este caso, sobre datos en línea (OLTP) que han de ser trasladados a bajo coste al almacén histórico (OLAP, Data Warehousing), para ser archivados mientras aumentan continuamente. La ventaja de la estructura de la ventana de tiempo radica en que la tabla 'B' puede primar OLAP (Data Warehousing) en su estructura física, mientras que la tabla 'añoencurso' puede primar OLTP en su estructura física. El bajo coste en el intercambio lo proporciona la operación *exchange*, que puede llegar a consistir tan solo en actualizaciones al diccionario en determinadas circunstancias; véase la nota incluida en la página 17-33 del manual 'Administrator Guide', capítulo '17 Managing Partitioned Tables and Indexes' en el apartado 'Exchanging Partitions', se puede ver en:

[https://docs.oracle.com/cd/B19306\\_01/server.102/b14231/partiti.htm#i1107555](https://docs.oracle.com/cd/B19306_01/server.102/b14231/partiti.htm#i1107555)

**Pregunta:** Enumere y explique la secuencia completa de instrucciones SQL que debería ejecutar para conseguir el objetivo fijado.

- 11) Recrear la situación resultante del punto 3.
- 12) Usar ISQL\*Plus, conectado con usuario SYSTEM para, aplicando los conocimientos de redefinición de tablas adquiridos en la práctica anterior y los conocimientos de particionado recién estudiados, redefinir adecuadamente la tabla B como una tabla con cuatro particiones mediante hash, atendiendo al campo fecha, de la forma:
- Partición 1 en el tablespace prueba\_temp.
  - Partición 2 en el tablespace prueba.
  - Partición 3 en el tablespace prueba.
  - Partición 4 en el tablespace prueba\_temp.

**Pregunta:** Enumere y explique la secuencia completa de instrucciones SQL que ejecuta para conseguir el objetivo fijado.

- 13) Comprobar que ya no hay desequilibrios en el espacio asignado.

**Preguntas:**

1. A pesar de no haber desequilibrios en el espacio asignado, ¿hay diferencias en la ocupación de las filas de las particiones?
  2. ¿Cuáles son las dos diferencias fundamentales entre la partición por hash y la partición por rango con respecto a la optimización?
- 14) Eliminar completamente la tabla B y reiniciar las secuencias. Crear de nuevo la tabla B, directamente particionada por hash como se indica en el punto anterior. Emplear

el procedimiento `inserta_en_b` para insertar 400.000 elementos. Comprometer la transacción.

**Pregunta:** Enumere y explique la secuencia completa de instrucciones SQL que ejecuta para conseguir el objetivo fijado.

15) Comprobar que no hay desequilibrios.

16) Eliminar completamente la tabla B y reiniciar las secuencias.

Crear de nuevo la tabla B, directamente particionada por listas de la siguiente forma:

- ‘Vocales’ con aquellas tuplas con valor del campo b (‘A’, ‘E’, ‘I’, ‘O’, ‘U’) en el tablespace prueba.
- ‘Consonantes’ con aquellas tuplas con el resto de valores en el tablespace prueba\_temp.

Emplear el procedimiento `inserta_en_b` para insertar 400.000 elementos. Comprometer la transacción.

**Pregunta:** Enumere y explique la secuencia completa de instrucciones SQL que ejecuta para conseguir el objetivo fijado.

17) Comprobar la situación que se obtiene.

**Pregunta:** Comentar cómo ha resultado el equilibrado entre las particiones y la ubicación de cada una de ellas.

En cuanto a la poda de particiones, según el método de particionado, el optimizador puede realizar poda de particiones para las búsquedas por la clave de particionado mediante disyunción (OR) de

- En el particionado por rango y en el particionado por listas: búsquedas exactas y en rango.
- En el particionado por hash: búsquedas exactas.