

Bases de Datos

Trabajo Práctico N°2

Fecha de Entrega: 15/11/2013

Fecha de Recuperatorio: 11/12/2013

Objetivos

- Evaluar el impacto de la utilización de diferentes estrategias de manejo de memoria en la implementación de ciertas operaciones sobre una BD.
- Analizar la importancia del módulo Buffer Manager en el contexto de un motor de BD.

Enunciado

El **Buffer Manager** es uno de los componentes más importantes dentro de un motor de BD. Su principal función es administrar un espacio de memoria de la BD, utilizado como una especie de memoria *caché*. El objetivo es que las diferentes aplicaciones que usan la BD y requieren páginas de disco, puedan recuperar la página de este espacio de memoria y accedan lo menos posible al disco.

El espacio de memoria administrado por el *Buffer Manager* puede ser organizado de diferentes formas y la estrategia para decidir cuál página reemplazar cuando ya no queda más espacio también puede variar.

1. Investigar estrategia de reemplazo de páginas de Oracle

El algoritmo que utiliza Oracle para manejar las páginas del Buffer Pool es conocido como *"Touch Count"* y es una variante del popular *LRU*.

En este punto deberán realizar un estudio sobre este algoritmo, describiendo su funcionamiento, la problemática que intenta resolver y sus ventajas/desventajas con respecto a algoritmos clásicos como LRU y MRU.

El análisis debe basarse en la última versión del motor: Oracle 11g. Es importante detallar las referencias de donde extraen la información.

2. Implementar estrategias de reemplazo de páginas

Aquí deberán incorporar al motor **UBADB** los siguientes algoritmos:

- **LRU**: se remueve la página de fecha de referencia más antigua.
- **MRU**: opuesto a LRU, se remueve la página de fecha de referencia más reciente.
- **Touch Count**: podrán basarse en el paper *"All About Oracle's Touch Count Algorithm"* (Shallahamer, 2004).

Consideraremos que una página es referenciada cuando se realiza un *pin* o un *unpin* sobre ella.

Su implementación deberá contener **tests de unidad** (usando *JUnit*).

3. Evaluación de estrategia “Touch Count”

El objetivo de este punto es comparar el desempeño de las estrategias implementadas en el punto anterior.

La comparación de resultados debe considerar:

- *Hit rate (HR)*: # páginas encontradas en memoria / # páginas solicitadas
- *Hit rate with Memory (HRM)*: # páginas encontradas en memoria / # páginas solicitadas que podrían haber sido hits

La diferencia entre estas dos métricas radica en que la segunda no considera como miss a la primera solicitud de una página (esta jamás podría haber sido hit pues nunca se había solicitado con anterioridad).

A los efectos del TP, vamos a considerar el primer pedido desde que se levantó la base, aunque en el mundo real se establece un lapso de tiempo para definir el “inicio” (hay bases de datos que no se bajan nunca o casi nunca).

Se pide:

- Mostrar gráficamente los resultados de la eficiencia (*hit rate*) para buffers pequeños y de mayor capacidad, con distintas trazas usando cada una de las estrategias.
- Realizar una tabla comparativa de eficiencia de las distintas estrategias en los diferentes contextos.
- Presentar un análisis sobre los resultados y sus conclusiones.

Para definir cada escenario pueden hacer uso de una herramienta para generar trazas dentro de *UBADB*.

Consideraciones

Para la corrección no sólo se evaluará que lo implementado funcione correctamente sino también la calidad del código generado. Con calidad nos referimos a usar comentarios, nombre declarativo para las variables o métodos, uso de métodos auxiliares, evitar duplicar código, etc. La idea es seguir utilizando este código en futuros cuatrimestres, por lo que es importante que traten de seguir las convenciones utilizadas a lo largo del código.

Entrega

La entrega deberá contar con el código que implementa lo pedido, un breve informe con:

- Detalles de implementación
- Decisiones tomadas
- Conclusiones

En los laboratorios, se llevará a cabo una demo con su tutor asignado.

Bibliografía

- Libro “Database Management Systems”, Raghu Ramakrishnan – Johannes Gehrke, 2º Edición, Capítulo 7.
- Paper “Principles of Database Buffer Management”, Effelsberg & Haerder, 1984.
- Paper “All About Oracle’s Touch Count Algorithm”, Shallahamer, 2004.