FICHEROS Y BASES DE DATOS (E44)

3º INGENIERÍA EN INFORMÁTICA

Tema 4.

Técnicas de Dispersión.

Definición y Manejo.

1.- Introducción.

2.- Funciones de Dispersión.

3.- Saturación Progresiva.

4.- Empaquetado de Registros.

5.- Otras Técnicas de Dispersión.

6.- Dispersión Extensible.

(Capítulos 10 y 11 del Folk)

(Capítulos 4 y 5 del Elmasri)

(Capítulo 3 del Date)

INTRODUCCIÓN

Concepto de Dispersión

- Algunas aplicaciones requieren que el número de accesos a disco necesario para recuperar información se reduzca a uno.

- Dado que el manejo de los índices multinivel puede tener un coste mayor, se deben definir estructuras de ficheros alternativas.

- La Dispersión permite definir la posición que ocupará un registro en el fichero, mediante la aplicación de una Función de Dispersión sobre la Clave de Acceso.

- Esta posición se utiliza en las operaciones de inserción y borrado, por lo que el número de accesos se reduce a uno.

- Un registro puede almacenarse en cualquier posición dentro del fichero, según la función de dispersión que se utilice.

- Por esta razón, esta opción sólo se puede utilizar en ficheros con registros de tamaño fijo.

- La dispersión no admite la utilización de claves repetidas, por lo que normalmente se aplica sobre la clave primaria.

- La aplicación de la función de dispersión genera un orden entre los registros, pero sólo respecto de la clave utilizada.

- Por lo tanto el acceso respecto de cualquier otra clave, requiere la utilización de otras estructuras de ficheros como los índices.

INTRODUCCIÓN

Colisiones

- El problema se presenta cuando más de un registro se asocia a la misma posición, es decir, cuando aparecen las Colisiones.
- Las claves que se asocian a la misma posición se les denomina Sinónimos.
- La aparición de colisiones puede aumentar el número de accesos requerido para acceder a un registro determinado.
- Para reducir el problema se puede,
  - Utilizar una Función de Dispersión Perfecta, que reduzca al máximo su número.
  - Definir variantes que reduzcan el número de colisiones que puedan aparecer.
- La primera opción es demasiado costosa, ya que existe una gran variedad de funciones, y además se requiere conocer las claves.
- Por lo tanto, el estudio se centra en la gestión de las colisiones, que dará lugar a diferentes variantes de esta técnica.
- Las opciones más comunes son las siguientes,
  - Definir una función de dispersión que realice una distribución adecuada de los registros.
  - Aumentar el tamaño del fichero para reducir la posibilidad de aparición de las colisiones.
  - Almacenar más de un registro en la misma posición, que referenciará a una página.

FUNCIONES DE DISPERSIÓN

Ejemplos

- Una función de dispersión se podría describir como sigue,
  - Se calcula el producto de la representación en ASCII de los dos primeros carácteres de la clave.
  - La dirección del registro se obtiene a partir de los últimos dígitos del resultado.
- Esta función puede producir un alto grado de colisiones, y además produce un bajo nivel de distribución de los registros.
- Una función que presenta mejores resultados es la siguiente,
  - Se obtiene la representación numérica de la clave.
  - Dicha representación se divide en partes, que deben de ser sumadas.
  - El valor acumulado no debe sobrepasar el máximo número entero representable.
  - El resultado se divide por un número que se relaciona con el tamaño del fichero, siendo el resto la dirección del registro.
- La principal cuestión es la elección del divisor,
  - Normalmente se utiliza un número primo o bien, un número con divisores mayores de 20.
  - Este número debe de ser mayor que el tamaño del fichero, y redefinirá dicho valor.

FUNCIONES DE DISPERSIÓN

Clasificación y Análisis

- Las funciones de dispersión se puede clasificar a partir de la probabilidad de asignación de un registro en una dirección como,
  - Uniforme, en las que los registros aparecen uniformemente distribuidos en el fichero.
  - Aleatoria, si un determinado registro puede ser asignado a cualquier dirección.
- La primera opción suele ser muy costosa, mientras que no existe ningún tipo de control sobre la segunda.
- Es por ello, que se estudian alternativas que permitan mejorar el comportamiento de estas últimas,
  - Buscar un patrón en la clave.
  - Particionar la clave y luego sumar las partes.
  - Dividir la clave por un número.
- Cuando estas alternativas no son útiles, se pueden analizar otras opciones,
  - Elevar la clave al cuadrado, y elegir como dirección los dígitos centrales.
  - Cambiar la base de la clave, y dividir el resultado por el tamaño del fichero.
- La primera opción requiere la utilización de aritmética específica, aunque suele dar un resultado adecuado para ciertas claves.

SATURACIÓN PROGRESIVA

Definición

- Una de las opciones más simples para resolver la aparición de una colisión en una dirección, es elegir el primer hueco no ocupado que se encuentre a continuación de aquél.
- Si se alcanza el final del fichero en este proceso, se debe continuar con la primera dirección del fichero.
- Esta técnica se denomina Verificación Lineal o Saturación Progresiva.
- La búsqueda de información también se inicia en la dirección asociada a la clave.
- Si el registro buscado aparece en el fichero, esta operación finaliza cuando se encuentra el registro con la clave asociada.
- En caso contrario, el proceso finaliza cuando se produce una de las siguientes situaciones,
  - Aparece un hueco no ocupado.
  - Se alcanza la dirección inicial.
- Mediante esta técnica, el número de accesos requeridos para acceder a un registro crece de modo ostensible al aumentar la densidad de registros en el fichero.
- Por esta razón, esta técnica sólo se debe utilizar cuando la densidad de registros resulta suficientemente baja, alrededor del 40%.

SATURACIÓN PROGRESIVA

Eliminación de Registros

- En la eliminación de un registro del fichero se debe de considerar que,
  - El registro pueda ser reutilizado.
  - El borrado no impida la localización de un registro.
- La primera condición se puede conseguir de modo sencillo marcando el hueco asociado como vacío, es decir, borrando el registro.
- Para impedir que un borrado interrumpa la búsqueda de un registro insertado, se debe de marcar de un modo especial.
- Esta marca indicará al proceso de búsqueda de un registro que el hueco correspondiente estuvo ocupado y se ha borrado.
- Por lo que respecta al proceso de inserción, lo identificará como un hueco útil.
- Este proceso puede eliminar la posibilidad de detectar la aparición de duplicados, ya que es posible reutilizar un hueco situado en una posición anterior.
- Las eliminaciones pueden aumentar de modo innecesario el número de accesos necesarios para acceder a un registro, para resolverlo,
  - Recolocar los sinónimos tras el borrado.
  - Reorganizar el fichero periódicamente.
  - Utilizar otra metodología

EMPAQUETADO DE REGISTROS

Frecuencia de Colisiones

- La Densidad de Registros en un fichero se define como el cociente entre el número de registros almacenados y el número de registros que caben en un fichero.

- La conexión existente entre la densidad de registros y el número de colisiones que pueden aparecer resulta muy compleja.

- Únicamente cuando se utiliza una función de dispersión aleatoria se puede realizar un buen análisis, mediante un estudio estadístico.

- La probabilidad de aparición de x registros asociados a una dirección en un fichero con una capacidad de N huecos y con r registros, se define por la función de Poisson,

$$p(x) = \frac{(r/N)^x e^{-r/N}}{x!}$$

- Siendo n la capacidad en registros del fichero, el número de colisiones se calcula como,

$$num\_colis = \sum_{x=2}^{n} n(x-1)p(x)$$

- El análisis de estas expresiones permite evaluar el alto número de colisiones que se pueden producir, y que crece al aumentar la densidad.

- Es por ello, que resulta interesante evaluar alguna opción que permita reducir el número de colisiones, y que permita aumentar la densidad del fichero.

EMPAQUETADO DE REGISTROS

Definición

- Siendo b el número de registros en un hueco, la relación entre el número de huecos y el número de registros almacenado se define,

$$n = bN$$

- Para un mismo valor de densidad, el número de colisiones disminuye cuando crece b,

$$num\_colis = \sum_{x=b+1}^{n} n(x-b)p(x)$$

- Este es el fundamento para el Empaquetado de Registros.

- Uno de los aspectos básicos es su tamaño, ya que el uso de dimensiones muy grandes puede aumentar el coste de acceso a disco.

- Normalmente se elige un tamaño intermedio como un cúmulo.

- Cuando el valor de b es mayor que 1, se debe introducir un contador que indique el número de registros ocupados en la página y marcar los registros que no han sido utilizados.

- De este modo, el algoritmo de inserción puede detectar si caben registros en la página y cuáles son.

- La eliminación puede realizarse mediante la utilización de la saturación progresiva, aunque la gestión puede complicarse.

- El algoritmo de inserción debe diferenciar las marcas de registros no utilizado y de registros reutilizable.

OTRAS TÉCNICAS DE RESOLUCIÓN DE COLISIONES

Definición

- Todas las técnicas que se describen tienen como objetivo reducir los problemas relativos a la aparición de colisiones.

- En la Doble Dispersión, si aparece una colisión se aplica una segunda función de dispersión, da como resultado un número, c, que no tiene ningún divisor común con N.

- La nueva dirección se obtiene sumando el valor c a la dirección original, hasta encontrar un hueco vacío.

- En este sistema los sinónimos pueden estar muy separados, aumentando su coste de acceso.

- La Saturación Progresiva Enlazada forma una lista con los sinónimos, reduciendo el coste de acceso a éstos, pero con inconvenientes,

  - Cada registro debe de poseer un campo en el que se almacene una dirección,

  - Toda dirección debe contener un registro con esa dirección, no puede contener un sinónimo de otra dirección.

- Para resolver los problemas, se puede enlazar con un Área de Saturación Separada.

- Por su parte las Tablas de Dispersión se aplican sobre índices, lo que permite manejar registros de tamaño variable.

DISPERSIÓN EXTENSIBLE

Planteamiento

- Las técnicas de dispersión descritas fijan el tamaño del fichero, además deben de reducir el número de colisiones, limitando la densidad de registros en el fichero.

- Cuando no se conoce el tamaño del fichero, y se desea eliminar la influencia de las colisiones, es necesario utilizar otras técnicas.

- Como en el caso de los índices, la solución se basa en la utilización de un árbol binario, pero su implementación difiere sustancialmente.

- En este caso, las hojas del árbol aparecen en un vector, el Directorio, y referencian a una página.

- El directorio se puede almacenar en memoria, reduciendo a uno el número de accesos.

- La función de dispersión es similar a las que se han descritos, pero en este caso no se calcula la división por el tamaño del fichero.

- El valor obtenido en la partición y acumulación de la clave es la Dirección del registro.

- Los primeros bits de esta dirección indican la página que almacenará el registro.

- El objetivo es limitar el número de páginas que ocupa el fichero, por lo que la misma página puede ser referenciada por varias hojas.

- Por su parte, la altura del árbol varía de modo dinámico, en función del número de registros que se vayan insertando.

DISPERSIÓN EXTENSIBLE

Desarrollo

- El directorio se compone de un Registro de Cabecera en el que se almacena la Altura del Árbol, más una entrada para cada una de las hojas del árbol.

- La altura indica el número de bits de la dirección que deben ser utilizados para localizar la página asociada a un registro.

- Una página puede ser referenciada por más de una hoja del árbol.

- Por esta razón, en cada página aparece un registro de cabecera que indica el número de bits que son comunes a las direcciones de los registros que contiene la página.

- Cuando una página se encuentra llena y se desea insertar un nuevo registro,

  - Se crean dos nuevas páginas, cuyo valor en el registro de cabecera se toma sumando uno al valor de la página original.

  - Los registros se distribuyen entre las dos páginas, tal que se cumpla que los primeros bits de sus direcciones coincidan.

- Si el valor de cabecera en la página original coincide con el valor en el directorio, es necesario aumentar la altura del árbol.

- El borrado de un registro puede producir la operación inversa, en donde se combinen dos páginas en una única página, y en algún caso se puede reducir la altura del árbol.