
SISINF

ARCHIVOS PERMANENTES

Versión 7.3

Kratos, S.A. de C.V.

La Tecnología en Software.

Derechos Reservados ☐. Prohibida la reproducción total o parcial sin permiso escrito de **KRATOS, S.A. de C.V.** El uso de programas que integran **SISINF** se vende y renta bajo contrato con **KRATOS, S.A. de C.V.**

CONTENIDO

CONTENIDO.....	2
PREFACIO.....	4
1) Conceptos Generales.....	5
1.1) Agrupación de la Información.....	5
1.2) Acceso a la Información.....	5
1.3) Descripción General de las Instrucciones.....	7
2) Instrucciones de Archivos Permanentes.....	8
2.1) Buscar un registro en disco.....	8
2.2) Buscar Recurrente.....	24
2.3) Leer un Archivo Permanente.....	40
2.4) Escribir un Archivo Permanente.....	40
2.5) Borrar un Registro.....	41
2.6) Borrar un recurrente.....	42
2.7) Borrar un archivo.....	42
2.8) Insertar un registro.....	42
2.9) Opción Disco.....	44
3) Archivos Compartidos.....	51
3.1) Conceptos Generales.....	51
3.2) Solución a actualización simultánea.....	55
3.3) Falta de integridad.....	58
4) Implantación de Archivos Permanentes.....	59
4.1) Problemas de compartir archivos.....	59
4.2) Generación de pedidos.....	59
4.3) Estrategia de Tarea Central.....	60
4.4) Archivo en Memoria Virtual Compartida.....	60
4.5) Buffers Locales.....	61
4.6) Estructura de los Archivos Permanentes.....	61
4.7) Errores en Archivos Permanentes.....	67
5) Tiempo Usado.....	70
5.1) Introducción.....	70
5.2) Tiempo de Respuesta.....	71

5.3) Tiempo de Proceso.....73
5.4) Estadísticas de Uso.....75
6) Errores de Ejecución.....77

PREFACIO

En este manual se describen los archivos del banco de datos de **SISINF** llamados Permanentes. El contenido es:

- ◆ En el capítulo 1 se describen los conceptos generales de estos archivos ya descritos en el Manual de DBD y los conceptos aplicables a las instrucciones para su manejo.
- ◆ En el Capítulo 2 se describe las instrucciones para el manejo de estos archivos.
- ◆ En el Capítulo 3 se describe la forma de programar para compartir por varios usuarios la información.
- ◆ En el Capítulo 4 se describe la forma en que se logra la compartición de archivos así como su formato interno.
- ◆ En el Capítulo 5 se discute en forma general los problemas de tiempo de respuesta, que aunque están involucradas todas las instrucciones del lenguaje, las que mas pueden afectar son las de Archivo Permanente.
- ◆ En el Capítulo 6 se describen los diferentes errores de ejecución

1) Conceptos Generales.

1.1) Agrupación de la Información.

Tal como se describió en el manual de DBD lo primero que tiene que hacer el diseñador de la aplicación es distribuir la diferente información que tendrá el sistema en los diferentes Archivos Permanentes que formarán la aplicación.

Así por ejemplo en un sistema comercial, toda la información relacionada con los clientes se tendrá en el archivo CLIENT, la información relacionada a artículos se tendrá en ART, ...

Dentro de cada archivo se deberá también distribuir la información en los diferentes tipos de registros que forman un archivo.

Así para un archivo de clientes, se tendrá en el archivo maestro la información general del mismo como puede ser su número, nombre, dirección, ... Del registro maestro del cliente dependerá la información de sus facturas formando el recurrente de facturas. De este tipo de registro un cliente en específico puede tener cero, una o varias facturas.

En resumen se tiene:

- ◆ Archivo Permanente. Archivo que contiene diferentes tipos de registros con toda la información relacionada.
- ◆ Registro Maestro. Registro con la información general del cual dependerán los registros con la información de detalle.
- ◆ Registro Recurrente. Registro con la información de detalle que puede ocurrir cero, una o más veces y que depende de un registro maestro o de otro recurrente.

1.2) Acceso a la Información.

Otro aspecto importante es como poder acceder la información de un Archivo Permanente y sus diferentes registros maestros y recurrentes.

El lenguaje tiene instrucciones para que una vez posicionado en un cierto

- 1.2) Acceso a la Información.

registro maestro se puede acceder tanto a la información del registro maestro como a la de todos los recurrentes que dependen de ese registro maestro.

Es importante señalar que solo se puede tener en un momento dado uno solo de cada tipo de registro (maestro o recurrente) disponible para acceder a su información. Así se puede tener un solo registro maestro del archivo clientes perteneciente a un cliente, de ese cliente solo una factura, ... en un momento dado, etc.

Ahora el problema está en cómo llegar a cada registro maestro. Para ello se tiene el concepto de llaves, es decir, variables que conociendo su valor permiten llegar al resto de la información. Así por ejemplo si el número de cliente es llave, conociendo un número específico, permitirá llegar al resto de la información.

El lenguaje permite llegar a los diferentes registros maestros mediante instrucciones que hacen uso de las llaves. Un caso especial es cuando la llave pertenece a un recurrente, permitiendo llegar tanto al recurrente como al registro maestro.

Una o varias variables forman un Grupo de llaves permitiendo el acceso de la información por dicho grupo de llaves. La llave formada debe ser única. Un archivo puede tener cero, uno o varios grupos de llaves.

Los grupos de llaves permiten varios tipos de búsqueda, los cuales se resumen en:

- ◆ **PRINCIPAL** Conociendo el valor de las variables llave del grupo se llega a la información.
- ◆ **SECUENCIAL** Se accede registro por registro en el orden que definen las llaves del grupo.
- ◆ **NO-PRINCIPAL** Se acceden los registros que tengan igual cierta variable llave del grupo de llaves.

Una vez que se ha llegado al registro maestro se tienen instrucciones para acceder los dependientes, o sea, los recurrentes. Estos accesos pueden ser de primero insertado a último insertado o bien de último insertado a primero insertado. También se tiene una opción de acceso por un ordenamiento específico dependiendo de la información de los mismos.

1.3) Descripción General de las Instrucciones.

Como ya se comentó, es necesario llegar a los registros para poder acceder la información tanto del registro maestro como de sus recurrentes. Para ello se dispone de dos instrucciones.

BUSCAR DISCO

BUSCAR RECURRENTE

Con la primera de ellas se accesa el registro maestro por sus llaves, con la segunda sus recurrentes. Una vez que un registro está accesado se encuentra disponible para ser procesado. Un caso especial es que la llave sea de un recurrente en cuyo caso se accesa el recurrente y todos los ascendientes hasta el registro maestro de dicho recurrente.

Una vez que un registro está disponible para ser procesado se puede leer la información de cada uno de sus campos, modificar la información o bien borrar el registro. Las instrucciones para hacerlo son:

LEER DISCO

ESCRIBIR DISCO

BORRAR DISCO

BORRAR RECURRENTE

Cuando la información NO existe en el banco de datos se puede insertar mediante la instrucción.

INSERTAR DISCO

Por último se tiene una instrucción de uso múltiple con diversas finalidades como es posicionar un grupo de llaves al inicio, un recurrente, ... dicha instrucción:

OPCION DISCO

Hasta aquí una descripción general de las instrucciones

2) Instrucciones de Archivos Permanentes.

2.1) Buscar un registro en disco.

Esta instrucción sirve para buscar un registro maestro o un recurrente en un cierto grupo de llaves.

El formato general de la instrucción es:

BUSCAR DISCO NOMB **EJECUTA #**

PRIN-Gx o **NO-PRIN-Gx** o **SEC-Gx** o **APROX-Gx** o **BUSC-Gx** u **ORD.** o **ULTIMO-Gx**

VAR-L VAR-L ...

En donde:

- ◆ NOMB es el nombre del archivo permanente y es el mismo que se usó en la definición de archivo en el módulo DBD.
- ◆ Si no se encuentra el registro buscado, se ejecuta la instrucción con etiqueta #.
- ◆ Se busca por alguna de las opciones listadas, es decir, por PRIN-Gx, NO-PRIN-Gx, SEC-Gx, etc.
- ◆ La letra x es el número del grupo de llaves con el que se busca.
- ◆ En las opciones de PRIN-Gx, NO-PRIN-Gx, APROX-Gx y BUSC-Gx es necesario especificar las variables llave (VAR-L) con cuyo contenido se hará la búsqueda. Si dichas variables son alfanuméricas, el orden para hacer la comparación dependerá del modelo y marca de la computadora.

Muchas instrucciones de **SISINF** trabajan de forma que la solución a problemas complejos se hace a través de opciones. Así por ejemplo, si se desea imprimir información en una línea se usará la instrucción de IMPRIMIR DETALLE, pero si se desea imprimir información en dos líneas y que ambas queden en la misma hoja, se usará la misma instrucción pero con la opción /.

En las instrucciones de BUSCAR DISCO con las **mismas** opciones se pueden solucionar diferentes problemas de programación, desde muy simples hasta complejos. Por lo anterior, se discute a continuación el uso de algunas de

estas opciones para los casos más comunes de grupos de llaves en el registro maestro:

- ◆ Cuando se conoce la llave de un registro y se desea buscar, se usará la opción PRIN-Gx. Es decir, la opción PRIN-Gx busca en el directorio indicado por el primer registro que cumpla por igual con las llaves especificadas.
- ◆ Cuando se desea buscar **todos** los registros que tengan igual una llave (o unas llaves), se usará la opción NO-PRIN-Gx. Es decir, la opción NO-PRIN-Gx busca la **primera vez** por el primer registro que cumpla por igual con las llaves especificadas; la **segunda vez** se buscará el siguiente registro que cumpla por igual con las llaves especificadas, etc.
- ◆ Cuando se desea buscar todos los registros de un grupo de llaves se usará la opción SEC-Gx. Es decir, la opción SEC-Gx busca la **primera vez** el primer registro, la segunda vez el segundo, etc. el orden con que se buscan los registros lo definen las llaves del grupo.
- ◆ Cuando se desea buscar el último registro de un grupo de llaves se debe especificar **ULTIMO-Gx** sin llaves. Cuando se especifican llaves se buscará el último que cumpla con igual a las llaves especificadas, si no existe un registro con esas llaves, se posicionará en el último registro con llaves inferiores al buscado.
- ◆ En la opción PRIN-Gx, si no se encuentra el registro buscado, se ejecuta la instrucción con etiqueta #.
- ◆ En las opciones NO-PRIN-Gx y SEC-Gx, si **ya no hay** mas registros se ejecuta la instrucción con etiqueta #.
- ◆ Cuando se especifica ULTIMO-Gx sin llaves y no hay información en el archivo se ejecuta la instrucción con etiqueta #, lo mismo sucede cuando se especifican llaves y son menores al primer registro.
- ◆ En estas opciones se busca el registro y se pasa la información del mismo, de disco a memoria, de forma que está **disponible** para ser procesado y que se pueda leer (LEER DISCO), modificar (ESCRIBIR DISCO), etc.

El siguiente ejemplo ilustra el uso de esta instrucción para las opciones descritas:

a) En DBD se define:

```
ARCHIVO CLTES 2 2
```

- 2.1) Buscar un registro en disco.

```

NUM-CLI  NUMERO DE CLIENTE' * R0 LL1 N1 0
NOM-CLI  'NOMBRE DEL CLIENTE' * R0 S   X 30
ZONA     'ZONA DEL CLIENTE'      * R0 LL1 NI 0

```

b) En un programa en donde se desee imprimir todo el archivo en orden secuencial por el grupo de llaves 1 se pondría:

```
PROGRAMA 'LISTADO SECUENCIAL'
```

```
_____
```

```
_____
```

```
90  BUSCAR DISCO CLTES EJECUTA 100 SEG-G1
```

```
_____ instrucciones para leer el registro e
```

```
_____ imprimirlo
```

```
EJECUTA 90
```

```
FIN
```

c) En un programa en que se desee sacar información para un número de cliente (al azar) y escriba en la pantalla se pondría:

```
PROGRAMA 'CONSULTAS A CLIENTES'
```

```
_____
```

```
_____
```

```
50  LEER TERMINAL EJECUTA 100 NUM-CLI
```

```
BUSCAR DISCO CLTES EJECUTA 90 PRIN-G1 NUM-CLI
```

```
_____ instrucciones para leer y escribir en la
```

```
_____ pantalla
```

```
_____
```

```
EJECUTA 50
```

```
90  ERROR 'CLIENTE NO EXISTE'  NUM-CLI
```

```
EJECUTA 50
```

```
100 FIN
```

d) En un programa en donde se desee imprimir la información de los clientes por zona (suponiendo que son 10 zonas)

```
PROGRAMA 'LISTADO POR ZONA'
```

```
_____
```

```
_____
```

```

ZONA = 1
50  BUSCAR DISCO CLTES EJECUTA 100 NO-PRIN-G1 ZONA
_____ instrucciones para leer el registro
_____ e imprimirlo
_____
EJECUTA 50
100 ZONA = ZONA + 1
SI ZONA <= 10 EJECUTA 50
FIN

```

e) En un programa donde se insertan nuevos clientes:

```

PROGRAMA 'Alta Cliente'
_____
_____
NUM-CLI = 1
BUSCAR DISCO CLTES EJECUTA 10 ULTIMO-G1
LEER DISCO CLTES NUM-CLI
NUM-CLI = NUM-CLI + 1
10 _____
_____ instrucciones para insertar
_____

```

Cuando se usa la opción **PRIN-Gx** en un grupo de llaves que sólo tiene una variable, se buscará el registro que tiene igual dicha llave (sólo puede haber uno), pero si se tienen varias variables llave en un grupo, se buscará el **primero que cumpla igual con las llaves que se especifican**.

Para aclarar más este concepto suponga que se tiene un grupo de llaves 1 con 3 variables llave llamadas A, B y C cuyos valores son:

Registro número	Llave A	Llave B	Llave C
1	1	5	2
2	1	6	2
3	1	6	3
4	2	5	2
5	2	8	3

6	2	9	2
7	2	9	3
8	2	10	2
9	2	11	4
10	4	5	1
.....			
1000	100	1	2

La siguiente tabla ilustra el registro buscado cuando se usa la instrucción de BUSCAR DISCO, opción PRIN-G1 y las llaves indicadas:

Caso	Llaves			Registro buscado
1	A = 2			4
2	A = 1	B = 6		2
3*	A = 2	C = 4		9
4*	B = 1	C = 2		1000
5	A = 2	B = 9	C = 2	6

Dado que no se permiten llaves duplicadas, cuando se especifican todas, sólo se podrá tener un registro que cumpla con ellas. Tal sería el caso 5.

En los casos 1 a 4 se buscó el primer registro que cumpla con igual en las llaves especificadas.

En el caso 1, aunque hay varios registros con la llave A = 2, se buscó el primero que cumplió por igual en la llave especificada, o sea, el 4. En el caso 2 de la misma forma, aunque hay varios con A = 1, B = 6, se buscó el primero que cumplió por igual en las dos llaves especificadas, o sea, el registro 2.

(*) La instrucción de BUSCAR DISCO con opción PRIN-Gx se puede usar en forma eficiente o en forma ineficiente. Su uso en forma eficiente es cuando no se brincan llaves, tal sería el caso 1 en que se buscó por la primera llave, el caso 2 en que se buscó por la primera y la segunda, o el caso 5 en que se buscó por todas. Su uso en forma ineficiente es cuando se brincan llaves, tal sería el caso 3 en que se buscó por la primera y tercer llaves brincando la segunda, o el caso 4 en que se buscó por la segunda y la tercera, brincando la primera.

En el caso 3 la forma interna de trabajo de esta instrucción sería buscar el primer registro con llave A = 2, o sea, el 4 y a partir de éste buscar en los directorios en forma secuencial aquel registro que tenga C = 4, que es el 9.

El uso de la instrucción de **BUSCAR DISCO** con opción **PRIN-Gx** en situaciones parecidas a este caso puede ser muy ineficiente si la búsqueda secuencial en el directorio es a través de muchos registros. Si es necesario brincar claves se deberá revisar esta situación.

El caso 4 es el más ineficiente, ya que es necesario buscar en forma secuencial en el directorio de dicho grupo de llaves desde el primer registro. No se recomienda el uso de esta instrucción en casos parecidos a éste por ser muy ineficiente.

El uso de esta instrucción en forma eficiente origina que el número de lecturas a disco sea normal y que por lo tanto los tiempos de respuesta o de proceso sean aceptables. Su uso en forma ineficiente origina muchas lecturas a disco y que por lo tanto los tiempos de respuesta o de proceso sean inaceptables.

A continuación se discute el uso de **ULTIMO-Gx** con los mismos valores que se usaron en el ejemplo anterior.

Caso	Llaves	Registro Buscado
1	Sin llaves	1000
2	A = 1	3
3	A = 2 B = 20	9
4	A = 2 B = 11 C = 2	8
5	A = 3 B = 20	9
6	B = 11 (Error de Compilación)	
7	A = 0 Salida por ejecuta	

El caso 1 de NO especificar llaves se posiciona en el último registro. En el caso 2 se encuentra el último con A = 1. En el caso 3 aunque el registro 9 tiene B = 11, es el último más cercano. Al igual en los casos 4 y 5 en que se encuentra el último más cercano. El caso 6 no es posible ya que NO se permite brincar llaves. El caso 7 es cuando la llave es menor al primer registro, se sale por la ejecuta.

A continuación se discute el uso de la opción **APROX-Gx**. Esta opción busca en el directorio indicado el primer registro que cumpla por igual a las llaves especificadas, y si no se encuentra dicho registro, el buscado será el siguiente mayor.

Para aclarar más la opción **APROX-Gx** suponga que se tienen los siguientes registros y los valores de sus llaves:

- 2.1) Buscar un registro en disco.

Registro	Llave
1	10
2	20
3	25
4	60
5	80

Si se busca con llave igual a 10 se tendrá el registro 1; con llave igual a 25 se tendrá el registro 3, con llave igual a 26 se tendrá el registro 4, ya que como no hay registro con llave 26, el buscado es el siguiente. Si se busca con llave 90 se ejecuta la instrucción con etiqueta # (BUSCAR DISCO NOM EJECUTA #), ya que no hay registro con llave 90 y no hay siguiente tampoco.

La opción **APROX-Gx** trabaja igual que la PRIN-Gx en lo relativo a su uso con varias variables llave. Las recomendaciones hechas en cuanto al uso de esta instrucción en forma eficiente o ineficiente son también aplicables.

A continuación se discute el concepto de **posicionamiento** que ayudará al uso de esta instrucción con opción **NO-PRIN-Gx** y **SEC-Gx** en problemas más complejos.

Cada vez que se ejecuta la instrucción de BUSCAR DISCO con opción PRIN-Gx, No-PRIN-Gx y SEC-Gx, y se localiza el registro buscado, se guarda en tablas internas cuál es posición de dicho registro para el grupo de llaves con que se efectuó la búsqueda, de forma que:

- ♦ La ejecución de otros BUSCAR DISCO con opción SEC-Gx buscará el **siguiente** registro a donde se está posicionando en este grupo de llaves.
- ♦ La ejecución de otro BUSCAR DISCO con opción NO-PRIN-Gx buscará el siguiente registro que cumpla por igual a las llaves especificadas a partir de donde está posicionado en este grupo de llaves.

Para aclarar más este concepto se tiene el siguiente ejemplo, para ello suponga que el registro maestro tiene **dos** grupos de llaves, cada uno con una variable N1 de cero decimales. Suponga que se tienen **5** registros con los siguientes valores:

Número de Registro	Valor llave grupo 1	Valor llave grupo 2
1	10	1
2	15	6
3	25	8

4	30	10
5	50	15

Se ejecutan las siguientes instrucciones:

- a) BUSCAR DISCO con opción PRIN-G1 con valor de la llave igual a 15. Se pasa de disco a memoria el registro número 2 que tiene dicha llave, de forma que está disponible y se puede leer, escribir, etc.
- b) BUSCAR DISCO con opción PRIN-G2 con valor de la lleva igual a 10. Se pasa el disco a memoria el registro número 4 que tiene dicha llave, de forma que está disponible para ser procesado.
- c) BUSCAR DISCO con opción SEC-G1. Se pasa de disco a memoria el siguiente registro del grupo 1, o sea, el 3, de forma que está **disponible** para ser procesado.
- d) BUSCAR DISCO con opción SEC-G2. Se pasa de disco a memoria el siguiente registro del grupo 2, o sea, el 5, de forma que está disponible para ser procesado.

Puestas estas instrucciones en forma de tabla se tendría:

Opción de la instrucción	Posicionamiento	Posicionamiento	Registro en
BUSCAR DISCO	Grupo 1	Grupo 2	memoria
a) PRIN-G1 llave 15	Reg2	Inicio	Reg2
b) PRIN-G2 llave 10	Reg2	Reg4	Reg4
c) SEC-G1	Reg3	Reg4	Reg3
d) SEC-G2	Reg3	Reg5	Reg5

Los siguientes comentarios son aplicables a este ejemplo:

- ◆ Cuando se efectuó la primera instrucción de BUSCAR DISCO (inciso a) se guardó la posición en el grupo de llaves 1, o sea, el registro 2, de forma que cuando se ejecutó esta instrucción con opción SEC-G1 (inciso c) se buscó el siguiente registro al 2, o sea, el 3.
- ◆ De igual forma, cuando se efectuó la segunda instrucción de BUSCAR DISCO (inciso b), se guardó la posición en el grupo de llaves 2, o sea, el registro 4, y cuando se ejecutó esta instrucción con opción SEC-G2 (inciso d) se buscó el siguiente registro al 4, o sea, el 5.
- ◆ Note que sólo se tiene un registro en memoria a la vez, el cual está
- 2.1) Buscar un registro en disco.

disponible para ser procesado, es decir, se puede leer, modificar, borrar, etc.

- ◆ Cuando NO se ha hecho ninguna instrucción de BUSCAR DISCO en un grupo de llaves, éste está posicionado al inicio en dicho grupo.

Por lo descrito hasta el momento la instrucción de BUSCAR DISCO se puede usar para un mismo archivo y en un mismo programa con diferentes opciones. El siguiente ejemplo ilustra lo anterior.

a) En DBD se definió

```
ARCHIVO CLTES 2 2
```

```
-----
```

```
NUM-CLI 'NUMERO DE CLIENTE' * R0 LL1 N1 0
```

```
NOM-CLI 'NOMBRE DEL CLIENTE' * R0 S X 30
```

b) En un programa en donde se desee imprimir el archivo CLTES en orden secuencial por grupo de llaves 1 pero omitiendo del cliente 1000 al 1500 se pondría:

```
PROGRAMA 'LISTA CLIENTES'
```

```
BUSCAR DISCO CLTES EJECUTA 30 SEC-G1
```

```
----- se lee NUM-CLI
```

```
SI NUM-CLI < 1000 EJECUTA 20
```

```
SI NUM-CLI > 1500 EJECUTA 20
```

```
NUM-CLI = 1501
```

```
BUSCAR DISCO CLTES EJECUTA 30 APROX-G1 NUM-CLI
```

```
20 ----- leer la información e imprimirla
```

```
EJECUTA 10
```

```
30 FIN
```

En este ejemplo se hace el BUSCAR DISCO con opción SEC-G1, se lee el número del cliente y se revisa si es menor de 1000 o mayor de 1500. Si es así, se lee la información y se imprime para regresar al BUSCAR DISCO y tomar el siguiente. Cuando se detecta el primer registro cuyo número de cliente sea mayor o igual a 1000 se hace un BUSCAR DISCO con opción APROX-G1 para buscar el cliente con número 1501 o el siguiente mayor (si ya no hay, se termina el programa), se guarda la posición del registro buscado, se lee la información y se imprime y se regresa al BUSCAR DISCO con opción SEC-G1, la cual buscará el siguiente registro a donde está posicionado dicho grupo y continuará

el proceso.

Note que en todos los ejemplos de esta sección en que se ha usado la opción SEC-Gx o NO-PRIN-Gx se asume que el posicionamiento en el grupo de llaves está al INICIO, lo cual se logra en dichos ejemplos en forma automática, si es el primer BUSCAR DISCO que se hace en el archivo.

Se tienen tres formas para que el posicionamiento en un grupo de llaves esté al **inicio** antes de ejecutar la instrucción de BUSCAR DISCO.

- a) Que la instrucción de BUSCAR DISCO sea la primera que se hace en el archivo.
- b) Que en la última ejecución de un BUSCAR DISCO en este grupo de llaves, el registro buscado no se encuentre.
- c) Ejecutar la siguiente instrucción.

OPCION DISCO NOMB INICIO-Gx

En donde NOMB es el nombre del archivo y x es el grupo de llaves. Esta instrucción tiene más opciones por lo que más adelante se hace una descripción completa.

Una vez que se ha discutido el **posicionamiento** se podrá revisar el aspecto del uso eficiente de esta instrucción con opción **NO-PRIN-Gx**. (En la opción SEC-Gx siempre será eficiente su uso, ya que se busca el siguiente registro).

La forma interna de trabajo de la opción NO-PRIN-Gx es la siguiente:

- ◆ Si el grupo de llaves está al **inicio** se efectuará internamente una búsqueda con opción PRIN-Gx. (o sea, la primera vez que se usa).
- ◆ Si el grupo de llaves **no** está al inicio se hacen búsquedas secuenciales en el directorio para encontrar el registro (o sea, la segunda, tercera, ... vez que se usa).

Por lo anterior, las reglas para su uso eficiente o ineficiente son:

- ◆ Si de las variables llave usadas en la opción hay muchas iguales la instrucción es eficiente.
- ◆ Si de las variables llaves usadas en la opción hay pocas iguales pero **NO** se brincan llaves, la instrucción es eficiente.

- ◆ Si de las variables llave usadas en la opción hay pocas iguales y SI se brincan llaves, la instrucción es **ineficiente**.

Como un grupo de llaves se puede usar para búsqueda con PRIN-Gx o con NO-PRIN-Gx combinar el uso eficiente de ambas puede no ser fácil, entonces se puede solucionar el problema creando dos grupos de llave pero con nuevos aspectos que revisar, como son tiempos de inserción y uso de área en disco. Por lo anterior, se deberá estudiar las frecuencias de uso, ya que puede suceder que el 90% de los programas que usen el grupo de llaves lo hagan con opción PRIN-Gx y que sólo uno o dos lo hagan con opción NO-PRIN-Gx, en cuyo caso estos programas se deberán cambiar a usar temporales y la opción ORDEN.

Hasta ahora se ha descrito esta instrucción con ejemplos en el registro maestro, pero con esta instrucción se pueden también buscar recurrentes que tengan llaves.

A continuación se tiene un ejemplo del uso de esta instrucción para buscar recurrentes con llave, también se describen ciertas operaciones que se hacen al inicio de esta instrucción. Suponga que:

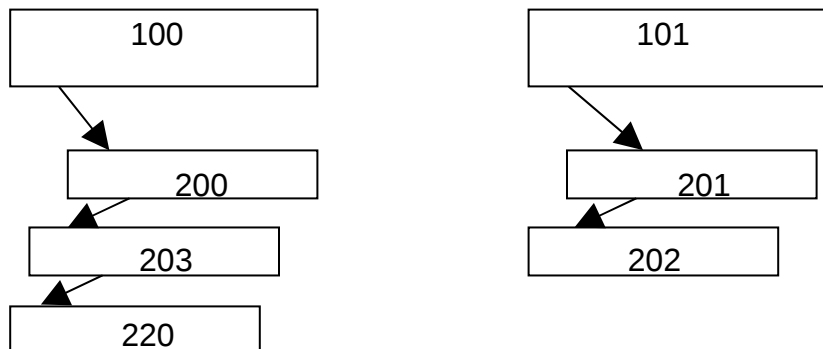
a) En DBD se definió:

```

ARCHIVO CLTES 7 4
-----
NUM-CLI  'NOMBRE DE CLIENTE' * R0 LL1 N2 0
NOM-CLI  'NOMBRE DEL CLIENTE' * S  X      30
-----
NUM-FACT '#FACTURA'          * R1 LL3 N1 0
FECHA-FACT 'FECHA FACT'      * R1 S    F
-----

```

b) Se tienen estos valores:



es decir, el cliente 100 tiene en el recurrente 1 las facturas con número 200, 203, 220. El cliente 101 tiene las facturas con número 201 y 202.

c) Se tienen las siguientes instrucciones de CLS.

```

                BUSCAR DISCO CLTES EJECUTA 20 SEC-G3
- - -   instrucciones en que se modifica el
- - -   recurrente 1 y el registro maestro
                EJECUTA 10
20    - - - - -

```

d) Se efectúa por primera vez BUSCAR DISCO.

- ◆ Si la primera factura es la 200 con la instrucción de BUSCAR DISCO ... SEC-G3 se pasa de disco a memoria el recurrente 1 con la información de la factura 200, pero también se pasa de disco a memoria la información del registro maestro del cliente 100. Ambos registros se pueden leer, modificar, etc., es decir, están disponibles para ser procesados.
- ◆ En forma general, cuando se hace un BUSCAR DISCO en un grupo de llaves de un recurrente y éste se encuentra, se pasan de disco a memoria dicho registro, el registro del cual depende, hasta el registro maestro de forma que todos ellos están disponibles para ser procesados y se pueden leer, modificar, borrar, etc.

e) Se efectúa por segunda vez BUSCAR DISCO.

- ◆ Si el recurrente con la información de la factura 200 se modificó y si la información del registro maestro del cliente 100 se modificó también se pasan estos registros de memoria a disco para actualizarlos.
- ◆ Se busca la siguiente factura que es la 201, se pasa de disco a memoria dicho recurrente así como el registro maestro del cliente 101.
- ◆ En forma general la instrucción de BUSCAR DISCO revisa si los registros disponibles para proceso (registro maestro y sus recurrentes) han sido modificados o borrados para hacer la actualización en disco y después buscar el registro de acuerdo a la opción. Note que al hacer la actualización los registros disponibles para proceso quedan NO disponibles de forma que si se sale por el EJECUTA # de la instrucción BUSCAR DISCO, NO hay registros disponibles para procesar en este archivo. En el caso de que se encuentre el registro buscado habrá nuevos registros disponibles.

f) Se efectúa por tercera vez BUSCAR DISCO.

- ◆ Se hace la actualización de la factura 201 y del cliente 101 si es que se modificaron.
- ◆ Se busca la siguiente factura que es la 202, se pasan de disco a memoria el recurrente con la 202 y el registro maestro del cual depende, o sea, el del cliente 101.
- ◆ Es decir, lo primero que hace la instrucción de BUSCAR DISCO es la actualización, independientemente que al buscar, algunos de los registros sean los mismos.

Por lo descrito en este ejemplo, las opciones de BUSCAR DISCO (y el posicionamiento) trabajan igual si se usan en grupos de llaves del registro maestro o de recurrentes.

Por último, se describen las opciones de **ORD y BUSC-Gx**. La opción de ORD. es para un caso especial cuando el registro maestro no tiene llaves, su uso es igual que la opción SEC-Gx y el orden en que son presentados los registros es el orden de llegada o inserción.

La opción BUSC-Gx es un caso especial de la opción PRIN-Gx y se usa cuando solamente se desea conocer si el registro buscado **existe** en el archivo, es decir, cuando se usa esta opción la instrucción de BUSCAR DISCO trabaja en la siguiente forma:

- ◆ No se realiza la actualización de registros disponibles para proceso, de forma que éstos quedan disponibles después de la instrucción y se pueden leer, modificar, ...
- ◆ Se busca el registro con las mismas reglas de la opción PRIN-Gx.
- ◆ No se modifica el posicionamiento que se tiene en el grupo de llaves.
- ◆ El registro buscado NO se pasa de disco a memoria y por lo tanto NO queda disponible para ser leído, modificado o borrado.

El siguiente ejemplo ilustra el uso de esta opción:

a) En DBD se definió:

```
ARCHIVO CLTES      7      4
```

```
-----
```

```

NUM-CLI 'NUMERO DE CLIENTE' * R0 LL1 N2 0
NOM-CLI 'NOMBRE DEL CLIENTE' * R0 S X 30
-----
NUM-FACT '#FACTURA' * R1 LL3 N 10
FECHA-FAC 'FECHA FACT' * R1 S F

```

b) Un programa para la carga inicial de facturas sería:

```

PROGRAMA 'CARGA DE FACTURAS'
*
* LEER EL CLIENTE
*
10 LEER TERMINAL EJECUTA 30 NUM-CLI
   BUSCAR DISCO CLTES EJECUTA 11 PRIN-G1 NUM-CLI
   EJECUTA 20
11 ERROR 'NO EXISTE EL CLIENTE' NUM-CLI
   EJECUTA 10
*
* LEER LAS FACTURAS
*
20 LEER TERMINAL EJECUTA 10 NUM-FACT ...
   BUSCAR DISCO CLTES EJECUTA 21 BUSC-G3 NUM-FACT
   ERROR 'FACTURA YA EXISTE'
   EJECUTA 20
21 - - - - instrucciones para insertar la factura
   EJECUTA 20
30 FIN

```

Con la instrucción de BUSCAR DISCO con opción BUSC-G3 se busca en directorio si esta factura ya se había cargado y **no se altera** el resto de la información.

Note que si se hubiera usado la opción PRIN-G3 se tendría que cambiar la lógica, ya que:

- ◆ Si no existe la factura (se efectúa el EJECUTA 21) se perdería la posición en el registro maestro del cliente cuyas facturas se están cargando, puesto
- 2.1) Buscar un registro en disco.

que el primer paso de la instrucción BUSCAR DISCO con opción PRIN-Gx es la actualización.

- ◆ Si existe la factura en el archivo, el registro maestro que está ahora en memoria es el de la factura y no el del cliente cuyas facturas se están cargando.

El cambio en la lógica consistirá en tener que buscar el registro maestro.

Lo descrito en la opción PRIN-Gx respecto a varias llaves y al uso eficiente o ineficiente de la instrucción es también aplicable a la opción BUSC-Gx.

En **resumen**, los 5 conceptos importantes de esta instrucción se describen a continuación. La forma general de trabajo de la instrucción es:

- ◆ Actualización en disco de los registros disponibles para ser procesados (registro maestro y, en su caso, los recurrentes) que han sido modificados o borrados.
- ◆ Busca en el directorio indicado de acuerdo a la opción y guarda la posición en dicho directorio.
- ◆ Pasa el registro buscado de disco a memoria y en caso de recurrentes pasa también de los que depende. De forma que quedan disponibles para ser procesados.
- ◆ En el caso de la opción BUSC-Gx sólo se busca en los directorios.

Cuando un registro está disponible para ser procesado, se pueden efectuar las siguientes instrucciones de **SISINF** para archivos permanentes.

- ◆ Leer cualquiera de sus variables con LEER DISCO.
- ◆ Modificar variables simples con ESCRIBIR DISCO.
- ◆ Borrar el registro con BORRAR DISCO (para el maestro o BORRAR RECURRENTE).
- ◆ Se pueden insertar recurrentes dependientes con INSERTAR DISCO.

La forma de búsqueda de las opciones es:

- ◆ En la opción PRIN-Gx se busca el primer registro que cumpla por igual en las llaves especificadas.

- ◆ En la opción NO-PRIN-Gx a partir de donde está posicionado, se busca en el directorio por aquel registro que cumpla por igual en las llaves especificadas.
- ◆ En la opción SEC-Gx se busca el siguiente registro donde está posicionado el directorio.
- ◆ La opción ORD. es para el caso de registro maestro sin llaves, se busca el siguiente registro por orden de inserción.
- ◆ La opción APROX-Gx busca el primer registro que cumpla por igual en las llaves especificadas y si no se encuentra, el siguiente mayor.
- ◆ La opción BUSC-Gx trabaja igual que la opción PRIN-Gx pero no se actualiza, no se posiciona y no se pasa el registro como en las demás opciones.
- ◆ La opción ULTIMO-Gx busca el último registro, si se especifican llaves el que sea igual a las llaves especificadas y si no existe el último anterior.

Se EJECUTA la instrucción con etiqueta # de la instrucción de BUSCAR DISCO cuando:

- ◆ En opción PRIN-Gx, APROX-Gx y BUSC-Gx el registro buscado no existe
- ◆ En opción NO-PRIN-Gx, SEC-Gx y ORD., ya no hay más registros.
- ◆ En opción ULTIMO-Gx sin llaves y no hay registros en el archivo. O bien con llave menor al primer registro.

Y en los tres casos NO hay registros disponibles para proceso en este archivo.

Un grupo de llaves está al **inicio** cuando:

- ◆ La instrucción de BUSCAR DISCO es la primera que se efectúa en el archivo.
- ◆ En la última ejecución de un BUSCAR DISCO el registro buscado no existe o ya no hay más registros.
- ◆ Se ejecuta la instrucción de OPCION DISCO NOMB INICIO-Gx.

Cuando un grupo de llaves está al inicio con opción SEC-Gx se busca el primer registro y con opción NO-PRIN-Gx se busca desde el primero.

- 2.1) Buscar un registro en disco.

Tal como se había especificado, esta instrucción se usa en problemas simples y en problemas complejos, por lo que se deberá conocer su funcionamiento general y el de cada opción.

2.2) Buscar Recurrente.

Esta instrucción sirve para buscar un recurrente de un archivo permanente.

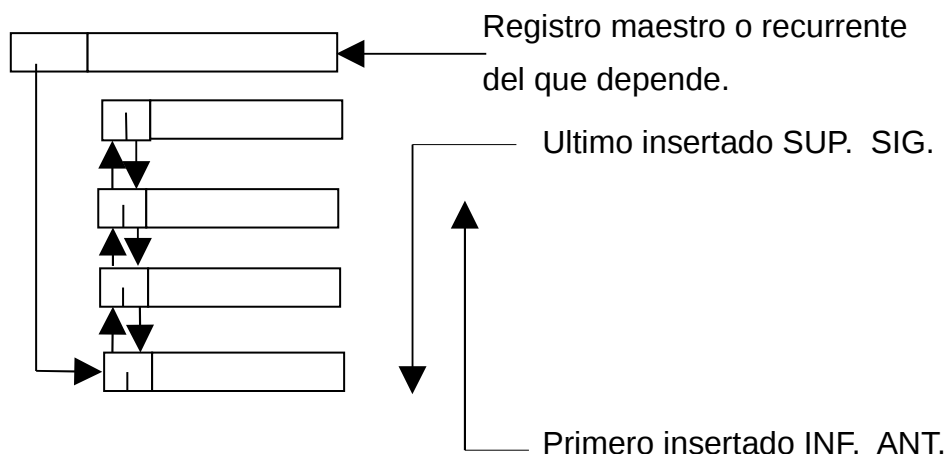
El formato general de la instrucción es:

BUSCAR RECURRENTE NOMB NUM-R EJECUTA #
INF. o SUP. o ANT. u ORD. (VAR-R VAR-R)

En donde:

- ◆ NOMB es el nombre del archivo permanente y es el mismo que se usó en la definición de ARCHIVO, en el módulo DBD.
- ◆ NUM-R es el número del tipo de recurrente que se desea buscar, así para una especificación de R1 en DBD se deberá poner un 1, para R2 un 2, etc.
- ◆ Si no se encuentra el registro buscado se EJECUTA la instrucción con etiqueta #.
- ◆ Se buscará por cualquiera de las opciones listadas, o sea, por INF., SUP., etc.
- ◆ En las opciones de INF., SUP., ANT., SIG. y ORD, si se especifica VAR-R, VAR-R ... (opcional) se modifica la forma de trabajo de la opción.
- ◆ Las variables VAR-R deben pertenecer al recurrente que se busca.

Para aclarar la forma de trabajo de las opciones SUP., INF., SIG. y ANT. es necesario conocer la forma en que se insertan los recurrentes. Gráficamente se tiene:



Es decir, los recurrentes siempre se insertan al final, de forma que se pueden buscar de primero insertado a último insertado, o bien de último insertado a primero insertado.

La forma de trabajo de las opciones **SUP.** e **INF.** es la siguiente:

- ◆ Si se usa la opción SUP. sin variables, se busca el último recurrente insertado.
- ◆ Si se usa la opción INF. sin variables, se busca el primer recurrente insertado.
- ◆ Si se usa la opción SUP. y una o más variables, se busca desde el último insertado por aquel recurrente que tenga igual el contenido de las variables VAR-R.
- ◆ Si se usa la opción INF. y una o más variables, se busca desde el primer insertado por aquel recurrente que tenga igual el contenido de las variables VAR-R.

Un ejemplo de cómo se busca al usar estas opciones sería:

a) Se supone que se tienen 5 recurrentes y el valor de una de las variables de dicho recurrente cuyo nombre es A.

	Registro Número	Valor de A
Último insertado	5	10
	4	15
	3	15

	2	8
Primer insertado	1	9

b) Se ejecuta la instrucción de BUSCAR RECURRENTE con las siguientes opciones y valores de A.

<u>Opción</u>	<u>Registro buscado (número)</u>
SUP.	5
INF.	1
SUP. con A = 15	4
INF. con A = 15	3
SUP. con A = 9	1
INF. con A = 10	5
SUP. con A = 8	2
INF. con A = 8	2

Note que si hay varios con el mismo valor la opción busca el primero. (SUP. con A = 15).

Al igual que en la instrucción de BUSCAR DISCO en que se tiene el concepto de **posicionamiento** en el grupo de llaves, en la instrucción de BUSCAR RECURRENTE se tiene el concepto de posicionamiento en el recurrente. Es decir, al ejecutar la instrucción de BUSCAR RECURRENTE, una vez que se ha buscado un registro, se guarda en tablas internas su posición.

La forma de trabajo de la opción **SIG.** y **ANT.** es la siguiente:

- ◆ Si se usa la opción SIG. sin variables, se busca el siguiente recurrente a donde está posicionado, el recorrido es de último insertado a primero insertado.
- ◆ Si se usa la opción ANT. sin variables, se busca el recurrente anterior a donde está posicionado, el recorrido es de primero insertado a último insertado.
- ◆ Si se usa la opción SIG. con una o más variables, se busca desde donde está posicionado por el siguiente recurrente que tenga igual el contenido de las variables VAR-R.
- ◆ Si se usa la opción ANT. con una o más variables, se busca desde donde está posicionado por el anterior recurrente que tenga igual el contenido de las variables VAR-R.

- ◆ En la opción SIG. con o sin variables, si el posicionamiento está al INICIO, se busca a partir del último insertado.
- ◆ En la opción ANT. con o sin variables, si el posicionamiento está al INICIO, se busca a partir del primero insertado.

Un ejemplo de cómo se busca al usar estas opciones sería:

a) Se supone que se tienen 5 recurrentes y el valor de una de las variables de dicho recurrente cuyo nombre es A.

	Registro Número	Valor de A
Último insertado	5	10
	4	15
	3	15
	2	8
Primero insertado	1	9

b) Se ejecuta la instrucción de BUSCAR RECURRENTE con la opción SIG. y sin variables.

Opción	Posicionamiento antes de Inst.	Registro bus- cado (número)	Posicionamiento después de Inst.
SIG.	Inicio	5	5
SIG.	5	4	4
SIG.	4	3	3
SIG.	3	2	2
SIG.	2	1	1
SIG.	1	-	Inicio

La última vez ya no se encontró registro y se EJECUTA la instrucción con etiqueta #. Esta opción se puede usar para buscar todos los recurrentes de un tipo de último insertado a primero insertado.

c) Se ejecuta la instrucción de BUSCAR RECURRENTE con Opción ANT. y sin variables.

Opción	Posicionamiento antes de Inst.	Registro bus- cado (número)	Posicionamiento después de Inst.
--------	-----------------------------------	--------------------------------	-------------------------------------

ANT.	Inicio	1	1
ANT.	1	2	2
ANT.	2	3	3
ANT.	3	4	4
ANT.	4	5	5
ANT.	5	-	Inicio

Esta opción se puede usar para buscar todos los recurrentes de un tipo de primero insertado a último insertado.

d) Se ejecuta la instrucción de BUSCAR RECURRENTE con opción SIG. y A = 15

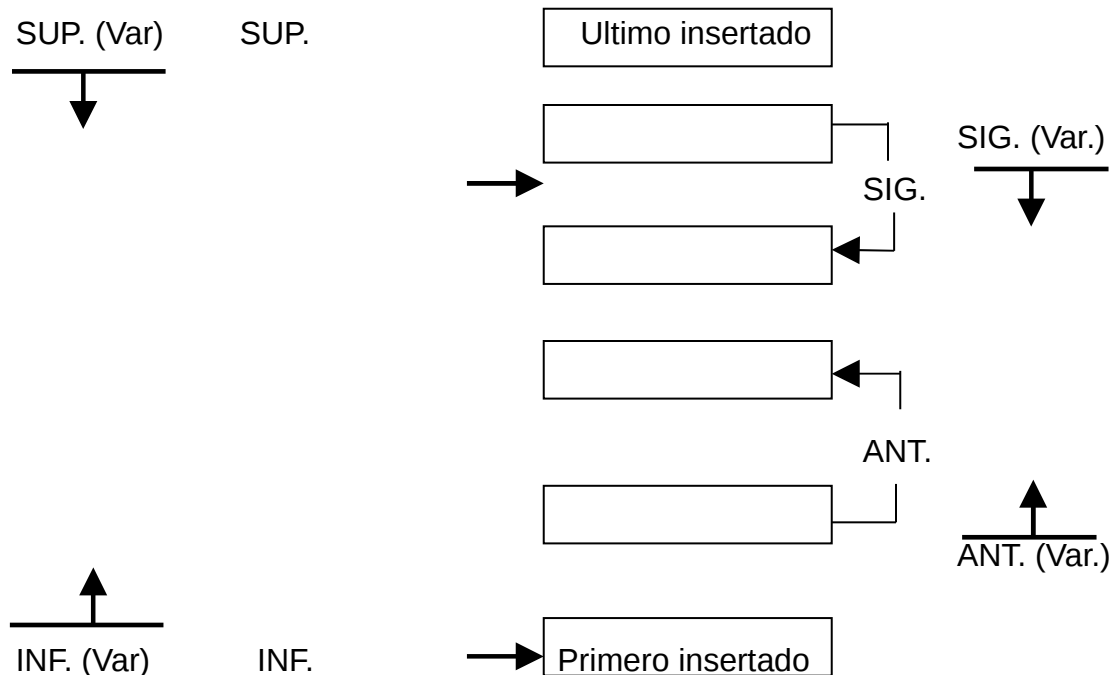
<u>Opción</u>	<u>Posicionamiento antes de Inst.</u>	<u>Registro buscado (número)</u>	<u>Posicionamiento</u>
SIG. con A = 15	Inicio	4	4
SIG. con A = 15	4	3	3
SIG. con A = 15	3	-	Inicio

Esta opción se puede usar para buscar todos aquellos que tengan igual una variable y el recorrido se hace de último insertado a primero insertado.

e) Se ejecuta la instrucción de BUSCAR RECURRENTE con las siguientes opciones y valores de A.

<u>Opción</u>	<u>Posicionamiento antes de Inst.</u>	<u>Registro buscado (número)</u>	<u>Posicionamiento después de Inst.</u>
SIG. con A= 8	Inicio	2	2
ANT. con A = 15	2	3	3
ANT.	3	4	4
SIG. con A = 9	4	1	1
ANT. con A = 16	1	-	Inicio
ANT.	Inicio	1	1

Gráficamente se pueden representar estas cuatro opciones de la siguiente forma:



En los ejemplos descritos el posicionamiento del recurrente estaba en ciertos casos al INICIO. A continuación se discute cómo se logra:

- ◆ Cuando un registro maestro o un recurrente se pone **disponible** para ser procesado, sus recurrentes **dependientes** están al inicio.
- ◆ Cuando se ejecuta la instrucción de BUSCAR RECURRENTE y el registro buscado no se encuentra, este recurrente está al inicio.
- ◆ Ejecutar la siguiente instrucción.

OPCION DISCO NOMB INICIO-Rx

En donde NOMB es el nombre del archivo y x es el número del recurrente. Esta instrucción se puede usar con otras opciones, las cuales se describen más adelante.

Después de haber descrito el funcionamiento de las opciones INF., SUP., ANT. y SIG. se comenta a continuación el funcionamiento general de la instrucción BUSCAR RECURRENTE.

La opción de ORD. se discutirá después.

Suponga que:

- 2.2) Buscar Recurrente.

a) En DBD se define (El R1 depende del R0)

```
ARCHIVO  CLTES          7  3
```

```
-----
```

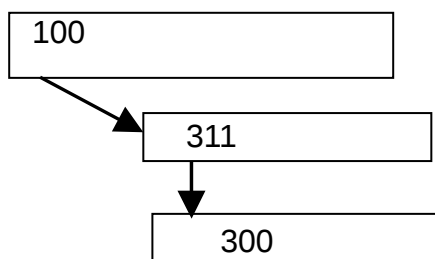
```
NUM-CLT  'NUMERO DE CLIENTE' *   R0 LL1 N1 0
```

```
-----
```

```
-----
```

```
NUM-FACT 'NUMERO DE FACTURA' *   R1 S  N1 0
```

b) Se tienen los siguientes valores:



c) Un programa que efectúe alguna modificación en las facturas sería:

```
7  BUSCAR DISCO CLTES EJECUTA 10 SEC-G1
```

```
8  BUSCAR RECURRENTE CLTES 1 EJECUTA 7 SIG.
```

```
----- instrucciones para modificar el
```

```
----- recurrente de facturas
```

```
EJECUTA 8
```

```
10 -----
```

```
-----
```

d) El BUSCAR DISCO pasa de disco a memoria el registro maestro del cliente 100 dejándolo **disponible** para ser procesado.

e) Se ejecuta por primera vez BUSCAR RECURRENTE.

- ◆ Se revisa que el registro maestro del que depende el recurrente esté disponible.
- ◆ Se busca de acuerdo a la opción, o sea, el último recurrente insertado que es la factura 311 y se pasa la información de ésta de disco a memoria de forma que el recurrente está disponible para ser procesado, se puede leer, modificar, borrar, etc.

- ◆ En forma general, lo primero que se hace al ejecutar la instrucción de BUSCAR RECURRENTE es revisar que el registro del que depende el recurrente esté **disponible**.

f) Se ejecuta por segunda vez BUSCAR RECURRENTE.

- ◆ Se revisa que el registro maestro del que depende el recurrente esté disponible.
- ◆ Si el recurrente de la factura 311 fue modificado se pasa la información del mismo de memoria a disco para que quede actualizado.
- ◆ Se busca el **siguiente** recurrente que será el de la factura 300, pasando la información de disco a memoria de forma que el recurrente está disponible para ser procesado.
- ◆ En forma general la instrucción de BUSCAR RECURRENTE revisa si el recurrente se modificó o se borró para hacer la actualización. También revisa si los recurrentes dependientes de éste y que estén disponibles fueron modificados o borrados para hacer la actualización de ellos.

g) Se ejecuta por tercera vez BUSCAR RECURRENTE.

- ◆ Se revisa que el registro maestro del que depende esté disponible.
- ◆ Si el recurrente de la factura 300 fue modificado, se pasa la información del mismo de memoria a disco para que quede actualizado.
- ◆ Se busca el siguiente recurrente y como ya no hay se EJECUTA la instrucción con etiqueta # y se pone el posicionamiento de este recurrente al INICIO.

El siguiente ejemplo ilustra el uso de esta instrucción en dos casos muy comunes al usar recurrentes.

a) En DBD se define:

```

ARCHIVO  CLTES          7  3
          (R1 depende de R0)
NUM-FACT 'NUMERO DE FACTURA' * R1 S N1  0
FECHA-FACT 'FECHA DE FACTURA'          * R1 S F
IMP-FAC   'IMPORTE DE FACTURA' * R1 S N2  2

```

b) En un programa donde se leen facturas e interesa **revisar** que no exista

- 2.2) Buscar Recurrente.

ese número de factura para un cliente específico se pondrían las siguientes instrucciones:

_____ (instrucciones para poner **disponible** el registro maestro del cliente)

5 LEER TERMINAL EJECUTA 11 NUM-FACT FECHA-FACT IMP-FACT
 BUSCAR RECURRENTE CLTES 1 EJECUTA 9 SUP. NUM-FACT
 ERROR 'FACTURA YA EXISTE'
 EJECUTA 5

9 Instrucción para insertarla en el archivo.

EJECUTA 5

10

Note que en este caso se usó SUP. NUM-FACT; se podría haber usado INF. NUM-FACT con resultados iguales.

El uso de BUSCAR RECURRENTE en este ejemplo es con propósito de validación y no para procesar el registro si se localiza.

c) En un programa en que se desean listar las facturas de un cliente:

_____ instrucciones para poner disponible
 _____ el registro del cliente.

100 BUSCAR RECURRENTE CLTES 1 EJECUTA 200 SIG.

_____ se lee e imprime la información
 _____ del recurrente.

200 _____ ya no hay más facturas

Note que en este caso se usó SIG. de forma que las facturas más nuevas se listaron primero, si se usa ANT. se listarán primero las facturas más antiguas.

Quando el registro maestro está disponible para ser procesado se pone el posicionamiento del recurrente al INICIO de forma que la primera vez que se ejecuta la instrucción BUSCAR RECURRENTE con opción SIG. se buscará el **último** insertado.

A continuación se discute la opción de **ORD**. como ya se describió con la opción SIG. se puede recorrer el recurrente del último insertado a primero insertado; con la opción ANT. se puede recorrer de primero insertado a último insertado, con la opción ORD. se recorre el recurrente con cierto orden (ascendente o descendente) requerido por el programador. Para hacer esto es necesario ORDENAR el recurrente y luego buscar los registros.

La siguiente instrucción se usa para ordenar un recurrente:

OPCION DISCO NOMA ORD-Rx A/D VAR-R1 A/D VAR/R2

En donde:

NOMA es el nombre del archivo cuyo recurrente se ordena.

ORD-Rx es la opción de esta instrucción y x es el número del recurrente.

VAR-R1 es el nombre de la variable que se usará como primera llave del ordenamiento.

A o D antes de la variable, indica si será ascendente o descendente.

Para aclarar más el uso de esta opción se tiene el siguiente ejemplo:

a) En DBD se definió lo siguiente:

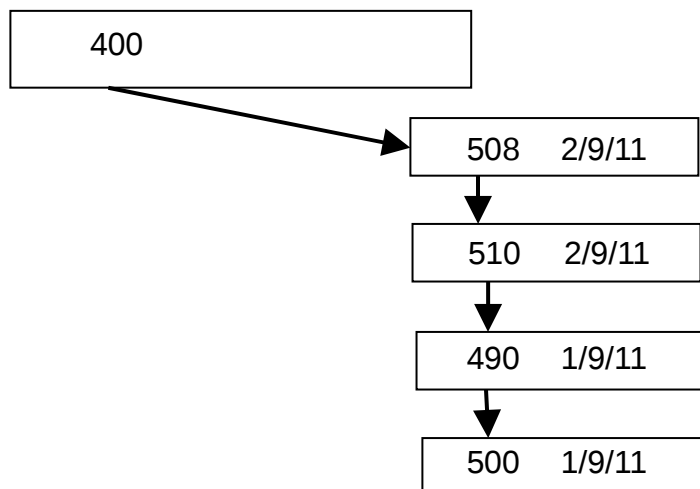
ARCHIVO CLTES 7 3

NUM-CLI 'NUMERO DE CLIENTE' * R0 LL1 N1 0

NUM-FACT 'NUMERO DE FACTURA' * R1 LL3 N1 0

FECHA-FACT 'FECHA DE FACTURA' * R1 S F

b) Se tienen los siguientes valores:



c) Un programa que liste las facturas ordenadas por fecha y número de factura sería:

PROGRAMA 'LISTAR FACTURAS'

----- opciones de impresión

10 BUSCAR DISCO CLTES EJECUTA 30 SEC-G1

----- se lee e imprime información general del cliente

OPCION DISCO CLTES ORD-R1 A FECHA-FACT A NUM-FACT

20 BUSCAR RECURRENTE CLTES 1 EJECUTA 10 ORD.

----- se lee e imprime información de cada factura

EJECUTA 20

30 FIN

d) Con la instrucción de BUSCAR DISCO se va buscando cliente por cliente, si se supone que se buscó el cliente con número 400 de forma que el registro maestro está disponible para ser procesado. La instrucción de BUSCAR DISCO puso como NO ORDENADO al recurrente de facturas.

e) Se ejecuta la instrucción de OPCION DISCO con ORD-Rx.

- ◆ Se ordenan en forma ascendente los 4 recurrentes de facturas del cliente 400, la primera variable para definir el ordenamiento es FECHA-FACT, la segunda es NUM-FACT, de forma que el recurrente de facturas quedaría ordenado de la siguiente forma:

490 1-9-11

500 1-9-11

508 2-9-11

510 2-9-11

- ◆ El recurrente ahora está ORDENADO.

f) Se ejecuta la instrucción de BUSCAR RECURRENTE opción ORD. por primera vez después de haberlo ordenado.

- ◆ Se revisa que el recurrente esté ORDENADO, es decir, que ya se ejecutó OPCION DISCO ORD-Rx en este recurrente.

- ◆ El recurrente de la factura 490 está disponible y se puede leer, modificar o borrar.

g) Se ejecuta BUSCAR RECURRENTE con opción ORD. por segunda vez.

- ◆ Se revisa que el recurrente esté ORDENADO.
- ◆ Si el recurrente de la factura 490 fuera modificado se pasa la información del mismo de memoria a disco.
- ◆ El registro de la factura 500 se pone como disponible.

h) Se ejecuta BUSCAR RECURRENTE con opción ORD. por tercera y cuarta vez, teniendo las factures 508 y 510 disponibles para su proceso.

i) Se ejecuta por quinta vez BUSCAR RECURRENTE.

- ◆ Como no hay más recurrentes se ejecuta la instrucción con etiqueta 10 y este recurrente se pone como NO ORDENADO.

Los siguientes comentarios son aplicables a BUSCAR RECURRENTE con opción ORD.

- ◆ Se asume que antes de la ejecución de esta instrucción se puso OPCION DISCO con opción ORD-Rx para indicar el orden deseado.
- ◆ La instrucción de OPCION DISCO se debe poner afuera del ciclo, si no se hace cada vez se ordenará el recurrente y el registro buscado será el primero.
- ◆ Si al especificar la opción ORD. se ponen variables, sólo aquellos registros con el mismo valor serán buscados.

- ◆ Cuando un registro se ha buscado, está disponible para ser procesado, es decir, se puede leer, modificar o borrar y sus dependientes pueden ser buscados, ordenados o insertar nuevos.
- ◆ Si se insertan registros en un recurrente después de haber ejecutado OPCION DISCO ORD-Rx en dicho recurrente, estos nuevos registros no serán encontrados con BUSCAR RECURRENTE opción ORD. Solamente los registros que existían al ejecutar OPCION DISCO ORD-Rx serán encontrados.
- ◆ Después de ejecutar OPCION DISCO ORD Rx se pueden ejecutar BUSCAR RECURRENTE con opción SUP., INF., ... en el mismo recurrente en combinación con BUSCAR RECURRENTE opción ORD.
- ◆ Si después de ejecutar OPCION DISCO ORD-Rx se ejecuta en el mismo recurrente BUSCAR RECURRENTE con opción SUP., INF., ... y el registro buscado se borra, dicho registro ya no será encontrado con BUSCAR RECURRENTE opción ORD.
- ◆ Si después de ejecutar OPCION DISCO ORD-Rx se ejecuta en el mismo recurrente BUSCAR RECURRENTE con opción SUP., INF., ... y el registro buscado se modifica (con ESCRIBIR DISCO), el registro modificado será obtenido con BUSCAR RECURRENTE opción ORD. Pero si la variable modificada fue llave del ordenamiento, el ordenamiento de los registros será el mismo que cuando se ejecutó OPCION DISCO ORD-Rx pero el contenido de las variables será el modificado.

El uso de la instrucción de OPCION DISCO ORD-Rx puede ser en forma eficiente e ineficiente.

A continuación se describe una serie de conceptos para aclarar esta situación.

La forma interna de ordenar el recurrente es la siguiente:

- ◆ Se revisa si el ordenamiento se puede hacer en memoria, en cuyo caso se pasan de disco a memoria **todos** los registros del recurrente y se **ordenan**.
- ◆ Si el ordenamiento NO se puede hacer en memoria, se genera un **archivo** secuencial y luego se ordena.

Si el ordenamiento es en memoria, el uso de esta instrucción es **eficiente**, si el ordenamiento se hace en archivos es ineficiente.

Los factores que determinan si el ordenamiento se puede hacer en memoria

son:

- ◆ Cantidad de memoria para hacer el ordenamiento.
- ◆ Tamaño del registro recurrente.
- ◆ Número de registros en el recurrente.

De estos factores el tamaño del registro es fijo y el número de registros en el recurrente es variable, ya que por ejemplo en el caso del archivo de clientes un cliente puede tener cero facturas, otros una factura, otras dos facturas, etc.

La cantidad de memoria para hacer el ordenamiento varía de programa a programa, la forma de asignarla es la siguiente:

- ◆ El módulo ELS tiene una cantidad de memoria fija para la ejecución de un programa de **SISINF**.
- ◆ El módulo CLS asigna la memoria NO usada por el programa de **SISINF** para que se use en los diferentes recurrentes que tienen la opción ORD. Note que es posible usar la instrucción de BUSCAR RECURRENTE con estas opciones en forma simultánea en varios archivos o varios recurrentes de un archivo.

El módulo CLS reporta al final de la compilación con la opción A la cantidad de memoria asignada a cada recurrente con esta opción y el número de registros máximo que puede tener el recurrente para que el ordenamiento sea en memoria.

Se deberá comparar el número de registros máximo con el promedio que tiene el recurrente y con esto estimar si es suficiente o no. De todas formas el módulo ELS lleva un contador interno del número de veces que esta instrucción se ha hecho en forma ineficiente y cada 10 veces se da un diagnóstico informativo.

En resumen, los 8 conceptos importantes de esta instrucción se describen a continuación:

1) La forma de trabajo cuando se usa las opciones SUP., INF., SIG. o ANT., es:

- ◆ Revisar que el registro maestro del que depende o el recurrente del que depende esté **disponible**.
- ◆ Actualizar en disco este recurrente y sus dependientes que estén

- 2.2) Buscar Recurrente.

disponibles para ser procesados, que han sido modificados o borrados.

- ◆ Se **busca** el recurrente de acuerdo a la opción y en el caso de INF., SUP., ANT. o SIG. se guarda su **posición**.
- ◆ El recurrente buscado queda **disponible** para ser procesado.

2) La forma de trabajo cuando se usa la opción ORD. es:

- ◆ Revisa que el recurrente esté ORDENANDO, es decir, se ejecutó OPCION DISCO ORD-Rx en dicho recurrente.
- ◆ Actualiza en disco este recurrente y sus dependientes que estén disponibles para ser procesados, que han sido modificados o borrados.
- ◆ Se busca el siguiente recurrente de acuerdo al ordenamiento.
- ◆ El registro es puesto como **disponible** para ser procesado.

3) Cuando un recurrente está disponible para ser procesado se pueden efectuar las siguientes instrucciones de **SISINF**.

- ◆ Leer cualquiera de sus variables con LEER DISCO.
- ◆ Modificar variables simples con ESCRIBIR DISCO.
- ◆ Borrar el recurrente con BORRAR RECURRENTE.
- ◆ Buscar sus recurrentes dependientes con BUSCAR RECURRENTE.
- ◆ Insertar recurrentes dependientes con INSERTAR DISCO.
- ◆ Ordenar recurrentes dependientes con OPCION DISCO ORD-Rx y luego buscarlos.

4) La forma de búsqueda de las opciones cuando no se especifican VAR-R es:

- ◆ La opción SUP. busca el último insertado.
- ◆ La opción INF. busca el primero insertado.
- ◆ La opción SIG. busca el siguiente recurrente a partir de donde está posicionado. Si el recurrente está al inicio se busca el último insertado.

- ◆ La opción ANT. busca el recurrente anterior a partir de donde está posicionado. Si el recurrente está al inicio se busca el primero insertado.

- ◆ La opción ORD. busca el siguiente registro de acuerdo al ordenamiento.

5) La forma de búsqueda cuando sí se especifican VAR-R es:

- ◆ La opción SUP. busca desde el último insertado por aquel que cumpla por igual en las variables.

- ◆ La opción INF. busca desde el primero insertado por aquel que cumpla por igual en las variables.

- ◆ La opción SIG. busca el siguiente recurrente a partir de donde está posicionado, que cumpla por igual en las variables. Si el recurrente está al inicio se busca desde el último insertado.

- ◆ La opción ANT. busca el recurrente anterior a partir de donde está posicionado, que cumpla por igual en las variables. Si el recurrente está al inicio se busca desde el primero insertado.

- ◆ La opción ORD. busca desde la posición actual en base al ordenamiento, aquel registro cuyas variables tengan el mismo valor al especificado.

6) Un recurrente está al INICIO:

- ◆ Cuando un registro maestro o un recurrente se pone **disponible**, sus recurrentes dependientes están al INICIO.

- ◆ Cuando se ejecuta BUSCAR RECURRENTE, opción SUP., INF., SIG., ANT. y el registro buscado no se encuentra.

- ◆ Cuando se ejecuta OPCION DISCO NOMB INICIO-Rx, en donde NOMB es el nombre del archivo y x el número del recurrente.

7) Un recurrente está NO ORDENADO.

- ◆ Cuando un registro maestro o un recurrente se pone disponible, sus recurrentes dependientes están NO ORDENADO.

- ◆ Cuando se ejecuta BUSCAR RECURRENTE con opción ORD. y el registro buscado no se encuentra, este recurrente se pone como NO ORDENADO.

8) Un recurrente esta ORDENADO:

- ◆ Después de la ejecución de OPCION DISCO con ORD-Rx.

2.3) Leer un Archivo Permanente.

Se leen las variables indicadas de un Archivo Permanente. El formato general es:

LEER DISCO NOMB VAR-R VAR-R ...

Los siguientes comentarios y restricciones son aplicables:

- ◆ NOMB es el nombre del Archivo Permanente y es el mismo que se usó en la definición de ARCHIVO, en el módulo DBD.
- ◆ Las variables VAR-R pertenecen al archivo NOMB deben ser el mismo tipo de registro.
- ◆ El orden en que se especifiquen las variables puede ser cualquiera, y no coincidir con la definición del archivo en DBD.
- ◆ No se necesita leer todas las variables del archivo, solamente las que se necesiten para el programa.
- ◆ El registro debe estar disponible para ser procesado, ello se logra por las instrucciones de BUSCAR DISCO, BUSCAR RECURRENTE o bien se acaba de INSERTAR.
- ◆ Se puede hacer varias lecturas a un mismo registro. (No se lee físicamente del disco).

2.4) Escribir un Archivo Permanente.

Se escriben las variables indicadas de un archivo Permanente modificando su contenido en el archivo.

El formato general es:

ESCRIBIR DISCO NOMB VAR-R VAR-R VAR-R

Los siguientes comentarios y restricciones son aplicables:

- ◆ NOMB es el nombre del Archivo Permanente y es el mismo que se usó en la definición de ARCHIVO, en el módulo DBD.
- ◆ Las variables VAR-R pertenecen al archivo NOMB y deben ser del mismo tipo de registro.
- ◆ El orden en que se especifiquen las variables puede ser cualquiera y no coincidir con la definición del archivo en DBD.
- ◆ No se necesitan escribir todas las variables del archivo, solamente aquellas donde se quiera modificar su contenido en el registro.
- ◆ El contenido de las variables en memoria no se altera después de ejecutar la instrucción.
- ◆ No es necesario que se lea el contenido de una variable para modificarlo.
- ◆ No se pueden escribir variables llave.
- ◆ El archivo debe estar disponible en el registro deseado por las instrucciones de BUSCAR DISCO, BUSCAR RECURRENTE o bien se acaba de INSERTAR.
- ◆ Se pueden hacer varias escrituras en un registro. (NO se escribe físicamente en el disco).

2.5) Borrar un Registro.

Se da de baja un registro maestro de información así como sus recurrentes de un Archivo Permanente. El formato general de esta instrucción es:

BORRAR DISCO NOMB

Los siguientes comentarios y restricciones son aplicables:

- ◆ NOMB es el nombre del Archivo Permanente y el mismo que se usa en la definición de ARCHIVO en el módulo DBD.
- ◆ El registro maestro debe estar disponible para ser procesado en el registro que se desea borrar por la instrucción de BUSCAR DISCO, o bien se acaba de INSERTAR.
- ◆ La baja del registro es lógica, es decir, se pone un indicador de baja.

2.6) Borrar un recurrente.

Se borra un registro recurrente de un Archivo Permanente. El formato general de la instrucción es:

BORRAR RECURRENTE NOMB NUM-R

Los siguientes comentarios y restricciones son aplicables:

- ◆ NOMB es el nombre del Archivo Permanente y es el mismo que se usó en la definición de ARCHIVO, en el módulo DBD.
- ◆ El recurrente debe estar disponible para ser procesado en el registro que se desea borrar, por las instrucciones de BUSCAR DISCO o BUSCAR RECURRENTE.
- ◆ NUM-R es el número del tipo de recurrente que se desea borrar. Poner 1 es para R1, 2 para R2.
- ◆ La baja del registro es lógica, es decir, se pone un indicador de dado de baja. Salvo el caso de recurrentes con opción M que sí son bajas físicas, es decir, se recupera el espacio.

2.7) Borrar un archivo.

Se borra el archivo que se especifica por NOMB. El formato general de la instrucción es:

BORRAR ARCHIVO NOMB

2.8) Insertar un registro.

Esta instrucción sirve para dar de alta o insertar en un Archivo Permanente.

El formato general de la instrucción es:

INSERTAR DISCO NOMB (EJECUTA #) VAR-R VAR-R VAR-R

Los siguientes comentarios y restricciones son aplicables:

- ◆ NOMB es el nombre del Archivo Permanente y es el mismo que se usó en la definición de ARCHIVO, en el módulo DBD.
- ◆ Las variables VAR-R pertenecen al archivo NOMB y deben ser del mismo tipo de registro.
- ◆ El orden en que se especifiquen las variables puede ser cualquiera y no coincidir con la definición del archivo en DBD.
- ◆ Aquellas variables que no se insertan por esta instrucción el sistema les asigna un valor de 0 para las numéricas y el caracter nulo para las alfanuméricas.
- ◆ Al hacer la inserción se actualizan los directorios de los grupos de llaves que se tengan. NO se permite que la llave formada por todas las variables del grupo de llaves esté duplicada.
- ◆ La opción de EJECUTA #, especifica que si existen llaves duplicadas el registro NO se inserte y se ejecute la instrucción con etiqueta #.

Precaución: La lógica interna es primero insertar el registro lógico y luego actualizar directorios. En caso de existir un registro con las mismas llaves, ocasiona consumo de área en disco; NO se recomienda esta opción para probar si existe o no.

- ◆ Si no se especifica EJECUTA # se desplegará error de ejecución en el caso de llaves duplicadas.
- ◆ En el caso de recurrentes, éstos se insertan en forma de pila y con esta instrucción se forman los apuntadores.
- ◆ NO se pierde posicionamiento de las instrucciones de BUSCAR DISCO o BUSCAR RECURRENTE al ejecutar esta instrucción.
- ◆ Si se BUSCA RECURRENTE con opción ORD. mientras el recurrente está ORDENADO y se insertan recurrentes en éste, éstos no serán encontrados por la instrucción.
- ◆ Cuando se inserta el registro y a continuación sus recurrentes dependientes, primero se deberá insertar el registro del cual dependen y a continuación sus recurrentes dependientes.
- ◆ En el caso de recurrentes el registro del cual depende el que se inserta debe estar disponible por las instrucciones de BUSCAR DISCO, BUSCAR RECURRENTE o INSERTAR DISCO.

OPCION DISCO NOMB **EXCLUSIVO** (NUM)

OPCION DISCO NOMB **DISCO** LIT/DIS-X

OPCION DISCO NOMB **ORD-RX** A/D VAR-R1 A/D VAR-R2...

OPCION DISCO NOMB **RECA** (NUM)

Los siguientes comentarios son aplicables:

- ◆ NOMB es el nombre del archivo permanente que se usó en la definición de la DBD.
- ◆ Solo se puede usar una opción a la vez.
- ◆ La opción de INICIO-GX sirve para poner un grupo de llaves al inicio. Se recomienda ver la instrucción de BUSCAR DISCO para ejemplos de su uso.
- ◆ La opción de INICIO-RX sirve para posicionar un recurrente al INICIO. Se recomienda ver la instrucción de BUSCAR RECURRENTE para ejemplos de su uso.
- ◆ La opción de LIBERAR escribe toda la información del archivo en disco y libera el registro maestro. Se recomienda ver la sección de archivos compartidos así como la de funcionamiento interno para ejemplos de su uso.
- ◆ La opción EXCLUSIVO sirve para manejar un archivo permanente en forma exclusiva. Se recomienda ver la sección de archivos compartidos para ejemplos de su uso.
- ◆ La opción COMPARTIDO sirve para manejar un archivo permanente en forma compartida, es decir, varios usuarios al mismo tiempo estén accedando el archivo.
- ◆ En la opción de EXCLUSIVO como en COMPARTIDO se puede especificar una variable N1 de cero decimales NUM. En dicha variable se almacenan valores de 0 a 99 para indicar el % de uso del archivo y con esto decidir si es necesaria una reconstrucción. También se almacena un -1 si el archivo NO existe, o un -2 si se puso la opción de MARCAR-ABIERTO y el archivo tiene este indicador.
- ◆ Con la opción de DISCO se especifica en LIT o en la variable tipo X DIS-X el nombre del dispositivo en donde está el archivo. Este nombre puede ser un directorio real o un nombre lógico que define su ubicación.

- ◆ El módulo ELS usa el dispositivo especificado en esta opción, si no se definió toma lo de la definición de ARCHIVO en la DBD, si ésta se dejó en blanco toma un valor definido en la CLAVE de ACCESO con que se inició la ejecución, si ésta está en blanco se toma un parámetro de instalación y por último si ésta también vale blancos se toma el 'Working Directory' o el 'Default Directory' o el grupo de HELLO, ... dependiendo del Sistema Operativo.
- ◆ Se recomienda ver la sección de BUSCAR RECURRENTE para ejemplos de uso de ORD-RX.

A continuación se describe la opción de RECA.

Los siguientes comentarios son aplicables:

- ◆ El módulo ELS tiene integrado un subconjunto del módulo RECA para ejecutar esta instrucción.
- ◆ El nombre del archivo a reconstruir es Nomb.
- ◆ Se puede especificar una variable N1 con 0 decimales o una constante con valor de 1 a 99 para definir el % deseado de uso para el archivo. Al reconstruirlo, ELS calcula el tamaño de los directorios y área de datos para que el archivo quede usado en ese porcentaje, en base a los registros que tiene actualmente (borrados y no borrados).
- ◆ El archivo se puede haber usado por la instrucción de OPCION DISCO EXCLUSIVO o COMPARTIDO.
- ◆ Después de esta instrucción el archivo se puede usar en forma normal.
- ◆ Cuando se compila el programa se guarda el dispositivo en donde está el archivo DBDSxx. Al ejecutar la instrucción con este dispositivo se abren los archivos internos ARCSxx, CAMSxx y TIPSxx. Lo anterior es importante si se llega a cambiar el dispositivo.
- ◆ Cuando se ejecuta esta opción se podrán tener los mismos errores que en la ejecución del módulo RECA con lo cual se procederá a diagnosticar y cancelar la ejecución. El código (C) que reporta tiene el siguiente significado:

001 Archivo ya se usó y hay buffer en memoria. Con Opción DISCO LIBERAR se escriben los buffers.

0002 El archivo lo tiene otra terminal. Se recomienda usar OPCION

DISCO EXCLUSIVO.

0003 El archivo NO existe.

0004 El archivo no está en ARCSxx.

0005 Error en registro 2. Es necesario reconstruir con RECA.

0006 Error en TIP se cambió la definición.

0007 No hay memoria para hacer ELS-RECA.

0008 Archivo NO ordenado.

0009 Error al convertir lógico a real.

01xx Error interno.

02xx Error en PEVx.

03xx Error en ORDEN.

1xxx Error en ARCSxx.

2xxx Error en CAMSxx.

3xxx Error en TIPSxx.

4xxx Error en archivo a reconstruir.

5xxx Error en archivo de salida.

6xxx Error en archivo de directorio.

7xxx Error en archivo para salvar memoria.

8xxx Error en archivo T990.

9xxx Error en archivo T991.

Cuando se ejecuta esta opción se generan dos archivos temporales, uno con la información de las estadísticas de la reconstrucción y el otro con la información de los registros perdidos y duplicados.

El formato del temporal T990 de estadística es:

- 2.9) Opción Disco.

Un registro con:

X60 nombre del archivo (directorio, nombre y extensión si se usa).

Un registro con:

X40 nombre extendido del archivo.

X8 fecha de última reconstrucción.

X8 hora de última reconstrucción.

X8 fecha esta reconstrucción.

X8 hora esta reconstrucción.

Un registro con:

N2 número de pedidos.

N2 número de duplicados.

Un registro con:

N1 número de recurrentes incluyendo el R0 (NTRM).

NTRM registros con:

N1 número del recurrente (0 = R0, 1 = R1, ...)

N2 número de registros leídos (incluye borrados y perdidos)

N2 número de registros borrados.

N2 número de registros perdidos.

Un registro con:

N1 número de grupos de llaves (NGLL).

NGLL registros con:

N1 número del grupo de llaves.

N2 número de llaves insertadas.

N2 número de llaves duplicadas (NDUP).

Un registro con:

N1 número de variables tipo texto, imagen o audio (NTIA).

NTIA registros con:

N1 número de variable, texto, imagen o audio (VTIA como 1,2,3, ...
NTIA).

- X30 nombre de la variable.
- N2 número de campos leídos (no incluye perdidos).
- N2 número de campos perdidos.
- N2 longitud promedio del campo.
- N2 longitud máxima del campo.

Por el formato del temporal se pueden hacer rutinas generales que exploren la información.

El formato de temporal T991 con la información de perdidos y duplicados es:

Un registro con:

- N2 número de perdidos (NP).

NP registros con:

- N1 tipo de error
 - 1 el R0 se pierde por error de lectura.
 - 2 el R0 se pierde por llave diferente.
 - 3 el RX se pierde por error de lectura.
 - 4 el RX se pierde por secuencia.
 - 5 el Texto/imagen/audio se pierde por error.
- N1 número de recurrente o número de la variable texto, imagen o audio VTIA definida en el T990.
- ... llave del R0 definida igual que en DBD.

Varios registros con:

- N1 número del grupo de llaves con variable llave duplicada.
- ... llave del grupo.
- ... llave del grupo.

La información de duplicados en el temporal está ordenada por grupo de llaves, primero para LL1 si hubo duplicados, luego LL2, ... Con NDUP del T990 de cada grupo de llaves se puede saber cuantos hay para cada grupo.

Este temporal se borra si no hay perdidos o duplicados.

El temporal de perdidos y duplicados es dependiente del archivo por lo que si se desea desplegar la información será necesario hacer las rutinas específicas.

El temporal de estadísticas tiene un formato general por lo que se tienen rutinas para su proceso. Si se quieren usar se deberá agregar un incluir del archivo ISRECD en la DBD; En los programas se tienen dos alternativas:

- ◆ Si se desea desplegar toda la información de estadística se deberán usar las rutinas de ISRECP.
- ◆ Si solo se desea saber si hubo perdidos o duplicados, se deberán usar las rutinas de ISRECR.

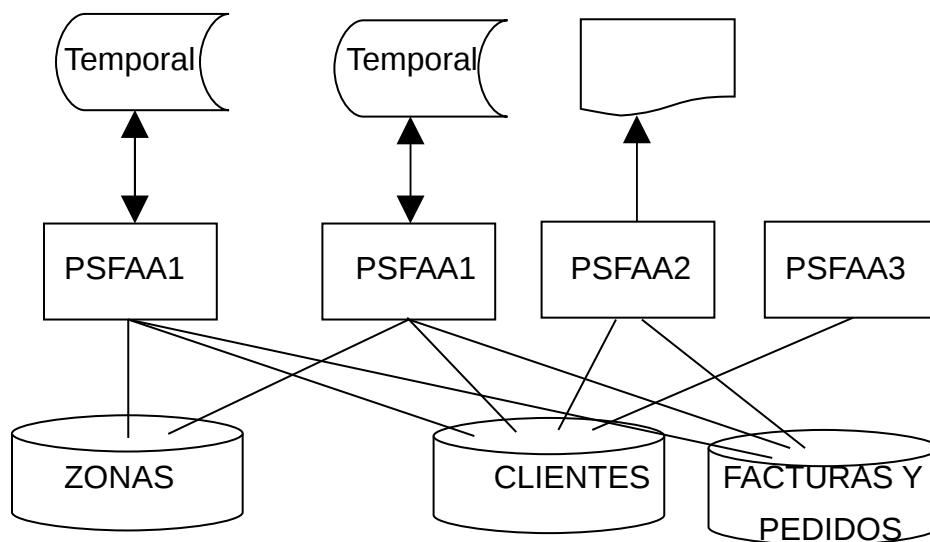
Se deberá revisar los tres archivos ya que en ellos está la documentación de su uso.

3) Archivos Compartidos.

3.1) Conceptos Generales.

En la operación normal de una computadora con terminales es común que el mismo programa se esté ejecutando desde varias terminales, o que diferentes programas estén accediendo el mismo archivo.

Un ejemplo gráfico de lo anterior es:



La siguiente es una lista de los programas involucrados en la gráfica y los archivos que accesa:

a) PSFAA1

- ◆ Se ejecuta en 2 terminales.
- ◆ Lee los archivos de clientes y zonas.
- ◆ Actualiza facturas y pedidos.
- ◆ Lee y escribe un temporal.

b) PSFAA2

- ◆ Se ejecuta en 1 terminal.
- ◆ Lee y actualiza el archivo de clientes.
- ◆ Lee y actualiza el archivo de facturas.
- ◆ Imprime un listado.

c) PSFAA3.

- ◆ Se ejecuta en 1 terminal.
- ◆ Actualiza el archivo de clientes.

El ejemplo anterior es posible bajo el lenguaje **SISINF** pero es necesario tomar ciertas consideraciones de programación para una operación adecuada.

El acceso de los archivos permanentes NO representa ningún problema cuando **varios** programas están leyendo el mismo archivo permanente, tampoco hay problemas cuando un solo programa lo está actualizando, es decir, está insertando registros o modificando registros previamente creados.

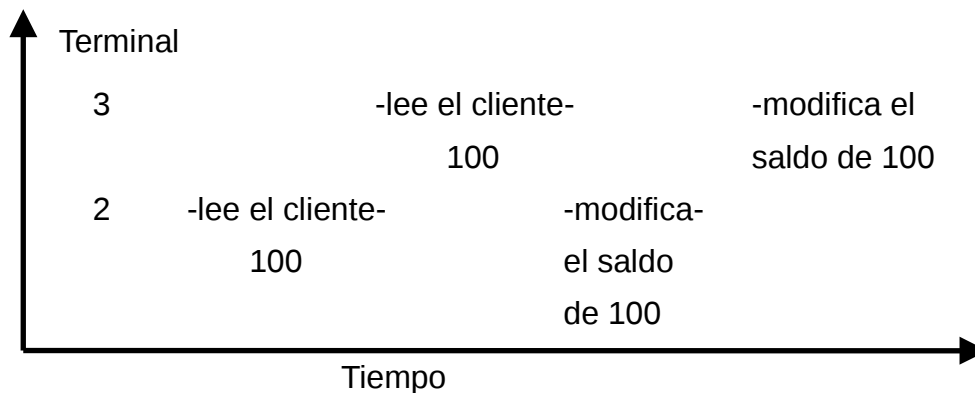
El problema es cuando varios programas están actualizando el mismo archivo.

A continuación se describe el problema y su solución:

El problema principal que se tiene en el caso de **actualización simultánea del mismo archivo** se puede ejemplificar de la siguiente forma:

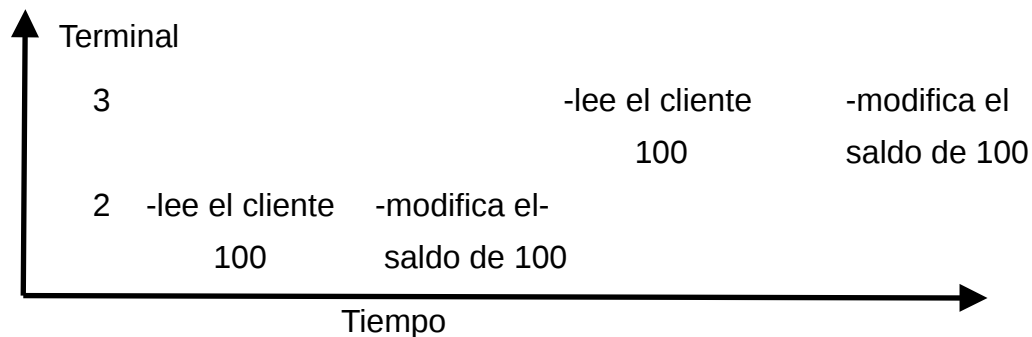
- ◆ El programa en la terminal 2 se lee el saldo del cliente 100 el cual es de \$1,500.00
- ◆ El programa en la terminal 3 se lee el saldo del **mismo cliente** (100) el cual es de \$1,500.00
- ◆ El programa en la terminal 2 modifica el saldo del cliente 100 haciéndole un cargo de \$2,000.00 de una factura quedando su saldo en \$3,500.00
- ◆ El programa en la terminal 3 modifica el saldo del cliente 100 haciéndolo un cargo de \$4,000.00 de una factura quedando su saldo en \$5,500.00
- ◆ El saldo queda en \$5,500.00 pero debería ser de \$7,500.00.

Gráficamente queda:



La solución al problema anterior es que se debe bloquear a nivel registro, es decir, el registro del cliente 100 lo tiene la terminal 2, nadie lo puede acceder hasta que la terminal lo escriba ya modificado.

Con esta solución gráficamente la ejecución queda como sigue:

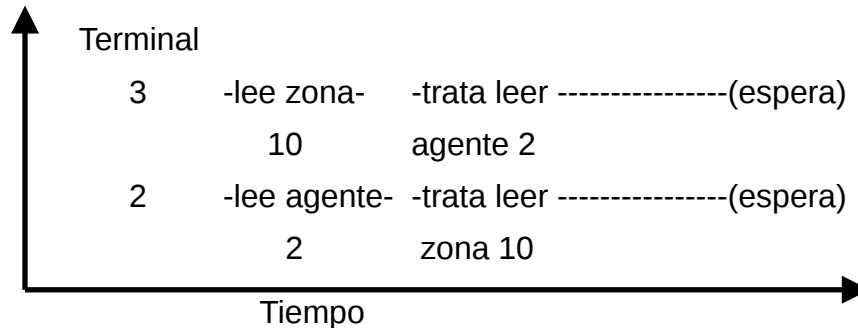


Pero un problema que se presenta con la solución anterior es el llamado 'Dead lock' y ocurre por problemas de programación. El siguiente ejemplo ilustra lo que sucede:

- ◆ El programa en la terminal 2 lee el registro del agente 2 (bloqueándolo).
- ◆ El programa en la terminal 3 lee el registro de la zona 10 (bloqueándolo).
- ◆ El programa en la terminal 2 trata de **leer** el registro de la zona 10 y se pone en espera hasta que sea liberado.
- ◆ El programa en la terminal 3 trata de leer el registro de agente 2 y se pone en espera hasta que sea liberado.

- ◆ Ambos programas están esperando indefinidamente.

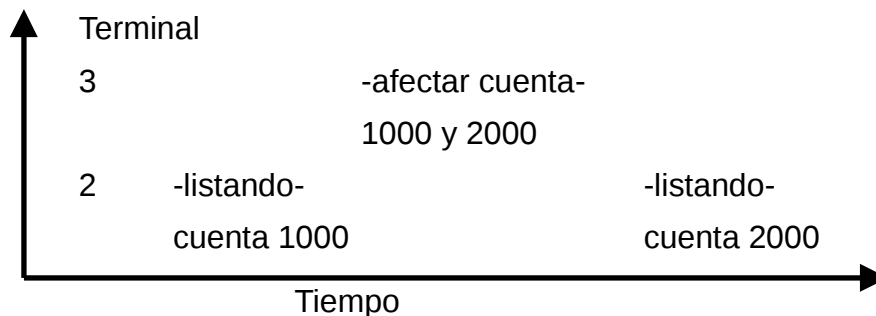
Gráficamente queda:



La solución al problema anterior es de **programación** y consiste en que **'si se tiene en una parte de un programa un registro bloqueado, no intentar bloquear otro'**, es decir, asegurarse que solamente se tenga un registro bloqueado por el programa en un mismo tiempo.

Otro problema de la **actualización simultánea** de un archivo es ocasionar una **falta de integridad en el proceso**. El siguiente ejemplo ilustra lo anterior.

- ◆ En la pantalla 2 se está generando una balanza de comprobación, se está listando la cuenta 1000.
- ◆ En la pantalla 3 se está afectando el archivo de cuentas, se hace movimiento que afecta a la cuenta 1000 y 2000.
- ◆ En la pantalla 2 se lista la cuenta 2000 que ya tiene incluido el movimiento que la 1000 no tiene, el listado quedará incorrecto (falta de integridad en el proceso). Gráficamente esto quedaría en la siguiente forma:



La solución al problema anterior es que el programa de listado tenga el archivo de cuentas de contabilidad en forma exclusiva y que ningún otro programa lo pueda acceder.

En resumen, se tienen tres problemas:

- ◆ Actualización simultánea de un registro. Solución: bloqueo a nivel registro en la actualización.
- ◆ 'Dead lock'. Solución: tener un solo registro bloqueado a un tiempo.
- ◆ Falta de integridad del proceso. Solución: tener en forma exclusiva el o los archivos.

3.2) Solución a actualización simultánea.

De los problemas del inciso anterior el de **actualización simultánea** se resuelve internamente por **SISINF**, el de 'Dead lock' es un problema de programación, por lo que se describen algunas reglas para que no ocurra.

Para la **actualización simultánea** de un mismo archivo, **SISINF** bloquea a nivel registro maestro en forma automática y si el registro que se desea bloquear lo tiene otra terminal se espera a que lo **libere**.

La forma en que trabaja este **bloqueo** automático, así como unas consideraciones que se deben tener se describen a continuación:

- ◆ La instrucción de BUSCAR DISCO una vez que se encuentra el registro buscado hace el bloqueo del registro maestro.
- ◆ La instrucción de BUSCAR DISCO una vez que se hace la **actualización** del registro maestro se **libera** el mismo.
- ◆ Cuando se ejecuta la instrucción de INSERTAR DISCO en un registro maestro, primero se hace la actualización del registro liberándose el mismo, y luego se bloquea el registro maestro que se acaba de insertar.
- ◆ Note que el bloqueo es a nivel registro maestro, por lo que NO es necesario bloquear los recurrentes, ya que estarán bloqueados en forma automática.
- ◆ Si se **busca** por un grupo de llaves en un recurrente, se busca en forma automática su maestro, por lo que si éste está bloqueado, el programa que hace la búsqueda entra en espera.
- ◆ El módulo CLS detecta para cada archivo que se usa en un programa si se ejecutan instrucciones de INSERTAR DISCO, ESCRIBIR DISCO, BORRAR DISCO o BORRAR RECURRENTE; si es así, dicho archivo se supone que está en modo **actualización**, por lo que todas las búsquedas

que se hagan bloquearán el registro, **independientemente** que se actualice o no.

- ◆ Si un programa sólo ejecuta instrucciones de LEER DISCO en un archivo se supone que está en modo **Lectura**, por lo que NO se bloquean registros al buscar.
- ◆ Es posible que un archivo lo estén accedendo varios programas en modo **actualización** y varios en modo **Lectura**. Los primeros tendrán bloqueo a nivel registro, los segundos NO.
- ◆ La opción BUSC. en BUSCAR DISCO no bloquea el registro en modo actualización, por lo que se deberá usar cuando solo se desee certificar la existencia de un registro.
- ◆ Al hacer el diseño de un archivo en la DBD es necesario tomar en cuenta el bloqueo del registro maestro, ya que si del mismo hay muy pocos y de los recurrentes dependientes hay muchos, se podrán tener problemas de operación. Un ejemplo de lo anterior sería poner a nivel registro maestro la información de compañías, y a segundo nivel (como recurrente) la información general del cliente. El resultado sería que no se podría actualizar la misma compañía desde dos terminales diferentes.

El problema de 'Dead lock' es de programación y se deberá solucionar en la forma que se diseña el programa, siguiendo la recomendación de NO tener más de un registro bloqueado a la vez.

Suponga los siguientes casos de programas:

- a) Uno o varios archivos en lectura, cero en actualización.
- b) Cero, uno o varios archivos en lectura, uno solo en actualización.
- c) Cero, uno o varios archivos en lectura, varios en actualización.

En los casos 'a' y 'b' no se tendrá problema de 'dead lock', en el caso 'c' se deberá estructurar el programa en una forma parecida a la siguiente:

* Parte I del programa.

se busca el registro maestro del archivo A.

se procesa la información del archivo A.

se libera el registro maestro del archivo A.

* Parte II del programa.

se busca el registro maestro del archivo B.
 se procesa la información del archivo B.
 se libera el registro maestro del archivo B.

* Parte III del programa.

* Parte IV del programa.

se busca el mismo registro maestro del archivo A
 que en la parte I.
 se procesa otra información en el archivo A. Se libera
 el registro maestro del archivo A.

* Parte V del programa.

Es decir, en cada **parte** del programa sólo se interactúa con **uno** de los archivos de actualización, pudiendo suceder que por las necesidades de programación se debe interactuar con el mismo registro en dos diferentes partes (I y IV).

La instrucción para liberar un registro maestro es:

OPCION DISCO NOMB LIBERAR

En donde NOMB es el nombre del archivo cuyo registro maestro se libera.

La forma general de trabajo de esta instrucción es:

- ◆ Se actualiza en disco los registros **disponibles** (registros maestro y en su caso recurrentes) que han sido modificados o borrados.
- ◆ Se libera el registro maestro.

Note que no queda disponible ningún registro del archivo en que se liberó, pero al posicionamiento en los grupos de llaves NO se altera.

3.3) Falta de integridad.

Para el problema de falta de integridad, **SISINF** tiene la facilidad de manejar archivos en forma exclusiva. Para ello se deberá usar la siguiente instrucción:

OPCION DISCO NOMB EXCLUSIVO

En donde NOMB es el nombre del archivo.

Los siguientes comentarios son aplicables:

- ◆ Esta instrucción se deberá de usar **antes** de cualquier otra en el archivo permanente.
- ◆ Si el archivo lo está usando otra terminal se pone en espera hasta que lo deje de usar.
- ◆ Si otra terminal quiere usar el archivo, se espera hasta que termine de usarlo este programa.
- ◆ La primera instrucción de archivo permanente que se ejecuta en un programa es la que **abre** el archivo (BUSCAR DISCO, INSERTAR DISCO, OPCION DISCO, etc.) por lo que se recomienda que, esta instrucción sea de las primeras de un programa para evitar 'dead lock' en espera por archivos.

4) Implantación de Archivos Permanentes.

4.1) Problemas de compartir archivos.

Para implantar los Archivos Permanentes de **SISINF** en forma compartida se tienen varias estrategias las cuales solucionan en diferente forma los tres problemas de compartición de archivos que se discuten a continuación:

- ◆ El primero de ellos consiste en que un registro maestro NO se puede modificar por dos programas de **SISINF** al mismo tiempo. Así en el ejemplo del archivo de clientes NO se puede estar actualizando el saldo del cliente en forma simultánea en dos terminales ya que quedaría mal esta información.
- ◆ El segundo problema consiste en que el mismo registro físico que puede contener uno o varios registros maestros y/o recurrentes, NO puede estar en buffers locales por varios programas para ser modificado al mismo tiempo, ya que si un programa lee de disco a un buffer local el registro físico del R0 de un cliente, lo modifica y luego lo escribe a disco y si otro programa hace lo mismo en el mismo tiempo, la información estaría incorrecta.
- ◆ Por último se tiene el problema de operaciones completas. Para explicarlo suponga el caso de cuando se inserta una llave, esta operación puede requerir de modificar varios registros físicos. Entonces si un programa está haciendo una inserción y otro una búsqueda, puede este último encontrar información inconsistente si la hace al mismo tiempo. Por lo anterior la operación se debe hacer completa antes que otro programa intente otra operación.

4.2) Generación de pedidos.

El módulo ELS genera, para ejecutar las diferentes instrucciones de archivos permanentes, unos pedidos los cuales serán procesados de acuerdo a la forma de implantar este tipo de archivos. Así para una instrucción de BUSCAR DISCO se generan dos pedidos, uno para buscar en el directorio del grupo de llaves de acuerdo a la opción y otro para hacer la lectura del registro.

Básicamente los pedidos son:

- 4.2) Generación de pedidos.

- ◆ Leer un registro lógico en una cierta posición.
- ◆ Escribir un registro lógico a partir de cierta posición.
- ◆ Insertar un registro lógico en la próxima posición disponible del archivo.
- ◆ Buscar en un directorio por una cierta llave de un grupo.
- ◆ Insertar en un directorio una llave.

4.3) Estrategia de Tarea Central.

Bajo esta estrategia de implantar Archivos Permanentes se tiene una tarea central llamada PEVX que procesará los diferentes pedidos de los diferentes ELS que se estén ejecutando (donde X es el número de la versión de **SISINF**)

La comunicación entre ELS y el PEVX es con un archivo de mensajes (Mail Box) o bien con memoria compartida y semáforos, ...

Con esta estrategia se resuelven todos los problemas ya que todos los registros físicos los tiene la tarea central y los pedidos se hacen de uno en uno por lo que las operaciones son completas. También esta tarea se encarga de que un R0 solo se modifique por un solo ELS al mismo tiempo, poniendo a los demás en espera.

4.4) Archivo en Memoria Virtual Compartida.

Esta estrategia consiste en tener el archivo en Memoria Virtual Compartida de forma que varios ELS pueden acceder esta memoria al mismo tiempo como si fuera un vector.

El módulo ELS tiene una subrutina llamada PEVM que es la que procesa los diferentes pedidos.

En esta forma de trabajo si varios ELS tienen el archivo, todos comparten el mismo vector y por lo tanto los registros físicos que forman el archivo, solo se tendrá una copia del vector.

Para lograr operaciones completas se necesita que el Sistema Operativo tenga un mecanismo de LOCK/UNLOCK. Con este mecanismo se logra tener el archivo en forma exclusiva para hacer la operación por el intervalo de tiempo

que sea necesario.

Por último para que solo un ELS tenga el mismo R0 de modificación se usará también el mecanismo de LOCK/UNLOCK o bien se usará el primer registro del archivo para tener tablas internas.

4.5) Buffers Locales.

Bajo esta estrategia, el Módulo ELS maneja los registros físicos en buffers locales en una subrutina llamada PEVBL la cual procesa los diferentes pedidos haciendo un LOCK al archivo, leyendo el o los registros físicos necesarios, luego se hacen las modificaciones que se piden, se escriben en disco los registros modificados y se hace un UNLOCK al archivo.

Las operaciones son completas ya que el archivo se tiene en forma exclusiva por el tiempo que dure el proceso del pedido, dado que los registros físicos se leen y escriben si son modificados también se elimina este problema. Por último se tienen las mismas políticas que en Memoria Virtual para el mismo R0 de modificación.

Para disminuir la lectura de registros físicos sobre todo cuando varias terminales solo están leyendo el archivo, se tiene el concepto de Buffer Válido/Inválido. Bajo este concepto, los buffers de ELS son válidos mientras otro ELS no los modifique en cuyo caso se hacen inválidos. Con esta estrategia primero se revisa si los buffer son válidos y si son se evitan las lecturas de aquella información que ya tiene ELS en sus buffers locales.

4.6) Estructura de los Archivos Permanentes.

Cada Archivo Permanente consta de un área en disco, que puede visualizarse como un conjunto de registros físicos o blocks (1, 2, N).

Estos blocks se agrupan en cuatro áreas:

- ◆ Area de Control.
- ◆ Area de TIP.
- ◆ Area de Directorios.
- ◆ Area de Datos.

Area de Control.

El Area de Control, que usa los registros 1 y 2 del archivo, contiene:

- ◆ El registro 1 se usa en algunas máquinas para el LOCK/UNLOCK del R0.
- ◆ El registro 2 contiene:
 - ◆ Tabla de apuntadores al inicio de cada área de directorio.
 - ◆ Apuntador al inicio del área de datos.
 - ◆ Apuntador al último registro de datos.
 - ◆ Suma de la información.

Si por algún motivo este registro se destruye o tiene información errónea, al tratar de usarlo se desplegará un diagnóstico; en este caso el archivo deberá reconstruirse.

Area de TIP.

El área de registros TIP sirve para verificar la consistencia en el uso del archivo por un programa, es decir, verificar que la “imagen” de DBD con que el archivo fue creado, sea la misma que la del programa que lo va a usar.

El área de TIP consta de tantos registros físicos, como tipos de recurrente (incluyendo R0) tenga el archivo.

La verificación de consistencia se logra de la siguiente forma:

- ◆ Cuando un programa se compila, en su imagen ejecutable (ESxxyy) se generan tablas de información de TIPOS, una por cada tipo de registros de cada archivo que **use** el programa.
- ◆ Cuando al ejecutar un programa se usa un archivo, que NO EXISTE, (es decir, se usa por primera vez, o después de purgarlo físicamente), el archivo se CREA. Una parte del proceso de creación consisten en grabar los registros TIP, cuya información se toma de las tablas de tipos que trae el programa para el archivo.
- ◆ Cuando al ejecutar un programa se usa un archivo que ya existe, al momento de “abrir” el archivo se efectúa la revisión de tipos, es decir, se verifica que la información de tipos del archivo coincida con la de las tablas de información de tipos que trae el programa.

Si esta información **no** coincide, se diagnostica y se suspende la ejecución del programa.

Las razones más comunes por la que se presenta este error son:

- a) El programa fue compilado con una DBD a la que se le hicieron cambios.
- b) El archivo fue reconstruido, haciendo cambios, pero éstos no se reflejaron en la DBD y/o en los programas.
- c) Hubo destrucción (total o parcial) del área de TIPOS en el archivo, por lo que hay que reconstruirlo.

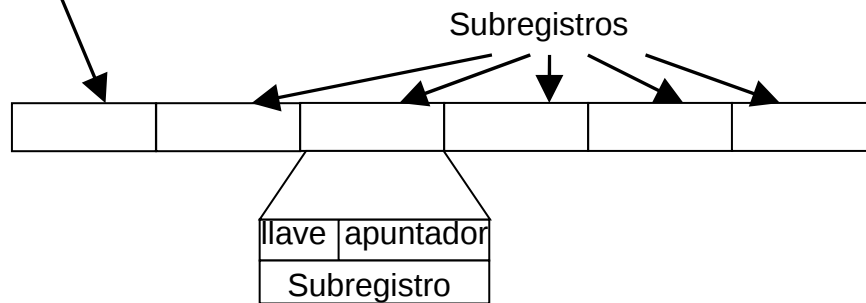
Area de Directorios.

Los directorios tienen las siguientes características:

- ◆ Se tiene un directorio por cada grupo de llaves que tenga el archivo.
- ◆ La estructura de los directorios es la llamada de árbol.
- ◆ Cada registro de directorio contiene un cierto número máximo de subregistros.
- ◆ El subregistro contiene el valor de cada variable del grupo de llaves y apuntador.
- ◆ Un directorio tiene "n" niveles, numerados 1, 2, 3, 4, ... El 1 es el superior, el último es el inferior.
- ◆ Del nivel superior solo hay un registro de directorio.
- ◆ El apuntador en el subregistro es a otro registro del directorio o en el caso de los registros inferiores al área de datos.
- ◆ Los registros inferiores tienen todas las llaves de ese directorio.

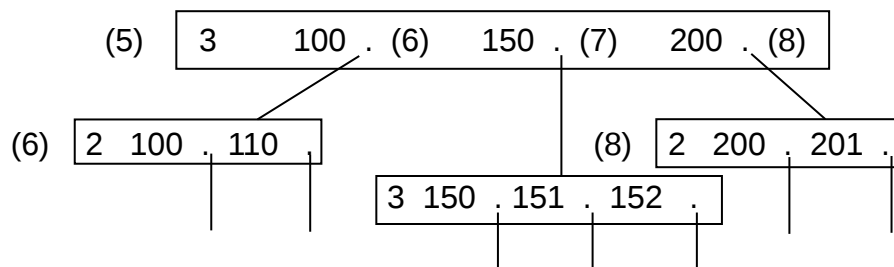
Gráficamente la estructura de un registro es:

Número de subregistros usados en este registro



La razón de llamarse estructura de árbol es la analogía, ya que el tronco sería el nivel superior, dicho tronco se divide en varias ramas, las cuales serían el nivel 2, estas ramas a su vez se dividen en otras, las cuales se el nivel 3 y así sucesivamente hasta llegar a las hojas que sería el nivel inferior. Por convención se designa nivel superior al tronco y nivel inferior a las hojas aunque esto sea contrario a la realidad en el caso de un árbol.

Suponga que se tiene como llave una variable N1 de 0 decimales; gráficamente el directorio para unos valores supuestos sería:



Los siguientes comentarios son aplicables:

- ◆ El número del registro de directorio se encierra entre paréntesis.
- ◆ El registro de directorio superior es uno solo y tiene por número el 5.
- ◆ Este directorio es de 2 niveles.
- ◆ El nivel inferior está formado por los registros 6, 7, 8.
- ◆ El primer subregistro del nivel superior contiene un 100 y el apuntador al registro inferior que es (6). El 100 es la primera llave del registro 6.
- ◆ El segundo subregistro del nivel superior contiene un 150 y el apuntador al

registro inferior que es (7). El 150 es la primera llave del registro 7.

- ◆ El registro número 6 contiene 2 subregistros, la llave 100 y su apuntador al área de datos y la llave 110 y su apuntador a dicho área de datos.

Descrito en otra forma, en el nivel inferior se tienen todas las llaves del directorio con su apuntador al área de datos, en el siguiente nivel hacia arriba, se tiene la primera llave de cada registro del nivel inferior y el apuntador al registro que pertenece. Si todas las llaves caben en un solo registro, este es el superior.

La instrucción de BUSCAR DISCO con opción PRIN. con un valor de llave 151, para el ejemplo que se describe como:

- ◆ Leer el nivel superior, en este caso el registro 5.
- ◆ Buscar entre que dos llaves están el 151, siendo entre 150 y 200. Tomar el apuntador de la menor, que en este caso es 7 y leer este registro.
- ◆ Dado que el registro 7 está en el nivel inferior, buscar por una llave igual a 151.
- ◆ Con el apuntador al área de datos, se lee el registro con llave 151.

Dado que el nivel inferior contiene todas las llaves del directorio, la instrucción de BUSCAR DISCO con opción SEC usa solo dicho nivel, para ello todos los registros de directorio de nivel inferior están apuntados. Así aunque NO aparece en la gráfica el registro 6 apunta al 7, el 7 al 8 y el 8 tiene una marca de fin.

El algoritmo de inserción trabaja de la siguiente forma:

- ◆ Buscar la llave que se inserta tal como ya se describió, hasta llegar al nivel inferior.
- ◆ Si ya existe diagnosticar ya que nos se pueden tener llaves duplicadas.
- ◆ Agregar en el registro de directorio la llave con su apuntador, si no hay lugar partir el registro en dos y modificar el nivel superior.
- ◆ Si se parte un registro de directorio y el nivel superior está también lleno, es necesario partir también el superior.

El número máximo de subregistros en cada registro de directorio depende de la longitud de la llave y de la longitud de apuntadores. En la mayoría de los casos es mucho mayor que el ejemplo que se describió.

- 4.6) Estructura de los Archivos Permanentes.

Una pregunta que se puede hacer es Cuantos niveles son necesarios? Para ello suponga una llave N1, y que el número máximo de subregistros en alguna máquina es de 80. Entonces en la mejor situación se tendrán 80, en la más desventajosa cuando todos los registros estén partidos se tendrá 40 por lo que:

NIVEL	1	40	80
NIVEL	2	1,600	6,400
NIVEL	3	64,000	512,000
NIVEL	4	2,560,000	40,960,000

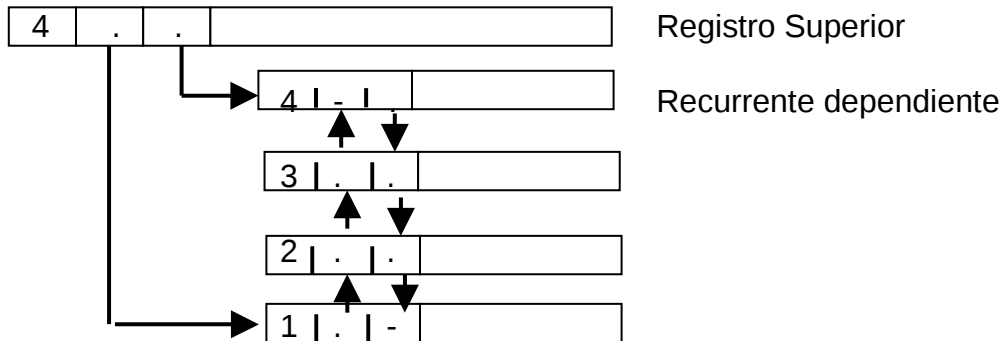
Un archivo con 2 ½ millones de registros necesita en la situación más desventajosa de 4 lecturas de registros de directorios para poder localizarlos.

Por último, se tiene el área de datos en la que los registros R0 y recurrentes se insertan en forma secuencial de forma que el módulo ELS se encarga del “bloqueo y desbloqueo” de los mismos. Gráficamente se tendría por ejemplo:

REGISTROS LOGICOS

R2	R0	R1	R1	R1	R2	R1	R1	R1	R0	R1
101	102	103	104							

Los recurrentes están ligados a su dependiente por apuntadores. Gráficamente se tendría por ejemplo:



En el registro superior se tiene el número de recurrentes que para este ejemplo es de 4 y en los dependientes se tiene la secuencia.

En resumen los archivos permanentes tienen cuatro áreas que gráficamente se puede representar así:

Registro	Registro	Directorio	Directorio	
2	TIP	LL 1	LL2	



La estructura de archivos descrita **permite tiempos de respuesta bastante buenos**, tanto en inserción al azar como en búsqueda secuencial.

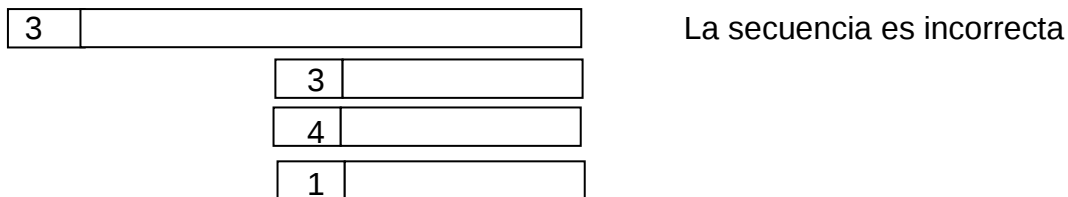
4.7) Errores en Archivos Permanentes.

El módulo ELS hace una serie de revisiones cuando hace operaciones con los Archivos Permanentes, dichas revisiones hacen que la operación sea más confiable. Si por algún motivo se encuentra algo incorrecto se diagnostica y se cancela el trabajo. A continuación se describen los errores más comunes y su explicación.

- ◆ NUMERO DE SECUENCIA INCORRECTO

Al ejecutar una instrucción de BUSCAR RECURRENTE la secuencia de los registros recurrentes no es la correcta.

Gráficamente:

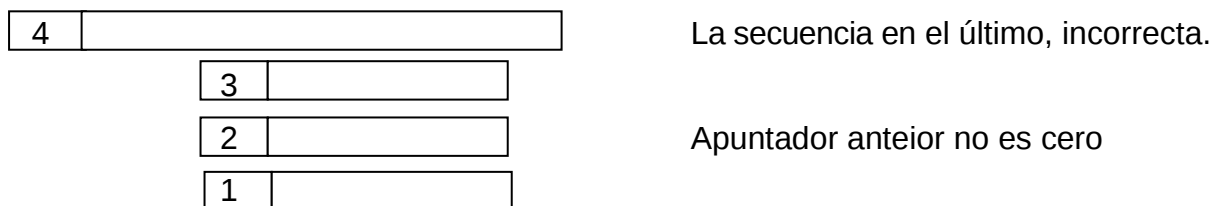


- ◆ APUNTADOR DE ANTERIOR NO ES CERO.

- ◆ SECUENCIA EN EL ULTIMO, INCORRECTA.

Cuando se inserta un recurrente se revisa que el apuntador para localizar el anterior sea cero y que la secuencia de este recurrente corresponda con el número de recurrentes que debe haber.

Gráficamente:

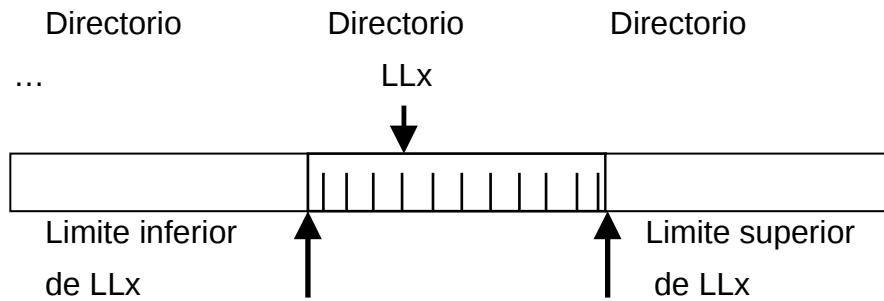


- ◆ FUERA DE LIMITES AL LEER DIRECTORIO.

◆ FUERA DE LIMITES AL ESCRIBIR DIRECTORIO.

Quando se leen los registros de directorios o cuando se escriben, para efectuar las instrucciones de BUSCAR o de INSERTAR, se revisa que el número del registro de directorio esté en los límites de ese directorio

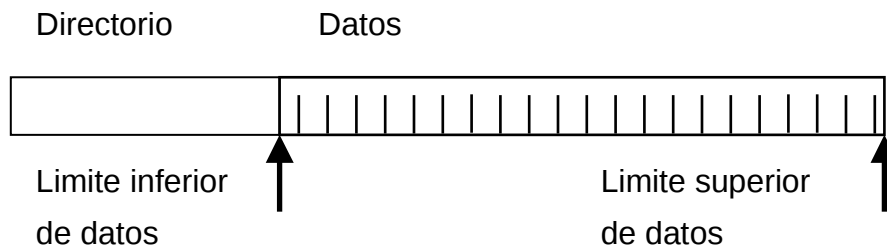
Gráficamente:



◆ FUERA DE LIMITES DE DATOS

Quando se lee o re-escribe un registro maestro o recurrente se revisa que esté en los límites del área de datos.

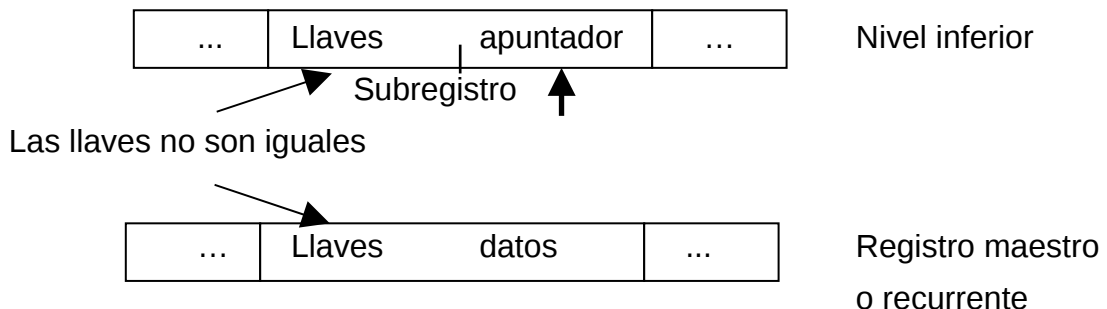
Gráficamente:



◆ LLAVES EN REG-DIR DIFERENTES

Al ejecutar BUSCAR DISCO las llaves en el directorio y en el registro no son iguales.

Gráficamente:



Las causas de estos errores son:

- ◆ Problemas de Hardware de la máquina, ya sea en los discos o en la memoria.
- ◆ Problemas en el archivo por fallas en el servicios de energía eléctrica o por caídas en el sistema operativo, que ocasiona que no se escriban los “buffers” de los archivos.
- ◆ Problemas del módulo ELS. Es posible esta situación si el error ocurre con mucha frecuencia en el mismo programa.

5) Tiempo Usado.

5.1) Introducción

En cualquier aplicación se tendrán programas de captura, programas de consulta, programas de actualización o cálculos y programas de reportes.

En los primeros tipos (captura y consulta) el **tiempo de respuesta** es un factor muy importante; en los dos últimos (cálculo y reportes), el tiempo de proceso es el factor importante.

En los programas de captura es necesario definir el nivel de validación que se tendrá, ya que dicho nivel afecta a los programas de actualización; así se puede tener los siguientes niveles de validación:

a) Se valida solamente por numérico, dígito, rango, se graba en un archivo secuencial la información.

b) Se hacen las validaciones especificadas en el inciso A y se revisa contra archivos para validar la información, la cual se graba en un archivo secuencial o en un recurrente de movimientos.

c) Se hacen las validaciones especificadas en el inciso A, se revisa contra archivos y se hace la afectación indicada por los movimientos.

Todos los niveles se pueden tener en **SISINF**, pero es necesario tomar en cuenta los siguientes comentarios:

- ◆ Si se usa la opción A será necesario validar contra archivos por medio de programas posteriores. Por experiencia en varios sistemas, el tiempo total de proceso (incluyendo los procesos posteriores) en la opción A es mayor que la opción B.
- ◆ En la opción B y C el problema de tiempos de respuesta es muy importante.
- ◆ La opción B tiene la ventaja de poder responsabilizar al usuario de la captura de la información y así, cuando está totalmente validada, ejecutar por el personal de operación el resto del proceso.
- ◆ La opción C se recomienda para aquellos casos en que se desea consultar

en cualquier momento la información ya actualizada.

Dado que el tiempo de respuesta y el tiempo de proceso son factores importantes en la operación de un sistema, en este capítulo se cubrirán ambos tópicos.

5.2) Tiempo de Respuesta.

El tiempo de respuesta se define para propósitos de este manual como el tiempo que se tarda la máquina desde que se oprime la tecla de "return" para que lea la información hasta que se posiciona el cursor para teclear la siguiente información en captura o se despliega la información pedida en consulta, es decir, el tiempo que la máquina se tarda en atender una "transacción".

El tiempo de respuesta dependerá de muchos factores, algunos de los cuales dependen, del equipo y otros de la forma de programar en **SISINF**. Para aclarar más estos factores se presenta una serie de casos relacionados con la captura de información.

- ◆ - El tiempo que se tarda entre la lectura de las diferentes variables de una instrucción de LEER TERMINAL dependerá del nivel de paginación o "swapping" que se tenga en el equipo en ese momento.
- ◆ - El tiempo que se tendrá entre dos instrucciones de LEER TERMINAL si entre ambas se tienen instrucciones que usen exclusivamente CPU y el número de estas es razonable, será igual al del caso A. Si entre ambas se tiene instrucciones de disco ya sea permanente o temporal se incrementa el tiempo de acuerdo a lo descrito a continuación: Usar instrucciones de LEER TERMINAL sin otras instrucciones intermedias, ocasiona el cambio de letrero de EJECUTANDO Y LECTURA en forma innecesaria.
- ◆ - Para implantar las instrucciones de archivos TEMPORALES **SISINF** se usan las facilidades que tenga el sistema operativo para la lectura y escritura de archivos secuenciales. Si el sistema operativo así lo permite, ELS escogerá el número de buffers y tamaño de block en forma óptima. En la situación más desventajosa se hará una operación de entrada o salida en disco por cada instrucción de TEMPORAL; En la situación más ventajosa se hará una operación de entrada o salida en disco por varias instrucciones de TEMPORAL. Este número dependerá del modelo y marca de computadora.
- ◆ La instrucción de BUSCAR DISCO origina tantos pedidos a PEVX como sea el nivel del registro más uno adicional. Así para una R0 origina 2

pedidos a PEVX.

- ◆ La instrucción de INSERTAR DISCO origina un pedido a PEVX por cada grupo de llaves que tenga el registro, un pedido adicional para la inserción del mismo y en el caso de recurrentes uno más.
- ◆ La instrucción de BUSCAR RECURRENTE sin variables origina un pedido a PEVX. La instrucción de BUSCAR RECURRENTE con variables origina tantos pedidos a PEVX hasta que la comparación sea igual (exitosa), o salga por el EJECUTA.
- ◆ En los casos de D a F será necesario conocer la relación de pedidos a PEVX y las lecturas o escrituras físicas del disco. Por experiencia en varias instalaciones se ha visto una relación de 3 para cuando se tienen programas de captura y de 3 a 10 cuando se tiene programas de proceso. La relación en estos últimos crece si se tienen muchos BUSCAR RECURRENTE. En caso en que por la forma del programa se está trabajando con los mismos registros físicos, la relación puede subir hasta 25 de acuerdo a lo observado. Los datos anteriores son observados y pueden cambiar positiva o negativamente en cada instalación.

Si entre dos LEER TERMINAL hay problemas de tiempo de respuesta, con lo descrito anteriormente se puede estimar el número de lecturas o escrituras que se hacen a disco y tratar de reducirlas en los puntos claves con cambios en las validaciones o en el diseño de los archivos.

Lo anterior determina en muchos casos el grado de actualización que se tendrá en los programas de captura sobre todo en aquellos sistemas que no es necesario actualizar en dicho momento.

Si con una buena programación y buen diseño de archivos aún se tienen problemas de tiempos de respuesta, se puede deber a un nivel alto de paginación o “swapping”, o a un nivel alto de actividad de disco (lecturas o escrituras). Los siguientes comentarios son aplicables:

- ◆ El problema de paginación o “swapping” excesivo se deberá obviamente a falta de memoria. Se recomienda consultar a personal de su distribuidor para una estimación de las necesidades del usuario.
- ◆ Si se mezclan programas de captura o consulta con programas de proceso o compilación y se asigna la misma prioridad, se tendrán problemas de tiempo de respuesta. La solución es que los problemas de proceso tengan menos prioridad
- ◆ Si se tiene un nivel alto de actividad en disco por lecturas y escrituras se

deberá solucionar por medio de hardware (otro controlador) o bien repartiendo mejor la carga entre las diferentes unidades de disco.

5.3) Tiempo de Proceso.

Una vez que se captura la información será necesario ejecutar los programas para hacer las actualizaciones y posteriormente sacar los listados necesarios. En ambos tipos de programas el tiempo de proceso de los mismos es importante por lo que un buen uso de las instrucciones de **SISINF** puede reducirlos. En esta sección se describe el consumo en recursos de cada instrucción.

Cuando se programa una aplicación o cuando se desea optimizarla, puede haber la duda acerca de cual es la forma más conveniente de usar las instrucciones, de tal manera que el tiempo de proceso sea mínimo. Dado que se pueden tener muchas combinaciones con resultados iguales, se describen a continuación ciertas recomendaciones:

- ◆ En la instrucción de FIN se cierran los Archivos Permanentes y Temporales que se usaron, se actualiza la contabilidad de **SISINF** y en la forma opcional se carga otro programa. Por el comentario anterior no es recomendable que al dividir un programa en dos o más queden de forma en que se ejecute el primero, luego el segundo, otra vez el primero luego el segundo, etc. La mejor forma es dividirlos de forma que se ejecute el primero, luego el segundo y se termine el proceso, ya que de esta forma la instrucción de FIN se ejecuta dos veces y en la forma descrita primeramente se ejecuta muchas veces.
- ◆ La instrucción de CONVERTIR está pensada para situaciones muy especiales en que por problemas de diseño o de nuevos requerimientos era necesario sacar la información caracter por caracter. Se usa también en otros casos especiales como cálculo del dígito, algunos formatos de impresión, etc. El tiempo de CPU usado en comparación con una instrucción de MOVER es mayor debido las validaciones y conversiones que son necesarias en la ejecución interna de la instrucción. Si se usa en forma indiscriminada se deberá pensar en el factor tiempo.
- ◆ La instrucción de LLAMAR involucra cerrar el archivo temporal, abrir el archivo del programa, cargarlo a memoria y pasarle la información del temporal. Si se usa dentro de un ciclo en el programa, se deberá pensar en el tiempo de ejecución necesario para efectuar las operaciones descritas.
- ◆ La instrucción de DETALLE necesita de un "buffer" especial en que se

forma la línea de impresión, el cual se mueve al “buffer” de salida con la instrucción de IMPRIMIR DETALLE. El uso de esta instrucción cuando se puede usar solamente IMPRIMIR DETALLE usa más tiempo de CPU.

- ◆ Se recomienda usar las instrucciones de SUMAR y RESTAR para el manejo de fechas y no tratar de simularlas con las demás instrucciones.
- ◆ La instrucción de BUSCAR TABLA compara en forma secuencial el valor específico contra la tabla. Por el comentario anterior se usará menor tiempo en búsquedas por igual en tablas grandes si se ponen los elementos más buscados en los primeros renglones.
- ◆ La instrucción de BUSCAR DISCO se puede usar en forma eficiente o ineficiente. En el capítulo de Instrucciones de Archivos Permanentes se ejemplifica sus usos para las diferentes opciones en forma eficiente e ineficiente. Se deberá revisar que sea eficiente.
- ◆ Cuando un programa no necesita de archivos compartidos se recomienda se ponga la instrucción de OPCION DISCO EXCLUSIVO, ya que en archivos compartidos es necesario hacer más revisiones que en exclusivo, lo que redundaría en mayor consumo de tiempo de CPU, así como más lecturas y escrituras a disco.
- ◆ Es mejor usar AREAS COMUNES que archivos temporales para pasar la información entre programas.

Es lógico que se comparen los tiempos de ejecución de programas hechos en **SISINF** con programas hechos en otros Lenguajes que generan código. Los siguientes comentarios son aplicables:

- ◆ El módulo ELS es un interpretador, pero como ya se especificó, esta interpretación no es identificar la instrucción ni las variables cada vez que se pasa por ella, sino que el módulo CLS deja un código especial que hace que el módulo ELS sea lo más rápido posible.
- ◆ El tiempo de CPU usado en las instrucciones de SUMAR, RESTAR, manejo de terminales e impresión es igual al que se obtendría de hacer lo mismo en otros Lenguajes que sean compiladores. Lo anterior se debe a que el número de instrucciones usadas en interpretar la instrucción comparando con las instrucciones necesarias para hacer lo indicado son mínimas.
- ◆ El tiempo de CPU usado por la interpretación de las instrucciones de TEMPORAL es también muy pequeño en comparación con lo que necesita hacer el sistema operativo para ejecutar la instrucción.

- ◆ El tiempo de CPU usado en las instrucciones de manejo de archivos permanentes es mínimo comparado con los tiempos de lectura o escritura en disco.
- ◆ El tiempo de CPU usado en la ejecución de las instrucciones Aritméticas y SI numérico, si la máquina NO tiene hardware de **aritmética decimal** es igual al que se obtendría en otros lenguajes ya que en ambos casos es necesario simular las instrucciones.

En algunos casos se puede obtener mucho mejores tiempos en **SISINF** ya que la aritmética usada es binaria y no es necesario hacer rutinas generales.

- ◆ El tiempo de CPU usado en la ejecución de las instrucciones aritméticas y SI numérico si la máquina sí tiene hardware de **aritmética decimal** es mayor en **SISINF** que en lenguajes que sean compiladores. Si en Lenguaje que son compiladores se tienen expresiones aritméticas cuyas variables tengan diferente longitud y decimales, esta diferencia puede ser mínima, también en el caso de hacer muchas divisiones y no tener esta instrucción por hardware.
- ◆ El tiempo de CPU usado en la ejecución de las instrucciones de MOVER, SI alfanumérico y EJECUTA será mayor en **SISINF** que en un lenguaje que sea compilador solo por el costo de interpretación.
- ◆ Por experiencia en sistemas trabajando en máquinas que SI tienen hardware de aritmética decimal, el tiempo total de proceso de sistemas en **SISINF** es igual al obtenido con otros lenguajes que sean compiladores.

5.4) Estadísticas de Uso.

La opción de Estadísticas de Uso que forma parte de la Contabilidad de uso de **SISINF** ayuda a hacer la optimización de sistemas en **SISINF** y hacer uso adecuado de los recursos computacionales.

Algunos de los problemas señalados en este capítulo pueden ser detectados mediante el uso de esta opción.

Los pasos a seguir para hacer uso de esta opción son:

- ◆ Se activa la Contabilidad de Uso de **SISINF**.
- ◆ Se saca un listado que contiene aquellos programas con mayor consumo de CPU.

- ◆ A dichos programas se les pone la opción de EST-USO en la instrucción de DOCUMENTACION y se compilan.
- ◆ El módulo ELS para aquellos programas que tengan EST-USO graba información con datos de la ejecución en el ESxxyy.
- ◆ Se ejecuta el PSDCA0 para obtener listados de estadísticas de uso.
- ◆ Se examinan los listados y en caso de problemas, se procede a su corrección.

El menú para hacer uso de esta opción es el PSDCA0. Personal de su distribuidor puede asesorar sobre el uso de esta opción. Es importante señalar que se puede imprimir una explicación de esta opción, así como de los listados. Para ello es necesario ejecutar el programa PSDCA0.

6) Errores de Ejecución.

A continuación se listan los posibles errores en ejecución y las posibles causas que pueden originarlos durante la ejecución.

ERROR	Posible Causa
075 ERROR AL ESCRIBIR PERMANENTE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Falta área en disco. ◆ Revisar código reportado por sistema operativo.
076 ERROR AL LEER PERMANENTE nomb (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por sistema operativo.
077 ERROR DEL PAGINADOR PROBLEMAS DE HARDWARE O SISINF.	<ul style="list-style-type: none"> ◆ Reiniciar sistema. ◆ Revisar si hay fallas de hardware. ◆ Si se repite, reportar a su distribuidor.
078 FUERA DE LIMITES DE DATOS EN nomb, PROBLEMAS EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo.
079 NO HAY ESPACIO EN LOCK (C) LIMITE MAXIMO.	<ul style="list-style-type: none"> ◆ Revisar Operación por cancelar ELS. ◆ Reiniciar sistema. ◆ Si se repite reportarlo a su distribuidor.
080 YA ESTA EN TABLA DE LOCK PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Reiniciar sistema. ◆ Revisar si hay fallas de hardware. ◆ Si se repite, reportar a su distribuidor.

ERROR	Posible Causa
081 NO ESTA EL REGISTRO EN LOCK, PROBLEMAS DE HARDWARE.	<ul style="list-style-type: none"> ◆ Revisar Sistema. ◆ Revisar si hay fallas de Hardware. ◆ Si se repite, reportar a su distribuidor.
082 NO HAY MARCA EN PEV, PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar que al ejecutar PEVX sea el PEV de la Versión que se ejecuta. ◆ Reiniciar Sistema. ◆ Recompilar Programa. ◆ Revisar si hay fallas de Hardware. ◆ Si se repite, reportar a su distribuidor.
083 ERROR EN SECUENCIA EN nomb PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
084 REGISTRO DOS DEL ARCHIVO INCORRECTO EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS. ◆ Si se accesa el archivo por un usuario de escritura y varios de lectura en el mismo R0 puede dar este error. Si se repite con frecuencia cambiar los usuarios de lectura a escritura.
085 SECUENCIA DEL ANTERIOR INCORRECTA EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.

ERROR	Posible Causa
086 APUNTADOR ANTERIOR NO ES CERO EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
087 APUNTADOR DE REG-FIS INVALIDO EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
088 NO HAY AREA EN EL ARCHIVO PARA HACER LA INSERCIÓN EN nomb.	<ul style="list-style-type: none"> ◆ Reconstruir para recuperar espacio de bajas o ampliar tamaño.
089 FUNCION INVALIDA EN PEV, PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de Hardware. ◆ Si se repite, reportar a su distribuidor. ◆ Revisar Sistema. ◆ Recompilar Programa.
090 LLAVES NEGATIVAS EN INSERTAR EN nomb, (C) NUMERO DEL GRUPO DE LLAVES.	<ul style="list-style-type: none"> ◆ Error de programación. Las llaves numéricas NO pueden ser negativas y las X no pueden tener caracteres especiales.
091 NGLL O NDNV INVALIDO AL INSERTAR, PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de hardware. ◆ Si se repite reportar a su distribuidor.
092 MAS DE 100 NIVELES AL INSERTAR EN nomb, PROBLEMAS EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.

ERROR	Posible Causa
093 FUERA DE LIMITES DE DIRECTORIO EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
094 FUERA DE REGISTRO AL INSERTAR EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
095 SE INSERTA CON LLAVES DUPLICADAS EN nomb, (C) NUMERO DEL GRUPO DE LLAVES.	<ul style="list-style-type: none"> ◆ Error de programación. No se hizo BUSCAR DISCO antes de INSERTAR para revisar si existían las llaves.
096 LLAVES DUPLICADAS EN SUPERIOR EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruir o bajar respaldo. ◆ Revisar Operación por cancelar ELS
097 SE EXCEDE CAPACIDAD EN DIRECTORIO EN nomb, (C) NUMERO DEL GRUPO DE LLAVES.	<ul style="list-style-type: none"> ◆ Reconstruir para recuperar espacio de bajas o ampliar tamaño.
098 ARCHIVO INVALIDO (C) NUMERO DEL ARCHIVO.	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de hardware. ◆ Si se repite, reportar a su distribuidor. ◆ Revisar sistema.
099 DIRECTORIO INCONSISTENTE EN nomb.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo.

ERROR	Posible Causa
100 NGLL O NDNV INVALIDO AL BUSCAR PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de hardware. ◆ Si se repite reportar a su distribuidor. ◆ Revisar Sistema. ◆ Recompilar Programa.
101 MAS DE 100 NIVELES AL BUSCAR EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo.
102 FUERA DEL REGISTRO AL BUSCAR EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
103 MAS DE 100 REGISTROS AL POSICIONAR EN nomb, PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
104 SE EXCEDE EL NUMERO DE USUARIOS, (C) LIMITE MAXIMO DE USUARIOS.	<ul style="list-style-type: none"> ◆ Reiniciar Sistema. ◆ Si se repite reportarlo a su distribuidor.
105 USUARIO INVALIDO EN PEV, PROBLEMAS DE HARDWARE O DE SISINF .	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de hardware. ◆ Si se repite reportarlo a su distribuidor.
106 SE EXCEDE MAXIMO DE ARCHIVOS (C) LIMITE MAXIMO DE ARCHIVOS.	<ul style="list-style-type: none"> ◆ Reiniciar sistema. ◆ Si se repite reportarlo a su distribuidor.

ERROR	Posible Causa
107 ERROR AL ABRIR PERMANENTE EN nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar directorio, área en disco, permisos, recursos del sistema. ◆ Revisar código reportado por el Sistema Operativo.
108 ERROR AL HACER LOCK DE REGISTRO PERMANENTE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
109 ERROR AL CERRAR PERMANENTE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
110 LONGITUD AL RECIBIR DE PEV, NO ES LA ESPERADA, PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar instalación de SISINF en lo referente al PEVX. ◆ Revisar si hay fallas de hardware. ◆ Si se repite reportar a su distribuidor.
111 NO HAY MARCA DE MENSAJE EN ELS AL RECIBIR DE PEV, PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Revisar la instalación de SISINF en lo referente al PEVX. ◆ Revisar si hay fallas de hardware. ◆ Si se repite reportar a su distribuidor.
112 ERROR DE COMUNI-CACION (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar instalación de SISINF en lo referente o PEVX.
113 NO SE PUEDE 'LEER DISCO' EN nomb, NO SE HA BUSCADO O INSERTADO.	<ul style="list-style-type: none"> ◆ Error de programación.
114 NO SE PUEDE 'LEER DISCO' EN nomb, EL REGISTRO ESTA BORRADO.	<ul style="list-style-type: none"> ◆ Error de programación.

ERROR	Posible Causa
115 NO SE PUEDE 'ESCRIBIR DISCO' EN nomb, NO SE HA BUSCADO O INSERTADO.	♦ Error de programación.
116 NO SE PUEDE 'ESCRIBIR DISCO' EN nomb, EL REGISTRO ESTA BORRADO.	♦ Error de programación.
117 NO SE PUEDE 'INSERTAR DISCO' EN nomb, NO SE HA BUSCADO O INSERTADO EN NIVEL SUPERIOR.	♦ Error de programación.
118 NO SE PUEDE 'INSERTAR DISCO' EN nomb, REGISTRO EN NIVEL SUPERIOR, BORRADO.	♦ Error de programación.
119 SE TRATA DE INSERTAR REC. OPCION M Y YA NO HAY LUGAR EN nomb.	♦ Error de programación.
120 SE CICLO EN TABLA INTERNA, PROBLEMAS DE HARDWARE O SISINF.	<ul style="list-style-type: none"> ♦ Recompilar programa. ♦ Revisar fallas de hardware. ♦ Si se repite, reportar a su distribuidor.
121 NO SE PUEDE 'BORRAR' EN nomb, NO SE HA BUSCADO O INSERTADO.	♦ Error de programación.
122 NO SE PUEDE 'BORRAR' EN nomb, YA ESTA BORRADO.	♦ Error de programación.
123 NO SE PUEDE 'BUSCAR RECURRENTE' EN nomb, NO SE HA BUSCADO O INSERTADO NIVEL SUPERIOR.	♦ Error de programación.
124 NO SE PUEDE 'BUSCAR RECURRENTE' EN nomb, SE BORRO EL NIVEL SUPERIOR.	♦ Error de programación.

ERROR	Posible Causa
125 LLAVES EN REG-DIR DIFERENTES EN nomb, PROBLEMA EN EL PROBLEMA EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo.
126 ERROR EN TABLA DE REGISTROS PROBLEMAS DE HARDWARE O SISINF .	<ul style="list-style-type: none"> ◆ Recompilar programa. ◆ Revisar si hay fallas de hardware. ◆ Si se repite, reportar a su distribuidor. ◆ Recompilar el programa.
127 ERROR AL BORRAR PERMANENTE EN nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
128 ERROR AL CREAR PERMANENTE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar directorio, área en disco, permisos, recursos del sistema.
129 EL ARCHIVO nomb, NO SE PUEDE BORRAR, SE ESTA USANDO POR OTRO USUARIO.	<ul style="list-style-type: none"> ◆ Revisar operación del sistema, por problema al compartir archivos.
131 ERROR AL SACAR FIN DE ARCHIVO DEL PERMANENTE nomb (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por sistema operativo.
132 SE CAMBIO LA DEFINICION DEL ARCHIVO nomb, (C) NUMERO DEL RECURRENTES CAMBIADO.	<ul style="list-style-type: none"> ◆ Recompilar el programa.
133 SE CAMBIO LA DEFINICION DEL ARCHIVO nomb, EN EL R0.	<ul style="list-style-type: none"> ◆ Recompilar el programa.
134 SE CAMBIO LA DEFINICION DEL ARCHIVO nomb, (C) GRUPO DE LLAVES CON EL CAMBIO.	<ul style="list-style-type: none"> ◆ Recompilar el programa.

ERROR	Posible Causa
135 NO SE PUEDE EJECUTAR OPCION DISCO ORD-RX EN nomb, NO SE HA HECHO BUSCAR O INSERTAR DEL SUPERIOR.	<ul style="list-style-type: none"> ◆ Error de programación.
136 NO SE PUEDE EJECUTAR OPCION DISCO ORD-RX EN nomb, SE BORRO EL SUPERIOR.	<ul style="list-style-type: none"> ◆ Error de programación.
137 ERROR AL ABRIR BRXXYY DE nomb (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar directorio, área en disco, permisos, recursos del sistema. ◆ Revisar código del Sistema Operativo.
138 ERROR AL CERRAR BRXXYY DE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código del Sistema Operativo.
139 ERROR AL ESCRIBIR BRXXYY DE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Falta área en disco. ◆ Revisar código reportado por el Sistema Operativo.
140 ERROR AL LEER BRXXYY DE nomb, CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
141 ERROR AL BORRAR BRXXYY DE nomb, (C) CODIGO DEL ERROR.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
142 EL REGISTRO DE nomb LO TIENE OTRA TERMINAL, CONSULTE A SISTEMAS.	<ul style="list-style-type: none"> ◆ Revisar operación del sistema, por problema al compartir archivos, ◆ Revisar programa por 'DEAD LOCK'
143 EL ARCHIVO nomb, LO TIENE OTRA TERMINAL, CONSULTE A SISTEMAS.	<ul style="list-style-type: none"> ◆ Revisar programación del sistema, por problema al compartir archivos.

ERROR	Posible Causa
144 SE EJECUTA 'OPCION DISCO EXCLUSIVO' EN nomb, PERO YA SE USO EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Error de programación.
145 SE EJECUTA 'BUSCAR RECURRENTE ORD. EN nomb, PERO NO SE HA HECHO 'OPCION DISCO ORD-RX'.	<ul style="list-style-type: none"> ◆ Error de programación.
146 INFORMACION DE LLAVES INCORRECTAS EN nomb, (C) GRUPO DE LLAVES.	<ul style="list-style-type: none"> ◆ Revisar si hay fallas de hardware. ◆ Si se repite, reportar a su distribuidor.
147 NUMERO DE RECU-RRENTES EN nomb, OPCION M INCORRECTO. PROBLE-MAS EN EL ARCHIVO.	<ul style="list-style-type: none"> ◆ Archivo con problemas internos reconstruirlo o bajar respaldo. ◆ Revisar Operación por cancelar ELS.
148 USUARIO CANCELADO POR OPERADOR.	<ul style="list-style-type: none"> ◆ Revisar mal uso del módulo RESP.
149 ARCHIVO CERRADO POR OPERADOR.	<ul style="list-style-type: none"> ◆ Revisar mal uso del módulo RESP.
150 NO HAY AREA EN TABLA DE CANCELACION.	<ul style="list-style-type: none"> ◆ Revisar mal uso del módulo RESP.
151 ERROR AL CONVERTIR LOGICO A REAL EN PERMANENTE.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
152 ERROR AL ESCRIBIR LO PENDIENTE EN nomb.	<ul style="list-style-type: none"> ◆ Revisar código reportado por el Sistema Operativo.
153 ERROR AL EXTENDER EL ARCHIVO nomb, POSIBLEMENTE NO HAY MAS AREA EN DISCO.	<ul style="list-style-type: none"> ◆ Falta área en disco. ◆ Revisar código reportado por el Sistema Operativo.

ERROR	Posible Causa
154 EL ARCHIVO nomb NO SE CERRO EN FORMA NORMAL Y TIENE MARCAR-ABIERTO.	♦ Archivo se debe reconstruir.
155 EL ARCHIVO nomb, NO ESTA DISPONIBLE EN ESTE MOMENTO.	♦ Revisar operación del sistema.
159 ERROR EN RECA DE ARCHIVO nomb.	♦ Revisar existan archivos de DBD. ♦ Revisar área en disco. ♦ Revisar errores descritos en sección 2.9. ♦ Ejecutar RECA externo.