IBM Technical Newsletter

IBM System/34
System Support
Reference Manual

This technical newsletter describes the changes that are available to the file allocation and the compress functions of version 8, modification 0 of the System/34 System Support Program Product (Program Number 5726-SS1). These changes affect only systems with three or four disks; systems with one or two disks are not affected.

The changes to the file allocation and the compress functions are identified by the PTF log number x8888, where x is a number that indicates the modification level of the PTF. To verify that these changes have been made to your system, use the DUMP PTF procedure to print or display the PTF log for the system library (#LIBRARY).

The changes to the file allocate and compress functions affect the description of the FILE OCL statement and the descriptions of the following procedures and utility programs:

- The BLDFILE procedure

- The RESTORE procedure

- The COMPRESS procedure

- The $FBLD utility program

- The $COPY utility program

- The $PACK utility program

- The $FREE utility program

The library allocation functions of the BLDLIBR procedure and the $MAINT utility program are not affected.

Note: Please file this technical newsletter at the back of the manual. When using the FILE OCL statement and the procedures and utility programs listed above, refer to this technical newsletter for additional explanation. To remind you that this newsletter modifies the description of the FILE OCL statement, and the descriptions of the procedures and utility programs listed above, make a note of this newsletter along side their descriptions in the manual.

The following is a summary of the changes made to the file allocation and the compress functions of the SSP, and is applicable *only* if you have a system with three or four disks. If you have a system with one or two disks, the file allocation and the compress functions are not affected.

CHANGES TO FILE ALLOCATION ON A THREE- OR FOUR-DISK SYSTEM

When you create a file, you can specify a location on disk for that file. If you specify A1 or A2, the SSP allocates the file in the following way:

- If you specify A1, the SSP allocates the file at the lowest address of available storage on the first disk. If not enough space for the file is available on the first disk, the system attempts to allocate space on disks 2 through 4 (or on a three-disk system, disks 2 through 3).

- Although disks 2 through 4 are physically separate, they are treated as one logical disk: A2 is the logical name. If you specify A2, the system allocates the file at the highest address of available storage on disks 2 through 4:
  - On a four-disk system, the file is placed at the last available segment of disk 4.
  - On a three-disk system, the file is placed at the last available segment of disk 3.

  If not enough space is available on the last physical disk of logical disk A2, the SSP attempts to find space on the other disks that make up A2. If not enough space is available on logical disk A2, the SSP attempts to allocate the file on A1 (the first disk).

You can also specify a specific block number for the location of the new file. The following table gives the beginning block number locations for disks 2 through 4:
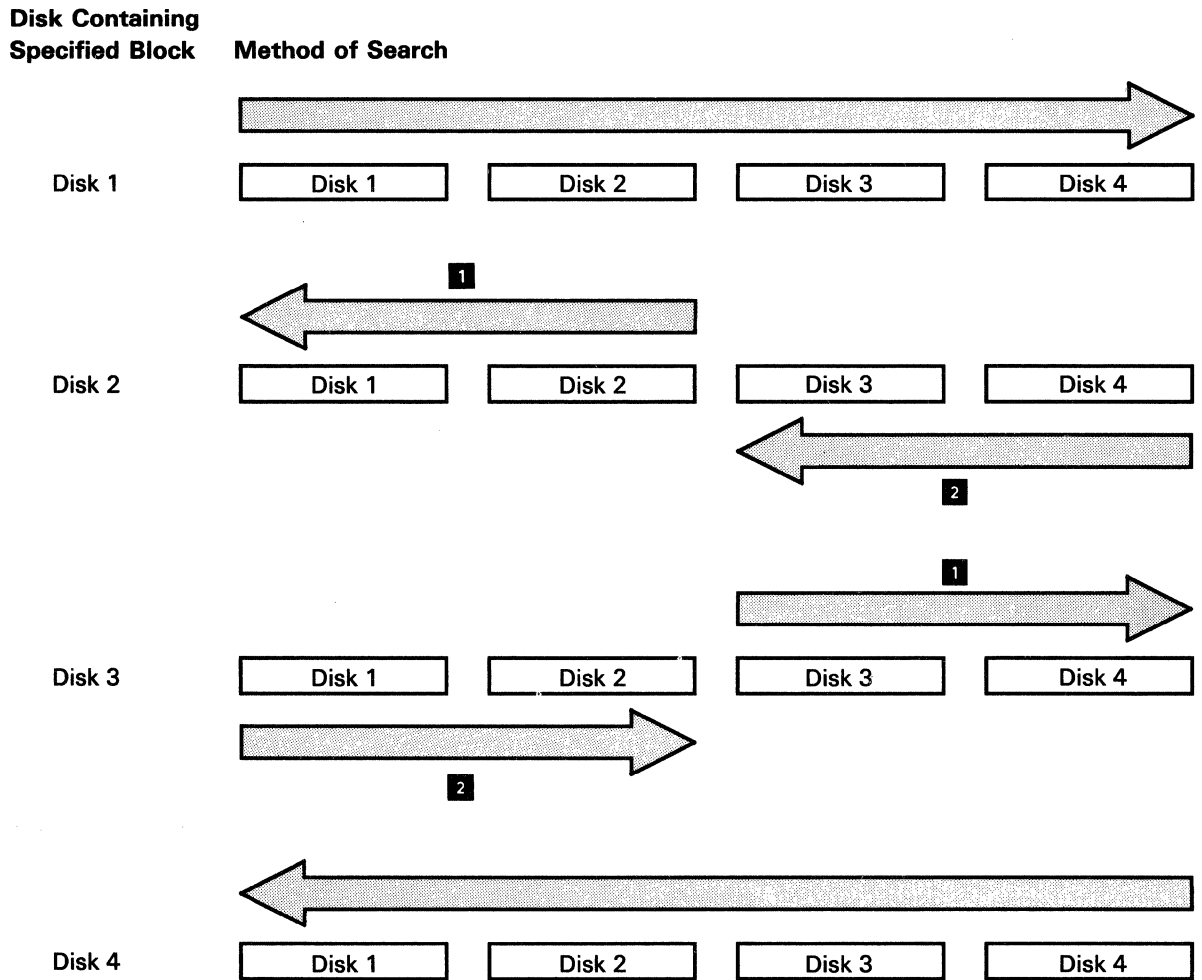
Disk	Beginning Block Number
Disk 2	25203
Disk 3	50406
Disk 4	75609

Prior to the change, if a block number was specified for the location of the new file, and if that block number location was not available, the SSP issued an error message and did not create the file.
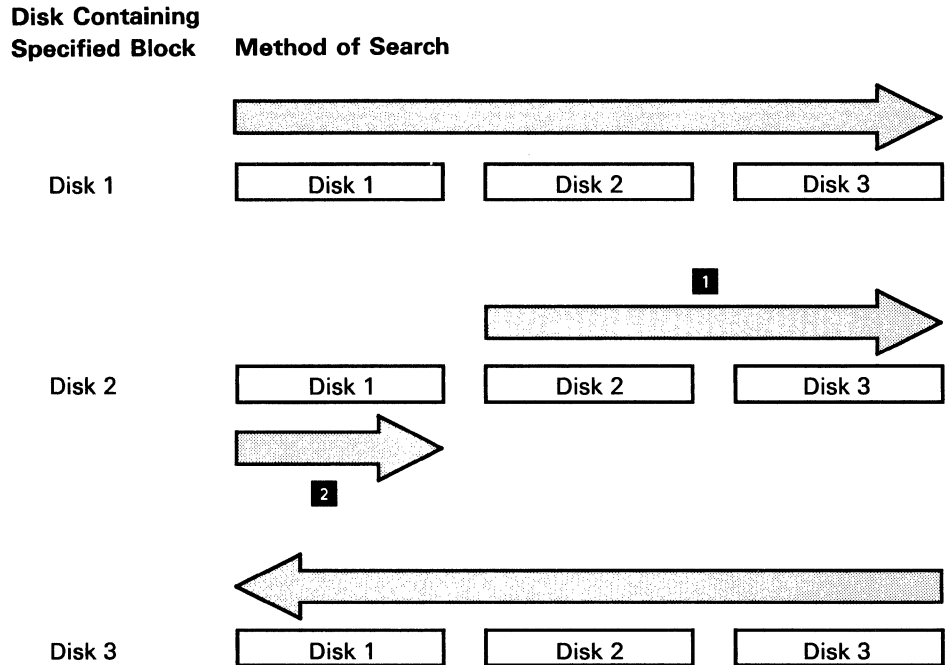
After the change, the SSP no longer issues an error message if the specified block number location is not available. Instead, the SSP first looks for available space elsewhere on the disk containing the specified block number. If space cannot be found on that disk, the SSP then looks for available space on the other disks on the system.

The way in which the SSP looks for available space depends on the number of disks on your system, and on the disk containing the specified block number. The following tables show how the SSP looks for available space depending on the disk containing the specified block number:
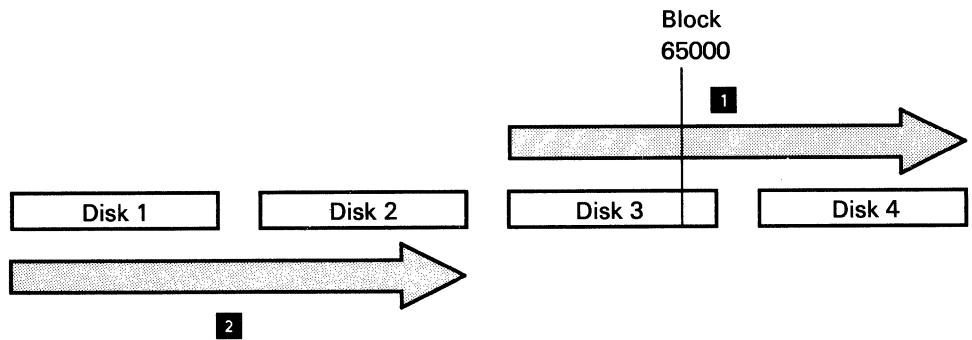
Four-Disk System

Disk Containing
Specified Block   **Method of Search**

Disk 1

| Disk 1 | Disk 2 | Disk 3 | Disk 4 |

Disk 2

| Disk 1 | Disk 2 | Disk 3 | Disk 4 |

Disk 3

| Disk 1 | Disk 2 | Disk 3 | Disk 4 |

Disk 4

| Disk 1 | Disk 2 | Disk 3 | Disk 4 |

Three-Disk System

**Disk Containing
Specified Block   Method of Search**

Disk 1

| Disk 1 | Disk 2 | Disk 3 |

Disk 2

| Disk 1 | Disk 2 | Disk 3 |

Disk 3

| Disk 1 | Disk 2 | Disk 3 |

For example, the following shows how the SSP looks for space if the file cannot be allocated at block number 65000 (disk 3 of a four-disk system):

Block
65000

| Disk 1 | Disk 2 | Disk 3 | | Disk 4 |

The following shows how the SSP looks for space if the file cannot be allocated at block number 26500 (disk 2 of a four-disk system):

Block
26500

| Disk 1 | Disk 2 | Disk 3 | Disk 4 |
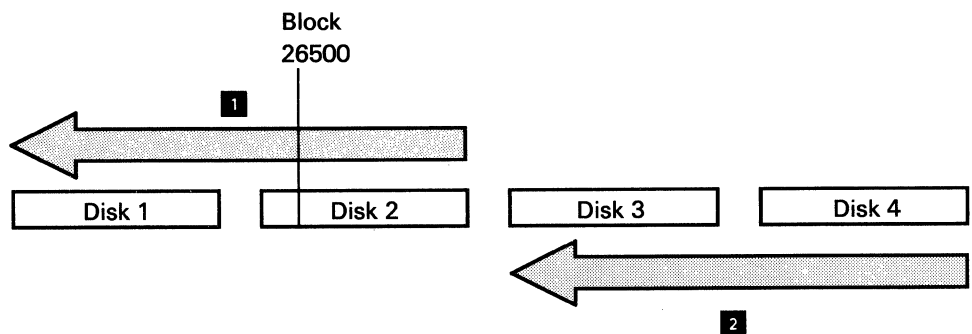
Because of this method of allocation, the file might not be placed at the block number location that you specified. Therefore, the file cannot be accessed by a reference to the location that was originally specified. You must run the CATALOG procedure to determine the actual location of the file.

If you do not specify A1, A2, or a block number location, the SSP attempts to allocate permanent (P) and temporary (T) files at the highest address of available storage on logical disk A2. If space is not available on the first disk searched, the SSP looks for space on the other disks that make up A2.

Scratch (S) and Job (J) File Allocation

Prior to the change, if a block number location or a logical disk was not specified for a new scratch (S) or job (J) file, the file was allocated at the first available space from the end of the system library (#LIBRARY) to the end of the last disk.

After the change, the search for available space starts at the beginning of disk 2 and continues to the end of the last disk. If space is not found, the search continues at the beginning of disk 1 and stops at the end of disk 1.

## CHANGES TO COMPRESS ON A THREE- OR FOUR-DISK SYSTEM

Prior to this change, the compress function caused files to be moved from one disk to another, maximizing the availability of disk space. That movement of files occurred regardless of whether the file had been placed on a specific disk. However, moving files from the disk where they were intended to reside could have slowed system performance.

With this change, the compress function of the SSP does *not* move files from the disks on which they reside. Files are, however, moved to one end of the disk on which they reside. The direction of the compress depends on the number of disks on the system:

Four-Disk System
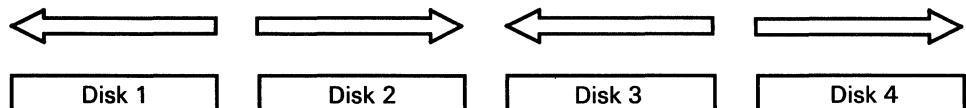
Three-Disk System
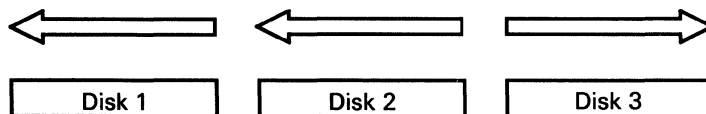
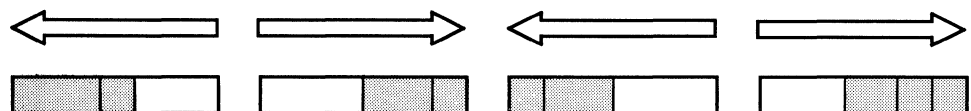For example, suppose you have a four-disk system. The shaded areas in the following figure indicate where data can be found:

Before COMPRESS is run:

If you run the COMPRESS procedure, the data on your disks will look something like this:

After COMPRESS is run:

On a four-disk system, if a file starts on disk 2 and ends on disk 3, that file is left where it is. If a file starts on disk 1 and ends on disk 2, or starts on disk 3 and ends on disk 4, the file is moved to the first of the two disks.

On a three-disk system, if a file starts on disk 1 and ends on disk 2, that file is left where it is. If a file starts on disk 2 and ends on disk 3, the file is moved to disk 2.

If you have a three- or four-disk system and you want to compress the disk in the way done prior to the change, thus maximizing contiguous free space, you must run the $FREE utility program at least twice. Use the following statements:

```
// LOAD $FREE
// RUN
// COMPRESS DISK-A1,FREE-HIGH
// END
```
First Run of $FREE

```
// LOAD $FREE
// RUN
// COMPRESS DISK-A2,FREE-LOW
// END
```
Second Run of $FREE

Because a compress operation moves files to different locations on disk, a file might not reside at the block number location that you originally specified. Therefore, the file cannot be accessed by a reference to that location. You must run the CATALOG procedure to determine the actual location of the file.