Published: 06/05/67

Identification

Use of the Multics Linker (Link_fault)
ft2ft3, setft3, f3catc
N. Adleman, D. L. Boyd

Purpose

During the development of Multics, it will frequently be desirable to have both the Multics linkage mechanism* and the pseudo-supervisor's linkage mechanism (MSPM BE.8) available at the same time.

This section describes the procedures which permit both mechanisms to be concurrently available. This coexistence of the different linkage mechanisms is made possible by using two different faults for the linkage fault: fault tag 2 from the pseudo-supervisor linker and fault tag 3 from the real (developing) Multics linker. The procedure ft2ft3 scans the linkage segments for a set of segments and changes all fault tag 2 (ft2) modifiers to fault tag 3 (ft3) modifiers. When a procedure in this set of segments attempts to use a link for the first time the resulting fault will be an ft3 rather than an ft2 and the Multics linker will be invoked by the fault tag 3 handler (procedure f3catc).

Usage

The procedure ft2ft3 is called from EPL as follows:

```
     call ft2ft3 (tab_ptr);

     declare tab_ptr (n) ptr;
```

Each element in the array of pointers, tab_ptr, points to the base of a linkage segment. These elements can be determined by calling the procedure segman in the pseudo-supervisor to retrieve a segment number for the associated text segment. For as long as the current (pseudo-) loader is used, the segment number for the linkage segment will be one more than the segment number for the text segment. Thus by adding one to the segment number for the text segment, the procedure ptr$baseptr (MSPM BY.14) can create the pointer to the base of the desired linkage segment. In EPL, an array of these pointers can be constructed as follows:

*Linker, Segment Management Module, procedure datmk_, etc.

```
declare textno fixed bin (17);

declare fb18 fixed bin (18);

declare b18 bit (18);

declare tab_ptr (n) pointer;

declare ptr$baseptr entry (bit (18)) ptr external;

declare pseudo_supervisor$segman external entry;

declare name (n) char (k);     /* name of programs in core
                                  whose linkage segments
                                  are to be changed */

iloop: do i = 1 to n;

call pseudo_supervisor$segman(name(i), textno);

     fb18 = textno + 1;        /* increment text segment
                                  number */

     b18 = fb18;               /* convert to a bit string */

     tab_ptr (i) = ptr$baseptr (b18);

end iloop;

                               /* an alternate method is by
                                  calling generate_ptr as
                                  follows: call generate_ptr
                                  (name(i)||".link",,
                                  tab_ptr(i)); */
```

Implementation

The procedure ft2ft3 extracts each pointer in the array argument and scans the corresponding linkage segment. In this scan all fault tag 2 modifiers (46 octal) are changed to fault tag 3 modifiers (47 octal). When all the pointers are extracted, and thus, all the linkage segments have been scanned, the procedure setft3 is invoked. It initializes the fault vector (assumed to be segment number 4) to transfer to the procedure f3catc on an ft3 fault.

Control returns to the caller of ft2ft3 when ft2ft3 has completed its processing.