

Published: 02/07/68,  
(Supersedes: BL.7.01; 04/05/67,  
BL.7.01; 03/19/67)

### Identification

initialization pre-link driver  
pre\_link\_1  
N. I. Morris

### Purpose

pre\_link\_1 is called by the Multics initializer from bootstrap2 (see Section BL.4.02) and from the segment loader (see Section BL.6.01). It is called after loading the first three collections from the Multics System Tape (see Section BL.1.01) to combine into the appropriate combined linkage segment the linkage just loaded and to "snap" all the links which are defined. To accomplish this, it must make calls to the lot\_maintainer in order to copy linkage sections and to set up the linkage offset table (LOT) and calls to pre\_link\_2 in order to "snap" links. (Some links may point to segments which are not yet loaded. These are ignored; they may be set on a subsequent call to pre\_link\_1).

### Combined Linkage Segments in Multics Initialization

Four combined linkage segments are provided in the Multics initialization environment: Two of these form part of the hard-core supervisor: "wired\_sup\_linkage" contains linkage sections for all wired-down hard-core supervisor procedures; "loaded\_sup\_linkage" contains linkage for the remainder of the hard-core supervisor. The other two, "wired\_init\_linkage" and "loaded\_init\_linkage", contain the wired and non-wired initialization linkage respectively.

### Usage

```
call pre_link_1 (slt_man_ptr);
```

"slt\_man\_ptr" is a pointer to <slt\_manager>/0.

pre\_link\_1 executes the following steps:

1. Pointers are obtained to the following segments by calling `slt_manager$get_seg_ptr (<slt_manager>|2)`:
  - a. `slt`
  - b. `lot`
  - c. `lot_maintainer`
  - d. `pre_link_2`
  - e. `loaded_sup_linkage`
  - f. `wired_sup_linkage`
  - g. `loaded_init_linkage`
  - h. `wired_init_linkage`
2. All SLT entries are scanned. For each text segment with linkage provided, the following is done:
  - a. If the "pre\_linked\_sw" is on, this entry is skipped.
  - b. A pointer to the segment's linkage segment is obtained from "text\_link\_segno".
  - c. If the "combine\_sw" is on, `pre_link_1` determines in which of the four combined linkage segments the segment's linkage section belongs. (This is determined by whether the segment is a supervisor or an initializer segment and whether it is wired-down or not). `lot_maintainer$copy_linkage (<lot_maintainer>|2)` is called to copy the linkage section into the appropriate combined linkage segment.
  - d. `lot_maintainer$set_lp (<lot_maintainer>|1)` is called to make an entry in the LOT or the segment and to set the linkage section's definitions pointer.
  - e. The "pre\_linked\_sw" in the SLT entry is turned on.
3. All SLT entries are scanned. For each text segment with linkage provided, the following is done:
  - a. A pointer to the segment's linkage section is obtained by making a call to `lot_maintainer$get_lp (<lot_maintainer>|0)`.
  - b. `pre_link_2$scan_linkage (<pre_link_2>|0)` is called to pre-link the segment.
4. Return.