Multics Task Report

MTR-087

To: Distribution

From: Steve Webber

Date: 05/01/75

Subject: Prelinking Project

This MTB gives tasks which are all related in some way to prelinking but very few of which are required for prelinking or require prelinking. Prelinking, as currently envisioned, could be implemented in today's Multics. However, many of the advantages to prelinking can not be fully achieved until other changes are made. These other changes are 1) more shared data, 2) less static storage, 3) copy-on-write, etc. Once the ability to optimize in these areas is at hand, prelinking becomes more attractive. The point being made is that, although prelinking is fairly independent of the other changes, the other changes benefit prelinking considerably. For this reason, the actual prelinking implementation will await some of these other changes and, hence, they are included in this MTR.

## Task Definition

There are three basic changes being proposed. these are:

1. Separate Static Storage,

2. Copy on write, and

3. Separation of name space and address space management.

(None of these changes is needed for prelinking and none require prelinking.)

The issue of separate static probably causes the most widespread changes in the system. In particular, the following are affected:

1. The linker must be changed to handle the ISOT correctly,

2. The standard object segment must change to accomodate the new section. This includes changes to the following:

   a. the binder,

   b. PL/I,

   c. ALM,

   d. object_info_ and immediate friends,

   e. bound_debug_util_, and

   f. other callers of object_info_.

The copy-on-write mechanism is isolated in a few programs and affects very few system and user programs. It does, however, probably require a more optimal or at least special cased signalling mechanism.

The name space manager changes underway are also not needed for prelinking. However, prelinking does want to separate reference name management from address space management so the prelinking tasks will await these changes which are incorporated as part of the redo of the address space manager.

A completely new makeknown mechanism is now working which allows use of directory segment numbers rather than pathnames. Not only is this more efficient, it also simplifies ring 0 by isolating all pathname management functions in a single procedure (which can eventually be removed from ring 0). So we see the prelinkiner is awaiting yet another task which is not strictly required.

The new address space manager changes introduce a whole new set of tasks related to specifying and implementing new hardcore interfaces which accept directory segment numbers rather than pathnames. These new interfaces are clearly independent of prelinking and the work on them will go on in parallel to the prelinking tasks at best.

The benefits to be gained from these new interfaces are hard to measure. Clearly they will provide a means of getting the same amount of work done for less expense, but the degree of improvement is unknown. For this reason, the interface development tasks will receive lower priority than the other tasks in the MTR. In fact, we may choose not to attack this problem in earnest until the new storage system is operational. (The reason we would do the primitives at all at this early stage is that in order to take advantage of the other changes and prelinking itself, we must recompile and/or reprogram a large

part of the system. Such a large scale reprogramming task might want to include use of the more efficient interfaces in order to avoid another reprogramming task later on.)

Detailed Status

As mentioned above, the address and name space changes are well along in implementation. Design discussions for certain aspects will be held in the near future.

The KST access mechanism is similarly nearly complete. The initiated mode concept is awaiting decisions on the copy-on-write strategy as these will both be incorporated in the same upcoming version of the address space manager.

The compiler changes have also begun. The "options (constant)" construct is working and the other tasks will begin soon or have already begun.

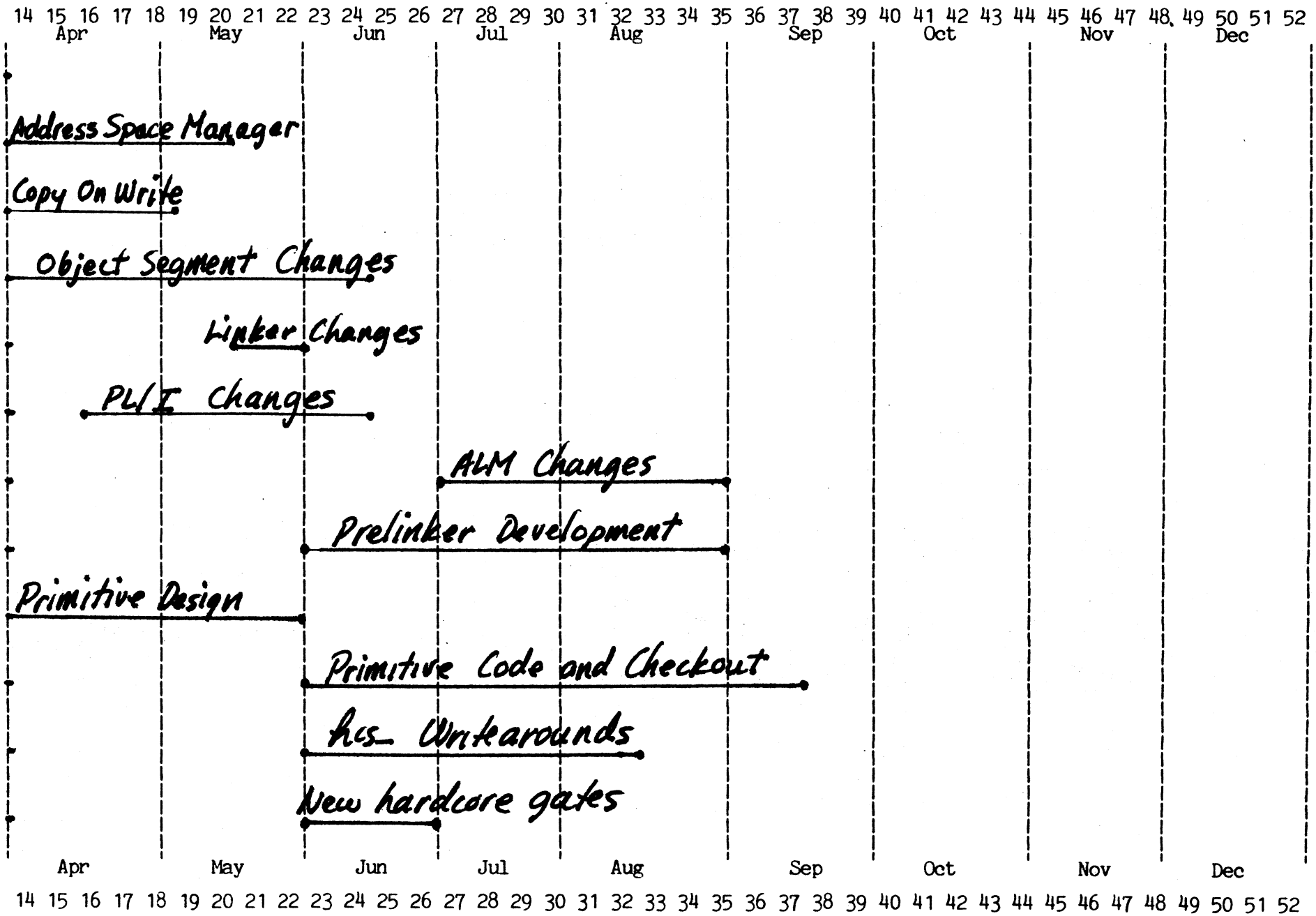
The ALM tasks will not be started until this summer due to personnel scheduling. These tasks, like many others, can be postponed as there is no strict requirement for them.

The object segment tasks are probably on the most critical path as many other tasks await the new object_info_. An MTB has been written describing the proposed changes and design discussions will be scheduled as soon as possible.

The copy-on-write issue is also awaiting a design discussion. The implementation here is simple except for the possibility of adding a feature (not publicized as a user interface, but wholly in the user ring) for handling system conditions with "static handlers".

The linker changes are scheduled to begin as soon as a working version of object_info_ is available. Groundwork for the linker changes were laid down in system 25-3. Once object_info_ and the linker are available, it will be possible to checkout and install other programs that create and understand new object segments.

The implementation of the prelinker itself will await installation of the other new features mentioned above. There are also several design issues to be clarified and there is much documentation to be prepared.


14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Address Space Manager
Copy On Write
Object Segment Changes
Linker Changes
PL/I Changes
ALM Changes
Prelinker Development
Primitive Design
Primitive Code and Checkout
Res. Workarounds
New hardcore gates
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

GROUP Multics Development Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA Name Space Manager

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Design new Reference Name Manager | Bratt<br>Webber | 03/01/75 | 03/15/75 | | done |
| Code and debug new Reference Name Manager | Bratt | 03/15/75 | 03/31/75 | | done |

GROUP Multics Development Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA Address Space Manager

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Design new KST format | Bratt<br>Stone<br>Webber | 03/01/75 | 03/31/75 | | done |
| Implement new makeknown interface to new KST | Bratt | 03/15/75 | 04/15/75 | | done |
| Implement new KST directory management | Bratt | 03/15/75 | 04/30/75 | | done |
| Add access information to the KST | Stone | 04/01/75 | 04/30/75 | | On schedule |
| Add initiated mode handling to the KST | Stone | 04/30/75 | 05/14/75 | | On schedule |
| Write writearounds to new ASM Kernel | Bratt<br>Stone | 04/01/75 | 04/31/75 | | done |

GROUP Multics Development Group DATE PAGE 1 / 1

PROJECT Prelinking Project AREA Miscellaneous Tasks

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Write MTB on copy-on-write mechanism | Webber | 04/01/75 | 04/14/75 | | done |
| Write MTB on limited static handler mechanism | Webber | 04/07/75 | 04/14/75 | | done |
| Implement secure restart mechanism | Webber | 04/01/75 | 04/21/75 | | done |
| Implement copy-on-write program for user ring | Webber | 04/15/75 | 04/30/75 | | On schedule |

GROUP Multics Development Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA New Primitives and Subroutines

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Write MTB proposing new primitives and changes to current primitives | Webber | 03/14/75 | 05/01/75 | | Draft in circulation for initial review |
| Coordinate design discussions on new primitives | Stone Webber | 04/15/75 | 05/15/75 | | |
| Design and Implement new storage system primitives | Many | 05/31/75 | 08/15/75 | 20 | Depends on design reviews and subsequent changes |
| Write new outer-level hcs_ interfaces | Many | 05/31/75 | 08/01/75 | | Initial version remains in ring 0 |
| Provide new hardcore programs to replace hcs_ interfaces (excluding directory control) | Stone Webber | 04/30/75 | 05/31/75 | | Requires added names on hcs_ or a new gate |
| Design and Implement new user-ring subroutines for new commands | Stone Webber | 05/31/75 | 06/15/75 | | Includes command_util_, expand_arg_, and get temp segs |
| Generate new MPM and SWG documentation for new command utilities | | | | | |

GROUP Multics Development Group DATE 04/29/75 PAGE 1 / 2

PROJECT Prelinking Project AREA Object Segment Changes

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Document proposed changes to the standard object segment definition | Weaver | 03/15/75 | 04/21/75 | | Circulated for internal review (twice) |
| Generate info segs describing object segment changes to users | Weaver | 04/22/75 | 04/29/75 | | |
| Code and checkout object_info_ and immediate friends | Weaver | 04/29/75 | 05/07/75 | | |
| Generate updated MPM documentation for object segment changes | Weaver | 04/15/75 | 05/15/75 | | |
| Convert callers of object_info_ in the system | Weaver<br>Barnes | 05/04/75 | 05/31/75 | | |
| Convert programs of bound_debug_util_ to use new object info | Weaver | 05/20/75 | 06/03/75 | | |
| Convert dump_ls and print_linkage_usage to use new object info | Weaver | 06/04/75 | 06/18/75 | | |
| Convert nonsystem callers of object_info_ (lisp, etc.) | | | | | needs coordination only |
| Design changes to binder for new object segment format | Weaver | 05/01/75 | 05/15/75 | | |
| Code and checkout changes to the binder for new object segment format | Weaver | 05/01/75 | 05/15/75 | | |
| Code and checkout binder changes for text-embedded-links | Weaver | | | | Low priority |
| Change linker to accept new style object segments (use of ISOT) | (Bratt?) | 05/15/75 | 05/29/75 | | |
| Document user-visible changes to the binder (MPM and info segs) | Weaver | 05/15/75 | 05/31/75 | | |

GROUP Multics Development Group DATE 04/29/75 PAGE 2 / 2

PROJECT Prelinking Project AREA Object Segment Changes

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Change generate_mst and system initialization to use new object segment format | | | | | Low priority. No new object in ring 0 until done |

GROUP Multics Development/Language Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA PL/I Changes

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Change the compiler to generate new style object segments | Wiatrowski | 04/22/75 | 05/22/75 | | |
| Change pl1 command to accept separate static option | Wiatrowski | 04/22/75 | 05/22/75 | | |
| Change compiler to operate correctly when separate static specified (addressing code) | Barnes | 05/22/75 | 06/12/75 | | |
| Change compiler to generate new entry structures | Wiatrowski | 05/14/75 | 05/28/75 | | |
| Change the trace command so that it does not use static storage of each program | Barnes | 04/29/75 | 05/14/75 | | Requires change to compiler not to allocate the storage |
| Code and checkout change to pl1_operators_ for new push operator | Barnes | 05/22/75 | 05/29/75 | | Makes use of ISOT |
| Change compiler to accept options (constant) | Schoeman | 04/13/75 | 04/18/75 | | done |
| Change stu_ for separate static | Barnes | 05/22/75 | 05/29/75 | | |

GROUP Multics Development/Language Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA ALM CHanges

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Add entry bound pseudo-op to ALM | Wiatrowski | 07/01/75 | 07/15/75 | | |
| Change ALM to generate new object segments | Wiatrowski | 07/15/75 | 07/29/75 | | |
| Change ALM to recognize *static | Wiatrowski | 07/29/75 | 08/13/75 | | |
| Change ALM to recognize join /static/ | Wiatrowski | 08/13/75 | 08/27/75 | | |
| Change pl1_operators_ to use new push macro for ALM programs | Barnes | 05/22/75 | 05/29/75 | | |
| Add text-embedded-links to ALM | Wiatrowski | | | | Low priority |

GROUP Multics Development Group DATE 04/29/75 PAGE 1 / 1

PROJECT Prelinking Project AREA Prelinker Tasks

| TASK DESCRIPTION | PERSONNEL | START | FINISH | M-W | CHANGES-STATUS |
|---|---|---|---|---|---|
| Design and Implement PLDT tools. | | | | | |
| Design new RNT structure for prelinking search rules | | | | | |
| Implement change to RNT manager for prelinking search rules | | | | | Includes primitives for setting and getting search rules |
| Design prelinker driver | | | | | |
| Implement prelinker driver | | | | | |
| Implement KST template building primitives. | | | | | |