To: Distribution
From: Betsy
Date: May 10, 1976
Subject: Approved MCR's from April 16-30, 1976

Attached are the MCR's that were approved from April 16-30, 1976. Please note that MCR's 1297, 1298 and 1327 are also included because they were not published upon their approval.
TITLE: Referencing Directory in iox-attach

AUTHOR: M. D. MacLaren

- Coded in [ ] PL/I [ ] ALM [ ] other -
  explain in DETAILED PROPOSAL
- Planned for System MR
- Fixes Bug Number(s):
- Documented in MTR
- User/Operations-visible Interface change? [x] yes [ ] no
- Incompatible change? [x] yes [ ] no
- Performance: [ ] Better [x] Same [ ] Worse
- Replaces MCR 1186

Objections/Comments:

Old interface will be retained indefinitely.

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

SUMMARY: Replace iox_$attach_iocb and iox_$attach_ioname by iox_$attach_ptr and iox_$attach_name that have a ref_ptr argument specifying the referencing procedure (if any) for the search rules. Change pl1io to pass a pointer to the user's procedure as ref_ptr when the user gives a title. Change ios_$attach to pass a pointer to the user's procedure. All other system callers of iox_ pass a null pointer.

REASONS: The name of an I/O module may be given at command level or it may be built into a program, and the search rules should be adjusted accordingly. The proposal allows explicit control in the general case, puts iox_$attach back the way it was before iox_, and treats the module name in the PL/I title option like a subroutine name.

IMPLICATIONS: There is a very-very small chance that the changes to iox and pl1io will affect some user.
**Note:** A complete description of the interpretation of the attach description has been added here. This duplicates a section in the Reference Guide.

**Entry:** `iox_$attach_ptr`

This entry point attaches an I/O switch in accordance with a specified attach description. The form of an attach description is given in "Multics Input/Output System" in Section IV of the MPM Reference Guide. If the switch is not in the detached state, its state is not changed, and the code `error_table$not_detached` is returned.

**Usage**

```fortran
declare iox_$attach_ptr entry (ptr, char(4), fixed bin(35));
call iox_$attach_ptr (iocb_ptr, atd, code);
```

where:
1. `iocb_ptr` points to the switch's control block. (Input)
2. `atd` is the attach description. (Input)
3. `code` is an I/O system status code. (Output)
4. `ref_ptr` is a pointer to the segment that is considered the referencing procedure. See "Notes" below. (Input)

**Notes:** The attach description has the following form:

```
module_name -option_1- ... -option_n-
```

The substrings `module_name`, `option_1`, ..., `option_n` must not contain blanks and must be separated by one or more blanks. The whole attach description may contain trailing blanks but not leading blanks.

The substring `module_name` determines the I/O module for the attachment as follows: If it does not contain any instances of greater-than or less-than characters (> or <), it is interpreted as a reference name, and the I/O module is found by the search rules. If `module_name` contains any greater-than or less-than characters, it is interpreted as the pathname (absolute or relative) of the I/O module.

The directory in which the segment pointed to by `ref_ptr` is located is used as the referencing directory for the standard search rules. If `ref_ptr` is null, then the standard search rule specifying the referencing directory is skipped. See "System Libraries and Search Rules" in Section III of the MPM Reference Guide.

The substrings `option_1`, ..., `option_n` must conform to the requirements of the particular I/O module. The I/O modules are described later in this document.
When the attachment is made, if the I/O module is not already initiated by
the specified reference name, it is so initiated. When module_name is given as
a pathname, the reference name is the final entryname in the pathname.

Entry: iox$_attach$name

This entry point is the same as the iox$_attach$ptr entry point except
that the I/O switch is designated by name and a pointer to its control block is
returned. The control block is created if it does not already exist.

Usage

```
declare iox$_attach$name entry (char(*), ptr, char(*), fixed bin(35));
call iox$_attach$name (switchname, iocb_ptr, atd, code);
```

where:

1. switchname is the name of the I/O switch. (Input)
2. iocb_ptr points to the switch's control block. (Output)
3. atd is the attach description. (Input)
4. ref_ptr is null or a pointer to the segment that
   is considered the referencing procedure.
   See "Notes" under iox$_attach$ptr. (Input)
5. code is an I/O system status code. (Output)
### Multics Change Request

**TITLE:** Delete the error_table_compiler from the system

**AUTHOR:** S. Webber

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>-Coded in [ ] PL/I [X] ALM [ ] other</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Planned for System MR</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Fixes Bug Number(s)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Documented in MTB</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-User/Operations-visible</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Interface change? [X] yes [ ] no</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Incompatible change? [X] yes [ ] no</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Performance: [ ] Better [X] Same [ ] Worse</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>-Replaces MCR</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STATUS</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Written</td>
<td>07.15.75</td>
</tr>
<tr>
<td>Status</td>
<td>Revised</td>
</tr>
<tr>
<td>Expires</td>
<td>02.19.76</td>
</tr>
</tbody>
</table>

**DOCUMENTATION CHANGES**

- **Document**
  - Specify One or More
    - MPM (Vol, Sect.) SWG (AK92)
    - PLMS (AN #)
    - MOSN (Sect.)
    - MPAM (Sect.)
    - MSAM (Sect.)

<table>
<thead>
<tr>
<th>Objections/Comments:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info Segs</td>
</tr>
<tr>
<td>Other (Name)</td>
</tr>
<tr>
<td>None (Reason)</td>
</tr>
</tbody>
</table>

**Use these headings:** Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

**SUMMARY:** Delete the program error_table_compiler from the system. Generate error tables using mexp.

**REASONS:** Fewer programs to maintain. Less confusion in that error table source is not a separate language of its own but rather (a slightly extended) ALM.

**DETAILED PROPOSAL:** Provide 4 macros in et_macros.incl.mexp to perform the equivalent functions. These macros would be:

- `et_head` header/setup
- `ec` define an error code
- `system_et` to close out the system error_table_
- `user_et` to close out a user error_table_

**IMPLICATIONS:** SWG documentation must be upgraded. In particular mexp must be elevated to SWG support or possibly have the name error_table_compiler added to it (users needn't really know what mexp is).

An exec_com should be provided to convert old style error_table_source to new style.
Multics Change Request

TITLE: Add prelinking to the system

AUTHOR: S. Webber

Category (Check One) 
Lib. Maint. Tools
Sys. Anal. Tools
Sys. Prog. Tools

Document Specify One or More

PL/I
ALM
Other

DETAILED PROPOSAL
explain
PL/I
ALM
Other

Fixes Bug Number(s) 355

User/Operations-visible Interface change? X yes □ no
Incompatible change? □ yes X no
Performance: X Better □ Same

Replaces MCR

Objections/Comments:

Use these headings:
Summary of Proposal, Reasons for Proposal, Implications,
Detailed Proposal.

SUMMARY:

Install a simplified prelinking program to be run during initialization to create prelinked template processes which are fully specified at bootload time.

REASON:

This prelinker will permit considerable performance gains for the initial release of the BASIC and FORTRAN Command and Editing subsystem.

IMPLICATIONS:

Since this version will not solve the general problems of prelinking such as variable search rules, online installations, new primitives, etc.,

there may be (internal) interface changes later on to facilitate introduction of these new features.
DETAILED PROPOSAL:

This version of the prelinker will generate prelinked template processes from PLDT's which are found in >system_library_1, hence, which came in on the bootload tape. (The directory in which the templates are placed will be specified in the PLDT's).

No problems arise due to changes in KST or RNT format since all templates are created anew each bootload.

There will be no capability for changing the prelinked subsystem until the next bootload, i.e., online changes for programs within the subsystem will have no effect until the next bootload.

There will be no means for changing the search rules dynamically within a prelinked process.

Other features described in MTB-169 will be implemented and/or utilized including:

1. copy-on-write
2. the ISOT
3. separate static
4. static handlers
3. control_args

  -brief, -bf
  -home_dir path, -hd path
  -process_overseer path, -po path
  -no_print_off, -npf
  -print_off, -pf
  -no_preempt, -np
  -no_start_up, -ns
  -account id, -ac id
  -force
  -change_password, -cpw

  can be selected from the following:

  suppress messages associated with a successful login. If the standard process overseer is being used, then the message of the day is not printed.

  set the user's home directory to the path specified, if the user's project administrator allows him to specify his home directory.

  set the user's process overseer to the procedure given by the path specified, if the user's project administrator allows him to specify his process overseer.

  cause the system to overtype a string of characters to provide a black area for the user to type his password.

  suppress overtyping for the password, since the user's terminal responds to the printer off control sequence.

  refuse to log the user in if he can only be logged in by preempting some other user in his load control group.

  instruct the standard process overseer not to execute the user's start_up.ec segment, if he has one, and if the project administrator allows him to avoid it.

  set the normal account identifier for the user to the one specified. (This control argument currently has no effect.)

  log the user in if at all possible if the user has the guaranteed login attribute. Only system users who perform emergency repair functions have the necessary attribute.

  change the user's password to a newly given password. The login command requests the old password before it requests the new password. If the old password is correct, the new password replaces the old for subsequent logins, and the message "password changed" is printed at the user's terminal. The user should not type the new password as part of the control argument.
If path ends in the characters ",direct", the specified procedure will be called directly during process initialization rather than by the "init_admin" procedure provided by the system. This means that the program specified by path must perform the tasks which would have been performed by the system's "init_admin" procedure.

If path ends in ",subsystem" the process created will be initialized by using template data bases existing in the directory specified by path. Use of this suffix causes the effect of having specified ",direct" as well.
TITLE: Install carry_total active function

AUTHOR: Steve Herbst

SUMMARY: Install a new active function, carry_total, that returns the number of requests in a given carry queue, for example:

\[ \text{carry_total } \text{-admin } \text{-ds MIT} \]

REASON: Useful in administrative exec_com's.
Name: carry_total

This active function returns the total number of requests in a given carry queue.

Usage:

```
[ carry_total -control_args- ]
```

where control args can be:

- **-admin**
  All requests, not just those belonging to the user. Status permission is required on the queue in order to use this control argument.

- **-ds XXX**
  Use the queue for destination XXX. The string XXX can be MIT, PCO, DEV or CISL. If this control argument is not specified, the default queue carry.ms is used.

- **-queue_path PN, -qp PN**
  Look for carry queues in the directory PN rather than the system default carry directory /daemon_dir/carry_dir.
TITLE: Improve the carry facility

AUTHOR: S. Herbst

SUMMARY:
Install the programs comprising the carry facility as tools. Currently, they are not officially installed.

Update the carry facility to do the following:

1. Send users mail when their requests are loaded.
2. Send users mail when a request cannot be dumped for whatever reason.
3. Be more crash-proof.
4. Change the access rules for an entry to be carried:

   r Carry.Multics.*
   r for the requestor

   on a segment or on all segments in a subtree, and:

   sma Carry.Multics.*
   sma for the requestor

   on the parent directory of the entry to be carried.
MCR 1691
Pg. 2 of 2

REASONS:

1. In case a crash occurs during loading, a segment can be loaded a second time. In the future, users will be warned not to update the contents of a carried segment until carry has mailed him a notification that it has been loaded.

2. Notification of errors while dumping currently has to be done by the person who looks at carry_dump error files. In the future, the owner of an entry that cannot be dumped finds out the same day.

3. Requests are often lost because a crash occurs during the absentee job that dumps carry requests. Only those requests that are on the tape at the time of the crash are carried. In the future, the carry_dump absentee.job will resubmit itself in case of a crash and cancel the resubmission when the tape is complete.

4. The above rules define minimum access requirements for an entry to be carried. w access is not required, because it can be forced given sma. There is no way to check for sma on the target parent, so we settle for sma on the parent here as an approximation.

IMPLICATIONS:

The last change affects the carry exec's, which are not installed and vary from system to system. The others affect the programs carry, carry_dump, carry_load and carry_dump_dp, which should be standard tools.
Note on MCR 1691

Documentation follows for the proposed new carry commands. Access requirements for entries to be carried have changed. Description of these new access requirements make use of the phrase "carry administrator". The old carry facility assumed the carry administrator, the user who dumps and loads carry tapes, to be Carry.Multics.*. The new facility regards anyone having r and d access to a carry queue to be a carry administrator for that queue.

Any user who has r and d access to a carry queue is able to make carry tapes from that queue. Therefore, any user having r and d access to a carry queue should also have appropriate access to entries named by requests in that queue. In most cases, only one user is allowed to run carries and only that user should be on the queue's ACL with r and d extended access. Access to >ddd>carry_dir>carry.ms at MIT, for example, should be changed to deny d access to *.SysDaemon.* and *.SysMaint.*. The access on a new carry queue is always adros to the creator, who is usually the carry administrator, and ao to everybody else.
Name: carry

This command requests that one or more segments or directory
subtrees be moved from one Multics site to another. It can also
be used to list or cancel pending carry requests. Requests are
stored in a message segment queue and are carried daily.

Usage:

    carry paths -control_args-

where:

1. paths are pathnames of segments and directories. For a
directory, the entire subtree is carried.

2. control_args can be:

   -destination DS,
   -ds DS carry to, or list or cancel requests for, the
destination DS. Queues are named DS.carry.ms. The
valid destinations depend on the particular Multics
site. Examples of destinations are:

   PCO  System M in Phoenix (default at MIT)
   MIT  MIT Service System (default at PCO)
   CISL  a synonym for MIT
   DEV  CISL Development Machine (from MIT only)

   The default queue carry.ms is used when this argument
is not supplied.

   -queue_path dirpath,
   -qp dirpath specifies the pathname of a directory in which carry
queues reside. This control argument allows the user
to use a carry facility other than the system default
carry facility. Queues are named DS.carry.ms in this
directory, where DS is specified by the -destination
control argument. If the -queue_path control
argument is not specified, queues reside in the
directory >daemon_dir_dir>carry_dir.

   -list, -ls list the user's pending carry requests for the
specified destination. If the -admin control
argument is also given, list all requests in the
queue.

   -cancel remove the carry requests named by paths from the
queue for the specified destination. Each pathname
must exactly match the pathname in the request, as
listed with the -list control argument. A message is
typed for each request cancelled.

   -admin allows the user to list all requests or cancel any
request in the queue, assuming he has proper access
(r and d extended access to the queue).

Notes:

In order to carry, the user must be registered by the same name at both sites. If any directories in the pathname of the entry to be carried do not exist at the target site, they are created.

Carries are performed by the carry administrator, who has r and d extended access to the given carry queues and requires appropriate access to the entries to be carried. In most cases, there is only one carry administrator, for example Carry.Project.*. The carry command checks that the following access requirements are met:

1. sma to the requestor and to all carry administrators on the parent directory of the entry being carried.

2. sma to the requestor and to all carry administrators on all directories in a subtree.

3. r to the requestor and to all carry administrators on a segment being carried or on all segments in a subtree.
Name: carry_dump

This command dumps carry requests from a given queue on a given tape.

Usage:

carry_dump path tape_number

where:

path is the pathname of a carry queue.
tape_number is the number of the desired tape.

Notes:

The user must have r and d extended access to the queue.

The following output segments are produced in the same directory as the carry queue:

A dump map named carry_dump.<date>.<time>.map.
An error file named carry_dump.<date>.<time>.ef.
A directory named carry_dump.<date>.<time>.map_dir, containing individual dump maps for the requestors. These are printed by the command carry_dump_dp.
A tape log named <tape_number>.tape_log listing the contents of the tape dumped.
An exec_com segment named <destination>.send_mail.ec which is dumped on the tape and used by carry_load at the target site to send mail to users when their requests have been loaded.
(Administrative document)

Name: remake_carry_tape

This command uses the segment <tape_number>.tape_log to remake a carry tape by a specified number.

Usage:

remake_carry_tape old_number new_number

where:

old_number is the number of the tape to be remade. There must be a segment <tape_number>.tape_log in the user's working directory.

new_number is the desired number of the remade tape.
(Administrative document)

Name: carry_load

This command loads a carry tape at its target site.

Usage:

carry_load control_path --debug-

where:

control_path is the pathname of a backup_load control file. The contents of this control file can be just ">**" to reload everything on the tape.

--debug specifies that the load is to be performed in "debug" mode. See backup_load for a description of this mode.

Notes:

The following output segments are produced in the directory containing the control file:

A load map named carry_load.<date>.<time>.map.
An error file named carry_load.<date>.<time>.ef.
An exec_com segment named <destination>.send_mail.ec which, after being updated with a list of errors from the error file, is used to send mail to users when their requests have been loaded.
Multics Change Request

TITLE: Fix quota bug in pool_manager

AUTHOR: J. C. Whitmore

Objections/Comments:

One or More

Objections/Comments:

- Codes in: XFL/I
- Planned for System MR 4.0
- Documented in MTB
- User/Operations-visible

Interface change? No

- Performance: Same

- Replaces MCR

Objections/Comments: None (Reason) No Interface Change

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

Reasons: The card input daemon gets a record quota overflow error code under certain conditions when trying to move quota back to a user directory in the card pool. This occurs because the pool_manager_close_user_pool entry fails to check if the quota move would make the access class directory quota non-terminal.

Summary: Modify the close_user_pool entry of pool_manager to check for a code of error_table_rqover or error_table_invalid_move_quota from hcs_quota_move. When this occurs, the pool_manager will move an extra two pages of quota from the pool root to the access class directory and attempt the move one more time.
### TITLE: Card Input Daemon bug fixes

**AUTHOR:** J. C. Whitmore

<table>
<thead>
<tr>
<th>-Coded in:</th>
<th>Category (Check One)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XPL/I</td>
<td>Lib. Maint. Tools</td>
</tr>
<tr>
<td>ALM</td>
<td>Sys. Anal. Tools</td>
</tr>
<tr>
<td>other</td>
<td>Sys. Prog. Tools</td>
</tr>
</tbody>
</table>

- Planned for System MR
- Fixes Bug Number(s): 355
- Documented in MTB

<table>
<thead>
<tr>
<th>User/Operations-visible Interface change?</th>
<th>Document Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>yes</td>
<td>Salvager</td>
</tr>
<tr>
<td>no</td>
<td>MPM (Vol, Sect.)</td>
</tr>
</tbody>
</table>

- Incompatible change? yes
- Performance: Better
- Replaces MCR

<table>
<thead>
<tr>
<th>Objections/Comments:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info Segs</td>
</tr>
<tr>
<td>Other (Name):</td>
</tr>
<tr>
<td>None (Reason):</td>
</tr>
</tbody>
</table>

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

**Summary:**

1) change control card parsing to allow a semicolon to terminate the deck id card (it will be optional).

2) change the quit handler to reset read the input command stream.

**Reasons:**

1) Currently a semicolon is required on the access id card and forbidden on the deck id card. This change will make the card formats consistent. The semicolon will be optional on the deck id card for upward compatibility.

2) The message coordinator can queue up several card daemon commands before they are read. These should be reset when a quit is sent to the Card Input Daemon.
The deck_id card has the following format:

DECK_NAME PUNCH_FORMAT

where:

1. **DECK_NAME** is the name used to identify the card image segment in System Pool Storage. It should be unique among the user's decks recently submitted. In the event of name duplications, the system card reading process appends a numeric component to the end of the supplied name and creates a duplicate card image segment for DECK_NAME.

2. **PUNCH_FORMAT** is the punch code conversion to use in reading the card deck. It must be MCC, VII PUNCH, or RAW. If this field is omitted, MCC format is assumed.

If name duplications are encountered then there may be more than one deck in System Pool Storage whose first component is DECK_NAME. The copy_cards command retrieves all of these copies when invoked.

The control cards should be produced on a standard IBM 029 key punch. The fields on the control cards are free format with spaces separating the fields. The only restrictions are:

1. the first access_id card must contain the PERSON_ID, plus the PROJECT_ID and first component of the ACCESS_CLASS if the PROJECT_ID and ACCESS_CLASS are specified.

2. the deck_id card must contain the DECK_NAME, plus the PUNCH_FORMAT if the PUNCH_FORMAT is specified.

Note: Use of a terminating semicolon is optional.

All characters on the control cards are mapped to lower case except those immediately following an escape character (backslash or cent sign). For example \MY\DECK.PL1 is mapped to My_Dock.pl1.

**Example**

Suppose user Doe working on project Proj, wishes to read a FORTRAN source deck into a segment called alpha.fortran, with an access class of "proprietary, my_company". The access_id card is:

\DOE.\PROJ  PROPRIETARY, MY_COMPANY;

and the deck_id card is:

ALPHA.FORTAN  MCC;

where MCC is the format of the data cards. The control cards followed by the data cards are submitted to operations personnel for reading. When the cards
Title: Provide separate definitions during Multics initialization

Author: Noel I. Morris

Objections/Comments:

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

Summary:
Modify the Multics System Tape contents to contain separate definitions. Modify Multics system initialization to load the definitions separately.

Reasons:
Currently, the definitions are written onto a Multics System Tape as part of the text. In the case of wired-down segments with many external references, one or more pages of wired-down memory can be wasted by needlessly wiring the definitions. Separating the definitions from the text will enable them to be loaded separately. It is hoped that up to 10 more pages of main memory can be made available with this scheme. In addition, the initialization "high water mark" will be lowered.

Detailed Proposal:
Modify the generate_mst command to write definitions separately on to the Multics system tape. Add a bit to the SLT entry to identify definitions segments.

Provide a fabricated segment to contain all loaded definitions. Modify bootstrap1 and segment_loader to load linkage segments directly into the appropriate combined linkage segment and to set the lot entry for the text segment. Modify bootstrap1 and segment_loader to load definitions segments directly into the definitions segment and to set the correct definitions pointer in the linkage section.

Modify pre_link_1 to remove the code which combines linkage sections and sets lot entries.

Implications:
System initialization should be faster.
**Title:** Rewrite the generate_mst command  

**Author:** Noel I. Morris

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td></td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td></td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
</tr>
<tr>
<td>BOS</td>
<td></td>
</tr>
<tr>
<td>Salvager</td>
<td></td>
</tr>
<tr>
<td>Ring Zero</td>
<td></td>
</tr>
<tr>
<td>Ring One</td>
<td></td>
</tr>
<tr>
<td>SysDaemon/Admin.</td>
<td></td>
</tr>
<tr>
<td>Runtime</td>
<td></td>
</tr>
<tr>
<td>User Cmd/Subr.</td>
<td></td>
</tr>
<tr>
<td>MPM (Vol, Sect.)</td>
<td></td>
</tr>
<tr>
<td>PLMS (AN #) Tools (MST Generator)</td>
<td></td>
</tr>
<tr>
<td>MOSN (Sect.)</td>
<td></td>
</tr>
<tr>
<td>MPAM (Sect.)</td>
<td></td>
</tr>
<tr>
<td>MSAM (Sect.)</td>
<td></td>
</tr>
</tbody>
</table>

**Category:** Documentation Changes  

**Status:** A  
**Date:** 04/20/76  
**Expires:** 10/20/76

**Objections/Comments:**

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

**Reasons:**
The current MST generator contains unreadable, convoluted, and obscure code. It is impossible to follow the flow of control through the program. It is close to impossible to implement new keywords.

**Proposal:**
Reimplement the generate_mst command so that the flow of control is straightforward and clear.
TITLE: Detect Fatal Process Error Loops

AUTHOR: T. Casey

SUMMARY:

1) Have answering service detect three fatal process errors (per line) within a one-minute interval, stop creating new processes, print an appropriate message to the user, and hang up the phone.

2) Define a new process termination wakeup message: "initxxxx" where xxxx is a word containing an error table code. This wakeup will indicate that process initialization is impossible with the given login (or default) parameters, and a new process should not be created even once more.

3) Have init proc stop calling syserr when it detects an error, but merely terminate the process with "initxxxx", replacing xxxx with an appropriate error_table_code.

4) Have the default initial procedures terminate the process with the "initxxxx" wakeup message when terminations are necessary.

REASONS:

Fatal process error loops tie up the Initializer process; some of them also generate many useless syserr messages (potentially causing messages to be lost).

IMPLICATIONS:

Problems in a user process will not cause problems for the system.
TITLE: Re-implement &command_line off for absentee

AUTHOR: Steve Herbst

Objections/Comments:

STATUS: Written 4/9/76

DATE: 04/20/76

MULTICS Change Request

Category (Check One)

- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools

- Multics Change Request
- User/Operations-visible
- Planned for System MR
- Fixes Bug Number(s)
- Documented in MTB
- Code in other
- 

Interface change? yes no

Incompatible change? yes no

- Performance: Better Same
- Replaces MCR

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

SUMMARY:

Make the &command_line off and &input_line off features work in absentee jobs. These were de-activated last year but are needed.

REASON:

absin file that does &command_line off followed by a file_output puts command lines in the output file. If a user writes an absin with &command_line off or intentionally adds an absin name to an exec_com that uses this feature, he should know not to expect a full script with his output.
TITLE: Install bind_fnp

AUTHOR: Susan Barr

Objections/Comments:

Multics Change Request

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

SUMMARY:
Install bind_fnp to create fnp core image segments from FNP object segments.

REASONS:
The current method of producing fnp core images requires that the system be run under the GCOS simulation. The size limit of 32 K makes it impossible to build a core image to support a configuration with 3HSLA'S.
**Name:** bind_fnp

The bind_fnp command produces a segment which contains a FNP core image from several unbound FNP object segments. It uses two driver segments: the search rules segment which specifies the directories to be searched; and the bindfile, which specifies the configuration that the FNP will support, the ordering of the object segments and the size of certain software tables.

**Usage:**

```
bind_fnp path -control_arg-
```

where:

1. **path** specifies the path of the bindfile. If path does not have a suffix of bind_fnp, it is assumed.

2. **control_arg** can be one of the following two optional control arguments:

   - **-list, -ls** produces a listing segment whose name is derived from the name of the bindfile, with the suffix changed to list. The listing segment is generated for the purpose of dprinting; it contains a copy of the bindfile and a load map as well as any error messages.

   - **-search, -se** is an optional control argument indicating that the user wishes to specify the search rules for the FNP object segments. If used, there must be a segment in the working directory containing an ASCII list of relative pathnames of directories to be searched in the order in which the search is desired. This list must have the same segment name as the bindfile, but with the suffix changed to search.
Following is a list of the delimiters used:

: keyword delimiter. It is used to identify a keyword followed by one or more parameters. A keyword that is followed by no parameters is delimited by a statement delimiter.

; statement delimiter.

, parameter delimiter (the last parameter is delimited by a statement delimiter).

/ begin comment.

*/ end comment.

**Keywords_and_End_Statement**

**hsla**  the parameter is the maximum number of high speed line adapters (HSLA) that this core image will support.

**lsla**  the parameter is the maximum number of low speed line adapters (LSLA) that this core image will support.

**memory**  the parameter is the amount of memory available on the FNP expressed in units of 1024 18 bit words.

**console**  the parameter can be:

- yes console can be configured.
- no console can not be configured.

**printer**  the parameter can be:

- yes printer can be configured.
- no printer can not be configured.

**order**  the parameters are a list of FNP object segments in the order they are to be loaded. The suffix objdk is assumed for all object segments.

**entry**  the parameter specifies the name of the entry at which execution is to begin after the segment is loaded into the FNP.
module

the parameter specifies the name of the FNP object segment that will be described by the following keywords.

type

the parameter specifies the type of FNP object segment. It can be one of the following: hsla, lsla or trace.

size

the parameter is the number of 18 bit words required for table space. The use of the size statement depends on the type specified for this module. If the type is trace, then size represents the size of the trace table. If the type is HSLA or LSLA, then the size represents the size of each HSLA table (or LSLA table). If a site is configuring less than the maximum HSLA's, bind_fnp will remove unused table space from the end of the module.

mask

the parameter is a 6 digit octal number that specifies the modules to be traced. This keyword can only be used if a trace type was also specified. This is used by system programmers for debugging FNP software. (See AN85, Multics Communications)

end;

specifies the end of information to be processed from the bindfile. It must be present at the end of the bindfile.

Notes

Default search rules:

1. Search the working directory.
2. Search >ldd>bos>355_object
3. If the segment is not found in 1. or 2., it is missing.

The segment containing the core image is stored in the working directory. It has the name of the bindfile without the suffix bind_fnp.
# Multics Change Request

**Title:** Install a new put_field

**Author:** R. Schoeman

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIM3 AN#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>355</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Salvager</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ring Zero</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ring One</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SysDaemon/Admin.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MGEN (Sect.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SysProgs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPAM (Sect.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSAM (Sect.)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Objections/Comments:**

Use these headings: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (Optional)

**Summary:** Install a new put_field.

**Reasons:**

1. Bug 1486 must be fixed. Presently, quick put list requests requiring the outputting of more than 848 characters fail.

2. put field chk, an operator which the compiler does not yet produce, has a bug which would keep it from working properly in some cases when it does eventually get put out by the compiler.

3. There is a very minor speed improvement.
<table>
<thead>
<tr>
<th>Headings are:</th>
<th>SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)</th>
</tr>
</thead>
</table>

**SUMMARY:** Fix bug in current 355 software's handling of reset-read.

**REASONS:** The current implementation of reset-read in the 355 can cause the beginning of a typed line to be thrown away while the remainder of the line is sent as input, without the user's being aware of it. As of MR 3.1 this is an occasional problem in echoplex mode; however, it will be a more frequent problem in polite mode starting with MR 4.0.

**IMPLICATIONS:** Occasional extra characters printed on the terminal in echoplex mode (see Detailed Proposal).

**DETAILED PROPOSAL:** Modify 355 control_tables as follows:

1. In polite mode: if a reset-read order arrives when a partial line of input is present, remember the fact and discard the entire line when the user types newline. (The error message that presumably accompanied the reset-read order will not appear until after the user types newline.)

2. In echoplex, "polite": echo an "G" character at the point where the reset-read arrived to indicate that all characters up to that point have been thrown away. The "G" character has been chosen because it is the default kill character; however, it will continue to be used even if the user changes his/her kill character.

**NOTE:** Coded in 355map.
**TITLE:** Fix isof fault handler.

**AUTHOR:** Richard Bratt

| Planned for System | not applicable |
| Fixed Bug Number(s) | not applicable |
| Documented in MTB | not applicable |
| Incompatible Change | no |
| User/Operations-visible Interface Change | no |
| Coded int (B)PL/I ( )ALM ( )other-see below |
| Performance | ( )better ( )same ( )worse |
| DOCUMENTATION CHANGES (specify one or more) |
| MPM (vol,sect) MPAM (sect) |
| MOSN (sect) MSAM (sect) |
| PLMs (AN#) |
| Info Segs |
| Other |
| None (reason) - PLM not yet written. |

**OBJECTIONS/COMMENTS:**

**HEADINGS are:** SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

**SUMMARY:** Prevent isof_fault_handler from attempting to reference pds$stacks in the outer ring.

**REASONS:** You can't get there from here.

**SPECIAL THANKS:** I wish to extend my special thanks to create_if_not_found links for allowing testing to proceed without detecting this error.
**SUMMARY:** Add three new per-process segment attributes. These attributes, which will be stored in the KST, allow processes to influence page control and segment control's behavior toward a segment or directory.

**REASONS:**
1. It has write permission to many segments. This causes it to force these segments out of the cache when they are dumped.
2. If only the dumper is using a segment it will still migrate to the paging device. This is highly undesirable for two reasons. First, the dumper will not reference these pages again (within a PD lap time), so the I/O to move these pages to the paging device is wasted. Second, these "useless" pages effectively reduce the size of the paging device.
3. The dumper knows it will not reference a dumped data segment for a long time, yet the AST replacement algorithm has no way of taking advantage of this knowledge. As a result, the dumper unnecessarily reduces the effective size of the AST.

**DETAILED PROPOSAL:**

Add three segment attributes to a KST entry.

1. **tpd** if set then prevent the system from moving pages of this object to the paging device if only this process
is using the segment.

2. allow_write if not set then do not grant write permission to this process.

3. allow_deactivate if set then allow this process to request deactivation of the object. Only if all processes using the object specified allow_deactivate will hardcore honor the request.

Add a new gate, phcs_$set_kst_attributes, to change these attributes (and the existing attributes: tms, tus, and audit).

Add a new primitive, phcs_$deactivate, which allows a process to suggest the deactivation of an object (see the description of the allow_deactivate kst attribute).

Change the dumper to call phcs_$set_kst_attributes to set tms, tus, tpd, allow_deactivate and to reset allow_write on segments which it dumps but does not use.

Change the dumper to call phcs_$deactivate after dumping a segment.
Name: change_kst_attributes

This command allows a user to change selected per-process attributes of an object.

Usage: change_kst_attributes -target- -attributes-

where:

1. target specifies the object whose kst attributes are to be changed. Either a relative pathname or an octal segment number may be specified. If the name looks like a segment number it must be preceeded by "-name".

2. attributes specifies those attributes which are to be changed. If an attribute is preceeded by the character "-" then the attribute is reset. Otherwise, the attribute is set. Attributes not mentioned are not affected.

tpd don't place pages of this object on the paging device on my account.

allow_deactivate I agree to user suggested deactivations of this object.

allow_write don't prevent me from writing into this object if I have permission to do so.

audit enable auditing.

Note: This command requires phcs_access.
**Name:** priv_change_kst_attributes

This command allows a user to change selected per-process attributes of an object.

**Usage:**

```
change_kst_attributes -target- -attributes-
```

**where:**

1. **target**
   
specifies the object whose kst attributes are to be changed. Either a relative pathname or an octal segment number may be specified. If the name looks like a segment number it must be preceded by "-name".

2. **attributes**
   
specifies those attributes which are to be changed. If an attribute is preceded by the character "-" then the attribute is reset. Otherwise, the attribute is set. Attributes not mentioned are not affected.

- **tpd**
  
don’t place pages of this object on the paging device on my account.

- **tms**
  
don’t update date-time-modified on my account.

- **tus**
  
don’t update date-time-used on my account.

- **allow_deactivate**
  
I agree to user suggested deactivations of this object.

- **allow_write**
  
don’t prevent me from writing into this object if I have permission to do so.

- **audit**
  
enable auditing.

**Note:**

This command requires hphcs_access.
deactivate_seg

This command allows a user to suggest the deactivation of a segment or directory to ring zero. This suggestion will be taken if the object in question is active and if all processes connected to the object have declared their willingness to have users influence the deactivation of the object (see change_kst_attributes and phcs_set_kst_attributes).

Usage: deactivate_seg -target-

where: target specifies the object to be deactivated. Either a relative pathname or an octal segment number may be specified. If the name looks like a segment number it must be preceded by "-name".

Note: This command requires phcs_access.
**Name:** reset_tpd

This command resets the transparent paging device switch of a directory entry. Resetting this switch allows pages of the segment or directory to be placed on the paging device. If the system is not using a paging device then this switch has no effect.

**Usage:** reset_tpd [-path-]

**where:** path specifies the relative pathname of the object whose transparent paging device switch is to be reset.

**Note:** This command requires access to the gate hphcs_.

**Name:** set_tpd

This command sets the transparent paging device switch of a directory entry. Setting this switch prevents pages of the segment or directory from being placed on the paging device. If the system is not using a paging device then this switch has no effect.

**Usage:** set_tpd -path-

**where:**

- `path` specifies the relative pathname of the object whose transparent paging device switch is to be set.

**Note:**

1. This command requires access to the gate `hphcs`.

2. The paging device is not flushed of pages of the object when the transparent paging device switch is set.
**Name:** phcs_$deactivate_seg

This gate allows a process to suggest the deactivation of a segment or directory to ring zero. This suggestion will be taken if the object in question is active and if all processes connected to the object have declared their willingness to have users influence the deactivation of the object (see change_kst_attributes and phcs_$set_kst_attributes).

**Usage:**

```c
declare phcs_$deactivate_entry (ptr, fixed binary (35));
call phcs_$deactivate_seg (segptr, code);
```

**where:**

- `segptr` is a pointer to an object.
- `code` is a status code.
**Name:** phcs$_s$set$_k$st$_a$ttributes

This gate allows a process to change selected per-process attributes of an object.

**Usage:**

```
dc1 1 kst_attributes aligned based (kstap),
    2 set unaligned,
        3 (allow_write,
            must_be_zero_1,
            must_be_zero_2,
            tpd,
            audit,
            explicit_deactivate_ok) bit (1),
        3 pad bit (39),
    2 value unaligned,
        3 (allow_write,
            tms,
            tus,
            tpd,
            audit,
            explicit_deactivate_ok) bit (1),
        3 pad bit (30);
```

declare phcs$_s$set$_k$st$_a$ttributes entry (fixed bin (17),
pointer, fixed binary (35));
call phcs$_s$set$_k$st$_a$ttributes (segno, kstap, code);

**where:**

- **segno** is a segment number.
- **kstap** is a pointer to the structure described below.
- **code** is a status code.
**Name:** hphcs_set_kst_attributes

This gate allows a process to change selected per-process attributes of an object.

**Usage:**
```plaintext
declare hphcs_set_kst_attributes entry (fixed bin (17), pointer, fixed binary (35));
call hphcs_set_kst_attributes (segno, kstap, code);
```

**Where:**
- **segno** is a segment number.
- **kstap** is a pointer to the structure described below.
- **code** is a status code.

```plaintext
dcl 1 kst_attributes aligned based (kstap),
  2 set unaligned,
    3 (allow_write,
      tms, tus, tpd, audit,
      explicit_deactivate_ok) bit (1),
    3 pad bit (39),
  2 value unaligned,
    3 (allow_write,
      tms, tus, tpd, audit,
      explicit_deactivate_ok) bit (1),
    3 pad bit (30);
```
14. bit_count  is the bit count associated with the segment.
15. did  specifies the secondary storage device (if any) on which the segment currently resides.
16. pad3  is unused space in this structure.
17. copy_sw  contains the setting of the segment copy switch.
18. pad4  is unused space in this structure.
19. rbs  contains the ring brackets of the segment right justified in the 8-bit field.
20. uid  is the segment unique identifier.

Entry:  hcs_$status_minf

The hcs_$status_minf entry point returns the bit count and entry type given a directory and entryname. Status permission on the directory or nonnull access on the entry is required to use this entry point.

Usage

```
declare hcs_$status_minf entry (char(*), char(*), fixed bin(1),
    fixed bin(2), fixed bin(24), fixed bin(35));
call hcs_$status_minf (dir_name, entryname, chase, type, bit_count, code);
```

where:

1. dir_name  is the same as for the hcs_$status_ entry point above. (Input)
2. entryname  is the same as for the hcs_$status_ entry point above. (Input)
3. chase  is the same as for the hcs_$status_ entry point above. (Input)
4. type  specifies the type of entry. (Output) It can be:
   0 link
   1 segment
   2 directory
5. bit_count  is the bit count. (Output)
6. code  is a storage system status code. (Output)

Entry:  hcs_$status_mins

This entry point returns the bit count and entry type given a pointer to the segment. Status permission on the directory or nonnull access to the segment is required to use this entry point.
TITLE: Change pull to use get_line_length_

AUTHOR: R.A. Barnes

SUMMARY:

Have pull use get_line_length_$switch rather than ios_$changelmode to derive the line length, which is used when formatting error messages.

REASONS:

The knowledge of how to calculate the line length should be in as few places as possible as mode strings get longer (as they have for MR 4.0). pull won't have to be changed to use a longer character string to pass to ios_$changelmode.
**MULTICS CHANGE REQUEST**

**TITLE:** Fix message coordinator to work with 28.3

**AUTHOR:** Robert S. Coren

**Planned for System:** MR 4.0

**Fixes Bug Number(s):** not applicable

**Documented In MTB:** not applicable

**Incompatible Changes:** no

**User/Operations-visible Interface Changes:** no

**Coded In:** ( )PL/I ( )ALM ( )other-see below

**Performance:** ( )better ( )same ( )worse

**DOCUMENTATION CHANGES (specify one or more):**

- MPM (vol,sect)
- MOSN (sect)
- PLM (AN#)
- Info Segs
- Other

**None (reason) bug fix**

**OBJECTIONS/COMMENTS:**

---

**SUMMARY:** Fix mc_tty_ to be able to handle returned modes string of more than 128 characters.

**REASONS:** With new modes invented for 28-3 (4.0) the string returned by hcs_mctty_order is longer than 128 characters.

**IMPLICATIONS:** None

**DETAILED PROPOSAL:** Change buffer in mc_tty_ for modes string to be 256 characters, and make it ignore error_table_$smallarg when returned by hcs_mctty_order ("modes").

**NOTE:** Submitted to MIT as emergency fix.
### MULTICS CHANGE REQUEST

**TITLE:** NSS Bug Fixes, round 3, emergencies  
**AUTHOR:** Bernard Greenberg  
**Planned for System:** MR 4.0  
**Fixes Bug Number(s):** not applicable  
**Documented in MTB:** not applicable  
**Incompatible Change:** no  
**User/Operations-visible Interface Change:** no  
**Coded in:** (H)PL/I ( )ALM ( )other-see below  
**Performance:** ( )better ( )same ( )worse  

### DOCUMENTATION CHANGES (specify one or more)

<table>
<thead>
<tr>
<th>HPH (vol,sect)</th>
<th>MPAM (sect)</th>
<th>MOSN (sect)</th>
<th>MSAM (sect)</th>
<th>PLMs (AN#)</th>
<th>Info Segs</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>61</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional) headings are:

**SUMMARY:** Fix critical bugs in NSS involving the On-line Salvager and disk dim.  
**REASONS:** The online salvager produces damage in certain instances, and the disk dim crashes the system erroneously in other circumstances.  
**IMPLICATIONS:** Greater reliability.  
**DETAILED PROPOSAL:** Fix an uninitialized variable in a syserr call in disk_control. Change the on-line salvager to correctly compute directory header size. Do not allow VTOCE freeing of active segments by salv_truncate if deactivation failed. Correct file-map size computation in salv_check_map.  

(installed emergency fixes, MSS 2b.2a)
SUMMARY: Add an option to the backup dumper (current 4.0 version) to dump only segments on a selected physical volume.

REASONS: Although the current plans for the storage system include schemes for moving segments between physical volumes on demand via system faults, interim measures are needed to condense logical volumes by removing a physical volume. A simple and reliable way to do this is to dump all of the segments on a given physical volume via a (partial) hierarchy walk of the complete dumper, while the system is running. The system may then be brought up without that physical volume, via changing logical (hierarchy) volume registration data, and the dumps so produced, plus all incrementals since, reloaded with the -pvname option. The connection failures which result from the vanished pack are transparent to the delete operation, modulo quota accounting.

IMPLICATIONS: An interim method of condensing logical hierarchy volumes, which is needed at MIT in the immediate future. A vtoce-check directory salvage must be performed after a volume is so vanished, to correctly recompute quota used totals, so that the reload will not record-quota overflow, and so that quota may be kept accurate.
Titre: Invoke Online Salvager when Problems are Suspected.
Auteur: Bernard Greenberg

Planifié pour le système: MK 4.0
N° de correction: non applicable
Documenté dans MTB: non applicable
Change incompatible: non
Interface utilisateur change visible: non

Code: (□) PL/I (□) ALM (□) autre voir ci-dessous
Performance: (□) meilleur (□) meilleur (□) pire

DOCUMENTATION CHANGES (specifiez un ou plus)

<table>
<thead>
<tr>
<th>MPM (vol,sect)</th>
<th>MPAM (sect)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSN (sect)</td>
<td>MSAM (sect)</td>
</tr>
<tr>
<td>PLMs (AN#)</td>
<td>61</td>
</tr>
<tr>
<td>Info Segs</td>
<td></td>
</tr>
<tr>
<td>Autres</td>
<td></td>
</tr>
</tbody>
</table>

OBJECTIONS/COMMENTS:

Headings are: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

SUMMARY: The Online salvager is currently invoked on crawlouts from the hardcore ring when a directory is locked for modification. Change the policy so that it is invoked whenever a crawlout occurs with a directory locked.

REASONS: The rationale behind invoking the Salvager when a crawlout with a locked-for-mod directory occurs is that the program which is crawling out may have left the directory in an inconsistent state. More usually, crashes or other mishaps leave directories in an inconsistent state, and a later user of the directory, who may be only reading it, crawls out while he has it locked because the directory itself is bad, e.g., contains bad relative pointers. The system should protect itself by salvaging any directory it suspects may have caused it to crawl out.

IMPLICATIONS: Greater reliability. The situation of "Every time I list the directory I get a crawlout, can someone please online-salvage it?" goes away. A more subtle implication is that the process of a directory reader now has one more reason for needing raw write access to directories on which he/she does not have modify permission. This contradicts some models of time-shared computer systems.
# Move FNP software to new library

**Summary**

Move FNP software from the BOS libraries to a new library.

**Reasons**

Since FNP software will soon be loaded by Multics instead of BOS, it seems like a good time to move the FNP software out of the BOS libraries. This will also provide a structure for new and different kinds of FNP and possibly RNP software.

**Detailed Proposal**

Create the directory `>ldd>communications` with the added name `comm`. Create the following directories:

```
>ldd>comm>fnp
>ldd>comm>fnp>source (addname s)
>ldd>comm>fnp>object (addname o)
>ldd>comm>fnp>info
```

The object directory will contain, besides all assembled object, the core image created by `bind_fnp`.

The info directory will contain the `bind_fnp` search and control files and the macro library file (`355_macros`), which will be moved from the tools library.
The hardcore object library will have to contain a link to \texttt{ldd} -> \texttt{fnp} -> \texttt{mcs} to allow \texttt{mcs} to be put on hardcore tapes. This link can be put in the BOS directory until MCS is loaded by Multics.

The FNP directory is required to allow for expansion of this library to contain other type of communications software.
### HEADINGS
- SUMMARY
- REASONS
- IMPLICATIONS
- DETAILED PROPOSAL (optional)

### SUMMARY

Install a new I/O assignment manager with very simple goals and a corresponding implementation. The new ioam will be used only to call the appropriate subsystem when the process which owns one of its devices is being destroyed.

The DST will be deleted at this time, since it is no longer used.

### REASONS

The current ioam provides features which are no longer used or required. Most of these were implemented in HCP, since it now manages peripheral I/O devices. Other features of the ioam are no longer used because of their inefficiency.

### IMPLICATIONS

Certain tools (gds, unassign_force) become obsolete and should be deleted.

### DETAILED PROPOSAL

Provide a new ioam with the following entries:

- `assign (devx, unassign_handler, code)`
- `unassign (devx, code)`
- `preempt (processid, devx, code)`
- `process_release (processid)`
The assign and unassign entries are used by the subsystem to register their
devices with the ioam_. The preempt entry is used by the subsystem to get
back one of its own devices, and will result in a call to the
unassign_handler passed in the assign call. The process_release entry is
used by deact_proc to force calls to the unassign_handler for each device
owned by the indicated process.

The NCP and tty DIM will be converted to call these routines.

NOTES

This MCR affects MCR 1626 and MTB 254, which recommend the unwiring of the
DST. This phase of implementation will be skipped, since this MCR proposes
the deletion of the DST.
MULTICS CHANGE REQUEST

TITLE: FNP load/dump hardcore primitives

AUTHOR: Mike Grady

Planned for System: MR5.0
Fixes Bug Number(s): not applicable
Documented in MTB: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: (M)PL/I ( )ALM ( )other-see below
Performance: ( )better ( )same ( )worse

DOCUMENTATION CHANGES (specify one or more)
MPH (vol,sect) MPAM (sect)
HOSN (sect) MSAM (sect)
PLMs (AN#) AN05
Info Segs
Other

OBJECTIONS/COMMENTS:

Headings are: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

SUMMARY

Add hardcore primitives to support loading and dumping of FNP's.

REASONS

This change will enable Multics to load and dump FNP's during normal system operation. It will also simplify FNP software maintainence (no more simultaneous BOS/hardcore installations), and should increase system availability since Multics will automatically reload an FNP after it crashes.
TITLE: Modify MCS for new binder

AUTHOR: Mike Grady

Planned for System: MR5.0
Fixes Bug Number(s): not applicable
Documented in MTB: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: ( ) PL/I ( ) ALM ( ) other-see below
Performance: ( ) better ( ) same ( ) worse

CATEGORY (check one):
( ) Lib. Maint. Tools
( ) Sys. Anal. Tools
( ) Sys. Prog. Tools
( ) 355
( ) BOS
( ) Salvager
( ) Ring Zero
( ) Ring One
( ) SysDaemon/Admin
( ) Runtime
( ) User Command/Subr

SUMMARY

Delete all pre-init code from MCS modules to simplify builds with new MCS binder.

REASONS

The old binder ran as a simulated 355 and was able to execute small pieces of code in each module that performed certain "pre-initialization" functions. The new binder will not be able to execute this code and these functions will be performed at actual 355 initialization time.

DETAILED PROPOSAL

Coded in 355map.
TITLE: Define New Salvager Options

AUTHOR: A. Kobziar

SUMMARY: Implement the following options for the directory salvager:

- **rebuild**
  - rebuild every directory.
- **pathname**
  - print the pathname of every directory.
- **check_vtoc**
  - check branch-vtoc connections.
- **long**
  - all of the above.
- **level n**
  - salvage only to a hierarchy depth of n.

Implement the following options for both directory and volume salvaging:

- **dump**
  - dump directories and/or vtoces that have significant inconsistencies.
- **debug**
  - output messages about the normal deletions of \( \text{process\_dir\_dir} \) and \( \text{system\_library\_l\_segments} \), and deciduous and per_process vtoces.
- **console**
  - direct output to the operator's console rather than to a printer, and suppress dump, debug, and pathname options.

Implement the following options for volume salvaging:

- **all**
  - salvage all registered but not in use disks.
- **check_dir**
  - check vtoce-branch connection.
- **copy**
  - salvage a copy of an in-use disk (necessary for a check_dir of all riv disks).

The salv card allows administrative specification of same options. Preceding an option with "no", as "nodump" will override the card specification.
**TITLE:** Change cutoff messages printed at login

**AUTHOR:** T. Casey

**STATUS**
- Written: 4/20/76
- Expires: 10/24/76

**Category (Check One)**
- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools
- BOS
- Salvager
- Ring Zero
- Ring One
- SysDaemon/Admin.
- runtime
- User Cmnd/Subr.

**DOCUMENTATION CHANGES**
- Specify One or More
  - MPM (Vol. Sect.)
  - PLMS (AN #)
  - MSAM (Sect.)

**Objections/Comments:**
- Info Segs
- Other (Name)
- None (Reason)

**Use these headings:** Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

**Summary:** Replace the messages "Your account is almost out of funds" and "Your account is near its termination date" by messages that have more useful information; and make the thresholds at which these messages start appearing settable by each project administrator for the users on his project.

**REASONS:** These messages are usually printed during the entire last month of the fiscal year, for every user, and are therefore, considered an annoyance, and ignored. Thus, people in real danger of being cut off do not get warned.

**DETAILED PROPOSAL:** Print the following messages (when appropriate):

- Your project's account has less than \( d \) days until its termination date (\( \sqrt{d} \)).
- Your project's account has less than \$\( d \) of its funds remaining.

Accept the following statements in PMF's.

- Warn_days: \( N \); /* warn users starting \( N \) days from cutoff*/
- Warn_percent: \( P \); /* warn users when less than \( P \% \) of funds left*/
- Warn_dollars: \( D \); /* warn users when less than \$D.00 \) left*/

If project administrators do not specify a value for any one of these variables, the defaults will be as they are now: \( N = P = D = 10 \). Those default values will continue to be used to flag projects near cutoff in the cutrpt. The figures set by the project administrator will only be used when deciding whether to print a message at login time.
**SUMMARY:** Fix several bugs related to the printing of lengths and records:

1. use of the -records control argument interferes with the printing of link paths, when both branches and links are being printed.

2. "Lengths" is printed in the long heading in all but one of the appropriate cases.

3. The brief heading always says "recs=-", even when lengths are being printed.

**Fix error checking to:**

1. not complain when the same (set of) control argument(s) is repeated.

2. when unable to get status on a directory (to see if it is really an msk), report the error but continue rather than aborting.
SUMMARY: hcs_make_seg behaves differently when given directory names of "" and "". Since the PL/I string comparison operator does not distinguish these two strings this behavior is confusing to users. As a result, it is proposed that the "if length(dname) = 0 then ..." test in make_seg be replaced by "if dname = "" then ..."".
Multics Change Request

| TITLE: | Fix bugs in as_init_ and load_ctl_ |
| AUTHOR: | T. Casey |
| STATUS | Written |
| DATE | 4/16/76 |

- Coded in □ XPL/I □ AIM □ other-explain in DETAILED PROPOSAL
- Planned for System MR4.0
- Fixes Bug Number(s) see below
- Documented in MTH 355
- User/Operations-visible Interface change? □ yes □ no
- Incompatible change? □ yes □ no
- Performance: □ Worse □ Better □ Same
- Replaces MCR

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DOCUMENTATION CHANGES</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCR 1812</td>
<td></td>
</tr>
<tr>
<td>Expires 10/23/76</td>
<td></td>
</tr>
</tbody>
</table>

Objections/Comments:

Use these headings: Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.

**SUMMARY:**

Change answering service initialization to

1) initialize work classes only after other required initialization has been done;
2) print error messages on initializer console for all errors that occur during initialization, even if callee may have already done so;
3) allow the priority scheduler to be turned on by installing a new MGT, if it was off at startup time.

**REASONS:**

1) work class initialization fails in some unusual circumstances.
2) when it does fail, no error messages are printed; the initializer process just returns to command level. The way as_init_ is written, there might be other errors that also cause this behavior.
3) If the priority scheduler is turned off to get around this problem, it cannot be turned back on until the next bootload.

**IMPLICATIONS:**

Failures during initialization will result in one to three different messages, rather than zero or one message.
SUMMARY: Fix tty_write so as not to put out extraneous white space at the ends of certain lines.

REASONS: When tty_write only converts part of the user-supplied output, and the portion converted ends in white space not followed by a newline, the white space is sent to the terminal; if the remainder of the line consists exclusively of white space, the white space sent with the first call should not be transmitted.

IMPLICATIONS: None.

DETAILED PROPOSAL: Have tty_write remember not only the current column position for each channel, but also the position implied by trailing white space; if it is necessary to break up the user's output and the first portion ends in white space, don't send it until it is known to be followed by a printing graphic.