

To: Distribution
From: Betsy
Date: 17 March 1976
Subject: Multics Change Requests

Enclosed are copies of Multics Change Requests which were approved from March 1 - 15, 1976.

Multics Project internal working documentation. Not to be reproduced or distributed outside the Multics Project.

TITLE: Install new MST copy/merge/list/excerpt/write.		STATUS	DATE
AUTHOR: Bernard Greenberg		Written	1/26/76
		Status	2/3/76 A 3/2/76
		Expires	9/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Category (Check One)	
-Planned for System MR 4.0		<input type="checkbox"/> Lib. Maint. Tools	
-Fixes Bug Number(s)		<input type="checkbox"/> Sys. Anal. Tools	
-Documented in MFB		<input checked="" type="checkbox"/> Sys. Prog. Tools	
-User/Operations-visible		355	
Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no		EOS	
-Incompatible change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no		Salvager	
-Performance: <input checked="" type="checkbox"/> Better <input type="checkbox"/> Same		Ring Zero	
<input type="checkbox"/> Worse		Ring One	
-Replaces MCR		SysDaemon/Admin.	
		Runtime	
		User Cmd/Subr.	
		MSAM (Sect.)	
Objections/Comments: add -brief control argument; remove abbreviations from Tools (except epm). Some facility for recording the author and contents of the tape must be developed for MST's.		Info Segs	
		Other (Name) MOH	
		None (Reason)	

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

Summary: Replace copy_mst with a new version, with increased function for editing MST's.

Reasons: 1) The installed copy_mst is in version 1 PL/1, with ios_, and has a needlessly difficult interface.
2) The need has often been felt for a tool to replace selected segments on a MST, or extract segments for inspection or other use.

Implications: 1) vastly improved interface for copy_mst (i.e., copy_mst tape1 tape2 vs. "Type # of tapes to be copied:").
2) The ability to copy multi-reel MST's, which have not happened in several years, will be lost.
3) The ability to copy several tapes in one invocation, which is highly nonstandard, and not particularly useful will be lost.
4) It will be very valuable for debugging and system development to be able to replace single segments on tapes. This implies an increased effectiveness of the CISL Development Site, whereby users can fix tapes made at the MIT Site with the Network Transfer of a single segment, or simply a patch.
5) An ability to create usable MST's with no traceable origin or documentation is introduced. This is a tradeoff in a debugging situation. Like all powerful things, such a tool must be used carefully.

Detailed Proposal: See attached documentation.

Name: copy_mst, cpm

copy_mst is used to copy a Multics System Tape (either a Multics or BOS bootload tape) onto another reel of tape.

Usage: copy_mst tapeid-1 tapeid-2

WHERE tapeid-1 is the tape reel identifier of the tape to be copied, and tapeid-2 is the tape reel identifier of the tape onto which the copy is to be made.

Name: merge_mst, mgm

merge_mst is used to copy a Multics System Tape (either a Multics or BOS bootload tape) onto another reel of tape, replacing selected segments with segments from the Storage System.

Usage: merge_mst tapeid-1 tapeid-2 : name-1 name-2 ... name-n

WHERE tapeid-1 is the tape reel identifier of the tape to be copied, and tapeid-2 is the tape reel identifier of the new MST to be made. name-1 to name-n are segment names (possibly star names, see "match_star_name" in the MPM). Any segment ^{on the tape} which matches any of the names will be sought in the current working directory, and replaced in the tape copy. If no names are supplied, "***" is the default. If a segment with a separate linkage segment is replaced, a separate linkage segment will be sought in the working directory to replace it. However, if there is none, the linkage segment will be obtained from the object segment in the working directory. If a segment on the tape being replaced is not an object segment, but the matching segment in the working directory is, only the text of the segment will be written to the new tape. A separate linkage section cannot be replaced without replacing the segment from whence it was separated.

Each name may be preceded by the "-stop" (-sp) control argument, in which case a call to debug is made immediately before the segment is written out, whether or not it has been replaced. This allows the user to inspect or modify the SLT entry in any arbitrary way. Before such a call is made, an informational message giving the address of the ~~XX~~ segment and the SLT entry is printed out.

An informational message is printed out every time a segment is replaced on the tape with one from the working directory.

COMPILATION LISTING OF SEGMENT call_out

Compiled by: Multics PL/I Compiler of November 24, 1975.

Compiled on: 02/24/76 1530.3 ast Tue

Options: list map optimize

```

1 call_out: procedure;
2
3 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * */
4 /*
5 /* This procedure demonstrates the interactions between a */
6 /* user process and the Initializer in order to request */
7 /* an auto call line be assigned to the user. This program */
8 /* performs several functions: */
9 /*
10 /*      1) Requests the Initializer to call a specified */
11 /*         phone number and assign the line to this */
12 /*         process. */
13 /*      2) Goes blocked to wait for confirmation from the */
14 /*         Initializer. */
15 /*      3) Sets up an event call channel to be invoked */
16 /*         if the line is hungup from the other end. */
17 /*      4) Attaches the line to the process through an */
18 /*         iox_ IO module. */
19 /*      5) Provides an entry to allow the user to */
20 /*         disconnect the line. */
21 /*
22 /* All of these functions must be performed by any user */
23 /* program which uses the Auto Call Facility. */
24 /*
25 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * */
26
27 dcl  nargs fixed bin,
28      arg_ptr pointer,
29      arg_len fixed bin,
30      arg char(arg_len) based(arg_ptr);
31
32 dcl (  dialed,          /* Internal static switches used to keep track of */
33      attached,        /* the progress of a call_out attempt */
34      opened ) bit(1) aligned internal static initial("0"b);
35
36 dcl 1 dial_manager_arg internal static aligned, /* Structure used in calls to dial_manager_, kept */
37     2 version fixed bin initial(1),          /* in internal static so that subsequent calls to */
38     2 phone_number char(22),                 /* call_out will use the same values. */
39     2 wakeup_channel fixed bin(71),
40     2 tty_name char(32);
41
42 dcl 1 reply aligned, /* Used in call to ipc_block, contains message */
43     2 channel fixed bin(71), /* from Initializer when ipc_block returns */
44     2 message fixed bin(71),
45     2 sender bit(36),
46     2 origin bit(36),
47     2 wait_list_index fixed bin;
48
49 dcl  infop pointer, /* Used to reference "hungup" message from ipc_ */
50     1 info based(infop) like reply; /* event call */
51
52 dcl 1 status_flags aligned, /* Returned by convert_dial_message_ */
53     2 dialed_up bit(1) unaligned, /* = line is dialed and assigned */
54     2 hung_up bit(1) unaligned, /* = line has been disconnected */
55     2 control bit(1) unaligned, /* Not used in Auto Call functions */

```

```

56     2 pad blt(33) unaligned;
57
58 dcl 1 wait_list aligned,                /* Used in calls to ipc_block. Note that 'chan' */
59     2 n_channels fixed bin initial(1), /* contains the same value as is kept in      */
60     2 chan(1) fixed bin(71);           /* dial_manager_arg.wakeup_channel           */
61
62 dcl  my_name char(32),
63     stream_name char(32),              /* For call to lox_attach_lname              */
64     dev_name char(32),                 /* Device name returned by convert_dial_message_ */
65     dim_name char(32),                 /* Dim name returned by convert_dial_message_   */
66     attach_description char(64);       /* For call to lox_attach_lname              */
67
68 dcl  locb_ptr pointer;
69
70 dcl  n_dev fixed bin,                  /* Number of tty lines attached by this process */
71     code fixed bin(35);
72
73 dcl  call_out$hangup entry (pointer),
74     convert_dial_message$return_io_module entry (fixed bin(71), char(*), char(*), fixed bin,
75     1 aligned like status_flags, fixed bin(35)),
76     dial_manager$dial_out entry (pointer, fixed bin(35)),
77     dial_manager$sterminate_dial_out entry (pointer, fixed bin(35));
78
79 dcl  com_err_ entry options (variable),
80     convert_ipc_code_ entry (fixed bin(35)),
81     cu$arg_count entry (fixed bin),
82     cu$arg_ptr entry (fixed bin, pointer, fixed bin, fixed bin(35)),
83     lox_attach_lname entry (char(*), pointer, char(*), fixed bin(35)),
84     lox_close entry (pointer, fixed bin(35)),
85     lox_detach_locb entry (pointer, fixed bin(35)),
86     lox_open entry (pointer, fixed bin, bit(1) aligned, fixed bin(35)),
87     ipc_block entry (pointer, pointer, fixed bin(35)),
88     ipc_create_ev_chn entry (fixed bin(71), fixed bin(35)),
89     ipc_decl_ev_call_chn entry (fixed bin(71), pointer, pointer, fixed bin, fixed bin(35));
90
91 dcl  error_table_$badcall external fixed bin(35);
92
93 dcl ( addr,
94     null ) builtin;
95
96

```

```

97
98 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
99 /*
100 /* This is the main entry point. It is called with a
101 /* telephone number to be called and the name of a stream
102 /* that the line, once assigned, should be attached to.
103 /*
104 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
105
106     my_name = "call_out";
107
108     if dialed
109     then do;
110         call com_err_ (error_table_$badcall, my_name,
111             "Call out already in use, "a previously dialed.", dial_manager_arg.tty_name);
112         return;
113     end;
114
115     call cu_$arg_count (nargs);
116     if nargs ^= 2
117     then do;
118         call com_err_ (error_table_$badcall, my_name,
119             "'/Usage is:      call_out <phone_number> <stream_name>");
120         return;
121     end;
122
123     call cu_$arg_ptr (1, arg_ptr, arg_len, code);
124     dial_manager_arg.phone_number = arg;
125
126     call cu_$arg_ptr (2, arg_ptr, arg_len, code);
127     stream_name = arg;
128
129 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
130 /*
131 /* We now have the phone number and the stream. Before we
132 /* can request that the Initializer place the call we must
133 /* set up an InterProcess Communication channel over which
134 /* the Initializer will signal the success or failure of the
135 /* call. Note that the channel must be passed to the
136 /* Initializer so that it will know what channel to use.
137 /*
138 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
139
140     call ipc_$create_ev_chn (wait_list.chan(1), code);
141     if code ^= 0
142     then do;
143         call convert_ipc_code_ (code);
144         call com_err_ (code, my_name,
145             "Unable to create ipc event channel.");
146         return;
147     end;
148
149     dial_manager_arg.wakeup_channel = wait_list.chan(1);
150
151     call dial_manager_$dial_out ( addr (dial_manager_arg), code);
152     if code ^= 0
153     then do;
154         call com_err_ (code, my_name,
155             "Unable to call out to "a.", dial_manager_arg.phone_number);

```

```

156         return;
157     end;
158
159     call ipc_block ( addr (wait_list), addr (reply), code);
160     if code /= 0
161     then do;
162         call convert_ipc_code_ (code);
163         call com_err_ (code, my_name,
164             "Error calling ipc_block.");
165         return;
166     end;
167
168     call convert_dial_message_return_io_module (reply.message, dev_name, dim_name,
169         n_dev, status_flags, code);
170     if ~status_flags.dialed_up
171     then do;
172         call com_err_ (code, my_name,
173             "Error: call out failed.");
174         return;
175     end;
176
177     dialed = "1";
178     dial_manager_arg.tty_name = dev_name;
179
180 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
181 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
182 /* Now that the line has been assigned, we must prepare a */
183 /* handler for a possible hang up at the other end. Note */
184 /* that we must use the same IPC channel that we previously */
185 /* passed to the initializer, as this is the only channel */
186 /* that is known to the initializer. */
187 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
188 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
189
190     call ipc_decl_ev_call_chn (wait_list.chan(1), addr (call_out$hangup), null(), 1, code);
191     if code /= 0
192     then do;
193         call convert_ipc_code_ (code);
194         call com_err_ (code, my_name,
195             "Unable to set hangup handler, line will be attached anyway.");
196     end;
197
198     attach_description = dim_name || " " || dev_name;
199
200     call lox_attach_ioname (stream_name, locb_ptr, attach_description, code);
201     if code /= 0
202     then do;
203         call com_err_ (code, my_name,
204             "Unable to attach stream ~a to device ~a, line will be hung up.", stream_name, dev_name);
205         go to term_dial;
206     end;
207
208     attached = "1";
209
210     call lox_sopen (locb_ptr, 3, "0", code);
211     if code /= 0
212     then do;
213         call com_err_ (code, my_name,
214             "Unable to open stream ~a, line will be hung up.", stream_name);
215         go to detach_locb;

```

```
216         end;
217
218         opened = "1"b;
219
220         call com_err_ (0, my_name,
221             "'a attached to ^a through ^a.", stream_name, dev_name, dim_name);
222
223         return;
224
225
```



```

226
227 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * */
228 /* * * * * * * * * * * * * * * * * * * * * * * * * * * */
229 /* This entry point is used to terminate an Auto Call */
230 /* connection. Since we allow only one active connection */
231 /* at a time and keep the channel name in internal static, */
232 /* there are no arguments to this entry. */
233 /* * * * * * * * * * * * * * * * * * * * * * * * * * * */
234 /* * * * * * * * * * * * * * * * * * * * * * * * * * * */
235
236 end_call_out:      entry;
237
238     my_name = "call_out$end_call_out";
239
240     if ~dialed
241     then do;
242         call com_err_ (0, my_name,
243             "No lines currently attached.");
244         return;
245     end;
246
247     if opened
248     then call iox_$close (locb_ptr, code);
249     opened = "0"b;
250
251 detach_locb:
252     if attached
253     then call iox_$detach_locb (locb_ptr, code);
254     attached = "0"b;
255
256 term_dial:
257     if dialed
258     then call dial_manager_$terminate_dial_out ( addr (dial_manager_arg), code);
259     dialed = "0"b;
260
261     call com_err_ (0, my_name,
262         "Connection terminated, ~a hung up.", dial_manager_arg.tty_name);
263
264     return;
265
266

```

```

267
268 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
269 /*                                                                    */
270 /* This entry will be invoked as an IPC event call                    */
271 /* procedure if the Auto Call line is hung up other than             */
272 /* by our user's request. Note that we first check to see           */
273 /* that the wakeup is valid, and, if so, we then perform a          */
274 /* variety of cleanup operations.                                     */
275 /*                                                                    */
276 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
277
278 hangup:   entry (info);
279
280         my_name = "call_out$hangup";
281
282         call convert_dial_message_$return_lo_module (info.message, dev_name, dlm_name,
283             n_dev, status_flags, code);
284         if code ^= 0
285             then do;
286                 call com_err_ (code, my_name,
287                     "Hangup wakeup received, but unable to convert message from dial_control.");
288                 return;
289             end;
290
291         if dev_name ^= dial_manager_arg.tty_name
292             then do;
293                 call com_err_ (0, my_name,
294                     "Hangup wakeup ignored for unknown channel ^a.", dev_name);
295                 return;
296             end;
297
298         if ^status_flags.hung_up
299             then do;
300                 call com_err_ (0, my_name,
301                     "Spurious wakeup ignored for ^a.", dev_name);
302                 return;
303             end;
304
305         if opened
306             then call iox_$close (locb_ptr, code);
307         opened = "0"b;
308
309         if attached
310             then call iox_$detach_locb (locb_ptr, code);
311         attached = "0"b;
312
313         dialed = "0"b;
314
315         call com_err_ (0, my_name,
316             "Hangup occurred, ^a hung up and detached.", dev_name);
317
318         return;
319
320 end call_out;

```

II. System Requirements for the Auto Call Facility

In order to make use of the Auto Call Facility, a site must first make changes to the hardware and to several system databases. The rest of this document covers:

1. Hardware Requirements
2. Configuration Cards
3. The Channel Master File (CMF) and the Channel Definition Table (CDT)
4. Channel Access Control
5. System Messages

1. Hardware Requirements for Auto Call

The Multics Auto Call Facility utilizes three hardware components. The first is an HSLA Auto Call board - board number HSC 351, supplied by Honeywell. The second is a Bell system auto call unit - unit number 801A ACU, supplied by the telephone company. The third component is a standard Bell 103A modem.

The HSC 351 board includes installation instructions and one cable for connection to the Bell 801A auto call unit. The Bell 103A modem is also connected to the HSC 351 board, in this case by a standard dataset cable which should be ordered with the modem.

2. Configuration Cards

The HSC 351 auto call board uses an entire HSLA slot, unlike some other HSLA boards which allow two channels/slot. Thus HSLA slot 5, which, with a dual asynchronous HSLA board, provides HSLA channels 5 and 21, will provide only channel 5, an asynchronous 300 baud line, when the HSC 351 is used. Configuration cards should therefore be modified to indicate a single 300 baud channel for the appropriate slot.

3. The Channel Master File and the Channel Definition Table

When the appropriate hardware and configuration card changes have been made the Channel Master File should be modified. The only difference between a standard entry and an auto call entry is that the auto call entry should indicate a service type of "autocall". For example: "service: autocall;". Additional

changes to such items as the charge rate for channel usage may be made but are not required. Once the Channel Master File has been updated it should be converted to a Channel Definition Table using the cv_cmf command and the CDT installed.

4. Channel Access Control

The Channel Master File provides the ability to limit use of a channel by Access Isolation Mechanism access class. This limitation is observed by the Auto Call Facility, but does not provide specific user/project oriented access control. In order to provide this additional control, the Auto Call Facility makes use of the Access Control Segment concept first implemented by the Resource Control Package. Before the Facility may be used, an Access Control Segment must exist in >system_control_dir>rcp for each available auto call channel. The segment(s) must have a name of the form <channel name>.acs, for example: "tty605.acs". When a user process requests that an auto call channel be assigned, the user's access to the Access Control Segment will be checked and the line will be assigned only if the user has at least "r w" access to the segment.

5. System Messages

With the addition of functionality to the Initializer, a number of new messages may appear in the Initializer's log and output. All of the new messages are produced by the procedure dial_ctl_ and are written to both the log and the Initializer console. In the messages listed below, the following generic terms are replaced with values as indicated:

<channel>

is the ASCII name of the tty channel involved.

<error message>

is an error message extracted from error_table_ because of an error code returned to dial_ctl_.

<process id>

is the 12 digit octal process id assigned to a process when it is created. In the messages listed below, the process id is used only when the Initializer is unable to locate the process in question.

<project>

is the project id portion of a process identifier. (ASCII)

<reason>

is a standard code for the reason for a process termination, for example, "new_proc".

<telephone number>
is the ASCII string representing the telephone number being called.

<user>
is the user id portion of a process identifier.
(ASCII)

"dial_ctl_: now terminating dial_out of <channel> for <user>.<project>"

This message will be outputted whenever an auto call line hangs up for whatever reason. In general, it will be preceded by one of the messages below indicating the reason (line hungup, user requested, etc) the connection is being terminated.

"dial_ctl_: dial_out line being hung, <channel>, reason: <reason>"

This message occurs when a process with an active auto call connection terminates. <reason> is the cause of the process termination.

"dial_ctl_: can't find process <process id>"

This message results when the Initializer receives an auto call request from a process that can not be found in the answer table.

"dial_ctl_: now dialing <channel> to <telephone number> for <user>.<project>."

This message is printed when the Initializer begins to dial an auto call line for a user. Note that this message can be used to determine responsibility for calls placed by the Auto Call Facility for such purposes as billing for long distance time used.

"dial_ctl_\$finish_dial_out: can't find process <process id>"

This message appears if the Initializer has initiated an auto call dial for a user and is then unable to locate the appropriate process once the dialing has been completed.

"dial_ctl_: <channel> dialed out for <user>.<project>"

This message is outputted when a successful dial out has occurred and the Initializer is giving the line to the user

process.

"dial_ctl: <error message> getting dial_out_status or tty info (<channel> for <user>.<project>)."

This message results when an error is encountered attempting to get the status of an auto call channel. <error message> is the ASCII string associated with an error_table_ code which details the error which occurred.

"dial_ctl: <error message> dial_out (<channel> for <user>.<project>) failed."

This message results from any other error returned to dial_ctl_ while attempting to complete a dial_out request.

"dial_ctl: user requested dial_out term: <channel>, <user>.<project>"

This message occurs when a user requests that the auto call connection be terminated normally.

New tty_Control Operations

The following are additions to the Control Operation section of the tty_ wakeup in the MPM (AG93), pages 3-12, 3-13, and 3-14.

dial_out initiates the dialing of a user specified phone number. This operation is allowed only for the process that originally attached the device (generally the initializer process). The info_ptr must point to the following structure:

```
    dcl 1 info_structure aligned,  
        2 num_chars fixed bin,  
        2 phone_number char(n);
```

where n is greater than or equal to info_structure.num_chars. The phone number dialed will be the numeric characters in the first (leftmost) num_chars characters of phone_number (non-numeric characters are ignored). Up to an 12 digit phone number may be dialed. Upon Completion of the dial out process, a wakeup is issued on the event channel associated with the I/O switch. The dial_out operation is considered active until either status is obtained using the dial_out_status order (see below) or the io switch is closed and detached. The I/O system status codes this order may return are error_table_\$big_arg or error_table_\$small_arg if the phone number to be dialed contains too many, too few digits respectively, error_table_\$undefined_order_request if the channel is already dialed up, or error_table_\$io_no_permission if the caller lacks needed permission.

dial_out_status

is used to obtain the status of an active dial_out operation. Valid status can only be obtained once (after valid status has been returned the dial_out operation is no longer active). Note that during the dial out process this operation may return an indicator that status is not yet available. Until status is available the dial out process has not completed. The process initiating the dial out can wait for a wakeup on the event channel for the tty channel before issuing the dial_out_status order. The info_ptr should be a null pointer. The possible values of code are one of the following I/O system codes:

```
error_table_$device_not_usable  
    No telephone auto-call unit attached to channel  
    (or no power to unit). This implies a physical  
    difficulty.
```

```
error_table_$invalid_state  
    Telephone channel is already being used.
```

New tty_ Control Operations

error_table_\$no_connection

The phone call was not completed.

error_table_\$no_operation

The channel can not support dial out. This implies a logical problem with the attempted dial out.

error_table_\$no_wired_structure

The dial out was successful but Multics was unable to provide the needed wired storage to support use of the channel. The phone line has been hungup. Another attempt may be successful.

event_info

is used to obtain the event channel associated with a tty switch. It is the same event channel returned by the "read_status" and "write_status" orders but this order call can be used even when the channel associated with the I/O switch is hungup. The info_ptr must be to a fixed binary (71) variable.

TITLE: AST lock metering		STATUS	DATE	
AUTHOR: R. Mullen		TVV	Written 2 Mar, 76	
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____	Category (Check One)		Status A 3/9/76	
		<input type="checkbox"/> Lib. Maint. Tools	Expires 9/9/76	
		<input checked="" type="checkbox"/> Sys. Anal. Tools	DOCUMENTATION CHANGES	
		<input type="checkbox"/> Sys. Prog. Tools	Document	Specify One or More
		355		
	<input type="checkbox"/> BOS	MPM (Vol, Sect.)		
	<input type="checkbox"/> Salvager	PLMS (AN #)		
	<input checked="" type="checkbox"/> Ring Zero	MOSN (Sect.)		
	<input type="checkbox"/> Ring One	MPAM (Sect.)		
	<input type="checkbox"/> SysDaemon/Admin.	MSAM (Sect.)		
	<input type="checkbox"/> Runtime			
	<input type="checkbox"/> User Cmmnd/Subr.			
Objections/Comments:		Info Segs		
		Other (Name)		
		None (Reason)		

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

Summary:

lock\$lock_ast to keep the following meters in the SST to be printed by fsm:
 total time the AST is locked
 total time spent waiting for AST to unlock
 total number of AST lockings

Reasons:

The AST lock is suspected of being a bottleneck in NSS. If $\frac{\text{locked_time}}{\text{real_time}} > .3$ it will imply a potential problem. If $\frac{\text{wait_time}}{\text{real_time}} > .5$ it will imply that Multiprogramming depth is increased by a small but intolerable amount.

Implications:

- AST lock interference may be reduced if need be by:
- A. Using per aste or per AST pool locks
 - B. Changing backup to dump according to dtem
 - C. Growing AST or critical ASTEPOOLS.
 - D. Unlocking AST during certain hardcore io operations.
 - E. Reconfiguration of ASTE pools by salvager
 - F. Attempt to not deactivate segs with pages on BulkStore
 - G. Backup to call hphcs_\$deactivate_seg.

TITLE: Command To Set Sons Volume		STATUS	DATE
AUTHOR: Van Vleck		Written	3/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Status	A 30/76
		Expires	9/9/76
-Planned for System MR 4.0		DOCUMENTATION CHANGES	
-Fixes Bug Number(s) _____		Document	Specify One or More
-Documented in MTB _____		355	
-User/Operations-visible		BOS	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		Salvager	MPM (Vol, Sect.)
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		Ring Zero	PLMS (AN #) 61
-Performance: <input checked="" type="checkbox"/> Better <input type="checkbox"/> Same		Ring One	MOSN (Sect.)
<input type="checkbox"/> Worse		<input checked="" type="checkbox"/> SysDaemon/Admin.	MPAM (Sect.)
-Replaces MCR _____		Runtime	MSAM (Sect.)
		User Cmd/Subr.	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY: Install highly privileged command to set the sons volume for a directory. The master-dir switch is not affected.

REASONS: In order to improve MIT performance it was necessary to set the sons volume of >pdd to the public hierarchy volume. The entry hphcs_\$set sons lvid already exists for the dumper; this change just supplies a command interface.

IMPLICATIONS: This command should only be used on >pdd or in situations where setquota is appropriate, since it falsifies master directory control records and quotas.

Name: set_sons_volume

This highly privileged command sets the sons_lvid for a directory, if the user has modify on the directory and its parent, and the directory is empty. The master directory switch is not affected.

Usage: set_sons_volume path lvolume
where path is the directory to be modified
and lvolume is the logical volume

Notes: The normal use of this command is in the system_start-up.ec, to set the sons volume of >pdd to a value other than "root". This command should only be used on >pdd or in situations where setquota is appropriate, since it falsifies master directory control records and quotas.

TITLE: Add &rf and &qf features to do		STATUS	DATE
AUTHOR: Steve Herbst		Written	2/27/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____	Category (Check One)		Status
	<input type="checkbox"/> Lib. Maint. Tools	Expires	A 3/9/76
	<input type="checkbox"/> Sys. Anal. Tools	DOCUMENTATION CHANGES	
	<input type="checkbox"/> Sys. Prog. Tools	Document	Specify One or More
	<input type="checkbox"/> 355	MPM (Vol, Sect.)	AG92
	<input type="checkbox"/> BOS	PLMS (AN #)	
	<input type="checkbox"/> Salvager	MOSN (Sect.)	
	<input type="checkbox"/> Ring Zero	MPAM (Sect.)	
	<input type="checkbox"/> Ring One	MSAM (Sect.)	
	<input type="checkbox"/> SysDaemon/Admin.	Info Segs	
<input type="checkbox"/> Runtime	Other (Name)		
<input checked="" type="checkbox"/> User Cmmnd/Subr.	None (Reason)		
Objections/Comments:			

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

SUMMARY: Add the &rf and &qf features to the do command, as natural complements to the recently installed &f feature. The string &rn is replaced by a list of the nth through last arguments, requoted. The string &qfn is replaced by a list of the nth through last arguments with quotes doubled.

Modes

The string `&qi` is replaced by a list of the *i*th through last arguments with quotes doubled. The string `&rfi` is replaced by a list of the *i*th through last arguments, requoted.

The `do` command has three modes, the long/brief mode, the nogo/go mode, and the absentee/interactive mode. These modes are kept in internal static storage and are thus remembered from call to call within a process. The modes are set by invoking the `do` command as follows:

`do mode`

where `mode` is one of the following: `-long (-lg)`, `-brief (-bf)`, `-nogo`, `-go`, `-absentee`, or `-interactive`.

If the long/brief mode is long, then the expanded command line is printed on `error_output` before it is executed or passed back. If the long/brief mode is brief, then the command line is not printed. The default for this mode is brief.

If the nogo/go mode is nogo, then the expanded command line is not passed to the command processor for execution. If the nogo/go mode is go, then the expanded line is passed to the command processor (if the `do` command was invoked as a command). If `do` is invoked as an active function, then the nogo/go mode is ignored. The default for this mode is go.

If the absentee/interactive mode is absentee, then the `do` command establishes an on unit for the `any_other` condition during the execution of the expanded command line. This is mainly of use in an absentee environment, in which any invocation of the default `any_other` on unit terminates the process. In the absentee mode, any signal caught by `do` merely terminates execution of the command line, not the process. A number of conditions, however, are not handled by `do` but are passed on for their standard Multics treatment; they are `cput`, `alm`, `quit`, `program_interrupt`, `command_error`, `command_query_error`, `command_question`, and `record_quota_overflow`. (For a description of these conditions see "List of System Conditions and Default Handlers" in Section VI of the MPM Reference Guide.) If the absentee/interactive mode is interactive, then `do` does not catch any signals. The default for this mode is interactive.

Quote-Doubling and Requoting

In addition to the parameter designators `&1 ... &9`, the `do` command also recognizes two more sets of parameter designators. They are `&q1 ... &q9`, to request quote-doubling in the actual argument as it is substituted into the expanded command line, and `&r1 ... &r9`, to request that the actual argument be requoted as well as have its quotes doubled during substitution.

Quote-doubling can be described as follows. Each parameter designator in the `command_string` to be expanded is found nested a certain level deep in quotes. If a designator is found to not be within quotes, then its quote-level is zero; if it is found between a single pair of quotes, then its quote-level is one; and so on. If the parameter designator `&qi` is found nested to quote-level *L*, then, as `control_argi` is substituted into the expanded command line each quote character found in `control_argi` is replaced by `2**L` quote characters during insertion. This permits the quote character to survive the

TITLE: Fix entryname bug in exec_com		STATUS	DATE
AUTHOR: Steve Herbst		Written	3/1/76
		Status	A 03/09/76
		Expires	9/9/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input type="checkbox"/> Sys. Prog. Tools <input type="checkbox"/> 355 <input type="checkbox"/> BOS <input type="checkbox"/> Salvager <input type="checkbox"/> Ring Zero <input type="checkbox"/> Ring One <input type="checkbox"/> SysDaemon/Admin. <input type="checkbox"/> Runtime <input checked="" type="checkbox"/> User Cmnd/Subr.	
		DOCUMENTATION CHANGES	
Objections/Comments:		Document	Specify One or More
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	Doc ok

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

SUMMARY: Fix bug in exec_com that causes the error message "Entry name too long" when the name of an exec_com is 29 characters long & does not end in .ec.

TITLE: Fix &attach bug in abs_io_		STATUS	DATE
AUTHOR: Steve Herbst		Written	3/1/76
SHW		Status	A 02/07/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Expires	9/9/76
-Planned for System MR 4.0		DOCUMENTATION CHANGES	
-Fixes Bug Number(s) X		Document Specify One or More	
-Documented in MTB		MPM (Vol, Sect.)	
-User/Operations-visible		PLMS (AN #)	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		MOSN (Sect.)	
-Incompatible change? <input type="checkbox"/> yes <input type="checkbox"/> no		MPAM (Sect.)	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same		MSAM (Sect.)	
<input type="checkbox"/> Worse		Info Segs	
-Replaces MCR		Other (Name)	
Objections/Comments:		None (Reason) doc OK	

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

SUMMARY: Fix bug in abs_io_ that causes user_input to remain attached to an exec_com (by &attach) after &quit if &detach is not specified before the &quit.

REASON: &quit should restore the old attachment.

Ver. 3	MULTICS CHANGE REQUEST	MCR 1692
741022		
TITLE: Implement new faster tty_read		STATUS DATE
AUTHOR: Robert S. Coren		Written 02/27/76
	JWG	Status A 3/9/76
		Expires 08/27/76
Planned for System: MR 4.0		
Fixes Bug Number(s): not applicable		CATEGORY (check one)
Documented in MTB: 262		() Lib. Maint. Tools
Incompatible Change: no		() Sys. Anal. Tools
User/Operations-visible Interface Change: no		() Sys. Prog. Tools
Coded in: (X) PL/I () ALM () Other-see below		() 355
Performance: (X) better () same () worse		() BOS
		() Salvager
DOCUMENTATION CHANGES (specify one or more)		(X) Ring Zero
MPM (vol,sect) Refguide,3 MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#) AN85		() Runtime
Info Segs tty_changes, canonicalization		() User Command/Subr
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY: Install new version of tty_read described in MTB262, which also implements canonicalization as described in MTB 251.

REASONS: Part of general upgrade of ring-zero typewriter DIM. Improves efficiency of input processing and makes it more consistent; allows for future addition of user-substitutable translation tables and special characters.

IMPLICATIONS: Canonicalization will be consistent, but may not always produce identical results to old software (see MTB 251). "Invisible" control characters will be discarded in "can" and/or "erkl" mode.

DETAILED PROPOSAL: See MTB 262.

tty_changes.info:

A new version of the ring-zero tty DIM will be installed soon which incorporates the following changes:

1. Control characters which do not change the carriage or paper position (i.e., any control character other than backspace, horizontal tab, newline, carriage return, form-feed, vertical tab, or space), cannot be input except as octal escapes if the terminal is in either "can" or "erkl" mode.
2. A lot of bugs are eliminated from canonicalization of input, and well-established rules are established for combinations of overstrikes, erase/kill characters, and escape sequences. Please type "help canonicalization" or dprint >doc>info>canonicalization.info for details.

canonicalization.info will be a non-standard info segment suitable for dprinting, which will contain the list of rules presented in MTB 251, and possibly some of the examples. A slightly revised version of MTB 251 will also be appended to the discussion of canonical form in Section 3 of the MPM Reference Guide.

TITLE: Fix bugs in system control		STATUS	DATE
AUTHOR: Paul A. Green TVV		Written	02-25-76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 3.1 -Fixes Bug Number(s) <u>MRF 9260</u> -Documented in MTB -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR		Status	A 319/70
		Expires	9/1/76
		DOCUMENTATION CHANGES	
		Document	Specify One or More
		MPM (Vol, Sect.)	
		PLMS (AN #)	AN 66
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY: 1) Fix test mode to work.

2) Fix input-wait timeout to work.

Reasons: Both features stopped working in the new system control.

Implications: None

Detailed Proposal: Replace sc_init.pll, system_control.pll,
mc_tty.pll

Multics Change Request

TITLE: Fix bugs in mrd_util_		STATUS	DATE
AUTHOR: Paul A. Green TVV		Written	02-25-76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> AIM <input type="checkbox"/> other- explain in DETAILED PROPOCAL		Status	A 3/9/76
-Planned for System MR 3.1		Expires	9/9/76
-Fixes Bug Number(s) MPR 9221		DOCUMENTATION CHANGES	
-Documented in MTB		Document	Specify One or More
-User/Operations-visible		MPM (Vol, Sect.)	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		PLMS (AN #)	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		MOSN (Sect.)	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same		MPAM (Sect.)	
<input type="checkbox"/> Worse		MSAM (Sect.)	
-Replaces MCR		Info Segs	
Objections/Comments:		Other (Name)	
		None (Reason)	

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

SUMMARY: Fix threading errors in mrd_util_ that have caused intermittent message coordinator problems at many sites.

Reasons: Better reliability, fewer crashes. Slightly better error messages

Implications: None

Detailed Proposal: Replace mrd_util_.pll and syscon_mseg.incl.pll

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1696</u>
TITLE: Fix bug in print_syserr_log		STATUS
AUTHOR: Larry Johnson		DATE
	THVV	Written 02/25/76
Planned for System: 4.0		Status A 30/76
Fixes Bug Number(s): not applicable		Expires 06/25/76
Documented in MTB: not applicable		CATEGORY (check one)
Incompatible Change: no		() Lib. Maint. Tools
User/Operations-visible Interface Change: no		(<input checked="" type="checkbox"/>) Sys. Anal. Tools
Coded in: (<input checked="" type="checkbox"/>)PL/I ()ALM ()other-see below		() Sys. Prog. Tools
Performance: ()better (<input checked="" type="checkbox"/>)same ()worse		() 355
DOCUMENTATION CHANGES (specify one or more)		() BOS
MPM (vol,sect)	MPAM (sect)	() Salvager
MOSN (sect)	MSAM (sect)	() Ring Zero
PLMs (AN#)		() Ring One
Info Segs		() SysDaemon/Admin
Other		() Runtime
None (reason)		() User Command/Subr
OBJECTIONS/COMMENTS:		

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

Summary

The -exclude and -match control arguments do not work properly if the syserr message is an "=".

Detailed Proposal

Change the control arguments to check the expanded text.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR 1697
TITLE: Fix bug in get_process_id_		STATUS DATE
AUTHOR: Larry Johnson	SHW	Written 02/25/76
		Status 03/19/76
		Expires 08/25/76
Planned for System: 4.0		CATEGORY (check one)
Fixes Bug Number(s): not applicable		() Lib. Maint. Tools
Documented in MTB: not applicable		() Sys. Anal. Tools
Incompatible Change: no		() Sys. Prog. Tools
User/Operations-visible Interface Change: no		() 355
Coded in: <input checked="" type="checkbox"/> PL/I () ALM () other-see below		() BOS
Performance: <input checked="" type="checkbox"/> better () same () worse		() Salvager
DOCUMENTATION CHANGES (specify one or more)		() Ring Zero
MPM (vol,sect) MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#)		() Runtime
Info Segs		<input checked="" type="checkbox"/> User Command/Subr
Other		
None (reason)		
OBJECTIONS/COMMENTS:		

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

Summary

The module get_process_id_ (which contains get_group_id_, get_lock_id_, get_pdir_, get_authorization_) keeps the values to be returned by these entries in internal static. All of these values are initialized with the first call to any of the entry points. The initialization check is incorrect, however, so that all of the values are initialized with every call to any of the entries. This results in two extra hcs_ calls every time this subroutine is called.

Detailed Proposal

Fix the initialization check so that all values are initialized once per process (per ring).

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR 1698
TITLE: fix miscellaneous bugs in tape_ansi_		STATUS DATE
AUTHOR: Ross E. Klinger TVV		Written 02/26/76
Planned for System: 4.0		Status 03/09/76
Fixes Bug Number(s): not applicable		Expires 08/26/76
Documented in MTB: not applicable		CATEGORY (check one)
Incompatible Change: no		() Lib. Maint. Tools
User/Operations-visible Interface Change: no		() Sys. Anal. Tools
Coded in: (X) PL/I () ALM () other-see below		() Sys. Prog. Tools
Performance: () better (X) same () worse		() 355
DOCUMENTATION CHANGES (specify one or more)		() BOS
MPM (vol,sect)	MPAM (sect)	() Salvager
MOSN (sect)	MSAM (sect)	() Ring Zero
PLMs (AN#) 57		() Ring One
Info Segs		() SysDaemon/Admin
Other		() Runtime
OBJECTIONS/COMMENTS:		(X) User Command/Subr

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

SUMMARY: while writing the tape_ansi_PLM, a number of minor bugs have been discovered, none of which have ever been reported. The majority of these bugs will never be encountered except under highly unlikely combinations of errors; nevertheless, they should be fixed.

REASONS: Increased reliability

IMPLICATIONS: Improved reliability



Name: list_mst

list_mst is used to find out what segments are on a Multics System tape (either a Multics or BOS bootload tape).

Usage: list_mst tapeid name-1 name-2 ... name-n

WHERE tapeid is the tape reel identifier of the Multics System Tape to be listed. name-1 through name-n are names of segments to be listed if they appear on the tape. Note that they may contain stars (see "match_star_name_" in the MPM), and the default is "***" (list everything) if no names are given.

A summary line giving the length of each segment, and its primary name is printed out for each segment listed. A special name is printed out for segments written with the mst generator "first_name" keyword, for the proper name of such segments does not appear on the tape.

Name: excerpt_mst, xmst

excerpt_mst is used to excerpt given segments from a Multics System Tape (either a Multics or BOS bootload tape).

Usage: excerpt_mst tapeid name-1 name-2 ... name-n

WHERE tapeid is the tape reel identifier of the Multics System Tape to be excerpted. name-1 through name-n are the names of the specific segments to be extracted. These names may contain stars (see "match_star_name_" in the MPM), and the default is "***" (divulge entire tape) if no names are given. If a given segment has a separate linkage segment on the tape, and has been extracted, the separate linkage segment will be extracted as well. Segments extracted are created in the current working directory. Bit counts are set from the SLT entry on the tape, as opposed to the actual length of the segment on the tape.

An informational message is printed whenever a segment is extracted.

Name: write_mst

write_mst is used to write short Multics System Tapes. This is usually useful only for writing tapes for BOS, either complete BOS tapes, or tapes to be used with the BOS LOADDM command (see the Multics Operators Handbook).

Usage: write_mst tapeid ~~names~~and-control-args-

WHERE tapeid is the tape reel identifier of the ~~M~~ tape to be written. ~~names~~and-control-args- are the names of segments to be written, and control arguments specifying collection marks and optional text/link decoding. The arguments may be any intermixture of

- 1) Segment specifications
- and 2) Collection mark specifications.

A collection mark specification is the control argument "-col" or "-collection". It causes sequential collection marks to be written every time it appears in the command line.

A segment specification is the (relative pathname of a segment to be written on the tape, possibly preceded by the optional control arguments "-text", "-link", ~~xxx~~and/or "-stop" (short names "-tx", "-lk", and "-sp"). The segment is written with the entry name supplied as its name on the tape. Both its bit count/current length and actual length are derived from its bit count in the storage system. Other than the bit count and current length, its SLT entry is given as zero.

For each segment, if the "-link" argument is supplied, a separate segment, consisting of the separated linkage of this segment, is written on the tape, with the name <name>.link.

If the "-text" argument is supplied, only the text of the segment is written, as opposed to the whole segment. This, as "-link", requires that the segment specified be a standard object segment. Note that "-text" and ~~xxx~~"-link" neither contradict nor interact with each other.

If the "-stop" argument is supplied, a call is ~~made~~ made to debug immediately before the segment is written out, so that the user may modify the SLT entry in any desired way. A message giving the location of the segment and the SLT entry is printed out before such a call is made.

TITLE: Add per-process segfault counter to system		STATUS	DATE
AUTHOR: R. Bratt		SHW	
-Coded in: <input type="checkbox"/> PL/I <input checked="" type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Written	6 February 76
-Planned for System MR _____		Status	P2/12/76 A 3/2/76
-Fixes Bug Number(s) _____		Expires	9/2/76
-Documented in MTB _____		DOCUMENTATION CHANGES	
-User/Operations-visible		Document	Specify One or More
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		MPM (Vol, Sect.)	Subroutines
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		PLMS (AN #)	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same		MOSN (Sect.)	
<input type="checkbox"/> Worse		MPAM (Sect.)	
-Replaces MCR _____		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY:

Add per-process segfault counter to pds. Add new entry to hcs_\$get_process_usage return structure to return segfault count.

Note: This is compatible since the return structure already includes an entry specifying the number of items to return.

Also, add per process vtoc read and write meters.

REASON:

This may prove to be a useful meter. It costs very little to provide.

Subroutine Call
4/30/73Name: hcs_\$get_process_usage

This subroutine returns information about a process's usage of Multics since it was created. It provides data about processor and memory usage.

Usage

```
declare hcs_$get_process_usage entry (ptr, fixed bin(35));
call hcs_$get_process_usage (info_pointer, code);
```

- 1) info_pointer is a pointer to the structure in which process information is returned (see Notes below). (Input)
- 2) code is a standard status code. (Output)

Notes

The format of the structure based on info_pointer is:

```
declare 1 process_usage,
        2 number_wanted fixed bin,
        2 cpu_time_used fixed bin(71),
        2 memory_usage fixed bin(71),
        2 number_of_page_faults fixed bin(35),
        2 amount_of_prepaging fixed bin(35),
        2 process_virtual_time fixed bin(71);
```

- 1) number_wanted is set by the calling program to specify the number of other entries in the structure to be filled in. The entry itself (the numbers wanted) is not included in this count. The value *9* would cause *9* entries listed below to be filled in. A smaller number, *n*, will cause the first *n* entries to be filled in. (Input)
- 2) cpu_time_used is set to the amount of processor time (in microseconds) used by the calling process. (Output)

```
2 number-of-segment-faults fixed bin(35),
2 number-of-bounds-faults fixed bin(35),
2 number-of-vtoc-reads fixed bin,
2 number-of-vtoc-writes fixed bin;
```

- 3) `memory_usage` is a measure of the primary (core) memory used by this process. The units of memory usage are page-seconds, normalized to account for the size of primary memory actually in use. (Output)
- 4) `number_of_page_faults` is set to the number of demand page faults this process has taken. (Output)
- 5) `amount_of_prepaging` is the number of pages prepaged for this process. (Output)
- 6) `process_virtual_time` is the amount of processor time (in microseconds) used exclusive of page fault and system interrupt processing time. (Output)
- 7) *number-of-segment-faults* is set to the number of segment faults this process has taken. (output)
- 8) *number-of-bands-faults* is set to the number of bands faults this process has taken. (output)
- 9) *number-of-vtoc-reads* is set to the number of VTOC read io operations the process has done (output)
- 10) *number-of-vtoc-writes* is set to the number of write VTOC io operations the process has done (output)

TITLE: teco extensions		STATUS	DATE	
AUTHOR: R. Bratt SHW		Written	2-10-76	
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR _____ -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____	Category (Check One)		Status	
	Lib. Maint. Tools		Expires	
	Sys. Anal. Tools		DOCUMENTATION CHANGES	
	Sys. Prog. Tools		Document	Specify One or More
	355		MPM (Vol, Sect.)	
BOS		PLMS (AN #)	Tools	
Salvager		MOSN (Sect.)		
Ring Zero		MPAM (Sect.)		
Ring One		MSAM (Sect.)		
SysDaemon/Admin.				
Runtime				
<input checked="" type="checkbox"/> User Cmmnd/Subr.				

Objections/Comments:

Info Segs
Other (Name)
None (Reason)

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

SUMMARY:

Provide new entry point, teco\$macro, which allows the initial macro executed to be changed. (For compatibility reasons we must use a new entry point rather than a command line argument.)

REASONS:

teco's inability to start anything but a user-oriented editing session greatly reduces its usefulness. The addition of a mechanism for specifying the initial command buffer to be executed allows teco to be called from command level to perform specific tasks, e.g., abbrev editing.

1. Loops cannot cross macro boundaries, i.e. a loop cannot start in one macro and end in another. This does not, however, prohibit the M command from being used within a loop.
2. A macro can modify itself if it is in a Q-register. Note, however, that the current invocation of the macro is not affected; only future accesses to the Q-register. If the macro is invoked by the EM command, the results of modifying the file are hard to predict: TECO reads the command string directly from the file.
3. When a macro is invoked by the EM command, it should be noted that the name of the macro is found in the Q-register named ". Thus several macros can be put in one segment with the first command in the segment being OQ". (Don't forget to put all the appropriate names on the segment).
4. If an M or EM command is given as the last command in one macro, the command is interpreted as a goto rather than a call. Thus, unlimited M's can be done in this manner although there is an implementation defined limit to the depth of calls.
5. When the TECO command is entered, a macro named start_up is searched for. If it is found, the arguments to TECO are put onto the pushdown stack and the start_up macro is executed. If no start_up macro is found, the string EI/filename/J is executed, where filename is the first argument to TECO. At the present time, there is a start_up macro in the TECO library. When the start_up macro is called, the first thing on the pushdown list is the number of arguments TECO was called with. The remaining items in the list are the actual string arguments to TECO going from left to right on the command line.
6. If TECO\$macro is invoked as a command, then TECO behaves as described above except that the first argument is not placed in the pushdown list. Instead, TECO looks for a macro named by the first argument and executes it in place of the usual start_up macro. If an error is encountered before TECO command level is reached (and the macro need not ever allow TECO to reach command level), then TECO simply returns.

CODING CONVENTIONS FOR MACROS

Since there are only a small number of Q-registers (95), each with a one-character name, there are serious problems in writing a set of macros that are compatible. A set of macros become incompatible if one macro uses a Q-register for long-term storage that any other macro uses at all. There are two ways this effect can be combatted. First, by establishing certain coding conventions, and second, by use of a documented macro library. Probably the most important coding convention is the specification of which Q-registers can be used inside a macro for

TITLE: COBOL Bug Fixing		STATUS	DATE
AUTHOR: William K. O'Neill (CEO-B) CCZ		Written	February 18, 1976
-Coded in: <input checked="" type="checkbox"/> PL/I <input checked="" type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 3.1 -Fixes Bug Number(s) <u>43,150,162</u> -Documented in MTB -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR		Status	A 318176
		Expires	<u>AUG 18, 1976</u>
		DOCUMENTATION CHANGES	
		Document	Specify One or More
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

Summary

- (a) Bug 143: A READ into a file that has an occurs may store the data in the wrong location. It is difficult to program around this problem.
- (b) Bug 150: The compiler will abort if the number of data names, plus procedure names, plus section names exceeds approximately 800. Many large programs will not compile.
- (c) Bug 162: The compiler will abort if bad syntax is encountered in the organization clause in the environment division. This is a frequent error for a new user of Multics COBOL to make.

TITLE: Add information-only entry points to		STATUS	DATE
AUTHOR: dir_control_error Lee Scheffler		Written	760218
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Status	A 3/2/76
-Planned for System MR 4.0		Expires	9/2/76
-Fixes Bug Number(s) unreported		DOCUMENTATION CHANGES	
-Documented in MTB		Document	Specify One or More
-User/Operations-visible		MPM (Vol, Sect.)	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		PLMS (AN #) AN75	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		MOSN (Sect.)	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same		MPAM (Sect.)	
<input type="checkbox"/> Worse		MSAM (Sect.)	
-Replaces MCR		Info Segs	
Objections/Comments:		Other (Name)	
		None (Reason)	

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

Summary: Add two new entries, dir_control_error\$contents_info and dir_control_error\$attributes_info. These entries will operate exactly as do the current dir_control_error\$contents and dir_control_error\$attributes, but will not generate auditing messages.

Several places in hardcore directory control (both oss and nss) call the contents and attributes entries solely to determine whether or not they may return a detailed error code to their callers. Change these calls (in append, del_dir_tree, find_entry, initiate and truncate) to call the contents_info and attributes_info entry points.

Reasons: Bug fix required for Air Force acceptance of AIM.

Eliminate 90% of the (erroneous) auditing messages in the syserr log, particularly the message generated each time hcs_\$make_seg is called on a segment that already exists.

Implications: There will be a better distinction between the two different uses of dir_control_error, namely, to determine the proper error code to return on a real error, and determining whether the user has sufficient access to get a detailed status code.

TITLE: Fix erroneous audit calls from set_privileges		STATUS	DATE
AUTHOR: Lee Scheffler		Written	760213
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) unreported -Documented in MTB -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR		Status	A 3/2/76
		Expires	9/2/76
		DOCUMENTATION CHANGES	
		Category (Check One)	
	Lib. Maint. Tools	Document	Specify One or More
	Sys. Anal. Tools		
	Sys. Prog. Tools		
	355		
	BOS		
	Salvager	MPM (Vol, Sect.)	
	X Ring Zero	PLMS (AN #)	AN 75
	Ring One	MOSN (Sect.)	
	SysDaemon/Admin.	MPAM (Sect.)	
	Runtime	MSAM (Sect.)	
	User Cmnd/Subr.		
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

Summary: Change set_privileges to call protection_audit only when there is an actual change to the system_privileges of the calling process. (It currently calls protection_audit on every call, whether or not an actual change to privileges occurs.)

- Reasons:
1. Bug fix required for Air Force acceptance of AIM.
 2. Remove excessive audit messages from the syserr log.

Implications: None

TITLE: Add check for valid entry pointer to protection_audit_\$access_denied		STATUS	DATE
AUTHOR: Lee Scheffler		Written	760218
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) unreported -Documented in MTB -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____	Category (Check One)		
	<input type="checkbox"/> Lib. Maint. Tools	DOCUMENTATION CHANGES	
	<input type="checkbox"/> Sys. Anal. Tools	Document	Specify One or More
	<input type="checkbox"/> Sys. Prog. Tools	MPM (Vol, Sect.)	
	<input type="checkbox"/> 355	PLMS (AN #)	AN75
<input type="checkbox"/> BOS	MOSN (Sect.)		
<input type="checkbox"/> Salvager	MPAM (Sect.)		
<input checked="" type="checkbox"/> Ring Zero	MSAM (Sect.)		
<input type="checkbox"/> Ring One	Info Segs		
<input type="checkbox"/> SysDaemon/Admin.	Other (Name)		
<input type="checkbox"/> Runtime	None (Reason)		
<input type="checkbox"/> User Cmmd/Subr.			
Objections/Comments:			

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

Summary: Change protection_audit_\$access_denied to check that its entry pointer parameter is a valid entry pointer (offset ^ = 0) before blindly charging ahead picking up attributes out of the branch pointed to.

- Reasons:
1. Bug fix required for Air Force acceptance of AIM.
 2. Auditing messages for cases where dir_control_error is passed a directory pointer (offset = 0) instead of an entry pointer contain garbage.

TITLE: Change convert authorization to properly convert null access class strings AUTHOR: L. Scheffler		STATUS Written Status Expires	DATE 760218 A 3/2/76 9/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> AIM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) 80552 -Documented in MTB -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR	Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input type="checkbox"/> Sys. Prog. Tools <input type="checkbox"/> 355 <input type="checkbox"/> BOS <input type="checkbox"/> Salvager <input type="checkbox"/> Ring Zero <input type="checkbox"/> Ring One <input type="checkbox"/> SysDaemon/Admin. <input checked="" type="checkbox"/> Runtime <input checked="" type="checkbox"/> User Cmnd/Subr.	DOCUMENTATION CHANGES Document Specify One or More MPM (Vol, Sect.) PLMS (AN #) AN75 MOSN (Sect.) MPAM (Sect.) MSAM (Sect.) Info Segs Other (Name) None (Reason)	
Objections/Comments:			

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

Summary: Change convert authorization \$from string to reject a null access class string if the name for level 0 has not been set to null.

Reasons:

1. Bug fix required for Air Force acceptance of AIM.
2. For compatibility with sites not using AIM, a convention was established that a null access class character string would correspond to "system_low", defined to be level 0, no categories. This should only be true if level 0 has a null name. If a site chooses a non-null name for level 0, a null access class string should be rejected as invalid.

Implications: Greater consistency

TITLE: Add audit message for illegal returns		STATUS	DATE
AUTHOR: Lee Scheffler		Written	760218
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Status	A 3/2/76
		Expires	9/2/76
		DOCUMENTATION CHANGES	
Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input type="checkbox"/> Sys. Prog. Tools <input type="checkbox"/> 355 <input type="checkbox"/> BOS <input checked="" type="checkbox"/> Ring Zero <input type="checkbox"/> Ring One <input type="checkbox"/> SysDaemon/Admin. <input type="checkbox"/> Runtime <input type="checkbox"/> User Cmmd/Subr.		Document	Specify One or More
Objections/Comments:		MPM (Vol, Sect.)	
		PLMS (AN #)	AN75
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

Summary: Change restart_fault to call protection_audit if an attempt is made to restart with a set of invalid machine conditions.

Reasons: This is an auditable protection event that has not previously been audited.

Implications: Auditing mechanism is more complete.

TITLE: Modify map355 to increase assembly limits		STATUS	DATE
AUTHOR: M. Grady		Written	2-25-76
		Status	A 3/2/76
		Expires	9/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input checked="" type="checkbox"/> Sys. Prog. Tools 355 BOS Salvager Ring Zero Ring One SysDaemon/Admin. Runtime User Cmnd/Subr.	
		DOCUMENTATION CHANGES	
Objections/Comments:		Document	Specify One or More
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY:

Modify the map355 command to produce a \$limit card for the GCOS assembly of 355map source which specifies 128k core limit for the assembly.

REASONS:

The latest changes to dia_man caused the memory usage during assembly to exceed the current 32k limit. This caused the assembler to bomb.

Multics Change Request

TITLE: Install FAST subsystem		STATUS	DATE
AUTHOR: S. Barr		Written	2-24-76
		Status	A 3/2/76
		Expires	9/2/76
<input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Category (Check One)	
-Planned for System MR 3.1		<input type="checkbox"/> Lib. Maint. Tools	
-Fixes Bug Number(s)		<input type="checkbox"/> Sys. Anal. Tools	
-Documented in MTB 248		<input type="checkbox"/> Sys. Prog. Tools	
-User/Operations-visible		<input type="checkbox"/> 355	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		<input type="checkbox"/> BOS	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		<input type="checkbox"/> Salvager	
-Performance: <input type="checkbox"/> Better <input type="checkbox"/> Same		<input type="checkbox"/> Ring Zero	
<input type="checkbox"/> Worse		<input type="checkbox"/> Ring One	
-Replaces MCR		<input type="checkbox"/> SysDaemon/Admin.	
		<input type="checkbox"/> Runtime	
		<input type="checkbox"/> User Cmmnd/Subr.	
		<input checked="" type="checkbox"/> subsystem	
Objections/Comments:		DOCUMENTATION CHANGES	
		Document Specify One or More	
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY:

- Install FAST as described in MTB 248 with two modifications.
1. If a path is given with the run command, it can specify a source segment or an object segment.
 2. Change the ready message from "r" to "r hmmm".

DETAILED PROPOSAL:

See attached description of the run command.

run

run

Name: run

The run command executes a BASIC or FORTRAN program. After execution it closes all input/output files and frees common blocks.

Usage

run {path}

where path is the pathname of an object segment. If path is not given, the run command compiles the temporary text and executes it. In order for the command to work, the entryname used with the last old, new, or save command must have a language suffix. If path is specified and the entryname has a language suffix, the run command expects the segment to contain the source program and compiles and executes it. If path is specified and the entryname does not contain a language suffix, the run command expects the segment to contain object code and executes it.

Example

```
! old test.basic  
r  
  
! run  
  (program execution)  
r  
  
! run std /object segment "std" in working directory  
  (program execution)  
r  
  
! run /temporary text is not changed by run  
  (program execution of test.basic)  
r
```

TITLE: Improve delete's error messages		STATUS	DATE
AUTHOR: S. Herbst		JWG	
<input checked="" type="checkbox"/> Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL <input type="checkbox"/> Planned for System MR <input type="checkbox"/> Fixes Bug Number(s) _____ <input type="checkbox"/> Documented in MTB _____ <input type="checkbox"/> User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no <input type="checkbox"/> Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no <input type="checkbox"/> Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse <input type="checkbox"/> Replaces MCR _____		Written	2-23-76
		Status	A 3/2/76
		Expires	9/2/76
		DOCUMENTATION CHANGES	
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	doc ok

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

SUMMARY:

Fix the delete command to include the entire pathname with its error messages.

REASONS:

Currently, only the entry name is printed. For example, if x does not exist, the command "delete xyy" produces the misleading error message:

"Some directory in pathname does not exist. y"

TITLE: Implement star convention in adjust_bit_count		STATUS	DATE
AUTHOR: S. Herbst		Written	2-23-76
		Status	A 3/2/76
		Expires	9/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Category (Check One)	
-Planned for System MR		Lib. Maint. Tools	
-Fixes Bug Number(s)		Sys. Anal. Tools	
-Documented in MTB		Sys. Prog. Tools	
-User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no		355	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		BOS	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse		Salvager	
-Replaces MCR		Ring Zero	
		Ring One	
		SysDaemon/Admin.	
		Runtime	
		X User Cmnd/Subr.	
Objections/Comments:		DOCUMENTATION CHANGES	
		Document Specify One or More	
		MPM (Vol, Sect.) AG92	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY:

Change the adjust_bit_count command to implement the star convention.

REASONS:

Now that bit counts are more important, this change makes it easier for users to maintain correct bit counts on their segments.

TITLE: Correct bug in detection of eof in ntape_		STATUS	DATE
AUTHOR: R. Bratt		Written	2-24-76
		Status	A 9/2/76
		Expires	9/2/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Category (Check One)	
-Planned for System MR		Lib. Maint. Tools	
-Fixes Bug Number(s)		Sys. Anal. Tools	
-Documented in MTB		Sys. Prog. Tools	
-User/Operations-visible		355	
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		BOS	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		Salvager	
-Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same		Ring Zero	
<input type="checkbox"/> Worse		Ring One	
-Replaces MCR		SysDaemon/Admin.	
		Runtime	
		X User Cmmd/Subr.	
Objections/Comments:		DOCUMENTATION CHANGES	
		Document Specify One or More	
		MPM (Vol, Sect.)	
		PLMS (AN #)	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason) fix bug	

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

SUMMARY:
Correct eof detection in ntape_ by changing incorrect constant.

Ver. 3 741022	MULTICS CHANGE REQUEST	JWG	MCR 1679
TITLE: Implement dump_fnp and patch_fnp Commands		STATUS	DATE
AUTHOR: Robert S. Coren		Written	02/25/76
		Status	03/2/76
		Expires	08/25/76
Planned for System: not applicable 4.0		CATEGORY (check one)	
Fixes Bug Number(s): not applicable		<input type="checkbox"/> Lib. Maint. Tools	
Documented in MTE: not applicable		<input checked="" type="checkbox"/> Sys. Anal. Tools	
Incompatible Change: no		<input type="checkbox"/> Sys. Prog. Tools	
User/Operations-visible Interface Change: no		<input type="checkbox"/> 355	
Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input checked="" type="checkbox"/> other-see below		<input type="checkbox"/> POS	
Performance: <input type="checkbox"/> better <input type="checkbox"/> same <input type="checkbox"/> worse		<input type="checkbox"/> Salvager	
DOCUMENTATION CHANGES (specify one or more)		<input type="checkbox"/> Ring Zero	
MPM (vol,sect)	MPAM (sect)	<input type="checkbox"/> Ring One	
MOSN (sect)	MSAM (sect)	<input type="checkbox"/> SysDaemon/Admin	
PLMs (AN#) AN85		<input type="checkbox"/> Runtime	
Info Segs		<input type="checkbox"/> User Command/Subr	
Other			
OBJECTIONS/COMMENTS:			

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

SUMMARY: 1. Implement command, dump_fnp, and control operation in ring-zero typewriter DIM, "dump_fnp", to allow a user to print the contents of specified locations of FNP memory on his/her terminal.

2. Implement command, patch_fnp, and corresponding control operation, to allow a user to modify specified locations in FNP memory from Multics command level.

REASONS: It has frequently been found to be useful, when investigating actual or suspected bugs in FNP software, to be able to examine the contents of FNP memory. The inability to do this without access to an FNP console has often been inconvenient. It is very occasionally useful to be able to modify the contents of a running FNP, either to test assumptions about or to correct known software bugs.

IMPLICATIONS: None.

DETAILED PROPOSAL: See attached command documentation. The two control operations will make use of a temp-wired buffer (using iobm) to pass the required data between ring zero and the FNP. Accordingly, a new segment, fnp_dump_seg, will be added to the system header for this purpose. The "dump_fnp" and "patch_fnp" operations will be privileged and highly privileged, respectively; new gate entries, phcs_\$tty_order and hphcs_\$tty_order, will be added. It is not proposed to make these two operations available through iox_\$control.

COMMAND DESCRIPTIONS

Name: dump_fnp

This privileged command enables a user to print the contents of specified locations in FNP memory on his/her terminal, in a format similar to that used by dump_segment.

Usage

```
dump_fnp fnp_id address nwords -control_arg-
```

fnp_id is the identifier of the FNP whose memory is to be dumped. It must be either "a", "b", "c", or "d".

address is an octal number indicating the starting address from which memory is to be dumped.

nwords is an octal number giving the number of 10-bit words that are to be dumped. It must not be greater than 4000(8).

control_arg may be -character (or -ch), and indicates, if present, that the contents of the specified memory **locations are** to be printed as ASCII characters as well as octal numbers. It may appear anywhere on the command line.

Name: patch_fnp

This highly-privileged command enables a user to modify specified contiguous locations in FNP memory.

Usage

patch_fnp fnp_id address values

fnp_id is the identifier of the FNP to be patched. It must be "a", "b", "c", or "d".

address is an octal number identifying the first location to be patched.

values are one or more octal values to replace the contents of consecutive words of FNP memory starting at the location specified by address; each value may be a maximum of 6 digits.

Note : The user is asked to verify the correctness of the patches before they are made; a record of the patches is entered in the syserr log and printed on the syserr console.

CONTROL OPERATIONS

dump_fnp is used to dump the contents of FNP memory.

Usage

```
declare phcs_$tty_order entry (fixed bin, char (*), ptr, fixed bin,  
                               fixed bin(35));
```

```
call phcs_$tty_order (fnp_no, "dump_fnp", info_ptr, ignore, code);
```

fnp_no is the number of the relevant FNP, such that $0 < \text{fnp_no} \leq 4$. (Input)

info_ptr points to the following structure: (Input)

```
dcl 1 dump_fnp_info aligned,  
    2 fnp_address fixed bin,  
    2 fnp_length fixed bin,  
    2 bufp ptr;
```

fnp_address is the address of the first word of FNP memory to be dumped.

fnp_length is the number of 18-bit words to be dumped.

bufp points to a buffer in which the dumped words are to be returned to the caller.

code is a standard system status code. (Output)

patch_fnp is used to modify specified locations of FNP memory.

Usage

```
declare hphcs_$tty_order entry (fixed bin, char (*), ptr, fixed bin,  
                                fixed bin(35));
```

```
call hphcs_$tty_order (fnp_no, "patch_fnp", info_ptr, ignore, code);
```

Arguments are as above; the buffer pointed to by patch_fnp_info.bufp contains the values to be patched into FNP memory.

TITLE: Change format of pl1_version_		STATUS	DATE
AUTHOR: R. A. Barnes		Written	2-23-76
-Coded in: <input type="checkbox"/> PL/I <input type="checkbox"/> AIM <input checked="" type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Status	A 3/2/76
		Expires	9/12/76
		DOCUMENTATION CHANGES	
		Category (Check One)	Document Specify One or More
	Lib. Maint. Tools	MPM (Vol, Sect.)	
	Sys. Anal. Tools	PLMS (AN #)	
	Sys. Prog. Tools	MOSN (Sect.)	
	355	MPAM (Sect.)	
	BOS	MSAM (Sect.)	
	Salvager	Info Segs	
	Ring Zero	Other (Name)	
	Ring One	None (Reason)	
	SysDaemon/Admin.		
	Runtime		
	X User Cmnd/Subr.		

Objections/Comments:

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

SUMMARY:

Change the format of pl1_version_ to include a release number.

REASONS:

pl1_version_ contains an ASCII string that the PL/I compiler places in each object segment to identify the compiler used to create the segment. This string, pointed at by std_symbol_header.gen_version, presently contains the date the compiler was installed, which differs at MIT and PCO. To uniquely identify a compiler, therefore, we need a release number.

DETAILED PROPOSAL:

A unique release number will be used with the installed compiler. Since we have been numbering the releases in our conversations with Toshiba, we will use that scheme in pl1_version_. pl1_version_ for MR4.0 might say, for example:

Multics PL/I Compiler, Release 20, of March 20, 1976.

TITLE: Fix bug in propagate		STATUS	DATE
AUTHOR: S. Herbst MDM		Written	2-23-76
<input checked="" type="checkbox"/> Coded in <input type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL <input type="checkbox"/> Planned for System MR <input type="checkbox"/> Fixes Bug Number(s) _____ <input type="checkbox"/> Documented in MTB _____ <input type="checkbox"/> User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no <input type="checkbox"/> Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no <input type="checkbox"/> Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse <input type="checkbox"/> Replaces MCR _____		Status	A 3/2/76
		Expires	9/2/76
		DOCUMENTATION CHANGES	
Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input type="checkbox"/> Sys. Prog. Tools <input type="checkbox"/> 355 <input type="checkbox"/> BOS <input type="checkbox"/> Salvager <input type="checkbox"/> Ring Zero <input type="checkbox"/> Ring One <input type="checkbox"/> SysDaemon/Admin. <input type="checkbox"/> Runtime <input checked="" type="checkbox"/> User Cmmnd/Subr.		Document	Specify One or More
Objections/Comments:		MPM (Vol, Sect.)	
		PLMS (AN #)	57
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

SUMMARY:

Fix bug in iox_\$propagate that causes a modes operation on a re-attached I/O switch to return error_table_\$not_attached.

REASONS:

In some cases, a modes operation is valid on an attached but closed switch. In others, the error code error_table_\$no_operation should be returned. Bug is caused by improper initialization of an attached iocb.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1682</u>
TITLE: Fix more problems in syserr		STATUS DATE
AUTHOR: Larry Johnson		Written 02/13/76
JWG		Status <u>3/2/76</u>
		Expires 08/13/76
Planned for System: not applicable		CATEGORY (check one)
Fixes Bug Number(s): not applicable		
Documented in MTB: not applicable		
Incompatible Change: no		
User/Operations-visible Interface Change: no		
Coded in: (<input checked="" type="checkbox"/>)PL/I (<input type="checkbox"/>)ALM (<input type="checkbox"/>)other-see below		
Performance: (<input checked="" type="checkbox"/>)better (<input type="checkbox"/>)same (<input type="checkbox"/>)worse		
DOCUMENTATION CHANGES (specify one or more)		
MPM (vol,sect)	MPAM (sect)	() Lib. Maint. Tools
MOSN (sect)	MSAM (sect)	() Sys. Anal. Tools
PLMs (AN#)		() Sys. Prog. Tools
Info Segs		() 355
Other		() BOS
None (reason)		() Salvager
		(<input checked="" type="checkbox"/>) Ring Zero
		() Ring One
		() SysDaemon/Admin
		() Runtime
		() User Command/Subr
OBJECTIONS/COMMENTS:		

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

Summary

The following bugs have been found in the syserr mechanism:

1. Messages could be logged out of order due to an incorrect sequence of locking the wired and paged logs to copy messages out.
2. Zero length messages could cause a crash when logged (on some hardware), or a crash when the log is validated at start up.
3. When equal messages are written on the console, they are padded with blanks out to the full length of the original message.

Detailed Proposal

Problem 1 will be fixed by changing the way in which the logs are locked. Currently, a syserr interrupt is handled as follows.

1. lock the wired log.
2. copy messages into wired stack
3. unlock wired log
4. lock paged log
5. copy messages in
6. unlock paged log

It is possible that while one syserr log interrupt is between steps 3 and 4, another processor could complete steps 1 thru 4 and log a message out of order. The sequence will be changed to the following.

1. lock paged log
2. lock wired log
3. copy messages into wired stack
4. unlock wired log
5. copy messages into paged log
6. unlock paged log

Multics Change Request

TITLE: Answering Service Changes for Auto Call		STATUS	DATE
AUTHOR: David M. Jordan		Written	02/26/76
		Status	A 03/09/76
		Expires	09/19/76
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.0 -Fixes Bug Number(s) -Documented in MTB 247 -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR		Category (Check One) <input type="checkbox"/> Lib. Maint. Tools <input type="checkbox"/> Sys. Anal. Tools <input type="checkbox"/> Sys. Prog. Tools <input type="checkbox"/> 355 <input type="checkbox"/> BCS <input type="checkbox"/> Salvager <input type="checkbox"/> Ring Zero <input type="checkbox"/> Ring One <input checked="" type="checkbox"/> SysDaemon/Admin. <input type="checkbox"/> Runtime <input type="checkbox"/> User Cmmd/Subr.	
		DOCUMENTATION CHANGES	
		Document	Specify One or More
		MPM (Vol, Sect.)	
		PLMS (AN #) assign a PLM	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name) SRB	
		None (Reason)	

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

SUMMARY:

Install the Answering Service portion of the Multics Auto Call Facility as described in MTB 247 and in the accompanying user and external site documentation.

To: Whom It May Concern
From: David M. Jordan
Date: February 27, 1976
Subject: The Multics Auto Call Facility

Auto call hardware and associated software allow a computer to place telephone calls to other computers or terminals over standard telephone lines. This allows a relatively inexpensive alternative to more formal networks for special purpose low speed communications.

The Multics Auto Call Facility provides the necessary support software to allow controlled and accountable use of auto call hardware by arbitrary user processes. The facility relies on the Initializer process to perform such functions as controlling access to the hardware, establishing a communications link, and allowing for resource accounting.

The remainder of this document is divided into three sections:

- User Documentation
- System Requirements
- New tty_ Control Operations

I. Use of the Multics Auto Call Facility

The Multics Auto Call Facility allows the user access to pre-dialed auto call lines. All control operations (line assignment, telephone dialing, and hangup processing) are performed by the Initializer in response to user generated Inter Process Communication requests or in response to hardware conditions. After a line has been assigned, the user may perform input and output over the line using the standard Multics I/O system `iox_`.

The remainder of this documentation describes three entry points of interest to the user:

`dial_manager_$dial_out`

to request that the Initializer assign an auto call line and place a call to a specified telephone number.

`dial_manager_$terminate_dial_out`

to request that the Initializer hang up an auto call line and deassign it from the requesting process.

`convert_dial_message_$return_io_module`

to interpret status returned by the Initializer to determine such information as channel name and line status.

Because all requests are handled through Inter Process Communications, various functions peripheral to auto call must be performed by the user process. In order to demonstrate the sequence of calls required and data structures needed, a sample program which performs the basic required functions is also provided.

Name: dial_manager_\$dial_out

This entry point is used to request that an auto call line be dialed to a given telephone number and, if the line is successfully dialed, that the line be assigned to the requesting process.

Usage

```
declare dial_manager_$dial_out entry (pointer, fixed  
    bin(35));
```

```
call dial_manager_$dial_out (addr(dial_manager_arg), code);
```

1. dial_manager_arg

is as described in Notes below. Note also that dial_manager_arg.version must be 1, and that dial_manager_arg.phone_number and dial_manager_arg.wakeup_channel must be supplied. Note also that dial_manager_arg.tty_name may be either a null string (in which case the system will attempt to assign any available auto call tty channel) or a specific tty channel (in which case the system will attempt to assign that channel only). (Input)

2. code

is a standard status code. (See Notes below.) (Output)

Name: dial_manager_\$terminate_dial_out

This entry point is used to request that the Initializer hang up an auto call line and unassign it from the requesting process.

Usage

```
declare dial_manager_$terminate_dial_out entry (pointer,  
        fixed bin(35));
```

```
call dial_manager_$terminate_dial_out  
    (addr(dial_manager_arg), code);
```

1. dial_manager_arg

is as described in Notes below. Note also that dial_manager_arg.version must be 1, that dial_manager_arg.tty_name must be supplied, and that dial_manager_arg.wakeup_channel must be supplied and have the same value as was used when the original dial_manager_\$dial_out was made.
(Input)

2. code

is a standard status code. (See Notes below.)
(Output)

Name: convert_dial_message_\$return_io_module

This entry point is used to process Inter Process Communication messages from the Initializer regarding the status of an auto call line. In addition to returning line status, this entry also returns the device name and IO module name for use in attaching the line through the iox_ I/O system.

Usage

```
declare convert_dial_message_$return_io_module entry (fixed
    bin(71), char(*), char(*), fixed bin, 1 aligned like
    status_flags, fixed bin(35));
```

```
call convert_dial_message_$return_io_module (Info.message,
    dev_name, dim_name, n_dev, status_flags, code);
```

1. Info.message

is the message portion of the structure filled in by ipc_\$block or supplied as an argument to an IPC event call procedure. For more information see the description of ipc_ in the MPM Subsystem Writers' Guide, AK92. (Input)

2. dev_name

is the name of the tty device assigned or null if no device was assigned. (Output)

3. dim_name

is the name of the iox_ IO module to be used with the assigned device. (Output)

4. n_dev

is the number of tty devices currently assigned to this process. (Output)

5. status_flags

is a structure with the following declaration:

```
declare 1 status_flags aligned,
    2 dialed_up bit(1) unaligned,
    2 hung_up bit(1) unaligned,
    2 control bit(1) unaligned,
    2 pad bit(33) unaligned;
```

The "dialed_up" bit will be on if the line is currently dialed up. The "hung_up" bit will be on if the line has hung up. The "control" bit is used only in reference to calls to dial_manager_ entries other than dial_out and terminate_dial_out. (Output)

6. code

is a standard status code. (See Notes below.)
(Output)

Notes

1. dial_manager_arg

dial_manager_arg is a structure used to pass a variety of information about an auto call request to the procedure dial_manager_. It has the following declaration:

```
declare 1 dial_manager_arg internal static aligned,  
        2 version fixed bin initial (1),  
        2 phone_number char(22),  
        2 wakeup_channel fixed bin(71),  
        2 tty_name char(32);
```

1. version

indicates the version of the structure that is being used. Currently this must be 1.

2. phone_number

for calls to dial_manager_\$dial_out, this is the telephone number to be called. Note that non-numeric characters will be ignored, so the user need not remove them from a telephone number string.

3. wakeup_channel

is an InterProcess Communication channel to be used in communications from the Initializer. Note that the channel should be the same for all calls used in reference to the same auto call session.

4. tty_name

is used for calls to dial_manager_\$terminate_dial_out to indicate which channel should be disconnected. In calls to dial_manager_\$dial_out, it must be either a null string (in which case the Initializer will attempt to assign any available auto call channel) or a specific tty channel to be used for the auto call attempt.

2. Status codes

dial_manager_\$dial_out and dial_manager_\$terminate_dial_out may return any of the following status codes:

1. error_table_\$bad_arg, if dial_manager_arg.version \neq 1.
2. Any codes returned by message_segment_\$add_file, if dial_manager_ was unable to place the details of a request into the message segment provided by the Initializer.
3. Any codes returned by ipc_\$block.
4. Any codes returned by convert_dial_message_.
5. error_table_\$action_not_performed for any other errors.

convert_dial_manager_\$return_io_module will return error_table_\$action_not_performed if any error is encountered.

In order to help the user understand the sequencing of calls required for use of the Auto Call Facility a sample program has been provided on the next few pages. The main entry point "call_out" accepts as arguments a telephone number and stream name. It calls dial_manager_\$dial_out to request that a call be placed to the specified number, and, if the attempt is successful, it then attaches the assigned line through the appropriate lox_ IO module.

The second entry point "call_out\$end_call_out" performs the functions required for a user to terminate an auto call session. In addition to requesting that the Initializer hang up the line, it also makes the appropriate lox_ calls to close and detach the stream.

The last entry point "call_out\$hangup" is designed to act as an IPC event call procedure to be invoked if the line hangs up for reasons other than a user request. It also performs the various functions required to clean up the lox_ attachment.

Note that this procedure is designed to handle only one auto call connection at a time and thus has various flags in internal static to insure that it will not be re-entered while a previous auto call session is still in progress.